SESSION

DEGREE PROGRAMS, CURRICULUM, AND COURSE DEVELOPMENT + ASSESSMENT AND RELATED ISSUES

Chair(s)

TBA

Curriculum Design, Development and Assessment for Computer Science and Similar Disciplines

I. Schagaev[1], E. Bacon[2], N. Folic[1] and N. Ioannides[1,*]
[1]Faculty of Computing, London Metropolitan University, 166-220 Holloway Road, London, N7 8DB, UK.
[2]School of Computing and Mathematical Sciences, University of Greenwich, Maritime Greenwich Campus, Old Royal Naval College, Park Row, Greenwich, London, SE10 9LS, UK.

Abstract - *Curriculum design and development for Computer Science and similar disciplines as a formal model is introduced and analysed. Functions of education process as knowledge delivery and assessment are analysed. Structural formation of curriculum design is presented using definitive, characteristic and predictive functions. The process of change in the discipline is described and analysed. The algorithm to determine the core of the discipline is developed. Functions of the core moving and merging are introduced and analysed. A new assessment technique, multiple choice answers approach (MCAA) with several right and not mutually exclusive answers from a larger set of answers, is introduced. MCAA, even though developed as an assessment technique, also facilitates the formation of reliable coverage of subject curricula. The rigorous formation of assessment becomes an essential segment of curriculum design and development. Levels of applicability are explained showing the practical effectiveness of the proposed approach. A trial application in a real module of computer science showed promising results. Schemes of student progress assessment are introduced including analysis of success and protection from guessing, using penalty functions. Schemes of further development and directions for further research are discussed*.

Keywords: Curriculum Design, Curriculum Development, Curriculum Core, Theory of Classification, Formalisation, Multiple Choice Answer Approach, Penalty Function.

1 Introduction

Computer Science is a relatively new and fast growing discipline for teaching. It absorbs different theoretical and technical results from different disciplines and creates a fusion which penetrates and influences many aspects of human life. Computer Science is, in fact, the theoretical base for the fastest ever growing area of technological development, that of Information Technology (IT). At first glance, Computer Science as a discipline should absorb its own technical applications (IT), but previous attempts to do this failed. It is, therefore, necessary to create a 'bridge over troubled water' and make *curriculum design*, *development* and *assessment* more connected and, where possible, rigorously designed.

The need for such an effort evidently grows day by day, as one of the fathers of Computer Science, Dijkstra, in his letter [1] to the Communications of the ACM, admits: "*I would therefore like to posit that computing's central challenge, viz. "How not to make a mess of it," has not been met. … You see, while we all know that unmastered complexity is at the root of the misery, we do not know what degree of simplicity can be obtained, nor to what extent the intrinsic complexity of the whole design has to show up in the interfaces. … To put it bluntly, we simply do not know yet what we should be talking about…*"

This work attempts to form a logical core of curriculum design for computer science and similar disciplines. Initially, the main terms used are defined and the role of education as a very important driving force in the improvement of life for the society at large is identified. Existing models of curriculum development and their drawbacks are briefly discussed. Curriculum design as information processing is then analysed and further developed by the introduction of three main functions in discipline construction: definitive, characteristic and predictive. The core of the discipline, the way of its selection and its main features: moving and merging, are analysed and discussed. The authors also focus on the issue of assessment and address this as part of curriculum development and a new type of assessment, *Multiple Choice Answers Approach* (MCAA) is introduced, and is supported by a scheme for its implementation as well as suggestions for a penalty function. Our goal is to create a new assessment methodology, which, apart from assessing the students, also enables us to estimate the quality of teaching delivery. The creation of an automatic (or semi-automatic) assessment procedure along with the development of an algorithm for the selection of the most important material for assessment will help in estimating the quality of teaching delivery. A result from such an assessment for a concrete module from Computer Science is presented. Finally, further work and open problems in the further development of our approach are outlined.

By considering the following facts about Computer Science and other similar disciplines, one realizes that the need for this work becomes multi-fold:

- Computer science in general suffers from a snowball of information, useful and otherwise [1], which cannot be handled, properly processed or justified for teaching purposes.

- Standard analysis of the learning outcomes mostly serves to indicate the general performance through the use of statistics and has very limited module or student meaning.
- The level of competency of students varies enormously.
- Even though social demand to help students with different backgrounds get high levels of education is present, the resources to match these demands are limited.

If the strategic aim of this work is to be achieved, Computer Science as a discipline for teaching must be modified to incorporate IT, tuned for purpose, thus resulting in a more efficient and effective teaching process (those who ruined us, help us).

2 Science, knowledge, skills, curriculum: definitions and classifications

The main terms used in this study are general and have a variety of meanings which depend on the human activity that they may be used for. It is, therefore, necessary to define the specific meaning of these terms for the work presented here. Complete description of each term can be found in [2].

- *Science* - 1. The study, description, experimental investigation and theoretical explanation of the nature and behaviour of phenomena in the physical and natural world; 2. Branch of systematized knowledge of study.
- *Knowledge* - 1. Information, understanding acquired through learning or experience; 2. The total body of known facts or those associated with a particular subject; 3. Justified or verifiable belief, as distinct from opinion *(Phil)*.
- *Skills* - Special abilities in particular field acquired by learning or practice.
- *Curriculum* - The courses offered by an educational institution or followed by an individual or group; *Latin* - running, course, course of study, programme.
- *Computer Science* - Study of the construction, operation, and use of computers.

A more holistic approach to the word *curriculum* assumes that it should be placed between the *aim of education* and the *learning outcome*, where the *aim* is “what we want to achieve” and the *learning outcome* is “what we are able to measure”. A major question raised here about the *learning outcome*, as this assumes to express the result of education in one sentence, is whether we can actually do this!? This term will not be used here and its applicability is out of the scope of this work.

Curriculum design was analysed by Aristotle: *“For the formal nature is of greater importance than the material nature”* [3] and Confucius: *“He who learns but does not think is lost; He who thinks but does not learn is in great danger”* [4], clearly identifying the necessity of reflecting on what one has learned. This work aims to build an algorithm of *Curriculum design* and *development* for *Computer Science* and similar disciplines using our own recent theoretical results and experience during the redevelopment of an existing module within the Faculty of Computing of the London Metropolitan University.

The terminology shows that *Science* differs from *knowledge* by the indirect introduction of *the Subject* (an agent) to receive *knowledge.* In spite of its well-deserved recognition, the Penguin Dictionary has, in our opinion, a serious mistake in the definition of the word *knowledge* by using the too general and absolutely not essential in this context term *information*. Long discussions on the interrelations of this term are presented in the number of books written by N. Wirth, E. Dijkstra, W. Turski and others. Our opinion about the relation between *knowledge* and *information* is shown below and the efficiency of this description is further elaborated below:

$$Knowledge = Information + Algorithm\ of\ its\ Application \quad (1)$$

Curriculum, in turn, is just another name for the program of study. Just like any other program it must be complete, efficient and reliable in order to provide *education* in the selected area or discipline. The education cycle is completed when its outcome has been returned back to the society (figure 1). The roles of the main agents (student and teacher), are analysed here in the framework of the education process.

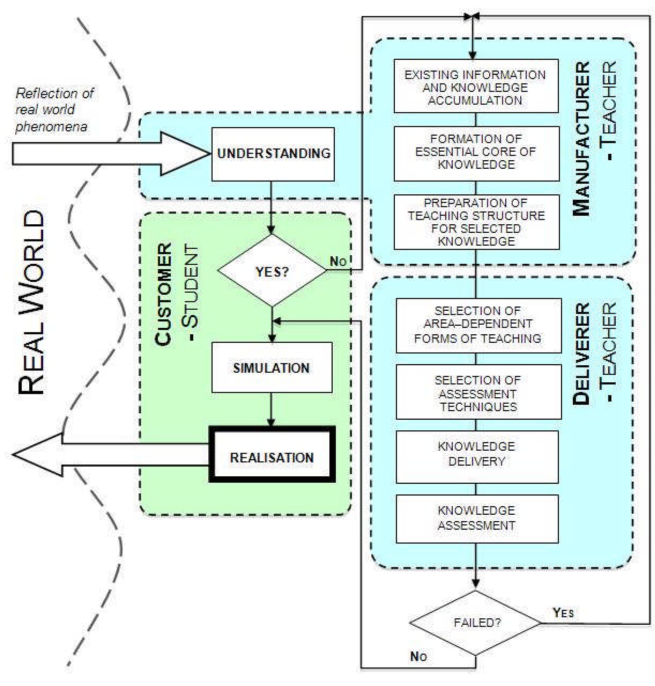


Fig. 1. ‘Return the Results of Education to Society’ Algorithm

As shown in Figure 1, feedback is the main measure of teaching success. During teaching the lecturer is the main *deliverer* of the products: *knowledge* and *skills*. The main *consumer* or *customer* is the student. The cycle of education is completed when *a customer* returns his/her results back to the society and, therefore, becomes *a deliverer*.

Confucius indirectly confirms the usefulness of this kind of modelling by presenting a measure of success for education by the production of better men after education: “*a goal is to create gentlemen who carry themselves with grace, speak correctly, and demonstrate the integrity of things*”. The model presented here goes a step further with the teacher becoming the main *manufacturer* and *deliverer* of *knowledge*, while the student changes their role from *consumer* during the learning process to a *deliverer* after it, when s/he returns back to society new experiences from the results of education, while Confucius claims that teacher is "a transmitter and not maker".

Any discipline of Computer Science assumes a certain amount of *knowledge* and *skills* in some proportion which plays a very important role in discipline design, and therefore, curriculum development. *Skills* are different from *knowledge* in the definitions presented above as they concern a special ability, not a wide understanding of the area. A good argument in favour of this differentiation is the variation in the area of application – *skills* are concerned with the application of particular *knowledge* or experience in a specific, well known environment, whereas *knowledge* is about the application of experience and learning outcomes in an uncertain and wide environment. This separation is important in further discipline structure and its curriculum development.

The equation *K+S = Successful Application* presents *Knowledge (K)* and *Skills (S)* as essential components of education. Rigorous separation of *K* and *S* helps to balance discipline structure by means of using the most appropriate instruments of the teaching/learning circle: lectures, tutorials, practicals, courseworks, as well as various assessment techniques. Three main but different types of segments (modules) of Computer Science as discipline are presented in Figure 2:

Fig. 2. Ratio of *knowledge* and *skills* in computer science disciplines; the blue (darker) segment presents *knowledge*.

Case A refers to areas within the discipline of Computer Science such as Algorithm Design and Problem Solving, Software Engineering, Network Technologies, Theory of Programming, Discrete Math, Theory of Data Bases, etc. Case B refers to areas such as Human Computer Interaction, Computer Aided Design, Computer Graphics, etc. Case C refers to areas such as Web Design, HTML, Java, Visual Basic, Modula-2, Pascal, Applications of Data Bases, etc.

In disciplines where $K > S$ the discipline must be built with a wide area of knowledge, be more abstract and, where possible, general in order to show the limits of the existing knowledge, as well as its place in the context of science. Areas of discipline which fall within this scenario require much more active lectures and seminars in the pure academic meaning of these words.

In disciplines where $K < S$ there should be more concern for deeper and practical aspects of one narrow area of skills and the applicability of these skills. Areas of discipline which fall within this scenario require many more practical sessions with small introductions of elementary or essential theory.

There is no doubt that the *curriculum* for these different cases should have different structure and forms.

The invention of measurement and efficiency of teaching and progress of learning by the use of the so-called Learning Outcomes (LO) caused both serious criticism and scepticism amongst practitioners. Indeed, when considering that 11 from 15 modules in Computer Science have nearly identical LOs and with all others being pretty much similar the immediate question which arises is: what do these LOs actually mean? Neither the structure of the discipline, nor its context and success of understanding can be described in two or three sentences.

It is clear from the above that new research is required in the area of structuring of knowledge delivery and derivation of schemes in order to measure the result of discipline delivery. The research work presented here focuses on schemes of knowledge delivery and assessing and in this context *Curriculum Development.*

3 Modelling of educational program development

Three very complex entities are involved in the process of education: *Science*, *Science Deliverer* (Lecturer) and *Science Consumer* (Student). Various authors [5][6][7] described the main elements of the course design process, their interrelation in time and their logical order. The most cited author [6] presents the generalized model shown in Figure 3.

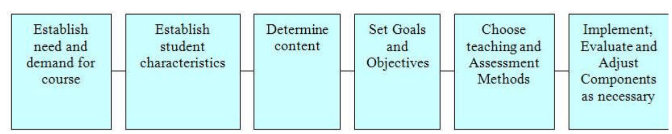


Fig. 3. A typical model of the course design process [6]

The model shown in Figure 3 suffers from poor logic: student characteristics are placed before content determination; goals and objectives, which are much more general terms and wider in meaning, are placed after determination of the context; teaching and assessment methods are selected based on goals and objectives; implementation and evaluation requires "adjustment" as necessary, but with no indication as to the meaning of the phrase "as necessary". More logical might be the sequence presented in Figure 4.

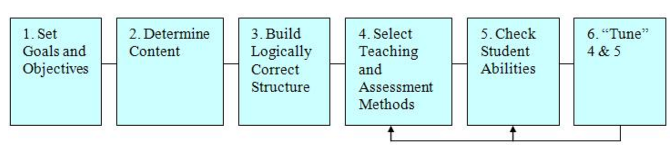


Fig. 4. Modified model of course design process.

But even this "logicalisation" (figure 4) cannot change the comments on the applicability, usefulness and efficiency and their measurement in the course design process. Both models shown in Figures 3 and 4 are too general to be useful. Furthermore, the resource dependence of the discipline and possible ways of its delivery are not even mentioned in the models of these authors [5][6][7].

We must also add that the context of the discipline should be involved, embedded and reflected in and during the *curriculum design process*. These arguments make obvious the need to develop a new model of the discipline and, therefore, curriculum design. The order is important: at first, one has to develop *a discipline*, and *then* the way of delivery. Below is such a sequence of discipline development:

1) Determination and definitions of the main elements in the discipline:
   - Description of connections between definitions.
   - Selection of main features of definitions.
2) Analysis of the schemes for discipline delivery (either existing, or new).
3) Specification of discipline delivery main elements:
   - Course structure.
   - Assessment instruments.
   - Performance issues and scheme of its measurement.

Work published by [8][9][10] concern a development of the *Theory of Information Processing,* described at the level of the main categories and resources. Here, the *input information* goes through an algorithm (processing), *new information* is created on the way and *the result* is produced and delivered. The process *of Learning* can also be analysed as *Information Processing* and vice versa: *The Processes of Teaching and Learning are in fact one entity – Information Processing.*

4 Curriculum Development as Information Processing

By accepting this point of view we could apply terms and results from the *theory of information processing* for various aspects and areas of pedagogy – in this case to the processes of discipline and curriculum development. This means that principles of classification for information systems can help build a course of learning with the highest possible understanding as well as a measure of this understanding.

The next important issue relates to the analysis of all possible resources necessary to form a *discipline* and its *curriculum*. Rigorous classification of resources and ways of their use and processing should be built to form the framework for all further steps.

The delivery of *knowledge* is a process of transforming a student from a state without *knowledge* to another state, where the *knowledge* obtained can be assessed and confirmed. The process of assessment can differentiate those students with the required level of *knowledge* and those without. A well-known sequence of steps to deliver knowledge includes:

1) *Knowledge* delivery.
2) Delivery of practical (application of knowledge).
3) An assessment.

All steps in this sequence should be completed at the right time and in such a manner that the system (University) should notice neither a fault nor the process of its elimination. Students are involved in all these steps. Some may not understand some elements and/or may not be able to "process" information when delivered in the course, but can catch up during the course. The passing of an assessment will then confirm a student as successful in acquiring the required level of knowledge. In some cases, however, expected failure, lack of confidence or change of interest may cause a student to change his/her course. Such cases will not be considered here due to the uncertainty and existing elements of subjectivity that such cases entail.

In reality, many more steps (in the *Knowledge Delivery Algorithm*) are needed to eliminate malfunctions and avoid student failures or even their withdrawal from a course. These actions depend on the various functions and features of learning and the teaching processes, their roles and "power" to provide knowledge for the students.

A *Knowledge Delivery System* can be analyzed as an information processing system using first order philosophical categories such as *matter* and *time*. For *Knowledge Delivery* it is *structure* and *time*. More importantly, the main function of *knowledge* (information) delivery must be considered at the first level of description, together with *matter* (structure) and *time* (see figure 5).

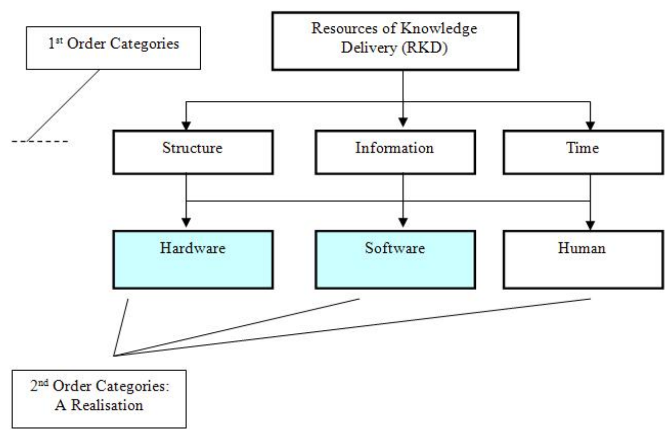


Fig. 5. Classification of resources to deliver *knowledge*

The levels shown in Figure 5 describe the structural core of the system to deliver *knowledge*. While *first order* categories present reflections of our understanding (they exist in the mind only), the *second order* descriptions (hardware, software) present real world objects as *executive elements* for the first order categories.

In general, the above mentioned sequence of steps towards delivering knowledge should generally be completed in scheduled time and in such a manner that the 'average' student could cope with the speed of its delivery as well as be able to continue learning. Furthermore, attention should be paid to the following sensible comments:

1) Time spent on exams and tests, in fact, is excluded from learning.
2) This sequence does not match IT use.

3) There is no direct evidence that lectures delivered were transformed into knowledge until the assessment was completed and passed.

The *first* statement suggests that the time of learning and the time of assessment (tests, exams) are separated for the student and is usually wrong to expect that under pressure of a possible failure the student continues to learn during the assessment.

The *second* statement suggests that knowledge delivery uses IT in one-way, and assessment uses IT in another. Knowledge delivery primarily concerns presentation aspects, while the use of IT in assessment is often based on multiple choice questions.

The *third* statement suggests that even the best developed and presented lectures can be unclear for students with different background. It becomes visible only during exams, and by then it is too late.

What options and resources do we have when creating a discipline and curriculum suitable for efficient teaching and learning? Again, according to [8][9][10] these are *structure*, *time* and *information*. At the same time, it is clear that testing time (assessment) should be eliminated from the algorithm of knowledge delivery (see *first* statement above) even though without assessment it will be impossible to receive feedback on the quality of education, either at the macro level – from the society at large (see Fig.1), or from the university. To resolve this issue well-known hardware design techniques of self-checking developed by [11] could help and should be used. In our case the most important feature of self-checking is *concurrency*, or *near*-concurrency, with functioning and *self-checking* even though the term *self-checking* in pedagogy has a one-to-one synonym: *self-assessment*. The difference lies in the quality of self-assessment which is very weak in comparison with self-checking as described by the Boolean logic function. As far as resources for *knowledge delivery* are concerned *time* cannot be counted, as it is beyond our control! Figure 5 can therefore be transformed as shown in Figure 6.

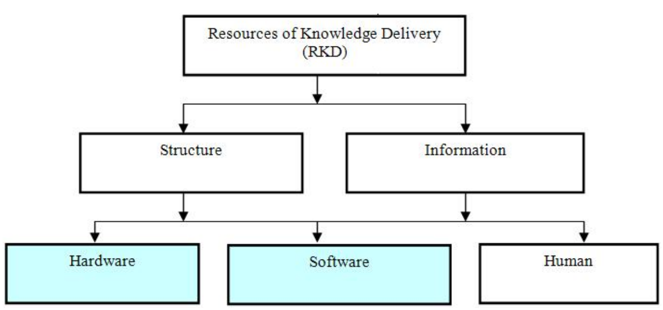


Fig. 6. Classification of resources to deliver *knowledge* excluding *time*

The two blue (shaded) boxes above present Information Technology elements, which can be involved in the process of *knowledge* delivery. We believe that further progress in the theory of *Knowledge Delivery* should be found in the selection of the principles and also in finding ways to load and reload it on the IT resources.

5 Principles of curriculum design

By direct analogy with [8][9][10] the *Knowledge Delivery System* must realize three closely connected functions: *Definitive*, *Characteristic* and *Predictive*.

A *Knowledge Area* has to have a *Definitive Function (DF)*, in which terms and concepts are called and nominated. DF answers the question "*What is it?*". The second function describes the interrelations between *Definitions* and c*haracterises* them and is called the *Characteristic Function (CF)*. *CF* answers the question "*how are these definitions connected?*". The *Predictive Function (PF)*, in turn, answers the question "*What if?*". Knowledge of DF and CF and their application are essential elements here. Application of definitions and characterisation of their interrelations enables the *use* of elements that have just been learned and predicts their behaviour.

But what is the core for the successful organisation of a learning process for one discipline and what is it for another? According to [8][9][10] the success of learning is the formation of a strong *PF* in the selected area of *Knowledge - discipline* and holds true for both new and well established disciplines. The success of *PF* depends on concurrent observation and satisfaction of two conditions:

- The first condition is accuracy, precision in the selection of the aim pursued, with analysis of a new or existing subject domain and the key required feature - kind of *aim integrity*.
- The second condition, not less important, is the internal structure of the course: rigorous approach in the construction and introduction of terms and concepts inside the course, as well as its assessment procedures.

To summarise, the requirements on the formation of a new curriculum note:

1) Any discipline, as an introduction of a theory, should be considered from the point of view of performance of three interconnected functions: *Definitive*, *Characteristic* and *Predictive*.

2) The discipline must be constructed with the strict principle of aim integrity, i.e. with the selection of the single feature (predictive function) necessary to achieve and maximise.

3) Each phase of introduction, presentation and detailed analysis of a discipline development must be rigorously analysed. Otherwise, the success of the course built around this discipline will be problematic.

4) Only really essential features and details of the analyzed discipline objects and phenomena must be included. For example, the course of computer science as information processing system should be discussed in terms based or directly connected with information; a course describing distributed systems and networks should present in some way categories of dimensions directly connected to these kinds of systems, as well as basic relative terms.

The applicability of the three functions mentioned above is the key to successful discipline and curriculum development. Essentially, the resources available should be analyzed, including any new technologies such as information processing. IT involvement in the process of course construction should be considered from the beginning. Here, we try to analyse the process of knowledge delivery from the point of view of three interrelated functions: DF, CF, PF and algorithm of learning, using the approach from [8][9][10].

Consider the sequence of functions discussed above:

$$\mathbf{DF \rightarrow CF \rightarrow PF} \tag{2}$$

At first one delivers definitions (DF) and then their interrelations (CF). The algorithm of learning completes when a learner has acquired the skill in the subject to predict the behaviour of the elements presented in the course, and PF exists. The power of prediction (PF) means understanding of the course material by the student. The bigger an individual PF is the better the student knowledge is. PF should be formally measured. Growth of the knowledge can be described as evolution and transition from nowhere to PF:

$$\mathbf{Nowhere\ and\ Nothing \rightarrow DF \rightarrow CF \rightarrow PF} \tag{3}$$

This evolution can be analysed and terminated when $PF_s > PF_{tr}$ (indices s: student and tr: threshold). Threshold indicates the required level of understanding, proven by assessment through analysis of predictive functions developed in the student's mind.

If a full set of definitions exists in a discipline it can count towards some development of DF. The question how to build the DF of the subject has its own importance and requires special research and elements essential for this paper will be introduced further.

Suppose for now that a teacher has prepared the DF of the course, or its part, and is ready to deliver it (*Nowhere and Nothing* → DF). The first iteration of learning will then consist of the delivery of DF from teacher to student. Once the student absorbs the essential definitions required for the understanding of the lecture material information from the course segment DF is completed and s/he is ready to learn further. Connections of elements – terms in the subject denote (determinate) CF: DF → CF.

In math CF is an essential set of formulae, in language course – grammar, etc. If we can deliver to a student an explanation on how terms of the course are connected then we can assume that CF does exist. Then the process of learning goes to another phase:

$$\mathbf{DF \rightarrow CF \rightarrow PF} \tag{4}$$

In a discipline, as a whole, it is impossible to expect that DF has to be learned as a whole before a student starts to understand CF also as a whole. Therefore, there must be a way to break DF and CF down properly into logical pieces.

To build well-balanced "golden" shape segments of DF one has to build as correct sequence of several DF_i in a curriculum as a whole:

$$\mathbf{DF = \{DF1,..DFi,..DFn, DFu\},\quad i = 1,n} \tag{5}$$

During the learning process a sequence of DF_i for i = [1,n] should be accompanied by a proper sequence of $\{CF_i\}$ with some shift in time, as presented in Figure 7. Any overlap in time with the delivery of DF_i and CF_i is acceptable and even useful. The length of overlap should be selected regarding the level and ability of students. Also, the size of a particular DF_i might be determined by a student's ability for learning. For good students any particular F_i might be bigger than for weaker students. This is especially important due to some recent political initiatives such as that for 'universities for all'.

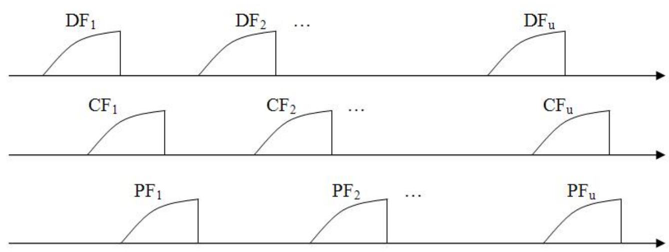


Fig. 7. Sequence of knowledge delivery

We denote DF_u and CF_u as universal (complete) functions for the discipline. Having DF_i and CF_i provides a possibility *to build* in the discipline or its part a predictive function: PF_i; for the whole discipline, having DF_u and CF_u enables to build PF_u for the discipline in the student's mind. Neighbourhood DF_i and DF_{i+1} might not be intersected (for the poor organised curriculum), but for the discipline as a whole a unification of all elementary definition (DF_i) takes place:

$$\mathbf{DF_1 \ldots \cup \ldots\ DF_i\ \ldots \cup \ldots DF_m\ \ldots \cup \ldots DF_n = DF_u} \tag{6}$$

The form of the curve for DF_i, CF_i and PF_i could also be different as well as the length of their intervals and would be defined by the features of the discipline and the personal preferences of the deliverer. It is not clear as yet what defines the form of curve and the length of the triangles except of the student and teacher abilities. It is, however, clear that the total workload of delivery of DF_i is a sum of workloads to deliver all elements as shown in the equation below:

$$W(DF_i) = \int_{t_i}^{t_{i+1}} \frac{dDF_i}{dt} \tag{7}$$

where: t_i and t_{i+1} define the interval of delivery for DF_i.

In a similar way, the total workloads to deliver CF and PF are shown below:

$$W(CF_i) = \int_{t_i}^{t_{i+1}} \frac{dCF_i}{dt} \tag{8}$$

$$W(PF_i) = \int_{t_i}^{t_{i+1}} \frac{dPF_i}{dt} \tag{9}$$

Naturally, discipline delivery should be done smoothly, with well-balanced functions of DF_i , CF_i, and PF_i as mentioned above. But proper management of their delivery seems to be a subject of special research, and includes human computer interaction aspects and specific features of the learning subject.

6 Curriculum Core

To build the *core* of a curriculum on a subject a set of Definitive Functions {DF} and one of Characteristic Functions {CF} must be constructed. The {DF} should contain the set of terms, in the same way as the Glossary of a book does, and the {CF} should contain the set of topics describing how definitions are connected, in the same way as the Index of a book does. By combining terms from the Glossary with the Index we have a semi-automatic procedure to form a set of essential questions required for *Assessment*. Up to now, except for pure math disciplines, the proper formation of a set of essential questions was hardly known. There is no doubt that statistics should be obtained to analyse the relevance of these questions to the core of the curriculum. Additionally, a logical sequence of the questions should be arranged for the *assessment*, as well as for the process of delivery of the discipline. It is clear that even a semi-automatic procedure might be quite useful, as it reduces teacher's workload in formation of assessment and, at the same time, guarantees quality. In one extreme there is a fully automated option in formation of *assessment* by direct programming of a set of terms and a set of questions. But this approach requires further research and approbation. In another extreme, a "manual" application of this algorithm should be taken.

If the *core* of the subject does exist, it does not mean that questions and terms taken from one book or lecture notes do relate to the core. Personal preferences of a teacher can be so high that they may completely mislead students and should be avoided. Thus, several books on the same subject from different authors could be selected and a *joint* Glossary $\{DF_j\}$ of terms, which includes terms that belong to at least one glossary from the analysed books and lecture notes, could be organised thus creating an *essential* Glossary $\{DF_e\}$ of terms which includes those terms only common to all Glossaries. Formally these two sets of glossaries are described as follows:

$$\mathbf{DF_j = DF_1 \cup DF_2 \dots \cup DF_{n-1} \cup DF_n} \quad (10)$$

$$\mathbf{DF_e = DF_1 \cap DF_2 \dots \cap DF_{n-1} \cap DF_n} \quad (11)$$

Building a set of questions around an essential glossary DF_e forms an <u>*essential*</u> set of questions CF_e and, in fact, complete as a whole a formal preparation of assessment. DF_e and CF_e combined form a core of the discipline assessment.

The core of the curriculum and its features present us with a special interest. Above we discussed in brief some elements of the algorithm *how to define a core*. Here we discuss features of the *observation* and *behaviour* of the core. We discover two phenomena in behaviour of the core – *moving core* and *merging core*.

A *moving core* occurs when curriculum design (and re-design, as this is a permanent process due to the appearance of new books, new papers and other sources of information) shows growth when using some descriptive elements (DF_i) and decrease of use for some others. This change of intensity in the use of different areas of the core is shown in Figure 8 by a different intensity of blue colour.

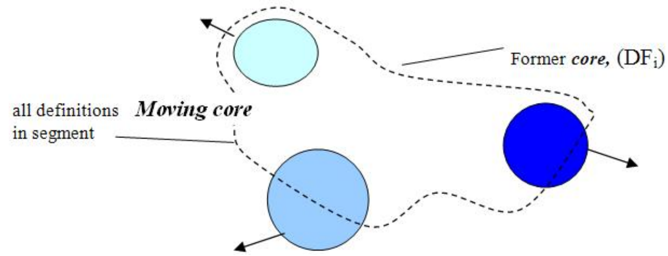


Fig. 8. Moving core of curriculum

This process of movement can lead to the separation of a discipline into several *new* disciplines. This separation should be detected early before it happens, as the discipline can loose its predictive function (PF) and become obsolete – theory can become very advanced to be practical (pure math case) and vice versa – practice can be developed much faster than theory and thus theory becomes almost useless, having only a descriptive function (networking). In the first example, knowledge does not connect with skills and is therefore hardly applicable to the society at large; in the second, the technological development has big advances in comparison with theoretical adaptation of the technological results. Both cases are very inefficient in terms of the model about the role of education presented in Figure 1. Keeping in mind this model, education should be effective from the point of view of improvement to the quality of life, and thus the early detection of the process of *moving* core seems to be important.

On the other hand, the *merging of several cores* can be caused by serious inventions, practical demand or revolution in technology, which accelerate the involvement of different disciplines into fusion as shown in Figure 9 where Computer Science is used as an example. The main reason of this fusion is the growth of Predictive Function resulting in the applicability of a new discipline to society.

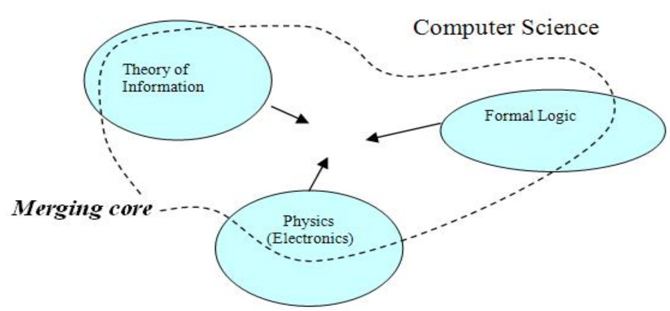


Fig. 9. Merging of several cores

Such processes and their detection are pretty complex but understandable and their algorithmisation and programming do not look impossible. This is a perfect area for the use of IT to assist the teacher to assemble the correct elements for the core of the discipline.

In the long run, these two processes can help detect changes in scientific areas related to the discipline automatically and assist the teacher to almost automatically adjust the curriculum and assessment to these changes. There is hope here that artificial intelligence (AI) and IT can be involved much more in the formation, correction and adjustment of curriculum for any scientific or technical discipline.

7 Assessment

Methods with which an assessment using the model of curriculum design and development proposed [12][13] can be formed are discussed in this section. By considering the requirements for assessment, a nearly formal assessment procedure could be realized that is also free from existing drawbacks. This procedure is called Multiple Choice Answers Approach (MCAA) and should *not* be confused with Multiple Choice Questions (MCQ) as the two are very different. A process for the formation of a questionnaire and a scheme to maximize the efficiency of assessment, thus increasing the objectivity of the curriculum, are also introduced.

7.1 Assessment requirements

We consider the most important requirements and features of assessment to be:

- Objectivity.
- Quantitative analysis.
- Time (and other resources) efficiency.
- Concurrency with the learning process.

The popularity of assessing using MCQ, especially during the last decade, driven by the help that it provided teachers with the complex and boring procedure of marking and calculation of assessment results, has been growing very fast. And even though MCQ is a relatively new invention, which enables the applicability of IT to be used as a tool for assessment, it was considered a panacea. However, even though successful in that, the MCQ approach, disappointedly, does not provide proof of efficiency and growth of knowledge. The fact is that it does not have a connection with real delivery of *knowledge* or a measure of the quality of knowledge delivery – an *assessment* of its result. The standard assessment used to be limited by definitive descriptions, i.e. without the use of the set of functions, *characteristic* {CF} and *predictive* {PF}, as described in [9][10].

According to [14], Bertrand Russell claims that: *"to be directly acquainted with something is to be in a position to give it a name in the strict logical sense, and to know something only by description is to know only that something uniquely fits the description"*. But to give something a name and to be able to understand and use that name correctly are two different things …

7.2 Formation of the Multiple Choice Answers Approach (MCAA) assessment

The *Multiple Choice Answers Approach* (MCAA) that enables the effective use of IT for assessment and matches the requirements set in section 7.1, is presented here. Compared with its predecessor, MCQ, the MCAA, which is based on the process of curriculum design and development presented in [12][13], enables the application of IT for assessment, increases the reliability of knowledge, eliminates guessing, and can help manage the assessment of thousands of students at the same time! Assume that the discipline in question, Computer Science or other similar, has sets of definitions (Definitive Function, DF), a set of key characteristics about how definitions are connected (Characteristic Function, CF) and, therefore, a set of known predictions (Predictive Function, PF) and that all elements of these sets are known. A question $\mathbf{q_i}$ can then be created which covers (includes) several terms from DF_i ($DF_i \subset DF$, segment of the discipline) and $q_i \in CF_i$ ($CF_i \subset CF$, questions on the segment of the discipline DF_i) as shown in Figure 10.

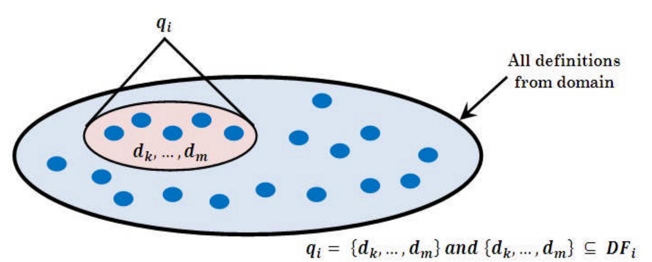


Fig. 10. Formation of a question using {DF} and {CF}

Suppose we organise a table from the elements of {DF} called *Working Table*, $WTDF_u$. Inside $WTDF_u$ we have overlapped areas in $DF_1, \ldots, DF_u$ ($DF_i \cap DF_j \neq \varnothing$, $i \neq j$) and where some definitions might belong to the various questions, and be "more important" than others:

$$\exists\, d_k \in DF_i : d_k \in DF_j, i \neq j \qquad (12)$$

By organising sub-tables $WTDF_i$ for each DF_i as an elementary segment of knowledge delivery – could be a lecture, seminar, etc – we can create a question $\mathbf{q_i}$ with explanations for each term $d_k, \ldots, d_m$ from DF_i as shown in Figure 11. All elements from DF_i should be covered by at least one question from CF_i. We can then form subsets of questions $\{q_i, \ldots, q_t\}$ to construct the full set $Q_i = \{q_1, \ldots, q_i\}$ where Q_i covers the whole area of definitions from DF_i. Very importantly, the same terms from $\{d_k, \ldots, d_m\}$ can be involved in the answers of different questions.

DFᵢ (elements of definitive function)

CFᵢ

	d_1	d_2	d_3	….	d_m	….	d_{k-1}	d_k
q_1	x		x		x			
q_2		x		x			x	
q_3	x			x		x	x	
…								
…		x	x					x
q_{m-1}								
q_m	x	x	x		x			x

Fig. 11. Connecting questions and definitions

Once table WTDF$_i$ is developed, the number of definitions required for the correct answer of question $\mathbf{q_i}$ becomes available. Furthermore, the table will also allow us to create more than one question with multiple right definitions from the same domain (DF$_i$), if necessary. All of them (say **m**) could be used together with all other possible definitions' domains relating to DF$_i$. The correct answer will then mean 100% hit of right **m** from **n**. In fact, the full set of questions for one fragment of the course will form CF$_i$. Using other definitions from the same DF$_i$ for the construction of different questions will actually enforce the student learning during assessment or the trial assessment exercise.

Formally, the procedure to create a question set is completed when the resultant questions cover all elements from DF$_i$. But this condition is not unique. We will, later on, show that there are other conditions with their own merits in forming a question set around DF$_i$.

The level of overlapping between working tables WTDF$_i$ developed for a module assessment will indicate how directly connected with the structure of the curriculum they actually are. The formation of a question set, however, has both objective and subjective aspects: coverage of the module and the students' abilities and psychology respectively. Thus overlapping of WTDF itself might be an interesting subject for further research.

7.3 Formation of the Questionnaire

Let's assume that several questions have been built around the same subset of terms: $q_i \in Q_i$ and $\{d_k, \ldots, d_m\} \subseteq DF_i$ and that questions from Q_i cover DF_i as shown in Figure 12.

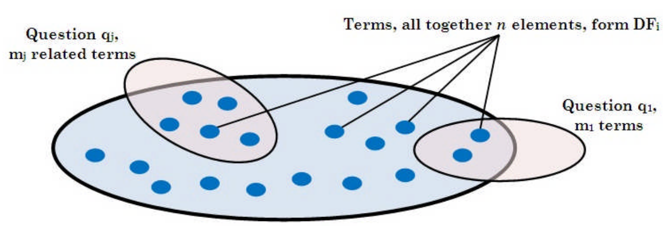


Fig. 12. Formation of DF$_u$

The procedure for the generation of set Q_i is completed when all terms of the subject (or its part) are involved in the questions and, therefore, are covered by answers of some or, at least, one question. Or formally:

$$(\forall d_k : d_k \in DF_i)(\exists q_i \in Q_i : d_k \in q_i) \qquad (13)$$

Alternatively:

$$\bigcup_{q_i \in Q_i} q_i = DF_i \qquad (14)$$

If the set of questions uses several terms more than once, we can say that the subset of mostly used terms forms a core of a discipline (or its part). The rule of belonging to the core of definitions in the discipline segment DF$_i$ is fairly simple – if there exist questions created on this segment of definitions (DF$_i$) and the intersection of these questions in the number of terms used from DF$_i$ are the biggest, then these questions and elements from the segment of definitions form a core of the segment, or formally:

$$CF_{i\text{-core}} = \{\forall q_j, q_k \in Q_j : q_k \Leftrightarrow q_k(DF_i), q_j \Leftrightarrow q_j(DF_i) \text{ and } |q_k(DF_i) \cap q_j(DF_i)| = max\} \qquad (15)$$

$$DF_{i\text{-core}} = \bigcap_{q_i \in CF_{i-core}} q_i \qquad (16)$$

The formation of a set of questions is the formation of CF, since these questions show how terms are connected and characterised. The selection of the core terms from DF can be combined with the selection of the core questions from CF. This double core, if determined, can be really useful for the formation of the course assessment, formally:

$$DF_{core} = \bigcup_{i=1}^{n} DF_{i-core} \qquad (17)$$

And:

$$CF_{core} = \bigcup_{i=1}^{n} CF_{i-core} \qquad (18)$$

The hierarchy of the importance of the questions can also be interesting as it helps to form an essential set of questions towards the required level of knowledge (grade) and adjust this level to different student abilities. Assume that for grade A we have one set of questions Q_a, for B we have Q_b , for C we have Q_c and $Q_a \supset Q_b \supset Q_c$, where A, B, C are grades of student marks. But the most important aspect of the MCAA approach is that it avoids the well known problem of MCQ, that of knowing by name. In fact, MCAA enforces students and learners to dig deeper for the meaning of definitions along with their interconnections and known predictive functions, thus pushing researchers and students to discover new knowledge – increasing the power of {PF}. R. Feynman [15], in his interview for the BBC emphasises the importance of such an approach, in our case for curriculum and assessment design and development: *"When you finished all that naming, you know absolutely nothing what that bird is; now let us look at the bird and discover what it is doing …"*.

8 MCAA efficiency

Suppose we have DF_i which has cardinality n – i.e. n terms in it and n = $|\{d_1,\dots,d_n\}|$ as shown in Figure 13. The level of their connections does not matter here.

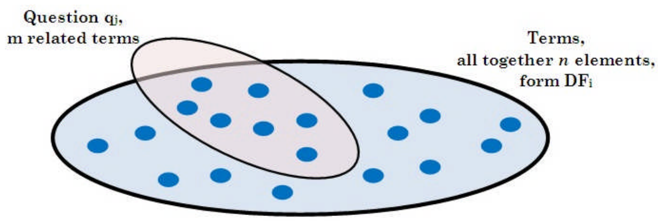


Fig. 13. Formation of questions q_i around definition domain DF_1

Assume that a set of questions $Q_i \subseteq CF_i$ which covers and includes all elements of DF_i has been built. As an example, for memory based questions in a computer architecture course, this could be: Memory: DRAM, SRAM, VRAM, Address, Data, etc. Full understanding of the meaning of the question q_j assumes a selection of right *m* related to the question terms, i.e. 100% correct answer for q_j is *m* from *n* possible choices. To avoid guessing by the students, the best ratio of *m* and *n* can be found, using features of binomial coefficients, to be:

$$2 \times m = n \qquad (19)$$

This relationship between the numbers of right and wrong answers in the proposed Multiple Choice Answers Approach can be proven by considering the number of different *m* options from *n* possible answers which is determined by the classic formula of binomial coefficients:

$$\binom{n}{m} = \frac{n!}{m!(n-m)!} \qquad (20)$$

Our task here is to find a maximum of (4) making the chances of guessing in MCAA negligible. Let's consider two extreme examples: when ***m = n-1*** and when ***m = 1***. For these two variants equal results can be found:

$$\binom{n}{1} = \binom{n}{n-1} = \frac{n!}{(n-1)!(1)!} \qquad (21)$$

Also, because we have n! in the upper part of the ratio in both variants it is easy to see that the minimum of ***m!(n-m)!*** is achieved when *m! = (n-m)!* and, thus, *n = 2m*.

Suppose that *n = 2a*, and *m = n/2*, then:

$$\binom{n}{m} = \binom{2a}{a} = \frac{(2a)!}{a!a!} = \sum_{i=0}^{a}\binom{a}{i}^2 \qquad (22)$$

For *n*=8 and *m*=4 the probability of the occasional hit on a right answer, even if the student knows that *m*=4, is 1/70 (1.43%) confirming the validity of this approach as a means towards excluding guessing and cheating during exams. Even semi-automatic procedures of formation of questions for assessment can decrease the workload of teachers. There is no doubt that there is much more to be done in the approbation of this approach in various universities and different disciplines.

There will, of course, be cases where the terms in the answer would be selected incorrectly and we would thus have: Selection ($\mathbf{q_j}$ (DF_i)) < ***m***. Furthermore, we may also face the case when all buttons might be pressed, ie ***n*** from ***n***. For these cases a *penalty function* should be established, which will decrease the mark for each wrong answer and only count the right ones.

Various forms of penalty function do exist and selection of some *best-fit* functions can be developed even though the approach towards the selection of best-fit functions requires further research. The distribution of answers, of course, varies. The best answers on questions $\mathbf{q_j}$ have hit ratio 1 (*m* right answers from *m* (the same) options).

The form and severity of the penalty function is the subject of special interest and may be different whether on assessment or training purposes. A penalty function can be constructed where its effect could be "severe" in examinations, whereas one might choose another form of "kind" testing scheme for training purposes. During tutorials, for example, student should be able to experiment with MCAA. The penalty function in this case should be somehow forgiving or even soft to enable more wrong answers without visible penalisation. On the contrary, during assessment exercises the penalty function should become severe to exclude guessing. Changing the questions for each tutorial by noting which questions a student has previously answered, provides us with the facility to automatically force student to answer all questions from the discipline, and, in fact, receive essential knowledge and be assessed concurrently with learning! We feel that this feature of MCAA makes this approach promising.

Precision of the answers depends on the ***n*** and ***m*** selections and their sizes. When a candidate does not choose all right boxes for questions correctly and does not hit any wrong answers his / her total mark can be calculated by deduction of the ratio $|\mathbf{m_r}|$ **/** $|\mathbf{m}|$ from the total mark (equal 1):

Mark in absence of wrong answers = (**1** - $|\mathbf{m_r}|$ / $|\mathbf{m}|$) (23)

Where: $\mathbf{m_r}$ is the set of not mentioned correct answers.

Thus, a candidate who misses 2 from 4 right answers gets a total mark of: 1- (2/4) = 0.5, or 50%

If the question is required to be weighted, say, cost 30 marks then the student with the sample answer above, will be awarded 15 marks.

A student, in turn, can make mistakes and may respond with incorrect answers. Again, the number of wrong answers should be weighted with the total number of wrong answers (equal to the number of right answers) and deducted from the total mark:

Mark when wrong answers present = (**1** – $|\mathbf{m_w}|$ / $|\mathbf{m}|$) (24)

Where: $\mathbf{m_w}$ is the set of wrong answers identified by student.

A generalized relative mark is calculated using the following set theory equation:

$$\mathbf{T_m = 1 - (1 / |m|) (|m - m_a| + |m_a \cap (n - m)|)} \qquad (25)$$

Where: $\mathbf{m_a}$ is the answer provided by the candidate; $\mathbf{m}$ is the right answer, $\mathbf{n - m}$ is the relative complement. Note that $\mathbf{m_r \cup m_w = m_a}$.

Further research regarding the transition from set theory equations into arithmetic equations to calculate the grade in the assessment is needed. Formalisation of assessment procedures by use of set theory Euler, Venn or Peirce diagrams might not be so easy. Forms of penalty function must consider two variables – the number of right answers and the number of all possible answers – as these are dependent on each other. Preliminary work on these diagrams failed to convert a decision rule from set theory into an arithmetic rule in order to get a grading rule. The first impression is that the resolution of uncertainties mentioned here might be found if models for DF, CF and PF and the scheme of assessment using MCAA could be described with more than two dimensions.

9 Trial Application

A trial application of the MCAA approach to assessment in real everyday practice (module: Network Technologies) in the Faculty of Computing (FoC) at the London Metropolitan University, showed very promising results. With MCAA, students achieved much better marks than those achieved by students on the previous year when MCAA was not used either for practice or for assessment. An example of a typical MCAA question used is shown in Figure 14.

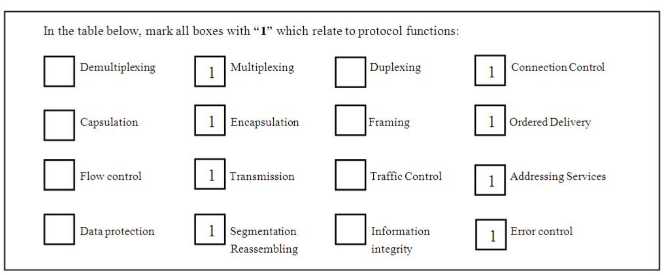


Fig. 14. An example of a typical MCAA question

Statistics accumulated, an essential part of module development procedures, for the specific module by the Assessments Unit of the university supports this claim. Comparative statistics of final results from this year, using MCAA, and last year are presented in Figure 15 and show a real trend of improvement in the module results – less failures and more students achieving A and B grades – confirming that the discussed changes and the way of implementing these changes seem to be workable. The result, as a whole, looks very promising even though it is too early to discuss efficiency of separate elements in any changes planned and realised.

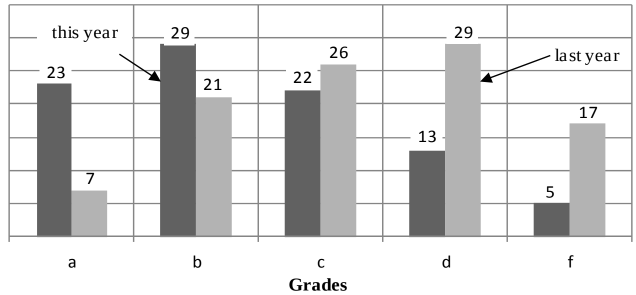


Fig. 15. Statistics of MCAA approach: 2 sequential years

10 Conclusions and Future Work

This work attempted to form a logical *core* of *curriculum design* for computer science and similar disciplines. Initially, the main terms used (science, knowledge, skills, curriculum and computer science) were defined and the role of education as a very important driving force in the improvement of life for the society at large was identified.

Existing models of *curriculum development* were briefly discussed and were found to be of a very generic nature. Their drawbacks were also identified as being applicability, usefulness and efficiency, and their measurement in the course design process. Curriculum design as *information processing* was then analysed and further developed by the introduction of three main functions in discipline construction: *definitive*, *characteristic* and *predictive*.

The *core* of the discipline, the way of its selection and its main features: *moving* and *merging*, were also analysed and discussed. The automatic formation of Glossary, Indexes, tracing their modifications, formation of assessment procedure and its realisation was shown to be one of the roles of IT. The two processes (moving and merging) were shown to be able to help detect changes in scientific areas related to the discipline automatically and assist the teacher towards almost automatic adjustment of the curriculum and assessment.

A new assessment methodology, *Multiple Choice Answers Approach* (MCAA), which apart from assessing the students also enables us to estimate the quality of teaching delivery, has been introduced and analysed. This was addressed as part of curriculum design and development, where the cycle of knowledge delivery and its progress were analysed using the same three functions: *definitive*, *characteristic* and *predictive.*

The features of the *Multiple Choice Answer Approach* assessment were analysed and the principles of the formation of multiple choice answers around a discipline, using the *core* of that discipline, were explained. A scheme for its realisation was shown to be through the creation of an automatic (or semi-automatic) assessment procedure and through the development of an algorithm for the selection of the most important material for assessment.

A process for the formation of an MCAA questionnaire and a scheme to maximize the efficiency of assessment, thus increasing the objectivity of the curriculum were also introduced. Special consideration was paid on a penalty function which even though it will consider the correct answers given it will also consider the incorrect ones which, in turn, will determine the level of penalty imposed.

A trial application of the MCAA approach to assessment on a real module at the Faculty of Computing at London Metropolitan University confirmed the applicability of this methodology with the analysis of statistics showing the growth in efficiency of teaching in this specific field of computer science.

Further development of this work will incorporate the formation of a framework and design of a tool for automatic curriculum design and update using the web and any available electronic resources.

Additional investigation on some open questions regarding the structure of a discipline in terms of models of curriculum design proposed. More specifically, what does it mean … :

- … if the Index in the module is much bigger than the Glossary: ($\{DF_u\} > \{CF_u\}$)?
- … if the Index in the book is much smaller than the Glossary ($\{DF_u\} < \{CF_u\}$)?
- … when the index and the Glossary are equal ($\{DF_u\} = \{CF_u\}$)?
- … when the index and the glossary are mixed, ie not separated ($\{DF_u\} \cup \{CF_u\}$)?

Development of the algorithm of accumulation and selection of elements for questions and glossaries as well as further extension of this algorithm towards the formation of essential questions to achieve iterative and growing understanding of the course or discipline (predictive function). Further use of set theory should formalise development of essential sub-dictionaries and questionnaires for any selected discipline.

Development of various assessment modes, including training and testing, using various supportive / advising schemes and various penalty functions – rewarding students for choosing the correct answers, even though they may not choose all the correct ones, might make assessment part of learning. Introduction and applicability analysis of special forms of penalty function for various modes of learning and testing aiming to create an applicable library, suitable for practitioners. Furthermore, the possibility of a grading function will be investigated in terms of overall level of knowledge achieved and "fine tuning" of required efforts in improving student knowledge.

Finally, the creation of an automatic framework for almost real time analysis of distributions of results obtained during the approbation period of the use of MCAA in assessment practice.

11 References

[1] Dijkstra, E. "The End of Computer Science?" Communications of the ACM, 44, 3, 92, 2001.

[2] Penguin, The New PENGUIN English Dictionary, 2001

[3] Jeans, J. The Mysterious Universe, New York: Macmillan, p.106 and p.138, 1930.

[4] ChinaCulture.org, Ministry of Culture, P. R. China, http://www.culturalink.gov.cn/gb/en_aboutchina/2003-09/24/content23084.htm (accessed: 8 Apr 2010)

[5] Toohey, S. Designing Courses for Higher Education, Buckingham, The Society for Research into Higher Education & Open University Press, 1999.

[6] Ramsden, P. Learning to Teach in Higher Education, London, Routledge, 1992.

[7] Diamond R. Designing and Improving Courses and curricula in HE, 1989.

[8] Schagaev, I. "Yet Another Approach to Classification of Redundancy", IMEKO Congress, 1990, Helsinki, Finland, pp 485-490, 1990.

[9] Schagaev, I. and Buhanova, G. "Comparative Study of Fault Tolerant RAM Structures", Proceedings of IEEE DSN Conference, Sweden, Goteborg, July, 2001.

[10] Schagaev I. "Redundancy Classification For Fault Tolerant Computer Design" Proc. of the 2001 IEEE Systems, Man, and Cybernetics Conference, Tuscon, AZ.

[11] Carter W. Digital Designing with Programmable Logic Devices, Prentice Hall, ISBN: 0133737217, 1996.

[12] Schagaev, I., Bacon, E. and Ioannides, N. "An Approach to Curriculum Design for Computer Science", 3rd World Summit on the knowledge Society, Corfu, September, in M.D. Lytras et al. Eds. WSKS 2010, Part I, CCIS 111, 493-499.

[13] Schagaev, I., Bacon, E. and Ioannides, N. "Curriculum Design and Development for Computer Science and Similar Disciplines". International Journal of Knowledge Society Research (IJKSR), 1, 3, 17-32, 2010.

[14] Analytic Philosophy, In Encyclopaedia Britannica, 2011. Retrieved from: http://www.britannica.com/EBchecked/topic/22568/analytic-philosophy (accessed: 09 Jan 2011)

[15] Feynman, R. (1981) BBC interview, BBC Horizon. Retrieved from: http://video.google.co.uk/ videoplay?docid=8777381378502286852# (accessed: 09 Jan 2011)

Adaptive Curriculum in Computer Literacy Course

Nasser Tadayon[1]**, Nathan Barker**[1]**, and Connie Nyman**[1]
[1]Department of Computer Science and Information Systems
Southern Utah University
Cedar City, Utah, USA

Abstract: *The adaption of constructivist change theory (Lueddeke) to the dynamic curriculum content related to teaching technology is applied through a computer literacy course. An increasing dependency on technology has reshaped educational systems worldwide to adjust their curriculum and embed the requirement of basic technical skills among graduates at all levels. In this paper, we present an analysis of data that supports the application of our continuous improvement process to the technology sections of a computer literacy course. We also demonstrate a more comprehensive analysis of the student survey results, which is the primary motivation for updating the topics in the course. The survey results rank the topics covered in the course based on student feedback on the usefulness and appeal of subjects. The appropriate modifications will be applied based on this feedback. Modifications will be discussed among faculty and a collective effort will determine the final change. Furthermore, the analysis of data identifies the topics most liked or disliked by a group of students based on their gender, classification, and major program.*

Keywords: Computer literacy, Education, Technology, Adaptive curriculum

1. Introduction

Information literacy, as part of the required and essential technical competency for all college graduates, has been one of the main topics of study worldwide. In the US, states have defined their standards and developed curricula for Higher Education and Secondary Education to meet their required information literacy. Based on Wikipedia, the most common definition of information literacy is "...the ability to know when there is a need for information, to be able to identify, locate, evaluate, and effectively use that information for the issue or problem at hand". The "Information Literacy Competency Standards for Higher Education" report from the Association of College & Research Libraries (ACRL) defined information literacy as a "set of abilities requiring individuals to recognize when information is needed and have the ability to locate, evaluate, and use effectively the needed information." The report indicates the importance of information literacy in higher education and points out the fact that information literacy is considered a key outcome for college students by the Western Association of Schools and College (WASC) and the Southern Association of Colleges and Schools (SACS).

Computer Literacy, on the other hand, is defined as the knowledge and ability to use computers and related technology efficiently, with a range of skills covering levels from elementary use to programming and advanced problem solving. The level of computer literacy one must achieve to gain an advantage over others depends both on the society one is in and one's place in the social hierarchy (Wikipedia). Employers in the 21st century want their workers to not just be information literate but also familiar with computer applications that they would use on a daily basis. Our ever-increasing dependency on high-tech tools for almost everyone has created a new environment that demands a basic understanding and application of emerging technologies. We define the computer literacy requirement for students at the Post-Secondary Education level to be "exploration of the concepts of microcomputer systems" which includes the knowledge and the use of a variety of computer software applications. Aligned with the rapid changes in technology, the elements of a computer literacy course need to change and hence it is imperative to frequently revise and update the course to include the latest technological advancements.

2. Computer Literacy

The ACRL report defines five standards and twenty-two performance indicators to classify someone as information literate. The standards are outlined as the ability of an individual to:

- Determine the extent of information needed
- Access the needed information effectively and efficiently
- Evaluate information and its sources critically and incorporate selected information into one's knowledge base
- Use information effectively to accomplish a specific purpose
- Understand the economic, legal, and social issues surrounding the use of information, and access and use information ethically and legally

Although most researchers emphasize information literacy that is common to all disciplines, learning environments, and levels of education, they rely extensively on information technology. Computer literacy and competency as a major and required component of information technology has been gaining more attention. With the rise of the internet age, many aspects of information literacy are associated with the use of computers (Association of College and Research Libraries).

This paper describes and analyzes the process used to adjust the curriculum with the rapid changes in technology for an introductory computer literacy course at Southern Utah University. The course is a required general education (GE) course and is divided into two sections--Applications and Technology. The Applications part of the course has specific outcomes derived from standards defined by the state computer literacy requirement. The topics used in the Technology section of the course are basically essential and updated knowledge of some basic computer concepts and software that are commonly used. These topics are adapted and changed based on the feasibility, the result of the students' surveys, and the faculty input in order to align with the dynamic technological competency needs of the students. Critical information technology competencies are also often taken for granted (Liao and Pope). This course is taught through hands-on projects and in-class activities (for in-class sections) that have been proven to be one of the most helpful strategies to learn technical concepts (Florence and Qi).

The technology section was assessed through assignments and tests split between two parts called Technology 1 and Technology 2.

During Fall-2011 semester, Technology 1 and 2 contained the following topics:

Technology 1:

- Intro to Firefox Extensions
- Intro to Cyber Crime, Electronic Commerce, and Ethics
- Intro to Audio Editing
- Intro to Video Editing
- Intro to 3D Computer Graphic
- Intro to the Command Line
- Intro to Cloud Computing
- Intro to Linux Mint

Technology 2:

- Intro to Networking
- Intro to Web Development
- Intro to Photo Editing
- Intro to Image Creation Using GIMP
- Intro to Computer Programming
- Intro to Open Office
- Intro to Web 2.0

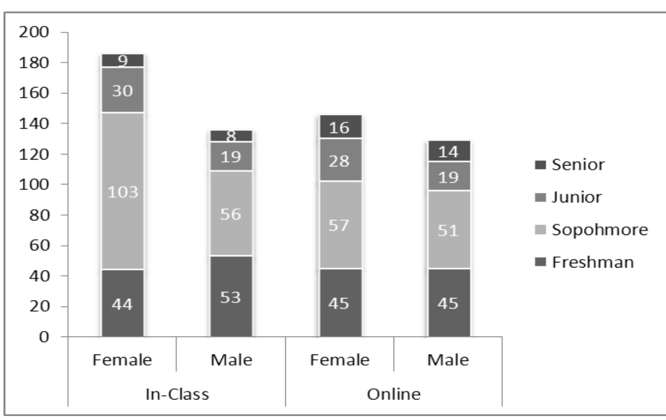
Computer literacy included, in addition to the above technology topics, sections covering Microsoft applications (Word, Excel, Access, and PowerPoint). At the end of each semester, a student survey asks students to rate their level enjoyment for all sections of the course.

The following is a report of the findings which compares the results obtained based on gender, classification, delivery method, and students' major study. The data collected during Fall 2011 are based on 624 students from 12 in-class sections (333 students) and 11 online sections (291 students).

3. Assessing the level of enjoyment

The ranking assignment contains a question for each topic in the form "Please rate the following assignment on a scale of 0 to 10, where 10 is enjoyed, 5 is neutral, and 0 is hated". For each topic, the students are asked to provide a numeric value in response to a question. The final question in the assessment process provides feedback for assessment and is mentioned as: "We try to expose you to some of the most interesting issues in today's technology. Because this area changes so quickly, we do not want to fall behind. Please list some suggested topics that you believe might be of interest to future students taking CSIS 1000."

Data from 624 students was collected and 27 of them were removed from the set of data considered due to error in the values entered (not within the range or unspecified). The data collected from 597 students were used for analysis. The following Figure demonstrates the overall student demographic.

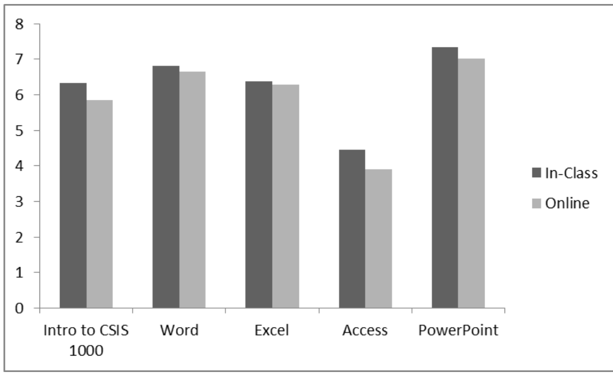


	In-Class		Online		
	Female	**Male**	**Female**	**Male**	**Total**
Freshman	44	53	45	45	187
Sopohmore	103	56	57	51	267
Junior	30	19	28	19	96
Senior	9	8	16	14	47
Total	186	136	146	129	597

Figure 1-Student demographic

3.1 Result of ranking the level of enjoyment for the application section

The average in overall ranking of the assessment data for each section is separated based on method of delivery (In-Class vs. Online). The results for the Application section are shown in Figure 2.

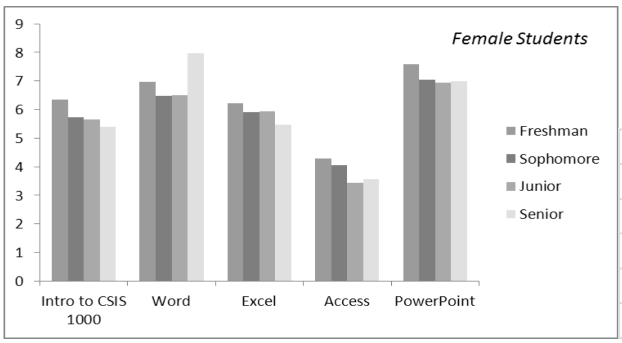


	In-Class	Online
Applications	Average Ranking	
Intro to CSIS 1000	6.337	5.858
Word	6.806	6.649
Excel	6.381	6.287
Access	4.458	3.911
PowerPoint	7.338	7.027
Overall Average	6.264	5.946

Figure 2- Average Ranking for Application

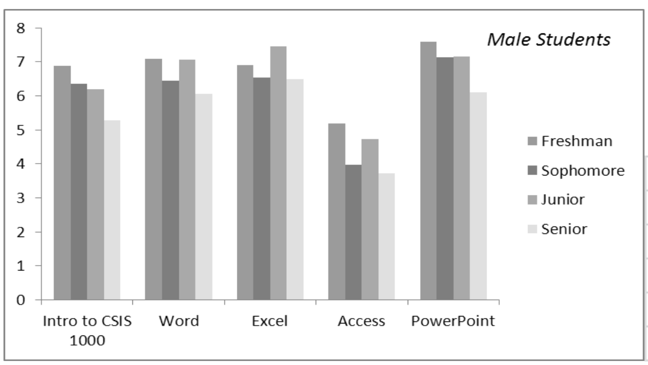
The data indicate that PowerPoint was the most liked and Access was the least liked subject among the topics in the Application section for both in-class and online. Average rankings of in-class students have been higher than online ones.

The rankings of 332 female students and 256 male students for both in-class and online based on their classification are shown in Figures 3 and 4.



Female	Intro to CSIS 1000	Word	Excel	Access	PowerPoint
Freshman	6.36	6.98	6.21	4.28	7.59
Sophomore	5.74	6.48	5.92	4.04	7.05
Junior	5.66	6.51	5.94	3.44	6.94
Senior	5.40	7.96	5.48	3.56	7.00
Average	5.87	6.73	5.97	3.96	7.17

Figure 3-Application Ranking by Female Students

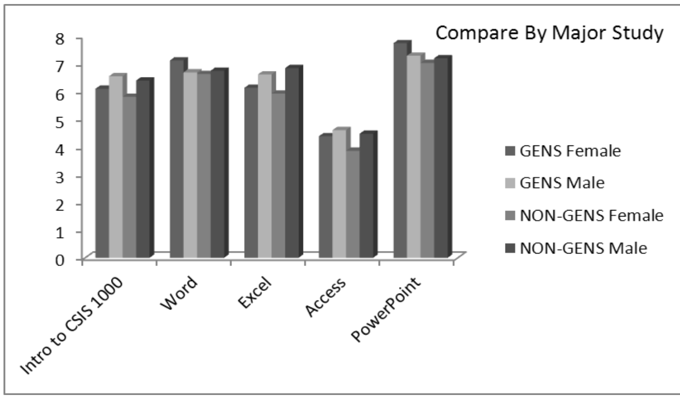


Male	Intro to CSIS 1000	Word	Excel	Access	PowerPoint
Freshman	6.88	7.10	6.91	5.18	7.60
Sophomore	6.35	6.45	6.54	3.98	7.14
Junior	6.18	7.05	7.45	4.74	7.16
Senior	5.27	6.05	6.50	3.73	6.09
Average	6.43	6.74	6.80	4.51	7.23

Figure 4-Application Ranking by Male Students

The results indicate that PowerPoint was the most preferred by male and female freshman students. The Word section was appreciated most by senior female students (mostly online) and least by senior male students. Overall, female students liked Word and PowerPoint the same as male students but male students liked Excel significantly more than female students. All topics except Word for female students and Excel for male students were enjoyed more by freshman students than others. Access was disliked more by female students than male students.

The figure below demonstrates the differences in rankings for 120 students within GENS (General Education) majors compared to 477 other (non-GENS) students. In general, the GENS (General Education) major students ranked the application section higher than non-GENS majors.



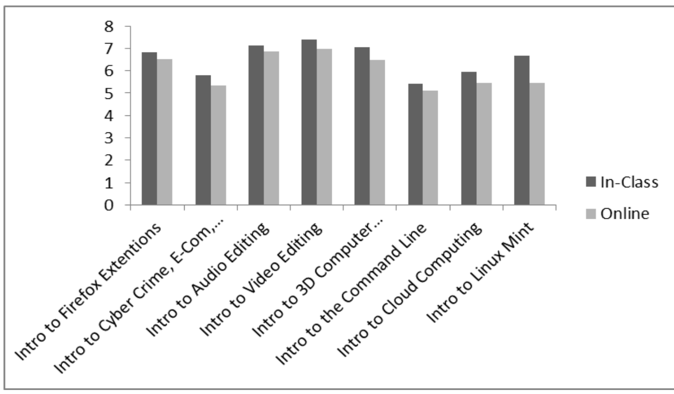
Major	GENS		NON-GENS	
Application	Female	Male	Female	Male
Intro to CSIS 1000	6.10	6.55	5.81	6.40
Word	7.13	6.69	6.63	6.75
Excel	6.14	6.62	5.93	6.85
Access	4.39	4.61	3.86	4.48
PowerPoint	7.75	7.30	7.04	7.20
Overall Average	6.30	6.35	5.85	6.33

Figure 5-Application Ranking by Major Study

Non-GENS male students provided higher rankings in all subjects than females whereas within the GENS majors, female students have ranked Word and PowerPoint higher than males.

3.2 Results of ranking the level of enjoyment for the Technology 1 section

The following figure compares the rankings of the Technology 1 section for in-class vs. online.

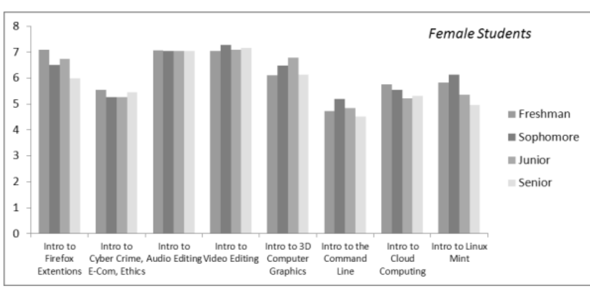


	In-Class	Online
Technology 1	**Average Ranking**	
Intro to Firefox Extentions	6.836	6.507
Intro to Cyber Crime, E-Com, Ethics	5.808	5.326
Intro to Audio Editing	7.143	6.872
Intro to Video Editing	7.386	6.994
Intro to 3D Computer Graphics	7.047	6.471
Intro to the Command Line	5.411	5.115
Intro to Cloud Computing	5.968	5.446
Intro to Linux Mint	6.669	5.459
Overall Average	6.534	6.024

Figure 6- Average Ranking for Technology 1 –In-Class and Online

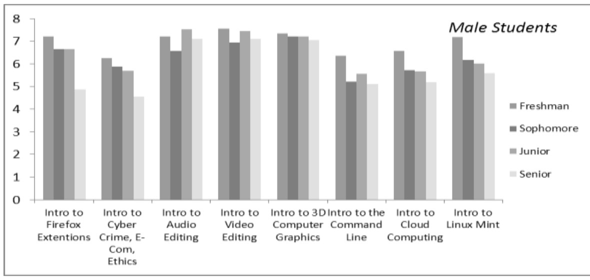
The results show that the in-class students have given a higher ranking for all topics within the Technology 1 section compared to online students (similar results were observed in the Application section). The “Intro to Linux – Mint” exercise had a relatively significant difference in rankings due to the difficulty of exercises without proper assistance.

The overall rankings of the level of enjoyment for all the topics within the Technology 1 section based on gender are shown below.



Female Technology 1	Intro to Firefox Extentions	Intro to Cyber Crime, E-Com, Ethics	Intro to Audio Editing	Intro to Video Editing	Intro to 3D Computer Graphics	Intro to the Command Line	Intro to Cloud Computing	Intro to Linux Mint	Average
Freshman	7.09	5.54	7.06	7.04	6.12	4.73	5.76	5.82	6.14
Sophomore	6.51	5.27	7.05	7.28	6.49	5.20	5.55	6.13	6.19
Junior	6.74	5.26	7.03	7.10	6.78	4.85	5.21	5.37	6.04
Senior	6.00	5.46	7.04	7.16	6.12	4.52	5.32	4.96	5.82
Average	6.67	5.36	7.05	7.18	6.42	4.96	5.53	5.82	6.12

Figure 7-Average Ranking of Technology 1 - Female

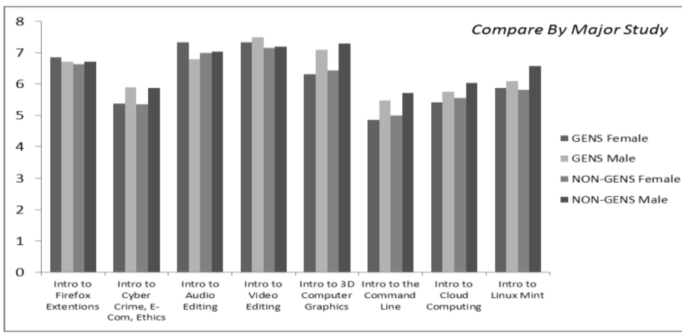


Male Technology 1	Intro to Firefox Extentions	Intro to Cyber Crime, E-Com, Ethics	Intro to Audio Editing	Intro to Video Editing	Intro to 3D Computer Graphics	Intro to the Command Line	Intro to Cloud Computing	Intro to Linux Mint	Average
Freshman	7.21	6.25	7.22	7.54	7.34	6.35	6.56	7.18	6.96
Sophomore	6.65	5.88	6.56	6.93	7.21	5.20	5.72	6.17	6.29
Junior	6.66	5.68	7.53	7.45	7.21	5.55	5.66	6.00	6.47
Senior	4.86	4.55	7.09	7.09	7.04	5.10	5.18	5.59	5.81
Average	6.71	5.88	6.98	7.24	7.25	5.67	5.98	6.47	6.52

Figure 8-Average Ranking of Technology 1 - Male

With the exception of "Intro to Audio Editing", which appealed to junior male students, male freshman students almost always liked the topics more. Female freshman students gave the highest ranks to “Intro to Firefox”, “Intro to Cyber Crime, E-Com, Ethic”, Intro to Audio Editing”, and Intro to Cloud Computing”. However “Intro to Video Editing”, “Intro to the Command Line”, and “Intro to Linux-MINT” were most liked by sophomore female students. Among female students, “Intro to 3D Computer Graphics” was liked most by juniors. Overall, male students liked all topics in Technology 1 except “Intro to Audio Editing” more than female students.

The figure below demonstrates the difference in rankings for students with the General Education major compared to other students for topics in Technology 1. In general, the GENS (General Education) major students ranked the Application section higher than non-GENS major students.



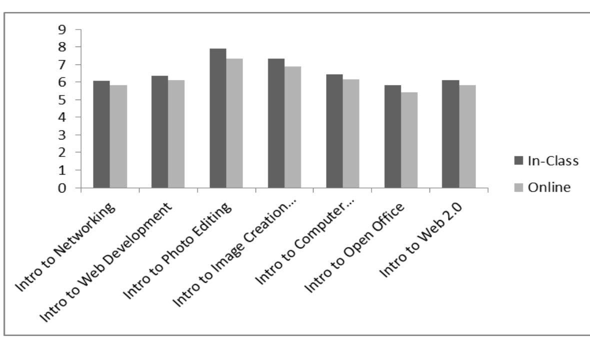
Major	GENS		NON-GENS	
Technology 1	Female	Male	Female	Male
Intro to Firefox Extentions	6.86	6.70	6.62	6.71
Intro to Cyber Crime, E-Com, Ethics	5.38	5.89	5.35	5.87
Intro to Audio Editing	7.32	6.79	6.98	7.03
Intro to Video Editing	7.32	7.49	7.14	7.18
Intro to 3D Computer Graphics	6.32	7.08	6.44	7.28
Intro to the Command Line	4.85	5.47	4.98	5.72
Intro to Cloud Computing	5.42	5.75	5.55	6.03
Intro to Linux Mint	5.87	6.09	5.81	6.57
Average	6.17	6.41	6.11	6.55

Figure 9- Technology 1 Ranking by Major Study

The overall average for several topics in Technology 1 was the highest among males in non-GENS majors. There were a few topics that GENS students liked more. “Intro to Audio Editing” and “Intro to Video Editing” were especially more popular among GENS students.

3.3 Results of ranking the level of enjoyment for the Technology 2 section

The following figure shows the rankings of the Technology 2 section for in-class vs. online.



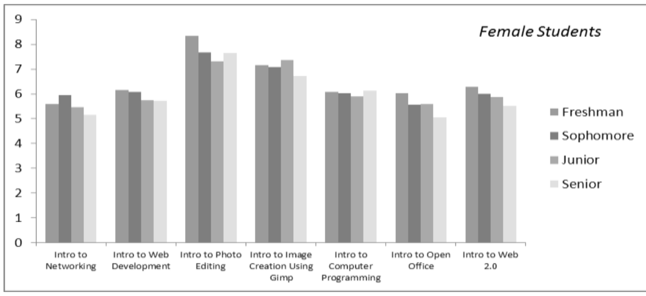
	In-Class	Online
Technology 2	**Average Ranking**	
Intro to Networking	6.072	5.808
Intro to Web Development	6.355	6.109
Intro to Photo Editing	7.888	7.317
Intro to Image Creation Using Gimp	7.332	6.881
Intro to Computer Programming	6.453	6.155
Intro to Open Office	5.845	5.401
Intro to Web 2.0	6.099	5.843
Overall Average	6.578	6.216

Figure 10-Average Ranking for Technology 2 – In-Class and Online

The results show that in-class students have given a higher ranking for all topics within the Technology 2 section compared to online which is consistent with all other topics.

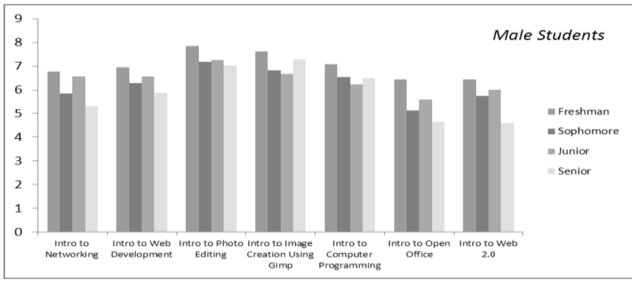
The “Intro to Photo Editing” exercise had a relatively high ranking, especially among in-class students.

The rankings in level of enjoyment for each of the topics within the Technology 2 section based on gender are shown below.



Female Technology 2	Intro to Networking	Intro to Web Development	Intro to Photo Editing	Intro to Image Creation Using Gimp	Intro to Computer Programming	Intro to Open Office	Intro to Web 2.0	Average
Freshman	5.60	6.14	8.33	7.16	6.07	6.01	6.28	6.51
Sophomore	5.96	6.07	7.67	7.08	6.03	5.57	6.00	6.34
Junior	5.46	5.75	7.32	7.35	5.91	5.59	5.87	6.18
Senior	5.16	5.72	7.64	6.72	6.12	5.04	5.52	5.99
Average	5.71	6.01	7.78	7.12	6.02	5.65	6.02	6.33

Figure 11-Average Ranking of Technology 2 - Female

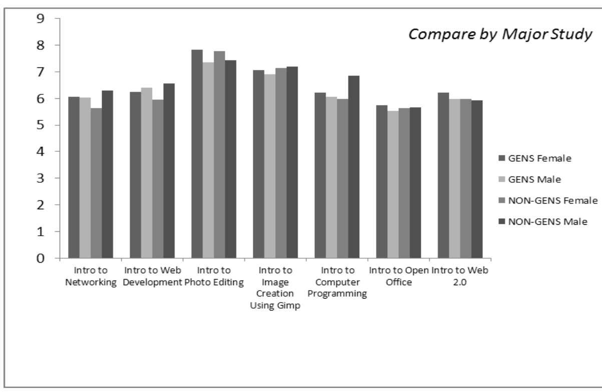


Male Technology 2	Intro to Networking	Intro to Web Development	Intro to Photo Editing	Intro to Image Creation Using Gimp	Intro to Computer Programming	Intro to Open Office	Intro to Web 2.0	Average
Freshman	6.78	6.96	7.85	7.62	7.07	6.43	6.44	7.02
Sophomore	5.85	6.28	7.17	6.83	6.53	5.14	5.73	6.22
Junior	6.55	6.55	7.26	6.66	6.24	5.58	6.00	6.41
Senior	5.32	5.86	7.03	7.27	6.50	4.64	4.59	5.89
Average	6.25	6.53	7.42	7.13	6.68	5.64	5.94	6.51

Figure 12-Average Ranking of Technology 2 - Male

All freshman male students liked the topics most and in many cases significantly more than other male students. Most of the topics were liked most by female freshman students as well. Female sophomore students liked “Intro to Networking” and female junior students liked “Intro to Image Creation using GIMP” the most. Senior female students liked “Intro to Computer Programming” more than other female students.

The figure below demonstrates the student ranking for GENS (General Education) majors compared to other non-GENS students for topics in Technology 2.



Major	GENS		NON-GENS	
Technology 2	Female	Male	Female	Male
Intro to Networking	6.06	6.04	5.63	6.30
Intro to Web Development	6.24	6.40	5.95	6.56
Intro to Photo Editing	7.83	7.36	7.77	7.43
Intro to Image Creation Using Gimp	7.06	6.92	7.13	7.19
Intro to Computer Programming	6.22	6.06	5.98	6.84
Intro to Open Office	5.75	5.52	5.62	5.66
Intro to Web 2.0	6.22	5.99	5.97	5.92
Average	6.48	6.33	6.29	6.56

Figure 13- Technology 2 Ranking by Major Study

The highest rankings are distributed among GENS female students and non-GENS male students on different topics. “Intro to Networking”, “Intro to Web Development”, “Intro to Image Creation Using GIMP”, and “Intro to Computer Programming” have the highest ranking by non-GENS male students. In specific, “Intro to Computer programming” was significantly more popular among non-GENS male students. Among other popular topics, “Intro to Photo Editing” was ranked relatively high by GENS female students.

4. Conclusion

The overall student rankings of all topics for Fall 2011 are shown in the following figure.

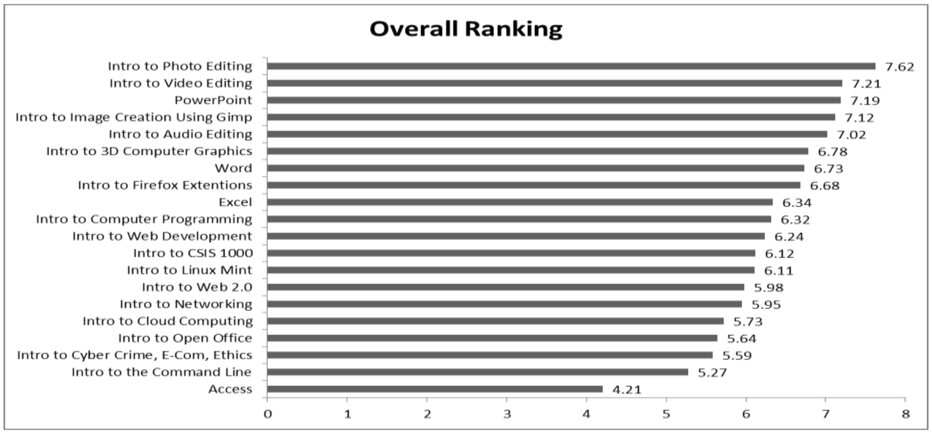


Figure 14- Overall Ranking of all topics sorted

A similar survey was conducted in the previous year in which the most liked topic for the students was “Photo Editing (using Gimp)”. The other topics liked by students were PowerPoint, Image Creation, and Audio Editing. Access was among the least liked topics followed by E-Commerce and Cyber Crime which were combined and modified for this year.

Before Fall 2011 and at the end of the academic year 2010-11 the topics and contents within the technology sections of the course were updated and modified for the new academic year. The modifications were as follows:

- “Intro to Video Editing” replaced “Intro to Human Computer Interaction”
- “Intro to Cloud Computing” replaced “Intro to Linux”
- “Intro to Image Creation Using GIMP” was added

The rankings of topics will be repeated during Spring 2012 and appropriate modification or adjustment to the topics will take place for next year.

References:

[1] Lueddeke, George. "Reconciling Research, Teaching and Scholarship in Higher Education" International Journal for the Scholarship of Teaching and Learning (Vol. 2, No. 1 (January 2008)).

[2] Jeffrey Brewer, Alka Harriger, and John Mendonca, “Beyond the Model: Building an Effective and Dynamic IT Curriculum” Journal of Information Technology Education (Volume 5, 2006).

[3] Association of College and Research Libraries. Information Literacy Competency Standards for Higher Education. Chicago, Illinois, 2008.

[4] Liao, Luby and Jack W Pope. "Computer Literacy for Everyone." Consortium for Computing Sciences in Colleges. 2008. 231-238.

[5] Tadayon, Nasser, Connie Nyman and Nathan Barker. "Effect of Technology in Computer Literacy Courses." FECS 2010. 129-134.

A Minor in Game Development

Serhiy Morozov, Kevin Daimi, and Yali Wu
Department of Mathematics, Computer Science, and Software Engineering
University of Detroit Mercy, Detroit MI, USA
{morozose, daimikj, wuya1}@udmercy.edu

***Abstract*—Game development is one of the fastest growing industries today. The demand for popular games is impressive and the prospects for employment are promising. However, many recent college graduates do not specialize in this field. A generic computer science degree is adequate, but it could be improved with more relevant training. We propose a dedicated minor that emphasizes software development to prepare students for future study or employment in the game industry. We anticipate that this minor will distinguish students from other recent graduates, increase enrollment, and improve feedback from existing students.**

***Index Terms*—Minor Design, Game Development, Curriculum Design, Assessment, Objectives**

I. Introduction

Many consider game development to be unworthy of an academic pursuit. Recent news regarding the sexual and violent content in games may have enforced this concern. Other types of media like movies and magazines contain the same kind of content, yet film and journalism schools still exits. Perhaps such double standards exist because games have traditionally been developed for children. However, recent statistics show an average gamer age to be 32.82 for males and 33.49 for females, just under the general population average [34]. Gaming is now one of the most popular forms of entertainment for all age groups, with 93% of all games rated "E" for Everyone [2], [22]. Games are relatively new, but they are as prevalent as the more traditional media, especially on campuses full of students and young faculty who grew up playing video games [25].

Gaming is an extremely profitable business with one of the fastest growth rates in the United States economy. In just a few years, the game industry has doubled in size and is now larger than Hollywood [1], [2], [21], [25], [31], [35]. In 2007, the industry earned $8.85 billion in revenue out of $18 billion in sales, which is a 43% increase from the previous years [11], [33]. Considering such rapid growth, the game industry will soon employ as many people as the US Navy, which is the largest in the world [22], [35]. For instance, Electronic Arts (EA) has doubled in size since 2000. It now employs 6,450 people, with more than half of them directly involved with game creation [25]. This company has established itself as an industry leader in supporting education efforts in game development. It has invested millions of dollars to support schools that train their future employees [25]. Consequently, many schools that initially hesitated to acknowledge game development as a serious application of software engineering now graduate students who are readily employed in this industry. The number of such game-related programs has increased more than 10 times over the last five years [25], [35]. There is no doubt that game development is on the rise.

Recent computer science graduates are excited to work for a game studio. Unfortunately, it is difficult for inexperienced students to stand out among others. In some cases the competition for a job opening could be as high as 1:100 [16]. It is more difficult to get a job at a small company, where software engineers make up a relatively small portion of the team and the majority of employees are involved with content creation, e.g., artists, musicians, and designers [15], [21], [33]. Large companies put more emphasis on the technology, typically hiring 65% computer scientists, 30% artists, and 5% designers [35]. We propose a dedicated game development minor that will prepare students for future employment at such large game studios.

However, a minor in game development does not restrict students' careers to a single industry. It offers a skill set that can be readily applied to many other disciplines. For instance, modern games often employ state of the art computer simulation technology that is typically directed at entertainment. The same technology may simulate other real-world scenarios to accomplish more functional goals, e.g., surgery simulation, psychological rehabilitation, and pain mitigation [2], [23]. In fact, the difference between a software developer at Electronic Arts and a software developer on Wall Street may come down to their clothing preferences [25].

II. Background

Game development courses attract new students. For example, computer science enrollment at Daniel Webster College has almost tripled in one year [10]. The University of Southern California anticipates 45% more applications with 80% of new students interested in game development. Denver University expects three times as many new game development majors [35]. Likewise, the schools observe a positive change in students themselves. The game development courses typically have better grades, fewer dropouts, and more positive feedback [32] because students enjoy the classes and can relate to the material.

The best games are not necessarily complex, so game development does not have to be overly technical. For example, small teams of freshman at Daniel Webster College developed commercial quality casino and board games with only one semester of introductory programming experience [10]. New students experienced an incredible confidence boost and realized the power of computer science [10], [12]. When compared with traditional programming courses, students were more invested, interested, and engaged in the course. They spent more time working on their project, wrote more lines of code, and achieved a better understanding of the course material [12], [31], [32]. Some students even continued to work on their games after the conclusion of the course [3], which indicates genuine interest.

Game development also motivates existing students. Simple game development projects promote a more thorough understanding of difficult programming concepts and encourage students to explore computer science further [3], [9]–[12], [17], [32]. Simple games allow students to focus on the strategy, rather than implementation details, which is a key concept in successful software design. No one can write a perfect game on the first try, so frequent changes are common. As a result, students recognize the need for modularity and incremental development, which are also core principles of software engineering. Furthermore, such projects improve communication skills, facilitate collaborative learning, and motivate students to work harder [3], [32].

Game development projects offer comprehensive practical experience. For instance, such projects give students an opportunity to learn by construction [3]. Students are intimately familiar with the architecture of their favorite games, which makes it easy to visualize concepts without implementation concerns. Sometimes students spend the entire meeting discussing fun ideas [28]. Many are too ambitious and far-fetching, but creativity empowers students because their opinions matter. A simple game project offers the three things that truly motivate students, autonomy, mastery, and purpose [19].

Planning such projects is difficult because the teacher must balance the entertainment and instructional goals. However, many educational games have been successfully developed to improve student learning in mathematics, computer science, physics, and even poetry [2], [27]. For example, drill and practice games increase the speed and accuracy of math skills [3]. They engage students in active learning, encourage collaboration, and promote social participation [13]. Games have even been developed for the Department of Defense (DoD) as a tool for teaching the latest theory of military operations [30]. This shows how game development can go beyond entertainment and have a significant influence on the way we approach learning.

III. Minor Objectives

Program objectives depict the career and professional achievements that the program is training students to attain. Since this minor will exist within the software engineering program, we incorporate many recommendations of the Accreditation Board for Engineering and Technology (ABET). The proposed game development minor has the following objectives:

1) Introduce game development methods, techniques, and tools.
2) Provide the knowledge necessary for understanding and implementing games.
3) Prepare students for the interdisciplinary nature of commercial game development.
4) Prepare students for further study or employment in the game development industry.

IV. Minor Outcomes

Outcomes illustrate what graduates are anticipated to comprehend and be able to perform in terms of knowledge and capabilities. The outcomes are consistent with the existing minors [5], [6] within the software engineering program. Upon completion of the game development minor, students will be able to do the following:

1) Apply various techniques, algorithms, and tools to solve game development problems.
2) Design a storyboard for a game.
3) Implement games using scripting and compiled languages.
4) Manipulate commercial graphic libraries.
5) Present their work orally and in writing.
6) Collaborate with other students on team projects.

V. Minor Requirements

The proposed curriculum includes fundamental techniques as well as practical programming experience relevant to any software engineering industry. We have adopted many suggestions from the International Game Developer Association (IGDA), which is comprised of industry representatives and academics [11], [16], [18], [22], [31]. Table I lists the required courses in the order of increasing complexity.

Our curriculum includes existing computer science courses as well as four new ones, dedicated exclusively to game development. The existing courses cover the conventional topics of introductory computer science. Students are required to have basic programming experience, which is covered in Introduction to Programming (C++) I and II, CSSE 1710 and 1720 respectively. However, industrial programming experience or coursework in a different programming language may also satisfy this requirement.

The game development topics are evenly distributed across three courses, CSSE 3430, 4250, and 4260. The first one provides a broad exposure, without focusing too heavily on one area. The other two courses cover the essential topics in game development. Also, students have an opportunity to apply their knowledge in a capstone project and earn the experience that is often required in the industry.

Even though game development is a massively interdisciplinary field, our curriculum has been carefully chosen to emphasize technical competence. Software development is the core component of game development [2], [23], which is highly valued by the industry. This curriculum will appeal to computer science and software engineering

TABLE I
MINOR REQUIREMENTS

Course	Course Name	Credits	Prerequisite Co-requisite
CSSE-2130	Programming in Java	3	CSSE-1720
CSSE-2430	Data Structures and Algorithms	3	CSSE-1720
CSSE-3430	Introduction to Game Programming	3	CSSE-1720 and CSSE-2430
CSSE-4250	Game Design I	3	CSSE-3430
CSSE-4260	Game Design II	3	CSSE-4250
CSSE-4940	Game Development Capstone Project	3	CSSE-4260

majors because of the three-course overlap. However, students from other departments who have taken basic programming courses may also be interested.

VI. Assessment

Assessment is a vital practice for any program. The assessment of the game development minor uses standard practices employed by the software engineering program. Program assessment will include the following:

- A capstone project demonstrating the skills developed throughout the minor.
- A comprehensive exam, which covers all the topics of the minor. It will be given to students during week 13 of their last semester.
- A senior survey, which is required of all graduating seniors. It will include the minor-specific questions.
- A research paper on an application field for each course in their minor.
- In class exams, quizzes, and homeworks measuring student comprehension of the class material.

Samples of exams, projects, and other assignments will be collected over a range of quality for evaluation of outcomes and syllabi content. Assessment information will be processed by the department assessment coordinator. The chair of the program will call a joint meeting of all participating faculty to facilitate any modification of the minor.

VII. Course Contents

CSSE-2130 Programming in Java
: Java Applications, Java Applets, Control Structures, Methods, Arrays, Strings and Characters, Object-Oriented Programming, Graphics and Java 2D, Basic Graphical User Interface Components, Exception Handling, Files and Streams, Java Utilities.

CSSE-2430 Data Structures and Algorithms
: Object-Oriented Principles, Standard Data Structures, Big-O Notation, NP Completeness, Stacks, Queues, Generic Data Types, Dynamic Memory, Recursion, Linked Lists, Circular Lists, Doubly Linked Lists, Trees, Binary Search Trees, Heaps, Graphics, Sorting Algorithms, Searching Algorithms, Object-Oriented Language Implementation of Data Structures and Algorithms.

CSSE-3430 Introduction to Game Programming
: Game Project, Game Creation, Vector Based Graphics, Bitmap Based Graphics, Simple Sprites, Sprite Animations, Keyboard and Mouse Input, Sound Effects and Music, Timing, Game Loop, Graphics Libraries, Game Programming Projects.

CSSE-4250 Game Design I
: Design Components and Processes, Game Concepts, Game Worlds, Character Development, Storytelling and Narrative, User Interfaces, Animation, Audio/Visual Design, Game Engines, Gameplay, Game Balancing, Level Design, Action Games, Strategy Games, Role-Playing, Sports Games, Vehicle Simulations, Adventure Games, Online Gaming, Prototyping, Projects.

CSSE-4260 Game Design II
: Agents, Percepts and Actions, Actions and Movement, Combining Steering Behaviors, Path Finding, Decision Making, Behavior Trees, Goal Oriented Action Planning, Hierarchical Task Planning and Partial Order Planning, Tactical and Strategic AI, Reinforcement Learning, Board Games, Anytime Algorithms, World Interfacing, Geometric Queries to Games, Intelligent Content, AI Tools and Content Creation, Scripting, Designing Intelligent Games, Case Studies.

CSSE-4940 Game Development Capstone Project
: This is a team-oriented project course in which teams design, implement, test, and document a computerized game. The instructor will either offer or solicit a selection of projects. Projects may include students from other disciplines.

VIII. Curriculum Discussion

The minor starts with basic programming, data structures, and algorithm courses. Once students have built up enough programming knowledge, they should feel comfortable writing basic applications and interacting with typical graphical libraries, i.e., the main frameworks used in the industry [23]. In the Introduction to Game Programming course, students will begin actual game development, which is a more specialized version of the traditional computer programming. This course is a prerequisite for the more advanced courses, which may not be taken concurrently. Such curricular design minimizes student investment in the minor, in case they choose not to pursue it further. The two dedicated game development courses (CSSE 4250 and 4260) will expose students to the theory of game design, the structure of the production process, and a team project that mimics industry requirements. They will offer both supervised and independent projects, which build teamwork as well as individual competence.

The minor ends with a project course where students develop an original game. It gives students the freedom to be creative, apply their skills, and demonstrate their knowledge. Capstone projects enhance team-work, software development experience, and proficiency with professional tools [4], [28], [31]. Furthermore, students gain valuable experience and build a significant portion of their professional portfolio, which will be invaluable for industry interviews [26], [35]. Upon completion, we will encourage students to present their work during the "Tech Day" event, a recurring exhibit held by the school of Engineering and Science to motivate high-school students to pursue a college degree. Many schools report more motivated freshmen [16] and higher enrollment [10] after such demonstrations. Some schools explicitly promote entrepreneurship and commercialization of student work [23]. Some even require students to purchase their own game development tools to retain the copyright to their work [21]. We also plan to incorporate these practices in the future.

IX. Complexity Considerations

The significant duration, budget, and personnel size make every game a serious software engineering project. Multi-million dollar games are usually released in just two or three years [21], [31]. They are a product of thousands of man-hours spent among hundreds of individuals [11], [23]. Our minor cannot handle projects of such scale and complexity. Below we outline a few ways to address these issues and increase student productivity.

A. Agile Methodologies

Even though students are typically excited about working on a game project, they often struggle to meet the deadlines due to over-ambition, procrastination, and poor teamwork coordination [28], [29]. Furthermore, games evolve quickly, so the code must be flexible, reusable, and maintainable [1], [31], [32]. The solution is an agile software development methodology, which promises timely delivery, especially when the requirements are unstable. It assumes collaboration and continuous refinement and it is the preferred methodology for game development projects in the industry as well as academia [7], [10], [26], [28]. We plan to incorporate it into our courses to ensure the projects are on time.

Agile methodologies are meant for mature software developers because of the frequent in-person interaction. Students lack such experience, but our class schedule can alleviate this problem. The Monday-Wednesday-Friday class schedule would ensure that students can attend frequent meetings, some of which may be supervised [28]. The team size of two-three students of approximately the same age and graduate standing is optimal for a personal feel and little coordination [21], [29]. Finally, peer reviews can motivate student collaboration and promote fair work distribution [10]. We plan to adopt these techniques to facilitate the adoption of agile methodology for student projects.

B. Mobile Gaming

Another way to simplify the inherent complexity of game development is to specialize in a particular platform. For example, mobile gaming is significantly smaller and simpler because it does not require substantial resources that may not be available on a mobile device [14], [23]. Virtually every student has a cellphone and half of them regularly play mobile games like Tetris, Pac-Man, and Solitaire [14]. These games have few simple rules and do not require a significant time commitment. Developing such games is exciting because students can quickly create a game, play it on their phone, and show it off to their friends [14]. Few engineering disciplines could boast such quick turnaround time and practicality. Mobile games would be perfect for student projects.

C. Scripting Languages

Game studios are primarily concerned with building more interesting games, but not necessarily solving engineering problems. Therefore, existing solutions and common low-level functionality developed by expert programmers is often reused. The various game engines, software architectures, design patterns, and generative programming techniques simplify the development of large and complex games [1], [8], [17], [20], [24], [33]. The programming courses should give students plenty of coding experience, so they should feel comfortable with any game development framework. Scripting languages abstract implementation details and let students focus on game design. Since they only have to write the unique parts of the game, students can be more productive [4], [17]. The existing artifacts take care of the technical challenges and allow developers to be more creative. We will encourage such creativity in our minor.

X. Future Improvement

Our current curriculum is directed towards game programming, a coding intensive occupation similar to computer science. It covers basic mathematics, physics, modeling, and artificial intelligence. Even though our courses briefly cover all of the related topics, a more in-depth insight into physics, computer graphics, and basic networking would also be beneficial [21]. A few dedicated courses could cover these topics in sufficient detail. In the future, basic animation and 3D modeling classes may be recommended to students who would like to have a more well-rounded game development education. This version of the curriculum would be geared towards liberal arts students, with more emphasis on asset creation. Such complementary versions of the minor would focus on the two respective areas of game production, design and development.

XI. Research Opportunities

As with many other software engineering fields, game development has plenty of open research problems. For instance, modern games exhibit an increasingly heavier dependence on static assets, e.g., textures, sounds, and animations [18]. This problem grows with each new game console enticing a new generation of gamers with more detailed graphics and sound. Eventually, no amount of details will be able to attract consumers, who demand longer and more immersive experiences.

As a result, the game itself must entertain and amaze the players. Currently, the story within a typical game unravels through a scripted sequence of scenes that change according to the player actions [18]. The structured approach has a finite amount of considerably distinct game plays. Artificial intelligence techniques, on the other hand, can create an interesting story on the fly. In fact, interactive drama is considered the pinnacle of game design [18]. The story within the game follows the rules of the virtual world, yet it has no predetermined course. A player can exercise free will and experience the consequences of his/her actions. The dynamically generated story is always different, so the players get to enjoy a longer gameplay. Students taking this minor may become interested in further artificial intelligence research specifically for such applications.

XII. Conclusions

Students enjoy playing and making computer games. Many of them fantasize about creating the next game that would define an entire generation, e.g., Super Mario, Tetris, or PacMan. For a long time such fantasies remained unattainable for young people who did not want to commit multiple years to software development. Fortunately, modern technological advances allow students to create relatively complex games within just a few semesters. Because much of the serious programming has already been done, only basic understanding of software development is required. This makes our minor accessible for a variety of students. However, software engineering majors may also find this minor interesting for the research opportunities that span beyond game applications. This minor will give students a chance to make the game they always wanted to play.

References

[1] A. Ampatzoglou and A. Chatzigeorgiou, "Evaluation of object-oriented design patterns in game development," *Inf. Softw. Technol.*, vol. 49, pp. 445–454, May 2007.

[2] L. Argent, R. Fajardo, S. Gjertson, S. T. Leutenegger, M. A. Lopez, and J. Rutenbeck, "Building a game development program," *IEEE Computer*, vol. 39, pp. 52–60, 2006.

[3] A. Baytak and S. M. Land, "A case study of educational game design by kids and for kids," *Procedia Social and Behavioral Sciences*, vol. 2, no. 2, pp. 5242–5246, 2010.

[4] B. Clark, J. Rosenberg, T. Smith, S. Steiner, S. Wallace, and G. Orr, "Game development courses in the computer science curriculum," *Journal of Computing Sciences in Colleges*, vol. 23, pp. 65–66, December 2007.

[5] K. Daimi, N. Al-Holou, and U. Mohammed, "An interdisciplinary minor in computer and network security," in *Security and Management*, 2010, pp. 505–510.

[6] K. Daimi, G. Grabowski, and K. Snyder, "A collaborative bioinformatics minor design," in *FECS*, 2009, pp. 341–345.

[7] P. J. Diefenbach, "Practical game design and development pedagogy," *IEEE Computer Graphics and Applications*, vol. 31, pp. 84–88, 2011.

[8] P. Gestwicki and F.-S. Sun, "Teaching design patterns through computer game development," *Journal on Educational Resources in Computing (JERIC)*, vol. 8, pp. 2:1–2:22, March 2008.

[9] T. Goulding, "Complex game development Throughout the college curriculum," *SIGCSE Bull.*, vol. 40, pp. 68–71, November 2008.

[10] T. Goulding and R. DiTrolio, "Complex game development by freshman computer science majors," *SIGCSE Bull.*, vol. 39, pp. 92–99, December 2007.

[11] M. Guimaraes and M. Murray, "An exploratory overview of teaching computer game development," *Journal of Computing Sciences in Colleges*, vol. 24, pp. 144–149, October 2008.

[12] M. A. Holliday, "Incremental game development in an introductory programming course," in *Proceedings of the 33rd annual on Southeast regional conference*, ser. ACM-SE 33. New York, NY, USA: ACM, 1995, pp. 170–175.

[13] F. Ke, "A case study of computer gaming for math: Engaged learning from gameplay?" *Computers & Education*, vol. 51, no. 4, pp. 1609–1620, 2008.

[14] S. Kurkovsky, "Making the case for mobile game development," *SIGCSE Bull.*, vol. 41, pp. 401–401, July 2009.

[15] S. H. Lee, G. H. Lee, H. H. Cho, D. H. Song, and S. Y. Rhew, "An empirical model of the game software development processes," in *Proceedings of the Fourth International Conference on Software Engineering Research, Management and Applications*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 371–377.

[16] K. Mikami, T. Watanabe, K. Yamaji, K. Ozawa, A. Ito, M. Kawashima, R. Takeuchi, K. Kondo, and M. Kaneko, "Construction trial of a practical education curriculum for game development by industry-university collaboration in japan," in *ACM SIGGRAPH ASIA 2009 Educators Program*, ser. SIGGRAPH ASIA '09. New York, NY, USA: ACM, 2009, pp. 7:1–7:6.

[17] P. Moreno-Ger, J. L. Sierra, I. Martínez-Ortiz, and B. Fernández-Manjón, "A documental approach to adventure game development," *Science of Computer Programming*, vol. 67, pp. 3–31, June 2007.

[18] J. Murray, I. Bogost, M. Mateas, and M. Nitsche, "Game design education: Integrating computation and culture," *Computer*, vol. 39, pp. 43–51, June 2006.

[19] D. H. Pink, *Drive: the surprising truth about what motivates us*. New York: Riverhead Books, 2009.

[20] E. M. Reyno and J. A. Carsí Cubel, "Automatic prototyping in model-driven game development," *Computers in Entertainment (CIE) - SPECIAL ISSUE: Media Arts and Games (Part II)*, vol. 7, pp. 29:1–29:9, Jun. 2009.

[21] A. D. Ritzhaupt, "Creating a game development course with limited resources: An evaluation study," *Trans. Comput. Educ.*, vol. 9, pp. 3:1–3:16, March 2009.

[22] J. D. Rocca, R. Hunike, W. Spector, and E. Zimmerman, "Forum: Game development, design and analysis curriculum," in *ACM SIGGRAPH 2002 conference abstracts and applications*, ser. SIGGRAPH '02. New York, NY, USA: ACM, 2002, pp. 16–16.

[23] N. F. Rodrigues, R. Simoes, and J. L. Vilaça, "A digital game development education project," in *VS-GAMES*, 2010, pp. 79–82.

[24] V. T. Sarinho and A. L. Apolinário, "A generative programming approach for game development," in *Proceedings of the 2009 VIII Brazilian Symposium on Games and Digital Entertainment*, ser. SBGAMES '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 83–92. [Online]. Available: http://dx.doi.org/10.1109/SBGAMES.2009.18

[25] S. Schiesel, "Video games are their major, so don't call them slackers," *New York Times*, November 2005.

[26] J. Schild, R. Walter, and M. Masuch, *ABC-Sprints : Adapting Scrum to Academic Game Development Courses*. ACM, 2010, pp. 187–194.

[27] B. Shelton and J. Scoresby, "Aligning game activity with educational goals: Following a constrained design approach to instructional computer games," *Educational Technology Research and Development*, pp. 1–26, November 2010.

[28] T. Smith, K. M. Cooper, and C. S. Longstreet, "Software engineering senior design course: Experiences with agile game development in a capstone project," in *Proceedings of the 1st International Workshop on Games and Software Engineering*, ser. GAS '11. New York, NY, USA: ACM, 2011, pp. 9–12.

[29] M. Q. Tran and R. Biddle, "Collaboration in serious game development: A case study," in *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*, ser. Future Play '08.

New York, NY, USA: ACM, 2008, pp. 49–56. [Online]. Available: http://doi.acm.org/10.1145/1496984.1496993

[30] D. P. Twitchell, K. Wiers, M. Adkins, J. K. Burgoon, and J. Jay F. Nunamaker, "StrikeCOM: A multi-player online strategy game for researching and teaching group dynamics," *Hawaii International Conference on System Sciences*, vol. 1, p. 45b, 2005.

[31] D. Volk, "How to embed a game engineering course into a computer science curriculum," in *The 2008 Conference on Future Play Research Play Share*. ACM, 2008, pp. 192–195.

[32] A. I. Wang and B. Wu, "Using game development to teach software architecture," *Int. J. Computer Games Technology*, vol. 2011, 2011.

[33] W. White, C. Koch, J. Gehrke, and A. Demers, "Better scripts, better games," *Commun. ACM*, vol. 52, no. 3, pp. 42–47, Mar. 2009.

[34] D. Williams, M. Consalvo, S. Caplan, and N. Yee, "Looking for gender: Gender roles and behaviors among online gamers," *Journal of Communication*, vol. 59, no. 4, pp. 700–725, Dec. 2009.

[35] M. Zyda, "Educating the next generation of game developers," *Computer*, vol. 39, no. 6, pp. 30–34, 2006. [Online]. Available: e

Network Security Course: A Demonstration of Project-Based Learning

Leonidas Deligiannidis, Charlie Wiseman, Mira Yun, Tom Goulding

Wentworth Institute of Technology
Department of Computer Science and Networking
550 Huntington Avenue
Boston, MA 02115, USA
{deligiannidisl, wisemanc, yunm, gouldingt}@wit.edu

ABSTRACT

Systems and Network Security is a very important topic in today's world where Information Technology is becoming part of our everyday lives. We access and interact with numerous services on the Internet that require us to exchange sensitive data. The Internet is a highly distributed system operated by many different entities and as such should not be trusted by end users. This is the reason that we need strong security measures to communicate in a secure way, privately, with other (trusted) entities. Users, whether consumers or businesses, retain no control how their information is routed among the many networks that comprise the Internet. Therefore, there is a strong need for cryptographic protocols to authenticate, gain trust, and establish a secure channel for exchanging data.

We are enriching our curriculum by offering several Computer Science advanced electives each semester (Fall, Spring and Summer). Students have the option of taking advanced elective courses that fit their interest. One of our newer courses, Cryptography and Network Security, addresses these issues of communicating securely over inherently insecure channels such as the Internet. In this paper we present the material taught in this class as well as the assignments and demonstrations utilized in the class room to motivate the students in learning.

Keywords - Network Security, Education

1. INTRODUCTION

Wentworth Institute of Technology is a baccalaureate degree granting institution. The Department of Computer Science and Networking has been in existence since the early nineteen eighties. Our department offers two degrees: one leading to a Bachelor of Science in Computer Science and another leading to a Bachelor of Science in Computer Networking. While the computer science degree program is a typical, traditional CS program, the computer networking program is more hands-on and exposes students to a wide skill-set in networking, computer science, and management. Both majors select four advanced Computer Science electives from a pool of 10-12 courses during their Junior and Senior years. One of the courses now offered regularly is the Cryptography and Network Security course. This course has been very successful, with excellent student engagement and feedback [21]. The material presented is difficult, but demonstrations using numerical and/or graphical examples engage the students. In fact, many of the students work on the more difficult extra credit assignments in addition to the regular course assignments. Many students ultimately make a carrier change because of this course.

The book used in the class is “Cryptography and Network Security: Principles and Practice”, by William Stallings, Prentice Hall. This is an excellent book and has been adopted by many professors offering similar courses at the college level. Most programming assignments are done on students' laptop computers using the Java language; a few of the assignments are done in C#. There are six programming assignments where we use symmetric encryption, digital signatures, and digital certificates, and two more assignments where we implement RSA public key cryptography from the ground up. There are two exams during the semester and a final exam at the end of the semester. Because of the popularity of the topics covered in class, we offer this course during the Summer semester every year and sometimes during the Spring as well.

In the rest of the paper we will focus on the programming assignments, demonstrations, and future material we plan on incorporating into this course.

2. BACKGROUND

It is suggested that hands-on, investigative teaching with associated exercises improves the learning performance of the students [17]. It is also shown that engaging students in hands-on exercises promotes active learning and helps students develop critical thinking skills [19][20] more than simply covering lecture material in class and leaving students with many unanswered questions. This improves their learning performance and as a result increases the likelihood of programs to retain their students [3][7][11][16][18][19][20][23]. A three year study was performed to determine why STEM majors switch to non-STEM majors [27]. They found that students switch majors because of lack of interest, teaching methodology ineffectiveness, and because they felt overwhelmed with the curriculum demands. Successful completion of the introductory courses for the first year is crucial in retaining students in a program and most lecture courses are notoriously ineffective in engaging students [6].

DefEx is a set of hands-on cyber-defense exercises that focuses on undergraduate development through understanding and problem solving related to security [28]. These exercises include code and system level hardening, problem detection, digital forensics, wireless access point security**,** cross-site scripting, command and SQL injection, file uploading, and a wireless access point treasure hunt game based on wardriving that requires students to utilize all their skills from throughout the course.

3. COURSE WORK

In this course, the students are given two exams during the semester and a final at the end of the semester. The students are also required to attend every lecture and complete six assignments. Extra credit assignments are provided for students who are interested in learning more. In the next section we describe the major programming assignments and demonstrations we perform to motivate our students.

3.1. Assignment in Symmetric Encryption

The Data Encryption Standard (DES) algorithm was the most widely used symmetric cryptosystem in the world. It is a block cipher that was selected by the National Bureau of Standards as an official Federal Information Processing Standard for the United States in 1976. DES is now considered to be insecure and has already been superseded by the Advanced Encryption Standard (AES) [30].

The DES algorithm takes a 64 bit plaintext block and a 64 bit long secret key and transforms them into a 64 bit long ciphertext block. Decryption must be performed using the same key as was used for encryption and the same algorithm in reverse to reproduce the original plaintext block.

The students learn about symmetric encryption algorithms and they complete one assignment where they use DES to encrypt and decrypt a file. In another assignment the students compare the performance of DES and AES by plotting how fast different size files are encrypted and decrypted.

3.2. Assignment in Hash Functions

Cryptographic hash functions play an important role in modern communication technology. The input to a hash function is a file or stream of any size and the output is a fixed size digital representation of the file that is normally less than 1KB and serves as the fingerprint of the original file (often called the message digest). It is impossible to reconstruct the original file if you only have the fingerprint. Moreover, changing a single bit of information in the input would result in a significantly different fingerprint. These algorithms are designed to avoid collision. In other words, it is very unlikely for two messages M and M' to produce the same fingerprint using cryptographic hash function H: $H(M)\ != H(M')$. Many cryptographic hash functions are based on the so called MD4 algorithm initially proposed in [24], and they have received the greatest attention.

Students write a program to compute the message digest given different input streams. Then they modify the input in order to produce substantially different digests. The students are challenged to find two inputs that produce the same message digest. Then, we demonstrate how they can break MD5 using the techniques described in [12][13]. Specifically, we produce two different executable files that have significantly different purposes, yet whose MD5 digests are identical. This shows that it is possible to have two different files with the same MD5 message digest and that using MD5 hashing to verify file downloads is not safe.

3.3. Extra Credit Assignment on Steganography

Steganography is a technique to conceal information so that only the communicating parties know about the existence of the information in any form [15]**.** Steganography uses a Cover Image which is an image that will have information embedded in it. Then there is the actual information itself that could be anything from plain ASCII text or another image to pdf files, sound files, etc. After the information has been embedded into the Cover Image using any of the Steganography techniques (such as Least Significant Bit Insertion) a Stego Image [4] is obtained. It is also possible to encrypt the information before embedding it, thereby adding another level of security [2].

We demonstrate two applications that use steganography. Both of them are written in Java and are accessible via webstart technology: HAE (Hide at End) [9] and Stego [29]. HAE demonstrates how one can hide any document of any size at the end of a PNG picture. The output looks exactly like the input but the file is bigger in size. This tool relies on the fact that PNG viewers ignore everything after the end-tag of the PNG image. So, right after the end-tag any other file or information can be appended. Students can use HAE to interactively hide pictures inside other pictures, retrieve them, and check file sizes.

Stego, shown in Figure 1, is an interactive graphical interface that allows one to hide text or images into other images, retrieve the secret text or image, and display them all in the same window. Stego uses the Least Significant Bit insertion technique. This means that every bit of the secret is inserted in the least significant bit of a byte that represents either the Red, Green, or Blue color of the RGB pixel of the original image (the Cover Image). Stego also allows the students to hide more than one bit per byte. The effect can be seen instantly on the graphical interface. A user can also save each image on the system to inspect the file size, run a hash function on them, etc, or even load saved Stego Images. This feature is provided from the pull down menu. Figure 2 shows how the Stego Image looks when the 5 least significant bits per byte are used to hide the secret image.

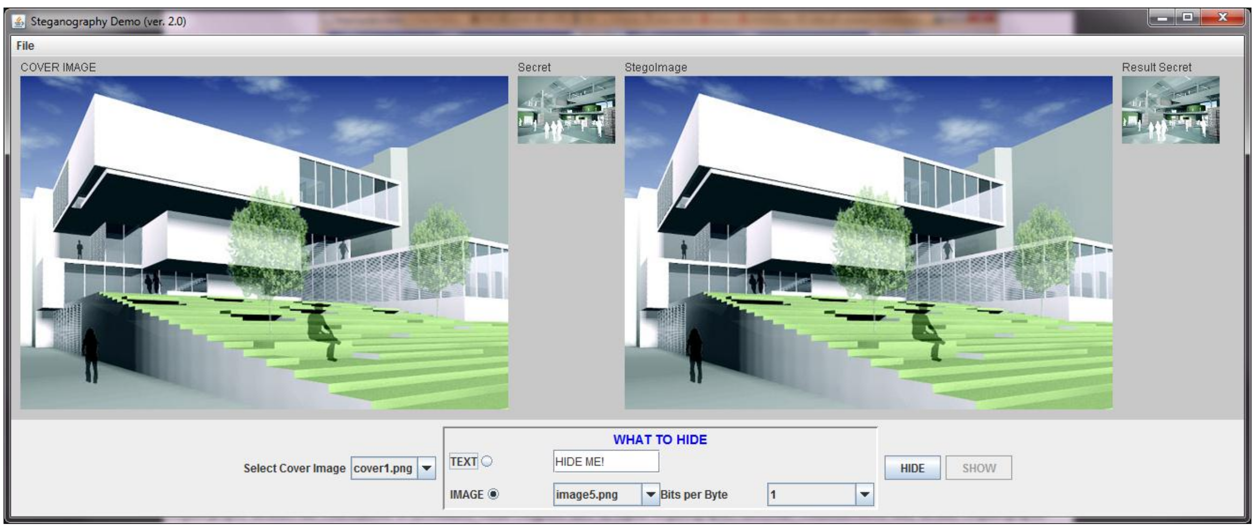


Figure 1. Graphical interface of the "Stego" steganography demonstration application. Four images at the top, from left to right: the Cover Image, the secret image, the Stego Image (output image containing the secret image), and the result secret image retrieved from the Stego Image. At the bottom, the user first selects a cover image, then the user selects what to hide (text or image) and either types in the text or selects a secret image. Then, the user selects the number of least significant bits to be used to hide the secret image. Finally the user can click on the "HIDE" button to create the Stego Image, and "SHOW" to retrieve the secret image hidden in the Stego Image.

Figure 2. A Cover Image (left) and a Stego Image (right), side-by-side where the 5 least significant bits per byte are used to hide a secret image. Visible alterations can be observed at the top of the Stego Image.

For extra credit, we give the source code of Stego to the students with one of the functions is removed. The function is used to retrieve a secret image from the Stego Image and must be replaced by the students.

3.4. Assignment in a Key Exchange Algorithm

One of the questions students have with symmetric encryption algorithms is key distribution. One of the simplest algorithms that we illustrate and describe with real numbers is Diffie-Helman. The students write a program to show how a "key" can be distributed to two entities without actually transmitting the key itself. The students have the option to work on an extra credit component. The extra credit is to use the Diffie-Helman key exchange algorithm to exchange keys in a secure client/server "chatting" application. They exchange a key and then they use a symmetric algorithm to establish a secure connection and transmit text and images over the Internet. Later in the semester, after we cover asymmetric encryption and we begin talking about PGP, they can modify their program so that the key exchange is implemented using RSA.

3.5. Assignment in Asymmetric Encryption

RSA [25] and elliptic curve [14] are two popular public key cryptosystems. Public key systems are the standard choice for sending secure information over an insecure channel or network connection. For example, a customer wants to send their credit card information to a web site. There is no way for the costumer and the web vendor to share a secret key as in symmetric encryption systems without the risk of a third party overhearing the secret key. Instead, public key systems use two keys: a public key that is shared with everyone and a private key that is kept secret by the receiving party. The public key and the private key share a mathematical link, but it is not possible (or is at least computational very difficult) to derive either key from the other.

RSA is one of the most commonly used public key cryptosystems in the Internet today. It is based on the premise that factoring very large numbers is difficult. "Large" numbers in modern practice are 1024-bit or 2048-bit numbers. That is, they are numbers on the order of 2^{1023} ($\sim 10^{307}$) or 2^{2047} ($\sim 10^{615}$). One of these large numbers is chosen to serve as the modulus for encryption and decryption and then a public key exponent and private key exponent are chosen following certain number theory principles. This is the general procedure for generating the associated public and private keys:

1) Choose two prime numbers: p and q.
2) Compute the modulus n: $n = pq$.
3) Compute Euler's totient function $\varphi(n)$: $\varphi(n) = (p-1)(q-1)$.
4) Choose the public key exponent e, such that $1 < e < \varphi(n)$ and e is co-prime with $\varphi(n)$.
5) Compute the private key exponent d, where $(de) \bmod \varphi(n) = 1$.

Step 4 is done easily by selecting a random prime number less than $\varphi(n)$ and then checking that it is not a factor of $\varphi(n)$. Step 5 is somewhat more computationally intensive, but can be computed with the Extended Euclidean algorithm. When this is completed, the public key is the pair (n, e) and the private key is the pair (n, d). Going back to the credit card example above, the web vendor would send the customer their public key so that the customer can encrypt the credit card information. The private key is kept secret by the vendor. The customer uses standard encoding and padding schemes to produce the message, m, to be sent from the credit card information. To encrypt the message and produce the ciphertext, c, the customer uses this formula:

6) $c = m^e \bmod n$

The encrypted message c is then sent over the Internet and received by the web vendor who decrypts the message with this formula:

7) $m = c^d \bmod n$

There are a few different approaches to an RSA assignment. There are three main components: generating the keys, encrypting and decrypting data, and sending information over a network. The full assignment has students completing all three phases to send a single encrypted file over the network. They first generate the public and private keys, and then write a client/server application that allows a user to select a file and transmit it from the client to the server. The server stores the keys and transmits the public key to the client. The client encrypts the file, sends it, and then the server decrypts it.

The first and last steps can be omitted due to time constraints if necessary. For example, instead of having students write code to generate the keys they could compute one key pair by hand and use that directly. Alternatively, they could use existing tools such as OpenSSL [22] to generate the RSA keys. The file transfer could also be optional. Students could simply encrypt the file on the local system with one program and decrypt it with another. This would alleviate the need to have code that actually transmits the file, but loses the key exchange between two systems.

3.6. Demonstrations

Every student that takes this class is interested in knowing how a virus is written and how they work. We demonstrate to the students how four different Keyloggers work. We explain the code, which is written in C, and they can experiment with the programs. We also demonstrate two viruses. One of them attacks the browser and the other attacks the operating system (of course we only simulate the deletion and modification of system files). We also demonstrate and explain how to copy DVDs and how they are protected. The code given is written in Java. Another demonstration involves sending "fake" email messages (messages where the sender information is modified). Throughout the course, we have the opportunity to talk about ethical issues. However, the students should learn about these techniques so that they can protect themselves in the real world.

4. CONCLUSIONS AND FUTURE WORK

We provide students with solid security fundamentals and hands-on exercises that are focused on wired networks through our course on Cryptography and Network Security. As a next step, we will incorporate wireless security issues into our course.

Wireless enabled devices such as laptops, smartphones, and tablets are becoming increasingly affordable and accessible. Wireless technology has become a crucial component in computer networking and security technology [8][32]. Despite the popularity of the topic, courses on wireless and security are not frequently offered at the undergraduate level; this is mainly due to the fact that it is still a relatively new topic that requires greater research. In our course, we will focus on security issues relating to current Wi-Fi (802.11) systems.

The first generation of the 802.11 standard included security protection in the form of Wired Equivalent Privacy (WEP) [1]. WEP was found to be vulnerable to various statistical weaknesses, especially in the encryption algorithm it employed to scramble data passed over the WLAN [5][26]. While attempts were made to correct the problem, it is still a relatively simple procedure to crack the protocol. Essentially, one can pull the password right out of the air. In response, the Wi-Fi Alliance stepped up to the challenge and created an interim "standard" called Wi-Fi Protected Access (WPA). However, WPA's pre-shared key (PSK) mode is also crackable due to a flaw that exists in the authentication procedure [10]. For the most part, if there is a password and a user is involved then the system is flawed. This fact, combined with the reality that most users select poor passwords, provides an opportunity that can be exploited.

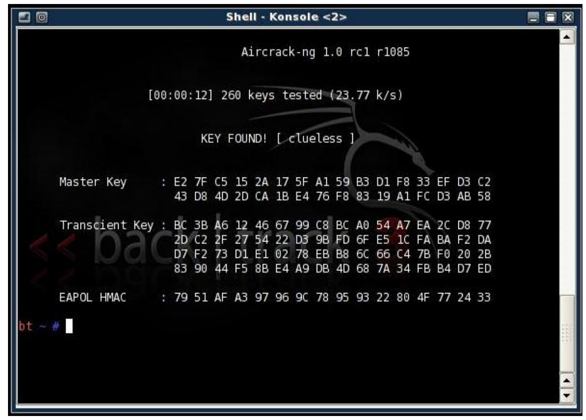


Figure 3. Cracking output of Aircrack-ng.

In order to provide hands-on learning experiences in wireless security, we will use devices and tools that are widely used in the real market. Linksys WRT54G series access points (802.11g broadband routers) and OpenWRT [31] firmware will be used for deploying and managing Wi-Fi networks. As shown in Figures 3 and 4, we will explore the vulnerabilities of the WEP and WPA-PSK protocols with key cracking tools and network protocol analyzers.

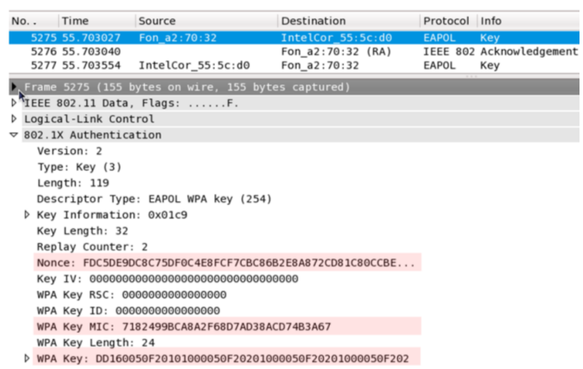


Figure 4. Wireshark screenshot of a WPA authentication frame.

Students will learn two WEP key cracking schemes: passive traffic collection and active injection. As a very basic approach, minimal WEP encrypted data will be passively collected. Approximately 20,000 to 50,000 packets are required to crack a 64 bit WEP key. From the collected data packets, we will use weak Initialization Vectors (IVs) to crack the WEP key as defined in the 802.11 standard. However this passive approach requires large amounts of data traffic and collecting time. As a more efficient technique, we can inject ARP request packets through fake authentication and association. The only known effective way to crack WPA-PSK is to force a re-authentication of a valid client. By forcing the connected client to disconnect, we capture the re-connect and authentication packets (i.e. the four-way-handshake) as shown in Figure 4.

Although students are mainly excited to learn about hacking and cracking skills, we believe that this course will allow students to be more engaged in the topic of wireless security. It will become an even better starting point for students to develop their interest in the topic of network security.

REFERENCES

[1] "IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Std 802.11-1997 , vol., no., pp.i-445, 1997.

[2] Akshay Choche and Hamid R. Arabnia, “A Methodology to Conceal QR Codes for Security Applications”. Proceedings of the International Conference on Information and Knowledge Engineering (IKE'11): July 2011, USA.

[3] Anderson-Rowland, M. R.; Blaisdell, S., Fletcher, S., Fussel, P., Jordan, C., McCarthey, M. A., Reyes, M. A., White, M., “Comprehensive programmatic approach to recruitment and retention in the college of engineering and applied sciences”, In Ptoc. Of the 29th Annual Frontiers in Education Conference, Vol.(1), 1999.

[4] Birgit Ptzmann, Information Hiding Terminology, First Workshop of Information Hiding Proceedings, Cambridge, U.K. May 30 - June 1, 1996. Lecture Notes in Computer Science, Vol.1174, pp 347-350. Springer-Verlag (1996).

[5] Boland, H.; Mousavi, H.; "Security issues of the IEEE 802.11b wireless LAN," Electrical and Computer Engineering, 2004. Canadian Conference on, vol.1, no., pp. 333- 336 Vol.1, 2-5 May 2004.

[6] Carol A. Twigg, “Using Asynchronous Learning in Redesign: Reaching and Retaining the at Risk Student”, JALN Vol.(8), Issue 1, Feb. 2004 pp 7-15.

[7] Courter, S. S.; Millar, S. B.; Syons, L. “From the students’ point of view: Experiences in a freshman engineering design course” Journal of engineering education, Vol.(87), no(3), pp 283-287, Jul 1998.

[8] D Ma; Tsudik, G.; , "Security and privacy in emerging wireless networks [Invited Paper]," Wireless Communications, IEEE , vol.17, no.5, pp.12-21, October 2010.

[9] HAE website http://faculty.cs.wit.edu/~ldeligia/PROJECTS/HAE/index.html

[10] Hal Berghel and Jacob Uecker, “WiFi attack vectors”. Commun. ACM 48, 8 (August 2005)

[11] Hong Wang, “From C to Python: Impact on Retention and Student Performance”, In Proc. of The 2008 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'09), pp.170-174, July 13-16 2009 Las Vegas NV, USA.

[12] http://www.codeproject.com/Articles/11643/Exploiting-MD5-collisions-in-C

[13] http://www.mscs.dal.ca/~selinger/md5collision/

[14] I. Black, G. Seroussi and N. Smart. ”Elliptic Curves in Cryptography”, Cambridge University Press, 1999.

[15] Investigator's Guide to Steganography Gregory Kipper Auerbach Publications 2004. Print ISBN: 978-0-8493-2433-8 eBook ISBN: 978-0-203-50476-5.

[16] Jacqueline C. Tanaka, Larry D. Gladney, “Strategies for Recruiting and Retaining Minorities in Physics and Biophysics”, Biophysical Journal Vol.(65), Jul. 1993 pp552-558

[17] Janet K. Yates, Madeleine Voss, Kuei-wu Tsai, “Creating Awareness about Engineering Careers: Innovative Recruitment and Retention Initiatives” 29th ASEE/IEEE Frontiers in Education Conference, Vol. 3, Nov 1999, San Juan Puerto Rico.

[18] Jerry M. Hatfield, John T. Tester, “LEGO Plus”, Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition, 2005.

[19] John T. Tester, David Scott, Jerry Hatfield, Rand Decker, Fonda Swimmer, “Developing Recruitment and Retention Strategies through "Design4Practice" Curriculum Enhancements”, 34th ASEE/IEEE Frontiers in Education Conference, Oct. 2004, Savannah, GA, USA.

[20] John T. Tester, Jerry Hatfield; “The Design4Practice Sophomore Design Course: Adapting to a Changing Academic Environment,” ASEE Annual Conference Proceedings, June 2005.

[21] Leonidas Deligiannidis, "Classroom Experiences: Disallowing Laptops during Lectures Improves Student Learning". In Proc. of The 2011 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'11), pp.217-221, July 18-21 2011, Las Vegas NV, USA.

[22] OpenSSL website. http://www.openssl.org/

[23] Parcover, J. A., McCuen, R. H., “Discovery Approach to Teaching Engineering Design”,

Journal of Professional Issues in Engineering Education and Practice, pp 236-241, Oct. 1995.

[24] R. L. Rivest, The MD4 message-digest algorithm, Crypto, LNCS 537 (1991) 303-311.

[25] R. Rivest, A. Shamir and L. Adleman, ”A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, Communications of the ACM, 21(2), 1978.

[26] Scott R. Fluhrer, Itsik Mantin, and Adi Shamir, “Weaknesses in the Key Scheduling Algorithm of RC4”, In Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography (SAC '01), Serge Vaudenay and Amr M. Youssef (Eds.). Springer-Verlag, London, UK, 1-24.

[27] Seymour, E. & Hewitt, N.M., “Talking about leaving: Why undergraduates leave the sciences”. Boulder, Co: Westview Press 1997.

[28] Sonja M. Glumich and Brian A. Kropa," DefEX: Hands-On Cyber Defense Exercises for Undergraduate Students" In Proc. of the 2011 Int. Conf. on Security and Management (SAM'11), July 2011, USA.

[29] Stego website http://faculty.cs.wit.edu/~ldeligia/PROJECTS/Stego/index.html

[30] U.S. Department of Commerce/National Institute of Standards and Technology, FIPS PUB 197, Advanced Encryption Standard (AES), Federal Information Processing Standards Publication, 2001. Available at http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[31] Website of OpenWRT https://openwrt.org/

[32] Yi-Sheng Shiu; Shih Yu Chang; Hsiao-Chun Wu; Huang, S.C.-H.; Hsiao-Hwa Chen; , "Physical layer security in wireless networks: a tutorial," Wireless Communications, IEEE , vol.18, no.2, pp.66-74, April 2011.

ProNIFA: A Tool Supporting Semi-Automatized Formative Assessment in STEM Teaching

Michael D. Kickmeier-Rust and Dietrich Albert
Cognitive Science Section, Knowledge Management Institute
Graz University of Technology, Graz, Austria

Abstract - *There is no doubt that modern technologies significantly changed the educational needs and practice; the impact of the continuously evolving technology will increase the demands of teachers and learners. NEXT-TELL is a multi-disciplinary European project that attempts to develop a European infrastructure to facilitate and broaden the use of modern information and communication technologies in order to tackle the educational challenges of today and tomorrow. A significant aspect in this context is a strong focus on modern, formative, non-numerical approaches to the assessment of learning in particular in STEM areas. In this paper we outline the challenges we see in the modern classrooms and we provide an introduction to a formal, cognitive theory that supports modern ideas of assessment and student appraisal. In addition, we present a software tool, named ProNIFA, that enables teachers and learners to utilize the strength of the formative assessment in an effective and semi-automatized way.*

Keywords: 21st century education; technology use in classrooms; formative, non-numerical, probabilistic assessment; Competence-based Knowledge Space Theory

1 21st Century Education

21st century education clearly is a big buzzword in today's media. The new millennium is accompanied by substantial technological evolutions; we became a highly diverse, globalized, complex, real-time media- knowledge- information- and learning society. Since the 1990s, the progress of media and technology was breath taking; during these one or two decades, we were facing the rise of a serious and broad use of computers at home (although the development started earlier, of course), the rise of the internet and how it revolutionized our society. We faced the spread of mobile phones and their evolution from telephones to omnipresent computer and communication devices; we see spread of mp3, twitch speed computer games and TV shows. We saw how our world got closer by changing the bridges over continents and oceans from 56k wires to hyper speed fiber glass networks.

But what does this mean for educational systems and the way our children learn and what they learn? Today's kindergarten kids will retire in 2070. Facing the pace of technological and societal changes and demands, we cannot predict what knowledge will be required in such a "far" future. But we are in charge to equip our children with the abilities and backgrounds to survive in that world. Our students are also facing many important emerging issues such as global warming, famine, poverty, health issues, a global population explosion and other environmental and social issues. These issues lead to a need for students to be able to communicate, function and create change personally, socially, economically and politically on local, national and global levels.

Formative assessment, defined as a bidirectional process between teacher and student to enhance, recognize, and respond to the learning, is one vital aspect in addressing those challenges Formative assessment is considered a promising approach to enable 21st century teaching since it potentially promotes self-reflection and self-directed learning processes and, more importantly, it facilitates the integration of new subject-specific knowledge into the student's existing knowledge network. It also helps adapting the teacher the educational processes to the individual needs and, therefore, making formal education more effective and also more enjoyable. At the moment, however, formative assessment is a time consuming interactive teacher-student communication process. The teacher must carefully monitor problem/task solution processes of each individual student, the teacher must understand how individual students think and learn, and the must help them to overcome conceptual difficulties, again on an individual basis. Considering today's situation and the daily routines in schools this desirable feedback process oftentimes drowns in limited time and resource context conditions. In this paper we want to introduce NEXT-TELL (www.next-tell.eu), a European initiative addressing the challenges of future school education. The major aim of this project is to support teachers in using modern technologies in the classes, to link various technologies with each other, and to benefit from synergies. NEXT-TELL also aim at supporting formative assessment processes and the aligned teaching with smart planning, assessment, and teaching technologies. Secondly, we want to introduce a formal, learning theoretic approach to non-numerical probabilistic assessment to support a formative appraisal of students and ideas of evidence-centered appraisal.

2 Numerical vs Non-Numerical Test Theories

Presently, western school systems are largely characterized by described the broad and rich spectrum of students' abilities and knowledge by single (numerical) value, the grade. This holds true on a small scale (e.g., on the level of tests or single examinations) and this holds true on a large scale (e.g., on the level of a final grade for a course). This approach, however, cannot express what students really and exactly can/know and what they cannot do/do not know satisfyingly. A good example for the weakness of the approach is the I.Q. (intelligence quotient), which attempts to characterize all the various abilities, strength and weaknesses of a person in many categories and disciplines (math, language, cognition, memory, etc.) with a single numerical value, lately.

As summarized by Falmagne, Cosyn, Doignon, and Thiéry [1], the origin of this popular test theoretical approach lies in the developments in 19th century physics. The frontiers between natural sciences, physics, psychology, and medicine we much more blurred as they are today and disciplines like "anthropometry" occurred, the "art of measuring the physical and mental faculties of human beings". Prominent proponents were Francis Galton, William Kelvin, or Carl Pearson (Figure 1). The predominant tenor was, if you cannot measure it, it is not science. Kelvin, for example, said "If you can't assign an exact numerical value, express it in numbers, your knowledge is of a meager and unsatisfactory kind".

The problem of this "charming" approach gets very clear by the following example (taken from [1]). Imagine an athletic quotient (A.Q.) and imagine 3 distinct sportsmen, the basketball player Michael Jordan, the tennis player Roger Federer, and the golf player Tiger Woods. All of them are, apparently, excellent athletes and deserve a high A.Q. but they are very different in their individual abilities. Depending on which scale one would take to measure the A.Q. (maybe the jumping height, maybe stamina?), the three sportsmen would receive very different appraisals. Now, an obvious response would be to compute a quotient of all the three men's main strength. But – what if another sportsmen is added, e.g., a long distance runner, a downhill racer, a chess player? For each distinct sportsman another measuring dimension could be added. And in the end, the resulting quotient would be able to precisely describe the real characteristics and strength of each individual sportsman.

The same problem occurs for formative assessment and the ECAAD approach, specifically in a limited resource classroom scenario. An approach for a fairer and more precise appraisal comes with ideas of non-numerical test theories. With the rise of mathematical combinatorics and with the rise of powerful computer technologies the more demanding approaches to precisely describe the various abilities of a student along multiple dimensions got in the focus for researchers and educators. In parts, the basic idea found its way into class rooms, for example, when students are described in narrative text instead of a single grade in the report card.

The approach we want to utilize for NEXT-TELL – in form of smart software services – is based on the probabilistic, non-numerical method of Competence-based Knowledge Space Theory (CbKST).

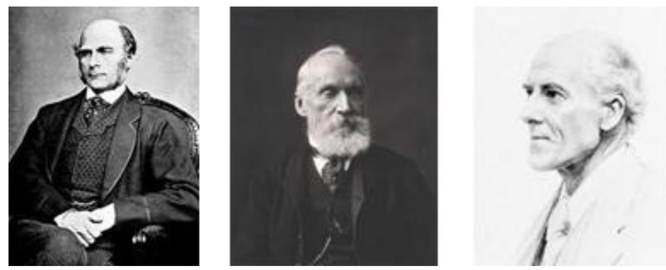


Figure 1. Proponents of "Anthropometry". From left to right: Sir Francis Galton, William Kelvin, and Carl Pearson.

2.1 The structure of Knowledge

CbKST originates from KST established by Jean-Paul Doignon and Jean-Claude Falmagne [2, 3], which is a well-elaborated set-theoretic framework for addressing the relations among problems (e.g., test items). It provides a basis for structuring a domain of knowledge and for representing the knowledge based on prerequisite relations (see Figure 2 for an example). While KST focuses only on performance (the behavior; for example, solving a test item), CbKST introduces a separation of observable performance and latent, unobservable competences, which determine the performance.

An empirically well-validated approach to CbKST was introduced by Klaus Korossy (1997, 1999); basically, the idea of the Competence-Performance Approach (CPA) is to

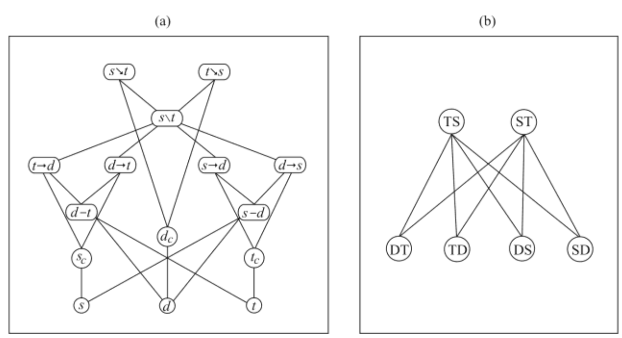


Figure 2. Panel (a) illustrates a prerequisite relation among a set of atomic competencies; panel (b) illustrates the prerequisite relation among related test items, induced by the competence structure. The example stems from learning the relations about time, distance, and speed (cf. [4]).

assume a finite set of more or less atomic competences C = {a, b, c, d, …} (in the sense of some well-defined, small scale descriptions of some sort of aptitude, ability, knowledge, or skill) and a prerequisite relation between those competences. A prerequisite relation states that competence a (e.g., to multiply two positive integers) is a prerequisite to acquire another competence b (e.g., to divide two positive integers). If a person possesses competence b, we can assume that the person also possess competence a. To account for the fact that more than one set of competences can be a prerequisite for another competence (e.g., competence a or competence b are a prerequisite for acquiring competence c), prerequisite functions have been introduced, relying on and/or-type relations. A person's competence state is described by a subset of competences of C, for example {a, b}. Due to the prerequisite relations between the competences, not all subsets of competences are possible competence states. To give an example, imagine four competences from the domain of basic algebra, the abilities to add, subtract, multiply, and divide numbers. Given four competences, the set of all possible knowledge states is 24. If we assume that the competences to add, subtract, and multiply numbers are prerequisites for the competence to divide numbers, not all of the 16 competence states are plausible. For example, it is highly unlikely that a child has the competence to divide numbers but not to add numbers. The collection of possible competence states corresponding to a prerequisite relation is called competence structure. Such competence structure also singles out different learning paths for moving from the naïve state { } (having no competences of a domain) to the state of possessing all of a domain's competences C.

So far, the structural model focuses on latent, unobservable competencies. By utilizing interpretation and representation functions the latent competences are mapped to a set of tasks (or test items) Q = {p, q, r, s, …} relevant for a given domain. Through the aforementioned example, Q might include a number of addition, subtraction, multiplication, and division tasks. The interpretation function assigns a set of competences required to solve a task to each of the problems in Q. Vice versa, by utilizing a representation function, a set of problems is assigned to each competence state, which can be mastered in this state. This assignment induces a performance structure, which is the collection of all possible performance states. Due to these functions, latent competences and observable performance can be separated and no one-to-one mapping is required. Moreover, learning or development is not seen as a linear course; equal for all children, rather, development follows one of a set of individual learning or developmental paths. More in-depth introductions to CbKST can be found, for example, in Doignon & Falmagne (1999).

In the context of NEXT-TELL, this approach to the modeling of the knowledge domain in the first instance and of meaningful learning sequencing in a second instance provides some key advantages:

One the one hand, we have a formal and computable method to analyze and define domains (i.e., the subject matter) on the level of (more or less) atomic entities of knowledge or ability – what term competency here. This formal nature in combination with a set of available tools and procedures enables to teacher to develop an individual structure of the subject matter, an individual curriculum, that involves the information and task from multiple sources (e.g., textbooks, books, websites, multimedia sources, peer activities, etc.). The flexible nature of the approach allow an easy mapping to other instances of NEXT-TELL, for example, the KSA modeling, the ECAAD methodology, software tools, and so forth. The prerequisite relations / functions induce competence structures which may serve as a means of planning, evaluating, and visualizing learning on the basis of admissible learning paths (cf. Figure 3). The knots of this Hasse diagram indicate meaningful competence states of a student while the edges indicate admissible transitions from one competence state to another by acquiring another competency.

- In terms of learning activity planning, a teacher may identify the starting point of an educational episode (e.g., a semester) by selecting the corresponding competence states in the figure and likewise she may identify the learning goal in a very precise manner.

- In terms of assessment planning, a teacher may select or construct items the perfectly suit the learning progress and competence states of the students.

- The Hasse diagram and associated graphs may be used to visualize the knowledge of students in a non-numerical and precise manner. This opens (a) a link to the use of open learner models (OLM) and it may serve (b) the ideas of negotiating and communicating appraisal outcomes (towards students and parents).

- The notions of inner and outer fringes enable very clear statements of what a student knows/can do at this very moment and it defines very clear indications of what a student can/should learn or practice next.

The CbKST approach originates from the field of intelligent and adaptive tutorial systems, that is, autonomous computer algorithms that attempt to interpret learner needs, learning progress, and problems within learning episodes and tailor educational measures accordingly. The aim is, essentially, to provide individual students with personalized and psycho-pedagogically optimal learning experiences and environments. The challenge within NEXT-TELL is, to translate this "computer-centered" approach to the 21st century classrooms and the use of human teachers. In addition, the approach has to be mapped and linked to the other important aspects and facets of NEXT-TELL such as OLM, ECAAD, or TISL.

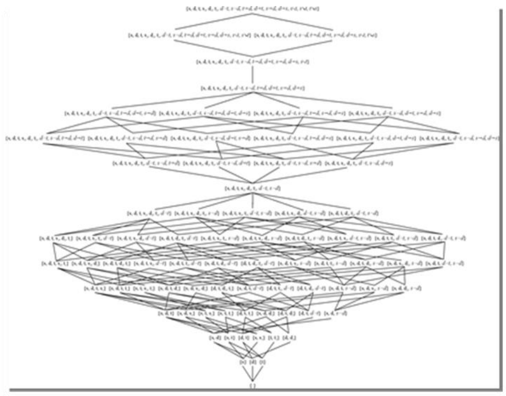


Figure 3. Illustration of a competence structure derived from the prerequisite relation shown in Figure 2, panel (a).

2.2 Practical Issues

The outlined approach includes three core components the domain/learning/learner modeling and the obtained planning of sequences of learning activities and assessment activities. The approach facilitates an evidence-centered approach to evaluating students' performance by allowing for a mapping of various activities (the evidences) to concrete competencies (the level of proficiency). Initially a conceptual analysis of the domain/subject matter is performed, subsequently the domain is modeled according the CbKST methodology (as briefly introduced above), followed by mapping the latent competencies to concrete learning an assessment activities. In the first instance, learning and assessment activities must be identified (by a teacher or learning designer) these activities are associated with the latent competencies in terms of required competencies (to be able to understand/perform a learning activity or to be able to master an assessment problem) and taught competencies (those that are taught or practiced with an activity). On the basis of the initially developed competence structure an activity structure is induced automatically. Subsequently, these modeling work leads to a well-defined learning sequence and it allows an assessment from multiple sources. The ECAAD planner serves as a means of the concrete analysis and planning work while the results may feed into an (open) learner model.

An important issue is the concrete application of such ideas in the classrooms. Establishing formative assessment scenarios including the necessary demands to learning analytics and the efforts required for visualization and communicating the information is highly resource demanding. Smart software is required that supports teachers. In the context of the NEXT-TELL project we developed a tool named ProNIFA which is supposed to promote and facilitate formative assessments.

3 Probabilistic, Unobtrusive, Formative-Assessment

ProNIFA stands for probabilistic non-invasive formative assessment and is developed in the context of the Next-Tell project. The tool, in essence establishes a handy user interface for CbKST-related services and functionalities. The services are running on a service and cover, broadly speaking, CbKST-related computation tasks, such as updating of probability distributions over competence structures. In addition to that, ProNIFA provides several authoring, analysis, and visualization features. The tool is a Windows application that utilizes various interfaces and links to online-based contents (Figure 4).

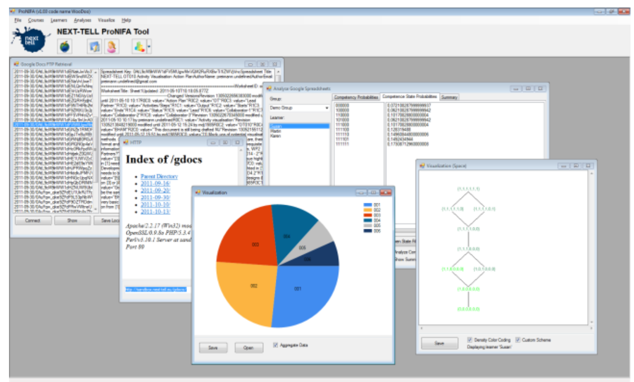


Figure 4. Screenshots of the ProNIFA tool.

A distinct feature in the context of formative assessment is the multi-source approach. ProNIFA allows connecting the analysis features to a broad range of sources of evidence. This refers to direct interfaces (for example to Google Docs) and it refers to connecting, automatically or manually, to certain log files. The only requisite is the availability of log files on the local computer, through HTTP access or via FTP. Using this level of connectivity, multiple sources can be merged and can contribute to a holistic analysis of learners' achievements and activity levels. As an example, ProNIFA enables a teacher to use the results of a Moodle test, exercises done in Google Spreadsheets, and the commitment displayed in a virtual meeting in a chat, to conduct a semi-automated appraisal of students.

The interpretation of the sources of evidence occurs depending on a-priori specified and defined conditions, heurists, and rules which associate sets of available and lacking competencies to achievements exhibited in the sources of evidence. Very basically, the idea is to define certain conditions or states in a given environment (no matter if a Moodle test or a status of a problem solving process in a learning game). Examples for such conditions may be the direction, pace, and altitude a learner is flying with a space ship in an adventure game or a combination of correctly and incorrectly ticked multiple choice tasks in a regular online school test. The specification of such state can occur in multiple forms, ranging from simply listing test items and the correctness of the items to complex heuristics such as the degree to which an activity reduced the 'distance' to the solution in a problem solving process (technically this can be achieved by pseudo code scripting). The next step of this kind of planning/authoring is to assign a set of competencies that can be assumed being available and also lacking when a certain state occurs. This assumption can be weighted with the strength of the probability updates. In essence, this approach equals the conceptual framework of micro adaptivity as, for example, described by [5].

3.1 Innovative visualizations

Summarized, ProNIFA collects information about learning-related activities from various sources and analyses and interprets these information in terms of learning progress and abilities. This can be done on the level of individual learners as well as on the level of groups and entire classes. The analyses are running in the background and, in essence, deliver probabilities sets and probability distributions. To support formative assessment, the results need to be worked up and visualized in an easy to understand, intuitive way. Presently ProNIFA provides several techniques to visualize assessment results. On the one hand there is a set of charting methods such as bar charts or line graphs that can visualize abilities in a rather conventional form. For example, the distribution of grades can be illustrated as a pie chart (cf. Figure 4). In addition, ProNIFA offers more innovative and distinct ways of visualization information. Three of them are presented and introduced in the following section.

3.1.1 Hasse diagrams

A Hasse diagramm is a strict mathematical representation of a so-called semi-order. Invented in the 60s of the last century by Helmut Hasse, entities (the knots) are connected by relationships (indicated by edges), establishing a directed graph. The properties of a semi-order are (i) reflexivity, (ii) anti symmetry, and (iii) transitivity. In principle, the direction of the graph is given by arrows of the edges; per convention however, the representation is simplified by avoiding the arrow heads, whereby the direction reads from bottom to top. In addition, the arrows from one element to itself (reflexivity property) as well as all arrows indicating transitivity are not shown. The following image illustrates a Hasse diagram (Figure 5).

Hasse diagrams enable a complete view to, partially, huge structures. Insofar, they are ideal for capturing the large competence spaces occurring in the context of CbKST-based assessment and recommendations. Very briefly, a Hasse diagram shows all possible (admissible) competence or knowledge states. By the logic of CbKST, each learner is, with a certain likelihood, in one of the competence states. This

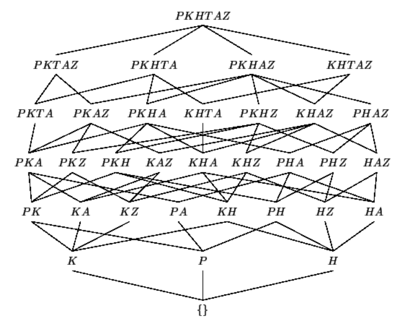


Figure 5. Example for a Hasse diagram.

allows coding the state likelihoods for example by colors and thereby visualizing areas and set of states with high (or vice versa low) probabilities. The simplest approach would by highlighting the competence state for a specific learner with the highest probability. The same coding principle can be used for multiple learners. This allows for identifying various sub-groups in a class, outliers, the best learners, and so on. A second aspect comes from the edges of the graph. Since the diagram reads from bottom to top, the edges indicate very clearly the "learning path" of a learner. Depending on the domain, we can monitor and represent each learning step from a first initial competence state to the current state. In the context of formative assessment, such information elucidates efforts of the learners, learning strategies, perhaps used learning materials, but also the efficacy of the teachers. Finally, a Hasse diagram offers the visualization of two very distinct concepts, the inner and out fringes. The inner fringe indicates what a learner can do / know at the moment. This is a clear hypothesis of which test/assessment items this learner can master with a certain probability. Such information may be used to generate effective and individualized tests. The concept of the out fringe indicates what competency should or can be reasonably taught to a specific learner as a next step. This provides a teacher with clear recommendation about future teaching on an individualized basis.

3.1.2 Pixel Clouds

Pixel clouds are a similar concept of representing ability on an individual or group level. In principle, the pixel clouds depicts each competence state (or skill/competency) as a single pixel. Each of the competence states is assigned a probability value which is color coded. The brighter a pixel is the higher is the corresponding probability, vice versa, the darker a pixel is the lower is the corresponding probability. The difficultly (or in other terms the structural location) of a competence or competence state in given by the position in the Euclidean space, ordered from left to right. This type of visualization has the great advantage that huge competency spaces can be grasped with a single sight. Despite maybe huge spaces, important information for teachers can be displayed on a single screen without the need for zooming. As show in the Figure 6, by this means also temporal information can be illustrated easily and quickly.

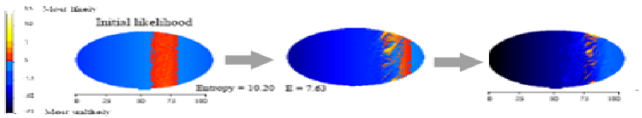


Figure 6. Pixel cloud illustrating an assessment process. With each piece of evidence (in an ECAAD sense), the picture of a learner's competencies is getting clearer and more stable.

3.1.3 Problem Spaces

A problem space is a formal and complete description of all possible solutions steps for a specific problem, represented from the starting state to the final (desired) state. All steps are represented on the basis of their admissibility according to the corresponding set of rules. The figure below illustrates the famous problem (game) the "Tower of Hanoi". In the context of formative assessment, this type of visualization can be used to illustrate the progress of students in problem solving situations. This might be on a small scale (within a short term problem, e.g., a test item) or on a large scale (e.g., in terms of a medium to long term project). It shall be highlighted that the term "problem" refers to a broad notion of task, a problem might well be a mathematical test item which is based on a set of calculation rules. The type of illustration (cf. Figure 7) is highly intuitive and, maybe more importantly, allows not only a snap shot of the present state but also indicates chosen solution paths.

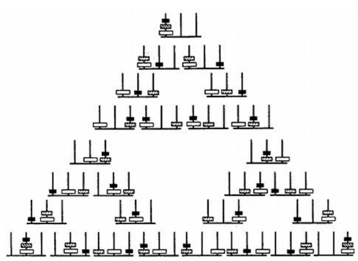


Figure 7. The image shows the problem space for the famous Tower of Hanoi problem/game.

4 Conclusions

It is evident that the future of teaching and learning necessarily needs to pursue innovative and novel paths. In particular in the STEM area, a clear and important trend is formative assessment, ideally on the basis of non-numerical information. This, unfortunately, is not a trivial attempt. It requires a highly amount of resources from a teacher to support the individuals in such a "formative" supportive and progressive way. The problem is that such resources are sparse and, facing today's classroom situations, formative assessment and appraisal of learner's cannot be mastered by teachers with a comprehensive and above all smart technology. In the context of the NEXT-TELL project we try to realize the necessary innovations and technical developments to empower teachers and to support teachers in their daily routines in order to realize better teaching and assessing. ProNIFA and CbKST as the conceptual backbone, are a promising approach. Presently, the ideas are investigated in European schools. First feedback shows that teachers highly appreciate this kind of assessment support. Of course, ProNIFA, as presented here, is just a small snapshot of the developments in the entire project. The fundamental aim is to cover the entire teaching cycle from planning, to conducting technology-supported teaching with various tools, to gathering evidence, analyzing and visualizing the student achievements, and finally to communicating and negotiating results and decisions.

5 Acknowledgements

The research and development introduced in this work is funded by the European Commission under the seventh framework programme in the ICT research priority, contract number 258114 (NEXT-TELL; http://www.next-tell.eu).

6 References

[1] Falmagne, J., & Cosyn, E., Doignon, J., & Thiéry, N. (2003). The Assessment of Knowledge, in Theory and in Practice. Institute for Mathematical Behavioral Sciences, UC Irvine, USA. Available online at http://www.scribd.com/doc/3155044/Science-Behind-ALEKS.

[2] Doignon, J., & Falmagne, J. (1985). Spaces for the assessment of knowledge. International Journal of Man-Machine Studies, 23, 175–196.

[3] Doignon, J., & Falmagne, J. (1999). Knowledge Spaces. Berlin: Springer.

[4] Albert, D., Kickmeier-Rust, M. D., & Matsuda, F. (2008). A formal framework for modelling the developmental course of competence and performance in the distance, speed, and time domain. Developmental Review, 28, 401-420.

[5] Kickmeier-Rust, M. D., & Albert, D. (2011). Micro adaptivity: Protecting immersion in didactically adaptive digital educational games. Journal of Computer Assisted Learning, 26, 95-105.

An Industry Focus Group Forum for the Assessment of New Engineering Graduates

K. Ferens
Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada
ferens@ee.umanitoba.ca

Abstract—This paper reports on an Industry Focus Group (IFG) forum, which took place on 20 October 2011. The objective of the forum was to obtain local Industry's perception and opinions of the strengths and weaknesses of new engineering graduates (from the Department of Electrical and Computer Engineering, University of Manitoba) at the time they enter the work force. The participants of the IFG forum identified missing knowledge, skills, and attitudes in new graduates. In addition, they identified key attributes which make engineering employees best-in-class in their field; these key attributes were intended to be targets to which new engineering graduates can aspire. The top priority identified strength of a new engineering graduate was attitude and the top weaknesses were tools, knowledge, and life-long learning. The reasons for having an IFG forum stemmed from the need to improve engineering education, to assess the performance of new engineering graduates, to obtain an external assessment of the program outcomes, and to satisfy the new accreditation requirements as specified by the Canadian Engineering Accreditation Board.

Keywords**-**Industry focus group forum, assessment, accreditation, CEAB.

1 Introduction

The Canadian Engineering Accreditation Board (CEAB) [1] has established new and additional accreditation criteria for undergraduate programs in Canadian Engineering educational institutions. These new accreditation criteria are focused on outcomes based assessment, and they state:

"The institution must demonstrate that the graduates of a program possess the attributes under the following headings. The attributes will be interpreted in the context of candidates at the time of graduation."

1. *A knowledge base for engineering*
2. *Problem analysis*
3. *Investigation (Validation)*
4. *Design*
5. *Use of engineering tools*
6. *Individual and teamwork*
7. *Communication skills*
8. *Professionalism*
9. *Impact of engineering on society and the environment*
10. *Ethics and equity*
11. *Economics and project management*
12. *Life-long learning*

The spirit of this requirement is that institutions must arrange for assessment of their graduates at the time they have left the intuition and have entered the work force.

There are many methods for performing an assessment of engineering graduates. These methods include surveys, interviews, and focus groups. In addition, conversational analysis and observations may be used [2]. Surveys and questionnaires are difficult to create because of interpretational, cultural, and societal differences in the readers. Interviews are more personal, but can be so to a fault, since some honest opinions may not be communicated in a personal contact for one reason or the other. Another interesting method calls for eaves dropping on student conversations during the process of design, review, and status sessions. Much can be learned by observing the behavior of students and what is said while students are in the design and development process. Related to this is a method to observe new engineering employees during their first few months at work. Observing how the employee reacts to new situations or expected scenarios can give insight in what is lacking or the strengths of new graduates. Of course, these observers can be industry members themselves, but academics would have a third-party view of the process.

Another form of assessment is based on measuring incremental abilities of students during the first several years enrolled in the program. This technique takes an integrative approach, and assumes that if a student has the ability of the parts, then the whole will take shape at the time of their graduation. Furthermore, the assessment criteria are based on the program's objectives, which may have been done in isolation and may not specifically address whether the criteria supports the goals and objectives of the industry. This method of assessment may establish reasonable processes, but may not assess how well the institution is doing to satisfy industry's needs. How can educators substantiate claims that student acquisition of certain topics at the completion of a degree will support the goals and objectives of industry?

This paper presents an external assessment of the engineering institution at the University of Manitoba, Canada. This work is an extension and continuation of the work reported in [3]. The extension was to include all disciplines in the forum, including Bio-systems, Computer, Civil, Electrical, and Mechanical and Manufacturing engineering. The extension also explored the strengths and weaknesses of new engineering graduates that are common to all disciplines. As required by the CEAB, the forum is a continual process, which is performed twice a year, and the results are fed back for program evolution.

The remainder of this paper discusses some related work. This is followed by a description of the IFG Forum – its motivation, composition, and intended faculty response. Then a description of the event, the forum, and the presentation and analysis of the results of the forum follows. Finally conclusions are given.

2 Related Work

Many studies have reported on the impact of engineering education on the local industry. In [3], local industry assessed a local engineering education institution to determine to what degree their expectations were being met. The approach investigated the areas of strengths and weaknesses of the engineering graduates and compared the results to their counterparts in universities abroad. A special questionnaire was sent to private sector company managers to elicit their opinions. While a questionnaire has its benefits, as [4] explains the quality of the information acquired depends on the extent to which subjects choose to respond honestly as well as their ability to report accurately. Likewise in [5], a survey study was performed to determine how well chemical engineering graduates were prepared for work in industry. In [6], the process of assessing engineering graduates is compared with the Quality Function Deployment (QFD) model, which has been used in industry successfully to develop products. In the QFD model, product requirements are carefully gathered from stakeholders before embarking on any development activities. This ensures that the final product meets the expectations of the customer. Likewise, engineering education instructors should consult with their major stakeholder, i.e., industry, to determine what local industry values and expects from those engineers who work within it. The paper [2] classified assessment methods in engineering education into descriptive and experimental studies. Descriptive techniques in engineering education include surveys, interviews, focus groups, conversational analysis, observation, and ethnography. Many examples were given, but none explored the benefits of a *forum* organization.

3 Industry Focus Group Forum Constitution

The Industry Focus Group Forum is supported by several advantages and benefits. The IFG Forum is an external assessment of an engineering institution – its program, educators, instructors, and people. The IFG consists of many industry representatives from many different sectors, who can offer diverse opinions related to their disciplines. The IFG forum is profoundly motivated to improve engineering education. Industry members openly give their opinions of the skills, knowledge, and attitudes of new engineering graduates. The educational institution acquires this external viewpoint and uses it to improve their programs. The IFG Forum is a process to improve educational programs, and thus can be claimed to meet accreditation requirements. The IFG members meet with faculty members to discuss educational matters on common ground – a foundation that has education and the desire to improve it deep at its core.

3.1 Composition of the IFG

The IFG was composed of industry members from several diverse disciplines: Bio-systems Engineering, Civil Engineering, Electrical and Computer Engineering, and Mechanical and Manufacturing Engineering. There were 25 industry members from these diverse disciplines, who were directly involved with the hiring, training, and assimilation of new graduates in their companies: there were seven directors, nine managers, three department heads, one team leader, and six R&D engineers. There was a broad range of industry sectors represented: hydro, structural, hospital, telecommunications, information security, aerospace agriculture, mechanical, pharmaceuticals, and bio-systems. The forum focused on a cross-section of different engineering functional areas including requirements, architecture, modeling, design, quality assurance, development, technology transfer, manufacturing, supply chain, regulatory compliance, profession, ethics, impact on society, and supervisory management.

The IFG forum was also attended by a contingent of 10 faculty members, including one associate dean, 6 department heads and associate heads, and two professors.

3.2 Motivation for the IFG Forum

There were several reasons why the Faculty of Engineering was interested in the existence of and participation in an IFG Forum. The objectives were to improve engineering education, to improve student learning, and to improve the educational institution.

1) To Be an External Assessment

The IFG forum was an external assessment of an engineering program. It obtained Industry's perception of the strengths and weaknesses of new engineering graduates at the time they entered the work force. As such, it was intended to provide an unbiased and third-party appraisal of the performance of an educational system. It was intended to separate the instructional material (inputs) given to students from the actual attributes (outputs) exhibited by students at the time they entered the workforce. An input given to a system may not always show in the output of the system. Furthermore, the external assessment was intended to avoid a possible conflict of interest, which may arise were the dispensers of instructional material asked to judge their own performance, i.e., to evaluate the impact of their own administration.

2) To Be an Assessment Tool

The IFG forum was used as an assessment tool, to measure the quality of students after the students have left the institution and have begun their careers in industry. It obtained Industry's perception of both the strengths and weaknesses of new engineering graduates at the time they entered the work force. The IFG identified missing knowledge, skills, and attitudes in new graduates. As well, the IFG identified key attributes which make engineering employees best-in-class in their field, with the intension that these key attributes can be strengthened and communicated to students throughout their studies.

3) To Satisfy CEAB Accreditation Requirements

The purpose of the forum was to satisfy the new "Continual Improvement" CEAB Accreditation Requirement: "*There must be processes in place that demonstrate that program outcomes are being assessed in the context of the graduate attributes, and that the results are applied to the further development of the program*" [1]. The IFG Forum *was* such a process whereby industry members identified and ranked the strengths and weaknesses of new engineering graduates, and that their assessment was given in terms of the CEAB attributes. Furthermore, the Faculty intends to use the identified strengths and weaknesses to improve and develop the program.

In addition, the purpose of the IFG Forum was to satisfy the new "Graduate Attributes" CEAB Accreditation Requirements: "*The institution must demonstrate that the graduates of a program possess the attributes ... The attributes will be interpreted in the context of candidates at the time of graduation*" [1]. The IFG Forum's assessment of new graduates was placed in the context of the CEAB attributes, and, furthermore, the assessment was done at the time the graduates entered the workforce.

The Associate Dean of Engineering gave a presentation to the IFG, which explained the Faculty's interest in the existence of and participation in the IFG forum. The Associate Dean stated that: To the Faculty, the IFG forum represents an external process, whereby (1) outcomes of new engineering graduates can be assessed at the time they enter the workforce; (2) outcomes can be assessed in the context of the 12 CEAB attributes; (3) the results of the forum can be fed back to the faculty and applied to the further development of the program; and (4) the assessment can be performed on an annual basis. Such a process would therefore be applied towards a partial fulfillment of the new CEAB accreditation requirements.

4) To Satisfy Local Industry Requirements

At the University of Manitoba, and most typically at other universities around the world, about 90% of engineering graduates proceed to work in industry. This means that Industry's needs should be at the forefront of the objectives and that industry's needs must be met. The more efficient and competitive industry becomes; the more successful academia is seen to be. This will translate to more federal and provincial funding for the Faculty. The purpose of the IFG Forum was to allow the Faculty to be accountable to local industry. Through the IFG Forum, the Faculty reaches out to local industry; asks them what their requirements are; uses this information to improve their program; and thus helps local industry be more productive, efficient, and competitive.

5) A Method of Requirements Gathering

Simply put, the IFG Forum was a method to gather industry statements of requirements. Once the statements were gathered, a team of engineering education researchers analyzed the statements, and translated them to actual requirements. The requirements were intended to be used by faculty to continually improve the program. The IFG Forum was an assessment instrument, and, as such, it is a method for gathering data to support evaluation of graduates of engineering.

3.3 Faculty Response

The results of the forum – the identified strengths and weaknesses of new graduates – has been made available to faculty members, with the intention that they be used as input to help reinforce, modify, shape, and restructure programs at the Faculty of Engineering, University of Manitoba. A presentation was given to Faculty Council on 24 January 2012, to explain the purpose of the forum; to show and analyze the results; to explain the importance of the faculty response; and to give faculty ideas about possible responses.

A report for each Forum was written and distributed to IFG members and to the Faculty of Engineering at the University of Manitoba. The past forum (February 2011) was shared with CEEA Conference in St. Johns, NL and the FECS conference in Las Vegas. As well, several seminars were given for students of the IEEQ program and ECE students, explaining what our local industry values and expects from those engineers who work within it.

Short term changes have already been made. Syllabus changes have already been made in courses ECE 3610 and ECE 3740. The course content of the Computer Engineering Curriculum was compared with an international standard. Excesses and deficiencies were identified and corrected.

The Faculty of Engineering created a long term plan. In this plan, they specified how the IFG and other stakeholders may contribute to the continual improvement of the programs within the institution.

4 IFG Forum Procedure and Results

The Industry Focus Group (IFG) forum took place on 20 October 2011 at a central location, away from businesses and away from the University. It took the better part of the day; many comments were revealed, discovered, and openly given. And faculty members listened attentively, taking notes, and asking questions. The moderator controlled the discussions. There was two phases: Phase 1 was a group discussion in which all engineering disciplines (Bio-systems Engineering, Civil Engineering, Electrical and Computer Engineering, and Mechanical and Manufacturing Engineering) took part. Phase 2 was a breakout session, in which the individual disciplines had separate discussion.

4.1 Role of Academic Participants

The moderator differentiated the roles of those participants from industry and those attendees from academia. The meeting's goal was to obtain the perceptions and priorities of industry participants. The primary role of academic participants was to be observers of the process and to clarify any curriculum issues. An added benefit is the opportunity to meet and network with industry professionals.

The moderator explained to the group that the central part of the session was identification of knowledge and skills that are strengths and weaknesses of new graduates as they enter the work force. Industry members will be the participants generating these ideas through discussions and brainstorming methods, and they will vote on the prioritization of the identified gaps. Academic guests can take part in the discussion at the session, but academic attendees shall not vote on priorities.

4.2 Phase 1: Strengths of New Engineering Employees

A brainstorming session was implemented in the first phase of the meeting to elicit and identify the strengths of new engineering employees (Table 1). The entire group took part in this session. The moderator listened attentively and captured the main points on PowerPoint slides, and the table leaders took detailed notes of the comments made by the participants at their tables.

Table 1 Group Identified Strengths of Engineering Employees. Listed in order of discussion.

#	**Statement of Strength (SoS)**
1	Creativity
2	Identify the problem
3	Communication
4	Initiative (self starter)
5	Ability to grow in company
6	Interpersonal skills
7	Innovative Resourceful
8	Attitude
9	Cross-disciplinary
10	Knowledge
11	Time management
12	Decisiveness
13	Adaptability
14	Project Management

1) Attitude

The best employees always have the best possible positive attitude towards engineering. They consider engineering to be a respectable field, and the work engineers do and the problems they solve have a positive impact on society. They exhibit this attitude in their everyday work; they are confident, competent, and have a high degree of performance. They understand and acknowledge the need for working together; communicating ideas among colleagues and the general public; and engaging in lifelong learning and to remain knowledgeable of contemporary issues. As such, the attitude trait of the best-in-class employee may be classified under the CEAB attributes of "individual and teamwork," "communication," "professionalism," "Impact of engineering on society and the environment," "ethics," and "lifelong learning."

2) Knowledge

The underpinning trait of the best-in-class engineer of a company or institution is knowledge, on both a broad base and specific knowledge about the company's business. One could not be technically creative, communicative, innovative, decisive, and come up with good designs for given problems, without having knowledge and exhibiting competence of basic science and fundamental engineering topics.

3) Creativity

Another identified strength of engineering employees was creativity. Obviously, before one can do creative engineering, one must have a depth of technical knowledge of facts, laws, and methods. Therefore, creativity is related to the CEAB attribute of "knowledge." Creativity is the ability to produce new things or new knowledge to solve some problem. Therefore, creativity can also be classified under the "design" CEAB attribute. A creative engineer finds their own work to do, rather than waiting for someone else to give them an assignment. In industry, creative engineers work more than 40/week for the sheer joy of it; they work in their spare time, during evenings and weekends, on their "secret" project. However, creativity needs to know bounds and limits; being productive at work must have higher priority than merely the sake of being creative.

4) Communication

Communication has always attracted attention in engineering. The IFG identified the best-in-class engineer has having the ability to communicate effectively with the engineering team and with the multi-disciplinary community at large. An engineer must have:

1. Fluency in written and spoken English.
2. Ability to make effective oral and written communications to technical and nontechnical audiences.
3. Empathy to listen, comprehend, and work with viewpoints from other stakeholders.
4. Ability to discuss and present arguments clearly and concisely.
5. Ability to represent engineering issues and the engineering profession to the broader community.

5) Initiative (Self Starter)

Initiative (self-starter) is a personal skill and attitude. It stems from a sense of responsibility and knowing that growth happens when roots find new sources of nutrition, not just from the hands that feed them. The best-in-class engineering employee knows what is required, and doesn't need to be told what to do and how to do it. They work intellectually on par and in parallel with their supervisors and managers, working toward common goals, without having to be explicitly informed of (spoon fed) those goals and objectives of the project and company. As such, the initiative trait of a new engineering employee can be classified under the CEAB attributes of "Individual Work, Professionalism, Ethics, and Life-long Learning."

6) Problem Identification

The IFG identified one of the strengths of engineering employees is their uncanny ability to extract and identify deep rooted and often hidden problems to be solved in a project. Many participants commented on the difficulty of identifying the actual problem in many complex projects. Many times in the past new engineering graduates started to work on a project without having identified the actual problem to be solved. Much company time and resources were mostly wasted because the employee worked on solving an indirect or peripheral problem to the project. This has occurred because of misinterpretation and a lack of full understanding of the problem.

An engineering employee with the problem identification strength exhibits abilities:

1. To critically investigate the issues surrounding the problem;
2. To evaluate information from a broad range of appropriate and acknowledged sources (stakeholders);
3. To explain the problem in clear, unambiguous, and simple terms; and
4. To discuss the problem's relevance to the project, and how it relates to the issues raised by the stakeholders.

The problem identification strength can be classified under the "Problem Analysis" and "Investigation" CEAB attributes, since a critical investigation is required to extract the actual problem from the statements and issues raised by the stakeholders, and this strength requires an ability to use appropriate knowledge and skills to identify, formulate, analyze, and solve complex engineering problems in order to reach substantiated conclusions.

4.3 Prioritization of Strengths

The group prioritized the identified strengths on a scale of 1 – 5, where 1 was the highest priority. The analysis reversed the priority scale such that 1 was the lowest priority and 5 was the highest priority. The total priority was obtained by summing over the priorities over all disciplines, i.e., over Bio-systems, Computer, Civil, Electrical, and Mechanical. As shown in Fig. 1, the top priority was attitude, followed closely by knowledge, creativity, communication, and initiative.

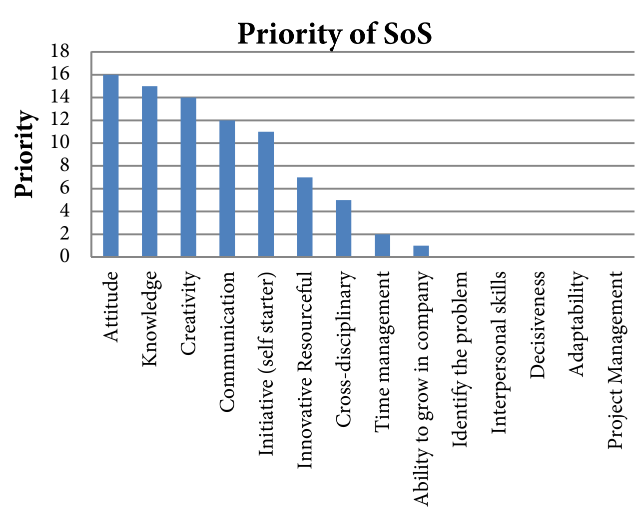


Fig. 1 Group prioritization of Statement of Strengths (SoS) of new engineering employees. In this figure, the priority was represented on a reverse scale of 1-5, where '5' was the highest priority and 1 was the lowest priority. The total priority was obtained by summing the priorities over all disciplines.

4.4 Mapping of Strengths to CEAB Attributes

The group identified strengths were mapped to CEAB attributes. Since many of the identified strengths were of a broad nature, they were mapped to one or more CEAB attributes. For example, the attitude attribute was mapped to "Individual and Teamwork," "Communication," "Professionalism," "Impact of engineering on society and the environment," and "Ethics and Equity."

Table 2 CEAB Attribute mapping of Group Identified Strengths of New Engineering Employees. P=Priority.

SoS#	P	CEAB Attribute	Statement of Strength (SoS)		CEAB Attributes
1	4	1,4	Creativity	1	A knowledge base for engineering
2		2,3	Identify the problem	2	Problem analysis
3	2	7	Communication	3	Investigation (Validation)
4	3	6, 8, 10, 12	Initiative (self starter)	4	Design
5		12	Ability to grow in company	5	Use of engineering tools
6		7	Interpersonal skills	6	Individual and teamwork
7	4	1,2,3,4	Innovative Resourceful	7	Communication skills
8	1	6,7,8,10,12	Attitude	8	Professionalism
9		6	Cross-disciplinary	9	Impact of engineering on society and the environment
10	5	1	Knowledge	10	Ethics and equity
11		11	Time management	11	Economics and project management
12		1,2,3,6	Decisiveness	12	Life-long learning
13		4,5,6	Adaptability		
14		11	Project Management		

4.5 Phase 2: Weaknesses of New Graduates

The second phase of the meeting implemented a brainstorming session to elicit and identify the weaknesses of new engineering employees. A breakout session required each discipline-table to discuss amongst themselves their perceived weaknesses of new engineering graduates at the time they entered the workforce.

4.6 Mapping of Weaknesses to CEAB Attributes

Each discipline-table was requested to identify weaknesses of new graduates entering their field, and to map those weaknesses to CEAB attributes. Due to time limitations, some discipline-tables completed only the identification of weaknesses. The table leaders performed the outstanding mapping of weaknesses to CEAB attributes on behalf of the discipline-tables. The results are shown in the following tables (Table 3, Table 4, Table 5, and Table 6).

Table 3 Bio-Systems Engineering specific weaknesses mapped to CEAB attributes.

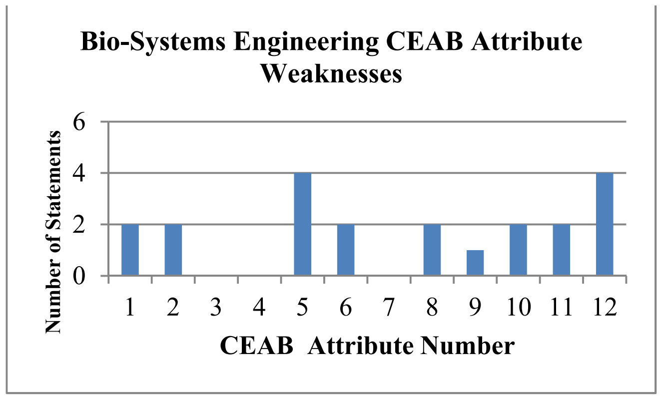


Table 4 Computer Engineering specific weaknesses mapped to CEAB attributes.

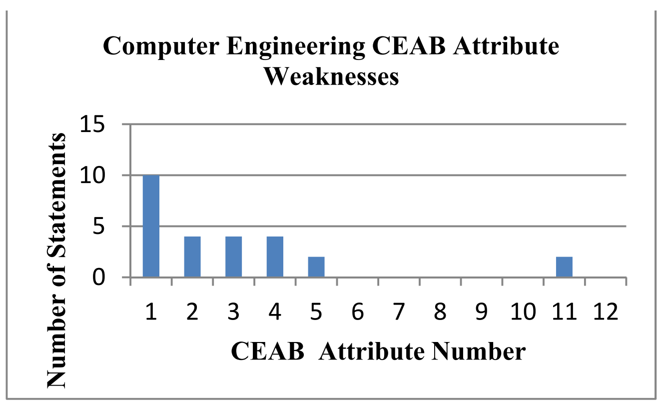


Table 5 Electrical Engineering specific weaknesses mapped to CEAB attributes.

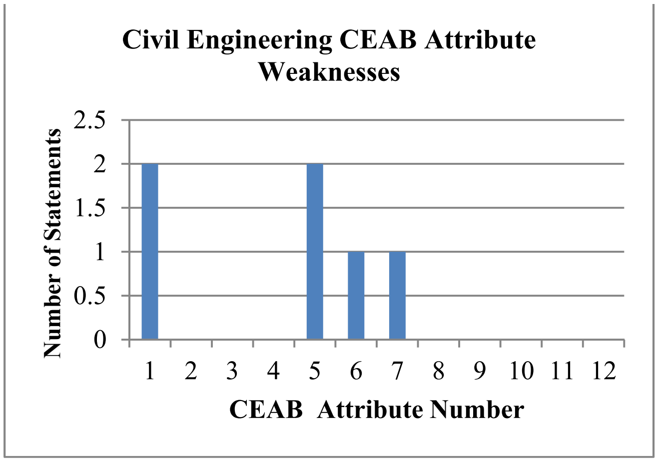
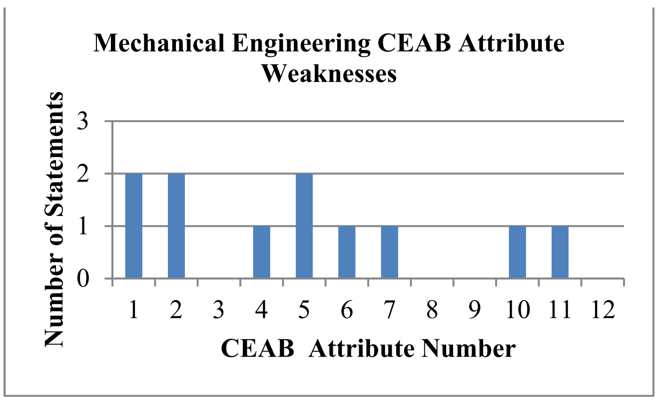


Table 6 Mechanical Engineering specific weaknesses mapped to CEAB attributes.



5 Conclusions

This paper reports on an Industry Focus Group (IFG) forum, which took place on 20 October 2011. The objective of the forum was to obtain local Industry's perception and opinions of the strengths and weaknesses of new engineering graduates (from the Department of Electrical and Computer Engineering, University of Manitoba) at the time they enter the work force. The highest priority strength of a new engineering graduate was attitude and the top weaknesses were tools, knowledge, and life-long learning. The IFG Forum is intended to be a continuous process, which is performed on a semi-annual basis, whereby requirements are gathered from industry; these requirements are mapped to CEAB attributes for validation; these identified strengths and weaknesses of new graduates are fed back to the Faculty for curriculum modification, to reinforce and strengthen the identified strengths and to improve the weaknesses. Over several iterations the feedback control system will evolve to a stronger and more industry relevant faculty of engineering.

References

[1] E. Canada, "Canadian Engineering Accreditation Board Accreditation Criteria and Procedures report 2011," [Online]. Available: http://www.engineerscanada.ca/files/w_Accreditation_Criteria_Procedures_2011.pdf. [Accessed 8 April 2012].

[2] K. Ferens and M. Friesen, "Industry Focus Group Forum for Engineering Program Assessment," in *International Conference on Frontiers in Education: Computer Science and Computer Engineering*, Las Vegas, 2011.

[3] S. A. Abu- Eisheh, "Assessment of the Output of Local Engineering Education Programs in Meeting the Needs of the Private Sector for Economies in Transition- The Palestinian Territories Case," *International Journal of Engineering Education,* vol. 20, no. 6, pp. 1042-1054, 2004.

[4] B. M. Olds, R. L. Miller and B. M. Moskal, "Assessment in Engineering Education: Evolution, Approaches and Future Collaborations," *Journal of Engineering Education,* vol. 94, no. 1, pp. 13-22, 2005.

[5] R. Martin, B. Maytham, J. Case and D. Fraser, "Engineering graduates' perceptions of how well they were prepared for work in industry," *European Journal of Engineering Education,* vol. 30, no. 2, pp. 167-180, 2005.

[6] P. Brackin, "Assessing Engineering Education: an Industrial Analogy," *International Journal of Engineering Education,* vol. 18, no. 2, pp. 151-156, 2002.

A Minor in Health Informatics

Yali Wu, Kevin Daimi, Serhiy Morozov
Department of Mathematics, Computer Science and Software Engineering
University of Detroit Mercy
Detroit, Michigan, 48221
{wuya1, daimikj, morozose}@udmercy.edu

***Abstract*—The fusion of information technology and healthcare practice is reshaping the healthcare industry and revolutionizing patient care. There has been a growing demand for health informatics and health information management professionals that possess knowledge and skills in both disciplines. The multidisciplinary nature of healthcare informatics requires an education that combines a technical foundation with course materials covering current clinical trends, government regulations, and healthcare-specific best management practices. In this paper, we propose a Minor in Health Informatics with an aim to better prepare graduates for entry into the healthcare industry. The program objectives, outcomes, curriculum design, as well as assessment methods are discussed in detail. It is our ultimate goal to meet the increasing demand for health informatics professionals in the healthcare industry.**

***Index Terms*—Health Informatics, Minor, Program Outcomes, Curriculum Design, Program Assessment**

I. INTRODUCTION

Over the last twenty years, we have seen a considerable rise in the implementation and use of health information systems and information technologies in healthcare globally. With the current United States government initiatives to move to electronic healthcare documentation as a mandate for a national healthcare information infrastructure by 2014 [21], there has been an increasing demand for health informatics and health information management professionals in the healthcare industry. As an emerging academic discipline, the primary goals of health informatics are to improve the overall health of patients by combining basic scientific and engineering insights with the useful application of these insights to important problems [2]. Haux [13] defined health informatics as the science of systematically processing data, information, and knowledge in medicine and clinical research. The field draws contributions from various disciplines such as computer science, information science and management to collect, analyze, and transmit medical information [11], [15], [24]. Health informatics is playing an increasing role in the management and effective use of data generated in clinical care [3], [9].

The multidisciplinary nature of health informatics requires an education that combines a technical foundation with course materials covering evolving clinical trends, government regulations of healthcare, and healthcare-specific best management practices [23]. Internationally, a number of health and nursing informatics programs have been developed in different countries around the world. Medical schools in the United States, France, Germany, and the Netherlands have incorporated health informatics into their medical curricula and training, and developed various programs such as double degree programs (Master of Nursing and Master in Health Information Science) [4], Health Informatics Master [10], and Master in Nursing with a specialization in Nursing Informatics [17], [18].

Despite the number of health informatics related programs, we were unable to find any programs that provide students with a Minor in Health Informatics alongside their major field of study. In this paper, we propose such a Minor in Health Informatics that focuses on concepts and issues surrounding technology and information management in today's rapidly changing healthcare environment. The program is designed for students majoring in healthcare, nursing, biomedical sciences, computer science, or software engineering, who have taken some prerequisite courses in computer science, and yet show interests in additional training in health informatics. The minor will complement students' major area of study by exposing them to broader aspects of technology and healthcare. It will prepare them for entry into the health informatics field, and enhance career advancement opportunities. Students who complete the Health Informatics Minor will be prepared to analyze, design, develop, implement, evaluate, and maintain heath information systems in a wide array of healthcare settings.

The rest of this paper is organized as follows: in Section II, we demonstrate the need for a Minor in Health Informatics by presenting various career opportunities in the healthcare industry and existing sources of health informatics education. Section III introduces an overview of the minor including program objectives, program outcomes and the mapping between them. We present the admission requirements, minor requirements, and assessment methods of the Minor in Section IV, V, and VI respectively, and we conclude in Section VII.

II. MOTIVATIONS FOR MINOR

As medical data and health information continue to grow in size and complexity, the demand for their organization and management increases dramatically [13], [24]. In this section, we present the career outlook for health informatics professionals, and the existing channels of health informatics education. The lack of a minor program in health informatics education motivates us to propose the design of a Minor in Health Informatics.

A. Health Informatics Career Outlook

Health informatics involves evaluating, implementing, and utilizing technology to manage all information related to the patient care delivery process: clinical, financial, technological, and enterprise. The multidisciplinary nature of health informatics leads to multiple health related professions being involved in the delivery of healthcare [6]. These professions not only involve individuals considered to be at the forefront of delivery, such as medical doctors, dentists, physician assistants, nurse practitioners, and nurses; but also include laboratory and imagery technicians, pharmacists, social scientists, and public health and clinical researchers [6]. The most common example of health informatics in action is the emergence of the electronic health record (EHR) [8], [21]. Regardless of the setting - whether a physician office, hospital or nursing home - health informatics and information systems professionals perform the vital role of organizing and managing health information to ensure quality, patient safety, and security.

The Bureau of Labor Statistics estimates that job opportunities for health informatics and information systems professionals will grow much faster than the average for all occupations by 2018 [22]. Hospitals and other healthcare settings are pushing hard to enter the digital world. Unlike many other businesses, the majority of hospitals and other health-care providers are still tied to the old fashioned paper records. Consumers, insurers, state and federal governments are demanding a switch from paper to electronic records and systems. This change requires present healthcare professionals and staff to acquire new skills in IT and health informatics . That need is so great that medical institutions are actually sending out doctors and nurses for IT training in order to fill the gap.

As the current federal stimulus package is dedicating $19 billion to speed the adoption of electronic health records [20], this further resulted in increased demand for health informatics specialists. According to the American Health Information Management Association (AHIMA) [1], the average yearly salary for entry-level medical records and health information management specialists in 2008 was $48,000, with an overall average salary of survey respondents being above $57,000 annually. These promising statistics have resulted in widespread interests in health informatics education. Along with regular IT courses, additional health informatics training would give students more career opportunities in the future.

A health informatics education also enhances career advancement opportunities. Becoming a health information manager, director or vice president is usually dependent upon experience or obtaining a higher level degree in health informatics, information technology, or a related field. Additional training in health informatics will prepare them for a wider range of roles at mid-to-high management levels in the healthcare information setting.

B. Existing Sources of Health Informatics Education

A number of health informatics related programs have been developed in different countries around the world. In Europe, there have been several pioneering medical, biomedical and health informatics master-level programs in the Netherlands [10], Austria [14] and other European countries [12]. In North America, there are a number of programs that offer students Master in Biomedical and Health Informatics [7], or Master in Nursing with a specialization in Nursing Informatics [17], [18]. In Canada, the University of Victoria, having both a School of Health Information Science and Nursing, allowed for various programs such as Health Informatics Master and double degree programs (Master of Nursing and the Master in Health Information Science) [4].

Covvey et al. [5] surveyed existing "channels" of health informatics education that produced IT professionals needed within the healthcare industry. They also classified them into four categories:

- *Health Informatics Academic Programs*: university or college-level formal curricula that are intended to produce IT professionals with Bachelor, Master, and/or Ph.D degrees prepared specifically for addressing health informatics challenges.
- *Health Informatics Elective-Augmented General Academic Programs:* academic programs such as Computer Science at universities and colleges offer students the opportunity to take electives or do projects or thesis on health informatics-related topics.
- *Self-Managed Health Informatics Post-Professional Education*: professionals with only a background in computer-related or health-related disciplines extend their initial background with additional certificates and/or self-education to develop health informatics related knowledge.
- *On-the-job Health Informatics Learning*: IT staff without any academic qualifications learn via on the job experiences.

Most of the Health Informatics academic programs are at the undergraduate or the graduate level. They either share curricula with related programs or have a separate track. Some of them are interdepartmental and usually affiliated with health science (31%), medical (25%), public health (16%), and computer science(16%) schools [6].

Despite the number of health informatics related programs, we were unable to find any programs that offer a Minor in Health Informatics alongside the major field of study. We envision a significant need for a Minor program in Health Informatics, that could offer distinctive health informatics qualifications, but without another full-fledged major academic program. Students majoring in another discipline, with some prerequisite computer science background, can benefit from this minor program to receive further training in health informatics.

III. Program Overview

The Minor in Health Informatics is designed to provide basic proficiency in areas surrounding technology and information management in today's rapidly changing healthcare environment. The course sequence provides a strong background in information technology, in addition to focusing on requisite skills needed to work with health informatics applications, specifically electronic health records and clinical information systems. The design goals of our program are listed as follows:

- *Goal 1:* Provide health informatics educational experience that includes healthcare related projects, and practical analysis and design skills, which highlight teamwork, problem solving, communication skills, and ethical values needed for the health informatics profession.
- *Goal 2:* Produce health informatics professionals who are ready to work in Health Industry. The graduates will be equipped with the needed practical experience in health informatics.

A. Program Objectives

Program educational objectives describe the expected accomplishments of graduates during the first few years after their graduation. The educational objectives of the Minor in Health Informatics are designed to:

- Develop the ability to formulate, apply, analyze, and evaluate appropriate solutions to solve health informatics problems, and assess the quality of health informatics tools and techniques.
- Instill among students the ability to understand, analyze, and assess the value of health informatics ethics, and demonstrate the ability to effectively communicate ideas and outcomes, both orally and in writing, in a logical manner.
- Improve students' ability to work productively and effectively in teams including interdisciplinary teams, and infuse effective leadership skills and systems thinking approach to adapt to the dynamic healthcare environment.

B. Program Outcomes

Program outcomes are statements that describe what students are expected to know and be able to do by the time of graduation. Upon completion of the minor program, students should be able to achieve the following outcomes:

(a) Graduates will be able to store, organize, manage, and interpret various healthcare data and knowledge.

(b) Graduates will be able to apply Health Informatics techniques, methods, and processes to various healthcare areas including Public Healthcare, Consumer Healthcare, and Clinical Care.

(c) Graduates will gain the skills needed to lead and motivate Health Informatics departments and teams, manage complex projects, and understand the ethical responsibilities and implications of Health Informatics.

(d) Graduates will be able to apply effective oral and written communications in a Health Informatics project setting.

(e) Graduates will be able to analyze, design, implement, and assess Health Informatics software components and systems using various design techniques and guided by quality metrics.

C. Mapping Outcomes to Objectives

The Minor's program outcomes are mapped to the program objectives. For each program objective, one or more program outcomes are designed to meet the stated objectives. We show this mapping in Table I. The columns in the table represent the program objectives whereas the rows represent the program outcomes. An 'X' means a mapping relationship between the two, whereas its absence indicates 'not applicable'. This mapping also facilitates the assessment of the Minor in terms of overall program quality and continuous improvement.

Our approach in defining program outcomes, objectives and their mapping is consistent with the approach used by ABET (Accreditation Board for Engineering and Technology) accredited programs. Although it is a minor as opposed to a four year degree program, following this program design approach ensures that the minor meet its expected objectives. As will be shown later, our curriculum design includes a set of carefully chosen courses to ensure that the course outcomes be mapped to the program outcomes as well. The required courses of the minor and how they map to program outcomes are discussed in Section V.

IV. Admission Requirements

In order to be enrolled in the minor, the applicants must have taken a set of prerequisite courses, as we show in Table II. Many of these prerequisites (e.g., mathematics courses) may be required as part of their own majors, and therefore they do not impose extra burdens on the students. An important part of health informatics involves the management and effective use of data generated in clinical care. Therefore, we include two prerequisite courses: Data Structures and Database Systems to ensure that students have a technical background in data and information management. These courses will also prepare them to understand more advanced topics in health information systems and healthcare data management.

In addition, it is possible that the minor coordinator may replace some of these courses with courses of a similar nature. For instance, Introduction to Programing (currently taught using C++ in the Computer Science program) can be replaced by other modern object oriented programming courses such as Java. The minor coordinator can also waive some of these classes for students with some work experience.

V. Minor Requirements

The Minor in Health Informatics consists of six courses (18 credits in total). They are given in Table III. Note that

TABLE I
MAPPING OUTCOMES TO OBJECTIVES

Outcome/Objectives	**1)** Develop the ability to formulate, apply, analyze, and evaluate appropriate solutions to solve health informatics problems, and assess the quality of Health Informatics tools and techniques.	**2)** Instill among students the ability to understand, analyze, and assess the value of health informatics ethics, and demonstrate the ability to effectively communicate ideas and outcomes, both orally and in writing, in a logical manner.	**3)** Improve student's ability to work productively and effectively in teams including interdisciplinary teams, and infuse effective leadership skills and systems thinking approach to adapt to the dynamic healthcare environment
a) Store, organize, manage, and interpret various health care data and knowledge.	X		
b) Apply Health Informatics techniques, methods, and processes to various health care areas including Public Healthcare, Consumer Healthcare, and Clinical Care.	X		
c) Gain the skills needed to lead and motivate Health Informatics departments and teams, manage complex projects, and understand the ethical responsibilities and implications of Health Informatics.		X	X
d) Apply effective oral and written communications in a Health Informatics project setting.		X	X
e) Analyze, design, implement, and assess Health Informatics software components and systems using various design techniques guided by quality metrics	X		

TABLE II
PREREQUISITES FOR MINOR

Course Title	*Credits*
Applied Probability and Statistics	(3 cr.)
Discrete Mathematics	(3 cr.)
Introduction to Programming	(3 cr.)
Data Structures	(3 cr.)
Database Systems	(3 cr.)

each course in the program is a 3-credit course, and is mapped to one or more program outcomes. The Software Engineering courses (Healthcare Software Engineering and Healthcare IT Project Management) are designed with the understandings that all assignments, case studies, and projects be in a healthcare field. The course descriptions are listed as follows:

- *Introduction to Health Informatics*: introduces the concepts and practices of health informatics, history and current status of information systems in healthcare, major clinical applications (medicine, nursing, pharmacy, laboratory, public health), decision support methods and technologies, analysis, design, implementation, and evaluation of healthcare information systems, new opportunities and emerging trends, health informatics projects.
- *Clinical Informatics*: discusses the basic theories and methods employed during the design, implementation, and management of clinical information systems, electronic health records, terminology and standards of clinical information systems, clinical configuration, user interface design, computerized physician order entry, clinical decision support, and clinical reporting.
- *Healthcare Software Engineering*: applies software engineering techniques in the healthcare industry, including software processes, software requirements engineering, system models, architectural design, object-oriented design, software reuse, verification and validation, software testing, software cost, quality management, process improvement in the development of health information systems, health informatics projects.

TABLE III
MINOR COURSE REQUIREMENTS

Course Title	*Credits*	*Program Outcomes*
Introduction to Health Informatics	(3 cr.)	(a), (b)
Clinical Informatics	(3 cr.)	(a), (b)
Healthcare Software Engineering	(3 cr.)	(a),(b),(c),(d),(e)
Consumer Health Informatics	(3 cr.)	(a),(b)
Public Health Informatics	(3 cr.)	(a),(b)
Healthcare IT Project Management	(3 cr.)	(b),(c),(d)

- *Consumer Health Informatics*: discusses the application of information and communication technology to support personal healthcare decisions and promote health literacy, consumer health information seeking behaviors, preferences and resources, models for the delivery of consumer health information, quality of consumer health information, design and implementation of consumer health information resources, provider-patient relationship and personal electronic medical records.

- *Public Health Informatics*: presents the concept of public health information, the information needs of public health professionals, barriers and requirements of a public health information infrastructure, data and process standards, electronic health records, electronic data exchange, security issues; data registries and sources.
- *Healthcare IT Project Management*: introduces project management concepts, project metrics, project planning, risk analysis and management, project scheduling, project tracking, software quality assurance, software configuration management, communicating project information, critical paths, case studies, and term projects in a healthcare field.

VI. Program Assessment

To assure overall program quality, we designed a program assessment process for a systematic evaluation of the extent to which the objectives and outcomes of the Health Informatics Minor are met. It uses a combination of assessment practices that include:

- Individual course assessment to ensure that each course is achieving its learning outcomes and supporting the program outcomes.
- An Exit Survey offered to students completing the program to solicit their feedback on the program and on how to improve it.
- An Alumni Survey used to discover how well our graduates feel they were prepared for their current position.
- An Employer Survey used to obtain the feedback of employers on how well our graduates are prepared for their positions.

The results of the assessment will be used to make recommendations for continuous improvement of the minor program.

VII. Conclusions

As medical data and health information continue to grow in size and complexity, the high demand for health informatics professionals is here to continue. In this paper, we presented a Minor in Health Informatics that would meet the increasing demand for health informatics professionals in hospitals and other healthcare settings. The minor will expose the students to broader aspects of technology and healthcare, prepare them for entry into the health informatics field and enhance career advancement opportunities. We discussed the program objectives, program outcomes, and curriculum design that meet the proposed outcomes. Admission requirements and degree assessment ensure overall program quality.

As a further improvement to the program, we are considering to add a course that focuses on health information exchange standards, methods, and models. A course on various issues surrounding health information exchange would be very pertinent to realize the vision of a national healthcare information infrastructure [21]. The course will cover data standards necessary to achieve interoperability within and among complex healthcare organizations, information security, privacy and confidentiality of health data including HL7 [19] and HIPAA [16] standards. Another possible improvement is to add a minor design project course that gives students comprehensive training of their practical skills before entering the real-world workforce. It is our intent to create a minor that could train students in both technology and healthcare and to prepare them to get where the world is going.

References

[1] AHIMA. (2012, Mar.). *2008 Salary Study Delves Deeper into Factors Driving Pay.* [Online]. Available: http://www.ahima.org/membership/salarystudy08.aspx/

[2] R. B. Altman, "Informatics in the care of patients: ten notable challenges," *The Western Journal of Medicine*, vol. 166, no. 2, pp. 118–122, 1997.

[3] J. Mantas, E. Ammenwerth, G. Demiris, A. Hasman, R. Haux, W. Hersh, E. Hovenga, K.C. Lun, H. Marin, F. Martin-Sanchez, and G. Wright, "Recommendations of the International Medical Informatics Association (IMIA) on Education in Biomedical and Health Informatics. First Revision.," *Methods of Information Medicine*, vol. 49, no. 2, pp. 105–120, 2010.

[4] E. Borycki, N. Frisch, M. McIntyre, and A. Kushniruk, "Design of an innovative double degree graduate program in health informatics and nursing: Bridging nursing and health informatics," *Journal of the European Federation of Medical Informatics*, vol. 7, no. 1, pp. 31–39, 2011.

[5] H. Covvey and A. B. Pidduck, "Health informatics education, working paper," *Waterloo HIP Position Paper*, vol. 58-59, no. 1, pp. 1–50, 1999.

[6] K. Daimi, and G. Grabowski, "A Professional Science Master Degree in Health Informatics," in *The 2011 International Conference on Bioinformatics and Biocomputing (BIOCOMP '11).* Las Vegas, 2011, pp. 373–378.

[7] G. Demiris, "Interdisciplinary innovations in biomedical and health informatics graduate education," *Methods of Information Medicine*, vol. 46, no. 1, pp. 63–66, 2007.

[8] D. Detmer, M. Bloomrosen, B. Raymond, and P. Tang, "Integrated personal health records: Transformative tools for consumer-centric care," *BMC Medical Informatics and Decision Making*, vol. 8, no. 1, p. 45, 2008.

[9] S. Garde and E. Hovenga, "Australian health informatics educational framework," *Australian Health Informatics Educational Framework*, vol. 2, pp. 1–15, 2005.

[10] W. Goossen, "Overview of health care and nursing informatics in the netherlands," *Health Informatics Journal*, vol. 2, no. 1, pp. 9–20, 1996.

[11] D. H. Gustafson, R. P. Hawkins, E. W. Boberg, F. McTavish, B. Owens, M. Wise, H. Berhe, and S. Pingree, "Chess: ten years of research and development in consumer health informatics for broad populations, including the underserved." *Studies In Health Technology And Informatics*, vol. 65, no. 3, pp. 169–177, 2002.

[12] A. Hasman, *Education and training in health informatics in Europe: state of the art, guidelines, applications*, ser. Studies in health technology and informatics. IOS Press, 1995.

[13] R. Haux, "Aims and tasks of medical informatics," *International Journal of Medical Informatics*, vol. 44, no. 1, pp. 9–20, 1997.

[14] R. Haux, "Biomedical and health informatics education at umit - approaches and strategies at a newly founded university," *International Journal of Medical Informatics*, vol. 73, no. 2, pp. 127–138, 2004.

[15] W. Hersh, A. Margolis, and P. Quiros, F. andOtero, "Building a health informatics workforce in developing countries," *Health Affairs*, vol. 29, no. 2, pp. 274–277, 2010.

[16] U.S. Department of Health and Human Services. (2012, Mar.). *Health Information Privacy.* [Online]. Available: http://www.hhs.gov/ocr/privacy/

[17] L. M. Hilz and C. B. Thompson, "Expanding horizons in nursing: The clinical informatics graduate program," *Online Journal of Nursing Informatics*, vol. 4, no. 1, pp. 1–16, 2000.

[18] P. G. Hinegardner and P. S. Lansing, "Nursing informatics programs at the university of maryland at baltimore," *Bulletin of the Medical Library Association*, vol. 82, no. 4, pp. 441–443, 1994.

[19] HL. Seven International. (2012, Mar.). *HL7 - health level seven.* [Online]. Available: http://www.hl7.org/

[20] P. II, "Medicare and medicaid programs; electronic health record incentive program. final rule." *Federal Register*, vol. 75, no. 144, pp. 44 313–44 588, 2010.

[21] J. Murphy, "The journey to meaningful use of electronic health records." *Nursing economic*, vol. 28, no. 4, pp. 283–286, 2010.

[22] U.S. Department of Labor. (2012, Mar.). Bureau of Labor Statistics. [Online]. Available: http://www.bls.gov/

[23] Y. Shahar, "Medical informatics: between science and engineering, between academia and industry." *Methods of Information Medicine*, vol. 41, no. 1, pp. 8–10, 2002.

[24] W. A. Yasnoff, P. W. O'Carroll, D. Koo, R. Linkins, and E. M. Kilbourne, "Public health informatics: Improving and transforming public health in the information age," *Journal of Public Health Management Practice,*, vol. 6, no. 6, pp. 67–75, 2000.

Communication Analysis through PBL

Norimichi Hiroshige, Weiqiang Kong, Shigeru Kusakabe, Tsunenori Mine,
Naoyasu Ubayashi, Akira Fukuda, and Keijiro Araki
Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University, Japan
744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan
{*hirosige, weiqiang*}*@qito.kyushu-u.ac.jp,*
{*kusakabe, mine, ubayashi, fukuda*}*@ait.kyushu-u.ac.jp, araki@csce.kyushu-u.ac.jp*

***Abstract*—PBL (Project Based Learning) has gained remarkable attention as an education method for fostering professionals in the field of ICT (Information and Communication Technology). It is expected that students could develop, through PBL, skills related to technologies in software development and project management, and moreover, communication skills in projects. Lecture contents and competency assessment methods have been well studied and become relatively clear with respect to developing skills of software development and project management. But this is not the case for communication skills, and it is thus hard to evaluate how students' communication skills change and progress through PBL. Based on meeting minutes that were taken in a PBL program carried out in half a year, we investigate this issue and report our results that relatively deeper (with respect to content profundity) communication has become possible.**

***Keywords*-Communication Analysis, Project Based Learning, Information and Communication Technology**

I. The PBL Program in Kyushu University

Information and Communication Technology (ICT) serves as a fundamental infrastructure that supports modern society and industry. In Japan, a program has been promoted since around 2006 primarily by MEXT and Keidanren[1], with multiple Japanese universities selected as target hubs of the program for fostering advanced ICT professionals. As a member university of the program, Kyushu University established and has been operating a Social Information Systems Engineering Course [2] in the Graduate School of Information Science and Electrical Engineering.

PBL is introduced into the course as one of the primary education methods. In general, each student group, consisting of approximately five members, concentrates on one subject in a duration from half a year to one year. The subjects are practical ones from companies engaging mainly in system development, the ones proposed by teachers, or the ones that students wish to devote theirselves to. Students are assigned to the group of their own desire.

II. PBL and Communication

Students are expected to learn and develop the following skills through PBL:

- Technology skills centered on software development;
- Project management skills;
- Communication skills necessary for advancing projects.

Among these, quite a lot of detailed and concrete educational materials have been available with respect to developing software development skills and project management skills, and assessment could be easily made for students' competency levels.

Regarding communication skills, however, the situation is opposite, that is, concrete educational materials are rare to use and assessing competency levels is extremely difficult. In particular, although PBL has been adopted in university curriculums, it is unclear how to evaluate whether students' communication skills have progressed and whether change has taken place. Therefore, we attempt to investigate and address this issue by carrying out our PBL program.

III. The Investigation Method

As PBL progresses, students may apply various means of communication such as oral communication, emails, or SNS etc., and thus, taking all communication as our analysis target is infeasible. However, there is a high ratio of oral communication among other means in our program since students conduct approximately 20 hours face-to-face groupwork every week. Therefore, we considered it as feasible to measure, by analyzing contents of meetings, how students' communication skills change, and thus carried out a quantitative analysis based on meeting minutes.

Students hold various meetings in PBL, and these meetings could be generally classified into two types. One is held within respective groups (called *group meetings* hereafter), that is, participants of the meetings are group members only; another is held between group members and *customers* (called *customer meetings* hereafter). Customers are university teachers or professionals from companies, who have proposed and offered the subjects to students. In a customer meeting, communication is conducted between customers (who are quite familiar with the subject of the group) and students (who are beginners with respect to technologies). Therefore, through analyzing minutes of customer meetings, we could investigate how communication between customers and students has changed.

[1]MEXT is short for Ministry of Education, Culture, Sports, Science and Technology in Japan. Keidanren is a comprehensive economic organization with 1281 companies as its members (www.keidanren.or.jp/english).

In addition to group meetings and customer meetings, there are also meetings in which project achievements are checked (reviewed). However, since these two meetings cost more than half of the overall meeting hours, we thus just focus on analyzing them.

IV. Investigation Targets

A. Groups

There are four student groups, each consisting of approximately five members and undertaking one research subject. Although all subjects in our PBL program are related to software development, their concrete research contents differs from groups. In addition, professionals from companies or university teachers, who have proposed the subjects for students, are assigned to corresponding groups to serve as group customers.

B. Meetings

As mentioned above, there are two types of meetings held by students.

1) Group Meetings: The kind of meetings whose participants are group student members only. For example, regarding software development experience, although there exists somewhat difference between students, students have at least development experience with Java; in addition, regarding specific knowledge related to the subjects of PBL (namely domain knowledge), almost all students are beginners. In this way, we can claim that communication between students who master technologies and knowledge of basically same levels is conducted in group meetings. By analyzing minutes of group meetings, we could investigate how communication among students has changed.

2) Customer Meetings: The kind of meetings whose participants are students together with company professionals or university teachers who serve as customers. Such meetings have been held mostly based on a once-a-week rule, although differ from group to group.

Customers certainly have thorough knowledge of the domains and technologies related to their group's research subject, but on the contrary, students only have a shallow understanding of the domains and related technologies. In this way, we could claim that communication between customers with high technological knowledge and students of beginner level is conducted in customer meetings. By analyzing minutes of customer meetings, we could investigate how communication between students and customers has changed.

V. Hypotheses

A. Communication among Students

As the progress of PBL, students may gain deepened domain knowledge and related technological knowledge compared to their shallow status in early stages of PBL. Due to this fact, we anticipate that they could be able to gradually conduct deeper communication as the progress of PBL. More concretely, we consider that, in the early stages of PBL, there are few deep discussions that will result in important pointing-out or discovering bugs, and are many shallow discussions that just make simple confirmation. Moreover, we anticipate that the number of deep discussions will increase gradually as with the progress of PBL.

B. Communication between Students and Customers

Compared to customers, students just have shallow domain knowledge and technologies related to their subject in the early stages of PBL. Due to this fact, we consider that discussions, in which students make proposals to customers, state creative suggestions, behave spontaneously, or discover problems in customer meetings might be extremely few. However, we anticipate that the number of such discussions will increase as the progress of PBL.

In addition, when questions-and-answers is performed between students and customers with respect to each discussion topic, there are many cases in which students response to the question proposed by customers and such conversation exchange repeats. We define in this paper repetition times of such conversation exchange as *reply times*. Similarly, we anticipate that reply times are small in the early stages of PBL since it is difficult for students at those stages to positively response to customers. Moreover, as the progress of PBL, we anticipate that students could gradually answer positively to customers, and consequentially, reply times will be increased.

VI. Analysis Perspectives and Measure Methods

In general meetings, questions-and-answers surrounding one topic are conducted among multiple participants to reach a conclusion. We define such a settled discussion (i.e., reached to a conclusion) as a *discussion thread*, and conduct for each discussion thread the following analysis.

A. The Method for Analysis of Change in Communication among Students

If a discussion thread just contains the conversation of a simple question and its answer, we could say that this discussion thread is of a shallow level with respect to content profundity. One the other hand, if bugs are discovered in a discussion thread, we could state that this thread is of a deep level with respect to content profundity.

Therefore, we define three "Content Levels" (to be defined later), judge the levels that each discussion thread belongs to, and investigate how spread of discussion threads changes.

Definition of Content Levels. There are shallow contents (profundity) as well as deep contents (profundity) in discussion threads. For example, the contents are shallow for questions-and-answers which are made just for simple confirmation. On the other hand, the contents are deep for

Table I
CONTENT LEVELS

Content Levels	Explanation
1: questioning (shallow)	A level of which conversations are made by speakers just for simple questioning or confirmation
2: pointing-out (middle)	A level of which bugs are not discovered as the discussion results, but realize items that will bring great influence due to low efficiency if just leave them as they are
3: discovering bugs (deep)	A level of which bugs are discovered as the discussion results

Table II
MEETING ABILITIES

Proposal ability	Whether technological ideas are proposed (Yes=1, No=0)
Creative ability	Whether creative or constructive suggestions are made (Yes=1, No=0)
Spontaneous ability	Whether spontaneous actions such as undertaking tasks by oneself etc. are accompanied (Yes=1, No=0)
Problem discovery ability	Whether problems are discovered (Yes=1, No=0)

discussion threads in which bugs are discovered as the discussion results. As such, content profundities of discussion threads are different. We define three levels in Table I. According to content profundity, each discussion thread is assigned a content level. In group and customer meetings, the content profundity level "discover bugs" is higher (or deeper) than the level "questioning".

B. The Method for Analysis of Communication between Customers and Students (I)

In customer meetings, discussions, in which students could make proposals (including technical ideas), state creative suggestions, behave spontaneously, or discover problems, to customers who have high technological skills, are desired. In other words, we could claim that a discussion thread is a good one if those mentioned elements could be found from the thread.

Therefore, we define four "Meeting Abilities" (to be defined later), count the numbers that each discussion thread belongs to, and investigate how spread of discussion threads changes.

Definition of Meeting Abilities. We judge a discussion thread as good communication if conversations related to technical ideas appear in the thread. Based on a similar consideration, we define four evaluation points in Table II, and judge whether the four points appear in each discussion thread. These four points are named as Meeting Abilities.

C. The Method for Analysis of Communication between Customers and Students (II)

In order to measure reply times between customers and students in customer meetings, we have taken detailed speaking-log level minutes, and counted reply times.

For example, initially customers start a conversation by posing a question to students. If students answer to this question, reply times are set as 2. Furthermore, if customers reply to the answer from students, the reply times are set to 3, and so on. There are twenty discussion threads if twenty topics are discussed in a meeting. We count for each discussion thread the reply times, and analyze its minimum, average, and maximum values. Since the workload for taking such minutes is quite heavy, we conduct this analysis for three groups only.

D. Measure Methods

The meeting-minutes format that incorporates the above mentioned perspectives are prepared and used by four groups of 2nd-year master students (in the 2nd semester of 2010) in two kinds of meetings – *group meetings* and *customer meetings*. We proceed with our analysis by arranging measure items to each phase involved in a group project.

VII. ANALYSIS

Among the analysis results, we report those cases from which results of great interest are obtained.

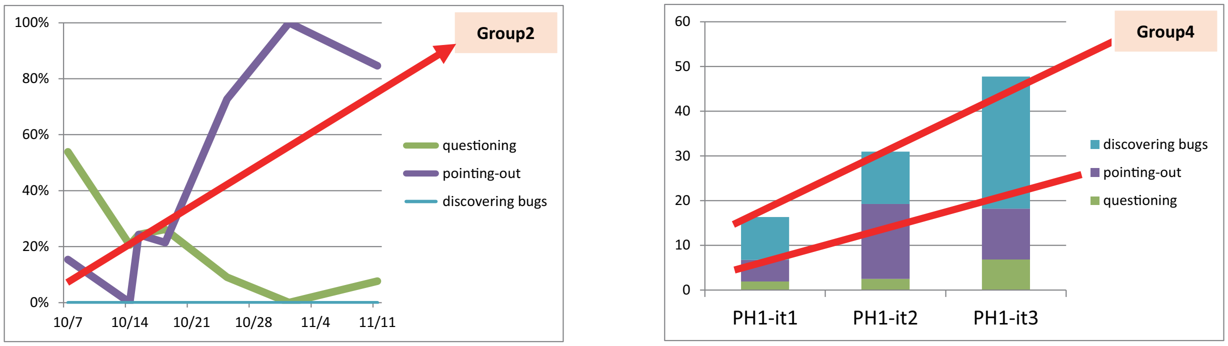


Figure 1. Analysis Results for Group2 and Group4

A. *Communication among Students*

Interesting results are obtained for two groups, which are illustrated in Figure 1.

First, regarding content level of Group2 (left-hand side of Figure 1), although there were many shallow level discussions like *questioning*, the percentage of *pointing-out* level discussions became higher gradually. Based on this, we could make the judgment that deep communication could be performed gradually in student meetings.

Second, the number of discussions in Group4 (right-hand side of Figure 1) increased in accordance with iteration[2]. In addition, the percentage of *discovering bugs* discussions has also increased to some extent. We therefore could make the same judgment as for Group2 that active and deep communication could be performed gradually in accordance with development iteration.

B. *Communication between Students and Customers*

For all the four groups, percentage of the items, from which abilities of proposal, creativity, spontaneousness, and problem discovery could be observed, remained in a dull and low level of 0%-10%.

We could make the judgment that, in half a year of PBL, students failed to reach a level that could make suggestions to customers and discover problems.

C. *Reply Times between Students and Customers*

This measure was carried out for Group3 only. On average, reply times were about five, the same as before. However, a trend of increase could be seen if we observe the maximum reply times as illustrated in Figure 2.

From this, we could make a judgment that tenacious discussions had been becoming possible between students and customers in customer meetings.

[2] A group project in PBL basically consists of multiple phases. Each phase contains several development iteration (i.e., repetition), e.g., to gradually introduce new functionalities etc. `PH` in Figure 1 denotes *phase* and `it` denotes *iteration*.

VIII. Investigation

The research subject of Group2 is to improve software quality by utilizing formal methods. In the survey phase, group members primarily learned the formal language VDM [1] and deepened their understanding of the target software system. Although there were initially many items of the level *questioning* in group meetings, the items of the level *pointing-out* (problems) had increased gradually. This indicated that, as the understanding to VDM and the target software system was deepened, group members were able to carry out deep discussions within group meetings, although they initially had not reached a level that could make suggestions or point out problems to customers.

Subject of Group4 is to develop airship control functionalities by utilizing the model-driven language SysML [3], with the eventual purpose to participate an airship competition. This group carried out development iteration for three times. As indicated by the analysis results shown in Figure 1, due to insufficiency of SysML studies and domain knowledge acquisition of airship competition, group members failed in early iteration to discover by themselves bugs that existed in design documents. However, it could be observed that bug discovery within group meetings had become possible in accordance with better understanding of SysML and acquisition of domain knowledge. This result conforms with the contents obtained from progress hearings with them, i.e., bugs existed in early iteration were discovered in group meetings held in later iteration.

Subject of Group3 is an industry development project. As a matter of fact, due to the high difficulty degree of requirement specifications, there were almost no conversations from students in customer meetings that showed proposal and creative abilities. This conforms with the results described in subsection VII.B.

However, although failed to make response to questions and requests from customers in early phases, students themselves did have impressions that they were able to do that in the later phases. This could also be seen from the results of *rally numbers between students and customers* shown in Figure 2.

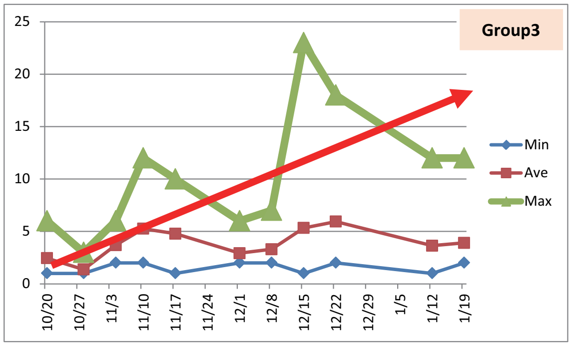


Figure 2. Analysis Result for Group3

IX. Results

By taking four groups in a half-a-year PBL program as targets, we have carried out a meeting-minutes based analysis of the communication among students, and respectively, between students and customers.

Regarding communication among students, we observed that three groups succeeded to conduct deep communication. Regarding communication between students and customers, there were no utterances that showed proposal ability and creative ability to the customers. However, we observed a trend that tenacious discussions had been becoming possible between students and customers.

X. Conclusion and Future Work

Results reported in the paper are based on a case study rather than an analysis with generality. According to phases and meeting-minutes contents (e.g., project management or technological contents), the purposes and perspectives may be different when considering communication. In addition, the ways of recording minutes may differ from groups or members. Taking the whole PBL process (in a half year) as a target, we conducted this time our analysis for each of its constituent phase, all based on meeting minutes. Next, we plan to carry out analysis by taking into account the above mentioned elements such as different perspectives etc. Moreover, extensions of our previous work [4], [5], [6], [7] related to this PBL program are also planned, in which meeting-minutes analysis may be included and treated as an important factor.

Acknowledgment

The research is carried out in the program of *Promotion Program to Foster Leading IT Specialists* supported by Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan.

References

[1] Martin C. Henson and Dines Bjørner. *Logic of Specification Languages*, Springer, 2008.

[2] Noriaki Sakamoto, Mitsuaki Fukase, Tsunenori Mine, Shigeru Kusakabe, Tsuneo Nakanishi, Yoichi Omori, Mohammad Mesbah Uddin, Keijiro Araki, Akira Fukuda, Hiroto Yasuura, and Teruaki Kitasuka. *Large Scale Business-Academia Collaboration in Master Education Course*. In the proceedings of the First International Conference on Computer Supported Education (CSEDU), pp. 159-166, 2009.

[3] Sanford Friedenthal, Alan Moore, and Rick Steiner. *A Practical Guide to SysML – The Systems Modeling Language*, Elsevier, 2009.

[4] Noriaki Sakamoto, Tsunenori Mine, Shigeru Kusakabe, Mitsuaki Fukase, Keijiro Araki, and Akira Fukuda. *Advantageous Effects of Internship Program for Fostering Highly Skilled ICT Personnel through Large Scale Business-academia Collaboration*. Journal of Information Processing, Vol. 49, No. 10, pp. 3388-3398, 2008. (In Japanese)

[5] Noriaki Sakamoto, Mitsuaki Fukase, Tsunenori Mine, Shigeru Kusakabe, Tsuneo Nakanishi, Yoichi Omori, Teruaki Kitasuka, Mohammad Mesbah Uddin, Keijiro Araki, Akira Fukuda, and Hiroto Yasuura. *An Approach to Foster Highly Skilled ICT Personnel through Large Scale Business-academia Collaboration*. Journal of Information Processing, Vol. 49, No. 8, pp. 2830-2842, 2008. (In Japanese)

[6] Norimichi Hiroshige, Naoyasu Ubayashi, Keiji Hokamura, and Akira Fukuda. *An Analysis about Effective Looking-Back in PBL*. In the proceedings of the workshop on Software Engineering Symposium (SES) 2010. (In Japanese)

[7] Norimichi Hiroshige, Naoyasu Ubayashi, Keiji Hokamura, and Akira Fukuda. *Evaluation of Project Based Learning in Kyushu University*. Research report of Information Processing Society of Japan 2010-SE-170(22), pp. 1-7, 2010. (In Japanese)

Developing Robot Programming Lab Projects

Xuejun Liang
Department of Computer Science, Jackson State University, Jackson, MS, USA

Abstract - *Robotics has played an important role in educations at different levels. But, robotics education at the college level is still ad-hoc. Many researchers have developed many great robotics course materials including lab projects. However, those materials are for teaching students at elite Research I universities rather than underrepresented students at Historically Black College and Universities (HBCUs). This paper presents ideas and details in adopting, revising, and developing robot programming lab projects that are suitable for underrepresented students at HBCUs.*

Keywords: Robotics Education, Robot Programming, Player/Stage, Tekkotsu

1 Introduction

Teaching an upper-level undergraduate robotics course at Historically Black Colleges and Universities (HBCUs) is challenging. The lack of suitable teaching materials is one of the biggest challenges, although there have been many great efforts in developing such robotics courses. For example, a Cognitive Robotics course has been developed at the Carnegie Mellon University [8, 11]. There is also an excellent list of robotics course materials [3]. But, those materials are prepared for teaching students at elite Research I institutions with a far more quick pace than in HBCU courses and were taught more on independent learning than in-class learning preferred by most HBCU students. Therefore, translating those materials into HBCU courses and making them suitable for HBCU students learning is necessary.

In addition, robot programming tasks in robotics competitions held along with the Annual ARTSI Student Research Conferences are valuable materials [1, 2]. But, it is difficult for our students to complete the whole tasks from scratch and without further detailed guidance, and therefore, there is a need to revise them in order to make them become useful and suitable robot programming lab projects.

In the rest of this paper, a brief overview of our robotics course will be given in Section 2. Robot programming with using the Player/Stage and with using the Tekkotsu will be introduced in Section 3. Three robot programming projects with using the Player/stage and two with using the Tekkotsu will be discussed in Section 4 and Section 5, respectively. The discussions for each project will focus on two aspects: 1. task and detailed steps to guide students to accomplish it, and 2. necessary knowledge for completing the projects, including programming skills, mathematical formulas, algorithms, and issues regarding to failures and uncertainty. Finally, a short discussion and conclusion is presented.

2 Overview of the Robotics Course

The robotics course is designed as an elective course for both senior undergraduate students and graduate students of Computer Science. It covers major topics on intelligent mobile robotics, including robot control architectures, sensing, localization, navigation, planning, and uncertainty. The course also reviews programming fundamentals in C++ language and introduces two robot programming software packages: the Player/Stage and the Tekkotsu. Students are evaluated on their homework assignments on major robotics topics, robot programming projects, midterm examination, and final examination. The course has been offered three times in the fall semesters over the last three years. The robot programming projects in each semester are updated with adding and/or removing some projects. The three course websites, one for each semester, are available to the public [5].

The robot platform used in our robotics course is the iRobot Create robot with the ASUS Eee PC on top of it [10]. As mentioned above, two robot programming software packages used in our robotics course are the Player/Stage [4, 9] and the Tekkotsu [8, 11]. In this paper, five robot programming projects will be discussed. Three of them are adopted and revised from Prof. Parker's robotics course entitled Software for Intelligent Robots [7]. They are waypoint following with using odometry data, targets searching and approaching by using behavior coordinating, and metric path planning by using wavefront algorithm. These three projects use the Player/Stage. The other two projects are developed based on the tasks in robotics competitions held along with the Annual ARTSI Student Conferences [1, 2]. In these two projects, a robot needs to navigate and localize itself within a maze, and to announce detected objects and their locations in the maze. But, the objects inside the maze and the navigation markers on the walls of the maze are different in these two projects.

3 Robot Programming Platforms

The iRobot Create is a popular mobile robot in robotics education. It uses differential drive and equips buttons for power, play, advance, and wheel drops (front, left, and right), bumps (left and right), IR sensor, wall sensor, cliffs sensors (left, front left, right, front right), encoders (distance, angle), and Leds. An ASUS notebook computer sitting on top of iRobot Create is functioned as the brain of the robot. The Player/Stage and the Tekkotsu are installed on the ASUS computer.

3.1 Robot Programming with the Player/Stage

As shown in Figure 1, Player server provides a network interface to a variety of real or simulated robots and sensor hardware. It commands robots and gets sensor data through device-specific connections. User robot control programs (clients) communicate with Player server through the TCP connection and hence can run on any computer with a network connection to the robot (Player server).

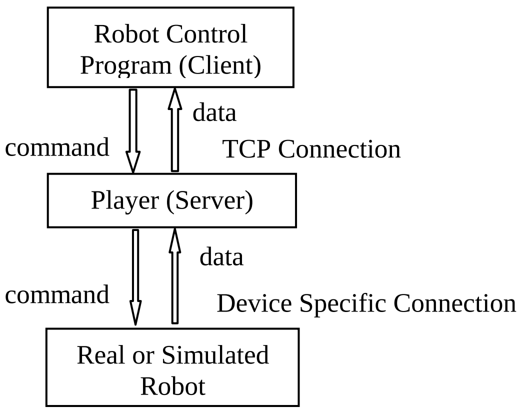


Figure 1: Player System Structure

Stage simulates a population of mobile robots moving in and sensing a two-dimensional bitmapped environment. Various sensor models are provided, including sonar, scanning laser rangefinder, pan-tilt-zoom camera with color blob detection and odometry. Stage devices present a standard Player interface. Few or no changes are required to move between simulation and hardware.

Programming with the Player/Stage, users need to provide two important files: world file and configuration file. The world (.world) file is needed when doing simulation using Stage. It describes things available in the world, including robots, items, and layout of the world. The configuration (.cfg) file contains the robot information called drivers and items in the world file if the client code interacts with them. The real robot drivers are needed to build in Player already and the simulation driver is always Stage.

The user robot control program is a client of the Player server. The client code receives inputs from sensors and controls hardware on robot through so called proxies. So learning proxies is needed to create client codes. Example 1 lists the main function of a client code in C++, which drivers the robot randomly without stop. This code shows a typical program structure of a client code, in which robots are defined and connected to the device proxies, and then in a control loop, sensing and acting interact with proxies. In the control loop of this example, the client code updates the proxies, generates a new random vector of distance and direction of motion every 3 seconds, and then translates it to a speed and a turn rate to drive the robot

```
int main(int argc, char *argv[])
{
  int randcount = 0;
  double speed, turnrate;
  Vector random(0,0);

  PlayerClient robot("localhost");
  Position2dProxy pp(&robot, 0);

  pp.SetMotorEnable (true);
  while(true)
  {
    // update the proxies
    robot.Read();

    // generate a random vector
    wander(randcount, random);

    // compute the speed and the turn rate
    translate(random, speed, turnrate);

    // command the motors
    pp.SetSpeed(speed, turnrate);
  }
}
```

Example 1: Main Function of Random Walk

Note that the control loop runs at about 10Hz, so 30 loop iterations is about 3 seconds. Hence, the wander function generates a new random vector when it is called every 30 times. This is implemented by using the function parameter randcount. Also note that the translate function applies the servo-loop control rules to compute the speed and the turn rate based on the distance and the direction of motion. In this example, it could be simple if the wander function generates a speed and a turn rate directly without using the translate function. But, when more robot behaviors, such as avoid obstacles, are needed, each behavior only needs to produce a vector of distance and direction of motion. Then, vectors from each behavior can be combined together as a single vector by using weighted vector addition.

3.2 Robot Programming with the Tekkotsu

Tekkotsu is an application development framework for mobile robots. It provides (1) lower level primitives for sensory processing, smooth control of effectors, and event-based communication, (2) higher level facilities, including an hierarchical state machine formalism for managing control flow in the application, a vision system, and an automatically maintained world map, (3) housekeeping and utility functions, such as timers and profilers, and (4) the newly added Tekkotsu crew [12], which enables programmers to use the built-in higher level robotic functions such as map-making, localization, and path planning.

Tekkotsu is object-oriented, making extensive use of C++ templates, multiple inheritance, and polymorphism (operator overloading). To write a robot control program, users need to define subclasses that inherit from the Tekkotsu base classes, and override any member functions requiring customization.

Two types of fundamental classes in Tekkotsu are behaviors and events. Users need to know the way to response or act when a behavior is constructed, activated, and deactivated, the way for a behavior to listen to events and to process events, and the ways to construct a state machine in Tekkotsu. Users also need to know the concepts of generator, source, and type of an event. Furthermore, when using the Tekkotsu crew, users need to know how to use different types of maps, how to localize the robot, how to detect and/or move to an object of interest, and how to get the location and shape information of objects of interest.

```
$nodeclass Randomwalk : StateNode {
  $nodeclass RandomNode: StateNode : {
    Vector random;
    virtual void doStart() {
      wander(random);
      postStateSignal<Vector>(random);
    }
  }
  $nodeclass TranslateNode: StateNode : {
    Vector random, result;
    double speed, turnrate;
    virtual void doStart() {
      random = extractSignal<Vector>(event);
      translate(random, speed, turnrate);
      result.setVector(speed, turnrate);
      postStateSignal<Vector>(result);
    }
  }
  $nodeclass DriveNode : WalkNode : {
    Vector result;
    virtual void doStart() {
      result = extractSignal<Vector>(event);
      double speed = result.getMagnitude();
      double turnrate = result.getDirection();
      setVelocity(speed, 0, turnrate);
    }
  }
  virtual void setup() {
    $statemachine {
      random: RandomNode
      translate: TranslateNode
      drive: DriveNode;
      random =S<Vector>=> translate
      translate =S<Vector>=> drive
      drive =T(3000)=> random
    }
  }
};
REGISTER_BEHAVIOR(Randomwalk);
```

Example 2: State Machine of Random Walk

In a Tekkotsu state machine, each state has an associated action: speak, move, etc. and transitions are triggered by sensory events, timers, or user's signals. State nodes are behaviors. Entering a state is activating it, and leaving a state is deactivating it. Transitions are also behaviors. A transition starts to work whenever its source state node becomes active. Transitions listen to sensors, timer, or other events, and when their conditions are met, they fire. When a transition fires, it deactivates its source node(s) and then activates its destination node(s).

A shorthand notation is used instead of C++ code to build state machines. The shorthand includes the state node definition, the transition definition, and the state node class definition. The shorthand is turned into C++ by a state machine compiler.

Example 2 lists the major part of a Tekkotsu state machine code of random walk. This application defines three state node classes and a state machine with three state nodes and three transitions. The random node generates a random vector of distance and direction of motion and then send the vector to the translate node, which computes a vector of speed and turn rate and then send the vector to the drive node, which drives the robot with the speed and turn rate for three second (3000 ms) and then transits back to the random node. Note that this application program has the same behavior as the client code in Example 1.

In addition to the concepts mentioned above, the following three basic skills of programming with the Tekkotsu are very important: (1) how to transit from one state to multiple states simultaneously so as to support parallel actions or behaviors, (2) how to transit from one state to one of multiple states based on different conditions so as to make a conditional transition, and (3) how to pass and/or share data among states so as to provide approaches of the data flow and the memory.

4 Projects with Using the Player/Stage

This section will present details of the three robot programming projects with using the Player/Stage. The three projects are Waypoint following, target searching, and path planning.

4.1 P1: Waypoint Following

The task of this project is to read a sequence of waypoints from a data file and then drive the robot to each waypoint one after another. The project is required to use the robot's odometry data and the servo-loop control approach. The following steps will guide students walking through the project.

1. Give students a skeleton world file and let them add details in the given file according to the requirements such as world size, simulation window size, etc.
2. Give students a skeleton client code which provides the program structure. The client program gets a sequence of waypoints from a file by calling the getWaypoints function and then enters an outer loop to iterate through each waypoint. The inner loop is a control loop to drive the robot to a waypoint, which is very similar to the control loop in Example 1. But, the loop will now terminate after the robot reaches to the waypoint and the wander function in Example 1 is now replaced by the gotoWaypoint function. Students only need to complete the following three C++ functions:

- getWaypoints. It reads a sequence of waypoints from a data file and store them into a queue.
- gotoWaypoint. It computes the distance and angle from the robot's current pose to the waypoint. If the distance is small enough, then return true. This will indicate that the robot has reached the waypoint. Otherwise, return false.
- translate. It is the same as the one in Example 1.

Please note that in order to help our students to complete these three functions, several points are worth to mention. First, students need to have C++ programming skills on File I/O and using the queue data structure, and understand the call-by-value and call-by-reference. Second, students need to know how to obtain the robot's current pose from the robot's odometry (encoder) data. Third, students need to know how to compute the distance between two points, the slope of a straight line, and the angle between two lines, and understand the difference between degree and radian. Fourth, students need to learn the rule-based servo-loop control approach in order to determine an updated speed and turn rate to drive the robot based on the distance and angle between robot and waypoint.

The second task of this project is, instead of simulation, to drive a real iRobot Create robot to follow a square path by using the same client program. Students are also asked to measure the odometry error and make changes to their client code to improve the performance. This exercise makes students get better understanding of uncertainty of sensory data.

4.2 P2: Target Searching

This project will generate robot control by sequencing and combining three behaviors: wander, avoid obstacles, and go to beacons. The final robot behavior should cause the robot to wander around avoiding obstacles until it detects a (for graduate students, previously unvisited) beacon, then move to the location of the beacon. Once the beacon is reached, the robot should go back into wander mode to search for another beacon. This will repeat infinitely. The following steps will guide students walking through the project.

1. Give students a skeleton world file and let them add details for defining and creating beacons and robot.
2. Give students a skeleton client code which provides the program structure, which is very similar to Example 1. In the control loop, after updating the proxies, three functions: wander, avoidObstacles, and gotoBeacon are called to active the three behaviors. Then, if no beacon is found, combine (weighted vector sum) the outputs from wander and avoidObstacles to get a single output vector, otherwise, combine outputs from gotoBeacon and avoidObstacles. Finally, the translate function is called to get a speed and a turn rate to drive the robot. Students only need to complete the following three C++ functions:
   - wander. It is the same as the one in Example 1.
   - avoidObstacles. It generates a vector of distance and direction of motion to avoid the obstacle, if there is an obstacle in front detected by using laser scanner. Otherwise, it generates a zero vector. Note that once starting avoiding, it is needed to continue avoiding for two seconds (the control loop runs at about 10Hz, so 20 loop iterations is about 2 second.)
   - gotoBeacon. It returns true whenever a (for graduate students, unvisited) beacon is detected by using Fiducial detector. Otherwise, return false. When a (for graduate students, unvisited) beacon is detected, it computes a vector of distance and angle from the robot's current pose to the beacon.

Note that the translate function has been implemented in P1. The function to combine (weighted sum) two vectors is given. In order to help our students to complete these three C++ functions, several points are worth to mention. First, students need to know how to produce a random number in a given range and how to do the coordinate transformation (rotation and shift) between the world coordinate and the robot's coordinate. Second, students need to know the implementation skills for the wander function to generate a new random vector when it is called every 30 times and for the avoidObstacle function to continue generating the same vector for 20 times once starting avoiding. Third, students need to know how to use the list data structure in C++.

4.3 P3: Path Planning

The task of this project is to implement the wavefront path planer. The path planner accepts as input a user's goal point and generates the waypoints of a path from a given starting point to the goal point. Then, the avoid obstacles behavior and the go to waypoint behavior from the previous projects are used to drive a robot to follow the path. The following steps will guide students walking through the project.

1. Give students a skeleton world file and let them fill in information for simulation window and map.
2. Give students a skeleton code of the wavefront path planner, which provides all detailed implementation of the planner except the following three functions, which are left for the student implementation.
   - grow. It grows the obstacles in the grid map for a single step, i.e. one grid cell farther. It scans the grid cells. If a cell is occupied, then mark the unoccupied neighbors of the cell as occupied.
   - propagate. It propagates the wavefront one grid cell farther. It starts from the grid cells with value i and the propagated cells get the value i+1. If the cell of robot's starting point has the same value i, then the wavefront propagation should be over, otherwise, try to propagate the wavefront to its neighbors. If there is no room to

propagate the wavefront, then the robot's goal point is unreachable; otherwise, the wavefront function can be called again and starts from the grid cells with value i+1.

- nextWaypoint. It computes the next waypoint. The next waypoint is a neighboring cell of the current waypoint and its value is 1 less than the value of the current waypoint cell. This function is called first with the current waypoint equal to the robot's starting point. It is called continuously until the new waypoint is the goal point.

Note that the implemented portion of the planner for students includes the conversion between the pixel map representation and the grid map representation of an image, the conversion between the world coordinate and the image coordinate, and the waypoints relaxing. In order to help our students to complete the three functions, two points are still needed to make. First, students need to know the grid cell representation of an image, the process of scanning the grid cells, and how to process the edge cells and corner cells easily. Second, students need to know what is the logic that makes the statement "there is no room to propagate the wavefront" to be true. In addition, students really need to pay attention to array boundaries.

5 Projects with Using the Tekkotsu

This section will present details of the two robot programming projects with using the Tekkotsu. These two projects have the same task, which is locating objects inside a maze.

5.1 P4: Locate Objects Inside a Maze

The task of this project is taken from the robotics competition held along with the 2nd Annual ARTSI Student Research Conference [1]. It is to get an iRobot Create/ASUS robot to navigate and localize itself within a maze, to announce detected objects and their locations in the maze.

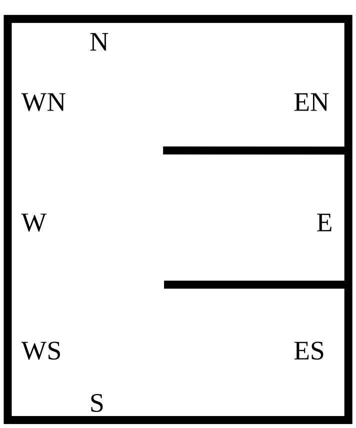


Figure 2: E-Maze with 8 Navigation Markers

The maze is shaped like the letter E as shown in Figure 2: a long corridor with three alcoves branching off from it. There is a bicolor marker at each end of each alcove or corridor, for a total of 8 markers: NE, N, NW, W, SW, S, SE, and E. Objects placed in the alcoves will be red, blue, or green balls, one per alcove. The project is graded on visiting each alcove of the maze, announcing detected objects and their locations, delivering a final report, and the overall run time.

This project is divided into three parts: (1) Travel though the maze by modifying the "following wall" behavior, (2) Travel though the maze and announce the balls the robot detects, and (3) Travel though the maze and report the balls the robot detects and their locations.

In Part 1, students need first to run a sample Tekkotsu's wall following program. Students will notice that the robot will not able to follow the wall unless its right side is placed near a wall, and the robot will not stop. Then, students are required to modify the sample code so that the robot will perform (a) go to a wall, (b) follow the wall, and (c) stop after a while. So the robot can travel through the maze along the walls regardless where the robot is placed inside the maze. Note that a state machine diagram to accomplish this task is given to students.

In Part 2, a sample Tekkotsu state machine code of looking for balls is given to students. This behavior will announce the balls in front of the robot. Students need to test the code with several runs, each run with different sets of color balls and different lighting conditions. Students may need to change the ASUS camera settings through Tekkotsu. Then, students are asked to add the looking for balls behavior in the state machine code in Part 1. So a robot can look for balls while it travels through the maze. Note that looking for balls and traveling through the maze are two parallel behaviors and the former behavior will send a signal to the latter behavior to stop the robot after the robot finds out the three balls. A state machine diagram to accomplish this task is given to students.

Please note that the robot may see the same ball several times. So the robot must have memory to remember which ball it has already seen. Meanwhile the robot should also remember the ordering in which it sees each of the three balls. To this end, a scheme of encoding the three colors is provided to students. In this scheme, Red is labeled as 1, Blue is 2, and Green is 4. A set of colors, for example, {Red, Green} will be labeled as 5 (1+4). A pair of colors, for example, (Blue, Green) will be labeled as 24 (2×10+4). A 3-tuple of colors, for example, (Blue, Green, Red) will be labeled as 241 (2×100+4×10+1).

In Part 3, a sample Tekkotsu state machine code of visiting markers is given to students. In this behavior, the robot will search for a bicolor marker by turning in its place and then move towards the marker; if the robot cannot find a marker after a while, it will move forward. In both cases, the robot will move forward and hit a wall of the maze. Students need to test this code first with the robot placed in different locations inside the maze and different lighting conditions. Students may need to change the ASUS camera settings.

Then, students are asked to modify the sample state machine code so that the robot will only find the bicolor marker at south, at southwest, at west, or at northwest. This means that the markers at the remaining four locations are ignored. Finally, students are asked to add the modified visiting marker behavior into the state machine code in Part 2. Then, students need to modify the announcement and the final report to include the location information of the detected balls. Note that the first ball the robot sees must be in the south alcove, the second must in the middle alcove, and the third must be in the north alcove, if one of the four bicolor markers at south, southwest, west, or northwest is found.

In addition to the above guidance, students must have basic object-oriented programming skills such as deriving a subclass and instantiating a class. Students must also learn Tekkotsu programming basics: behaviors, event, predefined state node classes and transition classes, and state machine formulations and semantics. Moreover, students need to know how to deal with uncertainty and failure. For example, bicolor markers are difficult to detect (false negative) and background blobs are easily recognized as a color ball (false positive).

5.2 P5: Another Version of P4

The task of this project is taken from the robotics competition held along with the 3rd Annual ARTSI Student Research Conference [2]. This task is the same as the one in P4 except the bicolor makers are replaced by the AprilTags [6] and the three color balls are replaced by three cubes with each covered by different AprilTags. Note that there are 20 navigation markers (AprilTags) on the walls of the E-maze as shown in Figure 3 so as to support a better result of the robot localization.

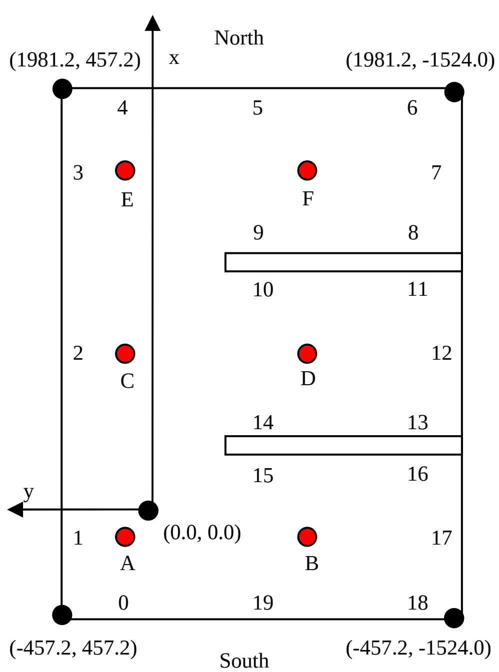


Figure 3: E-Maze, Twenty Navigation Markers (0-19), Six Waypoints (A-F), and World Coordinates of the Four Corners

In P4, each bicolor marker is treated as a topological navigation marker and the measurements of the maze are not used at all. On the other hand, in P5, a metric map of the maze as shown in Figure 3 is used in the robot localization. Meanwhile, moving robot to a particular waypoint inside the maze can be done easily by calling a pilot function that is newly added into Tekkotsu.

This project is divided into four parts: (1) Memorize and report which object the robot has observed in each alcove, (2) Look for the object in each alcove, (3) Drive robot to a waypoint in the maze, and (4) Putting it together.

In Part 1, students need first to run a sample Tekkotsu's pilot demonstration program. Students can use commands provided in this program to drive the robot forward or making a turn, check the robot location, localize the robot, and look AprilTags that are facing to the robot and report their IDs.

In this part, students are asked to add a command in this sample code to report which object the robot has observed in each alcove. Here, the robot needs to figure out which object in which alcove. We assume that when the robot is seeing an object in an alcove, it will also see at least one navigation marker on the wall of the same alcove. This may generally not be true. But, users can use commands to make the robot facing to an object and at least one marker at the same time. Note that students need to define three shared variables to remember the objects the robot has observed in each alcove.

In Part 2, students are asked to add a command to look for an object and its location. Now, we assume that the robot is inside an alcove, but it may not face an object. This new command allows the robot to look for an object for several times by making several turns until it finds one as well as its location or fails to find.

In Part 3, students are asked to add a goto <x y> command to drive the robot from its current position to location (x, y) inside the maze. Please note that the world coordinates of points in the maze should be used in the goto function. The coordinates of the four corners are shown in Figure 3. Note that the coordinates are using millimeters as measurement unit. Students can use the built-in path planning and execution function of the Tekkotsu pilot to implement the goto command. Or, students can implement the goto function on their own. In this case, we assume there is no obstacle in between the robot current location and the target location.

Due to the uncertainty, the robot may not be able to be close enough to the target location. In this case, the goto function should redo itself again until the robot is close to the target within a threshold distance (for example, 200 mm). If the robot is still not able to reach the target after redoing 3 times, the goto function should be stopped and return a failure.

Due to the same reason as above, the robot may hit walls before reaching to the target. In this case, the robot should backup a little bit and then the goto function should redo itself again. If the robot is still not able to reach the target after redoing 3 times, the goto function should be stopped and return a failure.

In Part 4, students are asked to put them together by creating a node class that takes a role of subtask scheduler. For example, a schedule can be goto A, goto B, look for object, goto A, goto C, goto D, look for object, … and finally report which object the robot has observed in each alcove. It is obvious that this schedule will let the robot find the object at each alcove, if each subtask is successfully executed.

6 Discussion and Conclusion

Our robotics course has offered three times in the fall semesters over three years starting from 2009. It covers all major topics on intelligent mobile robots, including robot control architectures, navigation, localization, planning, sensing, and uncertainty. A primary component of this course is robot programming. Over the three years of time, many robot programming projects have been adopted, revised, and developed for underrepresented students at HBCUs. In addition to the major projects presented in this paper, we have created several introductory projects in our robotics course for learning the basics of the Player/Stage and the Tekkotsu.

Teaching robot programming is challenging. More in-class teaching is always welcoming by students. Like most HBCUs, we don't have open labs and TAs for our robotics projects. Sometimes, the instructor has to spend extra hours in the lab to help students with their projects. Providing detailed and intuitive guidance to students is important and neccessary. Therefore, students will see the hope to complete the project and then spend their time doing the project.

In our robotics courses in 2009 and 2010, we used both Player/Stage and Tekkotsu. The first version of the project of localizing objects in a maze was added into our 2010 robotics course. But, we were left no time to do this project. In our 2011 robotics course, we used the Tekkotsu only. So, we got time to cover the second version of the project of localizing objects in a maze. Note that AprilTags are much easier to be detected than bicolor markers. This makes the robot localization more accurate. So, we can use the Tekkotsu built-in robot localization function in the second version. But, we can apply the method used in the first version to the second version without using the robot localization.

Compared with the Tekkotsu, the Player/Stage provides more general framework for robot programming. Users are easy to start with it and to test their own algorithms. On the other hand, the Tekkotsu builds a set of high-level interacting software components to relieve a programmer of the burden of specifying low-level robot behaviors [12]. This makes it possible to teach and practice more on robotics rather than programming details. But, there is a large learning curve to master the Tekkotsu fundamentals and its high-level software components. However, Tekkotsu is easy to use for the demonstration of robotics concepts. Meanwhile, the 3-D simulation software Mirage can be used along with the Tekkotsu for the simulation.

Acknowledgments

This work was supported by National Science Foundation award CNS-1042326. The author thanks Dr. David Touretzky for his help on the Tekkotsu.

7 References

[1] ARTSI Alliance, *2010 Robotics Competition*, Available at http://artsialliance.org/2010-Robotics-Competition

[2] ARTSI Alliance. *2011 Robotics Competition*. Available at http://artsialliance.org/2011-Robotics-Competition

[3] A. Dollar, D. Rus, and P. Fiorini, *Robotics Courseware*, Available at http://roboticscourseware.org

[4] B. Gerkey, R. Vaughan, K. Støy, A. Howard, G. Sukhatme, and M. Mataric, *Most Valuable Player: A Robot Device Server for Distributed Control*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), pp. 1226-1231, Wailea, Hawaii.

[5] X. Liang, *Introduction to Robotics* (*Fall 2011*, *Fall 2010*, *Fall 2009*), Available at http://www.jsums.edu/robotics/

[6] E. Olson, *AprilTag: A robust and flexible visual fiducial system*, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2011), Shanghai, China, May 2011.

[7] L. Parker, Software for Intelligent Robots, Available at http://roboticscourseware.org/fullcourses/mikey-test-course

[8] E. J. Tira-Thompson, G. V. Nickens, and D. S. Touretzky, *Extending Tekkotsu to new platforms for cognitive robotics*, Proceedings of the 2007 AAAI Mobile Robot Workshop, pp. 47-51, Menlo Park, CA.

[9] C. Toby, M. Bruce, and G. Brian, *Player 2.0: Toward a Practical Robot Programming Framework*, Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2005), Sydney, Australia.

[10] D. S. Touretzky, *iRobot Create/ASUS Notebook Platform for Tekkotsu*. Available at http://chiara-robot.org/Create/

[11] D. S. Touretzky, and E. J. Tira-Thompson, *Tekkotsu: A framework for AIBO cognitive robotics*, Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05), volume 4, pp. 1741-1742. Menlo Park, CA, 2005.

[12] D. S. Touretzky, and E. J. Tira-Thompson, *The Tekkotsu "Crew" Teaching Robot Programming At A Higher Level*, Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10), pp. 1908-1913, Atlanta, GA, 2010.

A Methodology for the Assessment of Pedagogic and Implementation Aspects of Laboratories

Muhammad Rashid[1], Imran A. Tasadduq[1], Yousuf I. Zia[1], Mohammad Al-Turkistany[1] and Saima Rashid[2]
[1]Computer Engineering Department, Umm Al-Qura University, Makkah, Saudi Arabia
[2]Education Department, Hazara University, Mansehra, Pakistan

Abstract—*Assessment of laboratories is an important requirement for any engineering accreditation program. However, a comprehensive assessment methodology is often ambiguous that can result in inconsistent and subjective assessments. Therefore, an assessment methodology is required to evaluate the effectiveness of laboratories. This paper presents a methodology for assessing the effectiveness of laboratories in terms of pedagogic as well as implementation aspects. The pedagogic aspects cover relationship between theory and laboratory practice, content level, activity level and learning environment. The implementation aspects cover facilities support, reliability, safety and laboratory manual. By assessing these pedagogic as well as implementation aspects, various practical abilities of the labs can be evaluated.*

Keywords: Engineering laboratories, assessment matrices, continuous improvement plan, ABET, accreditation

1. Introduction

Engineering is a practicing education where *doing* is important. Laboratories are essential elements in engineering because they involve a lot of practical *doing* [1]. While theory courses provide mathematical and scientific foundations, laboratories provide an ability to design and conduct experiments. The Criterion 3 (student outcomes) of Accreditation Board for Engineering and Technology (ABET) specifically states the ability to conduct experiments as well as analyze and interpret data as a major outcome of the engineering education process [2].

Despite the importance of laboratories, most of the attention has been given to theory courses and little has been written about laboratory instructions [3]. It may be due to the fact that there is hardly any career-related incentive for faculty to spend time in laboratory instructions. Therefore, this area of instruction is usually left to graduate students. We term it as a *neglected part of engineering educational research*. As an example, only less than 5 percent of the papers in Journal of Engineering Education from 2000 to 2010 used laboratory as a keyword [4].

In addition to the neglected behavior, very few research papers are available about the assessment of laboratories. A comprehensive summary of assessment methodologies is presented in [5]. However, these methodologies concern only the assessment of educational objectives of laboratories. For example, the work in [6] presents a qualitative assessment methodology for course outcomes from formal laboratory work products. Similarly, the work in [7] provides a methodology for assessing students' laboratory performance using Bayesian networks. Although, we presented an evaluation criterion for the assessment of engineering laboratories in our prior work [8]. However, our previous work was limited to the assessment of pedagogic aspects. To the best of our knowledge, no methodology exists for assessing the pedagogic as well as implementation aspects of laboratories. The lack of an established methodology may lead to subjective and inconsistent assessments.

Furthermore, Criterion 4 of ABET states that a program must show evidence of actions to improve the program [2]. In the context of laboratories, a continuous improvement plan consists of four steps as shown in Figure 1.

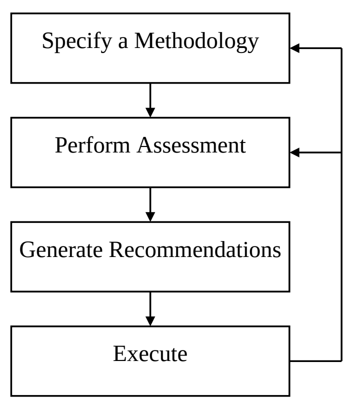


Fig. 1: Continuous Improvement Plan

It is evident from Figure 1 that a methodology is required before performing assessments. Based on the assessment results, recommendations are generated. In the execution phase, recommendations are implemented. The output from execution phase goes to step 1 (specify a methodology) or step 2 (perform assessments) to improve the effectiveness of laboratories in continuing fashion.

The objective of this paper is to propose an assessment methodology (step 1 in Figure 1) that covers pedagogic as well as implementation aspects of laboratory activities. Pedagogic aspects cover relationship between theory and laboratory practice, content level, activity level and learning environment. Implementation aspects cover facilities support, reliability, safety and laboratory manual. The proposed methodology is generic and can be applied to any engineering discipline. However, in order to illustrate the proposed methodology, examples are provided from the Computer Engineering (CE) domain. We have performed assessments (step 2 in Figure 1) for seven different CE laboratories. At the time of writing this paper, generation of recommendations phase (step 3 in Figure 1) and execution phase of the continuous improvement plan was in progress providing feedback to step 1 (specify a methodology) and step 2 (perform assessment). The details of these phases and feedback results will be reported in a follow-up paper.

2. Proposed Assessment Methodology

This section describes the proposed assessment methodology. It consists of two aspects: Pedagogic aspects (Section 2.1) and implementation aspects (Section 2.2).

2.1 Pedagogic Aspects

Pedagogic aspects include relationship between theory lecture and laboratory practice, content level, activity type and learning environment.

2.1.1 Relationship between theory and laboratory

There is a relationship between laboratory and the theory lecture that drives it. There can be four different possibilities:

a) Complete independence: Laboratory activities may be completely independent from the theory. In this situation, each laboratory depends only on the material covered in the previous laboratories. For example, a hardware description language (HDL) is taught in the laboratory environment. However, this completely independent laboratory experience is served as a pre-requisite for many other courses like digital system design, computer architecture, embedded systems design and so on.

b) Dependence across semesters: A second type of relationship between the theory lecture and laboratory is based on pre-requisites. The theory is covered during one term and the laboratory course is taken independently in another. For example, microprocessors and interfacing course provides the knowledge about programming for microprocessor-based design during one term. However, the projects based on this programming knowledge may be covered in the next term.

c) Loose coupling: In this format, the laboratory material and the theory lecture may or may not depend upon one another. For example, some experiments in a laboratory course may require a particular concept to be illustrated in the theory lecture. Similarly, the data analysis in some of the laboratory exercises may provide a solid ground for the subsequent theory lectures.

d) Close coupling: It implies that the curricula of the laboratory experience and the theory lecture are closely dependant. Instructor presents material in lecture that is supported by a laboratory experience. A typical example of this format is a programming language course. The instructor might discuss the laboratory material in the theory lecture, explaining what will happen in the next laboratory session, or review results from the previous session.

2.1.2 Content level

This attribute describes the type of content that is being addressed in the laboratory. We identify three major types:

a) Least sophisticated: The least sophisticated content involves purely mechanical knowledge of the system itself or of the application environment. For example, in a digital logic design laboratory, explaining to the students what an FPGA board kit is or how to use a development environment for software development in a specific programming language will be considered as least sophisticated contents.

b) Middle sophisticated: It involves implementation techniques within an environment. For example, in a digital logic design laboratory, middle sophisticated experiments will include basic building blocks of hardware description languages such as sequential and behavioral statements, test benches, synthesis process and so on.

c) Most sophisticated: It addresses problem solving to ensure that student can design the experiments as well as analyze and interpret data.

2.1.3 Activity type

While content level addresses the sophistication of contents, activity type on the other hand, describes the pedagogic activity in which students are engaged. We have identified the following activity types:

a) Tutorial: A tutorial can be used in a laboratory setting to provide the fundamental knowledge about laboratory safety precautions, required hardware resources, brief introduction of development environment and so on.

b) Project construction: The most obvious type of activity for a laboratory is to construct something new. Typically, students are given a problem and asked to develop a model to solve it and implement the solution. For example, in electronics laboratory, project construction may involve the construction of half wave and full wave rectifiers.

c) Experimentation and Analysis: In this case, students may first construct the solution, or they might be given the solution. The focus in this activity is to analyze the results. It allows the student to play with different variables in different scenarios. For example, half wave and full wave rectification experiment may analyse the effect of a reservoir capacitor upon the rectified waveform.

d) Exploration: The next logical step to experimentation and analysis is exploration. In this activity, students are given a scenario or a problem and are asked to provide theory and find answers. A typical example is the formulation of a mathematical relationship between a reservoir capacitor and the rectified waveform.

e) Interpretation: It involves interpreting the work of others. For example: understanding a piece of code and providing comments on it, studying a circuit design and explaining its functionality, reading documentation or a scholarly paper and so on. In addition to interpretation, students may learn to modify the work of others. An obvious activity is to correct code that contains educationally interesting bugs.

2.1.4 Learning environment

We focus on three aspects of learning environment [9][10]:

a) Teacher interaction: There may be three types of teacher interaction: graduate teaching assistant, a faculty member or a tutor with industrial experience. Graduate teaching assistants can be used when a moderate level of interaction is needed in the laboratory. A faculty member may be the laboratory instructor when a high level of interaction is needed. When a faculty member feels that students would benefit from contact with an individual who has industrial experience, a tutor with such an experience may be used as the laboratory instructor.

b) Student collaboration in laboratories: Different modes of student collaboration are possible in laboratories. We have identified two modes: students working independently and students working in groups. Advantage of students working independently is that each student gets the same experience and it is easier to assess student performance. The disadvantage is that the students do not gain the experience of working in collaborative groups, which is common in industry. Advantage of students working in groups is that students gain a deeper understanding of concepts when confronted with other ideas. They improve their communication skills, which is required by industry. The disadvantage is that all students do not gain the same experience and it is difficult to assess individual student contribution.

c) Assessment: There are many ways that an instructor can assess student performance in a laboratory. The method chosen depends on the nature of the laboratory assignment. For example examining a program and its output, reading and evaluating a formal lab report, asking questions about the laboratory exercise either in or outside of laboratory session, peer evaluation for group work and so on.

2.2 Implementation Aspects

Implementation aspects include facilities, reliability, safety and laboratory manuals. We have simplified the evaluation of implementation aspects of engineering laboratories by developing four different matrices as shown in Tables 1, 2, 3 and 4. The size of each matrix is variable depending on the required level of detail in the assessment.

Table 1: Facilities Support Matrix

S.No	Attributes	Yes = 1 No = 0
1	Is appropriate equipment available ?	
2	Is condition of equipment satisfactory ?	
3	Are data sheets available ?	
4	Is internet available ?	
5	Are computers connected on LAN ?	
6	Are multimedia available ?	
7	Is the space sufficient ?	
8	Are lab timings suitable ?	

Table 2: Maintenance and Reliability Matrix

S.No	Attributes	Yes = 1 No = 0
1	Is an administrator present ?	
2	Do system failure policies exist ?	
3	Do equipments provide reliable results	
4	Are faulty components replaced regularly ?	
5	Does a system exist for tracking ?	

Table 3: Safety Matrix

S.No	Attributes	Yes = 1 No = 0
	General Laboratory Safety Rules	
1	Are safety rules displayed ?	
2	Are labs properly locked ?	
3	Are laboratory hours displayed ?	
	Emergency Safety	
4	Is awareness present about fire	
5	Is awareness present about safety	
6	Is awareness present about evacuation?	
	Equipment Safety	
7	Are accessories properly operated ?	
8	Is awareness present about the waste?	
9	Are liquids properly handled ?	
10	Is equipment cleaned on regular basis ?	
11	Is proper temperature maintained ?	
12	Is anti-virus software installed ?	
	Power Supply Related Safety	
13	Is emergency power OFF installed ?	
14	Are protectors used to avoid surges ?	
15	Is equipment placed on static mates ?	

Table 4: Laboratory Manual Matrix

S.No	Attributes	Yes = 1 No = 0
1	Faculty is involved ?	
2	Manual is considering the context ?	
3	Manual is upgraded on regular basis ?	
4	Improvement plan is being followed ?	
5	Course co-coordinator is playing his role ?	
6	Is there feedback to course coordinator ?	
7	Quality control is being assured ?	

Each matrix consists of various attributes such that each attribute (a single element of a matrix) is a pre-determined question. Each question is a competency element that is assessed using a yes/no (1/0) scoring scheme. In other words, the value of each attribute is a binary number indicating the presence or absence of a certain attribute. The sum of the yes (1) scores becomes the grade for each matrix and the sum of the scores of four matrices becomes the final grade that can be awarded to the implementation aspects of laboratory. The following paragraphs describe four implementation aspects in terms of matrices.

2.2.1 Facilities support matrix

Proper facilities support improves the quality of learning and reduces the efforts. Adequate computing equipment must be available to support laboratory work. The amount of required equipment depends on the number of students involved and the estimated amount of time needed to complete each laboratory exercise. Table 1 shows the eight attributes of facilities support matrix that are required to be assessed.

2.2.2 Maintenance and reliability matrix

The technology must be reliable. Non-functional equipment or frequent crashes that cause loss of work will frustrate the students. Table 2 shows the maintenance and reliability matrix that enlists five attributes.

2.2.3 Safety matrix

A review of the literature on safety knowledge for engineering laboratories is presented in [11]. We have classified this knowledge into four categories as shown in Table 3.

a) General laboratory safety rules:

Safety rules must be displayed at a prominent location in the laboratory. Laboratories should be locked when unoccupied. The laboratory operating hours for students should be posted on the appropriate places.

b) Emergency safety:

Everyone present in the laboratory should be familiar with the locations and operation of safety and emergency equipment such as fire extinguishers, first aid kits, emergency power off system, fire alarm pull stations etc. Similarly, the familiarization with the emergency exits and evacuation plans is an important safety concern.

c) Equipment safety:

Proper handling of equipment is an important safety measure. For example, use of extension cords should be on temporary basis. Knowledge of correct handling, storage and disposal procedures for batteries, cells, capacitors, inductors and other high energy-storage devices is important. Equipment should be cleaned on a regular basis. Temperature in the room should stay cool to avoid overheating of equipment inside a lab. For labs with computers, external storage devices should be scanned for possible virus threats before using them.

d) Power supply related safety:

Every laboratory should be equipped with an emergency power off system. An electrical surge may cause a fire or shock the user. A surge protector is used to protect computers from electrical surges. Static electricity can damage electrical components inside computers. Consequently, the equipment should be placed on a static mat.

2.2.4 Laboratory manuals

Two good examples of developing laboratory manuals are presented in [12] and [13]. Consequently, a laboratory manual matrix used to evaluate its effectiveness is shown in Table 4. The challenges involved in the development of laboratory material are discussed below:

a) Faculty involvement: Developing laboratory materials require considerable amount of time. Since development of laboratory materials is not conventional research, so administrators usually does not give appropriate credit for the development of laboratory materials. As a result, faculty is generally reluctant to engage in difficult, time consuming activities if their efforts are not properly acknowledged in tenure, promotion, and salary considerations.

b) Context of contents: Due to the volatility of the engineering field, it is difficult to obtain *canned* materials. In other words, laboratory materials developed at one institution cannot be used in other institutions unless the context is identical.

c) Upgrading: The rapid evolution of the engineering field demands that laboratory materials must be upgraded regularly.

d) Continuous improvement plan: The development of laboratory manual is more properly viewed as an ongoing rather than a one-time effort. In the context of computer engineering, faculty is routinely expected to learn new computing hardware, operating systems, programming languages and system design tools in their spare time. Computing platforms routinely change, and such changes require that laboratory materials be modified accordingly.

e) Role of course coordinator: The role of the course coordinator is helpful primarily in larger institutions, where instructors share multiple sections of courses. Individual sections should be more or less synchronized. Course coordinator ensures that teaching assistants and other support personnel informed of the content and purpose of laboratory materials.

f) Feedback to course coordinator: Feedback from faculty and teaching assistants regarding problems in laboratory exercises or other materials is important.

g) Quality control assurance: It is necessary to provide a single point of maintenance for course materials such that updates are consistent with existing materials.

3. Case study

To test our approach, we applied the proposed assessment methodology to laboratories of Computer Engineering (CE) Department at Umm Al-Qura University. However, the proposed methodology is generic and can be applied to the laboratories of other disciplines also such as electrical engineering and computer science.

3.1 Computer Engineering laboratories

There are seven different laboratories: Computer Organization Laboratory (abbreviated as CO), Micro Computer System Design Laboratory (abbreviated as MC), Microprocessor Laboratory (abbreviated as MP), Advanced Logic Design Laboratory (abbreviated as AL), Computer Networks Laboratory (abbreviated as CN), Electronics Laboratory (abbreviated as EL) and Switching Theory Laboratory (abbreviated as ST).

3.2 Assessment of pedagogic aspects

The first three pedagogic aspects (relationship between theory lecture, content level and activity type) vary from one experiment to another while the fourth pedagogic aspect (learning environment) does not vary from one experiment to another. Consequently, we summarized the results for first three pedagogic aspects in a pedagogic table. We assessed the pedagogic aspects for all the seven laboratories. However, due to limited space, only the pedagogic table for computer organization laboratory is shown in Table 5.

The first two columns of Table 5 represent the serial number and title of each experiment. The third column shows the relationship between each experiment with the theory part of Computer Organization course. It is evident from Table 5 that first five experiments are loosely coupled with the theory while the last two experiments (6 to 7) do not depend on the theory lectures. The fourth column of Table 5 represents the content level of each experiment. Experiment number 1 is least sophisticated while experiments 2 to 5 are regarded as middle sophisticated. The last two experiments (6 to 7) are regarded as most sophisticated. The fifth column of Table 5 represents the activity type of each experiment. After getting a tutorial in the first experiment, the next four experiments consist of project construction. Finally, the last two experiments allow students some experimentation and analysis. The laboratory does not provide any exploration and interpretation activities to students.

3.3 Assessment of implementation aspects

The assessment results for implementation aspects are shown in Tables 6, 7, 8 and 9. These tables summarize the four implementation aspects (facilities, reliability, safety and laboratory manual) for seven different laboratories respectively. The first and second columns represent the serial number and description of attributes. Columns 2 to 9 represent the value of attributes for each laboratory. Finally, the last row in each table provides the total score of each laboratory.

3.3.1 Assessment of facilities

Table 6 shows that appropriate computing equipment is available in all the laboratories except electronics laboratory. Similarly, switching theory laboratory does not satisfy the working condition of equipments. It implies that some components are not working according to their technical specifications. Data sheets are present for each component in every laboratory. However, there is no internet or LAN access in any laboratory. Laboratories are lacking resources like printer, scanner and projector. Generally, the space is sufficient with the exception of microprocessor laboratory. Both the instructor and students are not satisfied with the laboratory timings.

3.3.2 Assessment of maintenance and reliability

Table 7 shows that there is no system failure policy exists to recover failure in laboratories. There is no database system for components tracking. Consequently, the replacement of faulty components is not an easy task.

3.3.3 Assessment of safety

Table 8 shows that every laboratory has a very low score in safety matrix. General laboratory safety rules are not displayed. Laboratory operating hours are not displayed for Micro Computer System Design laboratory (abbreviated as MC) and Switching Theory laboratory (abbreviated as ST). Similarly, none of the attribute in the categories of emergency safety and power supply related safety have scored the value of 1. Some of the attributes in equipment safety scored 1 such as temperature control and handling of liquids (if any). However, some attributes in the category of equipment safety are 0 such as installation of anti-virus software and awareness about waste management.

3.3.4 Assessment of laboratory manual

Similar to safety conditions, the quality of laboratory manual is at a low level as shown in Table 9. Almost every aspect of laboratory manual needs to be improved. One of the major concern is about the role of course co-ordinator. Enough feedback is not provided to him by laboratory instructors.

4. Conclusion

This article has proposed an effective methodology for the assessment of pedagogic and implementation aspects of laboratories. Pedagogic aspects covered are: relationship between theory and laboratory, content level, activity level and learning environment. Implementation aspects covered are: facilities, reliability, safety and development of laboratory manual. Proposed approach is extremely simple and

Table 5: Pedagogic table for computer organization laboratory at computer engineering department

S.No	Title of Experiment	Relationship	Content Level	Activity Type
1	Introduction to 8086 Assembly Language Programming	Loose Coupling	Least Sophisticated	Tutorial
2	Basic I/O Operations using Assembly Language	Loose Coupling	Middle Sophisticated	Project Construction
3	Memory Segmentation and Addressing Modes	Loose Coupling	Middle Sophisticated	Project Construction
4	ALU Instructions of 8086 Family of Processors	Loose Coupling	Middle Sophisticated	Project Construction
5	Use of Interrupt in Assembly Language	Loose Coupling	Middle Sophisticated	Project Construction
6	Mouse Applications in Assembly Language	Completely Independent	Most Sophisticated	Experimentation and Analysis
7	Semester Project to Evaluate the Progress of Students	Completely Independent	Most Sophisticated	Experimentation and Analysis

Table 6: Assessment results for Facilities Support Matrix (Abbreviations are used for each laboratory name)

S.No	Attributes	CO	MC	MP	AL	CN	EL	ST
1	Is appropriate computing equipment available ?	1	1	1	1	1	0	1
2	Is working condition of equipment satisfactory ?	1	1	1	1	1	1	0
3	Availability of data sheets ?	1	1	1	1	1	1	1
4	Is internet available ?	0	0	0	0	0	0	0
5	Are computers connected on LAN ?	0	0	0	0	0	0	0
6	Are printer, scanner and projector available ?	0	0	0	0	0	0	0
7	Is the space sufficient to house the facilities ?	1	0	1	1	1	1	1
	Total score of Facilities Matrix	**4**	**3**	**4**	**4**	**4**	**3**	**3**

Table 7: Assessment results for Reliability Matrix (Abbreviations are used for each laboratory name)

S.No	Attributes	CO	MC	MP	AL	CN	EL	ST
1	Is an administrator present to recover failures ?	0	0	0	0	0	0	0
2	Do system failure policies exist ?	0	0	0	0	0	0	0
3	Do equipments provide reliable results ?	1	1	1	1	1	1	0
4	Are faulty components replaced regularly ?	0	0	0	0	0	0	0
5	Does a system exist for component tracking ?	0	0	0	0	0	0	0
	Total score of Reliability Matrix	**1**	**1**	**1**	**1**	**1**	**1**	**1**

easy to implement. Computer engineering laboratories were assessed at Umm Al-Qura University.

The current version of our proposed assessment methodology provides a basic assessment skeleton which can be extended by introducing more implementation matrices such as laboratory inventory matrix. Similarly, matrices can be formed for the quantitative assessment of pedagogic aspects. Another probable direction is the formulation of a holistic methodology for the development of laboratory manuals. Pedagogic as well as implementation matrices can be used to assess the effectiveness of laboratory manuals.

References

[1] Lyle D. Feisel and Albert J. Rosa, “The role of the laboratory in undergraduate engineering education”, In *Journal of Engineering Education*, vol. 94, no. 1, pp. 121–130, January 2005.

[2] ABET Engineering Accreditation Commission, “Criterion for accrediting engineering programs”, 2011-2012. [Online]. Available: http://www.abet.org/eac-current-criteria/

[3] Jennifer M Case and Gregory Light, “Emerging Methodologies in Engineering Education Research”, In *Journal of Engineering Education*, vol. 100, no. 1, pp. 186–210, January 2011.

[4] Journal of Engineering Education, Website: http://www.jee.org/

[5] Barbara M. Olds, Barbara M. Moskal and Ronald L. Miller, “Assessment in Engineering Education: Evolution, Approaches and Future

Table 8: Assessment results for Safety matrix (Abbreviations are used for each laboratory name)

S.No	Attributes	CO	MC	MP	AL	CN	EL	ST
	General Laboratory Safety Rules							
1	Are safety rules displayed ?	0	0	0	0	0	0	0
2	Are labs properly locked ?	1	1	1	1	1	1	1
3	Are laboratory hours displayed ?	1	0	1	1	1	1	0
	Emergency Safety							
4	Is awareness present about safety?	0	0	0	0	0	0	0
5	Is awareness present about emergency ?	0	0	0	0	0	0	0
6	Is awareness present about evacuation ?	0	0	0	0	0	0	0
	Equipment Safety							
7	Are accessories properly operated ?	0	0	1	1	1	1	1
8	Is awareness present about the waste ?	0	0	0	0	0	0	0
9	Are liquids properly handled ?	1	1	1	1	1	1	1
10	Is equipment cleaned on regular basis ?	0	0	0	0	0	0	0
11	Is proper temperature maintained ?	1	1	1	1	1	1	1
12	Is anti-virus software installed?	0	0	0	0	0	0	0
	Power Supply Related Safety							
13	Is emergency power OFF present?	0	0	0	0	0	0	0
14	Are protectors used to avoid surges ?	0	0	0	0	0	0	0
15	Is equipment placed on static mates ?	0	0	0	0	0	0	0
	Total score of Safety Matrix	**4**	**3**	**5**	**5**	**5**	**5**	**4**

Table 9: Assessment results for Laboratory Manual Matrix (Abbreviations are used for each laboratory name)

S.No	Attributes	CO	MC	MP	AL	CN	EL	ST
1	Faculty is involved in the development of manuals ?	1	1	1	1	1	1	1
2	Laboratory material is considering the context ?	1	1	1	0	1	1	1
3	Laboratory material is upgraded on regular basis ?	1	0	1	1	1	1	0
4	Continuous improvement plan is being followed ?	0	0	0	0	0	0	0
5	Course co-coordinator is playing appropriate role ?	0	0	0	0	0	0	0
6	Is there any feedback to Course coordinator ?	0	0	0	0	0	0	0
7	Quality control is being assured ?	1	1	1	1	1	1	1
	Total score of Laboratory Manual Matrix	**4**	**3**	**4**	**3**	**4**	**4**	**3**

Collaborations", In *Journal of Engineering Education*, vol. 94, no. 1, pp. 13–25, January 2005.

[6] Lason L. Watai, Sarah A. Francis and Arthur J. Brodersen, "A Qualitative and Systematic Assessment Methodology for Course Outcomes from Formal Laboratory Work Products in Electrical Engineering", In *37th ASEE/IEEE Frontiers in Education Conference*, pp. 21–26, 2007.

[7] Ifeyinwa E. Djamel Azzi, Alan Hewitt and James Stocker, "A holistic approach to assessing students' laboratory performance using Bayesian networks", In *IEEE Workshop on Computational Intelligence in Virtual Environments, CIVE '09*, pp. 26–32, March 2009.

[8] Muhammad Rashid, Imran A. Tasadduq, Yousuf I. Zia, Muhammad Al-Turkistany and Saima Rashid, "Evaluation of Engineering Laboratories", In *IEEE International Conference on Education and E-Learning Innovations*, Tunisia, 2012.

[9] Michael Newby and Darrell Fisher, "An Instrument for Assessing The Learning Environment of A Computer Laboratory", In *Journal of Educational Computing Research*, vol. 16, no. 2, pp. 179– 190 1997.

[10] Sylvia Chong, Horn Mun Cheah and Ee Ling Low "Perceptions of student teachers in a blended learning environment", In *International Journal of Innovation and Learning*, vol. 8, no. 4, pp. 345–359, 2010.

[11] I. M. Dooba, A. G. Downe, and J. Jaafar, "Laboratory accidents in academia: An insident-report safety knowledge transfer model", In *National Postgraduate Conference*, pp. 1–6, September, 2011

[12] Anastasios Drosopoulos and Marios Hatziprokopiou, "Planning and Development of Lab Training Activities for Powerline Communications", In *IEEE Transactions on Education*, vol. 53, no. 3, pp. 384–389, 2010.

[13] Thaddeus Roppel, John Hung, Stuart Wentworth, and Scottedward Hodel, "An Interdisciplinary Laboratory Sequence in Electrical and Computer Engineering: Curriculum Design and Assessment Results", In *IEEE Transactions on Education*, vol. 43, no. 2, pp.143–152, May 2000.

A "Techy" Minor in Web Design and Development for Non-Technical Students

Ronald P. Vullo, Ph.D., and Catherine I. (Irving) Beaton, M.I.T.E.
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, New York 14623

Abstract - *It entrances students fluent in Facebook and Foursquare, but eludes those outside computing majors. Students recognize the benefits of computing skills and that web technologies have become a universal infrastructure. Web development skills are increasingly vital to non-computing disciplines so we have created a minor in web design and development for non-technical students. In it they develop significant technical depth, despite their non-computing backgrounds. This presents challenges, but makes for a vibrant classroom. The minor is taught using Active Learning so students learn and experience both the theory and practice of web development. These students generally have no programming experience and are gently introduced to the process in their first course and gradually exposed to more in-depth development throughout the rest of the minor. The course order is deliberately flexible allowing students to choose their own path to success. The minor's rapid growth reflects its relevance.*

Keywords: Curriculum, Education, Non-Technical, Programming, Web

1 Introduction

"The power of the Web is in its universality."
– Tim Berners-Lee, W3C Director and inventor of the World Wide Web[1]

Web development technologies are core to several computing degrees within the Golisano College of Computing and Information Sciences (GCCIS). Degree programs in Information Technology, Medical Informatics, New Media Interactive Development and Game Design and Development all have a significant web development component in their core courses. Students in degree programs often take a minor in web development. The authors support the belief that web development is integral to many computing disciplines, but also believe that it is increasingly vital to non-computing careers as well. For this reason, we have created a minor in web design and development that is both accessible to non-technical students, but also helps them develop significant technical depth of knowledge and skills. The minor attracts students from such diverse programs on campus as Marketing (E. Philip Saunders College of Business), Psychology (College of Liberal Arts), Media Arts and Technology (School of Print Media), and Professional Photographic Illustration (School of Photographic Arts and Sciences).

Bringing together students from such diverse fields of study presents some challenges, but also makes for a vibrant and exciting classroom environment. The entire minor is taught "hands-on" in an Active Learning environment where didactic material is mixed in with a considerable amount of lab work. Students learn and experience both the theory and practice of web development.

Although these are non-computing majors and generally have no programming experience, they are gently introduced to the process in their first course. Students are gradually exposed to more in-depth content in the primary course, and those threads are woven into each subsequent course. When they reach their last course, students are modifying open source code and writing their own simple programs. Over the duration of their classes, they learn concepts such as client-side vs. server-side processing, templates, code libraries, and AJAX. In addition to these technical topics they also learn about design, aesthetics, accessibility, content creation, usability, and the history and evolution of the technologies they are using.

The minor is designed for non-computing majors and students outside the computing field who wish to learn more than just the basics of Web usage. Students enrolled in this minor have little or no knowledge of web development and related technologies. The minor features courses covering image creation, video production, communication, development, and integration technologies applied to the Web. Students completing this minor have the broad range of skills and understanding necessary to design and build a basic, but complete, web presence.

This minor is not available to students whose home programs are in GCCIS. This is by design; there is another web development minor available for GCCIS students which is tailored to their needs and background. This minor sequence is designed to teach students the programming and other computing skills that reflect the depth appropriate for a minor in a degree program. These skills will not be a part of their major, but may support their chosen major as technology continues to grow as an important component in other degrees.

2 Program Structure and Sequence

The minor in Web Design and Development has a required first course (Web Foundations) three “middle courses” which can be taken in any order (Digital Image Creation, Digital Video Creation, and Rapid Online Presence) and a final course which serves to bring together and build upon the learning from the preceding four courses in an integrative manner (Web Integration & Application).

The program structure lays a common foundation for all of the students, allowing them to play equally on the same playing field. The various activities within the courses meet the needs of all four learning styles for students and provide components of stability and comfort for some students, and a sense of independence for others.

The sequence of the courses allows students to follow their own path, to some degree. After completing the foundation, students can elect from a selection of three courses. This option allows students to take control of their learning. They choose a course that matches their interests. Upon successful completion of the course, they now have increased motivation and desire, as well as confidence, to move on to the next course.

This flexibility of the minor is deliberate. David Kolb, a renowned researcher in the area of learning styles, suggests that there are four specific categories for learners:

- Accommodators – preference for well-controlled, hands-on method
- Divergers – risk-taking, experimenting, seat-of-pants learners
- Assimilators- want to be guided to ‘right’ answer
- Convergers – big picture to personalization[2]

While these categorizations are broad, they do represent different learner needs. For a traditional student (19-25 years old) the lure seems to be a flashy, fast, real/virtual experience. It has to ‘mean’ something to them. These learners are Generation X learners, and they lean more towards a structured learning environment with a ‘wow’ appeal. Therefore it becomes important to quickly support the creation of a product or process, so that they feel successful. This can present a challenge for instructors, especially in early courses where students lack sufficient skills to build something substantial.

Computing for non-majors can be an intimidating area of study for students. It is essential that they feel they can approach and master the content without feeling overwhelmed. The content of the courses is delivered in a variety of ways with a deliberate intent. For the first encounter with web/technologies, students are given an historical perspective on the internet and web page development. Some of this content is not new to all students. However, students do acknowledge that, while they are users of the internet and the web, they were unaware of much of its history – the how and why of the way it is now. The authors believe this historical context is important in establishing a solid foundation upon which to build knowledge.

Within the courses, there are a variety of activities to support different learning styles. Even in the first course, students hand-code HTML and CSS, create, edit, and manipulate digital images, create simple animations, and address web site accessibility issues There are opportunities for collaboration with colleagues through in-class exercises, and out-of-class assignments to encourage them to research their topics outside of class time.

While technology has become pervasive in society, industry has noted a need for other skills within the technology field. Writing skills, communication skills, and presentation skills are all sought after, in the computing field. It is for this reason that the courses also incorporates these components. The specific structure and content of each of the courses is provided below.

GCCIS-ISTE-105: Web Foundations

Students required to take this course: Students in the Minor in Web Design and Development for Non-Computing Majors, Students in the following degree programs: Advertising and Public Relations, Journalism, New Media Publishing, Professional and Technical Communications, and Psychology.

Students who might elect to take the course: Non-computing students wishing to learn web basics. (This course is closed to computing majors and will not be accepted for credit toward college of computing degrees.)

Goals of the course: This course provides an introduction and hands-on practice in developing basic web sites. Emphasis is placed on standards compliance and cross-platform development. It also provides an introduction to the fundamentals of web-delivered media.

Course description: A hands-on introduction to Internet and web foundations for non-computing majors. Includes HTML (Hypertext Markup Language) and CSS (Cascading Stylesheets), web page design fundamentals, basic digital image manipulation, and web site implementation and maintenance. Students will design and build their own web sites using the latest technologies and deploy them to the web for world-wide access.

Topics: 1. Web Pages
    1.1 Intro to Macintosh

1.2 History of Internet and Web
1.3 Basic Web Page Mechanics (HTML)
1.4 More HTML (Paths, Links & Images)
1.5 Advanced HTML and Validation
1.6 Adding Style with Basic CSS
1.7 CSS Layout
1.8 Copyright
1.9 Basic Navigation
1.10 Structure & Style (Menus & Lists)
1.11 Advanced CSS
1.12 Structure & Style (Swapping Stylesheets)
1.13 Color and Computers
1.14 Basic Design Principles for Web Sites
1.15 Fundamentals of Computer Graphics
2. Images
2.1 Pixelmap Images
2.2 Vector Graphics
2.3 Preparing Images for the Web
2.4 Editing Photographs)
2.5 Creating Web Backgrounds
2.6 Creating Images de Novo
3. Time-based Media (Animation and Video)
3.1 Animation
3.2 Video
4. Interaction (JavaScript)
4.1 JavaScript & DOM
5. Putting it all together
5.1 Designing a Larger Site
5.2 Site Design Critique
5.3 Prototyping a Site
5.4 Site Prototype Critique

Learning Outcomes: At the end of the course, the student will be able to:

- Create web pages using hand-coded HTML including CSS text formatting, images, tables, and CSS layout/positioning.
- Edit, create, and use digital images effectively by choosing and implementing correct image formats.
- Create and use small animated gifs to enhance web pages.
- Use SFTP software to upload and maintain web pages, including setting proper Unix permissions.
  in-class exercises and out-of-class projects

GCCIS-ISTE-205: Digital Image Creation

Students required to take this course: Students in the Minor in Web Design and Development.

Students who might elect to take the course: Students interested in learning digital image creation and manipulation for the web. This course is open to college of computing majors and may be taken as a free elective.

Goals of the course: Students will develop a deeper understanding of, and skill level working with digital imaging targeted for the web.

Course description: This course explores the creation and manipulation of digital images intended for use on the Web. Topics include basic digital photography, acquisition of images, and intermediate image manipulation. Students will do projects creating images suitable for the internet. Prerequisite: GCCIS-ISTE-105 or GCCIS-ISTE-140 (Web I for IT Majors)

Topics: 1. Digital Image Principles
1.1 Pixels
1.2 Additive and Subtractive Color
1.3 Bit Depth
1.4 Gamut and Color Spaces
2. Basic Digital Photography
2.1 Image Sensor technologies
2.2 Aperture
2.3 Exposure Time
2.4 Composition
2.5 Lighting
2.6 Planning a Shoot
2.7 Releases and Copyright
3. Scanning Images
4. Image Manipulation
4.1 RAW vs. JPEG
4.2 Color Correction and White Balance
4.3 Masks
4.4 Compositing Images
4.5 Removing Defects
5. Compression
6. Presenting images on the web

Learning Outcomes: At the end of the course, the student will be able to:

- Demonstrate knowledge of principles of Digital Image generation and display.
- Demonstrate knowledge of digital photography technologies and basic photographic optical principles.
- Take good quality digital photographs.
- Manipulate digital images.
- Appropriately present images on the web.

GCCIS-ISTE-206 Digital Video Creation

Students required to take this course: Students completing the Minor in Web Design and Development

Students who might elect to take the course: Students will develop a deeper understanding of, and skill level working with digital video targeted for the web. This course is open to college of computing majors and may be taken as a free elective.

Goals of the course: Students will acquire a basic competency in creating digital audio and video and its use on the Web.

Course description: This course explores the creation of digital video intended for use on the Web. Topics include basics of digital videography, acquisition of audio, editing, streaming, compression, as well as storytelling with video and integration into web sites and applications. Students will be provided with digital cameras for use in the course. Prerequisite: GCCIS-ISTE-105 or GCCIS-ISTE-140 (Web I for IT Majors)

Topics: 1. Storytelling with Video in a Digital World
- 1.1 Structure of a Story
- 1.2 Scripts & Storyboards
- 1.3 Fiction, News, Commentary, Comedy, etc.
- 1.4 Live vs. Edited Video

2. Technique
- 2.1 Lighting
- 2.2 Camerawork
- 2.3 Microphones and Audio Mixing
- 2.4 Animation, Stop-Action, Time Lapse

3. Digital Video Technologies and Techniques
- 3.1 Capturing
- 3.2 Editing
- 3.3 Captioning
- 3.4 Encoding
- 3.5 Streaming
- 3.6 Podcasting
- 3.7 Teleconferencing
- 3.8 Special Effects

4. User Interaction
- 4.1 Web Technologies for User Control
- 4.2 Integrating Audio and Video into Web Sites and Apps

Learning Outcomes: At the end of the course, the student will be able to:
- Create a Narrative
- Develop web-based audio and video productions
- Perform competent lighting technique
- Perform competent Videography technique
- Perform competent Editing technique
- Provide constructive peer critique and feedback
- Create an integrated web-video user experience

For this course students are provided with video cameras, and lighting equipment. This course is scheduled to meet once per week for a contiguous 3 hour block, usually in the evening and/or a lab which can be darkened for viewing video.

GCCIS-ISTE-305: Rapid Online Presence

Students required to take this course: Students in the Minor in Web Design and Development for Non-Computing Majors, Students in the New Media Publishing degree program.

Students who might elect to take the course: Non-computing students wishing to learn how to set up web sites that go beyond a few static pages. (This course is closed to computing majors and will not be accepted for credit toward college of computing degrees.)

Goals of the course: This course provides a practical overview of web development using rapid site development tools such as Blogs, Wikis, Content Management Systems, and Web Site Toolkits.

Course description: Although large-scale web sites still require considerable development effort, there are today several options for establishing a web presence using tools designed for non-programmers. This course gives students understanding of and experience with installing and customizing web sites using tools such as Blogs, Wikis, Content Management Systems, and Web Site Toolkits. Prerequisite:GCCIS-ISTE-105

Topics: 1. Blogs
- 1.1 What is a blog?
- 1.2 History of blogs
- 1.3 Blogging sites
- 1.4 Blog software installation and customization

2. Web Site Generators
- 2.1 Using software that creates web sites (i.e. iPhoto)
- 2.2 Customizing generated pages

3. Wikis
- 3.1 What is a wiki?
- 3.2 History of Wikis
- 3.3 Wikis vs. Blogs
- 3.4 Wiki software installation and customization

4. Web Site Toolkits
- 4.1 Types of Toolkits
- 4.2 Installing and using toolkits

5. Content Management Systems
- 5.1 What is a content management system?
- 5.2 Content management systems vs. toolkits, wikis and blogs

6. DIY: Integrating open source pieces
- 6.1 Mixing and matching toolkits, content management systems, wikis, blogs generated pages

Learning Outcomes: At the end of the course, the student will be able to:
- Evaluate and select appropriate tools and methods of rapidly creating web sites.
- Apply the basic principles of information architecture and navigation design to the development of a web site.
- Install and configure a selection of open source web applications such as wikis, blogs, etc.
- Modify the design/layout/navigation of the above systems to create customized web presence.
- Add and manipulate content in the above systems.

GCCIS-IST-405: Web Integration & Application

Students required to take this course: Students completing the Minor in Web Design and Development. (This course is closed to students not enrolled in the minor.)

Goals of the course: This course brings together and builds upon the learning from the preceding four courses in the Minor in Web Design and Development in an integrative manner.

Course description: The final course in the minor in Web Design and Development (for non-GCCIS majors). Students will develop a deeper understanding of technologies underlying the web and how to combine them. This course builds upon the work from the preceding four courses in the minor and emphasizes integrating multiple technologies and content sources to create sophisticated web sites and web applications for desktop and mobile devices. This course is not available to GCCIS majors. Prerequisites: GCCIS-ISTE-105, GCCIS-ISTE-205, GCCIS-ISTE-206, and GCCIS-ISTE-305

Topics: 1. Server-Side Programming
- 1.1 constructs
- 1.2 functions
- 1.3 data structures
- 1.4 includes and code organization

2. Maintaining State
- 2.1 cookies
- 2.2 session variables

3. Structured Content
- 3.1 normalization
- 3.2 structured files (CSV, INI, XML)
- 3.3 databases

4. Site Architectures and Approaches
- 4.1 Pages and Directories
- 4.2 Model-View-Controller
- 4.3 two- and three-tiered architectures
- 4.4 Integrating Web Services & Data Sources
- 4.5 Security Issues

5. Web Applications
- 5.1 AJAX
- 5.2 Mobile Apps

Learning Outcomes: At the end of the course, the student will be able to:
- Evaluate and select appropriate methods and technologies for creating multi-tiered web sites.
- Integrate multiple data sources into a web site or web application.
- Maintain user state and provide an individualized user experience.
- Take measures to address security issues of a data-backed web system.

For this course students are provided with accounts on a Unix web server supporting PHP and MySQL. Mobile computing devices (i.e. iPads and iPod touches) are also provided for development and testing of web apps.

3 References

[1] Berners-Lee, Tim. Retrieved April 10, 2012 from http://www.w3.org/WAI/.

[2] Smith, M. K. (2001). 'David A. Kolb on experiential learning', the encyclopedia of informal education. Retrieved April 10, 2012 from http://www.infed.org/biblio/b-explrn.htm.

Teaching Mathematics and Programming Foundations Early in Curriculum Using Real-Life Multicultural Examples

M. Barjaktarovic
Department of Mathematics and Computer Science, Hawai'i Pacific University, Honolulu, Hawai'i, USA

Abstract - *This paper is about teaching mathematics and programming early in and throughout the college curriculum to provide students a solid foundation that is enforced in many classes and via multiple means. Introducing concepts early and having constant repetition enforces the retention and mastery of the material. The solutions we propose in this paper are to be introduced in basic digital literacy classes for all majors, all the way up to classes for majors in STEM fields. Since brain research shows that we learn best based on what we already know, we have tied our methods into daily situations that students from many diverse backgrounds can relate to. We have tried these approaches in basic mathematics and programming classes and they seem to work.*

Keywords: formulas, programming, mathematics, curriculum, teaching, diversity

1 Introduction

A common question facing many instructors these days is the question of how to get all students up to par with their mathematical understanding. This question is related to the preparation as well as the enthusiasm of students. Many students these days arrive to college unprepared to tackle basic mathematical tasks and thus require remedial work. In this paper we will discuss some tested remedies for the problem to be introduced in classes that are not necessarily seen as mathematical, as well as more mathematical classes for majors only.

Based on seeing the work of many students as well as discussing the subject with other instructors, we came to a conclusion that the best place to start is learning how to use formulas. Any instructor who has taught anything related to mathematics or coding can vouch that many students have difficulty with substitution into formulas. Mastering formulas is critical for any mathematical work. For example, students can learn logic very quickly if they understand how to work with formulas, e.g. those learned in basic algebra. Formulas are abstract and thus can be hard to grasp by students. In addition, mathematics can be perceived as a difficult, boring, and abstract subject without much relevance to daily life.

We believe that it is necessary to start with the basic concepts and build understanding using examples that students can relate to and thus feel inspired to learn about. Neuroscience research [6] proves that we learn most efficiently if we can relate new information to what we already know. Mathematics is a completely new language and thus at the very start it does start with sheer memorization. This property can make mathematics seem elusive to those who did not learn the basics. However, once the basic language is mastered, we can teach mathematical concepts using what we already know.

2 Substitution and Programming Early

The basic literacy classes taught to all freshmen and required in most colleges and universities are a prime opportunity to introduce basic mathematical skills and even teach beginning programming. A spreadsheet cell with a formula can be seen as a "one line program." A spreadsheet can be read as a program.

The following approach was used to strengthen math skills and introduce students to programming in Basic Digital Literacy classes, required of all freshmen. The goal of the class is not to train programmers or mathematicians, but to train students to use spreadsheets. However, we use this opportunity to teach mathematics and programming as well.

Practically 100% of students reported that they understood what was being taught and they could code basic spreadsheet functions afterwards, as shown in their high hands-on exam achievements. The next step in research is to see if students of all majors can perform simple coding in Python, Ruby, or another easy-to-use imperative language. Our inclination is to believe that the students would have much easier time programming. Our experience is that computer science students can much more easily program in Java.

In the following example, we will teach several concepts:

- programming is, in essence, simply typing up what we already write on paper by hand in mathematics.
- we can design our own formulas and check if they are correct
- units are very important and formulas have to be adjusted to accommodate units

- a formula has to be designed so that each part is very clear
- variable names should be descriptive
- variables can be a conglomerate expressions involving other variables.

We will demonstrate these concepts using the topic that most students find interesting. Finding a topic that is interesting to all students, taking into account their diversity in gender, age, and cultural background, is rather challenging. We have experimented with many different topics, such as: cooking dinner, graduating, dress code, and many others. The best success with American students, regardless of their background, was achieved when discussing cars. Examples that involve clothing, sports, etc. always leave out female students (or conversely, make male students acutely aware that they are a majority when an example includes high heel shoes as well as sneakers.) Cars seem to be a topic that everyone can relate to with relatively high interest. Therefore, we will use an example of a car dealership.

2.1 Car Sales Example

Let us assume that a car dealership pays their salespeople a certain commission for every car sale made. How can a salesperson calculate how much money they have earned? First, we need to design the formula; then we need to code it so that the computer can do all the calculations.

2.1.1 Formula design (including units)

Let us use a very simple formula involving commission. We inform the students that our formula is rather simple and thus unrealistic, but is a good start towards more complicated real-life formulas. We will let the business majors provide the accurate formula used in real life.

Let us assume that the formula for earnings by commission is:

$$y = a\,x + b \tag{1}$$

where variables y, a, x and z are defined as:

y: the total earning
a: the price per car
x: the number of cars sold
b: the total bonus.

In order to match the units (for example, dollars), we need to multiply the bonus per car with the number of cars. Therefore:

$$b = a\,z \tag{2}$$

where variable z is defined as:

z: bonus per car.

The first step is to ensure that the formula we designed is correct. The formula does seem to make sense in terms of car sales. However, did we use correct operations? A good example for checking units is speed in miles per hour: if the units for speed are miles per hour, i.e. speed is measured in miles/hour, then the formula speed=time/distance is correct since all units match. The formula distance=time/speed is also correct. The important learning here is that units themselves are mathematically processed, just as if they were variables.

For the first quick check, students are asked to look at the units. Do units match? Yes, since we adjusted for bonus per car. Therefore, the formula was constructed correctly.

2.1.2 Writing formula on paper by hand

Having confirmed that the formula is correct and makes sense both in terms of the situation modeled as well as technically in terms of units and operations, the next step is to code the formula.

We all agree that this is how formula (1) looks when we write it in a math class "on paper" by hand:

$$y = ax + xz \tag{3}$$

Then we can plug in various values, such as:
a=\$20K/car, x = 5, z=\$500/car.

At this point, we are ready to code the formula.

2.1.3 Coding formula in a programming language

Coding can be difficult for students because they expect that a computer works like a human being and can clearly see and self-correct simple mistakes such as typos, misspellings, and extra or omitted characters. Often, beginners are likely to produce exclamations of statements such as "but I just forgot to put a comma!" and even statements such as "the computer doesn't like me!"

At this point we have not introduced the word "syntax" or "compiler" officially. This is not a programming class but a basic digital literacy class and there is no need to overwhelm the students with extra vocabulary. We did introduce earlier the words "program," "application," and "running the program or application."

We tell students that a computer program is like a robot – it is very literal and can understand only what it is programmed to understand. This imagery helps students to appreciate the rules of the syntax. Also, it makes them less frustrated and less likely to take it personally. With this approach, students actually enjoy programming and focus on it quite deeply. We believe that it resembles a video game in some way, where the player communicates with a robot. That makes it somewhat enticing.

Keeping in mind that we are indeed "communicating with a robot," it is clear that, if we code formula (3), we must communicate very clearly which mathematical operators we want to use. Also, computer programs cannot store units as easily as we are used to simply writing them out on paper. Also, we still cannot write units such as “20K” because computer does not recognize K as a part of a number.

In most programming languages, the multiplication is represented with *, and units are implicit. We do tell students that we are coding the formula in a programming language such as C/C++, Java, Python, or any other imperative-style language. Although the students are not in a programming class, they appreciate the learning.

Therefore, the basic code for formula (3) would look like the following:

```
a=20000
x=5
z=500
y=a*x + x*z
```

All students understand this, since this is extremely close to what they have seen in mathematics on paper.

2.1.3.1 Code structure and coding style (algorithm)

To introduce more “real” programming, we talk about giving variables more descriptive names, such as:

ppc: **p**rice **p**er **c**ar, in dollars per car
nc: **n**umber of **c**ars (no units)
bpc:**b**onus **p**er **c**ar, in dollars per car
e: **e**arnings.

Therefore, the code in most imperative languages such as Python would look like the following:

```
ppc=20000
nc=5
bpc=500
e = ppc * nc +  nc * bpc
```

We draw the student’s attention to the fact that code becomes readable “like a story” because we can read the variable names easily. We also draw their attention to the fact that this “story” can be written in a way that makes it easier or harder to read and understand. Sometimes, it is impossible to understand. For example, e can be calculated only after ppc, nc and bpc have been defined.

Thus, we implicitly introduce the concept of algorithms.

2.1.4 Coding formula using a spreadsheet

So far, we have worked with imperative style languages because their syntax resembles mathematics on paper by hand, and thus students can very easily relate to it. However, since they are not in a programming class but in a basic digital literacy class, we have to cover spreadsheets. During class, we use Excel as the platform since it is a common business application and is installed on the classroom computers. However, many students have Mac computers with their own version of spreadsheet, and many students use open source software, such as Libre Office. We tell students that the particular brand of spreadsheet is not important; the principles are the same.

What would code look like if coded in a spreadsheet? The most important concept and the most obvious difference between spreadsheets and traditional “on paper by hand” mathematics is how variables are represented. This feature is very ingenious and makes spreadsheets so useful and practical. In a spreadsheet, variable names are cell names.

The next difference (that is quite easy to understand because it does resemble “on paper by hand” mathematics albeit only by the looks) is the fact that we can write variable names in spreadsheet cells; only cells that have content starting with “=” are used for calculation. In this way, we avoid any discussion of commenting code written in, say, Python or C; and we implicitly introduce the idea that code can be commented for the ease of human users.

The last but not least difference is that spreadsheets can include units with the values. For example, numbers that represent dollar values can show $ next to them. This is yet another feature of spreadsheets that makes them very practical.

The next step is to introduce spreadsheet syntax. Our code in a spreadsheet would look like the following:

	A	B
1	**p**rice **p**er **c**ar, in dollars per car	20000
2	**n**umber of **c**ars	5
3	**b**onus **p**er **c**ar	500
4	**e**arnings	=B1 * B2 + B2 * B3

Figure 1 - Spreadsheet example

A, B, 1, 2, 3 and 4 is what the spreadsheet itself shows, and everything else we typed in. (Earlier, we have already introduced the concept of spreadsheet cell. Therefore, students know what the content of each cell is, e.g. that the content of cell A1 is “price per car, in dollars per car.”)

We invite students to experiment with their spreadsheet organization and include units. We ask them how they want their spreadsheet organized. What arrangement is the easiest to read and understand? Below are some ideas:

- columns arranged as variable names to the left and units to the right of column B
- columns arranged as variable name, units, value
- columns arranged like we are used to in sciences, with the header row that contains variable definitions and units
- column B marked with the proper units already, since spreadsheets can do that.

Thus we implicitly introduce the concept of technical communication as well as scientific notation.

2.1.5 Testing

When the spreadsheet is filled in with actual numbers and the values have been calculated, we ask the students to look at the results and see if they make sense.

For example, spreadsheets allow for different formatting of cells, e.g. some cells can be formatted as percentages or dollar amounts. For example, some students will format the cell containing the number of cars as dollar values. Some cells will be formatted as percentages instead of general numbers and thus will show the value as 100 times larger than the correct value. Students themselves pick on such mistakes and correct them.

2.1.6 Conclusion

Going from on-paper mathematics to imperative style languages is quite natural and easy to understand. Introducing coding using spreadsheet coding style is slightly more complicated because the syntax looks very different than what students are used to. All students can relate the patterns and see how the "on paper" mathematics matches the code, and how it all gets translated into spreadsheet-specific "code." Practically all students become capable of using formulas.

We make the concepts very clear and systematic and draw the attention to the fact that solving the problem starts with defining what the problem is, in English; designing a correct and appropriate formula; and finally calculating, in our case using a computer. The last step is to evaluate and test the results.

We emphasize that the results are not necessarily correct, even if calculated with a computer. Wrong code will produce wrong results. Students learn to appreciate that fact when they see simple formatting errors.

2.2 Using Real-life Formulas

In the same basic digital literacy class for all majors, we continue programming basic formulas into a spreadsheet. Below are some basic formulas conducive to teaching substitution and programming. Assume that we have variables X, y and Z:

In English, we say:	In math, we write it as:
Z is y% of X	Z = X*y /100
Z is X increased by y%	Z =X+ (X*y / 100)

Because * has precedence over +, () are not needed, so we can write the formula as following:

Z = X+ X*y / 100

We can also simplify the formula and write it as following:

Z = X*(1+y/100)

Z is X decreased by y%	Z = X – X*y / 100 Z = X*(1 - y/100)
X is what percentage of Z?	Call the percentage y y = X/Z *100

In order to make formulas more realistic yet easily understandable, we use examples of sales, taxes and tips. We also show students real-life tax forms with the similar calculations we do in class (and we self advertise and mention high earnings for such work.) For example:

- If a store makes $1000, and they must pay 5% tax (which is the real-life general excise tax in Hawai'i), how much is the tax?
- If a restaurant bill is $100, and the customary tip is 10% of the bill, how much is the tip?

Students often want examples to be related to "their" major. That is impossible in a basic digital literacy freshman class consisting of all different majors. Therefore, we tell students that the examples are applicable to any field, if we just change the text of the problem to fit the application they are interested in. For example:

- Hospital/hotel/business/ costs are $1000, out of which 14% goes on salaries. How much is paid for salaries?
- Ocean temperature rises 1 degree per year. If the current ocean temperature is 50 degrees, what is the temperature percentage increase?
- If dosage for an adult patient is 3 mg of the medication, and children get only 10% of adult dosage, how many mg of medication will a child get?

Below is an example of practicing coding in a spreadsheet. Students are given values in column B and asked to calculate values in column C, i.e. they are asked to map what they understand from "on paper" mathematics into spreadsheet formulas and code.

	A	B	C
1	Store	Profit	% of the total profit
2	Store1	1000	=B2/$B$5
3	Store2	2000	=B3/$B$5
4	Store3	3000	=B4/$B$5
5	Total:	=SUM(B2:B4)	=B5/$B$5

Figure 2 - Spreadsheet example (view formula)

Use the last formula: X is what percentage of Z? y=X/Z *100.

This spreadsheet also includes specific spreadsheet syntax, such as how to write a formula for a summation using a range of cells from cell B2 to cell B4 (i.e. cells arranged vertically). Also, students are asked to use $ to fixate the formula on a specific cell, since spreadsheets automatically adjust cell names when a formula is repeated by dropping the formula into neighboring cells using the mouse.

We ask the students to discuss when and why it is not necessary to include *100 in the formula. The answer depends on if the cell is formatted to represent a percentage or not. In other words, we review the concept of units, which are already programmed into the spreadsheet.

2.3 Using Multicultural Examples

Ethnomathematics can be briefly described as the study of mathematics practiced in different cultures [3]. Modern Western mathematics is based on Roman, Green and European influences. However, mathematics has been practiced in the rest of the world since time immemorial and offers many rich contributions. African, Chinese, South American, Indian, Turkish, Celtic, Arabic, and many other cultures have had advanced mathematics for centuries. For example, African music uses advanced polyrhythmic structures; some African villages and dwelling units are constructed as fractals [4].

A visual tour of many examples can be found on the Culturally Sensitive Tools website [4]. For example, Rhythm Wheels are an excellent tool for teaching division, GCD, and many other concepts related to patterns obtained by multiplicative increases using music. Designing exercises based on these tools is a whole topic in itself. It has been proven to work extremely well in local schools as well as internationally.

2.4 Using Visual Formulas

We found quilts to be easily understandable by students. Other choices would be ceramic tile, or perhaps even Celtic patters used for tattoos.

A quilt pattern is a formula, just given visually. Some quilt patterns, i.e. formulas, are given in [5]. Let us look at the breadfruit leaf pattern, as shown in Figure 3. Breadfruit is a staple food for Polynesians and grows in many local gardens. It is present very much in local artwork. It is actually possible and quite common to have quilts with a breadfruit leaf pattern, and they are also coveted souvenirs.

As is apparent from Figure 3, the pattern starts as a half-leaf and proceeds to one leaf and four leaves. Visually, it is obvious that the pattern is repeated in a particular way to obtain this effect. Thus we can formally introduce the concept of repeating formulas.

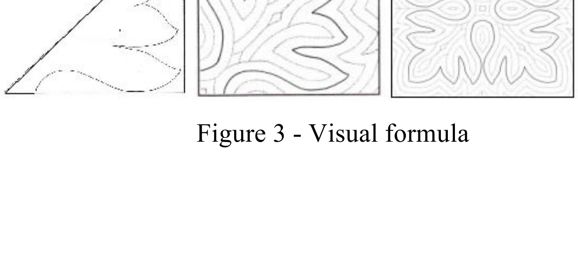


Figure 3 - Visual formula

We draw students' attention to the fact that each of the above images is a formula: half-leaf formula is used to construct one-leaf formula, which is then used to construct four-leaf formula. (Thus, we implicitly introduce the concept of recursive formulas.)

Let us discuss the one-leaf formula, just to keep it simple. Our conversation applies to the other two formulas. The formula does not say what color to use, it just tells the relationship of the different variables. In this case, variables are areas. One area is the leaf, another area is the background. (Regarding the one-leaf and four-leaf patterns, students sometimes ask: why not delineate areas bordered with the stitching lines and assign them different colors? We answer that indeed it is possible, but we are not working with a more complex example yet. For the sake of simplicity, let us work only with the traditional pattern, which has only two areas: the leaf and the background.)

We can call the breadfruit leaf as variable "leaf" and the background as variable "background". The attributes are: the type of fabric, and the color.

This way of thinking is intentionally introducing the concepts of variable attributes, which will become fields in a later programming class. Our intent is to introduce the concepts as early as possible and keep on repeating them.

2.5 Substituting one Formula into Another

A variable can be quite complex and consist of many different expressions and variables. For example,

$x = 2$

can be the result of calculating a more complex expression, such as:

$x = a * (y+2) - b^c * (z+5)^{25} / \sin \omega t$

where y, z, ω and t are all variables and a,b and c are constants.

For example:

$\sin(\log(0.001*34.555)^2 * 30000000 * \mathrm{sqrt}(44-34/34))^2$ is really $\sin(x)^2$, where

$x = \log(0.001*34.555)^2 * 30000000 * \mathrm{sqrt}(44-34/34)$.

Furthermore,

$x = a * b$, where

$a = \log(0.001*34.555)^2$

$b = 30000000 * \mathrm{sqrt}(44-34/34)$

Therefore, we can look at the same expression through the eyes of different patterns. For example:

$(x + y)^2$ can be looked at through the eyes of the pattern:

$$x^2 = x * x \quad (4)$$

in which case

$(x + y)^2 = (x + y) * (x + y)$.

The above looks like the pattern of distributive law, i.e. it fits the formula

$$(a+b)*(c+d) = ac + ad + bc + bd \quad (5)$$

Applying it to the above case, we get:

$(x + y)^2 = (x + y) * (x + y)$
$= x*x + x*y + y*x + y*y$
$= x^2+2xy + y^2$

Alternatively, $(x + y)^2$ can be looked at through the eyes of the pattern:

$$(a+b)^2 = a^2+2*a*b+ b^2, \quad (6)$$

in which case
$(x + y)^2 = x^2+2*x*y + y^2$.

Both approaches are correct. The first approach is more laborious, and second approach requires using a more advanced formula. Which approach to use? We emphasize to the students that all approaches lead to the same result, but some approaches are faster than others. It is practical to know which approach to take. We get that "feel" as to which approach to take only by practice. Just like a good cook knows how much salt to put in without measuring, we eventually get a feel for which formula to apply.

Once the basic concepts are mastered, students are asked to apply their knowledge to more complex expressions, such as:

Expand $(x + y - z)^2$ by splitting up the problem in several different ways.

Simplify $\cos^2(2*\omega*t) + 2*\sin(55*x)*\cos(2*\omega*t) + \sin^2(55*x)$.

2.5.1 Proper stacking of formulas

We tell students that formulas are like Lego pieces and require stacking in a proper order. In other words, whatever expression we use as a part of another expression has to match.

For example, factorial function does not allow non-integer input. We ask the students to try to expand expressions such as $(\sin x^2)!$ and $((\sin x)^2)!$.

Therefore, we implicitly introduce the notion of data type, one of the basic concepts used in programming.

2.5.2 Examples

Students can proceed to more advanced topics, such as logic, only when they thoroughly understand substitution. In our experience, students can apply substitution principles even if they do not fully understand the formula they are applying it to. For example, in a basic discrete math class students can simplify the following formulas before they even know what logical NOT (i.e. $\neg$) operator is. Below is an example of a problem that students easily solve.

Example: The formula called "double negation" is valid for all logical expressions p. The formula is: $\neg\neg p = p$

Do not get concerned about what logical expression is or what $\neg$ is. It is the same concept as we saw before. For example, just like saying: sin sin x. What is the implied order of parenthesis?

Apply the above formula to simplify the following expression, where p and q are some logical expressions and "and" is an operator, for example like + or - would be in algebra:

$\neg\neg(q)$
$\neg\neg(p \text{ and } q)$
$\neg\neg(p \text{ and } q \text{ or } p)$
$p \text{ and } \neg\neg(q)$

2.6 Formulas with Multiple Possible Results

Formulas that have "forking" qualities, i.e. different results based on certain conditions, are quite challenging to many computer science students, even good students. It somehow confuses them that the result is not the same every time we use the formula, perhaps because they think of mathematics as something rigid. However, we reassure them that they have seen such formulas probably practically every day of their lives. For example:

if it is rush hour, I take the highway
otherwise, I take Kapiolani Boulevard.

One such formula that tests substitution skills is the absolute value formula.

2.6.1 Example: absolute value formula

Definition of absolute value is as follows:

$$|x| = \begin{cases} x & \text{if } x \geq 0 \quad (7.1) \\ -x & \text{if } x < 0 \quad (7.2) \end{cases}$$

This is the definition written in mathematics "on paper by hand." We read lines (7.1) and (7.2) as:

$|x| = x$, if $x \geq 0$
$|x| = -x$, if $x < 0$

Students find it rather challenging to accept that absolute value can have two different results, based on the value of x. We believe that the mathematical way of writing the formula is a part of the confusion, because the formula is written contrary how we speak in daily life.

Students find it easy to accept the formula written in the programming style. Lines (7.1) and (7.2) written as code in an imperative language, such as Python, will look similar to the following:

if x≥0 then | x | = x
if x <0 then | x | = -x

We tell students that this is not code yet, since we cannot type in the above statements into computer. Code cannot use curly brackets because code is just a plain text file. Also, $|x|$ is not possible as the name for the result. Therefore, in coding, we make up a name for $|x|$, for example absx. That is why we document computer programs, because we need to say that absx means $|x|$.

Therefore, the code would look like outlined below:

```
if x≥0  then absx = x
if x <0 then absx = -x
```

The third way to write the absolute value formula is used by engineers, scientist, and logicians. This approach is routinely used by computer engineers and logicians when working with Boolean expressions. We can write the absolute value formula as a table:

x	$\|x\|$
≥0	x
<0	-x

Figure 4 - Table-based approach to formulas

The table-based way is very helpful for troubleshooting, especially if there are many absolute values requiring keeping track of different variables.

2.6.2 Combining forking and complex expressions

When the basic absolute formula is mastered, students are asked to test their substitution skills on more complex examples. For example, students are asked to expand the following expressions:

$|x + y + z|$

$|x| + |y|$

$|x + y - z| + |x|$

$|x^2 + y - z|$

$|(x + y)^2 - z|$

$|x|^2$

$|x^2|$.

2.7 Hands-on approach to teach substitution

We have devised a rather simple but fun and very practical way for teaching substitution. We use "games" that are used in early childhood education (Figure 5). We have not seen this approach anywhere else, but in our experience it works very well for teaching mathematics to anyone. The goal is to help students develop an eye for seeing patterns.

Students are asked to treat each place in the formula as a placeholder and enter the appropriate expression into it. For example, let's use the circles. Every circle has to contain the same expression. Therefore, we can look at the absolute value formula as shown below:

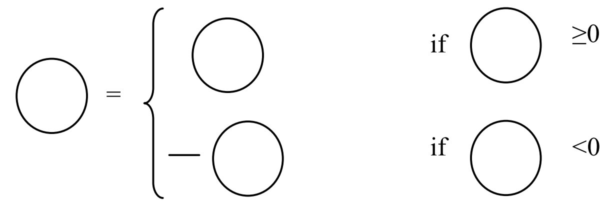


Figure 5 - Hands-on game-like approach to formulas

2.8 Conclusions

Approaches we tried worked on a very diverse group of students, ranging from freshman of all majors to computer science majors; diverse backgrounds based on gender, age, and cultural background, as well as different academic abilities and level of preparation. Student exams and quizzes show definite improvement.

3 References

[1] Milica Barjaktarovic. "Teaching Design and Testing in Computer Science Curriculum." The 2012 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'12), July 2012.

[2] Milica Barjaktarovic. "Training Effective Developers." The 2012 International Conference on Software Engineering Practice (SERP'12), July 2012.

[3] "Ethomatematics Digital Library." National Science, Technology, Engineering and Mathematics Education Digital Library (NSDL) funded by NSF, http://www.ethnomath.org/, 2005.

[4] R. Englash. "Culturally Situated Design Tools (CSDT)." http://csdt.rpi.edu/, 2003.

[5] Elizabeth Root Designs Hawaii. "Quilts Hawaii". http://www.quiltshawaii.com/freepattern.html, 2010.

[6] Colleen Politano, J. P. "Brain-Based Learning with Class." Portage & Main Press, 2000.

Development of Bioinformatics Foundational Courses in Undergraduate Curricula

Muhammad Ali
Department of Computer Science, Tuskegee University, Tuskegee, Alabama, USA

Abstract - *This paper describes the development of bioinformatics foundational courses for incorporation into undergraduate biology curricula. A sequence of three courses was developed with multi-disciplinary collaboration between the Departments of Biology and Computer Science at Tuskegee University. The focus was on teaching the effective use of bioinformatics tools, as compared to development of bioinformatics tools. The first two courses, Introduction to Bioscience Computing and Biological Algorithms & Data Structures, cover the basic computing and computer science foundation necessary for the informed use of bioinformatics tools. The third is an introductory course in bioinformatics. The courses were designed such that they can be taken by students majoring in biology or computer science. This work was supported by a NSF HBCU-UP grant.*

Keywords : Bioinformatics, Biocomputing, Biological algorithms, multi-disciplinary curriculum development

1 Introduction

Bioinformatics is a multi-disciplinary field that focuses on the discovery of biological knowledge using computational techniques. The need for core bioinformatics knowledge for biologists has often been emphasized [1] [2]. This paper discusses the efforts made by the Computer Science and Biology departments in Tuskegee University, Alabama, to develop introductory bioinformatics foundational courses for the biology curriculum, with an ultimate goal of introducing a computational genomics track in the Department of Biology. The paper first discusses the overall considerations for the contents of the sequence. This is followed by presentation of rationale for the selection of computing environment and contents for each course. A few earlier studies exist focusing on this issue [3] [4] [5]. This paper focuses on the development of undergraduate courses that equip undergraduates with the tools they would use in the industry, while laying a solid foundation for in-depth graduate studies. The paper also focuses on the successful integration of these courses into existing curricula that are constrained by credit hour requirements.

2 Considerations for the contents of the courses

It is possible to include in-depth treatment of computer science, mathematics, statistics and biology topics in a graduate-level bioinformatics curriculum. Consequently the initial bioinformatics curricula were at the graduate level. With the transition from the “Modern Synthesis” phase to the “New Biology” phase in the last third of the twentieth century, molecular biology has become increasingly important [6]. The amount of sequence data collected for various organisms is such that it is not possible to study them and compare them manually. The amount of such data is increasing exponentially. There is no recourse but to use computational means to analyze the biological data and solve biological problems. It has therefore become evident that some core bioinformatics knowledge needs to be imparted to undergraduate biology students. Present-day undergraduate biology curricula have very limited mathematical, statistical and computer science contents that are foundational knowledge for bioinformatics. This has called for a major shift in undergraduate biology curriculum for the 21st century.

The goal of the effort reported in this paper was to incorporate the essential bioinformatics related knowledge into an undergraduate biology curriculum in Tuskegee University, a small minority institution with fewer resources as compared to most other universities.

The in-depth treatment of bioinformatics that is possible in a graduate level bioinformatics curriculum is neither possible nor relevant to an undergraduate biology curriculum. This is because there is already shortage of available credit hours, and many of the graduates may not later follow a graduate bioinformatics curriculum. The focus needs to be on the basic uses of bioinformatics tools, rather than the development of these tools.

A survey of graduate level bioinformatics curricula shows considerable variation in content. There are three ends to this spectrum as shown in Figure 1. The central area ABC represents bioinformatics relevant knowledge. Curricula positioned towards the end-point C have high mathematics and statistics content [7]. Similarly, curricula positioned towards the end-point B have high computer science content [8], while those positioned towards the end-point A have high biology content [9]. Such variations in content take place mainly based on which department is offering a curriculum. Courses developed for a bioinformatics curriculum are influenced by the positioning of the curriculum with reference to Figure 1. For an undergraduate biology curriculum, it is more appropriate to position it towards end-point A.

An undergraduate biology curriculum should provide sufficient foundation in mathematics, statistics and computer science knowledge, while keeping the primary focus on biological applications.

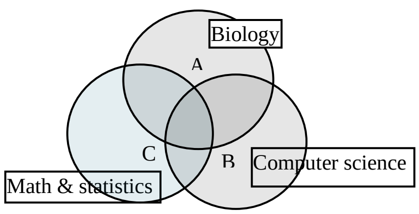


Figure 1 Constituents of bioinformatics discipline

Bioinformatics curricula also vary depending on whether the curriculum is targeted at bioinformatics tools developers, or the users of the bioinformatics tools. Almost all graduate level bioinformatics curricula have a high programming and algorithmic content, and thus are targeted at tool development in addition to developing proficiency in the use of bioinformatics tools. On the other hand most students graduating from a biology undergraduate program are not likely to become bioinformatics tool developers but are likely to be called upon to use bioinformatics tools across many biology/medical related jobs. Careers such as Clinical Geneticist, Medical Geneticist, Genetic Counselor, Genetics Laboratory Research Assistant and Genetics Laboratory Technician would require knowledge and use of bioinformatics tools [10]. In the case of undergraduate biology students, the bioinformatics courses should be targeted at the use of bioinformatics tools rather than tool development. Consequently in-depth treatment of mathematics, statistics and computer science contents was avoided. The courses contain enough computer science, mathematics and statistics content that are sufficient for the students to use the bioinformatics tools effectively. For example, a mostly qualitative treatment is presented pertaining to the development of substitution scoring matrices, rather than a rigorous mathematical treatment.

3 Development of the course sequence

The inclusion of multi-disciplinary knowledge in undergraduate biology curricula becomes difficult because of the competing need to include several new biology topics. The number of bioinformatics foundational courses needs to be kept to a minimum. A good understanding of basic probability and statistics is essential to the understanding and effective use of bioinformatics tools. An inter-disciplinary course BIOL 0202 Mathematics, Computers and Biosciences was developed with collaboration between the Mathematics and Biology department, to address this deficiency. A sequence of three bioinformatics foundational courses was developed with collaboration between Biology and Computer Science departments, to be offered after BIOL 0202. This sequence is designed to communicate enough bioinformatics knowledge to undergraduate biology students to teach them the effective use of bioinformatics tools and to be able to perform rudimentary programming that can help them to handle tasks peculiar to the problems they are solving and not supported by standard bioinformatics tools. These courses were also designed so they could be offered to students who are majoring in computer science and are interested in acquiring bioinformatics knowledge. The three courses are given in Table 1. The first and third courses have additional two hour computing lab components associated with them.

Table 1. Sequence of courses

Course Number	**Title**
BIOL/CSCI 0366	Introduction Biosciences Computing
BIOL/CSCI 0367	Biological Algorithms & Data Structures
BIOL/CSCI 0368	Introduction to Bioinformatics

4 Development of the Introduction to Biosciences Computing course

4.1 Selection of computing environment

Although the focus on tool development in undergraduate curricula should be minimal, there is nevertheless the need for some programming skills.

As a consequence of the increase in the computing needs in all fields over the past several decades, the computing tools at the undergraduate level steadily got upgraded steadily through slide rules, simple calculators, scientific programming calculators and on to graphing calculators. Now the computing needs have increased to a level that a typical graphing calculator is proving to be inadequate. As an example, simulation and modeling is becoming increasingly common in almost all fields of study, and cannot be supported by a graphing calculator. Students now need to be familiar with more sophisticated computing tools, especially those students who are studying in the science/engineering fields. In fact, we even need programming skills for effective use of common productivity tools such as spreadsheets and word processors. The more powerful functions of common productivity tools can be utilized only by writing macros. Such productivity tools are used by those working in technical as well as non-technical fields. The use of computing tools and the need for programming skills is now becoming increasingly important even in fields that are traditionally considered to be non-technical.

Graduate level bioinformatics curricula include programming courses in high-level languages such as Java, Biopython and BioPerl. Programming skills in such languages are essential for development of bioinformatics tools. Since only a few of the undergraduate biology students may continue into jobs or graduate programs that require programming skills in such languages, the teaching of programming skills in such languages is not advisable.

The next more powerful option compared to Biopython, C++, etc. is a computing environment such as MATLAB, Mathematica, R, etc. These environments are based on a very high level programming language. Programming in a very high level programming language such as R is easier as compared to programming in languages such as C++, and therefore allows more focus on problem solving.

R is a powerful general purpose computing environment, with a very high level programming language support. It is open source and therefore remains freely available to all students during and after leaving school. An extensive range of bioinformatics modules exist for the R environment and more are being developed in the open source environment, and therefore a wide variety of bioinformatics tasks can be undertaken in the R environment. For biologists, R meets the requirement as a general-purpose computing tool, a biosciences computing tool, and especially as a general-purpose molecular bioinformatics computing tool. Due to these relative advantages as compared to proprietary languages such as MATLAB, it was decided to use R package as the computing environment for this course.

4.2 Selection of contents of the course

The contents of the course were determined on the basis of:

a. The essential knowledge required for subsequent courses in the sequence.
b. Statistical skills required beyond what was covered in BIOL 0202
c. Developing the basic programming skills required to deal with unique computing tasks that may arise in the course of working with bioinformatics tools. Tasks such as sorting data, merging data, deleting selected portions from a data set, etc. require some basic programming skills.

It is very common for bioinformatics work to be done on Linux machines, and it was considered important to expose the students to the Linux environment. To achieve this, some basic Linux commands along with text editing were included in the course. Concepts typically used in bioinformatics work, including redirection, piping and shell scripting were also included.

Undergraduate biology students not only need to understand the computing relevant to molecular bioinformatics, they also need to know computing relevant to research activities typical for the biology discipline. This includes presentation of data acquired from lab experiments, by means of graphs. Many biology research activities involve the application of the scientific method, and students need to have a basic understanding of statistical techniques for data analysis.

Markov chain theory is fundamental to many bioinformatics techniques. While it is possible to use the tools without a good understanding of this topic, an informed and focused application of relevant bioinformatics tools will be facilitated by understanding the basics of Markov chain theory and its applications.

The following is a list of topics included in this course:

a. Introduction to Linux environment; basic Linux commands, text editing, redirection; piping; simple shell scripts
b. Introduction to R system; installation of R
c. Programming in R
d. Working with datasets in R
e. Basic data analysis and graphing
f. Probability theory
g. Univariate and multivariate probability distribution
h. Bayesian data analysis
i. Markov chain theory
j. Some algorithms based on Markov chains

5 Development of Biological Algorithms and Data Structures course

5.1 Selection of computing environment

It was decided to keep the computing environment the same as for BIOL/CSCI 0366 to avoid expending time and effort in yet another programming language that may not be of much use to the students after the course had ended. It was also decided to keep programming to a minimum, and teach algorithmic techniques using pseudo-code, keeping in mind the goal of producing informed users of bioinformatics tools, rather than developers of these tools. Students are asked to program some algorithms in R to demonstrate the feasibility of using R environment for bioinformatics work.

5.2 Selection of contents of the course

The initial topics in the course follow that of a typical algorithm and data structures course, starting with an introduction to algorithm analysis and complexity. Rigorous proofs are avoided, and focus is given to explaining the behavior of the algorithms. Basic molecular biology topics are covered, that are essential for a proper understanding of biological algorithms. Those algorithms are emphasized that are more relevant to bioinformatics:

a. Exhaustive search algorithms
b. Greedy algorithms
c. Dynamic programming algorithms
d. Divide-and-conquer algorithms

These topics are covered with reference to typical problems encountered in bioinformatics, especially sequence alignment.

As an example of bioinformatics specific algorithms and data structures, Biostrings defined in association with the Bioconductor package are introduced.

6 Development of Introduction to Bioinformatics course

6.1 Selection of computing environment

Alignment of a query sequence against large datasets is required frequently. These datasets are increasing exponentially in size, due to the rapid increase in the rate of sequencing. Now, computing requirements for even an introductory bioinformatics class require High Performance Computing. Even powerful workstations are inadequate.

Several different bioinformatics tools are required throughout the course. Keeping the datasets and the tools updated is a difficult task. Additionally, these computing resources will be available only on the campus, due to restrictions on off-campus access to university computing resources. The initial and recurring system administration costs for the necessary resources, is significant. For a department with no significant research in Bioinformatics, the recurring cost of maintaining and administering a HPC resource is difficult to justify for the purpose of supporting an undergraduate Bioinformatics course in a small institution.

A much better option for an undergraduate course at a small institution is to use web-based interfaces to Bioinformatics tools hosted on HPC resources installed at NCBI, EMBL and other organizations. These tools access large datasets. The tools are of high-quality and definitely adequate enough for an undergraduate curriculum. The tools as well as the datasets are frequently updated. The use of these web-based tools has the advantage of allowing students free access to the tools and datasets from anywhere, anytime. The resources remain available after graduation, and therefore the students' links to the subject material remains largely intact.

Web-based tools do not allow for automation of tasks and flexibility of processing possible with in-house HPC resources. Such automation and flexibility is necessary at the graduate level courses. However it is possible to conduct an undergraduate level course using web-based resources only. This is a necessary compromise to offset the high costs of in-house HPC resources

6.2 Selection of contents of the course

As stated earlier, it is desired that these courses should be designed such that they can be taken by computer science students also. These students do not normally take courses in biochemistry. It was decided to include some biochemistry topics that are directly relevant to nucleotide and amino acid sequences, so that the students develop a reasonable understanding of the structrues that are masked by a string of nucleotide alphabets. The central dogma of Genetics is briefly covered. Covering these topics is found to be beneficial to biology students as a refresher. Before starting on next major topic, a brief introduction to sequence alignment was done to give a feel of the most important activity in the field of molecular bioinformatics.

The next major topic is familiarization with the major NCBI and major European sequence databases, and methods to access them. The main resource used for teaching purposes is the NCBI databases, and the NCBI Entrez search engine for accessing these databases. While a large amount of sequence data are in the public domain, most publically available tools for processing these data are not user-friendly at this point in time, including those at NCBI. Consequently teaching students to use these tools effectively takes several lecture hours. It needs to be said that frequent updating of the features supported by these tools, while desirable, is also a factor that negatively impacts sustainability of consistent user-friendly interfaces for these tools. Hopefully these interfaces will improve in the next few years.

A powerful feature of these tools is the extensive cross-linking of related information and tools. A researcher can reduce his research time by using this feature effectively. It is important to make the students aware of this facility.

Another powerful feature in these search tools is the search query building facility. With the sequence and literature data increasing at a very high rate, the need to get very high quality search results, by significantly reducing false positives and false negatives is becoming crucial. The search query building feature needs considerable improvement as far as user-friendliness is concerned. The students find it difficult to understand and master this feature. Nevertheless, it is an important feature for obtaining good quality search results. Teaching students how to build effective search queries takes several lecture and lab hours.

The next topic covered was the search for appropriate scientific papers in PubMed. A good attempt has been made to standardize the user-interface across several tools available at NCBI, and in that sense the user-interface to PubMed is similar to that for searching sequence databases. However the lack of user-friendliness experienced with sequence search tools is also experienced while searching PubMed.

The lack of user-friendly interfaces may not be a handicap for a frequent user of these tools, but it is problematic when trying to teach a reasonable level of search skills amongst undergraduate students in a short period of time. Significant amount of classroom practice sessions was included in the course to develop these skills, in addition to the lab sessions. However sustainability of search skills remained an issue.

Another problem is lack of good-quality guidelines for using these tools. The documentation lags the frequent tool upgrades, and cause confusion when consulted, especially for students who need unambiguous guidelines to the latest versions of the tools. Detailed lecture slides/notes are required to bridge the gap and guide the students effectively. Lecture slides/notes need to be updated on a semester to semester basis, to keep up with changes in the search tools.

The next major topics are pairwise sequence alignment and the search of NCBI databases for sequences similar to a query sequence, using pairwise sequence alignment as the search technique. At the undergraduate level there is a need to teach students how to do sequence alignment, without in-depth statistical treatment. Statistical treatments are viable in a graduate program. At present most undergraduate biology students do not have a good grounding in the relevant statistics that form the basis for the alignment tools. The courses discussed earlier are meant to provide the requisite statistical background, but for a period of time there will be students in the class without the necessary statistical background. Realistically speaking, it will take several years for the notion to take root that mathematics and statistics have become an integral part of biological sciences. The sub-topics to be covered include alignment algorithms, the development of scoring matrices used in these algorithms and the adjustment of alignment parameters of NCBI BLAST tools. The NCBI sequence alignment tools do not allow the same level of flexibility for choosing search options as available in the command line versions, but it is more than adequate for an undergraduate course.

The problems with documentation are similar to those for the sequence search tools.

Multiple sequence alignment are also taught using web-based tools available at NCBI and EBI. Simple phylogenetic analysis is practiced using PHYLIP package for genetic phylogenetic trees and NCBI taxonomy database tools for species trees.

7 Conclusion

Inclusion of bioinformatics foundational knowledge in an undergraduate biology curriculum with computational genomics track is a challenging task. In the course sequence discussed in this paper, important bioinformatics concepts are taught, with the introduction of only the essential mathematics, statistics and statistics material necessary to provide clear concepts. The focus of the course sequence is on the use of bioinformatics tools, rather than the development of bioinformatics tools.

8 References

[1] G. Cooper, "Design and Implementation of an Undergraduate Bioinformatics Curriculum in an Online Environment.," 2007.

[2] M. Maloney, J. Parker, M. LeBlanc, C. T. Woodard, M. Glackin and M. Hanrahan, "Bioinformatics and the Undergraduate Curriculum," *CBE- Life Sciences Education*, vol. 9, pp. 172-174, 2010.

[3] D. T. Burhans, M. DeJongh, T. E. Doom and M. LeBlanc, "Bioinformatics in the undergraduate curriculum: opportunities for computer science educators," in *Proceedings of the 35th SIGCSE technical symposium on Computer science education (SIGCSE '04)*, New York, 2004.

[4] A. N. Kumar, R. K. Shumba, B. Ramamurthy and L. D'Antonio, "Emerging areas in computer science education," in *Proceedings of the 36th SIGSE technical*

symposium on Computer science education (SIGSE '05), St. Louis, Missouri USA, 2005.

[5] J. Cohen, "Guidelines for Establishing Undergraduate Bioinformatics Courses," vol. 12, no. 4, 2003.

[6] M. R. Rose and T. H. Oakley, "The new biology: beyond the Modern Synthesis," *Biology-Direct*, vol. 2, no. 30, 24 November 2007.

[7] Dept of Mathematics & Statistics, Hunter College, CUNY, "Mathematics and Statistics Programs," 2012. [Online]. Available: http://math.hunter.cuny.edu /graduate.shtml. [Accessed 25 February 2012].

[8] Department of Computer Science, New Jersey Institute of Technology, "NJIT: Computer Science: MS in Bioinformatics (MS BioInf)," 2012. [Online]. Available: http://cs.njit.edu/academics/graduate/ms-bioinf/index.php. [Accessed 25 February 2012].

[9] Dept of Biological Sciences, Carnegie Mellon Unversitiy, "Computational Biology & Bioinformatics-Dept of Biological Sciences-Carnegie Mellon University," 2012. [Online]. Available: http://www.cmu.edu/bio/research/compbio.html. [Accessed 25 February 2012].

[10] D. Collins, "Human Genetic Careers," 2011. [Online]. Available: http://www.kumc.edu/gec/prof/career.html. [Accessed 06 Nov 2011].

A Curriculum Coordination Project for Computer Science Transfer

Dr. Xiaohong (Sophie) Wang
Department of Mathematics and Computer Science
Salisbury University
1101 Camden Avenue
Salisbury, MD 21801 U.S.A.

Key words: Articulation, Transfer courses, Computer science courses

Abstract

The curriculum articulation between 2-year to 4-year institutions plays an important role in postsecondary education pipeline. An NSF funded program at Salisbury University sponsored a year-long curriculum coordination project in computer science with two of its regional feeder community colleges. Progress has been made in aligning the existing curriculum of the three institutions with national standards; increasing the number of transferrable courses; and mending the current course content to ensure the success of transfer students. Issues in the current transfer process have also been identified and both temporarily solutions and more permanent remedies are proposed. The effort by this curriculum coordination project is an important first step in the meaningful communication and collaboration among computer science faculty at the three institutions and it marks the beginning of an on-going effort to enable a seamless transfer process.

1. INTRODUCTION

According to the American Association for Community Colleges, nearly half of the undergraduate students are enrolled in 2-year colleges. The missions of community colleges are multiple: transfer preparation, associate degree and certificate programs, and continuing education. To serve a diverse student population, community colleges need to assist each student in determining which program best serves his/her educational and career goals. While associate degree and certificate programs are career-oriented and provide the specific knowledge, skills, and abilities for the workplace directly, transfer-oriented programs provide the academic foundation and pathway to a baccalaureate degree at a 4-year college or university.

As the number of 2-year college to 4-year college transfer students continues to grow, paying attention to the 2-year to 4-year college transfer is of extremely importance. However, according to Cuseo (2007), there is a significant gap between the number of students who enter 2-year colleges with the intention of transferring to 4-year institutions and the number who actually do transfer. For students who do transfer, they are likely to encounter significant post-transfer adjustment difficulties during their first term of enrollment at 4-year institutions. Many man-made barriers in policies and procedures interfere with a seamless transfer from 2-year to 4-year institutions. From the aspect of curricula, many confusion and difficulty exist regarding the transferability of courses from 2-year to 4-year colleges. Sometimes, 4-year institutions may refuse to accept transfer courses duo to curricular rigidity; or accept those courses as elective credits instead of credits toward general education or an academic major. Curricular changes in 4-year institutions are usually made without consideration of their implications for potential transfer students.

Currently transfer programs rely on formal cross-institutional articulation agreements. Two latest computer science curricular standards by the ACM and the IEEE Computer Society: “Computer Science 2008” for 4-year programs and “Computing Curricula 2009: Guidelines for Associate-Degree Programs in Computer Science”, establish transfer equivalencies between 2-year and 4-year programs to foster articulation in computing science education. The standards emphasize “effective and efficient articulation” among sending and receiving institutions and assign the responsibility for successful articulation to both students and faculty. An efficient and effective articulation requires accurate assessment of courses and curriculum content as well as meaningful communication and cooperation. When a 2-year college develops transition and articulation strategies for the institutions to which its students most often transfer, it should be flexible to modify course content to facilitate transfer credit and articulation agreements. The standards also recognize the importance of the receiving institution's faculty providing transitional preparation for transfer students. The reports remind faculty at receiving institutions to be sensitive to the needs and issues of transfer students.

In the summer of 2010, Salisbury University (hereafter SU) was awarded a 5-year NSF STEM Talent Enhancement Program grant to develop Bridges for **S**alisbury **U**niversity's **C**onnections to **C**areers for **E**very **S**TEM **S**tudent (SUCCESS) program. Students who initially enter college with the intention of majoring in science, technology, engineering, or mathematics (STEM) fields have substantially lower completion rates in these disciplines than do their peers who enter with aspirations for a non-STEM major (Huang et al. 2000). The Bridges for SUCCESS program is designed to increase the number of graduates in selected STEM disciplines by 75% within five years through expanded outreach, recruitment, enrichment, and retention activities. Using this grant, SU is working to create bridges for success by mentoring students from high schools and community colleges through baccalaureate degrees and then on to careers in a STEM field.

One of the five activities designed to create these bridges is to facilitate the seamless transition of community college students to SU STEM majors through academic and transition support. For academic support, SU established three curriculum coordination teams for computer science, physics, and earth science. Each team composed of faculty from SU and two of its regional "feeder" community colleges, Chesapeake College (CC) and Wor-Wic Community College (WWCC). The purpose of this curriculum coordination is to develop processes for evolving disciplinary standards, to improve student learning at both 2- and 4-year institutions and to improve the success of students transferring from the two community colleges to SU. The curriculum coordination teams selected the introductory courses within each discipline and worked to better align the curricula for those introductory and supporting courses. They also reviewed the discipline standards for each discipline and assessed the gap between their existing curricula. Issues that hinder the seamless transition process were identified and recommendations to work on those issues were proposed to their respective departments for consideration.

This paper reports the curriculum coordination activities for computer science (CS) program. The rest of the paper is organized as the following: Section 2 reports the activities conducted during the CS curriculum coordination. Section 3 discusses the CS transfer issues between WWCC, CC and SU and proposes recommendations to resolve those issues. Section 4 gives a brief summary of this year-long project.

2. CURRICULUM COORDINATION ACTIVITIES

As one of the selected STEM disciplines in SU Bridges for SUCCESS program, the CS curriculum coordination team was constructed by two faculty members from SU, CC and WWCC respectively. The team met seven times between August 2010 and June 2011. Between those meetings, the team actively communicated via emails. Several activities were conducted to: i) better align the curricula for introductory and supporting CS courses among SU, WWCC and CC; ii) map those introductory courses with the national standards; iii) enhance effective transfer of CC and WWCC courses through interaction and communication among faculty members; and iv) propose recommendations for issues in CS transfer process.

2.1 Alignment with National Standards

The CS team reviewed the latest national CS curricular standards: "Computer Science 2008 for 4-year programs" and "Computing Curricula 2009: Guidelines for Associate-Degree Programs in Computer Science". The "Computing Curricula 2009" strongly recommends that the entire CS I, CS II and CS III core course sequence and a minimum of two mathematics course: Discrete Structures and Calculus I, should be completed at 2-year institutions. The document also outlines the student learning outcomes (SLOs) for CS I, CS II and CS III.

After reviewing the SLOs for those three courses, the team created a mapping of the SLOs and their equivalents at SU, WWCC, and CC. The mapping results for CS I show that both SU and CC have either met or surpassed all eight SLOs and WWCC needs to meet or surpass four more SLOs. Both SU and CC have either met or surpassed all eight SLOs of CS II and WWCC needs to meet or surpass two more SLOs. For CS III both SU and CC have either met or surpassed seven out of the eight SLOs except the topic on information security. WWCC does not offer courses equivalent to CS III.

Two math courses, Discrete Structures and Calculus I, are recommended by the two standard reports for transfer CS programs. Both SU and CC offer those two courses. WWCC offers Calculus I but does not offer discrete mathematics. Since this curriculum coordination focuses on the CS courses only, mapping between the national standards for those two courses and SU, CC and WWCC was not conducted. However issues for current transfer and recommendations for better transfer of those two courses were given later in the paper.

The "Computing Curricula 2009" report also recommends the inclusion of additional intermediate computing courses to be transferred. However, whether each individual course will be transferred or not is determined by each institution based upon the resources of the 2-year institution and the size of the CS transfer program. In summary, out of the thirteen recommended courses, SU offers nine, CC offers eight, and WWCC offers four. The detailed result of this mapping is summarized in Table 1 in Appendix.

Both reports include many security topics in computing programs. However, the security topics are not presented in lower level CS curricula for all three institutions involved.

2.2 Review Existing and Identify New Transfer Courses

Another activity performed by the CS curriculum coordination team is to review the current CS course curricula of CC and WWCC to discover the content discrepancy of the existing transfer courses and identify new transfer courses.

WWCC's Transfer Program

Before the curriculum coordination project, there were two existing transfer courses from WWCC to SU: CS I and Calculus I. CS I has been accepted as a transfer course for many years. However, beginning fall 2010, SU added several advanced topics such as classes, objects, constructors and copy constructors, assignment operator overloading, destructors, operator overloading, pointers, and dynamic memory allocation to CS I. Although the transfer agreement for this course is not affected by those changes, to help transfer students succeed in CS II at SU, the inclusion of the above mentioned advanced topics to WWCC's CS I is highly recommended. Calculus I is another course transferrable to SU. However, the previous experience has shown that WWCC's transfer students have difficulty completing the successor course, Calculus II, at SU. Since the review and discussion on math courses is out of the scope of this curriculum coordination project, only recommendation on this issue is given in the end of the paper.

During the curriculum coordination project, two additional computer science courses from WWCC have been identified as transferrable to SU: Programming Fundamentals and Microcomputer Organization. WWCC faculty and SU faculty who teach those two courses met and determined that those two course content from WWCC satisfies the corresponding course requirements for SU's courses respectively.

CC's Transfer Program

There were six CS courses from CC transferrable to SU before this project: Microcomputer Organization, CS I, Discrete Structure, Calculus I, Calculus II and Statistics. These transfers have been in place for many years. Since the CS faculty at CC and SU maintain regular contact in the past, changes made to SU's courses have always been incorporated in CC's curriculum promptly.

One point worth noting is that, although CC's Statistics course has always transferrable, when SU added a lab component to its Statistics course, transfer students were required to take the 1-credit lab course to meet SU's full requirement for this course. Recently, recognizing the problem caused for transfer students, SU has eliminated the lab requirement for transfer students.

An agreement was signed between CC and SU to allow two additional CC courses to be transferred to SU. During one of the curriculum team meetings, CC faculty and SU faculty who teach Programming Fundamentals and CS II met and determined that those two course content from WWCC satisfies the corresponding course requirements for SU's courses respectively.

2.3 Course Transfer Agreement

ArtSys is the articulation system used by Maryland colleges and universities for transfer courses. In the past, ArtSys has recognized transfers of several individual CS courses from CC and WWCC to SU. However, there is always a delay in updating ArtSys with the new transfer courses due to administrative delay and rigidity. To be able work around those man-made barriers, the CS curriculum coordination team recommended the use of memo of understanding signed by relevant parties to deal with case-by-case transfer and newly identified transfers. During the year-long curriculum coordination activities, the team developed two memos of understanding. The first letter of understanding is between CC and SU. Since the CS program at CC has been terminated effective May 2011, the memo serves to facilitate the smooth transfer of two CC students to SU's CS program within the next year. This memo was signed between CC and SU to give them the credits for several courses taken at CC. Another memo of understanding was signed by WWCC and SU. It lists all existing and newly identified WWCC's CS courses transferrable to SU. This memo can be used to help incoming transfer CS students from WWCC temporarily before ArtSys gets updated with the new transfer courses.

2.4 Road Map for CS Transfer from WWCC or CC to SU

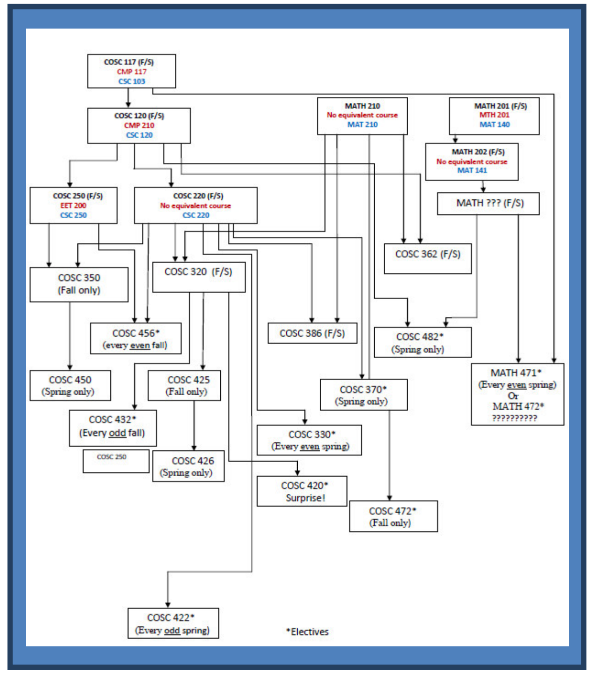
Another deliverable produced by the CS curriculum coordination team is a road map (see Figure 1) for CS transfer students from WWCC or CC to SU. In this road map, all courses transferred to SU have been incorporated to the chart. By examining this chart, any transfer CS students can see clearly where they are in the CS program at SU. The road map will be available on WWCC and CC campus so that they can be available easily to interested students. Again, in order to ensure a seamless transition, a student who is enrolled in a CS transfer program is responsible for meeting with his/her transfer advisor each semester and check SU's requirements periodically. However, it is also expected that if SU's CS program requirements and/or its CS course content change, the WWCC and CC will be notified in a timely manner.

3. ISSUES AND RECOMMENDATIONS

During the curriculum coordination process, the team has identified several existing issues that hinder the success of transfer students from CC or WWCC to SU. Those issues are summarized as followings:

1. Courses that transfer but do not fully prepare students for subsequent courses at SU. Experience has shown that many of WWCC's transfer students do not have sufficient amount of knowledge or skills to do well in subsequence courses at SU. There are many factors contribute to this. One factor is that insufficient amount of content is taught at WWCC. Another factor is that the changes made to those transferrable courses at SU also raise the expectation of transfer students.
2. Courses in SU's year 1 and year 2 curricula that are not required by the sending institutions' CS programs. For example, WWCC does not offer Discrete Mathematics, Calculus II, or CS II because their CS program does not require those courses. Thus WWCC's CS transfer students must take those courses at SU and hence will encounter some delay in their graduation.

Figure 1. Road map for CS transfer students (Black – SU, Red – WWCC, Blue-CC)



3. Different and/or new programming environments and operating systems are used at the receiving institution. For example, at SU, CS students are required to use Linux for CS II and many upper-level CS courses. However, some WWCC and CC students may not have experience with a UNIX/Linux program development environment before transferring to SU. This is very frustrating for transfer students and often causes a lot of anxiety when students transfer to SU.
4. ArtSys is not always updated in a timely manner. Difficulties in keeping ArtSys updated make it harder for prospective students to determine if a course or program is, in fact, transferable. Although everyone agree that it is important that when changes that affect transferability are made to a program or course, the changes should be reflected in ArtSys, no one seems to know who and how to get ArtSys to recognize a course or program as officially transferrable.

To help address those above identified issues, the curriculum coordination team provides the following recommendations:

1. The dialog among the CS faculty at the sending and receiving institutions established by this team should continue. Collaboration among the mathematics faculty at the three institutions must also be encouraged. Transfer course content and

student success in subsequent courses at the receiving institution must be monitored.

2. To strengthen the chance for success of WWCC's CS transfer students, WWCC needs to: i) meet or surpass several SLOs specified in the national standards for the CS I and CS II courses and ii) keep its CS II course content consisted with SU by including the advanced topics discussed earlier.
3. Several courses (Discrete Mathematics, Calculus II, CS II) in SU's year 1 and year 2 curricula are not required by the WWCC computer science programs. The team recommends that WWCC consider adding these courses when its resources allow this.
4. An effort must be made to accommodate students who are new to Linux at SU. The team recommends that SU provides a few self-paced or assisted lab sessions on the basics of using Linux for editing, compiling, linking, and debugging programs to students with no prior Linux programming experience.
5. The administrative issues in updating ArtSys need to be clarified. If guidelines on how to update ArtSys exist, they must be published and easily accessible by all relevant institutions. The contact person responsible for completing the process must be known to faculty at both the sending and receiving institutions.
6. Both the national standards for CS programs stress the incorporation of information security topics to computing programs. However, none of the three institutions involved have security topics in their lower level CS curricula. This situation needs to be changed as soon as possible in order to keep their CS curricula up to date.

4. SUMMARY

The main focus of this CS curriculum coordination team was to ensure a more seamless transfer for CS students from WWCC and CC to SU by reviewing the national standards and the current CS curriculum content, reviewing existing and new transfer courses, identifying the issues of the current transfer process, and providing recommendations.

When two national CS standards were reviewed, the SLOs for the three CS course sequences defined in the standards were examined, and a determination was made on where each of the three institutions stands for each SLO.

When this project began in fall 2010, only a handful of CS courses from WWCC and CC transferred to SU. Also before the project started, there was no communication between CS faculty at WWCC and SU. Due to the efforts of this curriculum coordination team, the number of transferrable CS courses has been increased significantly. This collaboration has also led to a better understanding of the CS curricula of the three institutions.

Course articulation is now well-defined via memos of understanding. These memos will assist the transfer of CS students when ArtSys does not reflect the most updated course articulation agreements. A "Road Map" was created so that transfer students can use it to see at a glance the SU course to which their associate level courses transfer, and their standing in SU's program.

Although 100% articulation is not possible at this time, the collaboration among faculty at three institutions during this curriculum coordination project marks the beginning of on-going effort to enable students to transfer in as seamless a manner as possible. Faculty members at the receiving institution are more sensitive to the needs of incoming CS transfer students, and steps are being taken to ensure that those needs are met. Those meaningful communication and cooperation and ongoing assessments of courses are essential for transfer students' success, and the effort by this curriculum coordination team has been an important first step.

5. ACKNOWLEDGMENTS

The author would like to thank Don Nicholson, Mary Lou Malone, and Sang-Eon Park and other faculty members from SU, CC and WWCC for working together on this project. This project is supported by SU Bridge for SUCCESS undergraduate research grant funded by the NSF.

6. REFERENCES

[1]ACM and IEEE Computer Society Interim Review Task Force, 2008: *Computer Science Curriculum 2008: An Interim Revision of CS 2001*.

[2]ACM Two-Year College Education Committee, 2008: *Computing Curricula 2009: Guidelines for Associate-Degree Transfer Curriculum in Computer Science*.

[3]Cuseo, J., 2007: *The Transfer Transition from 2-year to 4-year Institution: Critical Issues and Promising Practices*. http://www.uwc.edu/administration/academic-affairs/esfy/cuseo/

[4]Huang, G., Taddese, N., and Walter, E. 2000: *Entry and persistence of women and minorities in college science and engineering education*. Washington, D.C.: National Center for Education Statistics.

Appendix

Table 1. Additional (optional) Courses for 2-year Transfer Program Mapping

Courses Recommended	Equivalents at WWCC	Equivalents at CC	Equivalents at SU
Algorithm Analysis & Design	None	CSC 220	COSC 220, 320
Computer Organization & Architecture	EET 200 Microprocessors	CSC 250 Microcomputer Organization	COSC 250 Microcomputer Organization
Hardware Fundamentals	Introduced in CMP 115, Fundamentals of Computer Architecture	CSC 250, Microcomputer Organizations	COSC 250 and COSC 456 Computer Architecture
Human-Computer Interaction	None	None	COSC 425 & 426, Software Engineering I and II
Essentials of Computer Security	None	None	None
Intro. to Database Systems	CMP 255 Database Design & Management	Course exists but not required in CS program.	COSC 386 Database Implementation
Intro. to Software Engineering	None	Concepts introduced throughout programming courses.	COSC 425 & 426, Software Engineering I & II
Linux Operating System	Course that introduced Linux exists.	Course on Linux exists.	COSC350 System Programming, 450 Operating Systems. Use Linux in many CS courses.
Programming Languages	None	None	COSC 422 Organization of Programming Languages
Survey of Computing Disciplines	None	CIS 109 Introduction to Computers.	None
XML-Enabled Technologies	None	None	None
Digital Logic Circuits with Lab	None	CSC 250 Microcomputer Organization	COSC 250 Microcomputer Organization
Circuit Analysis with Lab	None	None	None

A Curriculum Development for the Undergraduate Degree Program in Information Technology

Yingbing Yu[1]
[1]Department of Computer Science & Information Technology
Austin Peay State University, Clarksville, TN, USA

Abstract - *Information Technology is an academic discipline distinct from computer engineering, computer science and management information systems. Information technology encompasses software engineering and development, computer networking and communications, Web technologies, computer security, database management, and digital media technologies. In this paper, we present a complete curriculum development for a new B.S major in Information Technology which was done through the combined efforts and inputs from both the academic and industry perspectives. It can serve as a reference for peer institutions to create a similar program in this area.*

Keywords: Information Technology Education, Curriculum Development, SIGITE, ABET

1 Introduction

Information technology (IT) has the impacts on every part of our lives. From Internet banking that helps us pay our bills instantly to the movie special effects we see on the big screen, IT is changing the way we see and experience the world. Behind the rapid innovation and development of IT are skilled professionals who keep our high-tech world moving. IT graduates understand this complex and fast-changing environment, and have the knowledge and skills to get things working. As computers contribute increasingly to our creativity, communications, entertainment, and well-being, the demand for IT graduates is continuing to grow.

Previously the author had the opportunity to serve as a faculty at a private college and work on a proposal to create a new major in information technology and develop the whole curriculum. We are introducing a new B.S. program in information technology with two concentrations in Web and Database Development and in Network Administration and Security. These broad, innovative programs go beyond the traditional boundaries of computer science and incorporate work from disciplines across the college.

The rest of this paper is organized as follows. Section 2 is the background review of IT programs in general. Section 3 presents the whole IT curriculum that consists of two concentrations and one minor degree. Section 4 discusses the assessment measures for the new IT program. The paper concludes with section 5.

2 Background

In the last decades, we have seen the trend that a growing number of colleges and universities developed the information technology curricula that were alternatives to the traditional computer science and information system curriculum. Rochester Institute of Technology has one of the earliest and probably one of the most comprehensive IT curricula [1]. Others schools that also offer the major include Brigham Young University [2], Georgia Southern University [3], George Mason University [4], New Heaven University [5], University of Massachusetts Amherst [6], Purdue University Calumet [7], University of Cincinnati [8], and so on.

Information technology in its broadest sense encompasses all aspects of computing technology. IT, as an academic discipline, focuses on meeting the needs of users within an organizational and societal context through the selection, creation, application, integration and administration of computing technologies. IT programs aim to provide IT graduates with the skills and knowledge to take on appropriate professional positions in Information Technology upon graduation and grow into leadership positions or pursue research or graduate studies in the field.

The goal of our IT program is to enable any interested student to confidently employ IT, and to secure an intellectual platform from which to develop capacity to innovate, using IT in his or her field. Our highly qualified faculty members are involved in professional development through different mechanisms. Most of our faculty members attend different professional meetings. A few faculty members also take additional graduate level courses to keep themselves current in their fields.

3 Information Technology Curriculum

This IT program consists of a common core that exposes students to a wide range of computing and technology topics, including the study of databases, hardware, networks, programming, and human/computer interaction. Advanced courses are selected from one of two concentrations: Web and Database Development or Network Administration and Security. A student also may select to complete a minor in another discipline.

Areas of application include webpage design and development, database administration and maintenance, and network development and administration. Typical initial job titles might be web developer, network technician, applications developer, and network security technician. With several years of experience job titles might be website administrator, network administrator, database administrator, and security manager.

The courses in the IT program consists of core courses plus specialty courses offered through concentrations. Core courses are courses required to be taken by all students in the program. This includes courses in the general education, in addition, to core courses that provide breadth of knowledge in all of the IT areas. In addition, some courses are cross-listed in both the IT and the existing Computer Science (CS) B.S. program.

3.1 Goals

The IT program is the result of combining existing Computer Science degree. The major will equip students with a broad background across fundamental areas of IT along with a depth of understanding in a particular area of interest. It will also prepare for graduate studies in fields such as networking, information security, database, web development and related information technology areas. The objective of the Information Technology degree is to provide students with the following:

- Meet the national accreditation criteria for the Accreditation Board for Engineering and Technology (ABET) [9].
- Empower students with more specialized technical skills and knowledge.
- Reflect the need of the industry and community.
- Meet the needs of all different segments of the student population.
- Possess the ability to integrate the components of information systems.
- Design and/or implement an information system to meet organizational needs.

3.2 Major Requirements

The IT program can be successfully completed in 8 full-time semesters with an average of 15 credits each semester. The 120 semester hour degree requirement consists of General Education requirements from the college, IT core courses, courses required for the chosen IT concentration area and electives.

At least 30 semester hours toward the BS in Information Technology degree must be earned at the college, and at least 45 semester hours of the degree requirements must be level 300 or above. Students pursuing the BS in Information Technology must complete requirements for at least one of two IT Concentration areas: 1) Network Administration; 2) Web and Database Development.

3.3 Core Courses

In addition to General Education requirements, including humanities and social sciences as well as mathematics and basic sciences requirements, the IT major requires IT core (36 credit hours), and concentration courses as described below. Each concentration includes a six-hour IT project course and a three-hour capstone course with the focus on the exit comprehensive examination. It is expected that the student will complete the IT design project over a period of two consecutive semesters.

The courses in the IT program consists of core courses plus specialty courses offered through concentrations. Core courses are courses required to be taken by all students in the program, in which some are cross-linking courses both offered in the IT and CS program.

- ITEC 110 Introduction to Information Technology
- ITEC 116 Introduction to Programming
- ITEC 216 Programming in C++
- ITEC 316 Object Oriented Programming
- ITEC 240 Discrete Structures
- ITEC 215 Web Page Development
- ITEC 320 Computer Hardware and Software
- ITEC 360 Human Computer Interaction
- ITEC 345 Database Management
- ITEC 325 Computer Networks

- ITEC 455 Information Technology Project
- ITEC 460 Information Technology Capstone

3.4 Concentrations

In addition to the core, students must take an IT concentration. The department will offer two concentrations, each of which requires a student to take 4 additional courses (12 credit hours).

1) *Concentration - Network Administration and Security*

- ITEC 325 OS Scripting
- ITEC 335 Network Administration
- ITEC 430 TCP/IP Internetworking
- ITEC 435 Network Security

2) *Concentration - Web and Database Development*

- ITEC 315 Client-Side Web Programming
- ITEC 350 Database Programming
- ITEC 415 Server-Side Web Programming
- ITEC 445 Database Administration

3.5 Elective Courses

Students need to take 2 additional elective courses (6 credit hours) to complete the IT degree requirement.

- ITEC 360 Human Computer Interaction
- ITEC 450 Special Topics
- ITEC 499 Information Technology Co-Op Experience
- COSI 121 Visual BASIC
- COSI 323 Advanced JAVA Programming
- COSI 330 Data Structures

3.6 Information Technology Minor

We also provide the options of a minor in Information Technology to those students who may just want to get a minor in Information Technology for their career. The following courses (15 credit hours) must be taken to be considered a minor in IT program.

- ITEC 120 Introduction to Information Technology
- ITEC 216 Programming in C++
- ITEC 305 Web Page Development
- ITEC 310 Introduction to UNIX
- ITEC 330 Computer Networks

4 Assessment Measures

As part of the program evaluation process, we have acquired valuable inputs from peer institutions with existing IT program and local major employers for the assessment of their needs. Based on these combined efforts and information, we defined assessment mechanism as the justification of creating the new IT major and effective measures as the desired outcome for the Information Technology major as follows.

4.1 Assessment Mechanism

- The enrollment in Computer Science major is down in recent years due to the shrinking job market.
- The employers from the local community and graduates expressed their concerns that our current majors are not adequately prepared for the job market.
- The major is expected to attract and retain more students who will be able to find more job opportunities especially in the Memphis Metropolitan Area, MS-AK-MS.
- The school has proposed to create a major in Information Technology to provide a better foundation for African American students, and especially women, in math and science.

4.2 Desired Outcome

- Acquire good knowledge in IT along with more specialized technical skills.
- Design and implement a system for a real application and be ready for employment at a professional level in industry.
- Be able to communicate effectively and efficiently with clients, users, and peers, both orally and in writing.
- Demonstrate independent critical thinking and problem-solving skills.
- Increase in enrollment, more job opportunities, and better prepared students for the job market. Demonstrate an understanding of the need for continued learning throughout their career.
- Work effectively in project teams to develop and/or implement IT-based solutions.

5 Conclusions

Information Technology is an academic discipline distinct from computer engineering, computer science and management information systems. IT encompasses software engineering and development, computer networking and communications, web technologies, computer security, database management, and digital media technologies. The IT professional is hired by organizations of all sizes in all industries. Students will receive a broad education across the IT spectrum as well as technical specialization in the areas of their choice.

We present the whole curricula development for the new B.S major in Information Technology. To empower students with more specialized technical skills and knowledge, the major will offer two concentrations: 1) network administration and security, and 2) web and database development. The B.S. program in IT should meet the students' needs in the metropolitan area and equip them with the skills and knowledge to take on appropriate professional positions in Information Technology upon graduation.

6 References

[1] http://www.it.rit.edu. Rochester Institute of Technology, Rochester, New York.

[2] http://sot.et.byu.edu/. Brigham Young University, College of Engineering and Technology, School of Technology.

[3] http://cit.georgiasouthern.edu/, Georgia Southern University.

[4] http://www.cs.gmu.edu/programs/masters/infs/bsit-is.html. George Mason University.

[5] http://www.newhaven.edu/9550/. University of New Haven.

[6] http://www.umass.edu/itprogram/. University of Massachusetts Amherst.

[7] http://webs.purduecal.edu/citg/. Purdue University Calumet.

[8] http://www.cech.uc.edu/it/. University of Cincinnati.

[9] http://www.sigite.org/. ACM Special Interest Group for Information Technology Education.

Modernizing the Real-Time and Embedded Curriculum

Kelvin Nilsen, Ph.D.
Chief Technology Officer Java, Atego Systems, Inc.
5930 Cornerstone Court West, Suite 250, San Diego, CA 92121, USA
kelvin.nilsen@atego.com

Abstract - *Traditionally, real-time and embedded software design and implementation has been the purview of computer engineering departments. A course dedicated to these topics, if it is offered at all, is often accompanied by aspects of hardware design and hardware device interfaces. Programming assignments usually use C or assembly language.*

This paper argues that traditional approaches to educating real-time and embedded engineers are outdated and fail to prepare students for typical modern real-time software engineering challenges.

Modern embedded real-time systems are very different than the embedded real-time systems of 20 years ago. Among trends that have not been addressed in the typical embedded real-time curriculum are first, a tremendous increase in size and complexity of embedded system software, with many systems doubling in size with each new product release, typically occuring every 18 to 36 months; second, a tendency to build each new product release primarily by integration of existing software components rather than by implementing all of the functionality for the new product from scratch; and third, the reality that reusable embedded and real-time software components are required to provide multiple decades of service on a rapidly evolving assortment of hardware and operating system platforms.

Keywords: Real-Time Java, Embedded Real-Time, Curriculum

1 Current Industry Practice

While it is still the case that a large number of embedded systems are still programmed in C and assembler, it is also true that increasingly, embedded real-time systems are being implemented in more modern object-oriented programming languages like Java [1]. Note, for example, that the highly popular Android platform uses Java as its preferred programming language [2]. Real-time Java has been used in the implementations of critical telecommunications infrastructure, manufacturing of semiconductors, automation of off-shore oil drilling rigs, multiple RADAR applications, situational awareness tools for military infantry, and Aegis warship weapons control including ballistic missile defense software [3-7].

The motivations for the increased use of Java in the implementation of embedded real-time software systems are several fold.

1. Typical development projects find that Java offers a two-fold gain in developer productivity and five- to ten-fold cost savings during software integration and maintenance activities in comparison with legacy languages like C and C++ [8].
2. The size of embedded software systems is growing exponentially, in close correlation with Moore's law. This is exhibited in Figure 1, which reports trends in NASA space mission software [9]. Similar trends have been reported by the consumer electronics and automotive software industries [10].
3. While computer processing speeds and memory capacities continue to increase while the costs of computing resources are declining, the costs of software engineers continue to rise. This motivates tradeoffs that allow software engineers to be more productive at the expense of slightly less efficient use of computing resources.
4. Whereas the embedded software systems of past decades were small enough that one or two programmers could implement all of the code for each new device in less than a year of development, today's embedded systems are far too large and complex to support that model. The typical difference between last year's product and next year's product is a doubling of code size, from maybe 1 million lines of code to 2 million lines of code! All of the new code, most of which is supplied as reusable software components by third parties, must integrate cleanly with code deployed in the prior product release and with all of the other components to be added to the new product release.

Some representative real-time Java applications are described in Table 1. Because these applications represent competitive advantages to their developers, certain details are omitted from some of the descriptions, as marked with the Ψ symbol.

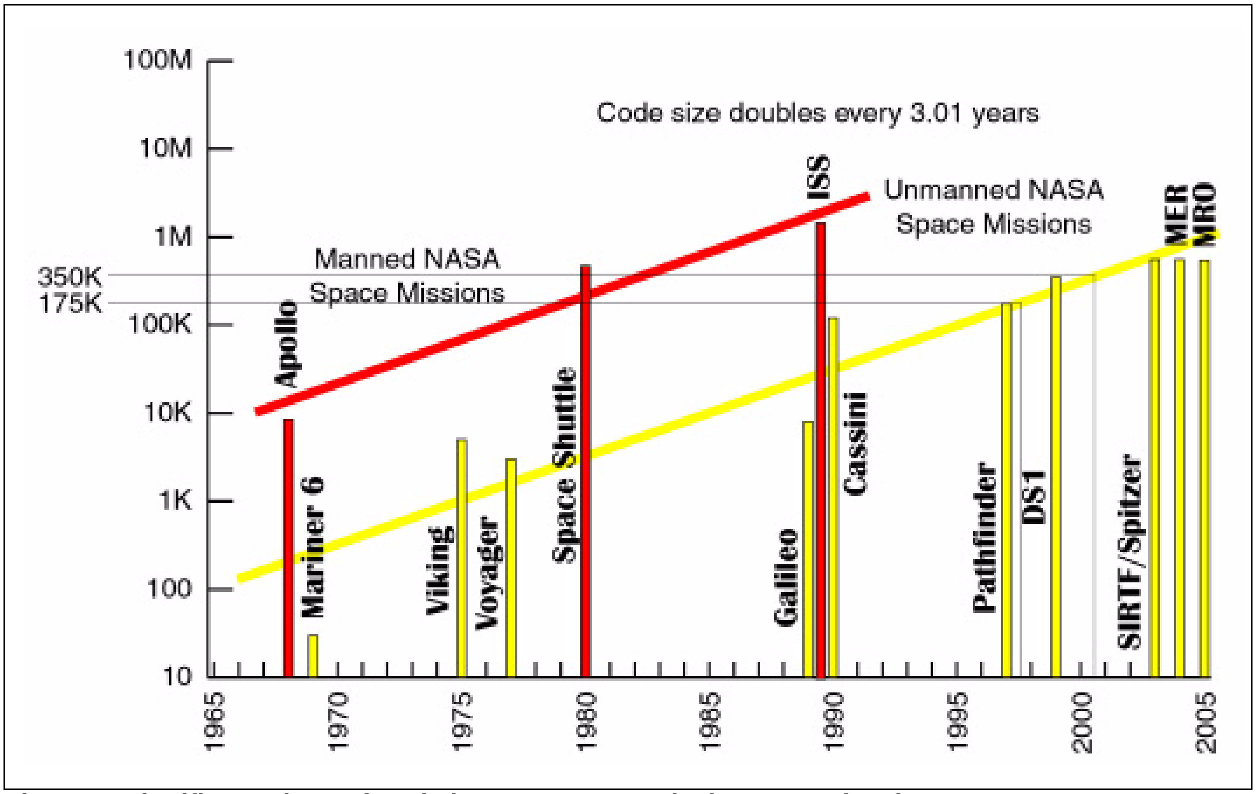


Figure 1. Significant Lines of Code in NASA Space Mission Control Software

Domain	Size	Periods
Fleet Telematics	**~6 man years**	**100 ms**
Semiconductor Manufacturing	**thousands of control points**	**1 ms**
Off Shore Drilling Rigs	**Ψ**	**10 ms**
Power Plant Automation	**thousands of control points**	**16.7 ms**
Helmet Mounted Situational Awareness for Foot Soldiers	**tens of man years**	**Ψ**
In-Flight Entertainment System	**tens of man years**	**100 ms**
Multi-Protocol Broadband Loop Carrier	**~10 man years**	**Ψ**
Long Haul Fiber Optic Telecommunications Switch	**~100 man years**	**40 ms**
Warship Weapons Control and Ballistic Missile Defense	**500K SLOC**	**sub-ms**

Table 1. Sample Real-Time Java Applications

2 Relevant Trends in Computer Science Education

In the 90s, the language of choice for most undergraduate Computer Science courses was C. Following its public release in 1996, Java has gradually taken over the role of dominant programming language for university instruction [11, 12]. Professors have found that the use of Java allows students to explore more interesting topics with less low-quality time spent by students, professors, and teaching assistants in lengthy debugging sessions. As a higher level language, Java completely eliminates the core dumps that frequently result from pointer arithmetic errors, array subscripting errors, the dereferencing of null and dangling pointers, and inappropriate type coercions.

Furthermore, the choice to use Java allows instructors to expose students to good object-oriented practice as one of the side benefits of each class room assignment. This is important because object orientation is one of the key mechanisms used by industry to manage the complexity of today's ever increasing software system sizes. Being familiar with object-oriented principles and techniques helps graduating students find employment.

3 Suggested Embedded Real-Time Curriculum Outline

The official curriculum recommendations for an elective course on embedded and real-time software reflect biases that were most appropriate in prior decades, when all of the software for each embedded real-time device could be written from scratch by one or two programmers in less than a year. Only four topics are suggested for the course [13]:

1. Process and task scheduling
2. Memory/disk management requirements in a real-time environment
3. Failures, risks, and recovery
4. Special concerns in real-time systems

This official list of topics leaves room for interpretation, allowing instructors to exercise discretion in choosing which "special concerns in real-time systems" they might want to include in the course. The reality is that if a course chooses to focus on the implementation of process and task scheduling and memory and disk management, there may be little time available to focus on topics relevant to the development of real-time applications.

The outline of topics provided below aims to educate students in general principles of real-time application development. While many real-world development efforts may choose not to use a high-level programming language, the principles learned in this class are for the most part relevant to all real-time development, regardless of language. The choice to use a high-level programming language for course assignments allows more interesting topics to be studied in greater depth.

1. **Rate monotonic scheduling theory**: An important question that must be addressed in developing a real-time application is how to analyze the workload to prove that all tasks will satisfy their real-time constraints. While many real-time scheduling theories exist, we recommend that an introductory course present rate monotonic theory, as it is the most widely used analysis technique and is relatively easy to understand.
2. **Empirical analysis of time and memory requirements for real-time tasks**: In order to apply schedulability analysis, it is necessary to understand the CPU time consumed by all of the tasks in the workload. This is computed as the product of worst-case task execution times and task execution frequencies. Assuring reliable operation of a real-time workload also depends on an awareness of the memory consumption of all workload tasks.
3. **Encapsulation of real-time attributes in reusable object-oriented real-time software components**: Modern real-time systems are constructed out of the integration of many independently developed software components. Over the lifetime of a real-time software component, it will most likely be deployed in many different contexts, on a variety of different processors and real-time operating systems. To facilitate easy porting and integration of a real-time software component, bundle each component with the ability to analyze and report its CPU time and memory requirements. Note that a component's resource requirements may be different within each execution environment.
4. **Account for infrastructure activities in analysis of real-time workload**: The complete system workload typically includes certain infrastructure services that are acting on behalf of the application. Typical infrastructure services include implementations of solid state and/or magnetic disk drivers, network libraries, messaging services, and, when appropriate, real-time garbage collection of dead memory.
5. **Comply with regulatory agency guidelines**: Certain embedded real-time applications must satisfy regulatory auditors prior to commercial deployment. This is especially true of safety critical applications, such as commercial aircraft avionics systems and rail transport automation. Discuss software process requirements, accumulation of artifacts to document and support audits of software process, peer review of intermediate artifacts, legal accountability, and testing methodologies.
6. **Distinguish hard real-time from soft real-time**: Help students understand that hard real-time systems are never allowed to miss deadlines, whereas soft real-time systems are designed to be robust in the face of occasional missed deadlines. Construction of hard real-time systems depends on formal analysis of worst-case resource needs. Construction of soft real-time systems usually relies on empirical measurements and statistical analysis to quantify likelihoods of compliance with real-time constraints. The domain of hard real-time techniques is usually small and simple repetitive algorithms. The domain of soft real-time methodologies often includes workloads that are highly dynamic and may include complex algorithms and heuristic approximations.
7. **Object-oriented techniques for hard real-time development**: Traditional object-oriented programming depends on practices that are difficult to analyze to the levels of rigor generally required in hard real-time systems. Discuss techniques to reduce and discipline polymorphic method invocations. Explore technologies that replace dependency on automatic garbage collection with safe stack allocation.

4 Suggested Curriculum Implementation

The recommended course structure relies on the use of real-time Java as the preferred foundation for course programming assignments. The reason is simple. Using Java for programming maximizes the learning experience while minimizing time spent in less productive peripheral pursuits. Furthermore, the use of Java offers improved coordination with other classes in the typical core curriculum of many computer science and software engineering programs.

Several anecdotes from industry experience help emphasize these benefits.

1. Calix, an industry leading vendor of broadband loop carriers to bring DSL connections into homes and small offices, chose to replace their C implementation of the management plane software with a Java implementation. They found that they were able to write the Java version of this software in half the time required to implement the C version. Furthermore, they found that the Java version had fewer bugs and more features. This represents a two fold improvement in developer productivity.
2. Intel found that they were able to construct a fault tolerant demonstration of new hardware by integrating off-the-shelf Java software components in only three days of effort. They reported that previous efforts to do build similar demonstrations out of off-the-shelf C and C++ components required three sold months of developmenet effort. This represents a twenty fold improvement in developer productivity.
3. Lockheed Martin reports that they were able to verify 3,500 requirements for a portion of the Java implementation of the Aegis warship software in only 5 months. With the previous version of the Aegis software, which had been written in Ada and CMS-2 (proprietary military language), their experience had been to verify only 3-4 requirements per day. This represents a nine fold improvement.
4. After rewriting the Aegis weapons system software in Java, Lockheed Martin found they were able to add support for “Standard Missile 6” in only 3 months of additional effort. Before the Java rewrite, they report that this activity would have required a full year of development.

These productivity benefits result because Java offers higher levels of abstraction, improved separation of concerns through better encapsulation of software components, and superior portability. This has made it easier to integrate 3rd party library components from external sources, easier to reuse software components originally developed and tested on different platforms, and easier to test and verify software functionality on fast, large-memory server machines instead of on the slow, memory constrained devices that are typical of embedded deployments.

In a university instructional setting, it means a class on principles of embedded real-time software development does not need to set aside and staff a special instructional laboratory for embedded real-time computing. Real-time Java technologies run on mainstream platforms like Linux and Windows as well. The jitter (unpredictable timing due to interference from other tasks) will of course be higher on a multi-user platform that is not designed to support real-time operation. Take this into consideration in establishing expectations for real-time determinsm.

For an enrichment activity, it is always interesting to see the Java code developed and tested on a shared server downloaded and executed on a dedicated single board computer. There are many such boards available now which are directly supported by off-the-shelf configurations of Linux and real-time Java virtual machines. Such boards are available today for under $150 [14], making it much easier today than in previous decades to experiment with true embedded systems development.

Realize, however, that including embedded development and integration activities significantly reduces the time available in a one-semester course to study topics in real-time application development. To do both embedded and real-time topics justice, it is probably best to offer two courses, one focused on real-time application development, and the other focused on embedded device development. Exercises in hard real-time development and device driver implementation probably fit best within the embedded course.

5 A Call to Action

This paper presents overview recommendations for changes in the embedded real-time curriculum. Developing these ideas into an off-the-shelf class requires significant effort. One of the reasons that embedded real-time development is not typically taught, or if taught, promotes outdated technologies and approaches, is because existing faculty are not generally familiar with the special needs of these industries. To make this course a reality at multiple universities, the following materials need to be developed:

1. A textbook needs to be available to cover the theory and principles of these topics. Currently, there does not exist a single textbook that covers all of the relevant topics. The textbook needs exercises and solutions to reinforce the learning planned for each topic.

2. A set of programming assignments needs to be developed along with sample solutions to each assignment. Supporting infrastructure software must be developed and provided to students so that they can focus their attention on the intended learning experience of each assignment without spending large amounts of time establishing the foundations on which their learning experience is to be based. A breakdown of principles to be reinforced and emphasized by each assignment should be provided. Programming assignments must align with classroom learning objectives.
3. A turnkey programming environment must be provided to make it easy for instructors to manage homework assignments. It must be easy to install and maintain this programming environment on readily available instructional computing platforms without adding significantly to the instructional lab staff's support burdens.

While the author of this paper is willing to contribute to creation of course materials, the nature of his responsibilities in supporting commercial customers does not leave time to effectively prepare the classroom materials by himself. A collaborative effort between industry practitioners and professional educators would be ideal. An expression of interest in this curriculum development from multiple educational institutions would help multiple individuals to justify their contributions to the overall effort.

6 Summary

The typical university treatment of embedded real-time development represents market requirements that are decades old. More modern real-time and embedded application development requires an understanding of principles and technologies that are not addressed in typical undergraduate or graduate programs in computer science and engineering. Suggested improvements to the embedded real-time curriculum would allow coursework to correlate better with current industry needs.

7 References

[1] K. Arnold, J. Gosling, D. Holmes. *The Java™ Programming Language, 4th edition*. 928 pages. Prentice Hall PTR. Aug, 2005.

[2] Jeff Friesen. Learn Java for Android Development. 656 pages. Apress. Sep. 29, 2010.

[3] "Calix Success Story", Atego white paper. 2 pages. May 2003. http://www.atego.com/pdf/PERC_CalixSuccess_e.pdf

[4] "Nortel Success Story", Atego white paper. 3 pages. Oct. 2003. http://www.atego.com/pdf/PERC_NortelSuccess_e.pdf

[5] "National Oilwell Varco Selects Aonix PERC for Java-based Robotic Drilling", RigZone. Sept. 25, 2006. available at https://www.rigzone.com/news/article.asp?a_id=36470

[6] Andrew Winkler. "The Modernization of the Aegis Fleet with Open Architecture". Keynote talk, Java Technologies Real-Time and Embedded Systems. York, UK. Sept. 28, 2011.

[7] Atego Press Release: "Thales Air Systems Selects Aonix PERC Ultra for Execution on Ground Radar Systems", Mar. 03, 2009.

[8] K. Nilsen. "Applying COTS Java Benefits to Mission-Critical Real-Time Software", Crosstalk: The Journal of Defense Software Engineering. pp. 19-24. June 2007.

[9] R. Lutz. "Software Engineering for Space Exploration", IEEE Computer, pp. 41-46. vol. 44, no. 10 (Oct. 2011).

[10] R. McMillan, "GM CTO Sees More Code on Future Cars", Computer World, Oct. 20, 2004.

[11] E. Roberts. "Resources to Support the Use of Java in Introductory Computer Science", ACM SIGCSE'04, Norfolk, Virginia, USA. March 3-7, 2004.

[12] E. Roberts, K. Bruce, R. Cutler, J. Cross, S. Grissom, K. Klee, S. Rodger, F. Trees, I. Utting, F. Yellin. "The ACM Java Task Force Project Rationale". 118 pages. Aug. 2006.

[13] *Computer Science Curriculum 2008: An Interim Revision of CS 2001*, Dec. 2008. Endorsed by ACM Education Council, Nov. 2008. Endorsed by IEEE Computer Society Educational Activities Board, Nov. 16, 2008.

[14] Mitch Frazier, "The BeagleBoard: $149 Linux System", The Linux Journal, Aug 6, 2008. http://www.linuxjournal.com/content/beagleboard-149-linux-system

Teaching Design and Testing in Computer Science Curriculum

M. Barjaktarovic
Department of Mathematics and Computer Science, Hawai'i Pacific University, Honolulu, Hawai'i, USA

Abstract - *Modern code consists of millions of lines of code. It is very difficult to design and test such code, especially if accompanied by a relatively high turnover of personnel and a lack of proper documentation - problems that are common in real life. How can we introduce enough and appropriate mathematics into the computer science curriculum so that students become excellent programmers? This learning needs to be conveyed to mathematics faculty as well so that they can help train future computer-related professionals. This paper presents the results of our experiences teaching coding using mathematics to diverse audiences, from non-major freshmen to more senior computer science majors.*

Keywords: design, testing, programming, curriculum, teaching

1 Introduction

Historically, computer science departments started within engineering departments, and eventually formed their own. The curriculum was modified to reflect the particular needs of computer science. While the engineering curriculum heavily depends on mathematics, the computer science curriculum was unfortunately left with a shortage of mathematics, according to many sources such as ACM Fellow David Lorge Parnas [5]. What type of mathematics is needed?

Modern code consists of millions of lines of code and requires implementation by multiple teams of programmers. Projects become more and more complex as software gains more features and more functionality. Additional pressures stem from the tight deadlines due to market pressures, particularly since we now compete in a global market. Architecting such non-trivial and complex code, implementing it, synchronizing the pieces, and keeping track of code versions and improvements, especially on a tight schedule, is a daunting task. Mathematics, properly used, provides quantitative help in design, implementation and testing. Engineers routinely deal with non-trivial problems, thanks to their rigorous education in mathematics. Computer science, to this day, wrestles with the quantitative approach to coding, for example handling the mythical man-month [4].

Object oriented (OO) approach provided help, largely by enforcing coding in a modular way and thus providing a well-defined structure. Currently, there is a multitude of OO programming languages. However, just coding in one of these OO languages is not enough to actually approach the entire software engineering process efficiently. Approaches such as model driven software engineering [10], unit testing [6], and test driven development (TDD) [3] have recently been devised to fill the gap. TDD was coined in 2002 and is currently popular with agile development teams [8].

Perhaps we can get some pointers from engineering, which has always used the OO approach and called it the "black box" approach. Areas such as control systems, communications, and many others depend on this approach. It is quite simple yet extremely powerful – each "box" takes an input, processes it, and produces an output. In the case of engineering, the processing can even be described mathematically via its transfer function. Computer science has acknowledged this kind of thinking in many ways, such as through the IPO approach (Input, Processing, Output) [14] and the "black box" and "white box" approaches [13]. However, these approaches are seldom taught in beginning and intermediate programming courses at most universities, while engineering students learn them from the start.

2 Mathematics for Computer Science

After talking with many higher-education faculty in mathematics and computer science and mentioning that computer science students need more mathematics, we find that most people have the initial response to think of "mathematics" as calculus, algebra, geometry, and other more "computational" and traditional facets of mathematics. Engineers use such mathematics. When we mention "discrete mathematics," many mathematics faculty tend to think of proofs and logic, and computer science faculty tend to think similarly. We would like to think of algorithms first.

In our experience, the mathematics necessary for efficient coding is the *algorithmic way of thinking*: algorithms, pseudo code, steps to take, levels of abstraction, and "chunks" of problems. This approach is the foundation of all mathematics; we can summarize it as "problem solving in steps." Advanced coding may use the "computational" aspects of mathematics, for example in 3-D animations, simulations, and engineering applications. In this paper we address the very basic foundation of coding.

In our experience we have seen that if students approach design and testing as a formalized, structured process, they learn very quickly. We ask students to design from one level of abstraction and proceed to refine into more detailed levels, and to define the results and/or behavior they expect at certain points in their code. Their coding skills improve greatly; their coding is faster and less demanding. This certainly helps with retention as well as their confidence and objective level of skill, which is then appreciated by their future instructors and employers.

Mathematics faculty are typically not aware of the needs of computer science students. In our opinion, it would greatly help if different disciplines communicate and coordinate the training. We envision curriculum charts as flow charts, where each class is represented as a box where it is clearly delineated what skills students should have when they enter and leave each class.

Given the existing conditions, we approach computer science training by (re)teaching the basic algorithmic way of thinking. This paper presents the result of our experience teaching coding to diverse audiences, from freshmen out-of-major to more senior majors. The language we use in this paper is Java, but any other language can be substituted.

2.1 The Big Picture

India, with its Institutes of Technology, and many other countries have proven that success in today's world depends on education and that science, engineering and technology fields (STEM) are critical. In the U.S.A., the situation is not ideal. Employers still complain that they cannot find enough talent, and the government is alarmed by the lack of students in STEM fields [7],[11],[12]. Agencies such as DARPA and many others [9] provide grants to increase the number and quality of our students. The US House of Representatives proclaimed December 5-10 as the National Computer Science Education Week. American students sometimes question the validity of their college degree and mention that many successful computer professionals do not even have college degrees. In other words, the common theme is that our higher-level education is not working as well as we would like it to. This is a huge problem that requires cooperation on many different levels. In this paper, we address only what can be done in computer science programming classes at the undergraduate level to increase the quality and quantity of the graduates.

Engineers routinely tackle non-trivial problems, thanks to their rigorous training in mathematics. The software development process could use mathematics more. Mathematics, used properly, provides a quantitative framework, the structure that enables efficient processing [1],[2].

3 Modular Approach in Practice

Through our experience in industry and academia, where we witnessed many "star" programmers as well as not so coveted ones, it became obvious that a good programmer programs in the OO style, regardless of the language used. OO makes coding and debugging easier. In OO, everything is chunked out into "objects" aka "boxes," making it is easier to follow the flow of the code design, to see what comes in and what comes out of each box, and understand how boxes connect to each other. It makes design and debugging much easier.

Informally, we use the modular approach in daily life because it is the most efficient way to proceed: "If I know what should come out of the box, then I can easily test it to see if it works correctly. If it does, then I can leave it alone. If not, then I can get into the box to debug it inside." This is the secret behind every successful car mechanic, medical doctor, and many other professionals.

We can describe the process as "thinking like a car mechanic." A good mechanic knows all the car parts, how they interrelate, what they do, and how they behave if they are working correctly or not. A good mechanic can debug very efficiently. If there is car noise, is it the muffler? The mechanic checks the muffler – it is ok - so she disregards the muffler. Is it the clutch? Yes. So she completely focuses on the clutch. Perhaps she can repair just some parts of it, or replace the whole clutch.

In computer science, we do not deal with chunks such as mufflers and clutches but with chunks of code. It is very important to know what the chunks are, what order they are in, and how to test them. We emphasize to the student that these are the basics of the modular approach, and, although it looks like simple common sense, it is an extremely powerful and practical paradigm. We explain to the students why.

"Chunks" allow interoperability and reusability. In computer science, using someone else's code is not called copying or cheating; it is called object oriented design and is highly encouraged. The idea is to make "libraries" that can be reused, which saves time and money. Just like with cars – it is possible to buy tires from many different vendors and have them all work.

Chunks can be on any level of abstraction, meaning a big chunk will have smaller chunks inside. For example, an engine has many parts inside. Chunks allow better design – what are the major portions of the solution? Each is a "chunk." Also, chunks allow for better testing – we can test before and after a chunk.

Chunks have three parts: the ingredients, the action to be taken and the finished product. In computer science lingo it is sometimes referred as input-processing-output (IPO) and sometimes as data and control flow. Data flow refers to input and output, and control flow is the processing part.

If we visualize it, it is like putting something into a machine that will do something to it, maybe transform it, and maybe generate something new. Just like in math: feed a few numbers to a calculator and let it calculate all kinds of things. Data flow is the numbers on the screen, and control flow is dictated by the calculator buttons you press. Below are some problems from real life that illustrate that.

Example: car. If it runs well, then there is no need to start looking under the hood. However, if it does not run and/or produces suspicious noises, we will have to open the hood and start troubleshooting. Is it the front, the back, the engine, the transmission, or the exhaust? Once we isolate the part that is causing problems, we further troubleshoot it, and eventually maybe we can just pull the malfunctioning part out and replace it. If code is written well, then it is structured exactly like that. Parts of well-done code are "plug and play." This is the OO principle.

Example: cooking dinner. If it turns out all right, we do not ask any questions and just enjoy a nice meal. However, if it turns out badly, then the victims of bad food start asking questions about details: did you put too much salt? Did you cook it long enough? Did you wash the food and refrigerate it properly?

In case of dinner, very often it is not possible to "plug and play" because everything is inter-related, e.g. if the spaghetti is cooked well but is smeared with sauce that is too salty, the whole plate of spaghetti might not be edible. We might have to start cooking from scratch. When code is written like that, without clear boundaries between parts, it is called spaghetti code**.** We emphasize to the students that spaghetti code is to be avoided because it is messy, slow, unreliable, most likely inaccurate, and definitely not scalable. Typically, it is not possible to repair spaghetti code; we have to write new code from scratch or continue to add more spaghetti mess to the spaghetti mess, which is a waste of time and effort.

Programmers are relatively well paid per hour; therefore the goal is to have code done as fast and also as accurately as possible in order to avoid later calls by the customer to fix the code. An additional concern is to have the code clearly structured and documented so that it is easy to modify, upgrade, scale up, and reuse it.

4 Specifics of Training

In order to teach these principles and thus mimic the process used by highly successful programmers, we ask students to submit the following with each homework assignment:

1. Code
2. Test cases: on-paper design, a specific plan, a checklist, of how you plan to test your code.
3. Pencil-and-paper results: what you expect as output from your code.
4. Test results: actual output of your code with your test cases as input.
5. Commentary

Test cases are used to prove that the code works correctly and is immune to wrong inputs as much as possible. Writing test cases is not about writing an essay on how you approached testing. Test cases are a specific checklist of all specific tests you plan to run. Pencil-and-paper calculations and test results are based on the test cases you designed; i.e. they are the same tests, except that pencil-and-paper is what you expect, and test results are what you actually got from your code.

Pencil-and-paper calculations are by-hand calculations of the test cases designed above. In other words: the tests cases you designed will produce some results. What should the results be? We need to figure it out on paper first.

Test results show that your code works in practice. It is enough to just cut and paste output from your code.

Commentary is necessary for documentation and also as a reflective practice. Writing commentary enhances learning and helps students integrate information that they obtained during coding. Commentary should include any problems encountered, how they were solved, and any insights. The very basic question to be commented on is: did you get the results you expected? Why or why not?

In our experience, students understand the coding portion of the software development process but may have difficulty with understanding the testing, especially if their textbook does not mention testing. Unfortunately, there are many popular undergraduate textbooks that contain neither testing nor design, and sometimes the instructor is forced into using such textbooks. The remedy is to write our own instructions and clearly explain to the students why additional material is necessary. However, the situation can still be awkward since students would prefer to believe that the textbook is excellent and has all of the needed portions.

4.1 Examples

Finding examples of possible illustrations is rather difficult. None of the examples we mentioned (car mechanic, cooking dinner) so far have been "interesting" and "catchy" to students. Finally, after many different trials, we found that the car wash example elicits the most enthusiastic response. Most American students, regardless of their background, seem to be quite concerned with the looks of their car and thus very passionate and well informed about the steps necessary to obtain a good outcome.

4.1.1 Example: testing if car washing service did your car properly

Below is the outline of how we explain the testing process to the students, using a car wash example.

Test cases are your plan for inspecting your finished car, i.e. a list of tests, i.e. a checklist of specific things you will do to test their work. The car wash shop also has a similar checklist. Their technicians should go through this checklist when they clean your car. You will follow your checklist when you get your car back from the service. You know what you expect from each test. If a test does not pass, you might refuse to pay for the service, complain to the management, and/or ask them to redo the job.

Pencil-and-paper results are the results you expect in theory, i.e. what you already know should be the results of your test cases. For example, a washed car should be spotless, without any dirt or stains, and with a clean interior and exterior. (In this particular case, we do not have to "do" any pencil-and-paper calculations since you probably already know how a car wash works.)

Test results are actual results. You will physically inspect your car when it is finished.

Commentary is what you would write on your feedback card to the shop, or on a site such as yelp.com. Was your car done properly? Did they miss anything? Did they do an exceptional job on something? Was the service good? Did you have to wait too long? Would you go back there? What have you learned? For example: empty the car seats before car wash, otherwise they will not wash the seats; do not give car to Nate but to Matt since he pays the most attention to detail; do not use car wax brand XYZ but use brand ABC.

For example, you might come up with the these test cases, expected results, and actual results, shown in Figure 1.

Checklist: (Test cases)	Expected output: (Pencil-n-paper calculations)	Actual results (Test runs)
Car body	shiny and clean	rain stains on doors
Windows	clean	windshield dirty
Under car	washed	muddy
Tires	shiny and clean	yes
Interior:	clean	yes
	no trash, dirt, etc.	dirt in seat folds some cigarette ash
	no stains	stain on front seat
	smells good	cigarette smell
Floor mats	clean	yes
Under mats	clean	yes
Under seats	clean	sand under front seat
Pockets	coins still there	no

Figure 1 – Sample car wash testing

We can add as many tests as we want, depending on how meticulously clean, aka "correctly done," we want the car to be. Software testing follows the same principle, e.g. a nuclear reactor control is tested (much) more than a toy video game.

4.1.2 Example of a full homework submission: sin(x)

This example was given to students as an example of a properly done homework submission.

Task: write code to ask user for a value x and then calculate sin(x).

Code:
Make a class that calculates sin(x), and a main program (aka test stub) that uses all methods from the class, to see if the class works properly. The test stub will be based on your test cases. In this particular example, the test stub will have a loop that goes through all angles.

Test cases:
Easy test cases: angles of 0, 90, 180, 360 degrees.
Wrong test cases: just a few is enough, e.g. enter character values as angles.
Complete test cases: sin(x) values for angles from -360 to 720 degrees, in increments of 10^o.

Pencil-and-paper results:
Do some (ideally all) test cases by hand. For this particular problem, do just a few test cases, not every single one. Check the easy cases and perhaps 1-2 more "more computationally intensive" cases that you have to use a calculator for.

Test results:
Run your code and copy and paste the results.

Commentary:
Does your code work as you thought it would? In Java, why is the result for sin(0) not 0? What have you learned?

4.1.3 Example of test cases: bank

This example was given to students as an example of testing and also design driven by testing.

Task: write code to simulate a bank account. If the balance is below $20, the account is considered inactive. Customers are allowed only 5 free withdrawals. After that, each withdrawal costs $1. At the end of each month, we calculate the interest.

Assume that we have four modules: OpenAccount, Withdraw, Deposit, and MonthlyProcess.

At this point, we do not necessarily introduce the terms "precondition" and "postcondition" officially. Depending on the level of our class, we might choose to talk about "what holds true at the input" and "what holds true at the output" of each module and thus is used in testing. We also introduce the concept of specifying what needs to be accomplished by the module, i.e. what the module "body" should do.

We also introduce the concept of using constants instead of hardwiring variable values, in order to help code readability.

```
MIN_BAL = 20              //minimum balance
NUM_FREE_WITHD = 5  //number of free withdrawals
WITHD_CHARGE = 1      //charge per extra withdrawal
INT_RATE = 0.05           //interest rate

OpenAccount module:
  Precondition:
    if deposit < 0, cannot open account

Withdraw module:
  Precondions:
    if withdrawal < 0, cannot withdraw
  Postcondion:
    if balance < MIN_BAL, status = inactive
  Body:
    balance = balance – withdrawal
    num_withdrawals = num_withdrawals + 1

Deposit Module:
  Precondion:
    if deposit < 0, cannot deposit
  Postcondion:
    if balance >= MIN_BAL, status = active
  Body:
    balance = balance + deposit

MonthlyProcess:
  Precondition:
    charge = 0; interest =0
  Postcondition:
    correctly calculated new balance; reset charges
  Body:
    charge = (num_withdrawals–NUM_FREE_WITHD)
                * WITHD_CHARGE
    balance = balance – charge
    interest = balance * INT_RATE
    balance = balance + interest
    charge = 0; interest =0; //reset the values
```

5 Thinking in Modules: Examples

This section presents examples from daily life that students can relate to. The examples are worked-out in OO style, using the box approach from engineering.

5.1 Prepare a spaghetti dinner for a group of 5 people

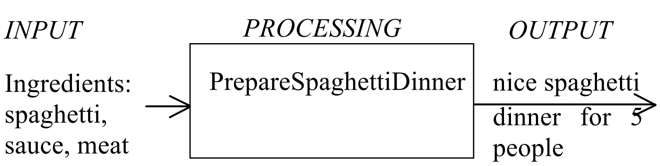


Figure 2 - High-level view of cooking spaghetti dinner

At the highest level, this is easy to test: we taste the dinner. Is it spaghetti? Does it taste good? Is there enough for 5 people? If so, we can eat. However, if there are any problems, e.g. we get dinner for 3, or the dinner tastes too salty, too bland, too cooked, then we start troubleshooting inside the PrepareSpaghettiDinner box.

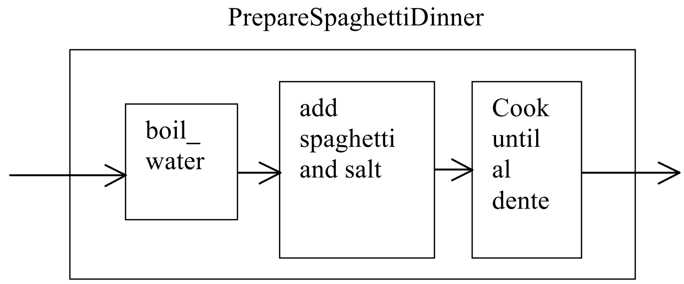


Figure 3 - Inside one module

Each one of these modules, aka boxes or chunks, is easily testable on its own. For example, “cook until al dente” box must produce al dente spaghetti. If not, then we exceeded cooking time or used too much heat. We look inside the box to find out exactly what step went wrong. The boil_water box must produce boiling water, not lukewarm water. We can identify where the problem is by looking at each box individually.

Students need to be reminded that each box must be given proper input in order to work. For example, the boil_water box requires a dish with water in it and some kind of heating device. If you try to input something else into the box, e.g. gasoline, the box cannot work. When given wrong input, some boxes will work but incorrectly, and some will crash.

Each one of the boxes has smaller chunks inside. For example, let us look into box called “boil water.”

boil_water box takes as input a warming device, an empty clean dish, and water.

boil_water box produces boiling water.

What is inside boil_water box? Anything we want, and that is the beauty of modular design. It can be boiled using microwave, stove, open fire, grill, oven, and many other devices. As long as it produces boiling water to pour over spaghetti, it works.

This is a great place to introduce big-O and other notations for measuring algorithm effectiveness. In computer science, one common box is called “sort numbers in increasing order.” By the second class in the major, our computer science students already know about five different algorithms to sort, and there are many more. They all “boil water” equally, however, some are boiling “better” than others. For example, boiling water in an oven is not all that efficient.

5.2 Organize a wedding.

The event cannot start until there are two people to wed. At the end, they have to be legally married. Thus we introduce the concept of precondition and postcondition.

The marriage box takes in: two people, guests, legal marriage representative.
The marriage box produces: two married people.
Inside the box, there are chunks such as:

send invitations
schedule wedding cake
get the gowns and the tuxes
pick the maids/men
book the venue
....

Each one of the higher-level chunks consists of lower-level chunks. For example:

```
send_invitations
   decide who to invite
   get their addresses
   pick the modality
      pick a design
      pick what to say
   send it
```

Each line is like a Lego piece. Notice how each piece leaves room for creativity. “Decide who to invite” can be implemented in many different ways – it can be done in person in your living room; over phone, over email, on Facebook, and so on. “Get their addresses” can be as easy as looking into email address book or a Rolodex, or can involve lots of calling. “Pick a design” can use paper or e-designs. An old fashioned wedding will go to a printer and send paper cards. A more modern wedding will do everything over email or social networking sites. In any case, they all pick a design and send invitations albeit in their own way.

We bring the attention of the students to to the indentations. Indentations mark a “chunk inside chunk.” Thus we introduce the concept of different levels of abstraction.

It is important to notice that the chunks do go in order. For example, it is not a good idea to send invitations until the venue is booked. Below is an algorithm that will fail:

glasses used for dinner will be 3 inches tall
send invitations
schedule wedding cake
get the gowns and the tuxes
pick the maids/men
book the venue

In this scenario, we are sending invitations before we know where and when the wedding is; we make gowns before we know who the maids are. In addition, we are concerned with small details such as what glasses will be used during dinner before we have worked out the important matters. Below is a better algorithm:

book the venue
pick the maids/men
send invitations
schedule wedding cake
get the gowns and the tuxes

In computer science, it is very important to read the code in English, just like the wedding example, and spot the flows before we actually run the code. If we try to run the code in a “guessing way” just to see what will happen and hope we stumble upon the right solution, coding takes a very long time and is quite a mysterious process that wastes a lot of time and money. Just like in the wedding example – we do not go by trial and error and try to send invitations to see what will happen. Everything is done intentionally and is planned.

5.2.1 Graduate someone from UniversityXYZ

On the highest level, the problem looks like shown in Figure 4:

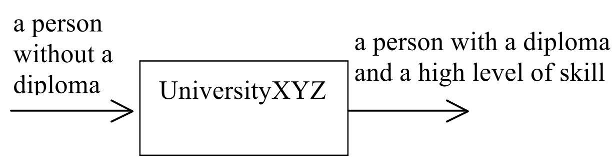


Figure 4 - High-level view of graduation process

On the highest level, we just test the person graduating – if they have the diploma and the skills, we are done. There is no need to look inside the box called UniversityXYZ. Otherwise, we go into the box and look into details. For example:

Did the University provide adequate educational opportunities?
Did the student study enough?
If necessary, we check in more detail: did they not study because they worked too much? played too much? did not study properly? no prerequisites? picked wrong major? lacked motivation? had family to take care of?

As long as we have not identified the problem, we ask more and more detailed questions.

6 Conclusion

Approaching software engineering training with a degree of formality and using algorithmic thinking and the modular approach helps students to learn faster and code more efficiently. Students themselves like the end result and positively comment on it via their evaluations and personal comments. Thus, this approach is helpful for retention and benefits employers and higher level classes.

7 References

[1] Milica Barjaktarovic. “Teaching Mathematics and Programming Foundations Early in Curriculum Using Real-Life Multicultural Examples.” The 2012 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'12), July 2012.

[2] Milica Barjaktarovic. “Training Effective Developers.” The 2012 International Conference on Software Engineering Practice (SERP'12), July 2012.

[3] Kent Beck. “Test Driven Development (TDD).” Addison-Wesley Professional; 1st edition, 2002.

[4] Fred Brooks. “The Mythical Man-Month: Essays on Software Engineering.” Addison-Wesley Publishing Company, 1975. Anniversary addition reprinted 1995.

[5] Nancy Eickelmann. “Interview to profile ACM Fellow David Lorge Parnas for SIGSOFT SEN”; SEN Vol. 24 Issue No. 3, May 1999. http://www.sigsoft.org/SEN/parnas.html

[6] IEEE Standards Board. “IEEE Standard for Software Unit Testing: ANSI/IEEE Std 1008-1987” in “IEEE Standards: Software Engineering,” (the Software Engineering Technical Committee of the IEEE Computer Society), Volume 2 Process Standards, 1999.

[7] Paul Jarley, Dean of Lee Business School. “Where are all IT Students.” Posted on February 20, 2012. http://business.unlv.edu/dean/where-are-all-the-it-students/

[8] Eric Landes. “Baking Test Driven development in Agile Software Development.” http://www.devx.com/enterprise/Article/46478, 2011

[9] Office of Sponsored Research. Sources of STEM Grants. Updated December 2011. http://www.csufresno.edu/grants/proposal/resources/stem.shtml

[10] Douglas C. Schmidt. “Model Driven Engineering.” IEEE Computer, Vol. 39, Issue 2, 25-31, Feb 2006.

[11] STEM Education Coalition home page. http://www.stemedcoalition.org/

[12] Mike Subelsky, Open Society Institute Baltimore. “Baltimore Should Become a Software Leader.” http://www.audaciousideas.org/2011/08/baltimore-should-become-a-software-education-leader/ Posted on August 2, 2011.

[13] William C. Wake. “Growing Frameworks in Java” chapter 12 “White Box and Black Box”. http://xp123.com/wwake/fw/ , 1994-2010.

[14] Niklaus Wirth. “Algorithms + Data Structures = Programs.” Prentice-Hall, 1976.

Using Real Execution Timings to Enliven a Data Structures Course

B. Boothe
Computer Science Department, University of Southern Maine, Portland, ME, USA

Abstract - *Which is faster, an array list or a linked list? In our data structures course we increase the engagement of our students by asking them to predict the outcome of a future race between competing data structures. We record their predictions for future validation, and subsequently as we analyze the data structures, students are more engaged in the analysis because it either lends credence or doubt towards the accuracy of their predictions. Some of their expectations get radically broken by the analysis, and in the end everyone gets surprised by the actual runtime results. Everyone benefits because the class is more engaging and the results are more memorable.*

Keywords: Real Execution Timings, Data Structures

1 The Starting Line

At the University of Southern Maine our initial programming curriculum is taught in Java. Our Data Structures course is the third course in our sequence of programming courses. In our second course, students use data structures such as Java's ArrayList and LinkedList and learn a bit about the internal mechanisms of how they work. They know that array lists have to periodically grow the internal array by creating a new one and copying everything over, and they know that it is painfully slow to insert at the start of an array list and move everything up. They have also been convinced linked lists are slick because of how efficiently they can insert new values between others without needing to copy anything. Accurate or not, these are their typical preconceived notions coming into data structures.

The first day of class I propose that we use the online ordered collection problem as a basis for analyzing and comparing the variety of data structures that we will encounter in this class. The online ordered collection problem is to take N items and insert them into the data structure while maintaining the data structure in sorted order at every insert. The online aspect refers to the possibility of using this data structure to perform a search at any point while it is being built. Since building the data structure involves searching for the positions to insert new members, this is a concise benchmark that incorporates both the cost of creating and growing the data structure as well as the cost of searching it.

After I have introduced the ordered collection benchmark, I then have students participate in a small group activity and make their predictions on which will be faster, array lists or linked lists. I pose this as a race between the best real programs we can write, running on a real computer. Which data structure will win?

I am not asking them to perform an order analysis, but simply to discuss it in their groups and use their intuition about these two data structures to predict the relative runtimes. Will one data structure be a little faster, a lot faster, or about the same as the other? The reader might pause to consider this question themselves.

I have used this activity for several years. The majority of groups always predict that linked lists will be either *a little faster* or *a lot faster* than array lists. For justifications they cite the high cost of inserting into the array and the cost of repeatedly growing the array to make it larger. In contrast, usually at least one group notes that the sorted array can be binary searched much faster than the linear search required of the linked list, and thus they predict that the array list will be faster. Finally, there is usually another group that remembers and utilizes order analysis to predict that both data structures will perform about the same. I do a lot of group activities and discussions in my class, and this disparity in opinion is perfect. All the better to grab their attention and later dispel their misconceptions.

With all of their predictions and justifications recorded on a transparency, I tuck this away to be brought out in a couple weeks after we have thoroughly studied and analyzed both of these data structures. Students are always disgruntled to learn that they will have to wait a couple weeks before they see the outcome of the race. They are eager to know the winner and whether their predictions are correct. Like a good writer, I know the value of foreshadowing.

2 Handicapping

Over the next couple weeks we discuss and analyze the implementation choices and asymptotic analyses of Java's ArrayList and LinkedList. The final analyses for building the ordered collection are shown in Tables 1 & 2. Overall both data structures have an expected execution time of $O(N^2)$.

Once they have seen that both the ArrayList and LinkedList will be $O(N^2)$, I give the students a chance to revise their opinions. Most students, having placed their bets on the LinkedList, stick with it beating the ArrayList, although maybe not by such a wide margin as some had predicted.

Table 1. Using an ArrayList to build an ordered collection.

building and growing	$O(N)$
searching	$O(N \log N)$
inserting	$O(N^2)$
total:	$O(N^2)$

Table 2. Using a LinkedList to build an ordered collection.

building	$O(N)$
searching	$O(N^2)$
inserting	$O(N)$
total:	$O(N^2)$

In addition to the traditional runtime analysis, I also do memory usage analysis of these data structures. For analyzing memory usage we simply count the array locations and/or visible object fields. These are summarized in Table 3. These memory footprints will play a role in explaining the real execution time results that the class will be seeing soon.

Table 3. Memory Footprints (as explained to students)

ArrayList	1N active, plus up to N/2 reserved for future growth
LinkedList	3N (data reference, previous node and next node references)

For memory usage they have seen that the object references stored in Java's array list are compactly stored in N adjacent memory locations, with up to an additional N/2 unused memory locations allocated as space for future growth.

For linked lists they have seen that each node in a doubly linked list has a data object reference as well as references to the next and previous nodes. This means that its memory footprint is 3N to hold N data objects.

At this point student attitudes are generally not overly interested in memory footprints since modern computers have enormous memories. In fact, in earlier classes I sometimes encourage students to be profligate with memory, if it can simplify their algorithms.

Over the duration of the semester we build a summary table of the data structures as we study them. For each new data structure, we note its key strengths and weaknesses, and we record its memory footprint and our analysis of its runtime on the online ordered collection benchmark (if appropriate). This table serves as a backbone for the class. As each new data structure is studied, we can quickly see its benefits and tradeoffs compared to the previously studied data structures.

3 The Race

Everyone takes pride in making a fast program, and thus students are always interested and eager to see the results from the race. The results are shown in Figure 1. This is a log-log plot of the execution times versus the problem sizes for 4 different data structures. At this point in the class, however, I show them a simpler version of this graph with only the performance curves for the ArrayList and the LinkedList. The lines for the ChunkList and TreeSet are added later as we analyze these data structures. I present only the single graph in this paper to save space.

These execution times are from a 2.4 GHz Intel Core 2 Duo with a 2MB L2 cache and 3.25 GB of RAM running Java 6. The testing code was written to be as efficient as possible. For array lists I used binary search to find the insertion point. For linked lists I used iterators. The objects inserted into the data structures are random Integer objects so as to minimize the amount of time spent comparing the objects and thus focusing on the time spent by the data structure manipulations themselves.

Students generally are quite startled to see the 60 times difference between the performances of these two data structures, especially since the majority had bet on the loser of the race! When I first created this activity I was myself surprised by the magnitude of the difference. I had expected the array list to win, but not by such a large margin.

4 Explanation

Why has our order analysis led us astray? Our expectation was that they were both $O(N^2)$, and thus they would perform similarly. Order analysis ignores constant factors, but a factor of 60 is startlingly large! It even appears from the increasing gap as the problem size grows, that measured in real time the LinkedList may in fact be of a higher order than the ArrayList.

I offer the class two explanations for the performance difference. I have not tried to quantify the importance of one factor over the other.

1. **A method call versus a memory access.** Focus on the $O(N^2)$ factors. For the linked list this was the searching time. Each insertion must linearly search down the list to find the insertion point. In total this

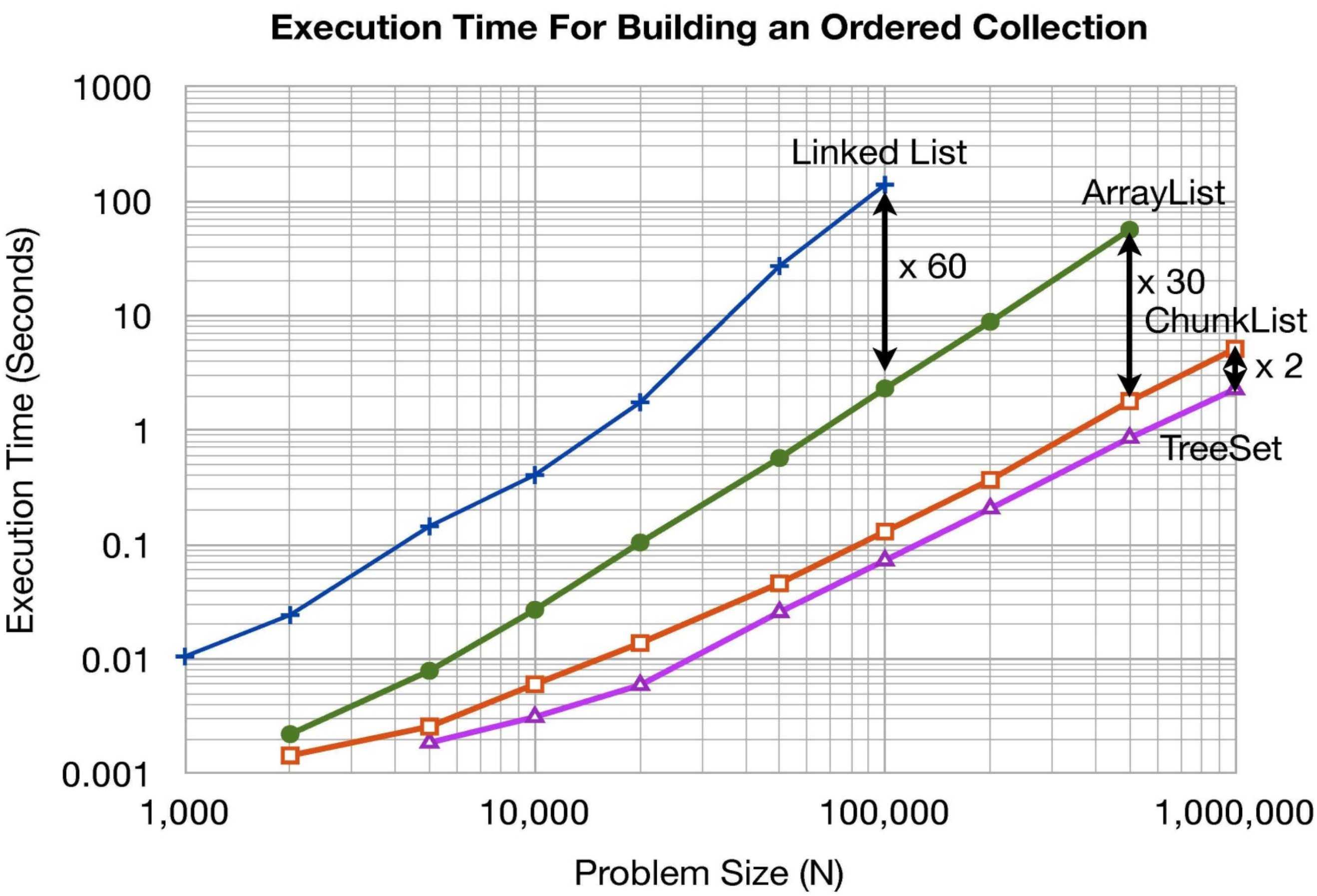


Figure 1. Execution times for building an ordered collection for 4 data structures.

involves $O(N^2)$ method calls to the comparator. In contrast, for the array list, the $O(N^2)$ factor was the insertion time incurred as each item was inserted somewhere into the array and everything in the array above the insertion point was moved up to make room for the new item. This involves N calls to System.arraycopy(), with each call moving $O(N)$ memory locations for a total of $O(N^2)$ memory locations moved. The difference is that arraycopy() is a tuned system routine very efficiently moving a contiguous block of memory [5]. When compared to the $O(N^2)$ individual comparator calls for the linked list, this is a big speed advantage for the array list.

2. **Memory Footprint.** In modern computers the time for a level 1 cache hit is more than 100 times faster than access to actual memory [1,2]. As the memory footprint of the data structure grows, more memory accesses are pushed to lower and slower levels of the memory hierarchy. In class we analyzed only the user visible fields and came up with an active memory footprint of 3 times larger for Java's LinkedList than its ArrayList. This means that the linked list is being pushed into lower levels of the memory hierarchy before the array list is. Furthermore the array copies are nice sequential accesses that work well with caches and pre-fetching compared to the linked list accesses which are jumping all over memory. We expect that this contributes to the increasing performance gap between the two data structures as the problem size gets larger. (At this point in class we do not complicate the picture by pointing out that the true memory footprint for Java's LinkedList is actually 7 times larger than that of an ArrayList due to hidden fields used in the representation of objects and the indirection table used for incremental garbage collection. [6])

Considering these two explanations, the factor of 60 difference in performance is more plausible, and those students who care strongly about fast execution times suddenly grant their old professor a little more respect when they realize he might be able to teach them more about the Zen of programming than they had expected.

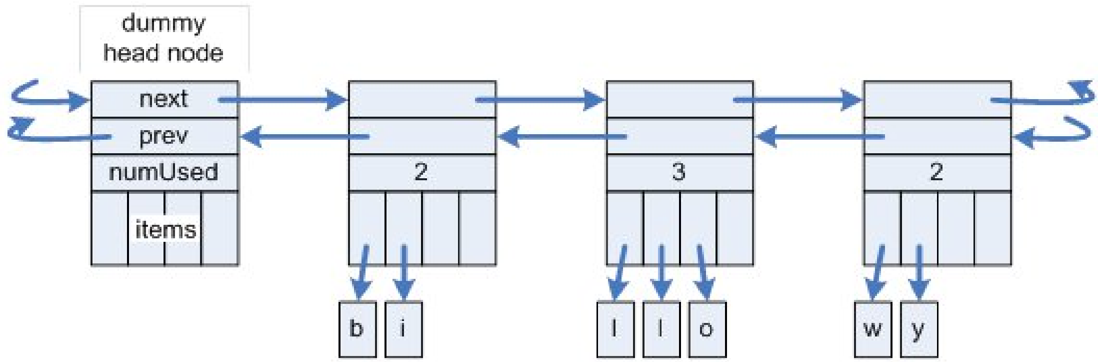


Figure 2. A ChunkLlist. A hybrid data structure that is a linked list of arrays. It is used to store an ordered list.

5 Hybrid Data Structures

Back in my days as a student, programming in C, we often needed to build a variety of simple data structures. These days programming languages provide excellent libraries of well-designed generic data structures. Data structures course have thus transformed into using these data structures and understanding their performance implications [4,7]. I, nevertheless, strongly feel that students ought to experience creating their own data structure from scratch. At some point in their careers they are likely to work on a complex software project that warrants creation of a new or custom data structure. I address this by having one assignment in which students implement a hybrid data structure, something incorporating aspects of two traditional data structures.

Several years ago I invented, for pedagogical purposes, a hybrid data structure that I called a ChunkList. A small example is shown in Figure 2. (I am not the first person to explore this hybrid data structure [3].) It is a linked list of nodes, but with each node containing an array of keys. When a node becomes full, it is split into 2 nodes. The purpose of this ChunkList is to implement an ordered list. Its add() method inserts a key in sorted order into the collection of keys. It is thus exactly what we need to implement the online ordered collection. Later students will see that its interface is in fact a subset of Java's TreeSet interface.

It is presented to students as a hybrid of a linked list and an array list having the best of both worlds. It has quicker searching than a linked list by jumping a chunk at a time through the list, but it also has quicker insertion than an array list by only inserting into an individual chunk, rather than inserting into a huge monolithic array. If chunks each hold 100 keys, then we have the possibility of searching 100 times faster than with a simple linked list. (All of that knowledge of linked lists has not gone to waste after all!)

One of my motivations for using a hybrid data structure is that having something more unique than what is found in textbooks decreases the chance that the less honest students will be able to find a pre-existing implementation to copy from. A quick Google search for ChunkList finds matches for the name, but I examined the top ranked matches and currently they provide nothing useful for the dishonest student who is trolling for a working program online. I would encourage anyone using this idea to think of a variation on the name. I have also seen this called an unrolled-linked list[3]. We might also call it a list of arrays or a blocked list.

Students are asked to implement this with a tunable chunk size and test it out with chunk sizes of 10 and 100. For extra credit students are asked to determine the optimal chunk size. I perform this analysis in class on the day they turn it in. It is an excellent opportunity to use a little calculus to minimize the execution time formula and demonstrate that all those math requirements that they grumble about weren't entirely immaterial to computer science. The analysis shows that the optimal chunk size is $\sqrt{N}$ and that the overall runtime for the online ordered collection problem using a ChunkList will then be $O(N\sqrt{N})$.

I now present to the class the earlier performance graph with the ChunkList added on. (In this paper the chunk list was already shown in Figure 1.) My ChunkList is a more sophisticated implementation than that of the students. It grows the chunk size in proportion to $\sqrt{N}$. As the total list size grows, so does the chunk size. From the performance graph this is clearly much better than the $O(N^2)$ of the ArrayList. The real execution time is 30 times faster than an array list, as the problem size reaches the maximum comparable size shown.

6 The Shoo-in

The eventual winner of course will be the binary search tree. Maintaining an ordered and quickly searchable collection is exactly what binary trees are intended for. Java's TreeSet is implemented with red black trees. It is the last line plotted in Figure 1. Its time for building the online ordered collection is $O(N \log N)$ which is asymptotically superior to the ChunkList's $O(N\sqrt{N})$.

The graph surprisingly shows only a factor of 2 advantage in real execution time of the TreeSet over the ChunkList. I was again surprised when I saw the actual performance data. At a size of N=1,000,000, the last data points shown on the graph, one might expect a performance difference of a factor of 50 for a $O(N \log N)$ algorithm compared to one that takes $O(N\sqrt{N})$. This is hard to explain. The contributing factors are:

1. **Red-black trees are complex.** Undoubtedly some of that factor of 50 is used up by the frequent internal rotations used in maintaining balance within the red black tree.

2. **ChunkList tuning.** The chunk list was tuned for optimal execution time. Through empirical tests I found that a chunk sizes of $13\sqrt{N}$ gave the fastest execution times. This larger chunk size gives more weighting to the faster array copy used in inserting into a chunk over the slower search of the linked list of chunks.

3. **TreeSet has a richer implementation.** For example the TreeSet provides both the Comparator and Comparable interfaces, whereas I asked the students only to implement the Comparator interface. Every option adds complexity and slows things down a tad.

4. **Java's TreeSet is just a wrapper for its TreeMap class.** In order to reuse code, Java's TreeSet is in fact a TreeMap with its data fields ignored. Every call to a TreeSet method is in reality an indirect call to a TreeMap method.

5. **Memory footprint.** Again the memory footprint size undoubtedly plays a role. The active memory footprint for the ChunkList is slightly over 1N, but the visible memory footprint for a TreeSet is 6N and the true footprint is 9N. A larger memory footprint pushes it into lower and slower levels in the memory hierarchy.

7 Conclusions

I have found using real execution timings and analysis of memory usage to be valuable tools in teaching data structures and for increasing student interest levels in the comparative analysis of these data structures. Students have great respect for real performance data and seem to trust it more than our mathematical analyses. Combining real execution times along with order analysis, and explaining the reasons for the discrepancies, has increased understanding of the value of asymptotic analysis and its limitations.

8 References

[1] Trishul M. Chilimbi, Mark D. Hill, and James R. Larus, "Cache-Conscious Structure Layout," In *Proceedings of the SIGPLAN'99 Conference on Programming Language Design and Implementation (PLDI),* May 1999.

[2] David A. Patterson and John L. Hennessy, "*Computer Organization and Design, The Hardware / Software Interface, 4th ed.",* Morgan Kaufmann 2009.

[3] "Unrolled Linked List", *http://end.wikipedia/wiki /Unrolld_linked_list*

[4] Elliot B. Koffman and Paul A. T. Wolfgang, "Data Structures, Abstraction and Design Using Java, 2nd ed.", Wiley 2010.

[5] Oracle Sun Developer Network, "JAVA SE 6 Performance White Paper", http://java.sun.com/performance /reference/whitepapers/6_performance.html

[6] Neil Coffey, "Memory usage in Java", http://www.javamex.com/tutorials/memory.

[7] R. Lister, I. Box, B. Morrison, J. Tenenberg, and D. S. Westbrook *"The Dimensions of Variation in the Teaching of Data Structures"*, In Proceedings of the 9th annual SIGCSE conference on Innovation and Technology in computer science education (pp. 92-96)

Developing and Teaching a Biometrics Course as Part of Computer Science and Engineering Curriculum

Weidong Liao, Osman Guzide, Reza Mirdamadi
Department of Computer Science, Mathematics and Engineering
School of Natural Sciences and Mathematics
Shepherd University
Shepherdstown, West Virginia 25443, USA

Abstract - *In this paper, we describe a biometrics course developed and taught at a four year college. The impact of the Biometrics course to STEM (Science, Technology, Engineering and Mathematics) fields, especially computer science and engineering, will be the focal point. Also included is the description of lab sessions developed as an integrated part of the biometrics course.*

The course development as discussed in this paper is sponsored by Research Enhancement Award from NASA WV Space Grant Consortium.

Keywords: Biometrics, Computer Science and Computer Engineering Education, STEM

1 Introduction

Historically, biometrics was a subject within biostatistics in history and it referred to the collection, storage and analysis of biological data regarding human beings. In the field of computer science and information technology, however, biometrics means hardware and software that can be used to measure and analyze characteristics of our body. Fingerprint has been the most commonly used characteristic. The other characteristics include Iris, facial, hand geometry, handwriting and voice pattern, etc.

For mathematicians, biometrics may be a subject in statistics. Biometrics applications take a sample, deriving a score, registering the derived score, deriving the new score when a user wants to get authenticated, and deciding if there is a match.

For engineers, biometrics means how to design a biometrics device. Before any biometrics application can be put into use, there must be certain types of biometrics devices, such as fingerprint scanners, facial recognition cameras, or voice recording devices. In addition, the companion software, including device drivers, SDK and applications, must be made available. It is therefore evident that biometrics is a comprehensive subject requiring solid background in STEM fields.

The Department of Computer Science, Mathematics and Engineering (CME) at Shepherd University have recently developed and a Biometrics and Security concentration. The CME Department offers four year degrees programs in the area of Computer Science, Computer Engineering, Information Technology, Mathematics, and Engineering. With great combination of academic background and industry experience, we are confident that we have sufficient skill set to develop a degree program in the field of Biometrics. We are confident that this program would be a popular concentration with great interest from students from the Eastern Panhandle of West Virginia and other neighboring states. Nevertheless, it has been challenging to develop and teach a Biometrics course to cover subjects ranging from biology, computer science, technology, engineering and mathematics.

The Biometrics course was developed after interviewing with and consulting our potential employers, students with interest in biometrics, and biometrics experts. By Fall 2011, this course has been offered three times. In this paper, we will describe our experience and lessons learned from developing and teaching this course.

The following sections are organized as follows. The next section describes the relationship between biometrics and STEM fields in general. Section 3 discusses the course description, the pre-requisite for this course and the textbooks that have been used when we were offering this course. After that, we discuss and describe two unique features for our biometrics classes: the term paper assignment and our lab settings to support this course and lab assignments we plan to have for student teams. Finally we summarize our ongoing efforts and provide observations we achieve from our biometrics classes.

2 Biometrics and STEM Fields

In this section we briefly describe the relationship between biometrics to each of these STEM fields: science, technology, engineering and mathematics.

2.1 Biometrics and Science

Biometrics has close relationship with biology, and thus background in biology will certainly be helpful for students to grasp biometrics concepts. In fact, the essence of biometrics is the translation from biological information to a series of numbers that can be stored in database. As a university with liberal art tradition, Shepherd University requires students take 46 General Studies credits. Among these 46 credits, 8 of them could be in biology.

We encourage students who are interested in Biometrics and Information Security degree program to take biology as part of their general studies credits. Meanwhile, biology students are welcome to take this course even if they haven’t taken the pre-requisite. We envision most students in CIS 361would have certain biology background.

2.2 Biometrics and Technology

Biometrics has been used in security for a long time. Since 2004, the US Homeland security and FBI have started to fingerprint visitors and immigrants in the international airports. European Union and Japan also want to follow. In the information security area, biometrics is commonly used to enforce authentication. That is, biometrics is widely used to confirm an entity as it claims to be. The biometrics can be used to confirm the message origin, to verify the integrity of the message, and to ascertain the receiver is intended receiver.

Information Security, Database Management System, Artificial Intelligence and Computer Graphics have been consistently offered at CME department, and the topics in these courses are also briefly described in Introduction to Computer and Information Sciences course (which is the pre-requisite for Biometrics course). Decent understanding to materials in these courses will certainly set the stage for the Biometrics course. On the other hand, Biometrics course will also be used as a course for students to reinforce their knowledge and concepts in Information Technology courses.

2.3 Biometrics and Engineering

Biometrics application relies on the success of physics, engineering and engineering technology. We can take fingerprinting (and related applications) as an example. At first, fingerprint scanners have to be envisioned and developed, which demands optics knowledge from physicists. Second, it requires engineers to have fingerprint scanners be manufactured. At last, engineers, engineering technologists, and IT professionals will work together to develop a full-featured biometrics system used in airport, banks, and Homeland security agencies.

We plan to encourage engineering students in our department to get involved in Biometrics course. Engineering students will be able to get chance to work with other students on Robotics projects using biometrics sensing systems, or other value-added biometrics system, such as biometrics based Sign-on/Sign-out systems that are used in human resource applications.

2.4 Biometrics and Mathematics

Biometrics requires certain background in probability and statistics. Background in sampling theory, regression analysis, nonparametric statistics and generalized linear models are essential to understanding the fundamentals of biometry and biometrics. Students in our industrial mathematics program would find Biometrics course beneficial. These students may also work on their capstone projects in biometrics area.

3 Course Description

CIS 361, Introduction to Biometrics, is a course targeting junior or senior students in Computer and Information Sciences (CIS) or Computer Information Technology (CIT) majors. This course is intended to teach fundamental concepts and theoretical background in biometrics and its applications. A lab session is also included in this 3-credit course to give students hands-on experiences in biometrics hardware and software.

This is the core course for students in the special track of Biometrics and Information Security. Students in other tracks in the CIS/CIT can take it as an elective course. This course is developed as an introductory course to cover the basics of biometrics and the mainstream biometric technologies. The underlying image processing concepts required to understand biometric techniques are also discussed. Other covered subjects are ethics, privacy concerts, and the future of biometric technologies.

In short, the following topics will be covered:

- Taxonomies of Biometrics Devices and Applications
- Basic Probability and Statistical Testing Methods
- Biometrics devices: finger printing, voice recognition, facial recognition, and iris scanning
- Biometrics and information security
- Social, Legal, and Ethical Concerns

Upon completion of this course, students are expected to be exposed to basic concepts of biometrics such as the fundamentals of fingerprinting, iris scanning, speaker verification, and hand geometry. In addition, students must also understand the underlying technologies behind biometrics, such as statistics and image processing, pattern recognition etc. A weekly lab is also designed for the provision of hands-on experience to students.

3.1 Prerequisite

The pre-requisite for this course is MATH 314: Statistics, and CIS 104: Introduction to Computer and Information Sciences, or permission from the instructor. Students from non-computing majors may not have taken CIS 104 but may have sufficient computing background. These students will grant permission to take this course. We anticipate a diverse group of students will be in this class. For example, students from biology program may be interested in this course. This would benefit the entire class as students from different background can form teams and work on term papers and projects together.

3.2 Textbooks

Two textbbooks have been used during the semesters when CIS 361 was being offered.

- Biometrics and Network Security, by Paul Reid, ISBN: 978-0131015494, Publisher: Prentice Hall PTR (Dec. 30, 2003)
- Introduction to Biometrics, by Anil K. Jain, etc., ISBN: 0387773258, Publisher: Springer (2011)

Only the first textbook was used when the course was offered for the first time in 2010 because the second textbook wasn't available in that time. Both textbooks are good choice for an introductory biometrics course. The first textbook provides appropriate coverage for using biometrics in network security. It fits our curriculum pretty well as students in our biometrics program would have to take network security course in their senior year.

Additional reading materials are assigned for students to be able to work on their term paper assignments.

4 Research and Term Paper Assignment

Students registered in the class are required to write a term paper in the subject of biometrics, and present the paper to the class. The term paper assignment was given to students in the second week of the semester so it is a semester-long assignment. The term paper assignment is team based with 2 or 3 students in a team. We give a list of topics for student teams to choose and it is mandatory that no two student teams could work on the same topic. In this way, there will be a good chance that most major topics in biometrics area were covered in student term papers, and students will very well benefit from the peer presentations as part of term paper assignments.

In Fall 2011, student teams worked on the following term paper topics.

- Biometrics and Facial Recognition
- Heart Beating Biometrics
- Iris Biometrics in Network Security
- The Future of Biometrics
- Voice Biometrics

5 Laboratory Settings

Department of Computer Science, Mathematics and Engineering has a PC lab dedicated to networking and security. We plan to equip this lab with essential biometrics devices for students to conduct hands-on lab in biometrics.

We have set aside departmental budget for the following software and hardware for biometrics lab sessions.

- MegaMatcher Software Development Kit. Including 1 Cluster Server, 2 Cluster Nodes, 1 MegaMatcher Server and 2 Client licenses.
- MegaMatcher additional Client License
- Futronic FS80 2.0 Pro USB Fingerprint scanners
- Security camera

As the alternative of term paper assignment, student teams also have option to work on a semester long lab project to develop a full-featured biometrics application consisting of biometrics devices, computing hardware and software applications. So far, our student teams have chosen a variety of topics in their programming oriented projects,, ranging from blood vessel authentication system, biometrics enhanced distance learning community, to a human resource management system that requests employees to use hand geometry or fingerprint to sign on when they arrive to work and sign out when they are ready to go home.

6 Extracurricular Activities and Related Courses

Our biometrics also plays its role in attracting students in computer science and engineering program. Many students in our Hacking and Network Security club have dedicated their interest in using biometrics to enhance network security. One student team are actively working on building a biometrics based access control system for lab doors, which involves computer hardware, networking, embedded system programming, and database techniques.

With collaboration between our faculty members, we also allow students to extend their projects and term paper in higher level security classes, including CIS 486: Network Security and CIS 485: Directed Research. In this way students are encouraged to work on more significant or research oriented projects. This effort is still ongoing.

7 Observation and Summary

Although specific assessment for the Introduction to Biometrics course is still being planned, the student feedback collected from regular course evaluation is very encouraging. The average score from students on “Rate the organization of the course material” is 4.3 out of 5. Students in particular appreciate the option of term paper and programming project as they believe this provides them chance to deepen their understanding to how biometrics systems work.

In summary, this paper describes a biometrics course that is designed for students in Biometrics and Security degree program and those in general STEM (Science, Technology, Engineering and Mathematics) programs. We believe the Biometrics course we developed would provide valuable reference for educators in Computer Science, Computer Engineering, and Information Technology areas.

We start to develop the Biometrics course plan in 2009. The initial result was presented on American Association of Engineering Education (ASEE) North Sectional Conference in March 2010. Since then the course has been taught three times. We plan to conduct a better designed course assessment when the class is offered again in Fall 2012.

8 Acknowledgment

The development of Introduction to Biometrics course is sponsored by Research Enhancement Award from NASA West Virginia Space Grant.

9 References

[1] D. Birch, "A World Away From Reality: Biometrics is the Wrong Way to Track Terrorists," Vol. 2002: Guardian, 2002.

[2] D. Maio, Maltoni, D., Cappelli, R., Wayman, J. L., Jain, A. K, "FCV2000: Fingerprint Verification Competition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, pp. 402-412, 2002.

[3] G. O. Williams, "Iris Recognition Technology," IEEE Aerospace and Electronics Systems Magazine, vol. 12, pp. 23-29, 1997.

[4] J. Daugman, "High confidence visual recognition of persons by a test of statistical independence," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, pp. 1148-1161, 1993.

[5] J. P. Holmes, "A Performance Evaluation of Biometric Identification Devices," Sandia National Laboratory SANDIA91-0276, 1991.

[6] Liao, W., Guzide O., “Work in Progress: On the Development of a Biometrics Course and its Relationship with STEM Curricula”, ASEE North Sectional Conference, March 2010.

[7] Lockheed Martin Team Wins Role on Key Department of Defense Biometrics Contract Vehicle, http://www.pressreleasepoint.com/lockheed-martin-team-wins-role-key-department-defense-biometrics-contract-vehicle.

[8] M. A. Dabbah, W. L. Woo, and S. S. Dlay, "Secure Authentication for Face Recognition," presented at Computational Intelligence in Image and Signal Processing, 2007. CIISP 2007. IEEE Symposium on, 2007.

[9] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," IBM systems Journal, vol. 40, pp. 614-634, 2001.

[10] Paul Reid, Biometrics and Network Security, ISBN: 0131015494, Prentice Hall PTR .

[11] Stephen J. Elliott, etc., Graduate Course Development in Biometrics, http://www.bspalabs.org/files/164/GradCourseASEE2002.pdf .

[12]. Wilson, A. R. Hicklin, H. Korves, B. Ulery, M. Zoepfl, M. Bone, P. Grother, R. J. Micheals, S. Otto, and C. Watson, Fingerprint vendor technology evaluation 2003: summary of results and analysis report, NIST Internal Rep. 7123, Jun. 2004.

Botzone: A Game Playing System for Artificial Intelligence Education

Haifeng Zhang, Ge Gao, Wenxin Li, Cheng Zhong, Wenyuan Yu and Cheng Wang
Department of Computer Science, Peking University, Beijing, China

Abstract - *The Botzone Intelligent Agent Platform is an online intelligent agent competition platform. The Botzone is designed to evaluate the intelligence of agents through competitions between them. The Botzone provides a context for users to learn basic AI knowledge and apply a variety of AI algorithms. It has been used as a teaching system and has served National Olympiad in Informatics (NOI) as a competition platform for many years. The Botzone is also an expandable system, which provides a set of facilities and clean interfaces, allowing users to design different competing situations and evaluate differing types of intelligent agents. In this paper, we introduce the structure, the applications, and the expansibility of the Botzone Intelligent Agent Platform.*

Keywords: Intelligent Agent, Online System, Education

1 Introduction

The potential for computer games as a tool for AI research and education continues to blossom [1]. Well-designed Games need great intelligence and good strategy. Designing an agent that plays the game is a challenging task, thus it provides an ideal context for practice of AI algorithms. Many game-based tools for CS and AI educators to use have been presented so far. Mcgovern et al. and Denero et al. have proposed several Java-Based games such as Space-war and Pac-man which can be used to teach introductory Artificial Intelligence [2, 3]. However, they don't provide a single platform for managing the various games used in CS education. Users can only develop and test their game agents offline and it is difficult for users to compare their agents with others'. Furthermore, those tools provide no uniform set of interfaces for educators to design different games to teach different algorithms.

The Botzone Intelligent Agent Platform is designed as an online turn-based strategy game playing system. Students major in Computer Science can create game agents and participate in the contest with others on the platform. In the process, they can learn basic programming skills and many Artificial Intelligence algorithms. Currently, such platforms for educators and researchers to use for turn-based strategy game development are limited. MGPS [4] is a similar platform, but it only supports normal-form games, which are only a small part of the turn-based strategy game.

A turn-based strategy game is a strategy game where players take turns when playing, distinguished from real time strategy where all players play simultaneously [5]. This type of game is commonly seen (such as chess and bridge), so it is easy to be understood by students. Furthermore, according to the complexity of the games, they can be applied in various courses. For example, the simplest Gobang game can be used in introductory classes while the Robocup game [6] can be used in advanced artificial intelligence classes. So the platform is suitable for students from undergraduates to graduates. We also provides a set of facilities and clean interfaces, allowing users to design different competing situations and evaluate differing types of intelligent agents.

We have applied the Botzone in an introductory programming course and an advanced algorithm course. We developed two turn-based strategy games for these two courses respectively. We try to lead students to experience the entertainment of programming through the Botzone. As a result, the students are more interested in such a study method, rather than in the traditional ones. In addition, we have applied the Botzone in a large-scale live competition. The competitors develop agents for an originally designed game. With the help of the Botzone, it is convenient for the organizer of the competition to select those competitors who have deeper understanding in applying AI algorithms.

We first review the structure and discuss the features of the Botzone. Then we present the main games on the Botzone and the applications of them in courses and competitions. Finally, we introduce the interfaces for developing new games, and invite users to develop new games or request for educational cooperation.

2 System Overview

The Botzone Intelligent Agent Platform is an online network system, which is now available for public use on our servers. Users may register accounts on the website and log in. Then they can view the description and the programming interfaces of the game. Once users have finished developing their agents, they can upload the source code of the agents using a simple web form. After that, the users can participate in the contests held on the platform with their own agents. They can also just start a single match with other agents on the platform.

A *match* is a single game process which is played by several agents. The agents take actions by turns until the game is over. The result of the game can reflect relative intelligent levels of the agents. A *contest* is a set of organized matches. It is participated in by a set of agents. The objective of a contest is to rank the agents by the relative intelligent levels. The matches of a contest can be organized in different competition systems, such as round robin[1] and Swiss system[2]. A series of contests can be established for a long-term project, such as a full-semester course or a large-scale competition.

The Botzone can be divided into three parts: the frontend, the storage system and the judge system. The relationship among the three parts is shown in the Figure 1.

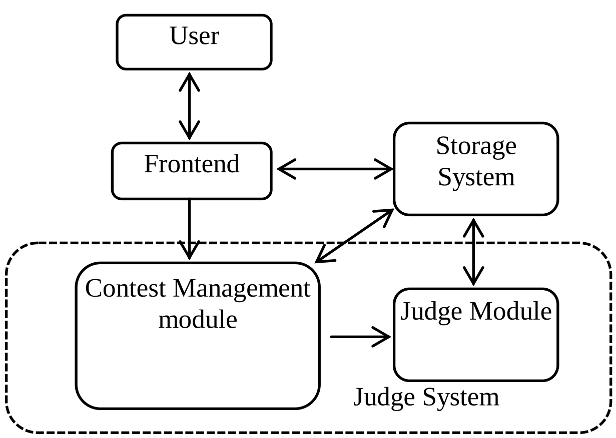


Figure 1: The structure of the Botzone System.

The frontend mainly refers to the Botzone website. Most of the interactions with users are completed by this part, such as registering accounts, uploading agents, attending contests, viewing results etc. The storage system consists of the storage of the executable files of the agents, logs, and the database storing information about users, agents and contests. The judge system consists of the contest management module and the judge module. The contest management module can run a contest according to a certain competition system. It decides the players of each single match, and then calls the judge module to run the match and record the logs, including the winner, the scores and the process of the match.

The Botzone has supported several functions which make it more entertaining as well as educational. First, it provides a visualized way to view the match process other than log files. There are FLASH animations for each game, which can replay the process of a certain match according to the log files. We find that users of the Botzone spend over half of the time on watching the FLASH animations. According to the feedbacks of the students, it also makes them more interested in developing agents and participating in the contests. Second, we allow users to choose their opponents among thousands of the agents on our platform. In this way, users can learn AI strategies from others' agents and improve their own algorithms. We have also provided a privacy function to protect users' agents. If one would not like to share his agents, he can set the type of the agent to be private and then the agent is no longer available to be chosen as opponent by others.

3 Applications

The Botzone now have three main games which are available for public use, Gobang Game, Blokus Game[3] and FourColours Game.

Gobang (Figure 2) is a simple and well-known abstract strategy board game:

> It is traditionally played with go pieces (black and white stones) on a go board (19x19 intersections). Black plays first, and players alternate in placing a stone of their color on an empty intersection. The winner is the first player to get an unbroken row of five stones horizontally, vertically, or diagonally. [7]

[1] Round-robin system is a competition organization method that gathers every n (defined by game rule) agents to play a match and total up the score of every agent.

[2] Swiss system is also a competition organization method that arranges several rounds. In every round an agent plays a match with its neighbor(s) in the current rank list, expecting the rank list is approaching stable.

[3] Blokus (pronounced like "block us") is an abstract strategy board game for two to four players, invented by Bernard Tavitian and first released in 2000 by Sekkoïa, a French company.[8]

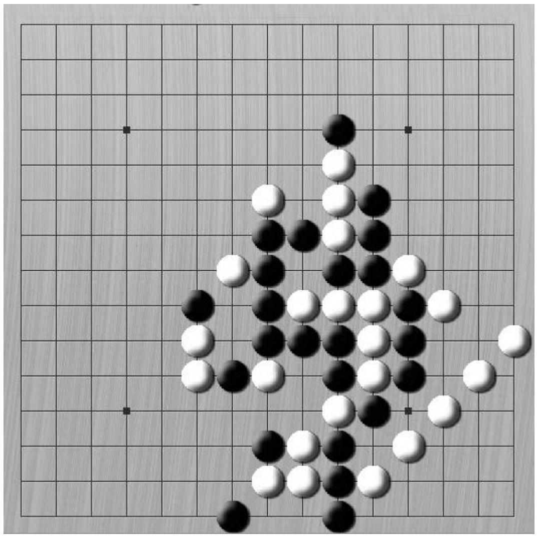


Figure 2: An example of Gobang game. In this game, the White wins.

Blokus (Figure 3) is a little more complex. The rules of Blokus are as follows [8]:

- This is a four-player game. Each player owns a set of prescriptive blocks with a unique color.
- Order of play is based on color, with blue going first, followed by yellow, red, and green.
- The first piece played of each color is placed in one of the board's four corners. Each new piece played must be placed so that it touches at least one piece of the same color, with only corner-to-corner contact allowed—edges cannot touch. However, edge-to-edge contact is allowed when two pieces of different color are involved.
- When a player cannot place a piece, he or she passes, and play continues as normal. The game ends when no one can place a piece. Result is decided by the sum of area of each player’s blocks.

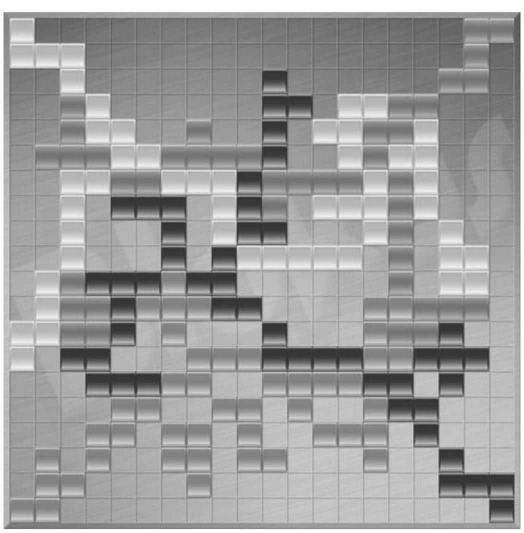


Figure 3: An example of Blokus game.

The game FourColours (Figure 4) was originally designed for National Olympiad in Informatics (NOI) 2010 and 2011. The idea of FourColours derives from the famous four color theorem. The basic rules are as follows:

- This is a game for two players. One controls Color 1 and Color 3, while the other controls Color 2 and Color 4. Color 1 goes first, followed by Color 2, 3 and 4.
- The two players paint a randomly generated planar map in turns. The map is consisted of 30-50 areas, which vary in both shape and size.
- The areas are of three types: lands, seas and mountains. The player can only paint lands. The color of a sea area is the color which appears most in the adjacent land areas. The color of a mountain area is the color which appears latest in the adjacent land areas.
- The adjacent areas can’t be painted with the same color.
- The game ends when no more area can be painted. The one who paints more total size of areas wins the game.

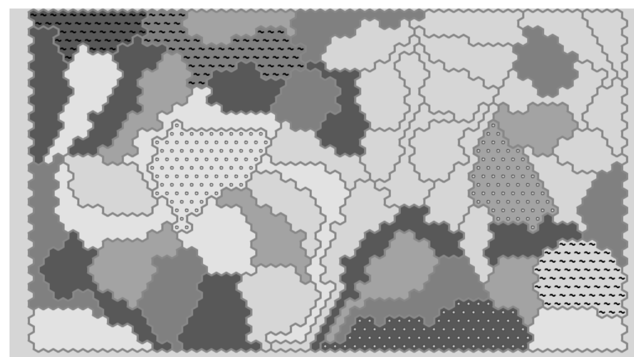


Figure 4: An example of FourColours game. Areas with tildes represent seas and areas with circles represent mountains.

These three games are all well suited to be used as a supplement to the Artificial Intelligence course or competition. They provide an ideal context for students to learn various search and pruning algorithms, especially min-max search with alpha-beta pruning, which is among the strongest algorithms in such problems [9]. Students can see the benefits and the necessity of pruning, because the game tree is too huge. And we find that the implementing of such game agents can deepen students’ understanding of the search and prune algorithms through the final exams and the questionnaire of our courses.

Furthermore, we find that playing games is also a good way for students to explore some learning algorithms. The interaction of machine learning and game playing has long been a topic since the early days of Artificial Intelligence [1]. It is difficult to come up with an ideal evaluation functions to evaluate each situation immediately. However, we can analyze the result and the log of hundreds or even thousands of matches and then get the knowledge of each situation and its outcome to improve our evaluation functions. Some possible approaches include decision trees, clustering etc.

The platform has been used in our courses for many years. We encouraged students to use all kinds of algorithms in the projects as long as the algorithm belongs to the field of Artificial Intelligence. We run a series of contests, and students are quite interested in such a project. In the advanced AI course last year, over 60% of the students kept on improving their algorithms throughout the whole semester and we have found many creative ideas. They enjoyed the assignment on the platform while learning the Artificial Intelligence. What's more, those who perform well in the project often get a high score in the related questions in the final exams.

In addition to being applied on courses, the Botzone has served NOI as a competition platform for three years. NOI is a nation-wide annual competition, aiming to improve high school students' programming skills and select members of national team to attend International Olympiad in Informatics (IOI). The Botzone helps NOI to examine competitors' programming skills and their familiarity with the related algorithms. The competition can also stimulate their interests in exploring the world of artificial intelligence further.

4 Expansibility

Since the beginning of developing our system, we have realized the importance of the expansibility. Only the few games on the platform can't satisfy the demand of various courses and competitions. The students will soon feel bored and lose interest in attending the courses and the competitions. In order to collect more interesting and educational games on the Botzone, we open up a new game interface to the public, hoping to make the platform more attractive. With the increase of the number of all kinds of games, students can learn more artificial intelligence algorithms and learn how to apply them in different games, since different games require different strategies and algorithms.

Users can design and develop new games by themselves and upload it to the Botzone. A new game needs the following files to be run well on the server:

Judge. Committers should design the rule of the new game, and implement the judge module. The judge module runs a match, maintains game process status, sends commands to agents, and outputs results. The judge module communicates with agents through controller, a built-in module in the Botzone, which plays a role as a messenger between the judge module and agents. The controller module interacts with the judge module and agents by command-line. The protocol between the controller module and the judge module is shown in the Table 1.

Display. Committers should provide a display module of the new game. The Botzone only support FLASH animations at present. The display module is in charge of displaying a single match, which is finished and has been logged on the Botzone server. The display module fetches match information from the Botzone server through HTTP request when it is played in a user's browser. Therefore, the process of every match can be replayed visually.

Introduction. Committers should provide an introduction of the new game. This introduction is prepared for the agent-developers of the new game. It should contain detailed explanation of the rule of the new game and the communication protocol between the judge module and agents.

Sample. Committers should provide a copy of source code of a sample agent for the new game. This sample is also prepared for the agent-developers. It could be equipped with a basic intelligent level, but it should cover all the cases on the communication protocol.

Table 1: Communication Protocol between the controller module and the judge module

	Instruction	Function
Controller To Judge	[PLAYERNUMBER] number	Tell Judge the number of players
	[PLAYERNAMES] name1 name2…	Tell Judge the names of the players
	[START]	Notify Judge that game started
	[TIMEOUT]	Tell Judge that the last agent got a time-out error
	[NEXT] pre_command	Feedback the output of the last agent
Judge To Controller	[PLAYER] player_num	Appoint the next agent
	[COMMAND] command	Send command to the next agent
	[KILL] player1 player2…	Stop one or more agents
	[GAMEOVER]	Notify Controller that the game is over
	[RESULT] score1 score2…	Output results
	[VIEWLOG] one line log	Output log file for the display module

We will review these files after committers upload them. The new games that are clearly defined, moderately challenging and fairly interesting will be accepted. In this way, we can expand the application of our platform.

Recently, we have accepted a new game named Snake (Figure 5), which is developed by an ordinary user. The main rules of Snake are as follows (similar to the Snake game in mobile phones):

- This is a game for two players. Each player controls a snake.
- The snakes can move ahead, turn left or right in every turn. It will die if it moves out of the map, or hits the body of either himself or the opponent.
- There are some stars coming out in succession, which may be eaten by snakes. When a snake eats a star, it will lengthen its body and gain a point. The longer, the better.

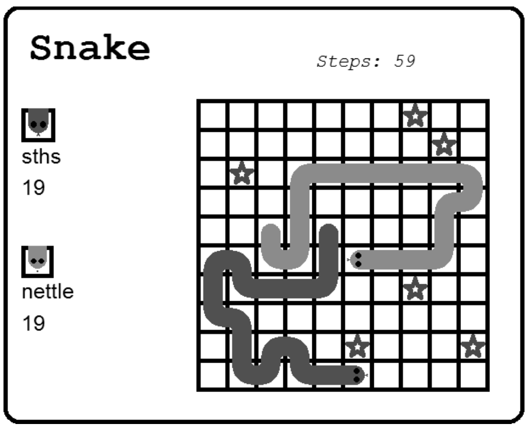


Figure 5: An example of Snake game.

With the increase of the variety of games on the Botzone, it can be applied in more courses such as game theory class or machine learning class, and the students can learn more related artificial intelligence algorithms and ideas.

5 Conclusions

The Botzone is an educational and entertained online game playing system. It can be used to teach students basic programming skills and algorithms in the AI field. It also can be applied to hold large-scale competitions. What's more, it is an expansible system so that it can provide services in more situations and for more users. In the recent years, the platform has been successfully applied in various courses and competitions.

The platform is now available for public use on the internet. We are eager for people to design new games for us and use our platform in courses and competitions. Those interested parties may contact the lead author with such requests.

In the future, we will continue improving our system and collect more excellent games developed by users. We will also consider developing the function that enables users to play game with the agents.

6 References

[1] Bowling, M.; Furnkranz, J.; Graepel, T.; and Musick, R.. "Machine learning and games". Machine Learning 63:211–21, 2006.

[2] Mcgovern, A.; Tidwell, Z.; and Rushing, D.. "Teaching Introductory Artificial Intelligence through Java-Based Games". AAAI Symposium on Educational Advances in Artificial Intelligence, North America, 2011.

[3] DeNero, J.; and Klein, D.. "Teaching introductory artificial intelligence with Pac-man". Educational Advances in Artificial Intelligence (EAAI), 2010.

[4] Macglashan, J.; Miner, D.; and Desjardins, M.. "A Game Playing System for Use in Computer Science Education". Florida Artificial Intelligence Research Society Conference, North America, 2010.

[5] Wikipedia. Turn-based strategy—Wikipedia, the free encyclopedia. [Online; accessed 7-February-2012].

[6] Kitano, H.; Asada, M.; Kuniyoshi, Y.; Noda, I.; Osawai, E.; and Matsubara, H.. "RoboCup: A Challenge Problem for AI and Robotics". RoboCup, pp. 1-19, 1997.

[7] Wikipedia. Gomoku—Wikipedia, the free encyclopedia. [Online; accessed 7-February-2012].

[8] Wikipedia. Blokus—Wikipedia, the free encyclopedia. [Online; accessed 7-February-2012].

[9] Allis, L. V.; Herik, H. J. V. D.; and Huntjens , M. P. H.. Go-Moku Solved by New Search Techniques. Computational Intelligence - CI 12(1): 7-23, 1996.

Accrediting Online Programs in Computing Disciplines

R. Byrd
School of IT and Computing, Abilene Christian University, Abilene, Texas, United States

Abstract – *Because of technology tools and collaborative pedagogies, online courses and programs are now able to provide a means of effective learning for more than just a small percent of non-traditional students. Teaching methods and materials, as well as preparation and times allotment are vastly different than what is effective in a traditional classroom setting, but once making the paradigm shift student learning can be improved for online learning. In many cases, online learning achievement surpasses that of the classroom setting. Social learning, advanced electronic course materials, and effective lab simulation tools are three areas that have significantly improved due to advances in Internet availability and reliability.*

Keywords: Accreditation, Online Programs, ABET, Distance Education

1 Introduction

1.1 Background

Distance education has been around at least since Isaac Pitman taught shorthand in Great Britain via correspondence in the 1840s [1]. By 1858, the University of London had created an international distance learning program [2]. Australia's Department of Correspondence Studies was offering programs by 1911 [3]. With the advent of audio and other multimedia in the 20th Century, distance learning programs flourished.

Today, the Internet is extending the reach and options of distance learning. Technology now enables schools to offer online programs with better student and instructor collaboration, flexible (asynchronous learning), problem specific feedback, automated interactive lessons or presentations, and simulations of almost anything conceivable. In reviewing literature for online delivery, statistical trends show an increase in the percent of online courses, as well as a larger diversity in the types of courses being offered online, and the types of organizations offering online courses. Online delivery has progressed to where accreditation of technical academic programs is now possible. However, developing an online academic program is more complex than simply uploading lectures and assignments to a website.

1.2 Overview

To develop an accreditable online program, it is important to understand the differences between online and face-two-face (f2f) teaching and learning. Seeing how and why the delivery modes differ will enable online development to be optimized for specific courses and audiences.

Following, is a discussion of the key factors that make an online program accreditable. In reality, the factors are the same as for a face-to-face program. The difference is in where the emphases are placed and what methodologies are involved in attaining the same criteria. Some of those different methodologies employ technology tools that, while they would be helpful in a face-to-face program, are essential in an online program.

Finally, some considerations for transitioning from a face-to-face program to an online program are presented.

2 Differences between online and f2f

Several questions arise when ascertaining the differences between online and face-to-face course delivery. Answers to these questions should help clarify the different learning environments and how the online environment may be optimized to further improve learning outcomes.

2.1 Who registers for online v. f2f courses?

Originally, online courses were not regarded as being credible. Only general education electives or adult learning and self-improvement type courses may have been acceptable. Concurrently, graduate level courses and programs where most of the student assessment consisted of grading research papers became common for online programs. Increasingly, students needing to "make-up" a course in their academic program may have registered for an undergraduate online course. Now, even within technical disciplines, complete degree programs are available online [4].

Students choose online over face-to-face programs for a variety of reasons, including: accessibility, convenience, and learning style. Whether due to work schedules conflicting with course offerings, travel distance, or other constraint, face-to-face courses and programs are simply not always accessible.

Convenience of online programs is represented by the ability to work on a course when and where a student desires to do so. With most online formats it is sometimes possible to work eight hours in a single day and then skip a day or two. Another convenience is the ability work on the course late at night after children or other commitments are not interrupting.

Some learners, whether due to personality type or even medical condition, do not perform well in a classroom environment. Online delivery offers an alternative to the rigid face-to-face structure. While student interaction via discussion forms, wikis, or other collaborative formats is now prevalent in online courses, face-to-face delivery might not include "team time" during the prescribed meeting time due to a perception that it is not an effective use of time.

2.2 How will learning or achievement differ?

A Department of Education meta-study collected research data from online courses taught between 1996 and 2008, reported that online delivery improves student outcomes or fosters even a better learning environment [5]. Other studies (e.g., [4]) report there is no difference in the learning level between courses taught online versus face-to-face. In fact, nearly every study in this review of literature referred to the level of achievement in some way. Each review had a set of specific circumstances and survey item wording. Clearly, the answer to the question, "which is better?", is not a simple one. It is apparent that type of course, learning style, need for flexibility, and other factors play an important role in determining effectiveness of an online course.

2.3 What activities will be different?

Face-to-face delivery offers the benefit of a set schedule that requires students to be on time and in-pace with the planned schedule each day. This is great for those who need a rigid schedule. Online delivery may allow for flexibility in completing assignments. There is still a need for communication, but it will be more two-way than one-way. One of the greatest differences between face-to-face and online delivery is the social learning context of the online mode. Social learning is often a predominant form of online learning.

Productive forums must be designed to enhance learning. Professors are no longer even instructors, but facilitators. Quizzes can be used, not to evaluate or assess whether a student read and learned the material, but to encourage and help the student get through the reading assignment [6]. This means that quizzes must be rewritten to accomplish a different purpose. Assignments must be designed to engage the student —now called the learner—to learn for him/herself and from peer learners.

Delivery media now have the flexibility of being, yes, a talking head (vernacular for a video of a lecturer), a PowerPoint presentation, an interactive Flash video, a team-based project where students must each perform some realworld activity, a computer simulation of a computer network, or any other creative medium that works to engage the student to learn the course or program objectives.

This drastic change in delivery media and activities will probably initially be more of an adjustment for the instructors than for the students. Institutional involvement will be required to support the online program on many levels, including delivery system technology and administrative support, professional development, instructional (i.e., course) design, multimedia technicians, and more.

3 What are key factors that make an online course/program accreditable?

The obvious short answer is, of course, "Meet the ABET criteria." But doing so will require looking at the same criteria from a different perspective. First, let's review a sample of the criteria.

3.1 What do the criteria require?

Following is an abbreviated representation of the ABET criteria for a computing program [7]. Other program criteria would be similar.

1 Evaluate student performance. Monitor student success. Advise students.
2 PEOs must be consistent with institutional mission, constituencies, and these criteria.
3 Documented outcomes that prepare students to attain PEOs.
4 A continuous improvement process is in place...
5 Curricular requirements are consistent with PEOs and designed so student outcomes are met.
6 Faculty have expertise and educational background consistent with its expected contributions.
7 Infrastructure provides an environment conducive to learning. Students are trained to use it.
8 Institutional adequately supports the quality and continuity of the program.
9 Program specific criteria are also met.

3.2 Which elements of the criteria may be affected differently by an online program?

Those desiring to accredit an online program would look in detail at every aspect of the criteria and ensure compliance, but the following considerations should enable those interested in the process to grasp how the perspective will need to change as they migrate from face-to-face delivery to an online delivery mode.

3.2.1 Student communication media for advising, evaluation, etc.

Since it is conceivable that a student may rarely, if ever, be on campus, it should be possible for all student administrative processes to be accomplished by other than face-to-face means. Of course, all program and course information should be available online. In addition, advising will need to be accomplished using new methods. Phone calls, texting, chatting, blogging, eportfolios, and videoconferencing can be combined with online interactive content and course registration. Deploying all these new techniques will require time to develop quality processes, train staff and faculty, and develop new formats of content.

Assessment methods, both for in-course objectives and student outcomes will need to be analyzed based on the new delivery mode. An advantage of the online delivery mode is that most of the student work, as well as student evaluations, will already be in electronic format and could be made available for review.

3.2.2 Consistent with institution and needs of constituencies

Being consistent with the institutional mission and needs of the constituencies, will require, first of all, buy-in for online delivery. If policies or mindsets are not favorable toward an online program they will need to be changed before continuing with program development. Showing precedence for online programs or showing equivalent outcomes, skill sets, or workplace effectiveness for students with online educational experience will help convince those who may doubt the efficacy of online delivery.

3.2.3 Meet the Student Outcome criteria [7]

Methods may be different in many cases but outcomes can be achieved, using technology, social learning techniques, and creativity.

a) Apply computing-related knowledge and skills appropriate to the discipline
b) Analyze a problem and identify and identify and define solution requirements
c) Design, implement, and evaluate [some sort of a solution]
d) Function effectively on teams to accomplish a common goal
e) Understand [the soft skills] of the discipline
f) Communicate effectively with a range of audiences
g) Analyze the local and global impact of [the discipline of people and culture]
h) [Be able to] engage in continuing professional development
i) Use current techniques, skills, and tools necessary for the discipline

3.2.4 Continuous Improvement

Assessment and evaluation processes may differ in data collection methods and sources. Direct over indirect methods will still be necessary, but there are more sources from which to collect direct data. For example, one aspect of engagement may be how many times a student comments on a forum. Such data is available from activity logs of many popular Learning Management Systems (e.g., Moodle, BlackBoard). This criterion, may actually be easier to attain, with a little thought, because the nature of online delivery complements continuous improvement.

3.2.5 Curriculum ensures PEOs and Student Outcomes are met

Mode of delivery may be transparent to the specific curriculum. Course currency will be more of an apparent issue with online than face-to-face. As online delivery is a more collaborative effort among facilitators and learners, learners will be more disposed to offer suggestions (and complaints) as to how to improve the course, thus keeping it current. This also applies to Criterion 4 above.

3.2.6 Faculty

Faculty will need to understand the differences between face-to-face and online teaching styles, student body diversity and uniqueness, time allotment, and types of teaching activities. This is critical to a successful online program and will require an initial investment of time and funding to ensure faculty are prepared for the difference.

3.2.7 Infrastructure provides an environment conducive to learning

Online delivery will require a web-based system requiring support. A blended program will require both the web-based system as well as physical student facilities. In addition to the server and application support, there will be a need for student and faculty support in the form of helpdesk services. Passwords are forgotten; network connections are sometimes problematic, the Learning Management System may not be intuitive for some students; simulation software may need configured on a student computer; etc.

3.2.8 Institutional Support

Financial, policy, faculty training, and course development support will be necessary for online delivery to work. Beyond what is needed to support a face-to-face accredited program, such as administrative support for managing assessment, etc., online program and course development requires a complete infrastructure of its own.

The registrar, university curricular committees, and campus offices will need to fully support an online program. Online course development stipends or release time will be necessary to build effective courses. An online course requires far more planning and preparation than does a face-to-face course. This is primarily because once the course begins, changing the formats, assignments, and other course aspects is more difficult.

Faculty who supply the course content are called subject matter experts (SMEs). Course developers are those who manage the actual creation of the course and are experts in which activities will accomplish a desired purpose. They will help word an assignment to make sense when there is no face-to-face interaction as there would be in a traditional classroom conversation. Technology/multimedia experts assist in or create lessons containing Flash, podcasts, and other broadcasting or creative lesson formats. To develop an effective online course, these roles must be present. Without them, there is only an electronic version of a face-to-face course: all the disadvantages with none of the advantages.

3.2.9 Program-specific outcomes

These will need to be addressed individually. The biggest question is probably that of hands-on requirements and other lab situations. The answer is basically: careful course design, simulation, and understanding “equivalent learning achievement”.

What are the key factors, again? Meet the criteria, meet the PEOs and SOs, satisfy the various constituencies.

4 Methods for transitioning from f2f to online

In some cases an institution may be able to offer a completely new degree and begin with an online program. Usually, the institution will want to migrate an existing face-to-face academic program to online or add online components to an existing program. While some of the following considerations have been briefly addressed above, they are explicitly mentioned here as transitional concepts.

4.1 Understand that online courses do not teach themselves [8]

It will not only take tremendous effort and resources to develop the online program, but also to deliver it. The unaware instructor will spend more hours grading assignments and participating in discussions for an online course than time spent in a face-to-face course. It is important to be involved in the social learning process. It is also important not to quickly provide answers to learner questions, but to allow the social learning environment to work.

4.2 Automate parts of the course whenever possible [8]

To prevent being overworked for a course, design the course so that each activity has meaning and spread out the activities that will require significant time to assess. Rather than a forum discussion on a chapter reading, pick a single key element of the chapter and develop an assignment around it that uses a discussion to engage students. Use a quiz activity as a tool to ensure the chapter is read, by setting the time for the quiz to be *during* the period of the reading. Students will then use the quiz to ensure they are understanding the material and it will keep them focused on the reading [6].

4.3 Enable calendars and "upcoming events" areas to remind students of events and deadlines

Most Learning Management Systems have options for course calendars that automatically post upcoming course activities and deadlines on the main course page. Plan on using these features to keep your students—and instructor—aware and involved in the course.

4.4 Determine what kind of technical support you have available

Whether web services, course development, or helpdesk, technical support will be a critical factor in the success of an online program. Work closely with the administration to ensure healthy communication and support are maintained in this area.

5 Tools and technologies that enable attainment of Student Outcomes in online courses

5.1 Methodologies as Tools

While technology is a key factor in developing and delivering an online program, the most effective tools are those processes and techniques (methodologies) that enable to students to become true learners. Appropriate use of discussion forums and group assignments can instill course objectives and concepts as much as any lecture or in-class assignment.

5.2 Software Technologies

There are also many technologies that can enhance learning, whether in a face-to-face program or online.

For computing programs within ABET/CAC, virtual machines (VMs) enable each student to have their own server for which they are solely responsible. Open source and proprietary versions are available. This researcher uses both, depending on the course.

Simulation software is useful for presenting concepts and enabling students to practice as if the computer network were real. Most of the networking certifications (e.g., Sysco) offer academic partnerships where faculty are the instructors, but the certification work is logged through the provider's online system. Other providers offer certification material as a subset of the complete course package [9]. In some cases, not having the "luxury" of face-to-face meeting forces the course developers to incorporate tools that are more realistic and less of an academic nuance.

Wireshark [10], for example, a network packet analyzer, can be used in an academic environment, but is also useful in an operational setting. Working closely with constituencies will provide insight as to which concepts are critical for a student to master and which workplace skills and experience will override learning of an academic concept. Each program discipline and set of constituencies will create a different perspective in this matter.

CyberSeige [11] is a computer and information security simulation training game. It is available from the Naval Postgraduate School, and was designed specifically as an educational tool.

5.3 Equivalent Learning

This is a key to understanding the benefit of online instruction: while it is possible that not quite as many facts or even a few concepts may not be completely understood by students in an online program, other benefits, such as practical experience with operational applications and services, may outweigh any minor so-called deficiencies in the overall learning of a student. Student Outcomes still need to be enabled, but they don't have to necessarily be enabled in the traditional manner.

6 References

[1] "Distance education," *New World Encyclopedia*. October 31, 2011. www.newworldencyclopedia.org/entry/Distance_education?oldid=956391

[2] University of London. March 14, 2011. http://www.londoninternational.ac.uk/about_us/index.shtml

[3] M. White. "Distance education in Australian higher education — a history". *Distance Education* 3(2), 255–78, 1982.

[4] J. Bourne, D. Harris, & F. Mayadas. "Online Engineering Education: Learning Anywhere, Anytime." *Journal of Engineering Education*, 94(1), 131-146, 2005.

[5] B. Means, Y. Toyama, R. Murphy, M. Bakia, & K. Jones (September 2010). "Evaluation of Evidence-Based Practices in Online Learning: A Meta-Analysis and Review of Online Learning Studies", *U.S. Department of Education: Office of Planning, Evaluation, and Policy Development*, http://www2.ed.gov/rschstat/eval/tech/evidence-based-practices/finalreport.pdf

[6] R. Byrd. "Does the early bird get the worm?: Analyzing response patterns for quizzes and assignments," *Southwest Teaching and Learning Conference*, Texas A&M University-Kingsville, System Center-San Antonio. March 27, 2009. http://swtlc.files.wordpress.com/2010/04/swtlc-proceedings-2009.pdf

[7] "Criteria for Accrediting Computing Programs, 2011 – 2012," *ABET* http://www.abet.org/accreditation-criteria-policies-documents/

[8] R. Sheridan. Reducing the Online Instructor's Workload: Tips on designing and administering online courses can save faculty valuable time while producing high-quality content", *Educause Quarterly*, 29(3), 2006. http://www.educause.edu/EDUCAUSE+Quarterly/EDUCAUSEQuarterlyMagazineVolum/ReducingtheOnlineInstructorsWo/157409

[9] "LabSim Educational Solutions," *TestOut*, http://www.testout.com/home/educator-resources/labsim-educational-solutions

[10] Wireshark. (2011). http://www.wireshark.org/

[11] CyberSeige (2011). http://www.cisr.us/projects/cyberciege.html

Successful ABET Accreditation at King Faisal University – Rubric based Assessment Plan for Continuous Improvement

Khalid Abdulmohsen Buragga[#], Muhammad Asif Khan[#]

[#]College of Computer Sciences and Information Technology
King Faisal University, Saudi Arabia
kburagga@kfu.edu.sa
asifkhan2k@yahoo.com

***Abstract*— An accreditation is recognition of the quality of a program and it indicates that the program meets standards and criteria set by an accrediting agency. Academic institutions can demonstrate their commitments to maintain quality of their programs by an accreditation. This paper presents experience of successful ABET accreditation for two programs i.e. computer sciences and information systems at King Faisal University, Saudi Arabia and an assessment plan for continuous improvement. A rubric based assessment plan has also been presented in this paper.**

***Keywords*— ABET, accreditation, assessment plan, assessment quality, quality**

I. Introduction

The increasing demand of highly skilled graduates in computer and information technology is at large throughout the world. To meet this demand College of Computer Sciences and Information Technology (CCSIT) was established in 2003 at King Faisal University, Saudi Arabia. The CCSIT comprises of four departments i.e. Computer Science (CS), Information Systems (IS), Computer Networks (CN) and Computer Engineering (CE). Currently two separate programs in Computer Sciences and Computer Information Systems offer Bachelor of Science degrees while the courses from Networking and Computer Engineering are taught in both the afore mentioned degree programs. The college offers two semesters in an academic year beginning in Fall and Spring in which each semester consists of 15 weeks.

In order to build a curriculum the college started developing programs based on the guidelines from ABET Inc, USA (Accreditation Board for Engineering and Technology) and Association for Computing Machinery (ACM) in 2003. The college always strives to improve the quality of the programs and in this connection an assessment of programs was conducted to preparing a self study report for National Commission for Academic Accreditation and Assessment (NCAAA) in 2007. The NCAAA was established in 2004 in cooperation with the Ministry of Higher Education, Saudi Arabia for continuing improvement in universities education [1]. There were eleven standards developed by NCAAA and the institutions in the kingdom were encouraged to apply for NCAAA accreditation. The CCSIT applied for the accreditation in 2007 and submitted a self-study report for one program namely Computer Science in March 2008. In November 2008 NCAAA arranged a reviewers' visit to the CCSIT in order to evaluate the programs. The reviewers provided some recommendations to improve quality of the programs and consequently the college reviewed curriculum of both the programs in 2009. In continuing improvement of curriculum of both the programs (i.e. CS and CIS) the Dean of CCIST expanded his vision and took a challenge to achieve international accreditation from ABET Inc, USA.

An accreditation is recognition of the quality of a program and it indicates that the program meets the standards and criteria set by an accrediting agency. Academic institutions can demonstrate their commitments to maintain quality of their programs by an accreditation. An accreditation status requires documentation and reports that show the academic institution is successful in fulfilling the required criteria [2]. Before the advent of ABET as a single entity the Computer Science Accreditation Commission (CSAC) conducted the accreditation for computer science program under the patronage of Computing Sciences Accreditation Board (CSAB). In 1998 ABET and CSAB signed a Memorandum of Agreement in November 1998 to integrate CSAB's accreditation services with ABET, with a transition time of approximately two years [4]. Later, in 2001 CSAB integrated into ABET and CSAC became CAC (Computing Accreditation Commission) under the patronage of ABET [5]. Now programs are accredited by ABET Inc, USA.

II. Program Outcomes

A program in any discipline must be able to equip students with knowledge, skills and abilities that are essential to enter either in work force, professional study or practice. In the beginning of curriculum development of both the programs different outcomes were set in order to prepare students competitive in the marketplace. Table I shows the old program outcomes for both CS and CIS programs.

TABLE I
PROGRAM OUTCOMES OF OLD CS AND CIS CRRICULUM

Program outcomes for CS curriculum	Program outcomes for CIS curriculum
a. knowledge of scientific principles that are fundamental to the computer science application areas b. ability to abstract, analyze and solve complex problems dealing with computer science application areas c. ability to complete a comprehensive design in one of the application areas involving the use of design standards and solve realistic constraints that include many of the following considerations: economic, environmental, sustainability, manufacturability, ethical, health, and safety, social and political d. ability to conduct experiments and interpret data related to his specialization e. good understanding of the expectations of a computer science employee who practices in an industrial or governmental organization f. commitment to continue developing knowledge and skills after graduation g. commitment to succeed personally and to assure employers success h. ability to deal with contemporary issues facing society in the local, global, Social, economic and political contexts as they relate to the practice of their Specialization	a. knowledge of scientific principles that are fundamental to the information technology application areas b. ability to abstract, analyze and solve complex problems dealing with information technology application areas c. ability to complete a comprehensive design in one of the application areas involving the use of design standards and solve realistic constraints that include many of the following considerations: economic, environmental, sustainability, manufacturability, ethical, health, and safety, social and political d. ability to conduct experiments and interpret data related to his specialization e. good understanding of the expectations of an information technology employee who practices in an industrial or governmental organization f. commitment to continue developing knowledge and skills after graduation g. commitment to succeed personally and to assure employers success h. ability to deal with contemporary issues facing society in the local, global, social, economic and political contexts as they relate to the practice of their specialization

In connection with the preparation of ABET accreditation and further development the college started to review the curriculum of both the programs in order to meet the criteria set by ABET. There were 40 courses identified for evaluation purpose and 34 courses selected to make modification/changes. The recommendations were given to make modifications in text books, grading, descriptions and outcomes of the courses. The recommendations were deliberated at different levels in the college before presenting to the university council for approval. The new curriculum has been implemented in Fall 2010, and now in the college both the programs have old and new curriculums in place. Table II shows the new program outcomes for both CS and CIS programs.

TABLE II
PROGRAM OUTCOMES OF NEW CS AND CIS CURRICULUM

Program outcomes for CS curriculum	Program outcomes for CIS curriculum
a. An ability to apply knowledge of computing and mathematics appropriate to the discipline b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution c. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs d. An ability to function effectively on teams to accomplish a common goal e. An understanding of professional, ethical, legal, security and social issues and responsibilities	a. An ability to apply knowledge of computing and mathematics appropriate to the discipline b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution c. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs d. An ability to function effectively on teams to accomplish a common goal e. An understanding of professional, ethical, legal, security and social issues and responsibilities

f. An ability to communicate effectively with a range of audiences g. An ability to analyze the local and global impact of computing on individuals, organizations, and society h. Recognition of the need for and an ability to engage in continuing professional development i. An ability to use current techniques, skills, and tools necessary for computing practice. j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices. k. An ability to apply design and development principles in the construction of software systems of varying complexity	f. An ability to communicate effectively with a range of audiences g. An ability to analyze the local and global impact of computing on individuals, organizations, and society h. Recognition of the need for and an ability to engage in continuing professional development i. An ability to use current techniques, skills, and tools necessary for computing practice. j. An understanding of processes that support the delivery and management of information systems within a specific application environment

The old curriculum requires 136 units in which 20 units are university requirements, 66 units are the college requirements and 50 units are needed for specialized courses. The new curriculum requires 121 units in which 8 units are the university requirements, 57 units are the college requirements and 56 units are needed for specialized courses. First two years of the curriculum are common to both the programs whereas the last two years develop students' skills and knowledge in their chosen specialization. The courses and their learning outcomes were completely reviewed and all the faculty members were involved in the process. Each faculty member of the relevant course was asked to re-design the learning outcomes of the course in order to map them with the program outcomes of each program. In the beginning of 2010 the college had applied for ABET accreditation and immediately started preparing the Self Study Report (SSR) that was required by ABET. The accreditation criteria set by ABET consists of the following items: (i) students (ii). program educational objectives (iii) program students outcomes (iv) continuous improvement (v) curriculum (vi) faculty (vii) facilities (viii) support (ix) program criteria.

The dean and the faculty members of the college worked incessantly and deliberated on each point to be included in the report. Faculty members and the staff at the college participated in the process overwhelmingly and provided the required data swiftly. After several weeks' deliberations, efforts and data collection the complete information was documented and the SSR was prepared and submitted to ABET in July, 2010. The report encompassed all the criteria set by the ABET and each criterion was explained full in all aspects.

III. ABET VISIT

In November, 2010 CCSIT successfully qualified ABET visit when a team of ABET reviewers visited the college and evaluated both the programs. All the assessment materials for both the programs were prepared and presented to the reviewers. At the end of the visit the reviewers provided verbal comments about their observation. We were satisfied and convinced with the comments as there was no deficiency in our programs and only two concerns and a weakness were identified. A draft statement was sent to us in February, 2011 in which two concerns and a weakness were mentioned in both the programs. In ABET terminology a concern means that *a program currently satisfies a criterion, policy or procedure, but the potential exists for the situation to change such that the criterion, policy, or procedure may not be satisfied*. Similarly, a weakness shows that a *program lacks strength of compliance with a criterion, policy, or procedure to ensure that the quality of the program will not be compromised. Remedial action is required to strengthen compliance with the criterion, policy, or procedure prior to the next review* [3]. One of the concerns was related to criterion 'Program Outcome ' (h) and it was stated that there was no evidence that the program enables students to recognize the need for and have an ability to engage in continuing professional development. Another concern was mentioned in criterion 'Curriculum' and it was described that (i) evidence that program outcome (h) is being achieved is not clear (ii) some text books are old editions (iii) the program's semester hours of general education minimally supports the need for preparing students to function in modern society. The weakness was identified in continuous improvement as according to the draft letter (i) a lack of specified targets for achievement levels (ii) the college had a lack of standardized methods of evaluation for students achievements, (iii) minimal evidence that program improvements are related to the evaluation of the assessment data and (iv) the program improvement process is not precisely documented.

We immediately started fixing up the concerns and the weakness mentioned in the draft letter. Concerns were eliminated immediately by establishing evidence of students continuing professional development, by adjusting the semester hours of general education and by including new editions of text books in the curriculum. In order to remove the weakness remedial actions were set-up. A target of achievement level was set on scale 4, an assessment method based on rubrics was developed and a plan to document the assessment process was crafted. We prepared detailed reply to the draft letter and sent to ABET in March, 2011. As a result, we received the letter from ABET granting full 6 years accreditation (until 2017) to both CIS and CS programs, a great success and great achievement.

IV. FUTURE PLAN AND DEVELOPMENT

In order to maintain the quality of a program it is necessary to embody continuous improvement in the program. A two-year cyclic assessment plan has been designed for 6 years (i.e. until 2017). Figure 1 shows the two-year cyclic plan.

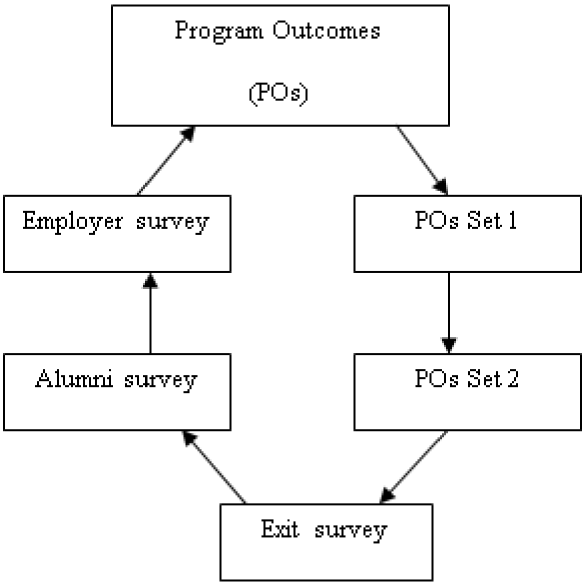


Fig. 1. A two-year cycle

As the Figure 1 shows the program outcomes are evaluated and in each semester a set of program outcomes will be evaluated. Similarly different surveys including exit survey, employer survey and alumni survey will be conducted in order to determine the level of achievement of the program outcomes. This cycle will be repeated three times until 2017

Two methods of assessment have been devised in order to evaluate the program outcomes i.e. direct assessment and in-direct assessment. The direct assessment method is comprised of students work i.e. rubric-based assessment (exams, assignments, projects, quizzes etc.) while the in-direct assessment is based on different surveys such as learning outcome survey, employer survey and alumni survey.

A. *Rubric-based Assessment*

ABET has recommended rubric assessment method for the evaluation of program outcomes as rubrics may become part of grading process and lead to expectation for students performance. A rubric consists of three components namely scale, performance indicators and descriptors. We have developed a rubric for each program outcome i.e. we developed a scale, set of performance indicators (3-4) and descriptors. Based on the learning outcomes of courses we have selected a set of courses for each program outcome. Table 3 shows a sample of a rubric that is developed for program outcome B. In order to determine the level of achievement in the program outcome B we have designed three performance indicators described in the Table III

TABLE III
AN EXAMPLE OF RUBRIC

Program Outcome B An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution				
Performance Indicator	**1 Unsatisfactory**	**2 Developing**	**3 Satisfactory**	**4 Exemplary**
Analyze a given problem or a system	Student is not able to analyze the given problem or the system	Student is partially able to analyze the given problem or the system	Student is mostly able to analyze the given problem or the system	Student is completely able to analyze the given problem or the system
Identify the computing requirements for a given problem	Student is not able to identify the computing requirements for the given problem	Student is able to identify some of the computing requirements for the given problem	Student is able to identify most of the computing requirements for the given problem	Student is able to identify all the computing requirements for the given problem
Define appropriate solutions to meet a given problem's specification	Student is not able to define appropriate solutions to meet the given problem's specification	Student is partially able to define appropriate solutions to meet the given problem's specification	Student is mostly able to define appropriate solutions to meet the given problem's specification	Student is completely able to define appropriate solutions to meet the given problem's specification

A scale consisting of four levels (1-unsatsifactory, 2-developing, 3-satisfactory, 4-exemplary) is devised and the description of each level is provided.

To determine the level of achieve of a program outcome by students one or more courses are selected. Each question in an assessment component (i.e. exam, quiz etc.) of a course that is related to a performance indicator is evaluated. The numbers of students who achieve each level against performance indicator are tallied and finally average of each program outcome out of 4 is calculated. It is important to note that the assessment is checked by two different instructors who do not have any connection with the course.

V. CONCLUSION

This paper has discussed the ABET accreditation experience and successful achievement of the accreditation. A future plan has also been presented for continuous improvement in order to maintain the quality and standards set by the ABET.

REFERENCES

[1] National Commission for Academic Accreditation and Assessment [http://www.ncaaa.org.sa].

[2] J.Enderle, J. Gassert, S. Blanchard, P. King, D. Beasley, P. Hale, & D. Aldridge, The ABCs of Preparing for ABET, IEEE Engineering in Medicine and Biology Magazine, 22(4), 2003, 122-132.

[3] Accreditation Board for Engineering and Technology , Glorrary [http://www.abet.org].

[4] Doris, L., Lee, S., and Kenneth, M., Integration of CSAB and ABET, In Proceedings of 30th ASEE/IEEE Frontiers in Education Conference, 1, 2000, USA.

[5] Curtis, C., Pankaj, M., and Marcello, V., Assessment of CAC Self-Study Report, In Proceedings of 34th ASEE/IEEE Frontiers in Education Conference, pp. 12-17, 2004, USA.

Data Representation in Digital Design, a Single Conversion Equation and a Formal Languages Approach

Hassan Farhat
University of Nebraska at Omaha

Abstract- *In the study of data representation in digital design and computer organization, we work with finite domains. We found that the early concepts of formal languages can be applied in the study. We found as well that covering the different data representation in sequence is beneficial. Traditional texts present the topics in a dispersed fashion.*

The contribution of the paper is in: (a) applying the concepts of an alphabet, a word over an alphabet, and a formal a language to a digital design and computer organization course; and (b) deriving a single conversion equation for the different data encodings used.

We first review the different binary encodings used in presenting numeric data. We then generate a generalized equation that can be applied to each of the main data types. By proper parameter substitutions we can convert between the different data types.

1 Introduction

Finite automaton forms one of the corner stones in the study computer science; it is also fundamental in the study of computer engineering. Finite automata and pushdown automata are found in the study of compilers in software [2, 3]. Finite automaton is also found in the study of hardware design [1, 7]. In [7] the design of the control unit and datapath of a processor can be modeled as a finite state machine (FSM) with a datapath (FSMD). Algorithmic state machines (ASMs) are generalization of finite state machines and are used throughout the design process at the register transfer level description of a computer.

At the early stages of studying digital design and computer organizations, the concepts of finite state machines are deferred towards the second half of the course. They are covered under the concepts state minimization and sequential circuits realizations. Instead, in the early part of the course, the study of integer data representation and Boolean function realization are covered. Later in the study, the different representations of real data type are covered. The study of data representations, is dispersed; at different parts of the course, the study includes unsigned data types for integer and fixed point real numbers, signed data types, and floating point number representation. When discussing data types we also discuss overflow and arithmetic errors for the many data representations.

The early parts of a formal languages course include, the definition of an alphabet, the definition of a word of arbitrary length of over an alphabet, and the definition of a formal language over an alphabet. We found these same concepts can be incorporated at the early stages of a digital design course. These concepts can be used to unify the coverage of all data types and present a common basis for the data presentation process. In this paper we propose a teaching method that covers the topics as an ordered sequence by starting with the concept of alphabet, a word over an alphabet, and a formal language. We then use the common features of encoding to derive a single conversion equation from a given code in the finite domain to a decimal equivalent.

The paper is organization as follows. Section 2 starts with the needed definitions of a formal language and how these definitions can be used to construct all possible words over the binary alphabet. The section includes the definition of concatenation of words and languages. Section 3 expands on the definitions applications; we progress to the different word interpretation in a digital design course where we explore the representation of the three major data types: unsigned integers, signed integers and floating-point representations. Section 4, covers reviews the IEEE 754 floating point representation. From the equations derived in sections 3 and 4, in section 5 we form a generalized conversion equation that applies all data types. The conclusion is given in section 6.

2 Alphabet, Words and Formal Languages

The definitions of an alphabet a word and formal language are found in [2, 3]. An alphabet is defined as any set of symbols, Σ. We normally work with a finite set of symbols. The cardinality of a set, Σ, is the number of elements in the set and is denoted as $|\Sigma|$. A word over the alphabet, Σ, is an ordered n-tuple $(a_1, a_2, \ldots, a_n)$ where each a_i is an alphabet symbol in Σ. The length of the word is the number of alphabet symbols in the word. We normally abbreviate the word by removing the parentheses and commas (represent as $a_1a_2\ldots a_n$). We denote the set of all words of length n as

$$\Sigma^n = \{w \mid w = a_1a_2...a_n \text{ where each } a_i \text{ is an element of } \Sigma\}.$$

The set of all words of any length over an alphabet Σ is defined as

$$\Sigma^* = \bigcup_{i=0}^{i=\infty} \Sigma^i ,$$

where ∪ represents the union of sets operator. The set Σ^0 is a special set that contains a special symbol "$\in$" (a word of length 0). The symbol is the identity symbol under concatenation. A formal language, L, over an alphabet Σ is defined as any subset of Σ^*.

The concatenation operator, ".", can be used to generate new words. The operator is defined over two alphabet symbols, two words, two languages, or any combinations of symbols words and languages. For example, let L be a formal language and "a" any alphabet symbol. The set a.L is defined as the set of all words that are prefixed with the symbol "a" followed by a word from L. For two languages L_1 and L_2 we define

$$L_1 . L_2 = \{w_1w_2 \mid w_1 \text{ is } L_1 \text{ and } w_2 \text{ is in } L_2\}$$

The concatenation operator is associative but not commutative. The operator can be used to generate all words of length n + 1 from a known set words of length n. We define

$\Sigma^{(n+1)} = \Sigma . \Sigma^n = \{a_iw_j \mid$ where a_i is an element of Σ and w_j is an element of $\Sigma^n\}$

The above can be used to determine the total number of words (all possible binary combinations) when Σ = {0, 1}. Recursively, one can show $|\Sigma^n| = |\Sigma|^n = 2^n$; the total number of possible binary combinations of length n bits. It can as well be used to construct a truth table recursively. Given the truth table for n variables, the truth table for n + 1 variables can be constructed from the truth table for n variables using the concatenation operator $\Sigma^{(n+1)} = \Sigma . \Sigma^n = \{0, 1\} . \Sigma^n$

This leads to the traditional method of generating the table, where the n+1 column (starting at n = 0) is composed of a sequence 2^n zeros followed with a sequence of 2^n ones.

The definition can also be of help in discussing the concepts of instruction sets and finite nature of the instruction sets.

3 The Three Common Data Types in Digital Design

In a typical digital design and computer organization course, the concept of binary arithmetic is considered early in the course and is covered over unsigned numbers. Additional data types and arithmetic on the types is covered later in the course. From the definition of an alphabet we can combine the discussion of the three data types and cover successively in the order of unsigned, signed, and floating point. In the discussion, we make use of the cardinality of binary words of length n as an independent parameter that limits the range of numbers under any representation (the number of words in any set of n-bit words is always 2^n). We review the three representations next.

Unsigned binary numbers: Given the n-bit binary (n-tuple word) number $x = x_{(n-1)}x_n \ldots x_0$, the decimal value of x, $dec_{unsigned}(x)$, is

$$dec_{unsigned}(x) = x_{n-1} \times 2^{n-1} + x_{n-2} \times 2^{n-2} + ... + x_1 \times 2^1 + x_0 \times 2^0 \quad (1)$$

The above function is used to generate the minimum and maximum binary numbers, minimum is a word with n zeros; a maximum is a word with n ones. Since the cardinality of Σ^n is 2^n, we conclude the maximum decimal value of n consecutive ones is $2^n - 1$. The definition also helps in the binary counting process. When counting, the first bit always alternates between zero and one. The second bit remains unchanged if the first bit value is zero. It changes (get complemented) only after the first bit is one. Similarly, the i^{th} bit remain unchanged until all less significant bits assume a value of one. This helps in the design of binary sequential counters later in the course.

Signed Numbers: Signed numbers are represented using one of the three conventions, signed magnitude, radix complement or diminished radix complement. When applied to binary numbers, radix complement and diminished radix complement become, respectively, two's (2's) and one's (1's) complement. Today's computers represent signed integers in two's complement form. Floating-point numbers are represented using signed-magnitude form. To form the 2's complement of an n-bit number, x, we form the binary subtraction $2^n - x$. Given an n-bit number x, as defined above, the decimal value of x, $dec_{twos}(x)$, can be found to be

$$\begin{aligned} dec_{twos}(x) &= (-1 \times x_{n-1} \times 2^{n-1}) + x_{n-2} \times 2^{n-2} + ... + x_1 \times 2^1 + x_0 \times 2^0 \\ &= (-1 \times x_{n-1} \times 2^{n-1}) + dec_{unsigned}(x_{n-2}x_{n-3}..x_0) \end{aligned} \quad (2)$$

The above equation and cardinality of the alphabet can be used to determine: (a) the range of positive; (b) range of negative; and (c) the minimum and maximum in each range.

For the negative part, the range of negative, from smallest to largest is determined from unsigned part (least significant n – 1 bits). A decimal value of the unsigned part of 0 results in the smallest negative value, $-2^{(n-1)}$. When the decimal part is all ones, its decimal value is $2^{(n-1)} - 1$. Hence, the largest decimal negative value is –1 corresponding to an n-bit word composed of a sequence of n ones. The range of non-negative numbers in binary and corresponding decimal value is derived similarly. The cardinality of a set is used to determine the range of the numbers.

Real numbers: Real numbers have two common representations, fixed-point and floating-point. While the set of real numbers is uncountable, the range over n-bit words in is finite and countable. Real numbers may need to represent very large numbers or very small fractions. To increase the range of values, floating-point representation is used. A real number, x, in fixed-point representation has a whole part, n bits, and fractional part, m bits, $x = x_{(n-1)} x_n \ldots x_0 . x_{-1} \ldots x_{-m}$. Similarly x in floating-point notation has an n-bit field representing the exponent part and an m-bit field representing the fractional part.

The decimal value of the fixed-point is determined using the equation

$$\begin{aligned} dec_{fixed-\mathrm{point}}(x) &= x_{n-1}\times 2^{n-1} + x_{n-2}\times 2^{n-2} + \ldots + x_1\times 2^1 + x_0\times 2^0 \\ &\quad + x_{-1}\times 2^{-1} + x_{-2}\times 2^{-2} + \ldots + x_{-m}\times 2^{-m} \\ &= \sum_{i=-m}^{i=n-1} x_i \times 2^i \end{aligned} \quad (3)$$

The range of fractional part can be computed using geometric series or by

$$\begin{aligned} dec_{fraction}(x) &= x_{-1}\times 2^{-1} + x_{-2}\times 2^{-2} + \ldots + x_{-m}\times 2^{-m} \\ &= 2^{-m}(x_{-1}\times 2^{m-1} + x_{-2}\times 2^{m-2} + \ldots + x_{-m}\times 2^0) \end{aligned} \quad (4)$$

Hence the range of the fractional part is 0 to $2^{-m}(2^m - 1)$.

The above representation and cardinality of a set of words is discussed here as well as when we look at the floating-point representation range. (Over a 5-bit word, the total number of possible words is 32. If these represent numeric data then the maximum number of data items is 32. Hence, 10100 could be a code for: the decimal value 20 (unsigned number), the decimal value –12 (2's complement), or 20/32 (fraction).)

The decimal value of a number represented in floating point form depends on the standard used. We consider this next and discuss the commonality between the different representations.

4 Floating-Point Numbers and Words Over the Binary Alphabet

When words represent floating-point numbers, the bits of a word are broken into 3 fields: a sign bit, a biased exponent field, and a fractional field. The standard floating-point representation used today is the IEEE 754 format developed around 1985. It applies to 32-bit and 64-bit representations. Earlier computers did not have a standard floating-point representation. Floating-point numbers are used to represent very large numbers as well as very small fractions.

We review the representation. Given a number of the form $N = x \times 2^y$, the number can be represented in binary using IEEE-754 32-bit standard. The representation has 3 fields: a sign bit, a fractional part (significand), F, and a biased exponent part, E. Biased exponents representation means the encoding includes a constant value added (bias) to the actual exponent. For an n-bit exponent field, the bias is 0 followed by n – 1 one bits. For the 32-bit IEEE representation (single precision) the floating-point format is shown in Figure 1.

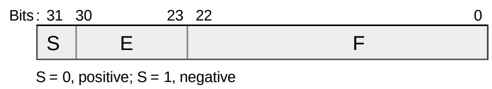


Figure 1

As can be seen from the figure, there are three fields: a 1bit sign field, an 8-bit exponent field and 23 bits fractional field. The sign bit is 0 or 1, representing positive and negative numbers. The exponent value used is biased exponent. Hence the added bias is dec(01111111) = $2^{(7-1)} - 1 = 127$.

IEEE 754 has 3 interpretations (representations) of the 32-bit word, denormalized, normalized and special cases. The interpretations are based on the binary representation of the biased exponent.

1. If $E = (00000000)_2 = (00)_{16}$ then the representation is denormalized.
2. If $E \neq (00000000)_2$ and $E \neq (11111111)_2$ then the representation is normalized.
3. If $E = (11111111)_2$ then the representation is special cases representation.

We next look at the decimal value equation for each. Denormalized word encoding: When $E = (00000000)_2$ the decimal value of the encoding is given by the equation

$$2^{1-bias} \times dec(.F) = 2^{-126} \times dec(.F) \quad (5)$$

Normalized word encoding: When $E \neq (00000000)_2$ and $E \neq (11111111)_2$ the decimal value of the encoding is given by the equation

$$2^{dec(E)-bias} \times dec(1.F) = 2^{dec(E)-126} \times dec(1.F) \quad (6)$$

Note the addition of 1 before the radix point in the fractional part.

There are two special cases corresponding to an E field composed of all 1 bits. One case represents infinite numbers. This occurs when field F is all zeros. The other case represents not a number (NaN). The NaN case occurs when E is composed of all one and the F field contains at least one non-zero bit.

The table in Figure 2 gives an illustration based on an example IEEE format but on a small word size, |E| = 2, |F| = 2. The figure includes nonnegative numbers only.

E F	E	E_adjust	F	F_adjust	Type	Decimal
00 00	0	0	0	0	denormalized	0
00 01	0	0	1/4	1/4	denormalized	1/4
00 10	0	0	2/4	2/4	denormalized	2/4
00 11	0	0	3/4	3/4	denormalized	3/4
$2^{1-bias} \times dec(.F) = 2^0 \times dec(.F), \quad bias = 2^{\lvert E\rvert-1} - 1 = 2^{2-1} - 1 = 1$						
01 00	1	0	0	1	normalized	0
01 01	1	0	1/4	5/4	normalized	5/4
01 10	1	0	2/4	6/4	normalized	6/4
01 11	1	0	3/4	7/4	normalized	7/4
10 00	2	1	0	1	normalized	2
10 01	2	1	1/4	5/4	normalized	10/4
10 10	2	1	2/4	6/4	normalized	12/4
10 11	2	1	3/4	7/4	normalized	14/4
$2^{dec(E)-bias} \times dec(1.F) = 2^{dec(E)-1} \times dec(1.F), \quad bias = 2^{\lvert E\rvert-1} - 1 = 2^{2-1} - 1 = 1$						
11 00	x	x	x	x	Special values	Infinity
11 01	x	x	x	x	Special values	NaN
11 10	x	x	x	x	Special values	NaN
11 11	x	x	x	x	Special values	NaN
Infinity: E = 11 and F = 00; NaN: E = 11 and F =01, 10, or 11						

Figure 2

5 Generating a Common Conversion Equation

Assume we are working with the set of data types over 32-bit words. Based on the discussion of alphabet and languages over the alphabet, we develop a common approach to computing the decimal value of a given binary encoding. In all the discussions we note that due to the finite cardinality, then the uncountable set of real numbers encoding function (for a given real number x, f(x) is the binary encoding of x) is such that f is not 1-to-1. Hence, the inverse of f does not exist. We show that all the computations can be written in the form

$$dec(\gamma) = k2^{\alpha} \times dec(\beta) \tag{7}$$

The proof is done by cases.

Case of unsigned numbers. Let the word encoding be $X = x_{(n-1)}x_{(n-2)} \ldots x_0$. By setting beta = X, k = 1 and alpha = 0, the proof follows.

Case of signed integers in 2's complement: Let the be $X = x_{(n-1)}x_{(n-2)} \ldots x_0$.

We know the decimal value of X is given by equation (2) as

$$\begin{aligned}
dec_{twos}(x) &= (-1 \times x_{n-1} \times 2^{n-1}) + x_{n-2} \times 2^{n-2} + \ldots + x_1 \times 2^1 + x_0 \times 2^0 \\
&= (-1 \times x_{n-1} \times 2^{n-1} + x_{n-2} \times 2^{n-2}) + x_{n-3} \times 2^{n-3} + \ldots + x_1 \times 2^1 + x_0 \times 2^0 \\
&= (x_{n-2} - 2x_{n-1}) \times 2^{n-2} + x_{n-3} \times 2^{n-3} + \ldots + x_1 \times 2^1 + x_0 \times 2^0 \\
&= dec((x_{n-2} - 2x_{n-1})x_{n-3} \ldots x_0)
\end{aligned}$$

$$\beta = (x_{n-2} - 2x_{n-1})x_{n-3} \ldots x_0$$

The proof follows for k = 1 and alpha = 0.

Case of real numbers in fixed-point notation: Let $x = x_{(n-1)}x_n \ldots x_0 . x_{-1} \ldots x_{-m}$. From equations 3 and 4 we have

$$dec_{fixed-point}(x) = \sum_{i=-m}^{i=n-1} x_i \times 2^i = 2^{-m}\left(\sum_{i=-m}^{i=n-1} x_i \times 2^{i+m}\right)$$

$$= 2^{-m} \times dec(x_{(n-1)}x_{(n-2)} \ldots x_0 x_{-1} \ldots x_{-m})$$

Note that $\beta = x_{(n-1)}x_{(n-2)} \ldots x_0 x_{-1} \ldots x_{-m}$ represents an n + m unsigned binary integer. The above equation is satisfied for $k = 1, \alpha = -m$.

Case of real numbers in floating-point notation, denormalized form. For this case, the encoding is broken into two parts, the biased exponent part and the fractional part. Let $E = E_{(m-1)} E_{(m-2)} \ldots E_0$ and $F = F_{(n-1)} F_{(n-2)} \ldots F_0$ correspond to the exponent and fractional parts, respectively. The number X is represented as EF. Using equation (5) for the denormalized part we have

$$\begin{aligned}
dec(x) &= 2^{1-bias} \times dec(.F) = 2^{1-(2^{\lvert E\rvert-1}-1)} \times dec(.F) \\
&= 2^{1-(2^{\lvert E\rvert-1}-1)} \times 2^{-\lvert F\rvert} dec(F) = 2^{1-(2^{\lvert E\rvert-1}-1)} \times 2^{-\lvert F\rvert} dec(F) \\
&= 2^{1-(bias+\lvert F\rvert)} dec(F) = 2^{2-(\lvert F\rvert+2^{\lvert E\rvert-1})} dec(F)
\end{aligned}$$

The definition is satisfied for $\beta = F, k = 1, \alpha = 2 - (\lvert F\rvert + 2^{\lvert E\rvert-1})$. Note that F is represented as an unsigned integer.

Case of real numbers in floating-point notation, normalized form: This represents the final case. From equation 6 we have

$$\begin{aligned}
dec(X) &= 2^{dec(E)-bias} \times dec(1.F) \\
&= 2^{dec(E)-bias} \times 2^{-\lvert F\rvert} dec(1F) \\
&= 2^{-(bias+\lvert F\rvert)} \times 2^{dec(E)} \times dec(1F)
\end{aligned}$$

The equation can be satisfied by assignment

$$k = 2^{(bias+\lvert F\rvert)}, \alpha = dec(E), \beta = 1F$$

Before we leave the discussion, we emphasize to the students that the under the same representation, unlike mapping decimal to binary, each of the functions above forms a 1-to-1 correspondence; i.e., two different encodings result in two different images (decimal values). We also emphasize that depending on the nature of encoding different binary codes may have the same decimal value. For example the fixed-point word 100.01 has a decimal value 4.25. Similarly and the floating-point word 1010001 (|E| =3, |F| = 4) shas the decimal value 4.25. This can be verified using the equation for the normalized floating-point representation[1].

$$dec(10100001) = 2^{-(bias+|F|)} \times 2^{dec(E)} \times dec(1F)$$

$$= 2^{-(3+4)} \times 2^5 \times dec(10001) = \frac{17}{4} = 4.25$$

6 Conclusion

In this paper we have introduced an alternative approach to teaching data representations in a digital design course. We have incorporated the use formal languages. In addition we have introduced a new general conversion equation. By proper parameter substitution, the equation can be in conversions given the common binary number encodings, unsigned integers, signed integers, fixed-point real numbers, and floating-point representations. For the floating-point encoding, we have incorporated the IEEE 754 standard.

References

[1] Weste N. and Karman E. (1993). *Principles of CMOS VLSI design A Systems Perspective 2nd edition.* Addison Wesley.

[2] Martin J. (1991). *Introduction to Languages and the Theory of Computation.* McGraw Hill.

[3] Barett W., Bates R., Gustafson D. Couch J. (1979). *Compiler Construction Theory and Practice, 2nd edition.* SRA publishing.

[4] Nelson V., Nagle H., Carroll B., Irwin J. (995). *Digital logic Circuit Analysis and Design.* Prentice Hall.

[5] Katz, R. and Gaetano B. (2005). *Contemporary Logic Design, 2nd edition*. Prentice Hall.

[6] Mano M., Kime C. (2003). *Logic and Computer Design Fundamentals, 3rd Edition*. Prentice Hall.

[7] Gajski D. (1997). *Principle of Digital Design*, PrenticeHall, 1997

[1] The conversion of a base 10 number to floating follows the following steps: (a) find the binary value of the number, (b) write as 2^e x 1.F, (c) add the biased binary exponent to e to form E, (d) the floating point number is represented as EF, with least significant bits of F filled with 0 and the sign bit (MSB) set to 0 or 1

Terminal efficiency analysis of a postgraduate degree, under the theory of hierarchical distances geometric structure

Casanova del Angel, F[1] and Rosas Sánchez, M. E[2]
1: SEPI ESIA, ALM Unit, IPN. Juan de Dios Bátiz S/N Edif. 10 Unidad ALM del IPN 07738 fcasanova@ipn.mx
2: ESIA Licentiate Degree, ALM Unit, IPN. Juan de Dios Bátiz S/N Edif. 10 Unidad ALM del IPN 07738 mrosass@ipn.mx

Abstract - *The demonstration of the main theorem of hierarchical distances geometric structure theory, which depends on a minimal distances criterion, from which a better reading of dendrograms resulting in better conclusions is demonstrated. The property of axial symmetry of the generating curve proves the affinity of triangles created by such, with state zero. Characterizing the generating curve as a countable sequence of sets, allows us to prove the existence of the transformation to linear affinity of such. A multivariate analysis of terminal efficiency of postgraduate degrees at ESIA, IPN Mexico, along its first 48 years of academic and scientific life and the development of students' graduation is presented.*

Keywords: geometrical structure, hierarchical distances, terminal efficiency, postgraduate degree and computer Engineering

1.. Introduction to the history of the postgraduate degree

On December 14, 1961, the General Director of Instituto Politécnico Nacional, IPN, Eugenio Méndez Docurro, presented before the School Consultive Techcnical Committee the advantages and rationale to establish the Graduate Section of the Higher School of Engineering and Architecture, ESIA, IPN. He based such presentation on two approaches: the capacity to establish postgraduate degrees at the ESIA and the type of courses, the specialties and modalities of each [1]. Based on such institutional decision, one of the oldest Sections of Graduates of the IPN was formally brought to life.

On July 16, 1962, registration into postgraduate degrees at ESIA, IPN, were formally started, with two master degrees in sciences, with specialty in structures and hydraulics [6]. Courses formally started on Wednesday, August 1, 1962. The syllabi of both master degrees were organized in half-year periods. On December 14, 1966, the School Consulter Technical Committee of the IPN discussed, in the fourth point of its agenda, the creation of six master degrees and a doctorate degree in sciences, among which there was a new master degree in sciences in planning. In 1978, an unprecedented expansion was intended, based on a proposal to create a master degree in sciences for every civil engineering specialty. Despite the boldness of such proposal, four new master degrees in sciences were established: architecture, environmental engineering, geology and soil mechanics. In 1981, mining and oil specialties for the master degree in geology, and architecture and architecture specialty with options for architectonic design and works construction and control were created. In 1983, the structural analysis, steel structures, cement structures, architecture and ports development specialties, for the master degree in hydraulics [4] were created. In 1981, under an agreement with Colegio de Ingenieros Petroleros, Petróleos Mexicanos and Instituto Mexicano del Petróleo, hydrocarbons administration, planning and economy specialty courses were taught.

In 1969, a four-year agreement between IPN and UNESCO was established, and the UNESCO-IPN Planning Center was created, where development plans were generated. In 1986, the master degree in architectural sciences became part of the Tecamachalco Unit, creating its own Section of Graduates. In 1997, the geology, administration, hydrocarbons planning and economy master degrees were relocated in the Ticomán Unit of ESIA, thus creating a new Section of

Postgraduate Studies and Research (SEPI) [5]. On June, 1998, through the execution of a General Academic Collaboration Agreement with the Universidad Politécnica de Madrid, a joint civil engineering doctorate degree with specialty in environmental hydraulics was established. Such doctorate degree was taught at SEPI ESIA, ALM Unit, IPN, with the support of the Civil Engineering –hydraulics and energy– Department of the UPM.

2. Terminal efficiency

Currently, one of the criteria used to assess the functioning of academic and research activities is terminal efficiency, as one of the main indicators showing the achievements of the corresponding education institution. Since the ESIA, ALM Unit, IPN is one of the schools of civil engineering with more students in Mexico, it is very important to know its terminal efficiency, both for licentiate and postgraduate degrees. On the top of the table, the number of graduates for each master degree in sciences officially known up to 2007, throughout 48 years, is shown. On the bottom of the table 1, terminal efficiency of the master degree in civil engineering up to date, which substituted the five previous ones in 2007, is: Structures shown.

Table 1. Historical terminal efficiency of master degrees in sciences.

Master degree in sciences	**Number of graduated students**	**Period**	**Year of defense of the first specialty thesis**	**Terminal efficiency annual index**
Structures	58	1962-2010	1970	1.20
Hydraulics	51	1962-2010	1975	1.06
Planning	66	1966-2010	1977	1.37
Soil mechanics	32	1981-2010	1987	0.66
Environmental engineering	114	1977-2010	1979	2.37
Doctorate degree				
Environmental hydraulics	5	1998-2010	2000	0.50
Master degree in engineering				
Structures	3	2009-2010	2010	1.5
Hydraulics	5	2009-2010	2010	2.5
Planning	5	2009-2010	2010	2.5
Geotechnics	0	2009-2010	-	0
Environmental engineering	10	2009-2010	2010	5

Source: Record of thesis recorded in the library of the SEPI ESIA, ALM Unit, IPN.

Correlation of terminal efficiency

The calculation of correlations or *degree of association* among variables was carried out from usual Euclidian distance d(j, j') among variables j and j'; that is: $d^2(j, j') = \Sigma^n_{i=1} x^2_{ij} + \Sigma^n_{i=1} x^2_{ij'} - 2 \Sigma^n_{i=1} x_{ij} x_{ij'}$. Since general terms of normed analysis in general terms in real space of dimension p, $\Re^p$, are points x_{ij} we have that: $\Sigma^n_{i=1} v^2_{ij} = \Sigma^n_{i=1} x^2_{ij'} = 1$. Every point-variable is on a sphere with radius 1 and center on the origin of main axes, which the correlation coefficient $c_{jj'}$ among variables j and j' is: $\Sigma^n_{i=1} x_{ij} x_{ij'} = c_{jj'}$.

Best correlated master degrees in sciences are: environmental engineering planning/ hydraulics, and structures/hydraulics, Figure 1. It must be remembered that, if two meteorological variables are *strongly correlated*, they are near from each other ($c_{jj'}$ = 1) or, on the contrary, as far from each other as possible ($c_{jj'}$ = -1), in accordance with linear relationship linking them is direct or inverse, and that when $c_{jj'}$ = 0 they are considered at an average distance or that variables j and j' are orthogonal.

Table 2. Correlations matrix.

	1	2	3	4	5	6	7
1	1.000						
2	0.652	1.000					
3	0.377	0.554	1.000				
4	0.564	0.656	0.695	1.000			
5	0.630	0.502	0.431	0.535	1.000		
6	-0.229	-0.018	-0.044	-0.035	-0.60	1.000	
7	-0.159	-0.098	-0.202	-0.178	-0.086	0.526	1.000

Where 1: Structures, 2: Hydraulics, 3: Planning, 4: Environmental engineering, 5: Soil mechanics, 6: Hydrocarbons administration and 7: Geotechnics.

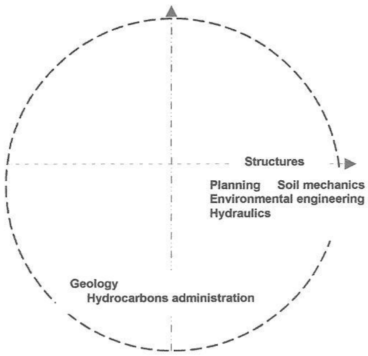


Figure 1. Circle of CP1 *vs* CP2 correlations

Best correlated master degrees in sciences are: environmental engineering planning/ hydraulics, and structures/hydraulics, Figure 1. It must be remembered that, if two meteorological variables are *strongly correlated*, they are near from each other ($c_{jj'}$ = 1) or, on the contrary, as far from each other as possible ($c_{jj'}$ = -1), in accordance with linear relationship linking them is direct or inverse, and that when $c_{jj'}$ = 0 they are considered at an average distance or that variables *j* and *j*' are orthogonal.

3. Analysis of main components of gross data

Factorial method chosen to describe data under study is the main component analysis, ACP, since it allows the direct search of simultaneous representation of sets under study *I* years of graduation and *J* master degrees in sciences [2]. The ACP applied on gross data K_{IJ} has the following factorial characteristics: variances on the first five main axes or own values are: X_1 = 3.3186, X_2 = 1.4787, X_3 = 0.7817, X_4 = 0.4838 y X_5 = 0.4223, while the inertia percentages explained by such axes are, respectively: 47.4%, 21.1%, 11.2%, 6.9%, and 6.0%. Main components are well defined. The first includes master degrees in sciences in environmental engineering, hydraulics and structures. The second main component includes the master degrees in science that have not belonged to this SEPI for a long time, hydrocarbons administration and economy and geology; while the third component is planning.

3.1. Hierarchical dendrogram of terminal efficiency

From a theoretical point of view, it may be considered that the theoretical development of hierarchical classification presents the problem of choosing classes, when at least to subclass pairs show the case of a minimal equation in distance δ on Ver[C_h], which is present when data is very fragmented. The traditional way to solve this has been to arbitrarily solve the pair of subclasses to be added, the first to be read. The right way, which mathematical structure depends on a minimal distances criterion, is supported with the demonstration of the theorem of *geometrical hierarchical distances* [3].

Theorem of geometric hierarchical distances. Given a partial hierarchical sequence of $C(\alpha)$, if there are two partial hierarchies to be added in the following choosing of classes $C_h(\alpha)$ y $C_k(\alpha)$ such as $C_h(\alpha) = C_k(\alpha)\ \forall\ h \neq k$, which represent the same minimum distance: $\delta(\{x_h\}, \{x_{h'}\}) = \delta(\{x_k\}, \{x_{k'}\})$, regarding class $C_r(\alpha)$, the partial hierarchy to be added to this last one depends on the geometric δ distances presented by classes $C_h(\alpha)$ and $C_k(\alpha)$ regarding C_r.

Demonstration. Let $C(\alpha) = C_0, C_1, ..., C_h, ..., C_{\alpha-1}$ be a partial hierarchy sequence. If there are two hierarchies $C_h(\alpha)$ and $C_k(\alpha)$ such as: $C_h(\alpha) = C_k(\alpha)\ \ \forall\ \ h \neq k$, which comply with equality of distance: $\delta^{h-1}(\{x_h\},\{x_{h'}\}) = \delta^{k-}$

$^{1}(\{x_k\},\{x_{k'}\})$, where $Ver[C_h] = Ver[C_k]$ so that the level indexes of classes are the same, that is, $\nu(C_h) = \nu(C_k)$, thus:

$$\inf\{\delta^{h-1}(C_h)\} = \inf\{\delta^{k-1}(C_k)\} \quad \text{and} \quad \inf\{\delta^{h-1}(\{x_h\}, \{x_{h'}\})\} = \inf\{\delta^{k-1}(\{x_k\}, \{x_{k'}\})\}$$

with x_h, $x_{h'}$ first-born elements and x_k, $x_{k'}$ benjamin elements of the corresponding classes C_h and C_k. Let us consider a third partial hierarchy $C_r(\alpha) \subset C(\alpha)$ and $r < h$, $r < k$, to which will be added one of the two partial hierarchies h or k order, using properties of ultra-metric distance. The geometric shapes those hierarchies may create are: equilateral, isoscelic and scalenic, just talks about triangular equilateral and isoscelic relationships, not about scalene relationships]. In accordance with the geometry made up by hierarchical classes $C_r(\alpha)$, $C_h(\alpha)$ and $C_k(\alpha)$:

$$\delta^{r-1}(C_r, C_h) \leq \sup\{\delta^{r-1}(C_r, C_k), \delta^{r-1}(C_k, C_h)\} \text{ y } \delta^{r-1}(C_r, C_k) \leq \sup\{\delta^{r-1}(C_r, C_h), \delta^{r-1}(C_h, C_k)\}$$

with the following inequalities:

$$\delta^{r-1}(C_r, C_h) \leq \delta^{r-1}(C_r, C_k) + \delta^{r-1}(C_k, C_h) \quad \text{y} \quad \delta^{r-1}(C_r, C_k) \leq \delta^{r-1}(C_r, C_h) + \delta^{r-1}(C_h, C_k)$$

which means it does not matter if the triangle generated by hierarchical classes $C_r(\alpha)$, $C_h(\alpha)$ y $C_k(\alpha)$ is equilateral or isosceles, arbitrarily choosing the partial hierarchy to be added. Now, if the triangle relationship is of the scalene type, one of the two distances to *r* class is smaller; that is, if $\delta^{r-1}(C_r, C_k) < \delta^{r-1}(C_r, C_h)$, which is easier to prove, as the inequality properties of the triangle

$$\delta^{r-1}(C_r, C_k) + \delta^{r-1}(C_k, C_h) < \delta^{r-1}(C_r, C_h) + \delta^{r-1}(C_k, C_h)$$

QED

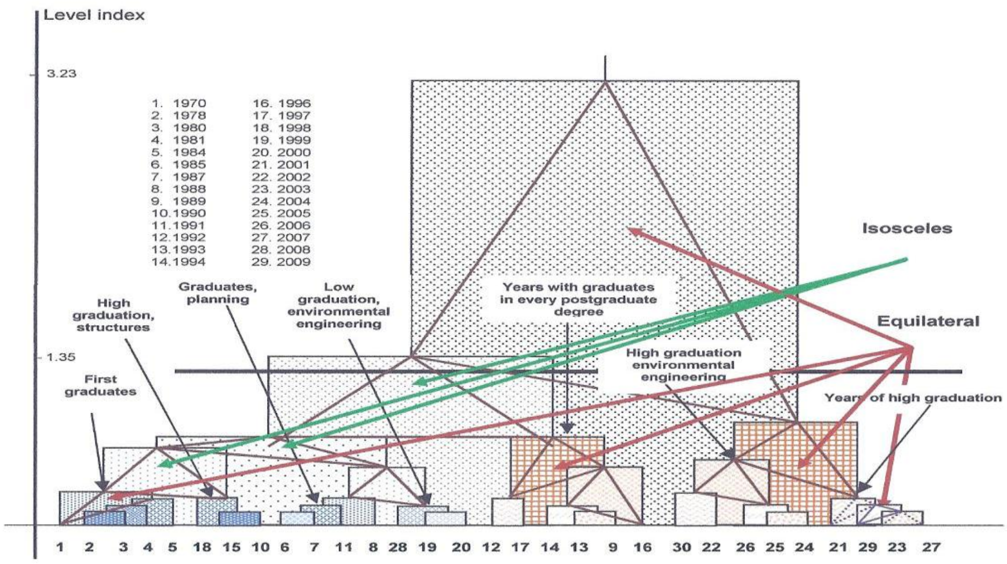


Figure 2. Hierarchical classification of terminal efficiency of postgraduate degree SEPI ESIA.

Figure 2, shows the hierarchy of relationship between the years of graduation of master degrees in sciences. Reading and interpretation is based on the value of hierarchical level index, shown on the left of the dendrogram, such being understood as the consecutive order of values from the product of the weight of the class under analysis and its diameter (distance $d(i, i')$ is the diameter of the smallest part of a hierarchy containing both *i* and a *i´*) [3]. The hierarchical dendrogram built is formed by three branches, which interpretation is absolutely congruent with knowledge on the topic. Here, the geometrical structure of the hierarchy is shown only in an equilateral and isosceles way. Isosceles geometry groups the phases of the dendrogram: high and low graduation. Equilateral geometry groups the phases making the difference between master degrees with most graduates.

4. Conclusions

From the theoretical point of view, it may be mentioned that the theoretical development of hierarchical classifications present the problem of electing classes when at least two pairs of subclasses present the case of a minimum equality in distance δ on Ver[C_h], which case is present when data is highly fragmented. The traditional way to solve this has been to choose arbitrarily the pair of subclasses to be added, the first one to be read. The right way to do it, whose mathematic structure depends on a minimum distance criterion, has been demonstrated based on the theorem of *geometric hierarchical distances.*

The property of axial symmetry of the generating curve proves the affinity of triangles created by such, with state zero. Characterizing the generating curve as a countable sequence of sets, allows us to prove the existence of the transformation to linear affinity of such.

To achieve the optimal terminal efficiency of the Section of Postgraduate Degrees, a real connection between professor and student must be fostered, in order that information moves in both ways, since a lot of students, along their lives, carry out professional practice highly contributing to the technological and scientific progress, which, together with professors as knowledge guides, may yield significant progress. As a result of the analysis carried out in this work, it must be noticed that one of the areas of knowledge of the postgraduate degree students are more interested in are environmental engineering and planning, offering the highest number of graduates in such specialties.

Acknowledge

This document was developed with part of the time devoted to the IPN-SIP 20100674 and IPN-SIP 20110437 research projects.

5. References

[1] Acta de la sesión celebrada el 14 de diciembre de 1961. Consejo Técnico Consultivo Escolar. Dirección General. Instituto Politécnico Nacional.

[2] Casanova del Angel, F. **2001**. *Análisis Multidimensional de Datos*. Editorial Logiciels. México.

[3] Casanova del Angel, F. **2010**. Structure géométrique des distances hiérarchiques. No. 41, pp. 27-47. La Revue MODULAD. ISSN: 1769-7387. www.modulad.fr.

[4] Catálogo de los estudios de posgrado ofrecidos en el IPN. 1984. Dirección de Graduados e Investigación. Instituto Politécnico Nacional.

[5] Gaceta Politécnica. **1991**. Órgano Oficial de Información del Instituto Politécnico Nacional. Publicación mensual, año XVII, número 317, mayo de 1991.

[6] Programa de estudios de posgrado *ESIA*. 1962. Finalidades, Objetivos y Justificaciones. Sección de Graduados ESIA. Editorial Politécnica. Agosto de 1962. México.

Discrete Mathematics as A Transitional Course

Chung-Chih Li[1], Kishan Mehrotra[2], and Chu Jong[1]
[1]School of Information Technology, Illinois State University, Normal, IL 61790, USA
[2]Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY 13244, USA

Abstract— *We argue that the first course of Discrete Mathematics in Computer Science curriculum should be seen as a transitional course to prepare students with necessary skills for rigorous mathematical arguments in their future study. We should emphasize improving mathematical skill over increasing advanced topics and Discrete Mathematics at this level should be taught as the first course together with CS1. In this regards, Discrete Mathematics textbooks available in the current market do not serve this purpose.*

Keywords: Discrete Mathematics

1. Introduction

There is no doubt that formal mathematical reasoning and arguments are indispensable in computer science (CS) [1] [2]. The ACM Computer Science Curriculum 2001 Report (CC2001) [3] and its 2005 and 2008 revisions [4] [5] considered discrete structures as foundational material for CS. The report stated:

> "*Computer science depends on mathematics for many of its fundamental definitions, axioms, theorems, and proof techniques. In addition, mathematics provides a language for working with ideas relevant to computer science, specific tools for analysis and verification, and a theoretical framework for understanding important computing ideas.*" ([3], page 40)

Six core areas were identified for the Discrete Mathematics (DM) course to cover, including basic materials such as functions, relations, sets, logics, and proof techniques. However, our impression from reading CC2001 is that, it is topics oriented, instead of skills oriented. Namely, it emphasizes understanding concepts over improving mathematical skills. We have a similar impression on most DM textbooks that emphasize introducing new concepts over practicing formal mathematical arguments using basic concepts.

1.1 Mathematical skills verse topics

We observed that most CS freshmen already acquired many nontrivial mathematical concepts such as algebra, geometry, calculus, and so on in their mathematical background. However, they were rather weak in formal mathematical reasoning and arguments. Unfortunately, as the field of CS expands so rapidly, CS educators never found enough time for them to teach new and important concepts to students. They would very hesitate to take time on simple concepts and slowly polish students mathematical arguments. But in the long run, it was easy to see how solid mathematical foundation could benefit students in advanced classes. We therefore argue that the one-semester DM course is necessary and it should emphasize improving mathematical skills over increasing new topics.

Consider the following problem we gave in the first class of the Theory of Computation (TOC) in order to assess students' mathematical skills. This was not merely an anecdotal example; one can easily find the same situation in almost every advanced CS class[6].

- Let A and B be sets. Prove that, $(A \cup B) \subseteq (A \cap B)$ if and only if $A = B$.

All students in the TOC class had taken DM as one of the prerequisites and most of them were senior students who have learned most advanced topics. But surprisingly, very few students could give satisfactory proofs for this almost trivial question. A typical mistake was to prove by showing some examples or drawing Venn Diagrams. Students didn't know how to treat "if and only if" in the theorem and didn't have the ability to construct abstract and formal mathematical arguments independent from specific examples. However, students' proofs did show that they had the concepts of sets and their operations and relations such as $\cup$, $\cap$, $\subseteq$, and $\in$. But having those basic concepts was not the key point of the problem. Our analogy is this: mathematical concepts are vocabularies of mathematics; but vocabularies alone do not make a language, we also need grammars. It seems to us that the DM course is the only course in CS curriculum where we can intensively teach student mathematics grammars. The assessment results showed that the DM course did not serve its purpose at all.

2. Common Approaches to Improving Discrete Mathematics Teaching and Learning

As the student's frustration built up, one could easily find a vast volume of literature on how to improve the performance in the classroom of DM. The dimension of the literature was very broad ranging from when and how to teach to what to teach [7]. However, we did not find any arguments emphasizing on training students' mathematical skills. In this section, we point out some common "wisdoms" that have failed in fixing the problem we just mentioned.

2.1 More applications?

There were many discussions around the problem that some students may not see the linkage between DM and their major [8]. Many articles have suggested that we should connect the topics in DM to the real-world CS problems in such a way students may be motivated to learn the topics [9]. However, motivation was doomed by frustration, not merely by the disconnection from the real world. Calculus may be even less useful in a programming-centered IT career, but students were more willing to learn with an attitude that "educated people should know something about it" [10]. We should cultivate students to view DM from an intellectual and scientific perspectives.

Secondly, given the time constraints, it is not practical to introduce any nontrivial applications in depth. On the other hand, students may also react skeptically to trivial applications that do not spell out the significance of this course. What is more important in the DM course is to prepare students for serious applications in their future study.

2.2 More programming?

More and more CS educators have been trying to increase programming works in the DM course (e.g.,[11], [12]). As CS researchers and educators, we certainly do not oppose programming, but we have a different agenda in the DM course. The more-programming approach was based on the following assertions: 1. Students can better learn abstract concepts through coding the concepts, and 2. Self-satisfaction motivates leaning and CS students in general are good or need to be good at programming.

For assertion 1, we would question: "What exactly is the concept that students can't understand until they complete some programs to witness the concept?" Consider the problem in the previous section again. There is no need for students to write any programs in order to better understand the concepts of sets, union, intersection, and so on. CS is not an experimental science. At this level, writing programs to test the theorem does not help students to construct a mathematical proof for the theorem. We believe that most of the students in our TOC class already have sufficient programming skills to implement those set operations, but such skills have nothing to do with the ability to construct a simple mathematical proof. For assertion 2, we must agree that self-satisfaction is an important factor in learning process, but self-satisfaction not necessarily facilitates effective learning. McMaster et al [11] let programming assignments play a "substantial portion" in students' grades. But if a student is satisfied with some irrelevant achievement, the student may be heading a wrong direction without knowing it because he/she is wrongly rewarded.

2.3 More topics?

CC2001 [3] identified six core areas in discrete structures:

DS1: Functions, Relations, and Sets
DS2: Basic Logic
DS3: Proof Techniques
DS4: Basic Counting
DS5: Graphs and Trees
DS6: Discrete Probability

All topics certainly have deep applications in computer science, and two semesters are needed to cover all these topics. However, it is not practical for many CS programs to has two DM courses. Therefore, the SIGCSE board approved its first sub-committee in 2003, *The committee on the Implementation of a Discrete Mathematics Course*. The charge to the sub-committee was "to work toward providing a few (≤ 6) practical models for a one-semester course that will meet the basic needs of undergraduates in a computer science Program". The committee's final report was published in 2007 [13]. In addition to DS1$\sim$3, the course outline suggested by the committee covered DS4 and DS5. However, we don't think they are necessary to be covered if we want to taught DM as the first course in CS program (section 3.4 and 3.2).

3. Our Pedagogical Philosophy To Discrete Mathematics

In the previous section, we have described some questions raised from the past efforts in improving the DM course. It seems to us that some efforts are irrelevant to the current problem and some approaches in fact compromise the integrity of the course. But what should we do otherwise? What do we expect from this course? In this section we try to provide our answers.

3.1 The purpose of Discrete Mathematics

Here is our proposition:

- Discrete Mathematics is an intermediate course to provide necessary mathematical *skills* for students to *transit* to more advanced studies.

We stress the words "*skills*" and "*transit*" in our statement. In other words, we want to transfer students from ones who understand mathematics with informal intuition and common senses to ones who can present mathematical and abstract objects in terms of precise definitions and express their relations in rigorous arguments. Thus, we consider the skills of managing mathematical arguments is the "principal" we want students to have after the course. Such skills are extremely important in our discipline. Even in writing an operational manual, for example, the author needs to define terminologies, domains of applications, and to organize logical sequences of operations and instructions for users.

3.2 Topics selection

As mentioned earlier, the most important objective of the DM course is to let students be familiar with the structure of rigorous mathematical arguments. Complicated concepts, however important to the discipline, are not suitable in this course because their complexity may distract the studentąŕs attention from the underlying structure of the arguments. We should use primitive concepts such as naive set theory, basic logical rules, easy discrete functions and relations, and so on as tools, not as targets, to train students for rigorous mathematical arguments and formal reasoning. Thus, topics in DS1$\sim$3 should be covered in the DM course. They are selected because their basic concepts are straightforward. Except Mathematical/Structural inductions, students already have most of the basic concepts before the class. Students don't have to spend too much time to understand the concepts before they can use them in their practices.

We suggest the topics to be taught in the following order: Sets, Logic, Mathematical/Structural Inductions, Relations, and Functions. For each topic, we start with a brief introduction that should cover all necessary concepts both in intuitive ways and precise definitions. Only one or two hours are needed to explain the basic concepts for each topic. Then, we should use the rest of the class time to practice how to use those terminologies, definitions, axioms, and assumptions in mathematical proofs and arguments.

Although logic is the backbone of any rigorous arguments to begin with, the topic will get too involved quickly as we proceed. Thus, we align with the approach that the naive set theory should come first [14]. Mathematical inductions are covered after logic as an application of Modus Ponens and logical inferences. The concept of relations is an extension of the concept of sets with extra properties. Likewise, the concept of functions is an extension of the concept of relations. In addition to mathematical/structural induction proofs, we have to explicitly emphasize on the patterns of mathematical proofs throughout the semester, including contradiction, counterexamples, contrapositive, constructive verse nonconstructive proofs, and so on. If students are well trained in DM, their mathematical skills should equip them to easily understand the presentations and techniques used in advanced subjects.

3.3 Mathematics as musical instruments

Practicing is the key to mastering. It is our belief that doing mathematics is just like playing musical instruments. In order to appreciate the music, one must at first listen to the music again and again. Consequently, if one wants to play and further master the musical instrument, one must practice repeatedly from the most basic techniques. Given this approach that has been widely accepted in music education, here we advocate to adopt its mathematical counterpart in CS education. It is clear that mathematical skills are developed not only by listening to the instructor in the classroom and reading textbooks, but also by repeatedly practicing easy arguments until one can use them as second nature.

Intuitive understanding will not carry students to advanced studies, because students can easily lose their intuition in complex proofs. However, any complex proof can be decomposed into small pieces and each of them is a simple application of some easy techniques. For example, many students considered using the pumping lemma to disprove some languages being regular as one of the most difficult proofs in TOC. (We restrict our discussion to regular languages for simplicity.) We observed that students in TOC had little or no problem at all to understand the lemma itself, although it was a new and nontrivial concept to them. But only a very small portion of students could fully understand the logic behind its application. We believe that if students were well trained in the DM course, they should be able to formulate the lemma and work on its negation by themselves for a proof by contradiction as follows. Let $P(L)$ be the predicate as follows.

$$P(L) : \text{L is a regular language.}$$

Define $Q(L)$ as follows.

$$\begin{aligned} Q(L) : \exists m \forall \omega (|\omega| \geq m \to \\ &\exists x \exists y \exists x ((\omega = xyz) \wedge (|xy| \leq m) \\ &\qquad \wedge (|y| > 0) \wedge \forall i (xy^i z \in L))). \end{aligned}$$

Thus, the pumping lemma states: $P(L) \to Q(L)$. By contrapositive, we have

$$\neg Q(L) \to \neg P(L).$$

To prove $\neg P(L)$, we need to argue that $\neg Q(L)$ is true. Using the following equivalences:

$$\begin{array}{rcl} \neg \forall x p & \equiv & \exists x \neg p \\ \neg \exists x p & \equiv & \forall x \neg p \\ p \to q & \equiv & \neg q \to \neg p \\ \neg (p \to q) & \equiv & p \wedge \neg q \\ \neg (p \wedge q) & \equiv & \neg p \vee \neg q \end{array}$$

we can formulate $\neg Q(L)$ as follows:

$$\begin{aligned} \neg Q(L) : \forall m \exists \omega ((|\omega| \geq m) \wedge \\ &\forall x \forall y \forall x (\neg (\omega = xyz) \vee \neg (|xy| \leq m) \\ &\qquad \vee \neg (|y| > 0) \vee \exists i (xy^i z \notin L))). \end{aligned}$$

The real difficult part is finding an ω and right i, but this is not the center of students confusion. Instead, students are confused by the logical meaning of $\forall x \forall y \forall x$ in a mathematical statement. The same confusion occurs in the converses of pumping lemmas, which is incorrect [15]. We believe that the lack of practicing in the DM course is the reason why students struggled in advanced courses.

3.4 Discrete Mathematics as the first course

We propose that the DM course should be taken in the first semester together with CS1 right after students are admitted to the CS program. Most CS programs make DM as a prerequisite to Algorithms and TOC. A common model is to schedule the DM course in the second year when students have taken CS1 and CS2, and probably are taking Data

Structures and DM at the same time. DM is not considered as a prerequisite to CS1, CS2, and Data Structures. This is justifiable only if the course emphasizes on the topics listed in DS4∼6 and the topics listed in DS1∼3 must be skipped or briefly skimmed. This model will make new and complicated topic to be the center of the DM course.

As we advocate in subsection 3.2 that the concepts used in the DM course should be easy enough for students to focus on the structure of arguments, we also think that reviewing those basic concepts and making them precise for students can benefit other courses including CS1 and CS2. In introductory programming courses, we observed that illogical thinking would result in frustration, poor programming, and mindlessly revising incorrect code without logical reasoning why the code was incorrect.

3.5 Rewrite textbooks in concise styles

There have been quite a lot of good textbooks for DM and we have to admit that we haven't seen a bad one for teachers; but, unfortunately, neither have we seen any good one for novices in mathematics. No matter how excellent we might think our choice was, we received constant complaints from students about the textbooks. All textbooks in our list have one common nature: they are very wordy. English is needed to explain concepts in intuition but the concepts of the subjects at this level in DM are almost trivial. We therefore believe that we should minimize the role of English and let students try to comprehend the subjects and theorems in terms of mathematical languages, i.e., definitions, axioms, assumptions, inference rules, deductions, logical consequences, and so on. Here we highlight some key features we are looking for in an ideal textbook for the DM course. To our knowledge, there is no DM textbook that meets the key features.

1) Basic and intuitive concepts should not be unnecessarily explained in lengthy discussion. Instead, students should re-learn those concepts in terms of precise mathematical definitions and use them to express those known properties and operations. For example, the entire chapter of Sets should be based on a few definitions given at the beginning of the chapter.
2) Any proof, however trivial, should not be omitted, e.g., $A \subseteq A \cup B$. The purpose is to let students be familiar with mathematical arguments, not to convince students that the theorem is correct.
3) In proofs, every step should be explained by explicitly pointing out which definitions, previous proven theorems, or logical rules are used. Phrases like "it is clear that" in most cases will just leave students wondering in the dark.
4) The textbook should provide enough problems and solutions for practicing.

4. Conclusions

Like many educators, we are aware of the chronicle frustration in teaching DM and the awful gap between our expectation and actual outcomes of the course. We decided to re-examine the fundamental purpose of the course and how it is implemented in CS undergraduate programs. While the research in teaching DM is not new to CS education community, here we take a different approach in hoping to fix the problem from the bottom. We give up complex subjects and make DM a truly transitional course. We believe that students should build up their mathematical skills first before they encounter more advanced topics. We hope that our speculation can help us reach some consensus and lead to a more consistent philosophy in the syllabi of the DM course.

References

[1] D. Baldwin, C. H. Smith, P. B. Hendersen, and V. Vadisigi, "CS1 and CS2 (panel session): foundations of computer science and discrete mathematics," in *Proceedings of the thirty-first SIGCSE technical symposium on Computer science education*, ser. SIGCSE '00. New York, NY, USA: ACM, 2000, pp. 397–398. [Online]. Available: http://doi.acm.org/10.1145/330908.331893

[2] P. B. Henderson, "Mathematics in the curricula," *SIGCSE Bull.*, vol. 37, no. 2, pp. 20–22, June 2005. [Online]. Available: http://doi.acm.org/10.1145/1083431.1083449

[3] ACM/IEEE-CS, The Joint Task Force on Computing Curricula, "Computing Curricula 2001, Computer Science, Final Report," December 2001.

[4] ACM/AIS/IEEE-CS, Joint Task Force on Computing Curricula 2005, "Computing Curricula 2005, The Overview Report," September 2005.

[5] ACM/IEEE-CS, The Interim Review Task Force, "Computer Science Curriculum 2008: An Interim Revision of CS 2001," December 2008.

[6] J. Krone, D. Baldwin, J. C. Carver, J. E. Hollingsworth, A. Kumar, and M. Sitaraman, "Teaching mathematical reasoning across the curriculum," in *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, ser. SIGCSE '12. New York, NY, USA: ACM, 2012, pp. 241–242. [Online]. Available: http://doi.acm.org/10.1145/2157136.2157208

[7] T. Whelan, S. Bergin, and J. F. Power, "Teaching discrete structures: A systematic review of the literature," in *SIGCSE*, Dallas, Texas, USA, 2011, pp. 275–280.

[8] R. Bravaco, B. Kim, M. B. Gousie, and L. Wilkens, "Math connections in computer science," *J. Comput. Sci. Coll.*, vol. 24, no. 6, pp. 57–61, June 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1529995.1530007

[9] S. S. Epp, *Discrete Mathematics with Applications*, 3rd ed. Belmont, CA: Books/Cole Publishing Company, 2004.

[10] A. Ralston, "Do we need any mathematics in computer science curricula?" *SIGCSE Bull.*, vol. 37, no. 2, pp. 6–9, June 2005. [Online]. Available: http://doi.acm.org/10.1145/1083431.1083433

[11] A. Decker and P. Ventura, "We claim this class for computer science: A non-mathematician's discrete structures course," in *SIGCSE*, Norfolk, Virginia, USA, 2007, pp. 100–104.

[12] K. McMaster, M. Anderson, and B. Rague, "Discrete math with programming: Better together," in *SIGCSE*, Covington, Kentucky, USA, 2007, pp. 442–446.

[13] SIGCSE Committee on the Implementation of a Discrete Mathematics Course, "Sigcse committee report," April 2007.

[14] C.-C. Xing, "Proof diagrams: A graphical tool for assisting set proofs," *CCSC*, vol. 22, no. 5, pp. 70–77, 2007.

[15] R. Johnsonbaugh and D. P. Miller, "Converses of pumping lemmas," in *Proceedings of the twenty-first SIGCSE technical symposium on Computer science education*, ser. SIGCSE '90. New York, NY, USA: ACM, 1990, pp. 27–30. [Online]. Available: http://doi.acm.org/10.1145/323410.319073

Comparison of elite and regional school computer science education and the impact on students' ability

***Shuhua Ren*[1], *Jie Liu*[2] , *David Olson*[3]**

[1]School of Information Science and Engineering, Dalian Polytechnic University, Dalian, P.R. China, 116034
[2,3]Department of Computer Science, Western Oregon University，Monmouth, OR, USA，97361

Abstract：

This paper compares teaching between Western Oregon University (WOU) and MIT through randomly selected courses. From the quantitative point of view, the paper explores the patterns and characteristics of teaching computer courses , and aiming the proportion of data for all aspects, the paper discussed the impact on the different aspects of students' comprehensive ability.

The facts have proved that the experiments and assignments, quizzes, midterm exams, term project and presentations, class attendance and participations, final exam and other aspects have very different impact on the students' ability.

We hope the discussions in this paper can provide possible areas for improving course and curricular design in Computer Science.

Keywords：Computer teaching, teaching mode, assessment methods

1 Introduction

Today, computer science education focuses more and more on students' ability to comprehend and to prepare them to be innovative. The traditional lecture-assignment-test approach is not effective to stimulate students' enthusiasm to improve their interests in the subjects.

Teaching is an important component in affecting a student's learning outcome. Activities used in teaching a class can differ greatly between one professor and another. Contents aside, it is very important to have the students interested in the topics a professor is covering. Therefore, the teaching process should be flexible, adaptable, targeted, and response to students needs, only then the students' interest may be stimulated and master the topics in a higher level. This is the conventional wisdom. This paper shows that we have to look at other areas to find differences.

2 Data Statistics Object

Course assessment and grading are essential components of teaching activities. The approach in assessment will determines the way of comprehensive training of students and the teaching effectiveness.

In order to identify assessment patterns, we randomly selected 24 of Western Oregon University (WOU) Computer Science courses and 27 courses in WOU's Department of Mathematics as our base for study [1]. As base for comparison, we also randomly selected 34 courses from MIT's (Massachusetts Institute of Technology) Electronics and Computer Science curriculum [2]. The selected courses belong to all level of undergraduate curriculum. WOU is a regional public university.

By means of extracting, classifying and analyzing the syllabi from MIT and WOU, we find that teaching and evaluation activities in these courses in our studied programs include labs, exercises, quizzes, midterm exams, term projects, presentations, classroom attendance and participation, and final examination.

3 The goal of our study

By means of analyzing on a variety of assessment types separately and accounts for a percentage of the total scores of courses, we attempt to quantify the differences among different programs and hope to find correlation between assessment methods and their impacts on students' learning.

4 Breakdown of statistics

Here WOU-C stands for WOU's Computer Science courses, WOU-M stands for the mathematics courses, MIT stands for Massachusetts Institute of Technology.

4.1 Labs & Assignments

Table 1：Labs & Assignments

The percentage in total score	Number of courses			Percentage of the total survey %		
	WOU-C	WOU-M	MIT	WOU-C	WOU-M	MIT
0%	4	3	2	16.67	11.11	5.88
5~10%	1		5	4.17		14.71
15~20%	4	5	7	16.67	18.52	20.59
25~30%	7	9	11	29.17	33.33	32.35
35~40%	6	4	4	25	14.81	11.76
45~50%	1	6	3	4.17	22.22	8.82
55~60%	1		2	4.17		5.88

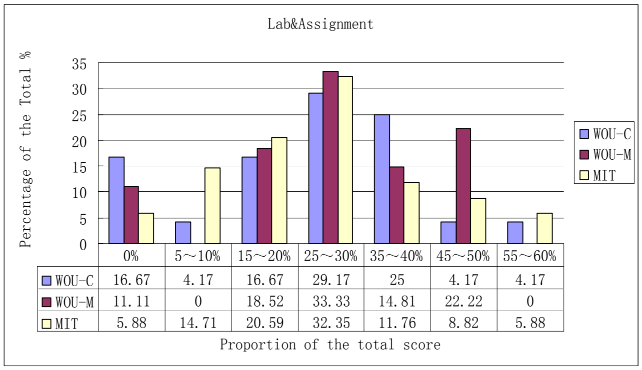


Figure 1. The distribution of labs and assignments as the percentage of students' grades

Finding #1：

Labs and Assignments in the proportion of the total score generally contribute to about 15% to 40% of final grades at WOU. MIT's courses weight for wider distribution, generally between 10% and 60%. Overall, relatively speaking, MIT and the WOU-C assigned to the Labs and Assignments ratio slightly higher than that of the WOU-M. Courses weight less on labs and assignments generally add projects and presentations.

The data show that, regardless of the elite or ordinary universities, labs and assignments are still an important means for assessment and training. On the amount of work, there are no significant differences between the elite and an ordinary University. The proportion in the total score of each school is basically the same, about 25% to 30%.

4.2 Quiz

Table 2：Contribution Quizzes

The percentage in total score	Number of courses			Percentage of the total survey %		
	WOU-C	WOU-M	MIT	WOU-C	WOU-M	MIT
0%	13	17	23	54.17	62.96	67.65
5~10%		4	1		14.81	2.94
15~20%	9	5	1	37.5	18.52	2.94
25~30%	1		6	4.17		17.65
35~40%	1		2	4.17		5.88
45~50%			4			11.76
55~60%		1	1		3.7	2.94

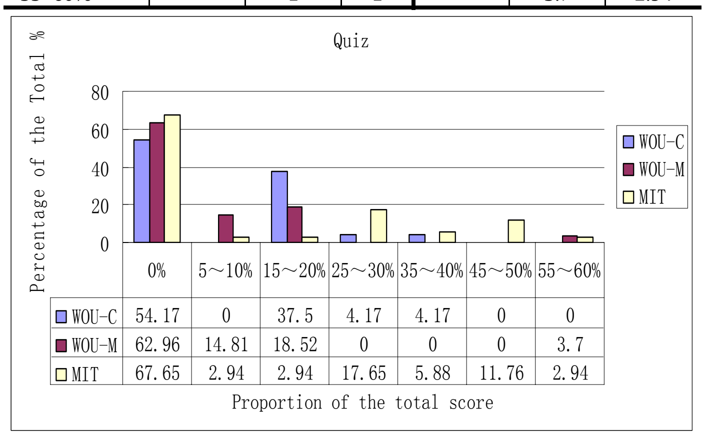


Figure 2. The distribution of quizzes as the percentage of students' grades

Finding #2：

Quizzes at WOU-M weight between 10% to 20% of final grades. WOU-C generally allocates between about 20 to 40 percent to quizzes; MIT generally accounts for about 25% to 50%, higher than the WOU-C and the WOU-M, and widely distributed. The number of quizzes generally is about 2 to 4 times. This shows that the elite has more stringent requirements for students.

Some classes treat quizzes and mid-terms the same. Quizzes generally are given at the end of each chapter or every one or two weeks and cover topics discussed in the previous weeks. If we add two mid-term exams, the examination of each course is very frequent. The quizzes can be effective to push students with a busy life to spend more time study.

4.3 Midterm Exam

Table 3：Midterm Exam

The percentage in total score	Number of courses			Percentage of the total survey %		
	WOU-C	WOU-M	MIT	WOU-C	WOU-M	MIT
0%	9	3	25	37.5	11.11	73.53
5~10%			1			2.94
15~20%	9	1		37.5	3.7	
25~30%	5	11	7	20.83	40.74	20.59
35~40%	1	10		4.17	37.04	
45~50%		2			7.41	
55~60%			1			2.94

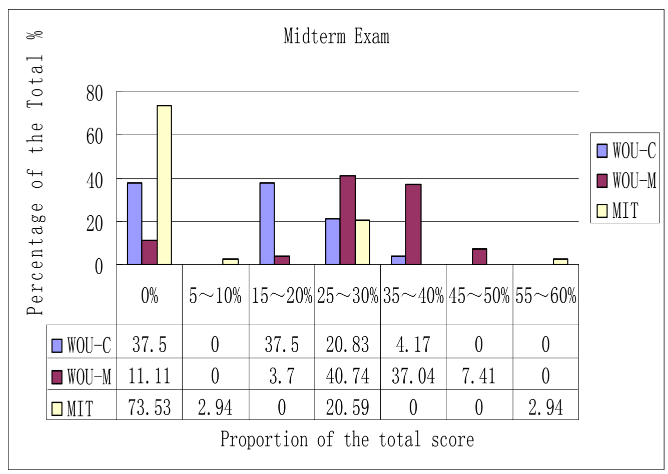


Figure 3. The distribution of midterm exam as the percentage of students' grades

Finding #3：

WOU-C weights mid-term exams generally between 15% and 30% of students' final grades; WOU-M's range is generally about 25% to 40%. MIT's mid-term exams weight generally between 25% to 30%. Math classes and MIT's mid-term exams are generally about 2 times. The results show that the proportion of roughly equal between elite schools and regional schools.

4.4 Presentation/Project

Table 4：Presentation/Project

The percentage in total score	Number of courses			Percentage of the total survey %		
	WOU-C	WOU-M	MIT	WOU-C	WOU-M	MIT
0%	13	22	19	54.17	81.48	55.88
5~10%	7	1	1	29.17	3.7	2.94
15~20%	2	2	4	8.33	7.41	11.76
25~30%	1	1	1	4.17	3.7	2.94
35~40%	0	1	3		3.7	8.82
45~50%			2			5.88
55~60%			2			5.88
65~80%	1		2	4.17		5.88

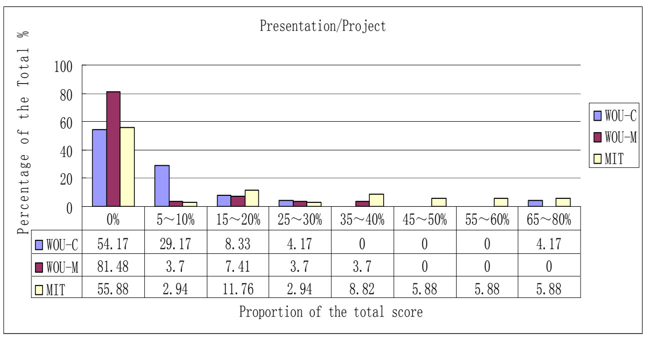


Figure 4. The distribution of presentation/project as the percentage of students' grades

Finding #4：

WOU-C courses have a range between 5% to 30%, the majority are about 5%~10%. WOU-M courses distribute between 5% to 30%, and the MIT range is between 15% to 80%. Lower division and services courses are less likely to have presentation, while upper division and special topic courses are more likely to have big term projects and presentations. MIT courses favor projects more.

Our experiences tell us that the impact presentations is very positive in enhancing students' ability to expand their knowledge, self-confidence, and presentation skills.

4.5 Attendance and participation

Table 5：Attendance and participation

The percentage in total score	Number of courses			Percentage of the total survey %		
	WOU-C	WOU-M	MIT	WOU-C	WOU-M	MIT
0%	18	21	16	75	77.78	47.06
5~10%	6	5	12	25	18.52	35.3
15~20%		1	5		3.7	14.71
25~30%			1			2.94
35~40%						
45~50%						
55~60%						

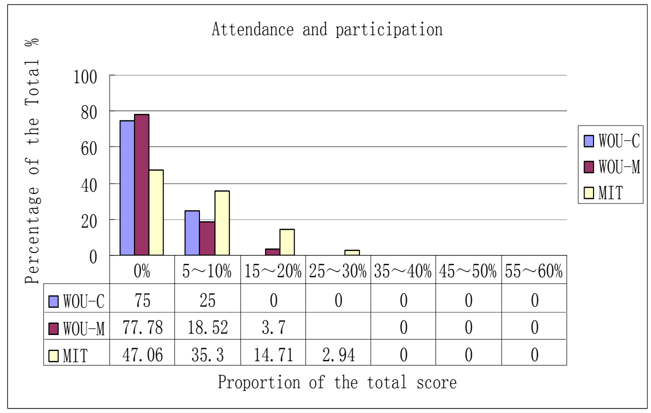


Figure 5. The distribution of attendance and participation as the percentage of students' grades

Finding #5：

The main focus is to examine the students' attendance and participations in the classroom, WOU-C and WOU-M accounting for between 5 and 10%; MIT's proportion is 5% to 20%, up to 30%. Various universities attach very different importance to the student's attendance and class participations.

4.6 Final Exam

Table 6：Final Exam

The percentage in total score	Number of courses			Percentage of the total survey %		
	WOU-C	WOU-M	MIT	WOU-C	WOU-M	MIT
0%	3	4	21	12.5	14.81	61.76
5~10%						
15~20%	3	17		12.5	62.96	
25~30%	5	6	4	20.83	22.22	11.76
35~40%	10		5	41.67		14.7
45~50%	3		3	12.5		8.82
55~60%			1			2.94

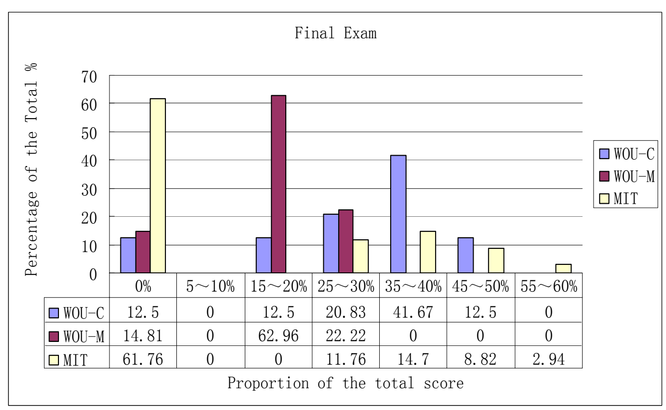


Figure 6. The distribution of final Exam as the percentage of students' grades

Finding #6：

The overall weight of the final exams in the total score is about 15% to 50%. WOU-M final exams account for the between 15% and 30%, with most course ranges between 15% and 20%. WOU-C's finals range between 15% to 50%, 35% to 40% being the majority. Both of the WOU programs weight less for their finals than that of MIT, which are between 25% to 50%. This indicates that elite emphasis more on final exams.

The final exam has a very important position in the performance assessment. Most schools generally give about 25% to 40% up to 60% allocations to the finals, probably because it provide opportunities to test students' mastery of knowledge and practical ability from many aspects of tops covered. Besides, it is the way the professors were trained.

4.7 Summary

Table 7：Summary of assessments and their distribution

Item	WOU				MIT	
	WOU-M-low	WOU-M-high	WOU-C-low	WOU-C-high	MIT-low	MIT-high
Lab& Assignment	15%	50%	15%	40%	10%	50%
Quizzes	10%	20%	15%	40%	25%	50%
Midterm Exam	25%	40%	15%	30%	25%	30%
Presentation/Project	5%	30%	5%	30%	15%	80%
Attendance and participation	5%	10%	5%	10%	5%	20%
Final Exam	15%	20%	35%	40%	25%	50%

5 Course Grading

As showing in Table 8, the mapping between a student's letter grade and her cumulative score is almost universal for both schools and all divisions.

Table 8：Mapping of cumulative scores and final grades.

No	Distribution	Grading
1	100%-92%	A
2	91%-90%	A-
3	89%-88%	B+
4	87%-82%	B
5	81%-80%	B-
6	79%-78%	C+
7	77%-72%	C
8	71%-70%	C-
9	69%-68%	D+
10	67%-62%	D
11	61%-60%	D-
12	59%-0%	F

6 Conclusions

The teaching and assessment methods between a regional university and MIT are not that much different. Even with different programs within the same school, learning activities and assessment methods are not that different either. Does this mean that had a student attended two different schools, the student would have received similar educations? We do not have the answer for that question.

We are plan to look into other areas of teaching. We will "take" an online class and compare it with the similar class we teach at WOU.

7 References

[1]Liberal Arts Core Curriculum Advising Guide.2011-2012, www.wou.edu/provost/aalc/advising/laccguide.php

[2] http://www.mit.com

[3]Liu J., J. Marsaglia, D. Olson, and W Zhang, “Teaching Foreign Languages Over The Internet”, in the Proceedings of “The 2008 International Conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government” July 14-17, 2008, Las Vegas.

[4]Liu J. and et al, “Teaching Parallel Programming and Building Parallel Computers”, in the Proceedings of “The 2005 International Conference on Frontiers in Education: Computer Science and Computer Engineering,” June 20-23, 2005, Las Vegas, pp. 149-155.

[5]Liu J., J. Marsaglia, and D. Olson, “Aiming for Success after Graduation”, in the Proceedings of “The 2005 International Conference on Frontiers in Education: Computer Science and Computer Engineering” June 20-23, 2005, Las Vegas, pp. 127-132.

[6]Liu J. and Marsaglia J., "A Mastery Based Multimedia Teaching Environ-ment For CS1," a research proposal submitted to NSF, June 1995.

[7]Liu J. and F. Liu, Teaching Parallel Programming with Multi-core Computers, The 2010 International Conference on Frontiers in Education: Computer Science and Computer Engineering, July 12-15, 2010, USA

[8]Liu J. and F. Liu, Preparing Software Engineering Graduates For IT 2.0 Era, The 2011 International Conference on Frontiers in Education: Com-puter Science and Computer Engineering, July 18-21, 2011, Las Vegas, USA

The Need for the Development of Graduate Schools of Telecommunications

Mark Cummings, Victor Clincy, & Victor Marshall
Kennesaw State University, Kennesaw, Georgia, USA

Abstract - *The telecommunications industry and the associated academic discipline have been evolving for over a century now. This paper will briefly explore the evolutionary process and then describe a new type of graduate-level organization for telecommunications education.*

Keywords: Telecommunications education, Graduate education

1 Introduction

The telecommunications industry and the associated academic discipline has been evolving for the last 150 years. This evolutionary process has brought us to the point where it is appropriate to consider a new type of academic organization. This paper will briefly explore the evolutionary process and then describe the new type of academic organization.

Today we have a wide variety of communications related academic disciplines or organizational structures that are converging in the telecommunications field. A partial list in rough order of historical evolution includes:

- Speech
- Journalism
- Electrical Engineering
- International Relations
- Law, Regulation, & Political Science
- General Semantics
- Broadcasting
- Computer Science
- Communications Theory
- Linguistics
- Semiotics
- Telecommunications Management
- Media Studies
- Law

The question that confronts us in the field of Telecommunications is, does dividing in this way, produce the best possible outcomes in the development of knowledge about and delivering that knowledge to future practitioners and academics ? The authors posit that the answer to this question is “no.” To help understand why this is so, the authors will paint a harsh black and white version of history. This is not meant to deny that there are many shades of grey, but the contrast is intended to make the discussion clear and easy to understand. Adding all the shades of grey would not change the ultimate conclusion, only make it more difficult to explicate. The pattern described herein relates in general to academia on a global basis. Because of political events in certain parts of the World, there are some places where the nature of government involvement and associated political processes produced some deviations[1]. These deviations (in part by accident) have been helpful in pointing the way forward.

2 Telecom Major Areas

It might be interesting to trace the evolution of these disciplines and identify which started to overlap with the others, but the authors leave that for another paper. Rather, here it will just be noted that with the emergence of the Internet, cellular, WiFi, Bluetooth, etc. all of these disciplines have converged in and around telecommunications. From the point-of-view of the telecommunications professional, it is no longer possible to confront a telecommunications question from one of these disciplines. Rather, all problems tend to involve a combination of three major areas:

- Technology
- Economics
- Policy

Recent examples of this convergence of these three areas in major telecommunications work include:

[1] For example in China where the government owned and operated telecommunications system in-house training center, became The Beijing University of Post and Telecommunications (BUPT) as telecommunications moved from being government owned into the commercial sphere.

- WiMAX and LightSquared – where what seemed to be promising when researched, studied and developed solely from the technology perspective encountered unexpected challenges from the economic (including business strategy) or policy areas.
- TV White Space – responding to an economic environment of spectrum scarcity produced by the growth of cellular with public policy based on Cognitive Radio technology to improve spectral efficiency.
- Conflict over Unlicensed vs. Licensed spectrum – where unlicensed proponents claim that public benefit from inherent technology and business structures that produce accelerated innovation and competition thus lowering prices versus licensed proponents claiming public benefit from quality of service assurances inherent in cellular technology and business structures.
- Content questions are driving current protocol technology research in GENI (NSF funded wired test bed for research on future internet technology) on alternatives to TCP/IP based on the unique needs of Social Networking.

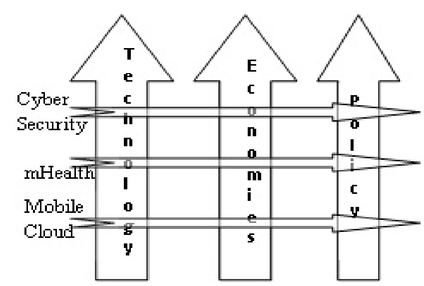
The convergence of these three areas is also apparent in Open Flow, Inter Cloud, and a long list of activities in cyber security and privacy issues.

So far, academia has responded to this multidimensional nature of today's telecommunications with academic programs housed in one of the previously existing organizations that attempt to involve other disciplines [2]. This has become commonly recognized [3]. Because housing even multidisciplinary programs in single discipline academic organizations creates an unavoidable bias towards the hosting organization's discipline, this is less than optimal, especially at the graduate level. So for example, multidisciplinary programs in Law Schools tend to be strong in policy, but not so much in business and technology. Similarly, programs in Engineering Schools tend to be strong in technology, but not so much in policy and business. Likewise, programs housed in Business Schools tend to be strong in business, economics, strategy, etc. but not so much in policy and technology.

Recent attempts to create multidisciplinary programs through cooperation of two or three schools within a university are an important development and a step in the right direction[4]. However, they lack the cohesive core that a focused school devoted to telecommunications would bring. Similarly, the creation of Schools of Information Technology is an important step forward and should be lauded, but they do not provide the full multidisciplinary focus on telecommunications required.

What is needed, then, are Graduate Schools of Telecommunications (GST) that are fundamentally multidisciplinary[5]. As such, they would not seek to displace the traditional academic organizations. There is and will be a need for focused academic work in each of these areas and the structures that have been developed should continue to provide a locus for that. GST's will provide a venue for the specialists in each of the related disciplines to come together and interact.

GST's can be organized around three major tracks with cross cutting areas of emphasis as shown in the illustration below.



The Tracks are the fundamental knowledge, tools and expertise that underlie the field. The Areas of Emphasis are the application of these to

2 See Appendix A for a list of these.
3 For example, "The diversity of the field of telecommunications is also reflected in the way telecom programs are structured within different universities. While many of the programs are present within schools of engineering, others have been offered by schools of sciences, communications, management, …" [1].

4 Please refer to summary table of Appendix A1.
5 Although still early in its gestation, general recognition of this need is indicated in the academic community by [1] and in industry by private conversations the authors have had with industry leaders.

specific problem spaces. Each Area of Emphasis will draw on different aspects of Technology, Economics, and Policy in different combinations.

2.1 Tracks

The tracks are general areas and each act as a container for a number of specialties. The boundaries between these tracks are not always sharply defined. For example, the history of telecommunications standards started as treaties negotiated by the Secretaries of State of the participating countries and failure to comply with the technical standards promulgated was defined as an act of war.

The *Technology Track* includes telecommunications related aspects of electrical engineering, computer science and sub specialties in each that revolve around semiconductors, sub systems, systems, transmission characteristics, protocols, encryption, identification, network design and optimization. It also includes communications theory, representation theory, and mathematical modeling.

The *Economics Track* could just as easily be titled the Business Track or the Strategy Track because it includes elements of all three. Specifically, the Economics Track includes telecommunications related aspects of macro-economics, micro-economics, finance, business strategy, marketing, supply-chains, international business, and operations.

The *Policy Track* includes telecommunications related aspects of standards development organizations, industry associations, country / regional / international regulation, law, import/export control, spectrum auctions, industrial policy, effects on society, and content control.

2.2 Areas of Emphasis

The areas-of-emphasis make use of a combination of the tracks to solve or explore particular problems. They are thus cross cutting by fundamental nature. Each may draw on different parts and/or aspects of the tracks. By their very nature, the cross cutting areas of emphasis may change over time. Some examples of areas that could be appropriate at the present time are: Cyber Security, Mobile Health (now called mHealth), Mobile Payments, Mobile Clouds, Augmented Reality, Disaster Preparedness / Recovery, Public Safety, and Defense.

3 Notes on Curricula

Because of conscious decisions, needs of local industry, student populations, and historical accidents, GST curricula will vary from institution to institution. This is as it should be. However, there are some things that all should have in common.

All GST's curricula should have set of requirements for degrees that require all students to take a core set of courses that cover all three tracks before specializing in one track or defining a program of study based on an area of emphasis. Each GST may define this core differently. Over time as experience develops in the various GST's, some common principles of constructing a core curriculum may develop[6].

GST degrees may evolve over time, but at this point it seems that a professional Master of Telecommunications and an academic Ph.D. of Telecommunications are required. It is likely that the Masters program will have a significant participation by students already working in the field. The Ph.D. program is likely to have a significant number of resident students involved in research programs. Providing interaction between these two groups of students will create an especially valuable environment. It will give the academic researchers exposure to the current real World problems in the industry while giving the students working in the industry exposure to the leading research in the field.

4 Notes on Staffing

Each GST will need to have a core group of dedicated faculty members in a balance across the three tracks. Additional faculty members should be drawn from the dedicated discipline areas. The dedicated discipline areas can be in the same university or other institutions in the area. Faculty should also be drawn from professionals in local industry. In this way, each GST becomes a regional resource.

---

6 For early work in this area see "A Framework for Defining Graduate Telecommunications Education" by Hammad Iqbal and Richard A. Thompson from the School of Information Sciences at the University of Pittsburg with the assistance of International Telecom Education and Research Association (ITERA) with data drawn from University of Pittsburgh (PIT), University of Colorado at Boulder (COL), Rochester Institute of Technology (RIT), Ball State University (BSU), Alaska Pacific University (APU), and Western Michigan University (WMU).

5 Conclusion

The telecommunications industry and the associated academic discipline have been evolving for the last 150 years. This evolutionary process has brought us to the point where, because of the field's fundamental interdisciplinary nature, Telecommunications graduate programs need to be created as stand alone academic organizations within universities. The resulting Graduate Schools of Telecommunications need to be structured so as to be regionally focused providing a venue for vibrant interaction of all associated academic disciplines and industry participants supported by a core, dedicated faculty.

6 Appendix 1

The first 11 listed below are what Google returns as the "Top Graduate Telecom Programs". It is interesting to note that they are all in the US and they are all housed in either a technical or business school. Google does not return more policy based programs such as the University of Pennsylvania School of Law that is shown 12th at the end of this list. The "Focus" section of each listing is actual self-describing text from the institution.

The following table summarizes these twelve programs in a framework that groups the programs by academic organizational home(s). This table also summarizes the degrees conferred. As indicated, these programs are housed in the business, computer science, engineering, information sciences and/or law programs at these institutions. Of the twelve programs, only three represent coordination between multiple academic disciplines: one program represents a joint effort between the business and computer science disciplines, and two represent a joint effort of the business, computer science and engineering disciplines. The remaining nine programs are offered by one of the following academic disciplines: computer science, engineering, information science, or law. The table also illustrates that most programs offer a masters degree, but two programs offer a doctorate, one offers a JD, one offers a professional degree, and another offers a certificate program.

6.1 Summary of Academic Programs

Institution	Academic Organizational Home(s)					Degrees / Certificates				
	Business School	**Computer Science**	**Engineering**	**Information Sciences**	**Law**	**Doctorate**	**JD**	**Masters**	**Professional**	**Certificate**
Johns Hopkins U in Baltimore, MA		X				X		X		
University of Maryland in College Park	X	X						X		
NC State University in Raleigh NC	X	X	X					X		
Northeastern University in Boston Mass	X	X	X					X		
University of Oklahoma in Norman OK			X					X		
University of Nebraska in Lincoln NE			X					X		
RIT, Rochester NY			X					X		
Southern Methodist University in Dallas Texas			X					X	X	
Boston University in Boston Mass				X		X		X		X
Syracuse University in Syracuse NY				X				X		
University of Pittsburgh in Pittsburgh PA				X				X		
University of Pennsylvania					X		X			

6.2 Detailed Program Information

NAME OF INSTITUTION: University of Maryland in College Park

LOCATION OF PROGRAM: Joint CS Department and College of Business

DEGREES GRANTED: Masters Degree

FOCUS: A Master of Science in Telecommunications, with specialized tracks in business and management, information systems security, networking and wireless.

URL: **http://www.telecom.umd.edu/**

NAME OF INSTITUTION: Johns Hopkins University in Baltimore, MA

LOCATION OF PROGRAM: Computer Science Department

DEGREES GRANTED: Masters Degree and Doctorate Degree

FOCUS: Targets computer science and electrical engineering graduate students. Students can select an option for telecommunications and networking. The program includes courses in network programming, communication systems engineering, telecommunications systems processing and advanced communications systems.

URL: **http://ep.jhu.edu/graduate-degree-programs/telecommunications-and-networking-option**

NAME OF INSTITUTION: North Carolina State University in Raleigh NC

LOCATION OF PROGRAM: Joint program offered by the Computer Science Department, Electrical & Computer Engineering Department, and College of Management

DEGREES GRANTED: Masters Degree

FOCUS: Program is intended for students with Electrical and Computer Engineering or Computer Science undergraduate degrees who wish to pursue careers in research, development, operations, and services in the networking and IT industry. It is designed to address a specific need in the networking industry in North Carolina, and, as such, is consistent with the land-grant mission of the University. The program is unique in the system of the University of North Carolina and nationally in blending the technical and management aspects of computer networking to offer students a truly unique educational opportunity.

URL: **http://networking.ncsu.edu/**

NAME OF INSTITUTION: Boston University in Boston Mass

LOCATION OF PROGRAM: Metropolitan College and the School of Information Sciences

DEGREES GRANTED: Certificate, Masters Degree and Doctorate

FOCUS: The program integrates knowledge of the computer science, engineering, managerial, and legal aspects of networking and telecommunications. The program is designed to provide knowledge and critical skills essential for success in this rapidly expanding field.

URL: **http://www.bu.edu/met/programs/graduate/telecommunication/**

NAME OF INSTITUTION: University of Oklahoma in Norman OK

LOCATION OF PROGRAM: School of Electrical and Computer Engineering

DEGREES GRANTED: Masters

FOCUS: There is a worldwide need for leaders who can apply advancements in computer and networking technology to the business of telecommunications. Specifically, there are business leaders trained in computational processes who have need for deeper knowledge of the technology of telecommunications and networking in order to make technically driven decisions. Similarly, there are engineers trained in the technology of communications who require a broader perspective in business parameters which affect new product and product life decisions. The Master of Science in Telecommunications Engineering develops on the synergy of these needs with a leadership program for the State of Oklahoma.

URL: **http://www.ou.edu/content/coe/ece/about_ece.html**

NAME OF INSTITUTION: Northeastern University in Boston Mass

LOCATION OF PROGRAM: Offered jointly by the Graduate School of Engineering and the Graduate School of Computer Science, in conjunction with the College of Business Administration.

DEGREES GRANTED: Master of Science in Telecommunication Systems Management

FOCUS: Program is designed for professionals currently in the telecommunications field who either wish to enhance their technical skills and credentials, or who wish to make a transition to the business side of telecommunications, especially to management or marketing. The disciplines from which students enter include engineering, computer science, the physical sciences, and mathematics. This program is one of only a very few master's programs in telecommunications in the United States that is truly multi-disciplinary.

URL: **http://www.northeastern.edu/graduate/programs/telecommunication-systems-management/**

NAME OF INSTITUTION: University of Nebraska in Lincoln Nebraska

LOCATION OF PROGRAM: College of Engineering

DEGREES GRANTED: Master of Science and Masters of Engineering in Telecommunications Engineering.

FOCUS: This graduate program provides advanced education and research to develop breadth of knowledge and depth of expertise in the engineering of telecommunication networks and systems. Specialized state-of-the-art laboratories and computer facilities are available including optical, wireless, and telecommunications laboratories; IBM supercomputer facilities; and high speed connections to Internet2 and AccessGrid.

URL: **http://www.unl.edu/gradstudies/prospective/programs/TelecommunicationsEngr.shtml**

NAME OF INSTITUTION: Syracuse University in Syracuse NY

LOCATION OF PROGRAM: School of Information Science

DEGREES GRANTED: Masters Degree in Telecommunications and Network Management

FOCUS: The Telecommunications and Network Management degree is a marriage of technology, policy and management. It is not an engineering or programming degree but gives IT professionals and new industry entrants a broader view of how to apply and use networking technologies to solve business problems.

URL: **http://ischool.syr.edu/academics/graduate/mstnm/index.aspx**

NAME OF INSTITUTION: Rochester Institute of Technology, Rochester NY

LOCATION OF PROGRAM: Dept of Electrical, Computer and Telecommunications Engineering Technology

DEGREES GRANTED: Master of Science Degree in Telecommunications Engineering

FOCUS: RIT offers a unique program focused on telecommunications that develops the advanced level of skill and knowledge needed by future leaders in the industry. The master of science degree in telecommunications engineering technology is for individuals who seek graduate education to advance into managerial and leadership roles in a dynamic telecommunications environment. This program can be completed through either online or on-campus study.

URL: **http://www.rit.edu/cast/ectet/ms-in-telecommunications-engineering-technology.php**

NAME OF INSTITUTION: Southern Methodist University in Dallas Texas

LOCATION OF PROGRAM: School of Engineering, Electrical Engineering Dept

DEGREES GRANTED: Masters Degree and Professional Degree

FOCUS: In recognition of the critical need for professional education in this field, the Lyle School of Engineering offers programs oriented toward the management of corporate communications and the design of telecommunication products and systems.

URL: **http://www.smu.edu/Lyle/Departments/EE/Programs/MS_Telecom**

NAME OF INSTITUTION: University of Pittsburgh in Pittsburgh PA

LOCATION OF PROGRAM: School of Information Science

DEGREES GRANTED: Masters in Telecommunications and Networking

FOCUS: Here at the iSchool at the University of Pittsburgh, your graduate studies will allow you to gain the critical skills and knowledge to be a leader in the global telecommunications field. Our curricula incorporates courses in network design and management, systems and wireless systems, network security, management and policy, and human communication and information processing.

URL: **http://www.ischool.pitt.edu/tele/index.php**

NAME OF INSTITUTION: University of Pennsylvania

LOCATION OF PROGRAM: Law School

DEGREES GRANTED: J.D.

FOCUS: Christopher Yoo, the John H. Chestnut Professor of Law, Communication, and Computer & Information Science is Director of the Center for Technology, Innovation & Competition at the University of Pennsylvania law School. His areas of focus include Telecommunications Law, Mass Media Law, Government Regulation,Telecommunications Law, Internet Law, Technology and Policy, and Cyberlaw/Internet Law. His recent research focuses on exploring how the principles of network engineering and the economics of imperfect competition can provide insights into the regulation of the Internet and other forms of electronic communications. He has been a leading voice in the “network neutrality” debate that has dominated Internet policy over the past several years. The Center for Technology, Innovation and Competition is dedicated to promoting foundational research that will shape and reshape the way legislators, regulatory authorities, and scholars think about technology policy. Through major scholarly conferences, symposia, faculty workshops, and other activities, CTIC is committed to providing a forum for exploring the full range of scholarly perspectives on these issues.

URLs:

http://www.law.upenn.edu/cf/faculty/csyoo/

http://www.law.upenn.edu/cf/institutes/ctic

7 Appendix 2

Below is an overview of the situation with Telecom programs in universities outside of the USA

7.1 China:

The Beijing University of post and Telecommunications (BUPT) is a complete standalone university focused on telecommunications. Based on the self description below and personal visits by one of the authors, BUPT can be said to have a technical focus. Other university programs in China appear to follow the US pattern with a heavy bias towards the technical side.

NAME OF INSTITUTION: BUPT

LOCATION OF PROGRAM: Standalone Telecommunications University

DEGREES GRANTED: M.S. & Ph.D.

FOCUS: BUPT has 11 schools, three research institutes and one center on Marxism teaching and research. The eleven schools are School of Information and Telecommunications Engineering, School of Electronic Engineering, School of Computer Science, School of Automation Engineering, School of Software Engineering, School of Economics and Management, School of Humanities, School of Science, International School, School of Network Education, and School of Ethnic Minority Education. The three research institutes are Institute of Network Technology, Institute of Optical Communication and Photoelectron and Institute of Sensing Technology and Business.

URL: **http://studyinchina.bupt.edu.cn/e_intro_bupt.html**

7.2 India:

Looking from the outside, it appears that India has fairly recently started specialized telecommunications programs in several Business Schools as illustrated by the two examples below. It appears that telecommunications technology is still being addressed in traditional Engineering and Computer Science programs.

NAME OF INSTITUTION: MIT School of Telecom Management Pune

LOCATION OF PROGRAM: School of Management

DEGREES GRANTED: Post Graduate Diploma In Management (Telecom)

MBA, M. Sc in International Business from University of Dundee, UK

FOCUS: Systems, Marketing, Finance and Human Resources

NAME OF INSTITUTION: International School of Telecom Technology and Management

LOCATION OF PROGRAM: Business School

DEGREES GRANTED: MBA

IIM Calcutta and XLRI Graduates along with the Top Executives of various Industries together formed International School of Telecom Technology and Management – ISTTM Hyderabad. ISTTM is an Innovation in MBA Degree. We believe, Academics and Real Time Experience is a perfect Combination to prepare next generation management Leaders.

URL: **www.isttm.com/new/default.aspx**

7.3 Japan:

NAME OF INSTITUTION: Yokohama National University

LOCATION OF PROGRAM: Engineering

DEGREES GRANTED: Ph.D.

FOCUS: The 2008 MEXT Global COE (Center of Excellence) Program selected the Innovative integration between Medicine and Engineering Based on Information Communications.

URL:http://gcoe.mict.ynu.ac.jp/en/index.php?overview

7.4 Europe:

There are a number of telecommunications programs in European universities. They seem to be primarily technically focused with a few Business programs offering Telecommunications as an area of emphasis.

8 References

[1] Hammad Iqbal and Richard A. Thompson, “A Framework for Defining Graduate Telecommunications Education”, In proceedings of International Telecommunications Education and Research Association Conference (ITERA) 2005, Las Vegas, NV, October 2005.

Note: *This is exploratory research that presents the case for the need of research in this area. Since these are largely uncharted waters, there is a paucity of existing research to cite within this field.*

Industry towards Embedded Curriculum Development

Pin Tao, Shiqiang Yang, Ninghan Zheng, Yongqiang Chen
Department of Computer Science and Technology Department, Tsinghua University, Beijing, China

Abstract - *Embedded system technology is a fast developing discipline in recent years. Universities are challenged with how to catch up with the industry speed and how to combine the industry technology with the university education. In this paper, we contribute our thinking and some practice work in the embedded curriculum development. By focusing on several platform and setup the strong cooperation with the industry, by adopting the marketed product in the lab, by accumulating the contribution from both the teacher and students year by year, the limit university education resource is collect together to make the curriculum better, the curriculum is highly evaluated by the student, university and industry.*

Keywords: Education, Embedded System

1 Introduction

As portable embedded devices and wireless communication are booming, they will have great impact on data and information services in the current human society and future world. These services would be data density, computation density, and communication density. Therefore, university education needs to improve itself to accommodate catch up with to this changing. For Computer Engineering, Computer Science, Electronic Engineering department in all over the world, embedded system curriculum plays an important role in the university mayor education as it can integrate many professional knowledge of their major together and give the students a system viewpoint for the future career life.

Embedded technology is mainly droved by the industry which introduces an important challenge to the university on how to build the curriculum towards the industry. In this paper, we will give our thinking and some practice work to solve this problem we faced.

2 Catch up with the latest chipset

With the quick development of the embedded technology, generation by generation advanced embedded chipset went to market for industry products, better and better. How to make the lab design catch up with the quick changing technology? How to training the students with the state-of-art platform? This is a great challenge for the teachers. We try to figure out the solution from the following three aspects.

2.1 Focus on some special platform

It is well known that there are thousands processors, software and solutions in the embedded system industry. Which one is the best for embedded education? It is a pity that no one can give a convincing answer. Actually, what the university should transfer to the student is the knowledge itself. For embedded system, the architecture, the power management, the bus, the real-time OS, the porting methodology, the debug method, etc. are the fundamental knowledge points for the students. We do not need to be distracted by many solutions, and just need to focus on one typical platform. It is enough for the education itself, and made us being more expert in the general knowledge and special platform. For example, our embedded system adopts the Intel embedded platform from year 2004, and we were familiar with the traditional EIA platform and the latest Atom based intelligent platform.

2.2 Setup the deep cooperation with Industry

As the embedded system technology is mainly droved by the industry, it is important for the university set up the strong and deep cooperation with the industry to get the detail information, resource from the industry and know the future direction of the industry. We join in the Intel university plan and act as the leading university in China, attend the Intel embedded education summit every year and many other workshops and seminars, join in the cooperation research projects and get the latest materials and hardware platforms frequently. And also at the same time, we sharing our contribution into the Intel university program, sharing our education slides with other universities in China. This cooperation greatly help us setup the world class embedded curriculum and training the best students for the world [1].

2.3 Update the curriculum materials every year

In order to catch up with the changing technology, we updating the curriculum resource every year, not only the lecture slides, but also the lab designs and lab platform itself. This updating comes from the industry, e.g. Intel, from the understanding and work of the teacher and teaching assistant, and also from the students who join in the curriculum. The teacher should work hard on the update task, otherwise he/she will find the courseware is not fit for the students of next year.

3 Adopt the marketed product

As far as we know, most of the embedded products are powered by Linux and other open source software. Under the GNU general public license, many manufacturers have release their implementation based on the open source, such as some network router, web cameras, etc. These marketed products also are good lab platform for embedded curriculum.

We adopt TP-LINK MR3420 wireless router in our Lab which is developed by TP-LINK, one of the top network device manufacturers. This device use an Atheros SoC chip with MIPS core as its main embedded processor, and running embedded Linux to provide the network data transfer. As their Linux implementation has been released in the website of company[2], we contact with TP-LINK on adopting their product in the university education. They give the positive feedback and support us to try it with the GNU tool chain. Finally we successfully download the image into the device and it works. After that, we made more improvement for this lab design. Firstly, we refine the source code and build process to fit the university education, for example permanently destroy avoided bootloader design. Secondly, we compile a detail instruction book for the students from the fundamental information to the step by step operations. Thirdly, we further adopt the OpenWRT solution [3] for this MR3420 device to make the lab system more powerful and more open. Finally, we replace the on board 4MB Flash memory and 16MB SDRAM with large chips which make the students can design more complex function for the device.

As the price of each device is less than $30, the total cost of about 30 sets device does not exceed $2,000. These 30 sets MR3420 give over 150 students every year the opportunity to get experience on how to build Linux image for a ready-to-use product. Most of the students highly evaluated this lab design and interest in adding various functions for the device.

The advantages of adopting marketed products lie in,

1) Promote the interesting of the students, the students are glad to DIY an industry product by themselves, glad to make it works in the real world, glad to modify, customize and add the function in the product.

2) Hardware is durable for lab management, the industry products normally are more durable than the experiment box specially for the education. This feature let the burden and cost of the lab management is lower.

3) Low price, the low total cost makes it possible to purchase the devices easily and upgrade the lab platform using the next generation products in the future easily.

4 Incremental education

We believe the education should be accumulated year by year incrementally, neither keep it almost unchanged in several years, nor starting over every three to five years. According to the incremental construction, the curriculum resource for students and teacher will become better continuously.

In order to make the incremental construction possible, we firstly establish a curriculum framework which is fit for incremental improvement. From the viewpoint of a curriculum, the resource of it can be contributed from three aspects, teacher and teaching assistant, students who join in the curriculum, the industry sponsors. Therefore, the teaching resource, including the syllabus and the lab design, should be a long term developing plan. It can setup a prototype in the original first year, and has the potential to be enriched in the future decades. For example, real time is a key knowledge point in the embedded system, we give the fundamental theory of real time OS and the simple real time performance measurement lab in the first year of 2006. And then we refine the lecture slides and lab design in the following years, such as enhance and change the work load in the lab, add the new schedule algorithm in the system and measure the real time response time in the following years. All the excellent student labs were reserved in the curriculum FTP site and their lab reports and possible source codes were provide for the following years students. The new students were asked to learn from the previous work and do some difference. Then the incremental education works fine.

5 Curriculum Evaluation

With the efforts of years' work, the courseware and labs update year by year according to the rapid progress of the embedded technology. As the course keep pace with the-state-of-art technology, it is well reputed by the students. According to the statistic result of about over 1500 courses in Tsinghua University, the embedded system curriculum is the top 15% student favorite course of all. The embedded system experiment platform was awarded the first prize of best teaching software platform of Tsinghua University in 2008. And the course has been award the National Best Selected Curricula of Ministry of Education, China. In 2011, one of the lab work of the students win the 2nd place of Imagine Cup worldwide embedded competition.

6 Conclusions

In this paper, we proposed our work in deeply combine the embedded education with the quick developed industry technology. By summarizing the knowledge points of the embedded system, we believe the university do not need to diverse in many industry solutions, university can focus their efforts on one platform and keep the strong cooperation with the industry. The industry also can provides the low cost, interesting and durable to use devices for the lab experiment, they are good choices for the university embedded curriculum. Under the fast changing environment, the curriculum can build up their resource year by year with an incremental framework. Then the contributions from the teacher, teaching assistants, students can be combined together for the next year.

7 References

[1] Pin Tao, Jolly Wang, Hai Shen, Lihong Mu, University-Industry Collaboration for Embedded Education, IT Education, Vol 13, p138-140, 2011

[2] GPL code center of TP-LINK products, http://www.tp-link.com/en/support/gpl/

[3] OpenWRT, https://openwrt.org/

SESSION

TOOLS AND SYSTEMS + WEB USAGE METHODS + LEARNING ENVIRONMENTS + ETHICS AND RELATED ISSUES

Chair(s)

TBA

AnalyzeD: A Shared Tool for Analyzing Virtual Team Collaboration in Classroom Software Engineering Projects

T. Kowark and H. Plattner
Hasso Plattner Institute for IT Systems Engineering
University of Potsdam
Potsdam, Germany
{firstname.lastname}@hpi.uni-potsdam.de

Abstract— *Classroom software engineering projects are viable aids for the education of computer science students. They allow to experience different technologies, methods, and development processes first hand, and intensify the learning effect compared to traditional lectures. But how to track whether the learning goals of such projects are fulfilled? Main focus in project evaluation often resides on results, since tracking the process itself can only be achieved with substantial manual efforts.*

We present AnalyzeD, a tool that simplifies monitoring of development processes within such settings. It enables observation of groupware systems for the occurrence of behavior that reflects defined learning goals like adherence to principles of the taught development processes or effective and efficient usage of the provided tools. More importantly, it allows sharing analysis results, thereby, breaking up the isolation of different classroom projects and allowing all users to benefit from previous insights into the inner workings of student teams.

Keywords: Collaborative software, Computer aided instruction

1. Introduction

Classroom software engineering projects have a long standing track record in aiding computer science education by enabling students to experience the theoretical foundations taught in software engineering lectures first hand in risk-free environments [2], [4], [19], [20]. Classroom projects, not only limited to the domain of software engineering, have also been a viable source for scientific insights into the inner workings of engineering teams and the effects of certain collaboration principles on project success and engineer productivity [6], [18].

Collaboration is nowadays increasingly focused on digital tools. Groupware systems, such as email lists, wikis, bug tracking systems, or version control systems, are ubiquitous in modern software engineering projects and much effort is put into connecting the remaining analog artifacts of software engineering projects to their digital counterparts. Analysis of these digital artifacts has shown to provide indicators for the current state of engineering teams and even the expected project outcome [21], [22].

Based on these findings, we developed AnalyzeD, a system that allows to capture, analyze, and compare such collaboration data from arbitrary project settings and groupware landscapes. Virtual collaboration artifacts in that regard are traceable actions within the provided systems, e.g., editing of a wiki page or creation of a new source code revision. The system further allows to share the results of the analysis of virtual collaboration artifacts directly with other users, which, in turn, can try to reproduce the observations within their respective project settings. Thereby, AnalyzeD extends the available database for verification or falsification of theories about the impact of certain virtual collaboration behavior on collaborative work beyond single institutions.

Such a system is especially interesting in classroom settings, since the teaching team to student ratio usually prohibits to observe each student or student team as closely as it would be necessary in order to constantly recognize deviations from the intended development processes or advised usage patterns for provided groupware systems. By providing a unified view on collaboration activities in all accessible collaboration tools, indicators for such deviations and misuses can be recognized and permit to provide more focused feedback to the students. Beyond that, the system offers the possibility to specify desired and unwanted behavior and share it with other users of AnalyzeD. This means, if indicators for good or bad project work have been identified, the entire user group of the tool immediately has access to this information and can search their own collaboration activities for occurrences of the described behavior.

This paper intends to make AnalyzeD known to a wider audience of software engineering lecturers and encourage them to use it within their educational software engineering projects or experiments. The remainder of the paper is structured as follows. First, we give an overview about related work in the field of virtual collaboration analysis. Then, we provide a detailed description of the system architecture. Finally, we present experiences made during a case study in a software engineering lecture. The paper concludes with an outline of next steps in the development of the system as well as further research work that can build upon the presented efforts.

2. Related Work

The idea of analyzing digital artifacts of team collaboration in software engineering projects was subject to prior research. The tools presented in related work, however, either lack the flexibility to be applied in arbitrary groupware setups or were isolated solutions that did not facilitate publication of analysis results to other users. In the following, we present an excerpt of publications about such tools and outline the similarities and differences to AnalyzeD.

Hackystat is an open-source tool that aims to automatically and unobtrusively collect, analyze, and interpret software development process and product measurements [10]. It allows to capture usage data from a variety of development tools and analyze this data through interchangeable client interfaces. The implementation is not designed to facilitate sharing of collaboration traces and analysis results since different installations remain isolated from each other. With regards to specifying desired or unwanted behavior, extensions of the platform were, for example, used to automatically determine whether programmers adhere to the principles of Test-Driven-Development (TDD) [9]. The efforts presented in this paper are a generalization of such extensions since they support not only singular use-cases within closed environments, but facilitate the encoding of collaboration behavior utilizing any possible combination of collaboration tools.

Microsoft's Team Foundation Server [5] is a collaboration tool suite that allows its users to analyze and compare their own collaboration behavior with other teams that use this platform. It is, however, limited to collaboration activities that are performed using the platform itself. If different version control, bug tracking, or email systems are used, the corresponding activities cannot be analyzed.

The Empirical Project Monitor is a tool that aggregates collaboration artifacts from different data sources, as well [14]. Similar to AnalyzeD, it relies on feeder applications that parse data sources like source code management systems, bug trackers, or email archives. A communication model is provided in order to detect potential problems within the collaboration behavior based on empirical studies. However, this model is encoded within the software itself and cannot be adapted to incorporate new knowledge without recoding and redeploying the system itself.

Wu et al. created a metric-based, multi-agent system that supports project management by gathering information from various sources of the development environment and deducing the current project status by means of intelligent agents that relate software development activities to the previously created project plan [23]. The system is applicable to various project setups, but the different installations remain isolated from one another. Insights about activities, which have effects on team performance cannot be made available to other projects without updating the evaluation logic of the software agents and redeploying the system.

A slightly different approach was taken by Reid et. al [16]. They introduced DrProject, an entire platform of project management tools that was specifically tailored to educational needs. By developing the entire groupware landscape by themselves, they simplified the process of collecting and analyzing virtual collaboration activities in order to better understand how students work together in classroom software engineering settings. This simplicity in data collection comes at the cost of realism with regards to the tools used and reduces generalizability of the analysis results since no abstract model of the collaboration activities is provided.

3. System Architecture

AnalyzeD is based on a previously developed platform for virtual collaboration monitoring [13]. This platform was rebuild to be available as an on-premise solution, which can be unobtrusively deployed alongside the infrastructure used for the projects under investigation. Further, it was extended with a central repository that allows to publish the results of virtual collaboration analysis and make them available to other users of the system.

3.1 Local Platform for Data Capturing and Analysis

The local analysis component of AnalyzeD is presented in detail in Fig. 1. It consists of three components. Firstly, specialized sensor clients capture collaboration events from the groupware landscape. For each used groupware tool, a specialized sensor client needs to be available. Regardless of the parsed systems, each client needs to implement an interface that allows the "Configuration and Control Service" to get basic information about required input parameters, to invoke the client, and to retrieve the results, i.e., the parsed collaboration events, in a common format. Thus, the system remains adaptable to newly created groupware tools.

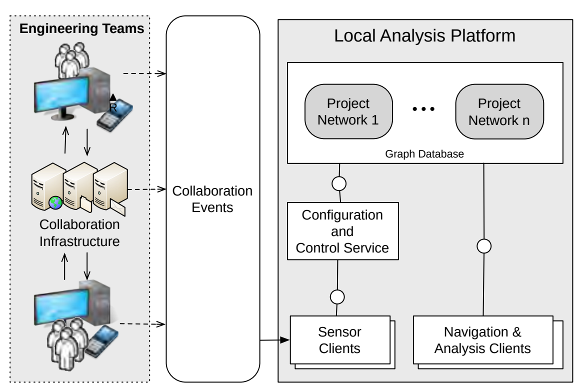


Fig. 1: Detailed Architecture of the local analysis platform component of AnalyzeD.

The data returned by the sensor clients is being aggregated into so-called Team Collaboration Networks (TCN) [21].

These networks are built upon ontologies that represent concepts of different collaboration tools, such as wikis, email lists, or version control systems. Thereby, the system can easily be extended just by creating a new ontology if new tools for collaboration are created that differ conceptually from the tools we know today. All nodes of the TCN can be linked by relations. Hence, these networks provide a holistic view that, for example, can model links from revisions to bug tracking items or from emails to wiki pages and allows to analyze whether the occurrence of such links might affect team collaboration. The TCN are stored within an off-the-shelf graph database.

The ontology-based system allows to add additional information to the TCN without having to change underlying data structures. If needed, new concepts or additions to existing ones can be modeled by means of an ontology, which then needs to be uploaded to the local analysis component. After that, sensor clients can be used to upload the corresponding data, e.g. calendar events, questionnaires filled out by project participants, or assessments of group or individual performance, and integrate it into the TCN. While currently not implemented, upcoming versions of the shared repository will allow users to upload these extension ontologies, too.

Analysis of the TCN can be performed by utilizing a SPARQL [15] endpoint, which is offered by the underlying graph database, or by using RDF/OWL compatible visualization and analysis tools.

3.2 Shared Repository for Virtual Collaboration Behavior

Analysis of said networks usually reveals reoccurring subgraphs that either have a direct impact on team performance or at least provide indicators for ongoing beneficial or detrimental collaboration behavior within the observed project teams. In [11], we formally described how to model such abstract subgraphs. In addition to the formal definition, we provided a graphical notation that allows to specify said behavioral snippets in a more accessible fashion.

Fig. 2 presents an example of this graphical notation containing all possible building blocks. Nodes are represented as circles that are labeled with possible classes of the node. The label can be extended by a set of valid tags for a node, which is depicted in squared brackets. Relations between nodes are represented by solid, directed edges. The label of an edge denotes the name of the relation and optionally defines cardinality constraints. Sequences can be expressed by dashed edges that connect two relations. Finally, attribute constraints are expressed within a separate rectangle that is attached to the respective node. Attribute constraints can be defined using SPARQL syntax.

As depicted in the architecture overview (see Fig. 3), we created a graphical editor (“Visual Creator”) for this notation. Thereby, users of the systems are able to model

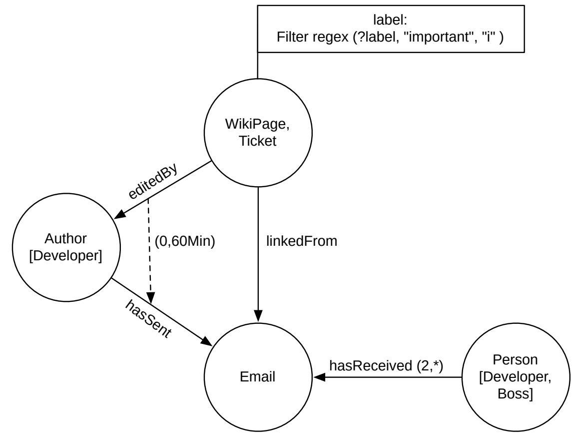


Fig. 2: Collaboration Pattern Example - Sending emails with hyperlinks to important tickets or wiki pages to fellow developers and superiors within 60 minutes after editing the respective artifacts.

behavioral snippets that represent, for example, learning goals within their classes. The graphical notations are translated into SPARQL queries for the purpose of detecting occurrences of the described behavior within the TCN under investigation. The “Matcher” component thereby takes into account characteristics of the respective networks and adopts cardinalities or terminology of tags. Finally, the “Observer” component allows to constantly monitor TCN for such occurrences and notify the user about newly detected instances. In combination, these components can serve as a freely configurable project management dashboard that simplifies the detection of indicators for desired or unwanted behavior within the monitored projects.

The content of the shared repository is available to each local analysis platform. Hence, all behavioral snippets are available for the analysis of local team collaboration networks and statistics about the detection of such behavior will be transferred back to the repository. Thereby, it becomes possible to determine whether observations are specific only to single projects or can be reproduced in different settings, as well.

4. Platform Application

In the following, we present experiences of using AnalyzeD within a classroom software engineering project. The lecture under investigation is a third year undergraduate software engineering class [12]. The 96 participants of the course were divided into two development groups, each consisting of eight teams. Those eight teams had the task of jointly developing a customer relationship management system (CRM).

At the beginning of the project, the students had to perform a mandatory preparation exercise regarding the

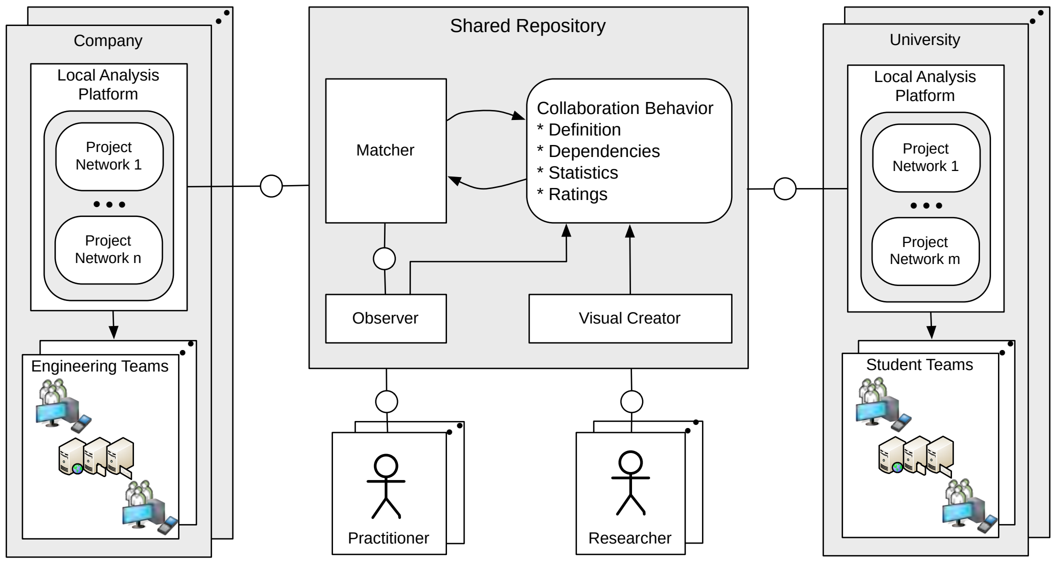


Fig. 3: AnalyzeD architecture overview showing the local analysis platforms that are connected with a central repository for virtual collaboration behavior.

used programming language and frameworks. They received tutorials for the groupware infrastructure, which consisted of dedicated email lists, Agilo [7] as a wiki and bug tracking solution, GIT [3] for version control management and Hudson CI [8] for continuous integration.

The project itself was performed using Scrum [17]. Each of the eight teams consisted of four to eight members, one of which took the role of the Product Owner (PO) and one served as the Scrum Master (SM). The POs had access to a team of customers embodied by members of our research group and performed initial requirements elicitation one week prior to project start. After that, all teams performed four sprints. Each sprint started with a sprint planning meeting and ended with a sprint review and a retrospection meeting. During sprint execution, the teams performed weekly Scrum meetings where they discussed ongoing developments and presented their work to the teaching team. Each sprint lasted three weeks and working time was officially limited to one day per week, however, additional work was not prohibited. The students were not allowed to freely choose their project teams but had to work with their colleagues from their concurrent capstone projects. By that, it was ensured that all teams could use dedicated rooms and workstations for the project.

The students received regular lectures that provided the theoretical background for the project work. Topics included different approaches for scaling Scrum to multiple teams working together in one project (e.g., [1]), introduction into development methods, such as Test- and Behavior-Driven-Development, and detailed tutorials for the tools used within the project.

From an educational point of view, the main focus did not reside on the overall project outcome, but on the adherence to the prescribed development process and usage of the presented tools in a way that contributes to project success. Therefore, we installed a team of four tutors that constantly monitored the progress of the teams and determined whether they adhere to the principles of Scrum and, if they do not, investigated the reasons for that. The tutors took part as silent observers in all meetings of the teams - weekly Scrum, sprint planning, sprint review, retrospection, SM, and PO meetings - and recorded their observations.

The presented iteration of the course is the second one. During the first iteration, we learned that manual observation during the meetings was not sufficient to completely capture the students teamwork. Firstly, teams avoided to mention problems in front of the teaching staff to prevent negative impacts on their grades. After the grades were announced, one team revealed that they thought co-located working would be rated higher than each team member performing their tasks individually. Hence, they tried to cover-up that one of their members was constantly working alone at home instead of the dedicated project space by excusing him during some regular meetings. Secondly, the tutors sometimes forgot to ask important questions and teams forgot to mention certain actions they took to improve their project work. For example, one project team was taking daily photos of their Scrum board and uploaded them into the wiki instead of using the provided ticket system. The tutor became aware of this fact only two weeks after the team started to exploit that behavior.

As a consequence of these examples, we decided to use

AnalyzeD as an additional tool for analyzing the virtual collaboration traces, which were created during the work that was carried out in between the scheduled meetings. The tutors, however, did not have access to the system. By that, we tried to ensure that they neither intentionally nor subconsciously tried to push the teams towards creating expected behavioral signatures.

4.1 Digital vs. Analog - Bug Tracking System Analysis

Being an agile software development process, Scrum does not focus on tools but on individuals and interactions. Hence, Scrum teams are supposed to find the most suitable tools for their work instead of dogmatically adhering to existing systems. Within the lecture, one intention was to teach the students that, even though digital bug trackers are viable aids within the software engineering process, they are sometimes inferior to analog solutions, such as physical Scrum boards.

The following observation was made during post-hoc analysis of the first iteration of the course: when teams started to focus on physical Scrum boards for ticket maintenance, the digital versions of the tickets were increasingly updated in bulk by designated members of the development teams. This permits to maintain an overview about the current progress directly within the team office, while simultaneously allowing to share the progress with other teams and the teaching staff through the groupware system with minimal effort. In order to detect this style of working, two behaviors were modeled.

Fig. 4 depicts that a ticket is changed by its owner. The owner in that regard is not the person that created the ticket, which often would be the PO, but the person who was assigned to implement the described feature or perform the desired task. We assumed that such behavior would mostly be exploited at the beginning of the project and decrease over time when small changes are carried out exclusively on the physical Scrum board.

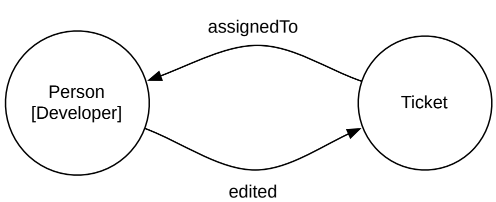


Fig. 4: Behavior 1. A ticket is changed by the developer that owns it.

The second behavior, updates by other team members that do not own the ticket, is depicted in Fig. 5. It is assumed to increase during the project as more changes are performed on the physical Scrum board and designated team members start to update the digital tickets every couple of days for their teammates.

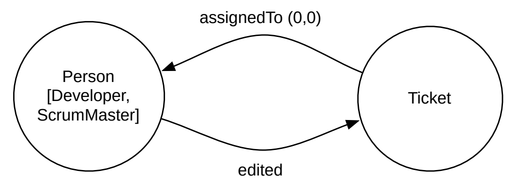


Fig. 5: Behavior 2. A ticket is changed by a team member that is not the owner of the ticket.

We used AnalyzeD to model these behaviors and monitor their occurrences. Fig. 6 shows the detected number of occurrences during the project for an entire development group. It is clearly visible that, overall, the assumptions about the occurrence rate of the behaviors held true. After many developers initially edited their own tickets on a frequent basis, turnaround was reached midway through the project, where changes by non-owners became more frequent.

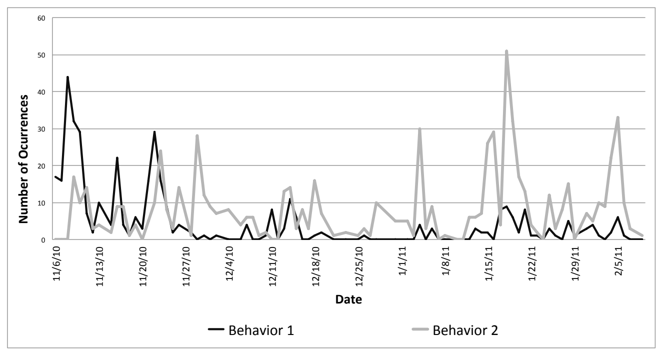


Fig. 6: Development of occurrences of the behaviors under investigation over the course of the project.

In addition to a graphical indicator for reached learning goals, the observation mechanism had a second positive aspect to it. By narrowing down the number of tickets worth investigating to just a fraction of the overall activity in the ticket system, i.e., those that were changed by their owners, it allowed us to provide more focused feedback to the students compared to just giving general advices to the entire class of 96 people. Resuming the example above, where the team tried to cover-up the student that refused to work with the entire group, AnalyzeD helped us to detect such behavior during project runtime instead of only in post-project evaluations and, thus, allowed us to react accordingly. Tutors could investigate the reasons for the behavior and tried to find suitable solutions in consultation with the teams.

4.2 Limitations

Digital collaboration is just a fraction of overall collaboration in software engineering projects. Therefore, the analysis of this data can only assist manual process observation, not replace it entirely. When applying the system, some aspects need to be taken in consideration to meaningfully analyze the collected data.

Connected tools can lead to false conclusions. If, for example, the bug tracking system is connected to the version control system by detecting ticket references in commit messages and automatically performing the corresponding ticket changes, focusing on only the ticket system during the analysis would be insufficient. Instead, it is then necessary to extend the respective ontologies in order to reflect the link between the two systems. Further, the behaviors modeled above need to be adopted to that situation and have to include source code revisions to detect tickets that were changed by their respective owners by referencing them in commit messages.

Another factor, which needs to be taken into account, are usage mistakes. Especially at the beginning of projects, students tend to misuse groupware tools due to inexperience. This could lead to frequent ticket updates without meaningful changes or a high number of source code revisions that fix errors of previous ones. In the case study, however, these unexpected usage patterns revealed that the tutorials for some of the tools were insufficient and additional training was necessary.

Finally, digital collaboration analysis is prone to observer effects. If students know that the way they use groupware tools is analyzed, it becomes much more likely that they adapt their behavior to generate supposedly ideal traces instead of focusing on doing what is best for the project. The students in our case study were informed upfront that their tool usage is being monitored and how we expect them to use the tools. Still, it was possible for us to detect the presented examples. While obviously not all students optimized their usage behavior to comply with our expectations, we cannot neglect the possibility that an unknown number of students did just that. Therefore, virtual collaboration analysis should only accompany thorough manual guidance and the results must always be put in context to the respective project setup.

5. Conclusion and Future Work

In this paper, we presented AnalyzeD, a system that allows to capture and analyze virtual team collaboration activities in an unobtrusive manner and publish the results of the analysis to a central repository that is available for all users of the platform. It is well-suited for classroom settings as it allows comparison of collaboration activity across different groupware setups and promotes the exchange of knowledge about detrimental or beneficial virtual collaboration behavior between software engineering teaching personnel of otherwise unrelated institutions.

The presented application is a simple example that showcases the general idea of the system. Due to the extensibility of the TCN approach and its ability to reflect not only the entirety of visible virtual collaboration activity in projects but add additional meta-information about project participants, more complex examples are likely to emerge. Use cases that will be investigated in upcoming experiments include: determining whether teams are dealing with the tasks at hand or if they are spending time on unnecessary tasks, detecting hot spot teams or team members that are overloaded with tasks due to extensive dependencies, and verifying if adherence to development principles like Test-Driven-Development can be detected by virtual collaboration analysis and if it is possible to quantify their effect on team productivity and performance. A shared platform that allows to publish such findings and enables other researchers and practitioners to falsify or verify them without the inherent delays of traditional publishing methods promises to be a viable aid for this kind of research.

6. Availability

The system is currently being prepared for open-source release under a yet to be determined license. Additional information and downloads are available at `http://epic.hpi.uni-potsdam.de/Home/AnalyzeD`.

References

[1] S. W. Ambler. Scaling agile software development through lean governance. In *SDG '09: Proceedings of the 2009 ICSE Workshop on Software Development Governance*, Washington, DC, USA, 2009. IEEE Computer Society.

[2] J. Bergin, C. Kussmaul, T. Reichlmayr, J. Caristi, and G. Pollice. Agile development in computer science education: practices and prognosis. In *SIGCSE '05: Proceedings of the 36th SIGCSE technical symposium on Computer science education*, pages 130–131, New York, NY, USA, 2005. ACM.

[3] Scott Chacon. Git - the fast version control system, http://www.git-scm.com.

[4] D. Coppit and J. M. Haddox-Schatz. Large team projects in software engineering courses. *SIGCSE Bull.*, 37(1):137–141, February 2005.

[5] Microsoft Corporation. http://www.microsoft.com/visualstudio/en-us/products/2010-editions/team-foundation-server/overview, Visual Studio Team Foundation Server 2010, 2011.

[6] S.H. Edwards. Improving student performance by evaluating how well students test their own programs. *J. Educ. Resour. Comput.*, 3(3):1, 2003.

[7] Agilo Software GmbH. Agilo for trac, http://www.agilofortrac.com/.

[8] Oracle Inc. Hudson ci, http://hudson-ci.org/.

[9] P. M. Johnson and H. Kou. Automated recognition of test-driven development with zorro. In *Proceedings of the AGILE 2007*, pages 15–25, Washington, DC, USA, 2007. IEEE Computer Society.

[10] P. M. Johnson, H. Kou, J. M. Agustin, Q. Zhang, A. Kagawa, and T. Yamashita. Practical automated process and product metric collection and analysis in a classroom setting: Lessons learned from hackystat-uh. *International Symposium on Empirical Software Engineering*, 2004.

[11] T. Kowark, P. Dobrigkeit, and A. Zeier. Towards a shared repository for patterns in virtual team collaboration. In *5th International Conference on New Trends in Information Science and Service Science*, 2011.

[12] T. Kowark, J. Müller, S. Müller, and A. Zeier. An educational testbed for the computational analysis of collaboration in early stages of software development processes. In *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS)*, January 2011.

[13] T. Kowark, M. Uflacker, and A. Zeier. Towards a shared platform for virtual collaboration analysis. In *The 2011 International Conference on Software Engineering Research and Practice (SERP '11)*, 2011.

[14] M. Ohira, R. Yokomori, M. Sakai, K. Matsumoto, K. Inoue, M. Barker, and K. Torii. Empirical project monitor: A system for managing software development projects in real time. In *International Symposium on Empirical Software Engineering*, Redondo Beach, USA, 2004.

[15] E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF, W3C Recommendation, http://www.w3.org/TR/rdf-sparql-query/.

[16] K. L. Reid and G. V. Wilson. Drproject: a software project management portal to meet educational needs. In *SIGCSE '07: Proceedings of the 38th SIGCSE technical symposium on Computer science education*, New York, NY, USA, 2007. ACM.

[17] K. Schwaber and M. Beedle. *Agile Software Development with Scrum*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[18] P. Skogstad. *A Unified Innovation Process Model for Engineering Designers and Managers*. PhD thesis, Stanford University, Center for Design Research, 2009.

[19] M. Staron. Using experiments in software engineering as an auxiliary tool for teaching–a qualitative evaluation from the perspective of students' learning process. In *ICSE '07: Proceedings of the 29th international conference on Software Engineering*, Washington, DC, USA, 2007.

[20] W.-K. Tan and H.-H. Teo. Training students to be innovative information systems developers: synergizing project-based learning with problem-based learning. In *SIGMIS-CPR '09: Proceedings of the special interest group on management information system's 47th annual conference on Computer personnel research*, New York, NY, USA, 2009. ACM.

[21] M. Uflacker. *Monitoring Virtual Team Collaboration: Methods, Applications, and Experiences in Engineering Design*. PhD thesis, Hasso Plattner Institute for IT Systems Engineering, Potsdam, Germany, 2010.

[22] I. Westphal, K. Thoben, and M. Seifert. Measuring collaboration performance in virtual organizations. *Establishing the foundation of collaborative networks*, pages 33–42, 2007.

[23] C.-S. Wu, W.-C. Chang, and I. K. Sethi. A metric-based multi-agent system for software project management. *ACIS International Conference on Computer and Information Science*, 2009.

wProjects: Dynamic Web Development Projects for Female High School Students

Elizabeth Thiry and Mary Beth Rosson
Center for Human-Computer Interaction
The Pennsylvania State University
University Park, Pennsylvania 16803 USA
exn152@psu.edu, mrosson@psu.edu

Abstract—**We report new developments for wConnect, an action research project aimed at recruiting more women into computer and information science. A key outreach activity for wConnect has been high school workshops, where we deliver hands-on web programming experiences. Over the past three years we have fielded and evaluated 11 workshops; nine used a Bridgetools end-user platform and two used wProjects, a Drupal-based tool. We report design and evaluation of the new tool, including some evidence that it improves on the earlier tools. We also report on new features allowing teachers to design their own workshop projects.**

Keywords-component; End-User Web Development; K-12 Education; Content Management Systems; Action Research

I. INTRODUCTION

Over the past two decades the proportion of women seeking education and careers in the computer and information sciences (CIS) has dropped dramatically [1]. Researchers concerned with this trend have investigated many factors that may contribute to the decline, for example the social context of learning [2]. Others have focused instead on interventions at the college level aimed attracting women and other minorities (e.g., introductory projects that emphasize media [3]). This paper reports anr effort aimed specifically at high school girls: workshops that introduce them to dynamic web development.

The high school workshops are an outreach activity of wConnect, an online community of women who support one another in their CIS education and career goals [3]. The workshops have been developed and fielded by undergraduate members of the community, typically conducted at a former high school of the workshop leader. In this paper we briefly introduce wConnect and its goals, the end-user tools we built to support the workshops, and early experiences fielding the workshops. This preliminary work has been reported in greater detail elsewhere [5][6]. The main body of this paper concerns how and why we have moved to a different web development tool for the workshops, along with reactions to the new tool from both student and teacher perspectives. We close by discussing current status, implications and future directions.

A. *wConnect, A Developmental Learning Community*

wConnect is a developmental learning community [7]; members with diverse levels of CIS interest and expertise interact and support one another in education and professional goals. The community includes high school girls who may not even (yet) be considering CIS education or careers; 1st and 2nd year undergraduates who have decided to pursue CIS education or are considering doing so; more advanced undergraduates in CIS majors; and female alumnae, graduate students, and other professionals working in the CIS field.

In the past three years, wConnect has developed through participatory action research [8]. A core team of undergraduate members has created an online community space and a number of associated online activities. The community space emerged over three different stages: a research prototype, a site that used the Facebook public API, and one built in Drupal; see [9] for a detailed design discussion of the wConnect online space.

In addition to interacting with and supporting one another in online activities, wConnect community members organize outreach events to engage with young women in middle and high school, as well as undergraduate women who have not yet developed an interest in CIS as part of their career goals. The work reported in this paper concerns one outreach activity that has been underway since project inception: hands-on high school workshops that introduce web development activities.

B. *End-User Web Development as an Outreach Activity*

A primary goal for wConnect is to engage and inspire young women who are not planning to pursue CIS education, to help them see how it might fit into their personal plans. One way wConnect pursues this goal is with high school workshops where girls create their own websites. We focus on web development (e.g., rather than robotics or media computation) because the web is a pervasive platform for data and services that can be recruited to serve many different real world interests. We also know from studies of the CLICK tool that nonprogrammers can create basic database-centric web applications with little training and background expertise [10][11].

To support the workshops we built a custom workspace in Bridgetools, a Java-based research toolkit that supports a range of fully interactive and web-based content objects [12]. At the top of Figure 1 is a screenshot of a Bridgetools workspace. Users can open and edit a database (upper right), one or more queries (middle), and web pages (left). At any point these objects can be opened and used in a browser (lower left).

The high school girls create *dynamic web applications*: web pages that update dynamically as a function of changes to a database and/or queries submitted to that database. Our goal has been to convey how a web page can be a "viewer" for a set

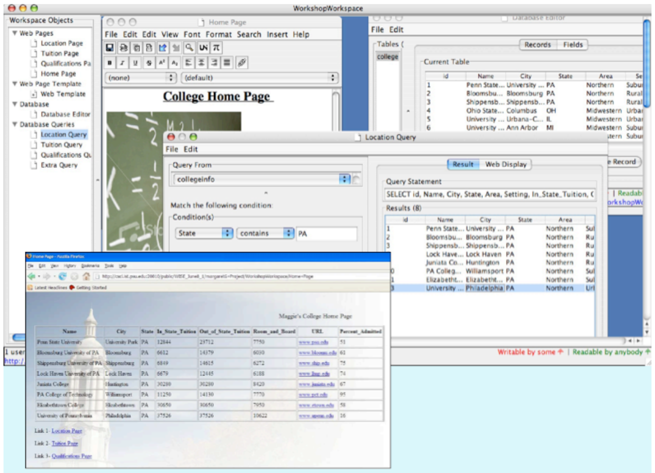


Figure 1. Bridgetools workspace, with a page editor (upper left), database (right), query editor (center) and resulting page (bottom left).

of stored data. Thus the young women were introduced to the basic concepts of stored information (records in a database), queries that extract subsets of the stored information, and web pages that present the results of queries. They learned these concepts by adding, editing and deleting their own records and creating queries against the records. The web page editor uses a special tag for embedding a named query and dynamically rendered query results as part of a page. In Figure 1 the student has stored information about universities she was investigating; the page lists all of her findings that are in Pennsylvania.

In the past three years we have conducted nine workshops using the Bridgetools workspace; five were held in university classrooms and four at area high schools. All workshops lasted from 60-90 minutes, during which the students learned to use the tools, generating tables of data, queries and associated web pages. As reported elsewhere [5][6], workshop outcomes were promising, with most students enthusiastic about their web project creations and demonstrating basic comprehension of the database, query and web display concepts.

The Bridgetools infrastructure enabled us to quickly assemble a workspace that could be used for personal web development projects. However, as articulated in [13], we came to recognize that the Bridgetools approach is not sustainable as an outreach platform for the wConnect community. For Bridgetools workshops that occurred in school settings, we regularly experienced technical problems: some school networks had firewalls blocking access to the research lab server hosting these tools; others had workstations with out-dated versions of Java or Java WebStart. Even when the activity ran as intended, preparation for each session was complex and tedious for the wConnect team, requiring a number of Bridgetools objects to be created and initialized for each participant (e.g., see the left side of workspace in Figure 1). These problems have limited the dissemination of the workshop concept to teachers and other interested parties. Further, Bridgetools is complex and wConnect members do not have the skills to debug or extend it when needed. Because dissemination and long-term sustainability are central project goals for wConnect, it became critical for us to develop a more generalizable set of end-user tools for use in the workshops.

II. wProjects

As we began work on a new tool for the workshops, several issues were foremost. Because of parallel work developing an online community space, wConnect members had built a body of expertise with Drupal and its open source community. Drupal runs reliably on most web browsers without advance work or testing in school classrooms. Although the platform constrains designers to generic web forms interaction (versus Bridgetools' object-specific editors), this user interface style has become familiar to general web users. All of these issues made Drupal an attractive option for wConnect [13].

Prior to beginning Drupal development, we confirmed that the content management system would support our basic pedagogical goals: adding, editing and deleting records, creating queries against the records, and finally building a website that uses both static data as well as the dynamic data delivered through their queries. As a bonus, we expected that Drupal's large open source community could be leveraged to help in dissemination and long-term maintenance of the tool. Finally, because Drupal is a web application itself, the issues of access to our servers and installation of software can be eliminated; all that the high school students need is an account, access to the internet, and no blocking of our tool's URL.

As we built the Drupal-based authoring tool (we call the new tool wProjects) we found that in addition to addressing many of our concerns about Bridgetools, Drupal brings the benefit that *any authorized user can design and specify their own workshop web project* using the same forms-based interface provided for student projects. This adds to the impact and generalizability of wProjects because it expands the target population of end users to be not only students learners but also teachers, mentors or other educators. A characterization of a teacher "web project design" scenario was shared in [13]; here we add to this with some early teacher-user experiences.

The core of wProjects is a set of custom modules that work with a MySQL database (part of the Drupal platform) to 1) gather and store specifications for a project (e.g., the number and structure of the data tables it requires); and 2) store and manipulate instances of projects generated by students (e.g., records, queries, web pages). The custom modules follow Drupal community standards, they are written in Php and use the Drupal API. Each module handles one basic task within wProjects: project specification (i.e., for a workshop leader); project instantiation (i.e., for a workshop participant); input, editing and deleting of table records; query creation and editing; and page creation and editing.

Figure 2 depicts the pieces of wProjects used by Sally (a pseudonym for one recent participant) to add records to a table, create a query, and work with saved queries. In a) Sally created her table by making multiple passes through a web-forms dialog that presents one form for each field in a table named "Careers". The parameters of the table – name, number and names of fields, and brief prompts to guide user input – have

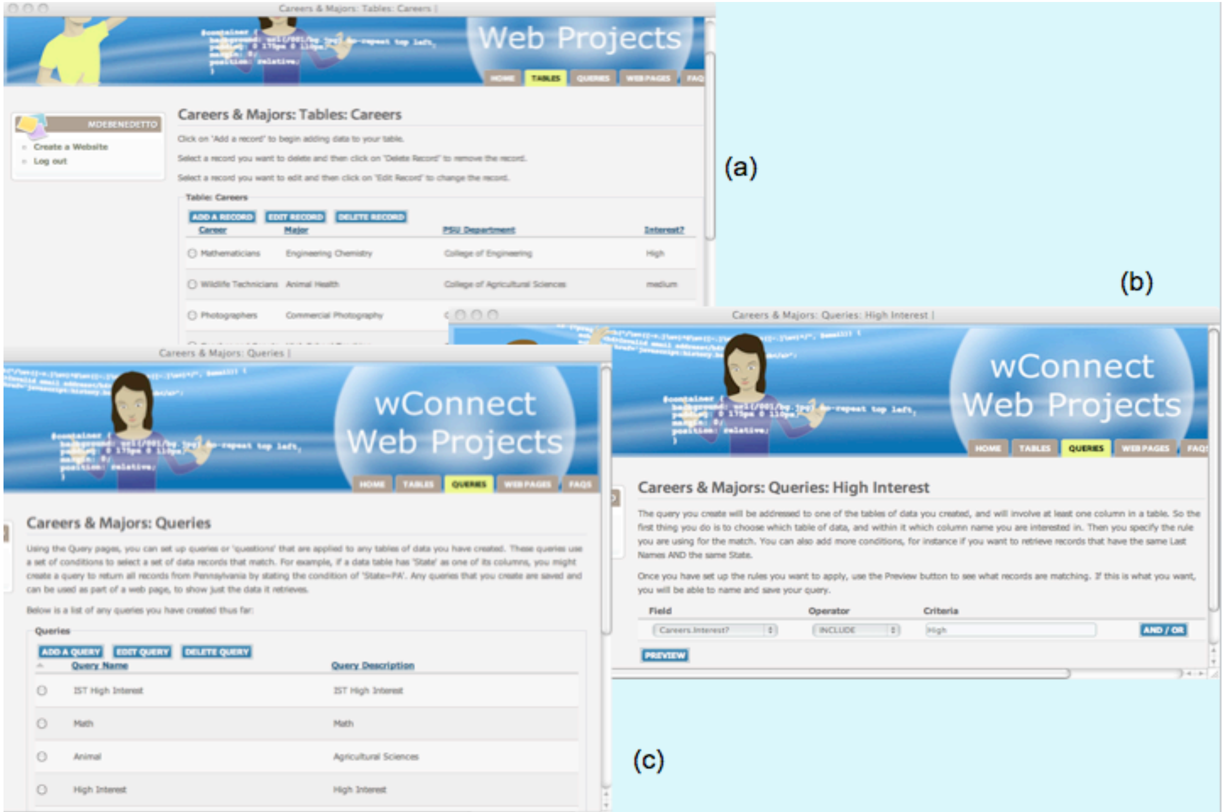


Figure 2. Core database and query features in wProjects, including a) data entry; b) query formulation; c) query selection and use.

been pre-specified by the web project designer. In this Careers and Majors project, girls are guided to explore web-based resources related to careers and majors, collecting pairings of varying interest in the table. At any point, Sally can use the Query tab to navigate to forms designed for specifying and previewing queries on her data table. For example in b) she has used a drop-down menu to select the "Interest" field and enter the string "High" as an includes criterion; when she presses the Preview button, the query will be applied and she will see what if any data are retrieved from her table. If she is satisfied with the result, she can c) name and save her query for later use. Sally can build as many queries as she likes.

Also at any point, Sally can use the Web Page tab to start working on web pages for her project (Figure 3). Once she has created one page she can return to edit that page further or create new pages. In either case she uses a version of an open-source HTML editor that we extended to support query content (`fckeditor`). This editor provides standard functionality for web page editing, including the option to work directly with the HTML source. For the Careers and Majors project we also pre-fetched evocative images but invited girls to find or generate content that they like.

The key connection between the web page and the database is the integration of query objects. We offer a simple selection dialog: users can press the customized query icon and to see a pop-up menu of saved queries. So, when Sally chose her saved High Interest query, `[High Interest]` was added at the text insertion point (a). When a web page is saved and opened, the query results are gathered and rendered as a table using the current Drupal theme (b). Girls are invited to change their page themes as desired; the wProjects designers have developed a

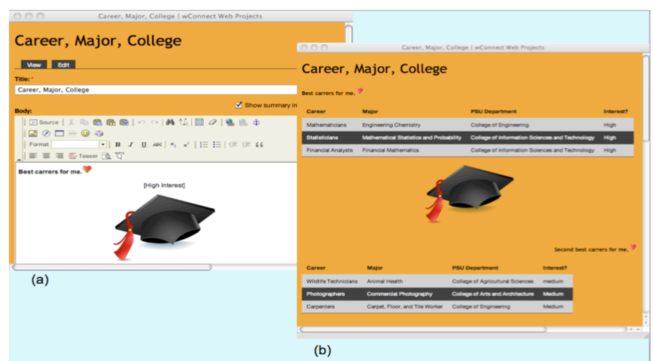


Figure 3. The (a) web page editor (with query insertion) and (b) formatted pages provided by wProjects.

small set to choose from, including a "vanilla" theme of black text on white (not surprisingly, no girls used that theme!). Once Sally has a data table, queries, and web pages, she can move flexibly among them, editing, adding and building connections as desired. In her case, she decided to insert five queries one after the other on one page. The work is saved for the participant in case she returns to wProjects for further work on this or other projects in the repository.

The implementation of wProjects is fairly straightforward because it reuses and extends existing Drupal modules. As stated earlier, the core concept is to store a project designer's specifications (along with meta-data about the owner, date of creation, etc.) as a MySQL record. Each time a project is instantiated, the pre-specified "tables" and "fields" are set up in another database table, this time with meta-data that connects the project to the student's wProjects ID. As data are added, they are stored in this project-specific table. When queries are specified, the conditions are stored along with the user data and the table to which the query is directed. Web pages are standard nodes in the Drupal implementation, with information about who, when, and for which project they were created. The query extension to `fckeditor` was modeled on the Bridgetools editor, with the exception that we added a menu icon to simplify the user interface. We turn now to students' and teachers' reactions to the wProjects web projects tool.

III. Student Reactions

We recently conducted two workshops using wProjects, the first in a classroom in our university (participants were from a local high school) and the second in a rural high school in our region. Nine girls participated in the first, ranging from 9th through 11th grade; eight were in the second, ranging from 9th thrugh 12th grade. Of the 17, one had been exposed to some programming in a summer camp, but none had taken a high school programming course. For a 4-item CIS efficacy scale excerpted from prior work [14] (Cronbach α=.78 for these 17 girls), the mean was 4.1 on a 7-point scale – right at the neutral value. The girls were recruited by a teacher contact at each school and completed a short background survey as part of their application and informed consent.

Both workshops presented the "Careers & Majors" project. The workshop leader (a different one at each site) introduced the project concept and web resources useful for investigating career interests, and the wConnect project leader made brief comments about our own College's undergraduate programs. The remaining time was spent on project development in three subtasks – table editing; query formulation; and web page editing. For each subtask, the leader demonstrated the process first, then the girls were given time to explore on their own. Interaction among the girls was encouraged as they searched for information, images or other content for their pages. The first workshop was time-limited as the girls had to catch a bus back to their school (about 50 minutes); in the second more time was available (about 90 minutes). The workshop closed with a take-home sheet about the wConnect community and future work with wProjects, and a brief feedback survey.

The feedback survey included the four reaction ratings that had also been used in earlier Bridgetools workshops. One probed the girl's understanding of the web project goals; a second her feelings of success in completing it; a third how much fun she had; and a fourth how encouraged she felt about attempting similar web projects in the future. All ratings were on a 5-point scale and average values ranged from 4.5 to 4.7; all were significantly greater than the neutral value of 3 ($t(16)=12.9$; $t(16)=11.8$; $t(16)=15.0$; $t(16)=7.6$; $p<.001$ for all measures). This general finding replicates findings from our earlier Bridgetools workshops [6]; most girls believed that they "got" the concept of a web project and enjoyed themselves.

Given that the general workshop goals were the same for the Bridgetools versus wProjects instances, we wondered how the girls' reactions might compare across the two tools. As we argued in an earlier design paper [13], wProjects is simpler and more reliable than Bridgetools, but achieves this through a rather tedious forms-based, sequential user interface. Thus we decided to make an exploratory contrast between the wProjects workshops and the most recent Bridgetools session (conducted at an area high school in Spring 2010, with 16 participants ranging from 9th to 10th grade). Although the web projects in focus for the two workshops were different (the earlier workshop used a "Places We've Been" project documenting vacations or family visits), the basic tasks of table editing, query building, and web page editing were the same. Figure 4 graphs a contrast for the four sets of subjective reactions; the mean ratings for understanding, felt success and fun are significantly greater for wProjects ($t(33)=2.1$; $t(33)=2.3$; $t(33)=2.5$; $p<.05$).

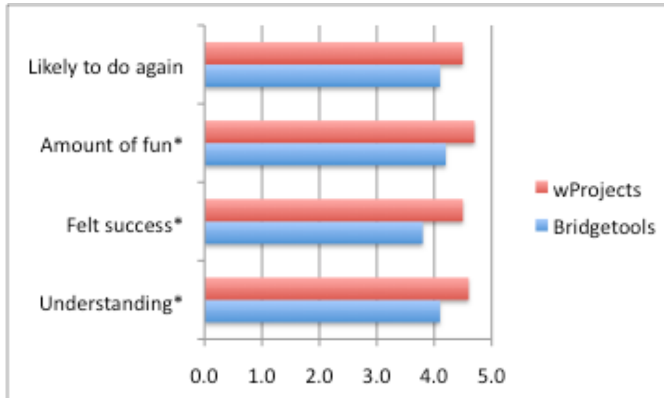


Figure 4. Ratings contrasts for Bridgetools versus wProjects.

It is important to emphasize that these quantitative contrasts are of an exploratory nature. We did not set up an experimental test of the two tools but rather took advantage of similarities in the two situations to draw a preliminary contrast. For instance, it is possible that the different project focus contributed to differences in the two sets of reactions. Nonetheless, these intial results are at the least suggestive that not only is wProjects meeting our higher-level wConnect requirements, but may be an improvement over the former tool.

Other data collected from the high school girls provides further evidence that the new wProjects tool is meeting our needs. The girls added from 3-7 records in their Career table (median 4); composed from 1-10 queries (median 2); and made from 1-3 web pages (median 1). Not surprisingly given the longer amount of time (90 rather than 50 minutes), the girls in the second workshop created more records and queries.

In addition to rating scales and simple production measures, we asked several open-ended questions, including explanations of the ratings for fun and feeling encouraged and things they liked most or least about the activity. The girls who worked with wProjects emphasized how the tool let them work with career interests (e.g., "*I enjoyed looking at all the different careers and what they require. I also liked being able to keep track of them in a nifty table. I'll be using this :-*"). Others focused more directly on the tool and the experience of building personal web projects (e.g., *"Because its very creative and fun to put your ideas in something like a website*"). These comments are just what we hoped to see, that the girls liked working with the tools and that they also felt a personal connection to their project content. At the same time, some girls are simply not interested in web development, even if they enjoyed the single session with us (e.g., *"I'm not into this sort of thing after I explored it"*). Of particular interest given our decision to change tool direction, one girl from the Bridgetools workshop raised tool access as an issue, "*The idea is cool to think about but I don't have the software to continue"*.

As a simple indicator of comprehension we included two probes about queries. One presented a hypothetical task of creating two web pages with different table content and asked how to do it. 10 of the 17 girls gave the right answer: create two queries, two web pages, and add one query to each. Six other girls were partially correct, some emphasizing web pages and others the queries. One said she did not know. A second question described updates to a table of data and asked what would happen if the page with the query were refreshed. 14 of the 17 correectly replied that the page would display the changed data; two incorrectly stated there would be no change and the final said she did not know. These results suggest that most but not all of the participants grasped the key concepts of query creation and use, including dynamic updates.

We also presented a simple "transfer" question, asking the girls to suggest other uses for a tool such as wProjects. All but one offered their own ideas, although several were variations on the college exploration theme. Other suggestions included a father's business data; book reading and associated notes; class schedules; recipes; places they visit; vocabulary notecards; grades; and other websites in general. These responses indicate that most girls understood the web project concept enough to apply it to other personal interests.

Finally, to help explain the range in subjective reactions, we repeated an analysis conducted with earlier workshop data [6], documenting the relationship of individual differences on a composite of the four workshop ratings. We used the entire set of 33 participants and created indices for computer playfulness [15], CIS career orientation [14], and workshop positivity. After first ensuring construct reliability, we regressed workshop positivity on playfulness and career ID. The resulting model accounted for 33.6% of the variance in positivity ($F(1,31)=8.1$, $p<.01$) and showed strong effects of both variables (β values were .45 and .36). This simple regression confirmed the pattern seen in our earlier work: pre-existing positivity about CIS careers, and intrinsic enjoyment when using computers are independent and important predictors of girls' reactions to the web development projects [6].

TABLE I. CONSTRUCTS USED IN INDIVIDUAL DIFFERENCE REGRESSION

Playfulness (7 items, α=.84)	**Career ID (3 items, α=.88)**	**Positivity (4 items, α=.83)**
5.1 (SD=.87)	4.3 (SD=1.0)	4.3 (SD=.67)

Note: scales ranged from 1-7 for Playfulness and Career ID, 1-5 for Positivity.

IV. TEACHER REACTIONS

In parallel with the two workshops using the Careers and Majors project (specified by wConnect members), we sought preliminary feedback from teachers who attempted to use the project specification features of wProjects. Recall that any authorized user can specify a project concept – this involves choosing a project name, writing a brief description of the learning goals, and deciding how many tables and what characteristics are needed in the database. In addition to field labels for table columns, the project designer also provides a prompt that will be associated with each input field to guide the eventual user (e.g., "Type your level of interest, for example High, Medium or Low").

We invited seven teachers with whom we had collaborated in some way in the past to participate. Two were from middle school and five from high school; four were male and three were female. In the end, five teachers (two males, three females) tried out the wProject design features and completed a brief feedback survey. They worked with the tool at their convenience, in response to an email invitation that contained an overview document (containing an example use of the tool) and links to the tool and the online feedback survey.

Although all five participants completed the feedback survey, only three of them (the females) completed an actual project specification. One project was aimed at exploring goals students have set in a learning enrichment program; a second was information about a novel programming tool the teacher has been investigating; the third was for tracking reactions and ratings to books read during independent study. One of the other teachers believed he had created a project (but he did not describe its goals); the other said he only got as far as thinking about the web page.

Not surprisingly, the teachers who successfully specified a project were quite positive about their experience. On a scale from 0-10, with 0=Never and 10=Many Times, they each chose "8" as their likelihood of using the tool again. The other two teachers both entered values of "3". They found the experience somewhat confusing, the tool not very easy to use, and the overview document lacking in useful examples. In contrast the three who created projects reported that the tool was quite easy to use and is likely to be useful by others (e.g., "*I think it is a great way quickly set up a project. I think others will be able to adapt and use it to meet their needs"* or *"I'd like to follow the project so that I can use it in my classroom"*).

With respect to concerns, several teachers expressed confusion about how to "test" their projects. In wProjects this is done by returning to the home page and choosing to "Create a Website" – the new project is now listed along with those specified by others. However this requires a designer to change mindset from that of a project designer to that of a project user (learner); we plan to add a short-cut to a project creation dialog for new projects as soon as they are saved. Two of the teachers also commented on the rigidity of the dialog: we use Drupal's database in a simple fashion to record the specifications as they are input and it is not possible to revise a project once a table has been specified (e.g., change its name or structure). We are working to find a more flexible solution for this.

Finally, several teachers felt that the overview and/or the tool itself would be more useful if a larger number and more diverse examples were included. This desire for more examples is consistent with other studies of end-user programming by teachers [16]. Adding more to the overview works against our general commitment to minimalist instruction [17], so we are hesitant to expand that document. However as wProjects goes into use more generally, many examples will be available.

As this summary conveys, our initial feedback from teachers was mixed. For the three teachers who grasped the concept, wProjects worked well; for other two it seemed mysterious and hard to use. Without a larger group of teachers and further probing it is unclear what might have led to these contrasting reactions. We hesitate to suggest gender-related explanations given the small sample size. However the teachers' pedagogical orientation may have contributed – for example, one of the women is an enrichment teacher, another is an award winning technology innovator, and the third is a high school AP computer science teacher who is exploring novel ways to attract rural students into computing. In contrast the two men were more conventional educators, one teaching business technology courses and the other advanced chemistry. Regardless, we should either target a more specific teacher-user population or revise the tool to better meet the expectations and needs of all teachers. One simple direction to consider is to first invite the teachers to take on a "student" persona, doing concrete work with a sample project. In fact this is a better example of active learning [17], and might do a better job of orienting and motivation teacher-designers than our carefully written overview document.

V. Discussion

The wProjects tool mitigates many issues we encountered when using Bridgetools: it can be accessed by anyone with Internet access, it leverages a large and active open source community, as well as a growing wConnect expertise, and it presents a simple and familiar web-forms interface. As a side benefit, workshop projects can now be created by any educator or mentor with an appropriately authorized wProjects account. Our initial trial suggests that the student outcomes are at the least equivalent and possibly better than what we achieved with Bridgetools. The teacher usage experiences were mixed but also hold promising signs that educators will be able to design wProjects activities useful for their teaching. We turn now to several new or ongoing issues raised by this research.

A. Tradeoffs in End-User Web Development Tools

Although wProjects seems to address our goals for broader dissemination and use of the workshop concept, the use of a Drupal platform has downsides. For instance, supporting end-user creation of web projects as part of a broad-featured CMS may not be the best way to communicate to learners that they are creating their own unique web applications. Although learners create personal content, their projects function as pieces of the encompassing wProjects application, borrowing its user interface themes and modules. In Bridgetools there was a cleaner distinction between the editing tools (i.e., the Java workspace) and the web application (opened and used in a distinct web browser). Although this is a genuine difference in the two approaches, wProjects users seem unconcerned, perhaps because they have no real concept of what "counts" as an independent web application. Nonetheless, we are working to increase the conceptual separation between wProjects and the websites it is used to create; one simple step is to store project URLs and make them accessible from outside the wProjects tool. This may be particularly attractive to students who want to show off their projects to friends or family.

With respect to dissemination and adoption by others, we have improved our position: anyone with a web browser and a wProjects account can design and/or create web projects, and any organization willing to install and maintain a Drupal server can create their own version or adaptation of wProjects. Nonetheless, installation and maintenance of the Drupal server is a cost that not all organizations (e.g., public schools) can bear, so our departmental server may end up hosting many projects if wProjects is successful and adopted widely; this may be an internal maintenance issue for us at some future date. Additionally, wProjects is linked to the Drupal community and the wConnect team must keep the modules up to date as future releases are made available; currently this is not an issue as community members have the expertise to handle the updates. Over time however wConnect may lose its "Drupal experts" causing problems for future maintenance. Finally, like any web-based application, wProjects must be regularly tested and updated as new web browers are released.

Our reuse of Drupal base functionality such as input forms has speeded development of wProjects, but has also produced a tabs and web-forms interface that we view as tedious, even while it is familiar to most web users. We would like to explore other input and output options, perhaps by finding or building tools analogous to `fckeditor` but optimized for creation or editing of tables and queries (e.g., the spreadsheet-like data editor and query wizards built for CLICK). In his dissertation work, Park is investigating a rather different paradigm for web development by novices, one that relies on a simplified approach to HTML and CSS [18][19]. Although this research is aimed at a somewhat different user population and goals, we will follow his progress intool design carefully to see if we can build from his ideas or tool components.

B. End-User Web Development as an Outreach Activity

In general, the workshops conducted by wConnect have been successful: participants have enjoyed the activities and most seemed to grasp the general concepts of data tables, queries, and dynamic web pages. At the same time, the web projects did not appeal equally to all girls – they were most effective for girls who arrived with a pre-existing identification with CIS, or a personal orientation of playfulness when using computers. One question as we continue is whether we can organize activities that are highly attractive to girls without these characteristics. For example, we may want to expand wConnect's outreach repertoire to activities other than web programming, perhaps media manipulation activities similar to those explored by Guzdial and his colleagues [3].

It is also possible that characteristics of the tools made the web programming less attractive to some girls. Beckwith and her colleagues [20] have made arguments of this sort in the context of spreadsheet debugging. Because of the outreach workshop format, it has not been possible to analyze the girls' use of the tools in detail and to see whether they are experiencing problems tied to their individual characteristics.

We also recognize that while the girls succeeded in building the dynamic web projects and many were able to answer questions about queries and updates in the feedback survey, we do not really know how much they learned (especially in any long-term sense) from their workshop experiences. If they went back to wProjects and started a different project, would they be able to succeed? We suspect so, but cannot be certain without further testing. Given many girls' readiness to suggest other application ideas, we are also curious to see whether and how well a high school girl can specify her own web project.

One disappointment is the relative lack of follow-through by participants. A few have joined wConnect, but do not attend online activities, contribute content, and so on. The wConnect team has been brainstorming ways to extend the engagement with these young women, for instance inviting a group of girls to help in mentoring another group. The general appreciation of the Careers and Majors project points to another idea – create web projects that are useful in and of themselves, so that girls may return to expand and use them.

C. Teachers as Web Project Designers

Although we obtained mixed feedback, we are encouraged by our early experiences with teachers designing new projects. For now, we will recruit teachers known for their technology innovation and education outreach, working with them to improve the project design dialog and documentation. One point we have already grasped is that most educators will not be motivated simply by the desire to teach students about dynamic web applications. Although this is the specific goal of the wConnect workshops, teachers working within their own context will have their own teaching goals. They may be perfectly content for the students to learn some web programming along the way, but their focus will be on science, literature or whatever content subject they teach.

We also need to learn more about the problems of the teachers who aborted their projects. One way to do this is to bring teachers in as lab study participants, where we can observe their efforts and reactions more closely. Another is to advertise wProjects more broadly but make its use contingent on completing a more detailed survey afterwards, or an interview where we can probe individual differences among users that might be leading to differential experiences.

VI. CONCLUSION

wProjects appears to be a reliable and useful tool, but we have many enrichments planned. In addition to improving the web-forms user exeprience, wConnect members are creating a module that will allow designers to see all of the learners who complete their projects and other associated data. We are also planning to allow collaboration across wProjects, so users can see and potentially merge with other users' content, creating queries for a richer dataset. We are also eager to integrate wProjects and its users into the online community, giving users the option to share their projects with others. Once we do this, we hope to use wProjects in a more direct and pervasive fashion to attract and engage young women in CIS activities.

ACKNOWLEDGMENT

We thank the wConnect team members who organized the workshops, student and teacher participants, Dejin Zhao and Nan Zang, who helped with the early prototyping of wProjects. This work was supported by the NSF as part of the Broadening Participation in Computing initative, CNS-0634337.

REFERENCES

[1] U.S. Department of Education, NCES, Higher Education General Information Survey (HEGIS), "Degrees and Other Formal Awards Conferred" surveys, 1970-71 — 1985-86; 1986-87 — 2007-08.

[2] L. Barker, C. McDowell and K. Kalahar, "Exploring factors that influence computer science introductory course students to persist in the major", in Proceedings of SIGCSE 2009, pp. 153-157.

[3] M. Guzdial, "A media computation course for non-majors," in Proceedings of SIGCSE 2003, pp. 104-108.

[4] M.B. Rosson, H. Sinha, D. Zhao, and J.M. Carroll, "wConnect: A developmental community for women in computer and information science," in Advanced Learning Technologies, 2010.

[5] M.B. Rosson, A. Ioujanina, et al., "A scaffolded introduction to dynamic website development for female high school students," in SIGCSE 2009.

[6] M. B. Rosson, H. Sinha, T. Hansford, and J. Mahar, "Offering early success experiences in software construction.," in ICALT 2008 Washington, DC, 2010, pp. 458-460.

[7] M. B. Rosson and J. M. Carroll, "Developmental learning communities.," Journal of Community Informatics, vol. 2, 2006.

[8] W. F. Whyte, "Advancing scientific knowledge through participatory action research.," Sociological Forum, vol. 4, pp. 367-385, 1989.

[9] M. B. Rosson, E. Thiry, D. Zhao, and J. M. Carroll, "Developing an online community for women in computer and information sciences: A design rationale analysis.," presented at the HICSS 44, 2011.

[10] J. Rode, *User-Centered Design of End-User Web Development Tool*. PhD Dissertation, Virginia Tech, 2005.

[11] M.B. Rosson, H. Sinha, M. Bhattacharya, and D. Zhao, "Design planning in end-user web development," *JVLCs*, 19, 468-484.

[12] M.B. Rosson, D. Dunlap, P. Isenhour, and J. Carroll, "Teacher Bridge: Building a community of teacher developers. *Proceedings HICSS 42*.

[13] M.B. Rosson, E. Thiry, D. Zhao and J.M. Carroll, "Teachers as designers: A tool enabling teachers to specify dyamic web projects for students," in International Symposium on End-User Development: IS-EUD 2011, in press.

[14] M.B. Rosson, J.M. Carroll and H. Sinha, "Orientation of undergraduates toward careers in computer and information science: Gender, self-efficacy and social support," ACM TOCE, in press.

[15] S. Wiedenbeck, S. Xiaoning, and T. Chintakovid, "Antecedents to End Users' Success in Learning to Program in an Introductory Programming Course," in Visual Languages and Human-Centric Computing, 2007. VL/HCC 2007. IEEE Symposium on, 2007, pp. 69-72

[16] S. Wiedenbeck, "Facilitators and inhibitors of end-user development by teachers in a school," in Visual Languages and Human-Centric Computing, 2005 IEEE Symposium on, 2005, pp. 215-222.

[17] J. Carroll, The Nurnberg Funnel: Designing Minimalist Instruction for Computer Skills, MIT Press, 1990.

[18] T.H. Park, "Creativity support in web design tools for end-user developers," in IEEE 2009 Visual Languages and Human-Centric Computing: VL/HCC 2009, Corvallis, OR.

[19] T.H. Park and S. Wiedenbeck, "First steps in coding by informal web developers," in IEEE 2010 Visual Languages and Human-Centric Computing: VL/HCC 2010, Leganes-Madrid, Spain.

[20] L. Beckwith, S. Sorte, M. Burnett, S. Wiedenbeck, T. Chintakovid and C. Cook, "Designing features for both genders in end-user programming environments," in IEEE 2005 Visual Languages and Human-Centric Computing.

Facilitating Peer Review in an Online Collaborative Learning Environment for Computer Science Students

Cen Li, Zhijiang Dong, Roland H. Untch, and Michael Chasteen
Department of Computer Science, Middle Tennessee State University, Murfreesboro, TN, USA

Abstract - *High student drop out and failure rates in entry-level computer science (CS) courses have contributed to a lower number of qualified CS graduates nationwide. Among various underlying causes that lead to this phenomenon, several unsatisfactory behavioral traits have been identified in CS students, including lack of motivation and combativeness towards the opinions of peers. Many CS educators have attempted to address these problems by developing learning tools to increase students' motivation through engaging educational activities. An alternative, and possibly better, solution is to combine learning activities with social activities to create an engaging learning environment that foster collaboration among student peers. In this work, we experiment with this idea by developing a peer review system as an integral component of an online social network that features many of the collaborative features found in popular websites, such as Facebook. Results from controlled experiments and student feedbacks indicate that not only did students' learning performance increases with use of the system, so did the students' sense of community.*

Keywords: peer review; peer learning; computer supported collaborative learning; online social network; CS2

1 Introduction

Distressingly high rates of failure and dropout have been reported nationwide among students enrolled in entry level computer science (CS) courses [13]. Studies investigating this problem have identified a lack of motivation, persistence, and passion towards course material, procrastination on assignments, unwillingness to support or aid others, and combativeness towards the opinions of peers as contributing factors [27]. To mitigate some of these factors, educational researchers have found that peer review and the fostering of learning communities can effectively enhance the learning process and make these courses more enjoyable to the students ([17], [23]). However such enhancements come with a cost of time and effort. This paper describes a successful system (*PeerSpace*) that, using web technology to reduce burdensome administrative overhead, integrates peer review into a peer-friendly, collaborative social learning environment. We report on our findings from experiments that peer review, as facilitated using PeerSpace, improves student learning outcomes. Additionally, we observe that students using PeerSpace have a demonstrably higher sense of community than those who do not. Section 2 provides background information on peer review and the PeerSpace system. It is also here that we state our conjectural thesis that peer review integrated into a peer-friendly learning environment is effective in improving student learning and promulgating a sense of community. Section 3 deals with the mechanics of peer review in PeerSpace. Section 4 contains the results of our experiments with our approach. Section 5 concludes the paper and suggests future work.

2 Background

2.1 Peer Review

In a course setting, peer review is a process wherein students evaluate the creative work or performance of other (peer) students and give constructive feedback [24]. It presents students with the opportunity to learn through evaluation and reflection [2], thereby strengthening the overall cognitive thinking capacity of participating students.

The most common use of peer review is found in English composition courses where peers swap essay drafts with each other and leave feedback in the form of corrections and suggestions to the author. Students who participate in peer review are shown to be better readers and have developed stronger self-editing skills. With guidance from their teachers, the students learn to perform reviews that are constructive and helpful rather than destructive and harsh, which benefits students receiving feedback and cultivates their ability to accept criticism from others.

Peer review has been applied to many fields of study such as biology, psychology, nursing, and computer science with successful results [24]. The most common occurrence of peer review in computer science appears in introductory CS courses ([10], [11], [18], [21], [25], [26], [28]). These courses typically appear in the first year and introduce programming in a language such as C++ or Java. Results in this area are very good and range from an increased sense of community [10] to better understanding of the material presented [18]. This success has been carried over to more advanced courses such as operating systems [6], computer architecture [7], scientific writing [1], web technologies [9], software engineering [29], parallel processing [8] and game design [19]. Peer review has also been used in atypical manners, for example, in a compiler course [22] where groups can borrow code from other students to use as a framework for the final portion of a term project if the group's own framework didn't turn out as well as expected.

In a typical peer review process, each student occupies two roles: as a *peer reviewer* to evaluate other students' work and as an *author* to receive feedback from peers. As a peer reviewer, each student is required to practice critical analysis skills and give constructive feedback. Additionally, peer reviewers have the opportunity to see multiple solutions to the same problem. In the context of computer science, a student performing a review learns to appreciate good programming and good documentation as they strive to understand the work they are reviewing; consequently these students are more likely to practice better programming and documentation in their future work. As an author, each student will receive multiple feedbacks from peers, and learn to accept constructive criticisms and suggestions from their peers. It has been observed that students tend to be more willing to accept criticism from their peers than from the instructor. Learning from their peer's programs and suggestions can be more effective in improving the quality of their own programs. Additionally, this process generates a greater sense of community because students are helping each other evolve into a more mature and skilled programmer.

Next we review a number of existing systems developed to support peer review.

2.2 Related Work

In an effort to improve an already verified method, Hundhausen et al. moved their paper based peer review to an online studio-based learning environment (OSBLE) [11]. The system supports and uses both individual and group review techniques. Reviews are done individually before the group comes together to create one review out of all the members' reviews. Students can place comments on specific lines or line ranges, along with marking the severity of the problem, and then drag them over to the group review if the group likes the assessment that was made. The students can do individual reviews from any computer, but the group reviews are done during lab time under a moderator. After the review process, students can view their feedback and resubmit their work. It is important to note that OBSLE does not support any kind of anonymous reviewing and does not use a rubric like most implementations. While this is in contradiction with most methods, results from the system seem to have been positive. Overall, Hundhausen et al. report that the transition to an online environment sped up the review process and made reviewing work a more efficient process for students. However, OSBLE is more than just a peer review tool as it supports basic course management. Future work has been planned to add more social features to OBSLE such as profiles and giving authors a way to rate the feedback they received.

Harri Hämäläinen et al. originally experimented with an open-source peer review system called MyReview [16] for a master's level course [8]. They concluded that advanced students could see a benefit in looking at each other's code. But the use of a system not designed with code review in mind was too distracting. Therefore, Ville Hyyrynen et al. (including Hämäläinen) decided to create a custom implementation of a peer review system called MyPeerReview ([12], [15]). This design is different from the previous examples because it is built off of an existing content management system (CMS) called Drupal [4] which already has basic features including users, profiles, and navigation tools. Because of this, only a module was needed to add support for assignment submission and peer review. The system itself does not support anonymous peer review, but the students were asked, not forced, to remove identifying information from their files before making a submission. Reviews are done individually and guided by a rubric constructed inside Drupal. In reference to the procedure, instructors have to manually "open" an assignment for students to submit work, and put it on "hold" when the deadline is reached. There were not a lot of features implemented in MyPeerReview initially, but the results were positive. Students felt like peer review was a worthwhile assignment, and the process of submitting and reviewing assignments went smoothly. However, several improvements were suggested, including adding a rating system, automated timing for deadlines, and using a rubric more similar to what the instructor uses.

Like the plans for online studio-based learning environment (OSBLE) developed by Hundhausen et al. [11] and the MyPeerReview system developed by Ville Hyyrynen et al. ([12], [15]) the peer review system developed for this work is part of a larger system that supports many features in addition to peer review. Like the approach taken by MyPeerReview, a content management system (CMS) is used as the basis of the system development. Elgg CMS [5] was chosen because it comes with many desirable social network features such as user profile, blog, and forum.

2.3 PeerSpace

PeerSpace ([3], [14]) is an innovative online collaborative learning environment developed for CS education with the purpose of bringing real-world activities into a virtual environment. One of the major benefits of this is the elimination of time and effort wasted in organizing an event because of scheduling or other conflicts found in real-life. PeerSpace's asynchronous, online environment creates a more efficient work flow that allows students to participate in tasks such as a discussion or group assignment without the instructor or other students being present.

PeerSpace is made up of individual components, called modules, that are classified as either social or learning modules. The corresponding social and learning activities supported by the modules interact as depicted in Figure 1.

The s*ocial* modules provide the students with a variety of tools to support asynchronous and synchronous peer communication. The social modules stimulate the development of long-term social connections for CS students and support features found in popular social networking sites, such as Facebook, including profiles, statuses, blogs, forums, chats, and games. These modules are designed to help the students in building peer support and trust and thus help the students feel they are part of a learning community. These are essential in making collaborative learning among student peers successful and enjoyable.

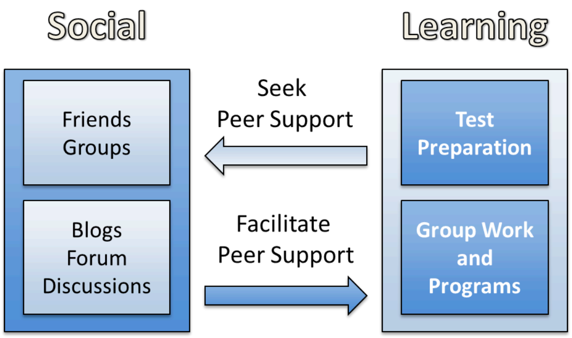


Figure 1. Module interaction model

The *learning* modules provide a set of tools that support collaborative learning activities such as group homework, peer code review and collaborative programming. The shared knowledge these activities instill affords the students a deeper understanding of course-related concepts and a broader knowledge of course-related topics. The idea of putting the learning activities amid the social activities strengthens the support among students and expands a student's social network with new friends.

It is our conjecture that peer review, a proven learning approach, integrated into a peer-friendly, collaborative learning environment is effective in improving student learning while providing a constructive atmosphere for students to develop a sense of community.

3 Peer Review in PeerSpace

3.1 System Architecture

The two main components supporting peer review in PeerSpace are the Assignment module and the Peer Review module. Both modules interact with the Subversion (SVN) server through the Grading Using Subversion (GUS) component. Each module provides two distinguished sets of features for two different types of users, i.e. the instructors and the students. Figure 2. illustrates the architecture of the peer review tool in PeerSpace.

The Assignment module handles the process of creating and submitting assignments. Instructors can create assignments through the instructor side of the Assignment module. Once an assignment is created, students can submit one or more of those assignments. The submissions are stored in the SVN server for the instructors to retrieve and grade. The instructors can subsequently put graded/marked assignments back to the SVN server so that students can obtain them through the Assignment module.

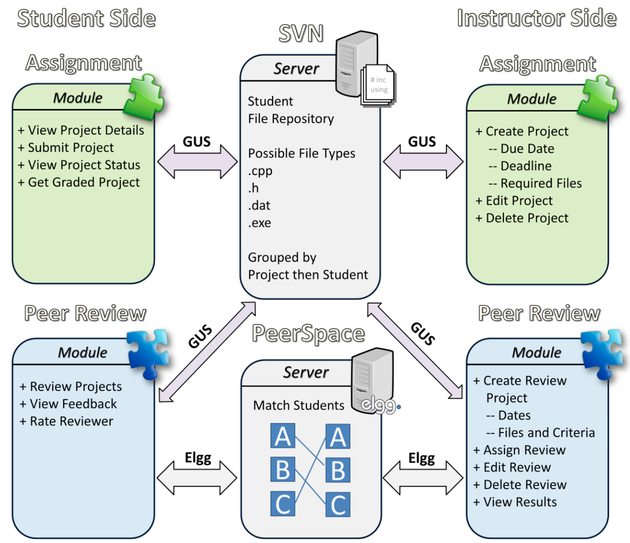


Figure 2. Architecture of Peer Review

3.2 Procedure to Perform Peer Review

The Peer Review module allows the students to review their classmates' assignments. First, an instructor creates a peer review assignment of an existing regular assignment for his/her class in the Peer Review module. Instructors must provide two dates for each peer review assignment: assign date that specifies the time to assign students' works to reviewers, and deadline date marks the end of the allotted review time. Then when the assign date reaches, the Peer Review module will assign students' submissions to reviewers. During this phrase, all submitted assignments will be retrieved from the SVN server and. copies of these assignments are stored on the PeerSpace server for the reviewers to view or download. Now reviewer can perform the evaluation on assigned works before the deadline. All completed peer reviews are stored in the Elgg database.

For CS projects, a student typically needs to submit a number of files, including source code, data files, and executable files. For peer review, instructors can specify which of the files and any extra files students should have for reviewing. Additionally, instructors can specify the files as viewable files, downloadable files, or both. Files marked as viewable are converted into images to prevent plagiarism. Since the peer review process is anonymous, all identification information is removed from viewable files during conversion. Students can retrieve downloadable files. Furthermore, instructors can specify files to be used in the review process that were not listed in the required files for the assignment. This allows instructors to do things such as upload special test cases from their personal computer or select a file that was added to the students' repositories after the submission deadline.

Students may be assigned to review one, two, or three programs for each review assignment. This allows students a better chance to learn through the peer review process because they can see more than one additional response to the assignment and receive more feedback about their solutions. The form to create peer review assignments is shown in Figure 3.

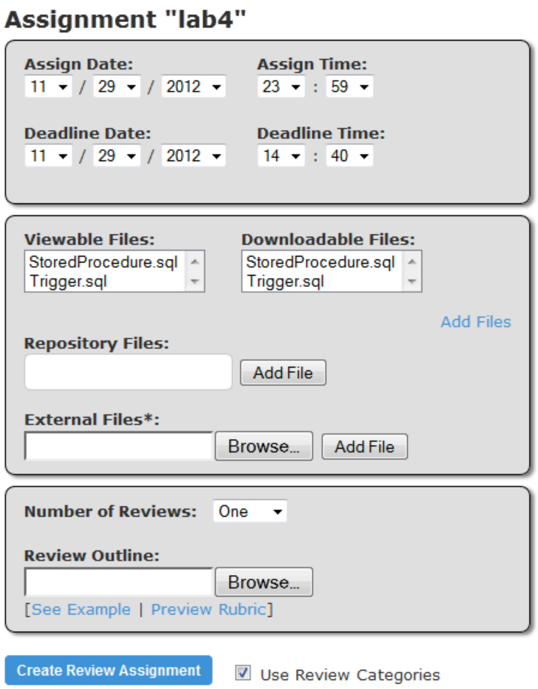


Figure 3. Peer review assignment creation form

Instructors can guide the peer review process by providing a review rubric for students to follow when performing a review. This allows customization of peer review assignments to fit the assignment being reviewed. The rubric allows instructors to create custom point values for criteria provided in the rubric as well as section headers to control the flow of the review and comments to clarify how to assess specific parts of the assignment. Instructors can preview how their rubric will be displayed to the students.

3.3 Manage and Monitor Peer Review Assignments

Once a peer review assignment has been created, it can be assigned to the students manually or automatically. If the instructor wants to assign the peer review assignment immediately after creating it, then a peer review can be assigned manually immediately. Otherwise, it can be assigned automatically at pre-specified time.

After a peer review assignment has been created, it can be edited if small changes need to be made. If the peer review assignment has yet to be assigned, then instructors can alter the assign date and time. The instructor can also change the deadline of the peer review assignment.

Peer Review assignments can be deleted at any time. Deleting a peer review assignment before it was assigned will simply delete the template used to assign the peer reviews to the student. If the peer review assignment is deleted after it has been assigned then the entire peer review assignment will be purged from the system, including all of the students' files and peer reviews. Peer review assignments can be deleted one at a time or collectively. The collective delete operation can be used at the end of a semester to clear all student assignments from PeerSpace.

After a peer review assignment has been assigned, instructors can view the progress of the class as well as the individuals through the "Grade" view. A list of all assigned peer reviews for a particular assignment is available in the Peer Review module for the instructor to see which reviews have been completed. A fully completed review listing contains the author's name, picture, and score the author received as well as the reviewer's name, picture, and rating the reviewer received. The list of peer reviews can be sorted by author, score, reviewer, or rating to make it easier to view the progress and performance of the review assignments.

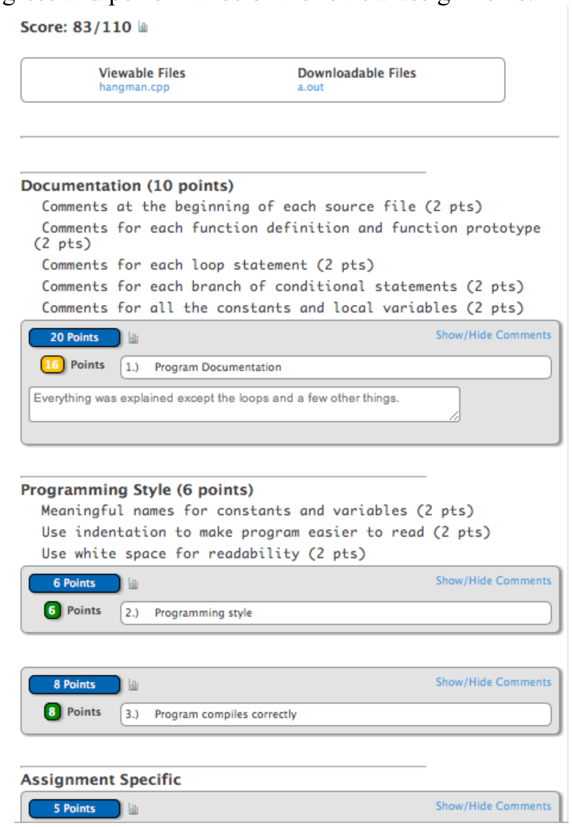


Figure 4. Partial view of a peer review completed by a student

The instructor can view an individual peer review assignment as soon as it has been reviewed. The instructor view contains all of the author's work, the reviewer's assessment and comments, as well as the score. Figure 4 shows a partial view of a peer review completed by a student.

After the review, the author has a chance to rate the feedback he or she received. The rating comes from the author's perceived quality of the review. The author does this by evaluating the reviewer's accuracy, helpfulness, knowledge, and fairness. Each category used in the rating process can be given a score of one at the lowest and five at the highest. Therefore, the lowest possible rating is a four and the highest is a twenty. Both the score and the rating are shown to the instructor when an individual peer review assignment is viewed. An example can be seen in Figure 5.

Figure 5. Results of an individual peer review assignment

4 EXPERIMENTAL ANALYSIS

Students from two sections of CS2 participated in this study. The same instructor taught both sections, and the same set of assignments and exams were used. Each section has 27 students. One section is chosen as the experiment group and the other as the control group at the beginning of the semester. Students from both groups have access to PeerSpace and all its tools. The only difference between the two is: students in the experiment group performed peer code review in their class with the support using the peer code review tool of PeerSpace, while no peer code review was performed in the control group. Peer code review started from the first part of the semester. Each student was required to review 2 programs submitted by other students in the class. During the semester, average scores on programming assignments, tests, and final course averages were collected and compared between the experiment and the control sections. Furthermore, a Modified Group Environment Questionnaire (MGEQ) was given to students in both sections at the end of the semester to collect and compare their sense of group learning environments. The Modified Group Environment Questionnaire (MGEQ) [20] features four subcategories that consider the individual attraction to group activities and group integration aspects of both social actions and learning tasks. These categories are made from groupings of the eighteen questions found in the MGEG. Table 1 lists all the questions under their respective categories.

All questions use a Likert scale with "strongly agree" having a value of 1 and "strongly disagree" having a value of 5. For questions 1, 2, 3, 4, 6, 7, 11, 13, 14, 17, and 18, "strongly disagree" is desired so a higher score is better. Therefore, a lower score is better on all the remaining questions.

Each category has an overall value calculated by adding the results from each question in the category together. Questions where "strongly agree" is desired are flipped so that all questions are uniform in their worth. This allows comparisons to be done for each category. Both grades and survey results are analyzed using a two-tailed t-test with a threshold of 0.1.

Table 1. MGEQ questions in four categories

#	**A. Individual Attraction to Group (Social):**
1	I do not enjoy interacting with members of this class.*
3	I am not going to miss the members of this class when the semester ends.*
5	Some of my best college acquaintances/friends are from this class.
7	I enjoy the atmosphere in other classes more than this class.*
	B. Individual Attraction to Group (Learning):
2	I'm unhappy that this class is not challenging enough.*
4	I am unhappy with the amount of effort my fellow classmates devote to this class.*
6	This class does not give me enough opportunities to improve my programming skills.*
8	I like the way group activities enhance the class.
9	This class is one of my most important college classes.
	C. Group Integration (Social):
11	Members of our class would rather work alone than work in groups.*
13	Class members rarely interact with one another.*
15	Members of this class would enjoy spending time together outside of class.
17	Members of this class do not interact outside of lectures and labs.*
	D. Group Integration (Social):
10	Class members help one another.
12	Instead of blaming someone else, members of our class generally take personal responsibility if they are unable to complete an assignment.
14	Class members have conflicting aspirations for what they want out of the class.*
16	If members of this class have problems with the material, other members of the class express concern and help them with these problems.
18	Class members are reluctant to communicate with one another.*

4.1 Study the effectivenss of peer code review on improving student learning

Table 2 presents the average scores on programming assignments, tests, and final course averages of the two groups. Statistically significantly better student learning performance was observed in the experiment class in all three areas. The programming assignments (PA) were examined individually to see if peer review had an effect from assignment-to-assignment. Because the experiment section had a higher average on every PA, the progression and details of each PA were reviewed. Table 3 contains the average of each PA for the control and experiment sections. The first peer review assignment was over PA2. Therefore, it is only to be expected that the first two PAs show little difference between the control and experiment section. More importantly, PA3, PA7, PA8, and PA9 are all statistically significant when compared to the control section. As the experiment section completed more peer review assignments, the difference between the two sections grew larger. These results show that peer code review activity lead to better student performance in programming assignments as well as in the overall learning in the course.

Table 2. Average scores of the experiment and the control section students

	Control Avg. (Std.)	Experiment Avg. (Std.)	T-Test
Programs	73.05 (21.84)	88.93 (12.37)	***0.013***
Tests	65.08 (26.76)	82.29 (17.73)	***0.033***
Overall	67.63 (25.03)	87.09 (11.88)	***0.006***

Table 3. Average scores of the 9 programming assignments from in the experiment and control sections

CS2 PAs	Control Avg. (Std.)	Experiment Avg. (Std.)	Difference	T-Test
PA 1	92.22 (17.41)	94.78 (19.58)	02.56	0.691
PA 2	92.38 (08.70)	93.78 (10.82)	01.40	0.682
PA 3	89.31 (11.24)	95.72 (09.30)	06.41	***0.078***
PA 4	73.31 (32.96)	82.00 (34.84)	08.69	0.462
PA 5	71.38 (37.45)	85.94 (25.48)	14.57	0.190
PA 6	72.94 (37.59)	85.00 (23.46)	12.06	0.264
PA 7	58.19 (41.94)	92.39 (09.91)	34.20	***0.002***
PA 8	57.38 (47.78)	90.06 (24.14)	32.68	***0.015***
PA 9	50.31 (44.94)	80.67 (31.71)	30.35	***0.028***

4.2 The effectiveness of peer review on improving students' sense of collaborative learning environment

Responses from the CS2 students in the experiment group show, at the end of the semester, they have a significantly higher sense of learning community and better group learning environment than those in the control group. Group environment survey results (Table 4) show the difference observed in the mean values of Individual Attractions to the group (social), Group Integration (social), and (c) Group Integration (learning) are statistically significant between the two groups, i.e., the two tailed t-test results are less than 0.1. T-test results along individual questions show the student response from the experiment group are significantly higher than those from the control group along the following set of questions:

a. I do not enjoy interacting with members of this class.
b. I am unhappy with the amount of effort my fellow classmates devote to this class.
c. Class members help one another.
d. Members of our class would rather work alone than work in groups.
e. Class members rarely interact with one another.
f. Members of this class would enjoy spending time together outside of class.
g. If members of this class have problems with the material, other members of the class express concern and help them with these problems.
h. Members of this class do not interact outside of lectures and labs.

This shows that the students in the experiment group interact more with their classmates, are more willing to work collaboratively, and are willing to help each other, thus a better overall sense of classroom learning community.

Table 4. CS2 MGEQ subcategory results

Category	Control Avg. (Std.)	Experiment Avg. (Std.)	T-Test
Individual Attraction to Group (Social)	11.77 (2.12)	13.94 (2.79)	***0.0290***
Individual Attraction to Group (Learning)	18.23 (2.83)	19.63 (2.98)	0.2117
Group Integration (Social)	10.00 (4.18)	15.00 (2.97)	***0.0008***
Group Integration (Learning)	16.30 (3.75)	18.88 (2.55)	***0.0375***

4.3 The effectiveness of peer review on improving students' sense of community

Table 5 compares student sense of community between the experiment and the control group by comparing the effects on individual attraction to group and group integration along learning (Learning in community) vs. social interaction (Connectedness in community). It is observed that the students in the experiment group have a significantly higher sense of belonging to a learning community than the students from the control group. While the response also shows the average sense of connectedness among the students in the experiment group is slightly lower than that from the control group, the overall sense of community among the experiment group is still higher (though not statistically significantly higher).

Table 5. Comparing student responses about their sense of community between the control and the experiment groups

	Control Avg(sdv)	Experiment Avg(sdv)	T-Test
Connectedness in community	32.6(6.9)	31.9(2.3)	0.72
Learning in community	29.8(5.8)	33.1(3.3)	**0.06**
Overall sense of community	62.4(9.5)	65(4.0)	0.32

5 Conclusions and Future Work

In this paper, we described an automated peer review system comprising a complete suite of instructor tools to facilitate a peer review process. We promulgated an approach that combines peer review with a social network to create a centralized place for students to complete work and contribute to the community, allowing social and learning components to feed each other and drive students. We observed that this allowed for a peer review process that is seamless and provides both instructors and students with a continuous feel.

Additionally we provided the results of experiments that sought to ascertain effectiveness of peer review on student learning and on promoting collaborative learning. The results show our approach to peer review benefits students both in learning and in supporting a peer-friendly collaborative environment. Results from the learning performances show the experiment section had significantly higher grades in all types of assignments as well as a better sense of community. We conjecture that the sense of community grew because peer review is a collaborative learning activity that fosters constructive criticism that pulls students closer together.

As future work, we plan to give instructors the ability to manually pair peer reviewers; this would allow instructors to exercise greater control over the peer review process. Additionally, we are continuing to gather data from controlled experiments in additional course offerings and subjects with a view of establishing greater external validity of our positive results.

6 ACKNOWLEDGMENT

This work is funded by the NSF CCLI/TUES grant (DUE-0837385), NSF REU grant (IIS-1038172), and the Sponsored Research program at MTSU.

7 REFERENCES

[1] C. Bauer, K. Figl, M. Derntl, P. P. Beran, and S. Kabicher, "The student view on online peer reviews," *SIGCSE Bull.* 41(3), pp. 26-30, July, 2009.

[2] B. Bloom. (1956).Taxonomy of Educational Objectives. Handbook I: Cognitive Domain. David McKay Co..

[3] Z. J. Dong, C. Li, and R. H. Untch, "Build peer support network for CS2 students," In *Proceedings of the 49th Annual Southeast Regional Conference*. Kennesaw, GA, USA, pp. 42-47, 2011.

[4] Drupal. Retrieved February 29, 2012, from Drupal: http://drupal.org.

[5] Elgg. Retrieved February 29, 2012, from Curverider Limited: http://elgg.org.

[6] E. F. Gehringer, "Electronic peer review and peer grading in computer-science courses," *SIGCSE Bull.* 33:1, pp. 139-143, February, 2001

[7] E. F. Gehringer, "Building resources for teaching computer architecture through electronic peer review," In *Proceedings of the 2003 workshop on Computer architecture education: Held in conjunction with the 30th International Symposium on Computer Architecture (WCAE '03).* San Diego, CA, USA, Article 9, 2009.

[8] H. Hämäläinen, J. Tarkkonen, K. Heikkinen, J. Ikonen, and J. Porras. "Use of Peer-Review System for Enhancing Learning of Programming," *Ninth IEEE International Conference on Advanced Learning Technologies (ICALT '09)*, Riga, Latvia, July 15-17, pp. 658-660, 2009.

[9] H. Hämäläinen, V. Hyyrynen, J. Ikonen, and J. Porras, "MyPeerReview: an online peer-reviewing system for programming courses," In *Proceedings of the 10th Koli Calling International Conference on Computing Education Research*, Berlin, Germany, pp. 94-99, 2010.

[10] C. Hundhausen, A. Agrawal, D. Fairbrother, and M. Trevisan, . "Integrating pedagogical code reviews into a CS 1 course: an empirical study," In *Proceedings of the 40th ACM technical symposium on computer science education*. Chattanooga, TN, USA, pp. 291-295, 2009.

[11] C. Hundhausen, A. Agrawal, and K. Ryan. "The design of an online environment to support pedagogical code reviews," In *Proceedings of the 41st ACM technical symposium on computer science education*. Milwaukee, WI, USA, pp. 182-186, 2010.

[12] V. Hyyrynen, H. Hämäläinen, J. Ikonen, and J. Porras. MyPeerReview: an online peer-reviewing system for programming courses. In Proceedings of the 10th Koli Calling International Conference on Computing Education Research, Berlin, Germany, pp. 94-99, 2010.

[13] P. Kinnunen, and L. Malmi, "Why students drop out CS1 course?" In: *Proceedings of the 2nd International Workshop on Computing Education Research*. Canterbury, UK, pp. 97-108, Sept. 2006.

[14] C. Li, Z. J. Dong, R. H. Untch, M. Chasteen, and N. Reale, "PeerSpace - An Online Collaborative Learning Environment for Computer Science Students," *IEEE 11th International Conference on Advanced Learning Technologies,* Athens, GA, pp. 409-411, 2011.

[15] MyPeerReview. Retrieved February 29, 2012, from SourceForge: http://sourceforge.net/projects/mypeerreview/.

[16] MyReview. Retrieved February 29, 2012, from SourceForge: http://myreview.sourceforge.net.

[17] A. Perez-Prado, and M. A. Thirunarayanan, "Qualitative comparison of online and classroom-based sections of a course: Exploring student perspectives," *Educational Media International*, 9(2), pp.195-202, 2002.

[18] K. Reily, P. L. Finnerty, and L. Terveen. "Two peers are better than one: aggregating peer reviews for computing assignments is surprisingly accurate," In *Proceedings of the ACM 2009 international conference on Supporting group work (GROUP '09)*, Sanibel Island, FL, pp. 115-124, 2009.

[19] A. Repenning, A. Basawapatna, and K. H. Koh, "Making university education more like middle school computer club: facilitating the flow of inspiration," *In Proceedings of the 14th Western Canadian Conference on Computing Education (WCCCE '09)*, Roelof Brouwer, Diana Cukierman, and George Tsiknis (Eds.), Burnaby, British Columbia, USA, pp. 9-16, 2009.

[20] A. P. Rovai, "Development of an instrument to measure classroom community," *The Internet and Higher Education*. 5:197-211, 2002.

[21] J. Sitthiworachart and M. Joy, "Effective peer assessment for learning computer programming," In *Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education*. Leeds, UK, June 28-30, pp. 122-126, 2004.

[22] H. Sondergaard, "Learning from and with peers: the different roles of student peer reviewing," In *Proceedings of the 14th annual ACM SIGCSE conference on Innovation and technology in computer science education (ITiCSE '09)*, Paris, France. pp. 31-35, 2009.

[23] K. Swan, "Building learning communities in online courses: the importance of interaction," *Education Communication and Information*, 2(1), pp. 23-49, 2002.

[24] K. Topping, "Peer Assessment Between Students in Colleges and Universities," *Review of Educational Research*, 68(3), pp. 249 -276, 1998.

[25] A. Trivedi, D. C. Kar, and H. Patterson-McNeill, "Automatic assignment management and peer evaluation," *J. Comput. Small Coll.* 18(4), pp. 30-37, April, 2003,.

[26] D. A. Trytten, "A design for team peer code review", In *Proceedings of the 36th SIGCSE technical symposium on computer science education*. St. Louis, MO, USA, pp. 455-459, 2005.

[27] W. M., Waite, and P. M. Leonardi, "Student culture vs. group work in computer science," In *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education*, Norfolk, VA, USA, pp. 12-16, March 2004.

[28] Y. Q. Wang, Y. J. Li, M. Collins, and P. Liu, "Process improvement of peer code review and behavior analysis of its participants," *SIGCSE Bull.* 40(1), pp. 107-111, March 2008.

[29] W. J. Wolfe. "Online student peer reviews." In *Proceedings of the 5th conference on Information technology education*, Salt lake city, UT, USA, pp. 33-37, 2004.

Strategies for Teaching Ideation and Ethics in Computer Science

Kendall E. Nygard[1], Lisa Bender[1], Gursimran Walia[1], Kevin Brooks[2], Krishna Kambhampaty[1], and Travis E. Nygard[3]
[1]Department of Computer Science, North Dakota State University, Fargo, North Dakota, USA
[2]Department of English, North Dakota State University, Fargo, North Dakota, USA
[3]Department of Art, Ripon College, Ripon, Wisconsin, USA

Abstract -- *Although social, ethical, and professional standards have been included in some undergraduate computer science departments' curricula for over twenty years, many faculty members who teach computer science ethics classes continue to look for fresh approaches to teaching ethics and social implications courses. Traditionally, this type of course has focused on viewing computer technologies through an ethical lens. However, there is a movement towards placing more emphasis on the social implications of technology. The global use of computer technologies continues to develop and rapidly change. This rapid growth affects everyone's lives in fundamental ways, which motivates an increased emphasis on social implications. Facebook and related social media systems are prominent examples of the social impact phenomenon. The IEEE/ACM Computer Science Curriculum 2013 has acknowledged this shift. Students in computer science and information technologies need to develop a mature appreciation for the societal impacts of the technologies that they likely will have a part in creating and maintaining. They must also learn how to appreciate diversity and multiple perspectives as well as to reason about and apply ethical decision making strategies to ethical problems they may encounter. This paper discusses a successful course in ethics and social implications that helps students develop their ethical reasoning skills and an appreciation for the complex impact that technologies have on our society.*

Keywords: ethics; social implications, education; team projects

1 Introduction

All computer science academics will agree that technical issues are central to any computing curriculum. However, just learning technical theory and skills is not enough and does not constitute a complete educational program in any computing field. Students must also develop the communication skills, teamwork skills, and ethical analysis skills required of practicing professionals. They must also develop an appreciation and understanding of the social and professional context in which computing is done and the important part they will play in the process. Students must develop the ability to ask serious questions about the social impact of computing technologies as well as the ability to evaluate the possible answers to those questions. Ideation is the process of generating, developing and communicating ideas. Within the structure of our course, we b elieve that ideation is essentially the process of the student gaining maturity in their perspectives of social and ethical issues by generating, developing, and communicating their ideas in a team project setting. "Future practitioners must be able to anticipate the impact of introducing a given product into a given environment. Will that product enhance or degrade the quality of life? What will the impact be upon individuals, groups, and institutions?" [1] The computer science faculty members at North Dakota State University (NDSU) have had these same sentiments for several years prior to this famous excerpt from the Computing Curricula 1991. Social Implications of Computers has been taught as a stand-alone course at NDSU for nearly 25 years. The reason for the sentiment is simple: computer technologies affect every society and our computing students need to learn how to reason about the social ramifications and ethical problems they may be confronted with, and stand against the pressures to be unethical. At NDSU the course has large enrollment, often in the vicinity of 100 students. Since project teams are formed with typically three students, we often have more than 30 teams active in a semester. While many universities have embraced the inclusion of ethics into the undergraduate computer science curriculum, some universities have not, and not all universities with ethics curricula have stand alone courses [2, 3]. IEEE has recognized that although it is wise to include material on social, professional and ethical issues in conjunction with technical courses, a standalone course is necessary to cover the topics in this knowledge area [1, 17].

This course material is important not only for undergraduate students, but for graduate students as well. Researchers applying for funding from the National Science Foundation are required to provide evidence that they teach ethics of responsible research [4]. Although there are a number of textbooks available to support an ethics course [5, 6, 7, 8], many of them do not adopt a multi-perspective approach, and none of these books cover the breadth of material that we believe is appropriate for the course. The use of computer technology is rapidly changing and expanding. Thus, we assert that it is important to expose the students to as many of the cultural, social, legal, and ethical issues in the discipline of computing as possible in order to broaden their appreciation and understanding of complex issues. Faculty new to the instruction of this type of course may find it difficult to choose which content areas to focus on or to decide upon what types of assignments best aid a student in understanding the course material. Academic papers often provide insights to specific successful techniques used in other ethics courses, but a review of the past decade of papers at ICER, ITCSE, and SIGCSE found that there were very few papers about the teaching of ethical, social and professional issues (about 1 percent of the total) [9]. More curriculum examples are needed in this area.

Ethics is a system of moral principles pertaining to the rightness and wrongness of certain actions [10]. Ethical skills include abilities such as interpreting situations, respecting others and cultivating conscience [11, 12]. As future responsible professionals, students need to employ these skills to help to understand the impact that their software and services may have on end users, as well as the impact that these will have on many others who do not directly interact with their software and services. Because technological change is unpredictable, we need students to be thinking of technology as more than just tools for their functional purposes and have them consider the potential impact of the entire spectrum of technology in our society. Social implications refer to anything that fundamentally alters social structure of groups of people in some way. Students feel that the focus on social issues helps them make the connection between the technologies being studied, and how they affect society [13]. Connolly [9] also recommends that this type of course should emphasize the social context aspects as this leads to more “social responsibility because it exposes our students to a more socially-nuanced understanding of their profession.” The proposed revision of the Computer Science Curricula 2013 also concurs with this direction and places much more emphasis on examining the social context of technology [17].

This paper presents an effective approach to teaching a standalone course on ethics and the social implications of computers with focus on a high diversification of subjects and their social implications. This course is also aimed at promoting ideation and teaching students how to recognize, analyze, and resolve ethical and social dilemmas arising from computing technologies.

The remainder of this paper is structured as follows. Section 2 provides background information on our goals of an ethics course. Section 3 describes the topics that are covered in the course. Section 4 discusses homework projects. Section 5 provides a discussion of the major group project for the course. Section 6 concludes the paper and provides a discussion of the lessons we have learned.

2 Course Background and Goals

As part of their Bachelor’s program in Computer Science, NDSU requires the CSCI 489 course, called Social Implications of Computing. It is a semester-long course that covers historical perspectives about, as well as a contemporary understanding of, the many ethical and social issues that have arisen from the introduction of the computer, internet, and related technologies. This class is also cross listed as a graduate level course that is available in the Master of Computer Science, Master of Software Engineering, and PhD programs in Computer Science and in Software Engineering. The course has a junior-level prerequisite. The reason for this prerequisite is to help ensure that students have the technical knowledge and intellectual maturity to support in-depth ethical analysis. Not only do students need maturity to appreciate the ethical and social issues, they also need to have an understanding of different technologies in order to consider their ethical implications.

The goals of the course are as follows:

- to raise awareness of real-world ethical issues involving computing technologies and help the students develop an understanding of the primary types of social, ethical, and legal issues in computing and information systems
- to help students develop a personal framework for understanding and analyzing how computing and information systems give rise to ethical dilemmas, social, cultural, and legal issues
- to assist students in understanding different methods in responding to many of the ethical dilemmas involving computer technologies
- to develop students’ critical thinking skills
- to develop students’ writing skills
- to develop students’ ability to work within teams
- to develop students ability to present, both orally and in writing, alternative viewpoints and their validity

3 Content Coverage

Our course requires students to purchase a social, professional, and ethical textbook by Sara Baase called “A Gift of Fire: Social, Legal, and Ethical Issues for Computers and the Internet”. Although the text provides a logical structure, it lacks discussion of several important technical areas, and it does not challenge the students to view the impact of computing from multiple perspectives. Because of this, the instructor presents many additional case studies related to the topics covered in the text, as well as additional topics such as intelligent agents, biological computing, net neutrality, media and communications, computer and games addiction, green computing, social media, and the semantic web. These case studies involve real-world situations which make the topics more engaging to the students. A major group project is the primary method used to engage students in viewing, understanding, and respecting ethical and social issues from different perspectives.

The major topics covered in the course are as follows:

- Ethical Frameworks
- Privacy
- Freedom of speech

- Intellectual Property
- Computer Crime
- Computers and Work
- Whistleblowing
- Media and Communications
- Social Media
- Net Neutrality
- Free and Open Source Software
- Green Computing (Sustainability)
- Evaluating and Controlling Technology
- Errors, Risks and Reliability of Systems
- Biometrics
- Professional Ethics
- Assistive Technology
- Artificial Intelligence
- Intelligent Agents
- Semantic Web
- Biological Computing
- Computer and Games Addictions

Many of these topics are recommended by IEEE/ACM standards [1, 17] and are covered in typical ethics, social and professional courses. However, we feel that there are many other topics that should be addressed to enable the students to obtain a more thorough awareness of ethics in computing.

Social Media and Media and Communications are two additional topics that we cover. The rise of social media has fundamentally altered how many people communicate with others and has had a profound impact on society. Social media has created many new areas where the application of ethics is required.

Intelligent Agents typically observe an environment and often learn or use knowledge gained from the observation. This raises the question about what these applications might do to us. Some of the obvious social implications that we discuss relate to the augmentation of intelligence, categorization of society, and target marketing. We also discuss ethical issues such as privacy, power and control, risks and responsibilities, and quality of life.

Search engines have had a profound impact on the way that people seek and gather information, The Semantic Web is the next step that will alter the Web by making it easier for users to find, combine, and share information. Some of the social and ethical issues discussed include communication, collective intelligence, knowledge, education, privacy, power and control, risks and responsibilities and quality of life.

Biological Computing (bioinformatics) applies computer science and technologies to the fields of medicine and biology. The field of bioinformatics has grown rapidly and has the potential to impact everyone, but particularly healthcare providers, patients, social workers, legal and medical professionals, pharmaceutical companies, and lawmakers. There is much concern with computing implications on potential medical applications. There are myriad ethical concerns related to privacy, potential misuse of data, and public and social policies. We address these issues for this topic.

The computer video game industry is one of the fastest growing sectors in the economy [18] and is causing huge societal impacts. In the Computer and Games Addictions topic we discuss debates regarding the misuse and dependency of children, youth and adults on video games as “addiction” and the many positive and negative impacts video games have on society.

We discuss a wide variety of topics that are applicable to the students’ use of technology, and we try to achieve an appropriate balance between ethical analysis and coverage of social issues as they pertain to technology. We believe this helps make the course more interesting, increase participation and increase student engagement.

4 Homework

Homework is important in the development of the student. In a study conducted by Hill et al. [19], homework was positively linked to student achievement. They indicate that homework is an inexpensive way of improving student academic preparation without increasing staff. However, determining whether assignments should be completed individually or in groups is a major decision for computer science faculty.

Individually-assigned homework promotes and strengthens individual capabilities and learning processes [21]. There is often no guarantee that an individual participates in group work and when students submit homework assignments done by others they miss the chance to learn. Individual homework fosters individual accountability.

Group work also has its advantages. Working in groups often aids students in learning complex material. They can also learn quality skills and insights from each other because of the variations in student experiences and backgrounds. Group work can assist in learning effective project team skills, and it fosters socialization and professional networking [20]. We feel that is it important to include a balance of both individual assignments and group assignments. We believe that group work promotes creative thinking and ideation.

Throughout the semester, we require the students to research and report on a number of issues using various types of assignments. All standard homework is individually assigned, and the major project for the course is group work.

The first homework assignment we give is a simple case analysis assignment. It involves a) presenting the student with a case on a real-world ethical issue, b) giving the students a simple framework, illustrated in Figure 1, to use in analyzing the social and ethical issues in the case, c) requiring the student to thoroughly research the case to understand the issues, and d) requiring the students to produce a written report of their analysis based on the framework provided.

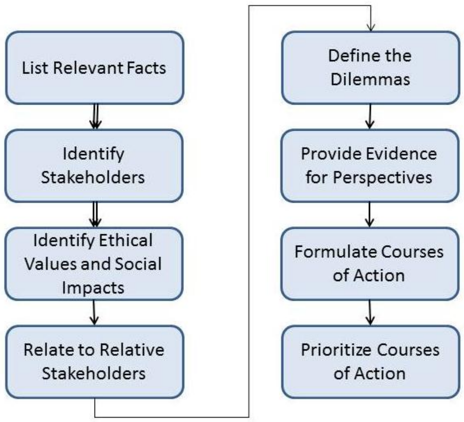


Figure 1 Social and Ethical Analysis Framework

Cases are a good way to develop the student's ability to analyze and synthesize, resolve conflicts, practice decision-making, engage in critical thinking, and develop writing skills [22]. One of the purposes of the assignment is to evaluate each individual student's initial writing and critical thinking skills. Because the framework used to analyze the case is similar to the framework used by the student teams in the major group project, this assignment also exposes the students to what is expected in the future major project and provides the students with a means of practicing and expressing their analyses.

The second assignment involves writing an editorial on an ethical or social issue involving a computing technology specified by the instructor. This is similar to the classic "letter to the editor" assignment in that it requires the student to define and defend a position. The students are required to research the topic in order to understand the issue, develop a position, and then build a solid argument to support their claim. Students are coached in the classical rhetorical devices of ethos, logos, and pathos in the process of developing their position paper. The process of choosing a position develops a student's decision making skills; the process of building and defending their position is an excellent exercise for building critical thinking skills; and, writing the editorial develops writing skills.

The third assignment involves a movie analysis. There are many movies that illustrate, either directly or indirectly, the impact that computing technologies can have on society. Two of our recent favorites are "The Social Network" by Columbia Pictures and "Barbershop Punk" by Evil Twin Productions. Films are challenging to fully understand, analyze, and critique because they contain large amounts of information. Important elements include the narrative, selection of music, use of computer-based special effects, juxtapositions of imagery, angles of shooting, speed of the narrative, allusions to other films, choice of settings and costumes, acting, and directing. Analyzing a movie provides a change of pace in this course. Most students find the movies entertaining and engaging, which seems to re-energize them for the task of performing another analysis activity. We not only ask the students to analyze the ethical dilemmas and social impacts of the main technology in the film, but we also ask them pointed questions about specific technologies, such as social networks or net neutrality, and we require them to justify their responses. Through this process, students deepen their understanding and appreciation of social norms and more fully address the principles that groups use to define their values, beliefs, attitudes and behaviors.

A fourth assignment explores the ethical and social issues that arise in the realm of biological computing. As in any topic with medical connections, this assignment tends to personalize the views of students toward social and ethical choices and forces them to directly consider the consequences of alternative available actions.

The last individual assignment is another case analysis, this time based on a different computing technology than the first assignment. In form and function, this assignment is identical to the first assignment; however, there are slightly different goals. Because the students are more practiced in using the ethical and social analysis framework presented at the beginning of the semester, the grading is more rigorous. We also evaluate each student's current level of writing and critical thinking skills to aid in our course learning outcomes assessment.

5 Major Group Project

An important assignment in the course is the major group project. This project incorporates active learning in that it requires students to solve ethical-reasoning problems using skills that they were exposed to in the course. Active learning shifts responsibility from the instructor to the student. Active learning and other participatory techniques are also very effective for teaching about computers and society. Active learning emphasizes exploration of attitudes and values over passively listening to instructor lectures [14].

This project is composed of a series of deliverables in which student teams produce a multi-vocal response to an ethical dilemma. In this project, student teams take a more in-depth look at a topic of their choosing. Student engagement and interest in the project is enhanced by letting the teams choose their project topic.

Each team is charged with identifying a major ethical dilemma or question that is related to social, legal, and ethical issues of computing technologies. Table 1 shows some of the recent topics carried out by student teams

.

Table 1. Recent Project Topics

Computers and the environment	Computers in law enforcement
Ownership of ideas	Open source software
Impacts of social media	Social issues for smartphones
Distance learning	Violence in computer games
Cyberspace communities	Spam
Identity theft	Privacy on the web
Hacktivism	Internet censorship
Children on the internet	Technology and terrorism
Telemedicine	Identification and biometrics
Predator drones	Digital Distraction
Copyright issues in music	Addiction to video games
Trust in the semantic web	Information warfare

Then the teams must analyze these ethical questions, evaluate their importance, research and understand the viewpoints of multiple stakeholders, and carry out analyses that consider the ramifications of possible courses of action. It is called a “multi-vocal” assignment because each team member examines the question from the perspective of a specific stakeholder. The teams are explicitly required to not come to a consensus on an answer to the ethical question (such that the particular points of view are right or wrong), but instead deeply explore the diverse opinions on the topic. More specifically, the students are to show, or recreate and analyze, some of the debates that might occur among stakeholders. The process of bringing out the multi-vocal aspects and considering the multiple viewpoints helps the students become more mature in their ethical thinking. We encourage each member of the team to adopt the role of a particular stakeholder. This type of “role playing” helps the students to consider other viewpoints and generate more creative analysis and solutions to the topic dilemmas [23]. We find evidence that students can transition from initial thinking that is shallow to much more thoughtful and mature considerations. This kind of activity also improves ideation, creativity, and critical thinking.

This type of project demands an intense interaction among students. Team members need to share ideas and reflect on the ideas of others. Teamwork incorporates values such as cooperation and collaboration., Goals of working collaboratively as a team include developing understanding of perspectives other than your own, respecting those perspectives, and only then trying to offer counter-arguments. This teamwork activity aids in developing critical thinking skills, in that some of the characteristics of strong critical thinkers are being inquisitive, open-minded, and flexible, as well as being willing to consider alternatives.

5.1 Project Deliverables

There are five deliverables in the group project, which are phased in, and much feedback is provided along the way through assessment comments, rubrics, and consultation with the instructor.

The project starts with topic selection. Students choose their own topic; however, the topic must be approved by the instructor. In addition to the topics shown in Table 1, student teams can select a topic of their own choosing that is approved. The list of topics is purposely open-ended to allow students to think creatively about the breadth of technology and associated societal impacts.

The five deliverables of the project include a vision statement, project proposal, interim report, final report, and project presentation.

The purpose of the vision statement is to help the students focus on developing a solid and focused ethical question that is complete, clear and concise. We also ask them to reflect on why this question is important to the members of the team and to write a justification of why others should find this question important. Limiting the focus of the assignment to these three points helps the students to create a clear vision for their project and to explain their motivation for choosing their topic. The vision statement also helps the instructor screen the student’s projects early on to make sure their topics are narrowed appropriately and to ensure that the the projects have a reasonable scope and depth.

The project proposal concerns the presentation of the project plan. Reflected in the project proposal is the background research that student teams have performed, the forming of their topic, their argument that the topic is valid, and all of the ethical questions they are evaluating. They also clearly identify who has a stake in their questions, and which ethical values are the most pertinent to their ethical questions. The teams are required to plan their research by identifying their research methods, possible information sources, methods for recording and sharing team discussions, and research findings. They must also explain why these are appropriate. Next, they articulate a preliminary plan for how they would write their final paper and design their presentation. They also list the qualification of the team members and convince the reader that they have the skills and resources to perform the project. Lastly, they provide a breakdown of the tasks that each member will perform, and then provide a schedule that shows the planned coordination of the project.

The interim report is a pivotal piece of the project. By the time of the interim report, much of the project should be completed. The interim report includes updated introduction and methodologies sections from the project proposal. The teams also provide a much more thorough and complete description of the stakeholders and the ethical and social impacts of the identified question(s) as they apply to these stakeholders from their individual perspective. The interim report also presents the evidence that they have at this point that supports the validity, or lack of validity, in ethical terms, of the various perspectives and action choices that the stakeholders have available to them. They must also make an effort to evaluate the chosen action ramifications by tracing through and describing what might happen as a result of taking each specific action. Lastly, they provide a much more detailed plan on the approach they will use for the in-class presentation, as well as a more detailed format for the final report.

The final report is similar in style to the interim report; however, this report should have a sense of thoroughness and completeness and make points that may have still been under development at the time of the interim report. We also ask the students to analyze the feasibility of their previously-proposed action choices and provide an overall conclusion that summarizes and defends the major findings of the project.

The in-class project presentations are one of the last project deliverables. Oral presentations have benefits for both the student presenters and the audience. The student teams demonstrate their mastery of the material through an oral presentation and hone their presentation skills. The audience benefits from learning about unfamiliar topics that have not been covered in the classroom, which enhances the course curriculum.

5.2 Forming Teams

There are basically three ways of forming student teams: instructor selection, self-selection, or joint selection. Research on teams or small groups in the classroom supports either instructor selection or joint selection [24]. We choose the instructor selection method and assigned teams randomly. There are typically three people on a team. No team training was provided, but the instructor and the GTA monitored team progress carefully and provided mentoring whenever teamwork issues arose.

5.3 Communication Contract

One of the biggest team problems we have observed in the past concerns communication. This difficulty includes students finding times in their schedules to meet regularly in person. They use different email accounts, Facebook, text messaging, etc. in attempts to communicate; however, these attempts often fail because each member uses these tools differently. Problems surface when one team member attempts to communicate via these tools and other team members do not respond or do not respond in a timely manner. This results in the team member attempting to initiate the communication feeling disrespected and disempowered. In a previous study [16], we used a social networking site called Ning in an attempt to remedy some of the team communication problems. Although somewhat successful, the primary problem using this tool was the learning curve for using the tool. Students often preferred to use tools they were already familiar with, such as email and Facebook.

To address this problem, we impose a requirement for each team to form a communication contract that includes reaching agreement on the following elements:

- Mediums – what are the communication channels to which team members will commit? These include such media as email, phone, twitter, Facebook, Skype, text messages, and instant messaging. Each team member documents which channels are available and preferred by them. For example, if the team commits to communicating by email, they must specify which email address should be used.
- Schedules – when are team members available to meet in person or be available via the agreed-upon media? This includes day-of-week or time-of-day constraints imposed by work class schedules, family commitments, etc.
- Structure – what are the standards for which communication methods are to be used for each situation? For example, the team may agree to schedule in-person meetings using Facebook, send information about the reports using email, and only call each other if there is an emergency. Also, by documenting acceptable "response times," team members will be patient in waiting for a response. For example, the group may agree to respond to all emails within 24 hours, so that when a member sends an email, they know they can expect everyone to respond within a day.

The communication contract eases misunderstandings and avoids problems and complaints that often occur in teamwork.

5.4 Peer Assessment and Evaluations

Grading the essays that are produced by the teams for their projects is a very time consuming task. The literature has many examples of using rubrics to ease this burden [15] We establish rubrics for each deliverable in advance and share them with the students so that they understand exactly how they will be evaluated, and apprehensions are thus eased. Due to the projects having a long history of being required in the course, the guidelines for teams to follow and the grading rubrics for each deliverable are very detailed and well-developed.

Not all students on the teams will have equal abilities, but we actually view this as a net positive. In industry managers learn to delegate tasks based on who can do what, and students should learn these skills. We want true team functioning on the project, and we give all team members the same grade. The better teams are able to work with this balance, lift the weak links, and make good use of strong team members.

The last grading element is peer reviews of teammates. During the presentations, students are assigned the task of reviewing all presentations by other teams and scoring the presentations using a rubric provided by the instructor. The students rate the team presentations and provide comments in strength and weakness categories. Peer review is another form of active learning and engages the audience in the presentations. An additional benefit of the peer reviews is that these reviews discourage students from skipping class after they have presented their own work.

6 Discussion of Lessons Learned

Student feedback about the course has been positive. Students comment that they enjoy the wide variety of topics that are covered in the course and enjoy discussing societal effects. The group project allows them to further explore a computing topic that they are interested in and to share their work with the rest of the class. We typically find that there is a lot of enthusiasm during the presentation from both the presenting teams and the audience. Students comment that they like learning about the many different topics that are presented by the student teams that are not necessarily covered in class.

Student often complain that they have difficulties finding time to meet or that some team members are not contributing equally to the project. Some teams are also slow to truly grasp the requirements of the project. We handle these situations by scheduling a team meeting with the instructor for discussion and negotiation. Additional meetings are not usually required. We are currently developing a framework for validating learning outcomes that we plan to report on in the future.

7 References

[1] IEEE-CS/ACM Joint Task force on computer science curriculum 2008: an interim revision of CS 2001, 2008, pp. 92-94.

[2] C.L. Spradling, L.K. Soh, and C. Ansorge, "Ethics training and decision-Making: do computer science programs need help," in *Proceedings of the 39th SIGCSE technical symposium on Computer science education*,. 2008, pp. 153-157.

[3] A. Ghafarian, (2002). "Integrating ethical issues into the undergraduate computer science curriculum," The Journal of Computing Sciences in Colleges 18(2), 2002, pp. 180-188.

[4] M. Zaidman, "Students in Computer Ethics-Speak up," The Journal of Computing Sciences in Colleges 26(2), 2010, pp. 251-255.

[5] S. Baase, *A Gift of Fire: Social, Legal, and Ethical Issues for Computing and the Internet*. Upper Saddle River, NJ: Prentice Hall, 2008.

[6] J.M. Kizza, *Ethical and Social Issues in the Information Age*. London: Springer-Verlag, 2007.

[7] R. Spinello, *Cyberethics: Morality and Law in Cyberspace*. Sudbury, MA: Jones and Bartlett Learning, 2011.

[8] H.T. Tavani, *Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing*. Hoboken, NJ: John Wiley and Sons, Inc., 2011.

[9] R. Connolly, "Beyond Good and Evil Impacts: Rethinking the Social Issues Components in Our Computing Curricula". In *Proc.of the 16th annual joint conference on Innovation and technology in computer science education ITiCSE*, 2011, pp. 228-232.

[10] Dictionary.com. Internet: http://dictionary.reference.com/browse/ethics.

[11] C. Huff, L. Barnard, and W. Frey. "Good computing: A pedagogically focused model of virtue in the practice of computing (part 1)". *Journal of Information, Communication and Ethics in Society*, 6(3), 2008, pp. 246-278.

[12] C. Huff, L. Barnard, and W. Frey. "Good computing: A pedagogically focused model of virtue in the practice of computing (part 2)". *Journal of Information, Communication and Ethics in Society*, 6(4), 2008, pp. 284-316.

[13] F. Zaidi, "A Student's Perspective on the Importance of Teaching Social Issues in the I.T. Curriculum". In *Proceedings of the 2011 conference on information technology education SIGITE*. 2011, pp. 83-85.

[14] C. Bonwell, and J. Eison. "Active learning: Creating excitement in the classroom". ASJE-ERIC Higher Education Report No, 1, George Washington University, Washington D.C., 1991.

[15] B. Moskal, K. Miller, and L.A. King. "Grading essays in computer ethics: rubrics considered helpful". In *Proceedings of the 33rd SIGCSE Technical symposium on Computer Science Education*, 2002, pp.101-105.

[16] K. Nygard, L. Bender, G. Walia, J. Kong, and K. Gagneja. "Collaboration using Social Networks for Team Projects". In The International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'11), 2011, pp. 86-92.

[17] IEEE-CS/ACM Joint Task Force on Computing Curricula: Computer Science Curricula 2013 Strawman Draft, 2012, pp.163-172.

[18] http://www.theesa.com/games-improving-what-matters/economy.asp

[19] S. Hill, S. Spencer, R. Alston, and J. Fitzgerald. (1986, Fall). "Homework in our schools," Education, 107 (1), 1986, pp. 58-70.

[20] P. Heller. "The Unfortunate Motivation Behind Group Work" Internet: http://paulrheller.com/2010/08/the-unfortunate-motivation-behind-assigning-group-work, retrieved March 26, 2012.

[21] R. Or-Bach and A.Yadin. "The importance of emphasizing individual learning in the collaborative learning era". *Journal of Information Systems Education*. 21(2), 2010, pp.185-194.

[22] M. Serra, D. Baneres, E. Santamaria, and J.M. Basart, "Acquiring Ethical Communicative Tools for an Online Ethics Training". In *Proceedings of the 29th ACM international conference on Design of communication*, 2011, pp. 285-288.

[23] J. Flowers. "Ideation: It's More Than Just Brainstorming," http://jcflowers1.iweb.bsu.edu/rlo/brainstorming2.htm, 2006, retrieved on March. 12, 2012.

[24] S.B. Feichtner, and E.A. Davis. "Why some groups fail: A survey of students' experiences with learning groups". *Journal of Management Education* , 9(4), 1984, pp. 77-88.

Individual Student Capstone Projects Focused on Community Engagement / Service Learning

Donald R. Schwartz, Andrew Habers, Oliver Galicki and James Carter
Department of Computer Science
Millsaps College
Jackson, MS USA

Abstract - *In several previous papers, we presented various service-learning/community-engaged group projects. [1][2][3][4][5] While these have been highly successful, our department has also decided to explore a new area – in addition to group projects, we experimented with* ***individual*** *service-learning/community-engaged projects. Computer Science students at Millsaps have several unique opportunities to design, develop and implement large-scale, year-long projects. The most popular of these come in the forms of Honors Projects and Senior Seminars. Three individual projects completed this school year are described in this paper.*

Keywords: *Curriculum Development, Preparing graduates for academia, Preparing graduates for industry, Evaluation strategies, Transition to graduate studies*

1 Introduction

Our College invites qualifying Juniors (those whose overall GPA is at least 3.3) from all majors to participate in our Honors program.[6] The Honors program gives our top students the opportunity to engage in intense one-on-one research with a faculty mentor of their choice, in a research area of their choice. These year-long projects start in the Spring of the Junior year and conclude during the Fall semester of the Senior year. Honors projects are similar to Masters projects, but are aimed at our undergraduates.

During the Fall semester, eligible Juniors, working closely with their chosen faculty mentor, define their thesis topic and submit a proposal describing the topic and focus of the intended project. Proposals from across campus are evaluated by the Honors program faculty committee. Approved projects begin in the Spring semester, during which students enroll in Honors 1. Just after Spring break, each student submits a progress report to the thesis committee, which consists of the project mentor, a second reader familiar with the subject matter, and a member of the Honors program faculty committee. If the committee determines that sufficient progress has been made to reasonably complete the project, the student is allowed to sign up for Honors 2 in the Fall. Those students then use the remainder of the Spring semester, along with the summer, to complete their projects. By the end of the first week of the Fall semester, students must submit a draft of their honors thesis to their thesis committee. The beginning of the Fall semester of Honors 2 is focused on the completion of the honors thesis, which is due by Fall Break. Students whose theses are deemed to be acceptable will then schedule a 45-minute defense of their research at some point before Thanksgiving. Final versions of the thesis are due in February, when Millsaps sponsors an Honors Conference, which is the capstone event. During the conference, students present their work to the wider Millsaps community (students, faculty and administration), family and friends. Students who successfully defend their theses graduate "with Honors" and are individually recognized during Commencement exercises.

As can be expected, Honors projects mentored by faculty in the Computer Science department also include the completion of an implementation. While some projects focus on a specific area of research, our department has made it a practice to allow students to design, develop and implement large-scale projects for actual "customers", either on-campus departments or offices, or local, city or state-wide not-for-profit entities.

In addition to the possibility of completing an Honors project, all Millsaps students must complete a Senior Seminar. In Computer Science, we offer a year-long Senior Seminar [7], part of which includes the completion of a senior project, which is a year-long research project, performed with faculty supervision, on an appropriate topic selected by the student. Ideally, senior projects combine ideas and material from several upper-level courses and extend far beyond what would be expected of projects in any particular course. These individual projects begin during the Fall semester of the senior year and culminate in a Senior Project Presentation during Comprehensive Exam Week near the end of the Spring semester. Naturally, students give multiple progress report presentations during Senior Seminar, which meets weekly throughout the school year. Once again, our department has recognized that service-learning (community-engaged) projects for on- and off-campus entities can sometimes be very appropriate for Senior Projects.

Two Honors Projects (the Millsaps Volunteer Tracker System and the EduChart Project) and one Senior Project (for the Mississippi Department of Health) that were completed this school year are described in this paper.

2 The Millsaps Volunteer Tracker System

The purpose of the Millsaps Volunteer Tracker system [8] is to provide an online interface for Millsaps student volunteers to enter, view, and analyze hours of service. It was developed in order to streamline the current process of recording student volunteer hours. The logging of volunteer hours is currently used for College grant applications, Sorority and Fraternity nationals, and student volunteer awards, and this tool meets each of these needs.

To accomplish these goals, the software platform used was a web-accessible database developed in the MySQL environment, along with both static and dynamic WebPages. The Web Pages were developed using a combination of PHP, HTML, CSS, and JavaScript. In order to streamline the development process, WampServer [9] version 2.0, containing Apache version 2.2.11, PHP version 5.3.0, and MySQL version 5.1.36, was used as a host to the database and PHP scripts. Volunteer Tracker was designed using the Google Chrome web browser, although compatibility with other browsers was also implemented, including Internet Explorer version 8.0 or later, Mozilla Firefox version 6.0 or later, and Google Chrome version 13.0 or later.

The database consists of following tables:

Users – information required for each user including student ID, user-name, encrypted password, access level and other personal information.

Organization – information pertaining to all organizations, including the organization ID, name, website, and other contact information.

Community_partner – information related to each community partner, including name, description, and other contact information.

Project – information regarding each project in the system. Each project is associated with one organization. The project name and subsequent information such as start date, end date, description, etc. is also stored.

News – stores news posts, which are created by system administrators and viewable by all users upon logging in to the system.

Sessions – tracks each user's session in the system and applies a time-out in which users are automatically logged out after a set amount of time.

Member_of – associates users to organizations, since a user can have multiple organization associations and each organization can have multiple users.

Sponsors – associates community partners to projects, since a project can have multiple community partner sponsors, and community partners can sponsor multiple projects.

Worked_on – information associated with each user's hours volunteered on projects. It consists of four fields: studentID, projectID, dateWorked, and numHours.

User_group – a simple table developed for easily adding or editing access-level group names.

Organization_type – similar to the user_group table, simply stores the unique ID and names of the twelve different organization types.

Sign_up – developed as a future expansion, but not included in the scope of this project. If implemented in the future, the sign_up table would allow for users to sign up for future projects if the user limit has not been reached for that particular project.

Volunteer Tracker was designed with data validation in mind, with two levels of security to ensure that the proper data is being inserted into the database. The first level of data validation resides on the client side in each form submission. Every field in every form of Volunteer Tracker goes through a data validation function, first making sure that no illegal characters (which are sometimes used for nefarious reasons) exist, and second making sure that the data input is what the field is actually requesting. The second level of data validation resides on the PHP server and will strip illegal characters from every input field on the form, preventing malicious attacks such as SQL injections. Both levels of data validation ensure that the system database is protected from erroneous data and common malicious attacks.

There are three levels of access within the Volunteer tracker system: User, Service Manager, and System Administrator. Each level of access was designed to coincide with the current hierarchy used by Millsaps College to track volunteer work. The most basic level of access, the User level, was designed for student access, allowing them to perform the two most crucial actions that Volunteer Tracker allows: entering hours and viewing service transcripts. All users also have the ability to view news updates at the home page and edit personal user information. Service Managers are students and volunteers who have been promoted to manage a particular group of volunteers. In addition to all permissions allowed for Users, Service Managers are able to enter hours for themselves, enter hours for other students, and add new projects into the system. The highest level of access, the System Administrator level, has full control over all features of the system, including the ability to add, edit, and remove Users, Organizations, Community Partners, News, and Projects.

In order to gain access to the Volunteer Tracker system, the user must enter a valid username and password into the login page. A link to register new users is also provided. All passwords are immediately encrypted using the NSA's cryptographic hash function SHA-1. [10]

If a login attempt succeeds, a PHP session variable is created which holds the user's information and security level. Sessions are managed by updating the Sessions table with timestamps from any system activity. Sessions are time-limited to increase the security of Volunteer Tracker, limiting the possibility of an old session being hijacked on an idle computer. Users are also able to manually terminate their sessions by logging out or closing the browser.

After successfully logging in to Volunteer Tracker, the user is presented the main menu as shown in Figure 1. Service Managers and Users main menu is pared down to their respective access options.

Figure 1 System Administrator Main Menu

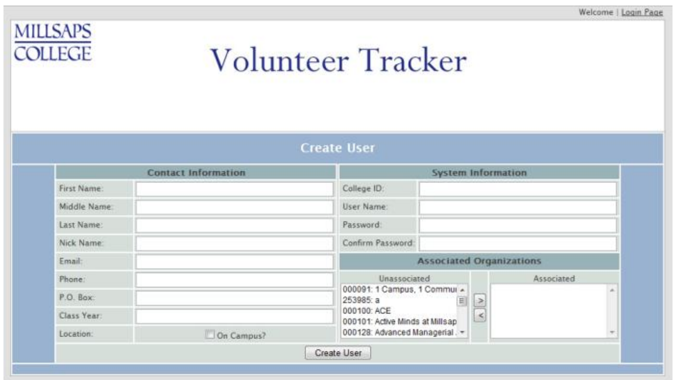


Figure 2 Create new user form

The form for a new user to create an account is shown in Figure 2. A similar form is presented for editing user data. We've also included forms for adding/editing/deleting Organizations (Figure 3), Community Partners, and News Posts.

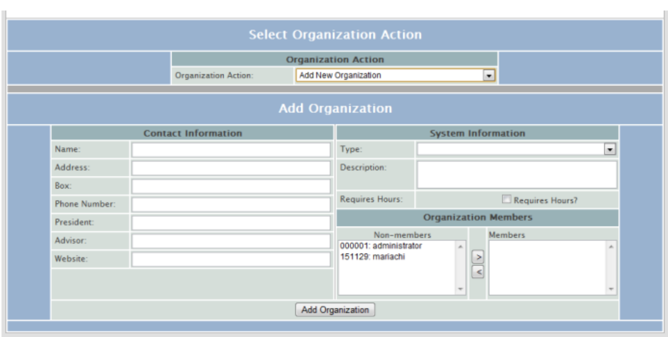


Figure 3 Organizations

Reporting is an important tool implemented in Volunteer Tracker. Service Managers and Administrators are able to run an array of reports, detailing information such as the user with the most hours, the organization with the most hours, specific user's hours, hours for an organization by totaling hours of associated projects, hours for an organization by totaling hours of members, total hours for the entire college, and total money raised by college. All reports are easily printable by clicking the print report button, which opens a printer friendly window along with a printer dialog box.

3 The EduChart Project

The purpose of the EduChart project [11] is to develop a web-based computer application to facilitate the tracking of clinical internship scheduling and patient charting for the Emergency Medical Technology Program at Holmes Community College (Holmes CC) in Ridgeland, Mississippi. To successfully receive a Certificate of Paramedicine from Holmes CC, students must complete 1000 hours of field and clinical internships across the state, along with 70 credits of didactic classroom learning. For both students and instructors, the software application helps to eliminate the time-consuming clerical duties and handwritten paperwork that are currently associated with the internship portion of the program. Students benefit from being able to view their scheduled shifts, submit patient care reports, and quantitatively view their skill progression statistics online. Instructors are able to focus more of their time on teaching, as the application will greatly reduce their current data-entry and paper-handling load.

EduChart includes two levels of access. Users in the system are classified as either "instructors" or "students." All users designated as instructors are full EduChart administrators. Per client request, EduChart is not structured to differentiate between instructor and program director, as both perform the same set of administrative tasks. The student level designation is, as the name implies, used for students enrolled in the paramedic program. Students are not granted any administrative rights in the program.

The EduChart web application is built on top of a server-hosted database. The MySQL Community Server 5.1 database platform was selected and used for the development of EduChart.

The database consists of the following tables:

Class - the details about a year-long offering of the Holmes CC Paramedicine Program. A record is defined by a classID, className, classStartYear, and classStatus.

ePCR - the details about an individual patient encounter and represents the **e**lectronic version of a physical **P**atient **C**are **R**eport. A record is mainly defined by a reportID, creationDateTime, shiftID, and studentID, and secondarily defined by various medical-related fields that pertain to the emergency call.

Instructor - the details about individual instructors within the program at Holmes. A record is defined by a Holmes_ID, instructorFirstName, instructorLastName, instructorEmail, instructorPhone, and username.

Location - the details about individual clinical sites at which shifts can be assigned and completed. A record is defined by a locID, locName, locBase, locType, and locStatus.

Shift - the details about a specific shift that an instructor has created in EduChart. A record is defined by a shiftID, locID, shiftStartDate, shiftStartTime, shiftLength, and preceptorsAssigned fields. (A preceptor is an EMT supervisor who evaluates the students.)

Shift_participation - links a student with a specific shift, which allows specific students to be evaluated on their performance during a specific shift by a preceptor.

Student - demographic information about the student.

User_permissions - the usernames, passwords, and permission levels for all users.

Instructors who successfully log into EduChart are presented with the menu shown in Figure 4.

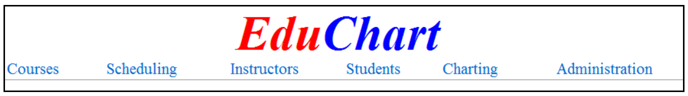


Figure 4 The Instructor Menu

The Courses tab of the Instructor Menu gives the user access to manage all aspects of courses. Through the tab, users can add a class, modify a class, and inactivate a class.

The Scheduling tab of the Instructor Menu gives the user access to manage all aspects of shifts and shift scheduling. Through the tab, users can create a shift (Figure 5), modify a shift, cancel a shift, assign a shift to a student, unassign a shift from a student, and view scheduled shifts.

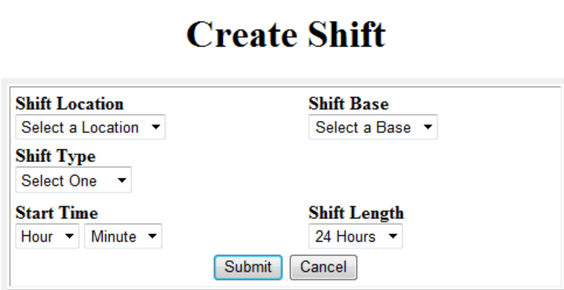


Figure 5 Creating a Shift

The View Schedules tab allows the user to view all scheduled shifts by three different search techniques: by Location, Student or Date. Allowing the search of scheduled shifts through all three methods makes it extremely easy to locate exactly what information is needed, whether it is for distribution to a certain location's preceptors or a certain student within a certain date range, etc.

The Instructors tab of the Instructor Menu gives the user access to manage all aspects of instructors. Through the tab, users can add instructors, modify instructor information, and remove instructors from the system.

The Students tab of the Instructor Menu gives the user access to manage all aspects of students. (Figure 6) Through the tab, users can add students, modify student information, and inactivate individual students from the system.

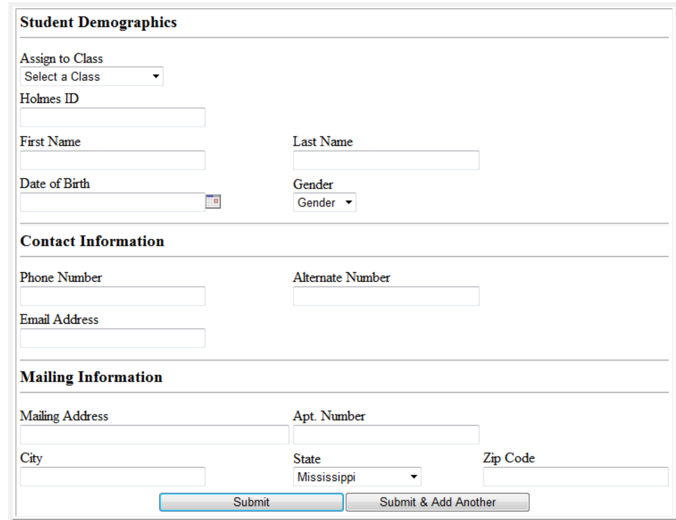


Figure 6 Student Demographics

The Charting tab of the Instructor Menu gives the user access to manage all aspects of students charting and their performance analysis. Through the tab, users view completed reports and monitor student skill progress. (Figure 7 and Figure 8)

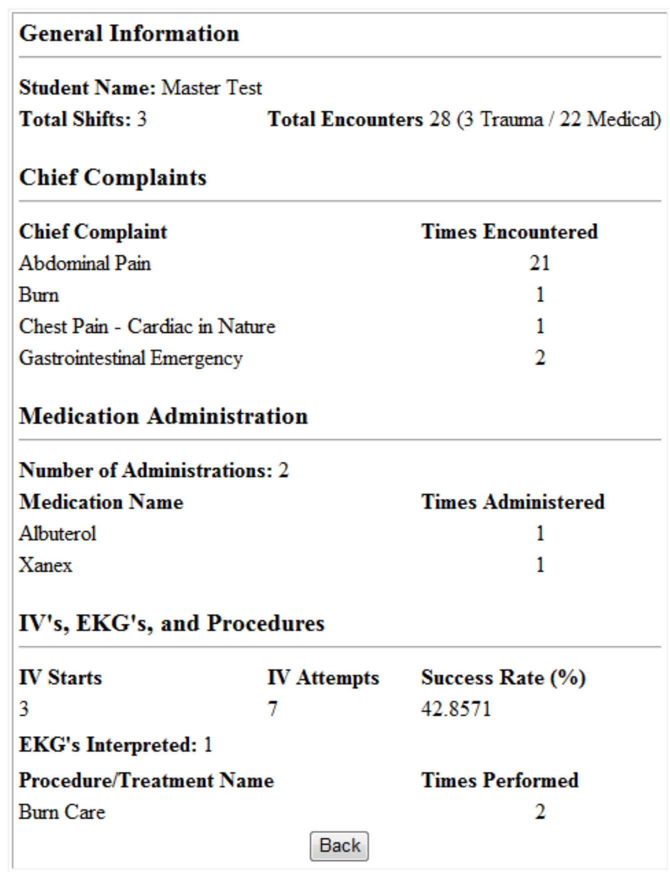


Figure 7 General Student Information

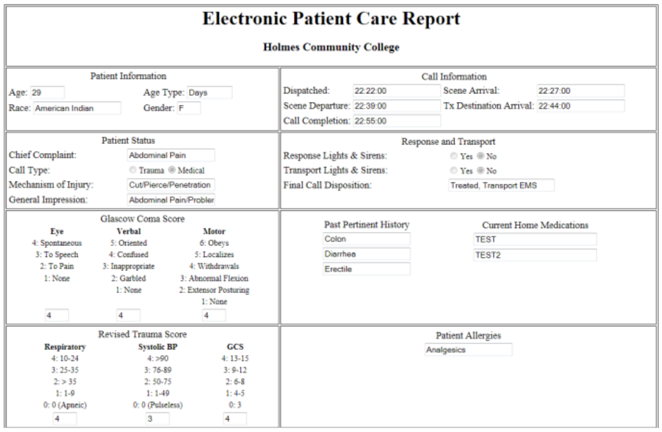


Figure 8 Electronic Patient Care Report (ePCR)

The Administration tab of the Instructor Menu gives the user access to manage certain administrative options. Through the tab, users can lookup user passwords, add shift locations, and modify shift locations.

Limited functions, including the ability to update their contact information, request specific shifts, and view their progress, are available for Student users.

4 Department of Health Project

In the Spring of 2010, Millsaps College was asked by the Mississippi Department of Health to produce an application which would provide surveys to parents of children with disabilities. These surveys assist in providing quantitative data as to the effectiveness of the department's ability to achieve desired outcomes. The project also provided a means for the department to create a statewide directory of Healthcare Service Providers, who specialize in working with children with disabilities, as well as their families. The project was successfully completed. [4] In the Summer of 2010, the Department developed new regulations which required all applications to be implemented in a .NET Framework-friendly language. This reimplementation was completed as a Senior Project. [12]

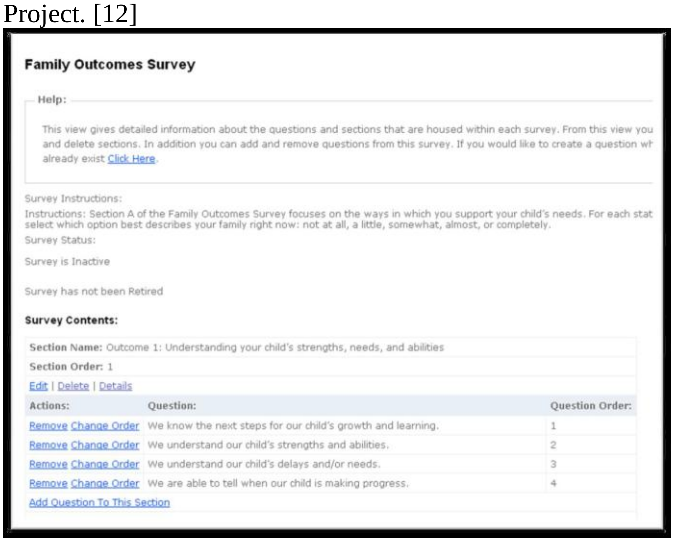


Figure 9 Family Outcomes Survey Creation

Staff members of the Department of Health travel all over the state to collect quantitative data describing the lifestyle of families who care for children with disabilities. These questions provide feedback as to the effectiveness and need for the Department of Health to provide services for these families. Our system allows the Department to create a variety of surveys (Figure 9). Delivery of each survey is provided in a way which provides anonymity (Figure 10). Additionally, the survey is only offered to those parents who care for children with disabilities.

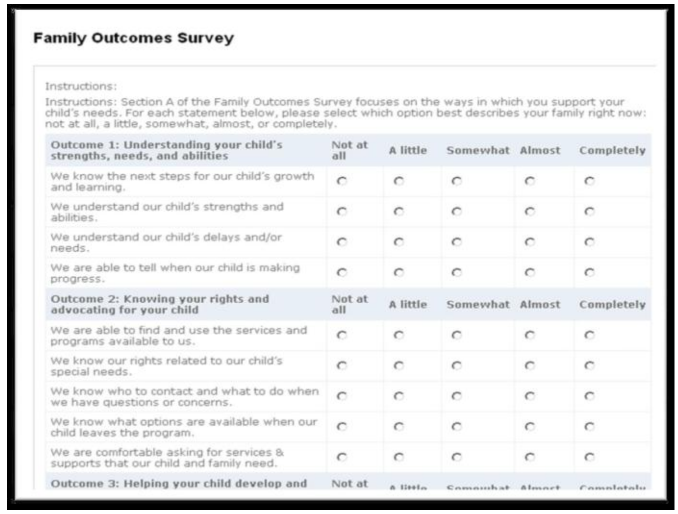


Figure 10 Family Outcomes Survey

After the survey data is collected, it is necessary to perform analyses on the data using a number of metrics, as provided by the survey. The population who takes the survey is measured using methods of categorization, including race, ethnicity, and location. Mississippi is divided into nine health districts. Each of the eighty-two counties in Mississippi is a member of one of these Health Districts. Additionally, there are recognized municipalities within these counties. This information is used to gather valuable information about how the Department of Health can better address the needs of Children with Disabilities and the families.

The Mississippi Department of Health also collects information about healthcare service providers from all over the state. Information is gathered in regard to the type of services that are offered, location, hours of operation, who they aim to serve, and the obligation on the part of the client if services are rendered (Figure 11). The information gathered by the department of health is used by parents of children with disabilities, to find the right professional to provide for the needs of their child. Through the use of this directory, children with disabilities get the healthcare they need to live better lives.

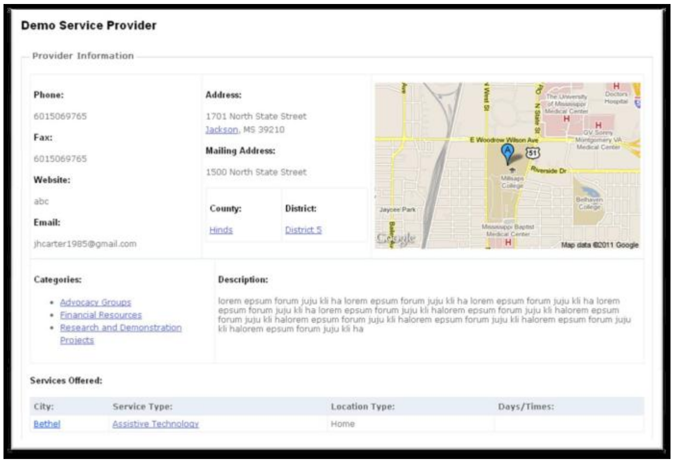


Figure 11 Data Service Provider

The implemented project satisfies the following list of requirements, as described by the Department of Health:

- The application needed to be available over the web and usable by clients of and staff for the Mississippi Department of Health.
- The application had to be compatible with the .NET Framework.
- The survey had to be simple to use for clients.
- The application needed to be able to organize data based on identifiable information about the clients.
- The application needed to allow staff to easily view the results of the survey in a format that could help inform decision making.
- The application needed to provide a method of collecting information about Healthcare Providers in Mississippi.
- The application needed to provide a means for navigating entries within the directory of Healthcare Service Providers.
- The application needed to separate identifiable information about users from the survey data received.
- The application needed to be aware of the geographical divisions of the Mississippi Department of Health.
- The application needed to recognize the authentication of multiple users and apply different authorization levels to such users.

The application was developed using Visual Studio Professional, was written in C#, and used ASP.net for its server side logic. SQL Server was used as the database technology used to house information. Queries were handled using Language Integrated Queries, which provided fourth generation language functionality within C#. The building of the Model Objects was assisted by the ADO.net framework. This application was built to run on the IIS 7 Web Server. Additionally all components were compliant with .NET Framework 4.

5 Current Status and Conclusions

There are many issues that must be addressed when allowing students to propose individual service-learning/community-engaged capstone projects. Many of these same issues also arise for group projects.[1][2] Obviously, the overall scope of the project must be appropriate. Many on- and off-campus clients propose projects that are either too ambitious or too limited in scope to be suitable for individual projects. Ideally, a good working relationship between the faculty mentor and the "client" must be established. Clients must understand (at the beginning of the project!) what is expected of them and what is expected of the student. Critical to the success of the project is the ability of the student (and the client) to discover and understand the system requirements and limitations. This requires that the client be readily available to answer questions and provide data throughout the development life cycle. Clients (and students) must clearly understand that requests for changes to the project may not be possible after an agreement has been made on the project specifications. The faculty mentor is the final "decider" in terms of what functionality the project will (and will not) support.

We are happy to report that each project was very successful. Each of the Honors projects was presented to a committee of Honors Program faculty. Each project earned a perfect score in all five areas of the overall evaluation. The senior project was presented to the Computer Science Faculty as well as representatives from the Mississippi Department of Health and was unanimously approved.

The Volunteer Tracker was installed on a new server and made available to the Millsaps community during the Spring 2012 semester. Training sessions for new users have been completed and the system is currently "in use". So far, things seem to be running very smoothly during initial usage.

The Mississippi Department of Health received our project during the Fall 2011 semester. They report that they have been swamped readjusting programs to new regulations, so have not yet installed our new system. The current target date is August, 2012.

Unfortunately, due to unforeseen changes and modifications (during the Fall semester) to the EMT Program at Holmes CC, the EduChart system was not installed this Spring. However, should things change and Holmes CC decides to use it, our system is ready-to-go.

References

[1] Schwartz, Donald R., "Service Learning and Software Engineering: Managing 'Real World' Group Projects", *Proceedings of the 2007 International Conference on Software Engineering and Data Engineering (SEDE-07),* Las Vegas, July 2007.

[2] Schwartz, Donald R., "The Benefits (and Drawbacks!) of "Real World" Software Engineering Semester Projects: A Case Study", *Proceedings of the 2009 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS-09)*, Las Vegas, July 2009.

[3] Schwartz, Donald R., "CLOWNS – A Software Engineering Semester Project Currently In-Use in Kindergarten Classes", *Proceedings of the 2008 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS-08)*, Las Vegas, July 2008.

[4] Schwartz, Donald R., "Incorporating Service Learning into Capstone/Software Engineering Group Projects", *Proceedings of the 2010 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS-10)*, Las Vegas, July 2010.

[5] Schwartz, Donald R., Jonathan F. Spencer, and Adam T. Huffman, "Service Learning, Software Engineering, and Hurricane Katrina: A Case Study", *Proceedings of the 2006 International Conference on Software Engineering Research and Practice (SERP '06)*, Las Vegas, June 2006.

[6] http://www.millsaps.edu/academics/honors_program.php

[7]http://www.millsaps.edu/academics/computer_science_research.php

[8] Habers, Andrew. "Student Volunteer Tracker", an Honors Thesis for the Department of Computer Science, Millsaps College, 2012.

[9] http://www.wampserver.com/en/

[10] http://www.w3.org/PICS/DSig/SHA1_1_0.html

[11] Galicki, Oliver. "EduChart - Development of an Emergency Medical Technology Education Management System", an Honors Thesis for the Department of Computer Science, Millsaps College, 2012.

[12] Carter, James. "First Steps Early Intervention Program", a Senior Project for the Department of Computer Science, Millsaps College, 2011.

A Semantic Web Framework for Teaching Logic Circuits

Francisco-Edgar Castillo-Barrera
Engineering Faculty
Universidad Autónoma de San Luis Potosí
Dr. Manuel Nava 8, Zona Universitaria poniente
78290 San Luis Potosí, México
ecastillo@uaslp.mx

Carolina Medina-Ramírez
Department of Electrical Engineering
Universidad Autónoma Metropolitana, México
Av San Rafael Atlixco 186, Col.Vicentina
09340 Distrito Federal, México
cmed@xanum.uam.mx

Jose Emilio Labra Gayo
Department of Computer Science
Universidad de Oviedo
C/Valdes Salas s/n 33007-Oviedo, España
jelabra@gmail.com

S. Masoud Sadjadi
School of Computing and Information Sciences
Florida International University (FIU)
11200 SW 8th St, Miami, USA
sadjadi@cs.fiu.edu

Abstract

Ontology-based systems have been used to facilitate teaching and learning. Ontologies have proven to be a very useful artifacts to represent a domain and as an important component in specific applications for giving semantics. In logic circuits domain, ontologies have been employed for teaching logic gates (xor, or, not, nand), and this approach has been deemed as an effective way for capturing and using the knowledge of the logic gates on assembling circuit systems. This knowledge can be reused by new students gaining time and reducing circuits manufacturing costs. In addittion, the correct assembling among logic gates and the right output of a circuit can be validated by using semantic techniques. In this paper, we describe a semantic web technique based on a core ontology, a reasoner and SPARQL queries for teaching and learning circuits based on logic gates. We use an example and a prototype to explain our approach.

1. Introduction

New scenarios for self-learning of logic circuits are neccesary during the design phase. These scenearios have to ensure the funcionality expected by the students and simulating the behavior of their circuits. These scenarios help them to prevent economic losses. Another important factor to consider is the circuit models reusability [19]. The time for developing a complex circuit using a semantic circuit repository decrease the cost of the project and reduce the learning curve of new students in the project. In this context, semantic technologies seem relevant. We can use Ontologies[32] in order to represent a circuit based on logic gates (and, or, not, etc.) and to verify the circuit design. Each connection of the circuit can be validated by means of ontology properties [32] and reasoners [27]. The new knowledge obtained for each part of the circuit assembled, can be stored in an ontology [21] written in OWL-DL [39] by means of metadata (is stored as an XML file), in this way the knowledge is capitalized [30][23]. This knowledge can be used by new developers or new members of the project to reduce manufacture time. In consequence, the company decreases costs. The circuits behavoir can be modelled by SPARQL queries. In fact, a complex circuit could be represented by one SPARQL query.

The rest of the paper is structured as follows. In Section 2 we give the related work for teaching logic circuits based on ontologies and SPARQL queries. In Section 3 we briefly explain concepts about Semantic Web Techniques, Ontologies, Core Ontologies, Reasoners and SPARQL queries. Section 4 we describe our approach for teaching of logic circuits in a Semantic Web Framework. In Section 5 we show the feasibility of our technique by describing an example and a prototype called Itzamna. Finally, in Section 6 we conclude our work.

2. Related work

The ontologies based on logic gates for teaching is mostly represented by work of Robal et al.[29] who wrote an ontology-based intelligent learning object for teaching the basics of digital logic. Robal's ontology is oriented for teaching the basics of digital logic, our ontology is made for validating the right connections among logic gates, for verifying the right output of the logic circuit built, and Students can create new circuits reusing the ontology. Another work by Sosnovsky and Gavrilova [33] was made by teaching and learning C programming based on the designed ontology.

3. Semantic Web Techniques

The Semantic Web [35][3][24] is the Tim Berners-Lee vision for representing information in the World Wide Web. He defines it as a web of data that can be processed directly and indirectly by machines [35]. This is a collection of standards, a set of tools [7], and a community that shares data. Semantic Technology is a concept in computer science which goal is to give semantics to data[10]. Supported by semantic tools [31] that provides semantic information about the meaning of words (RDF, SPARQL, OWL, and SKOS). *Semantic Web Techniques are methods and techniques based on semantic tools which allow us to manipulate information too*.

3.1 Ontologies

Ontologies are the key for Semantic Web goals. An Ontology [11][13][35][31][32] is defined by Gruber as *a specification of a conceptualization* [11] which defines the terms used to describe and represent a domain of knowledge, also is the model (set of concepts) for the meaning of those terms, thus defines the vocabulary and the meaning of that vocabulary, are used by students and applications that need to share domain information. More specifically, an ontology is a formal representation of knowledge with semantic content which allows us to obtain information. Such information can be retrieved by performing SPARQL queries [28] or using a rule-based inference engine [34]. In our case, the logic circuits is the domain of knowledge.

3.1.1 Core Ontologies

In philosophy, a **Core Ontology** [6] is a basic and minimal ontology consisting only of the minimal concepts required to understand the other concepts. It must be based on a core glossary that humans can understand. A **Core Ontology** is a complete and extensible ontology that expresses the basic concepts in a certain domain of knowledge. In this work we have built a core ontology which consists of a logic gates glossary which students of circuits understand well. We consider that these kind of ontologies support the reuse. Building these kind of ontologies do not require a complex methodology [9] to follow it, in fact, following the Ontology Development 101: A Guide to Creating Your First Ontology [8] or An eXtreme method for developing lightweight ontologies [16] are enough.

3.2 SPARQL Query Language

SPARQL is a query language for the Resource Description Framework (RDF) [22] which is a W3C Recommendation [38]. RDF Schema (RDFS) is extending RDF vocabulary for describing taxonomies of classes and properties. We use Web Ontology Language OWL [39] which extends RDF and RDFS. Its primary aim is to bring the expressive and reasoning power of description logic to the semantic web. In our learning scenario, Querying language is necessary to retrieve information [17] and verifying the correct output of the circuits. At this moment, we only have decided to explore semantic queries in SPARQL instead of applying another action such as: production rules [34].

3.3 Reasoners

A reasoner [27] is a program which its main task is checking the ontology consistency. It verifies if the ontology contains contradictory facts, axioms or wrong properties among concepts. Besides, new knowledge can be inferred after applying it. The most popular reasoners are Cerebra [25], FACT++ [37], KAON2 [26], Pellet [27], Racer [14], Ontobroker [5], OWLIM [20]. Pellet is an open-source Java based OWL-DL reasoner. In our verification process we use Pellet for checking the consistency of the logic circuit ontology and classify the taxonomy. We select the Pellet reasoner, because it gives an explanation when an inconsistency was detected.

4 Teaching Scenario in a Semantic Web Framework

Itzamna[1] is a factory framework of circuit models based on semantic techniques which focuses on maximising the level of reuse using logic circuits. One of the most important features of this framework is enabling knowledge reuse in logic circuits modelling using Semantic web techniques [4].

The aim of this framework is to allow to learn about logic circuits using a friendly interface and a graphical desing. Our main contributions are: we define a framework that allows us to reuse logic circuit for building new circuits. Second, our approach supports the validation of the output values obtained from the logic circuit during the design phase and Finally, our framework supports the learning and teaching of circuits based on logic gates. A prototype of the

[1] Itzamna is the name of an upper god of wisdom in Yucatec Maya mythology

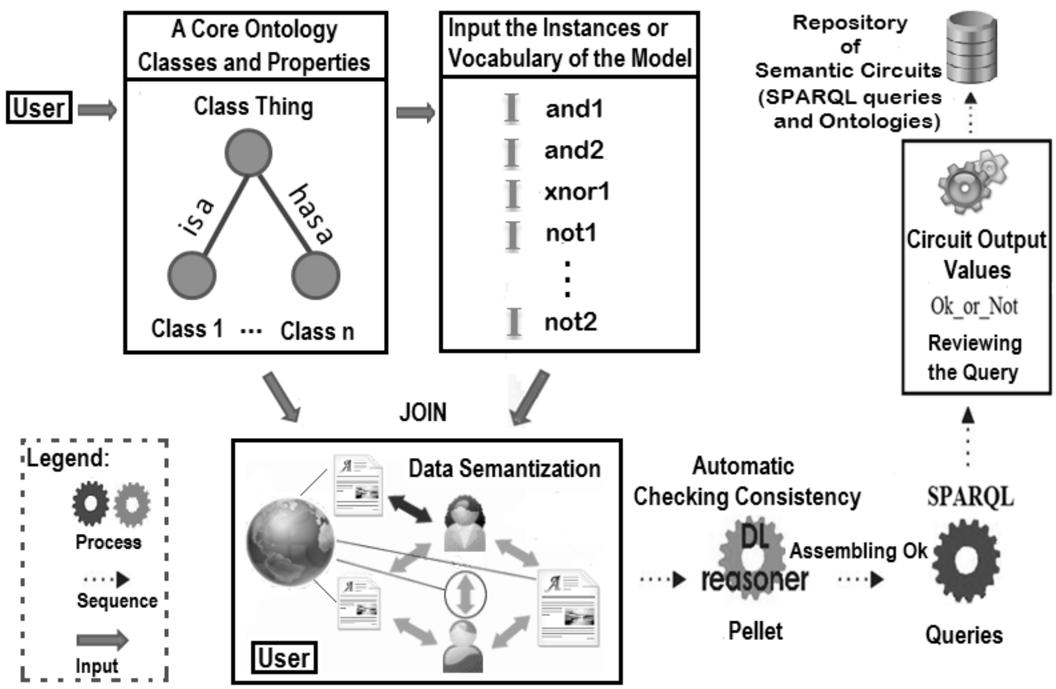


Figure 1. Semantic Web Techniques for Teaching Logic Circuits

framework involves a visual editor. The tool makes use of the library Flamingo and the Ribbon component [18] implemented in Java. We have used Jena API [11] and Java language [36] for programming that and NetBeans IDE 7.0 [2]. Each logic gate is represented in a graphic way and it can be assembled with another, see Figure 2. This is the first task to do by the students during his learnig process. Second, they have to introduce the information about the name and bit values of the circuits in the ontology. We have called this **Data Semantization process**, see Figure 1. This information also can be introduced by means of a text file (the *option Create Instances Vocabulary*). Itzamna transforms the user vocabulary (logic gates that the user needs for building his circuit) from a text file into an ontology instances. The third step will be to check the ontology consistency part of the Semantic Verification, see Figure 3. Semantic verification is the process which uses a core Ontology and Semantic Technologies (SPARQL queries) to guarantee the correct construction of logic circuits with specific connections and outputs. The semantics of assembling the logic gates are described with object properties. An important aspect of the logic gates to consider during the assembling is the Input and Output connections. A logic gate has one output, but different number of input connections. The logic gate connections are based on the output of one of them using as input in the others.

4.1 A Core Ontology for Logic Circuits

We propose a core ontology called **OntoCoreCircuit** which has the minimum concepts (logic gates) neccesary to represent the 1-bit Comparator circuit. And, Or, Xor, Not, Nand, Nor and Xnor are universal gates and they do not require to be validate by experts. Besides, we only need 3 or 5 competency questions to validate the ontology [12]. A Logic Gates Ontology was created for capturing and verifying information about the new logic circuit model during the graphical design. OntoCoreCircuit Ontology is built by means of classes using **n3** notation and relations among concepts. This is used by the ontology, because is a valid RDFS and OWL-DL notation. The Ontology use RDFS and OWL-DL language [39][15][40]. They are fundamentally based on descriptive logic languages. This Ontology consists of 3 Classes (*Circuit*,*Gate* and *Bits*), 35 Instances (*:and*, *:or*, *:not*, etc.), 10 Object properties (*:isTypeGate*, *:andOutput*, *:hasInput1*, etc.) and 1 Datatype property (*:hasName*).

5 Building a 1-bit Comparator in Itzamna Framework

A 1-bit comparator is a hardware electronic device that receives two bits (A and B) as input and determines whether one bit is **greater than**, **less than** and **equal** to the other bit.

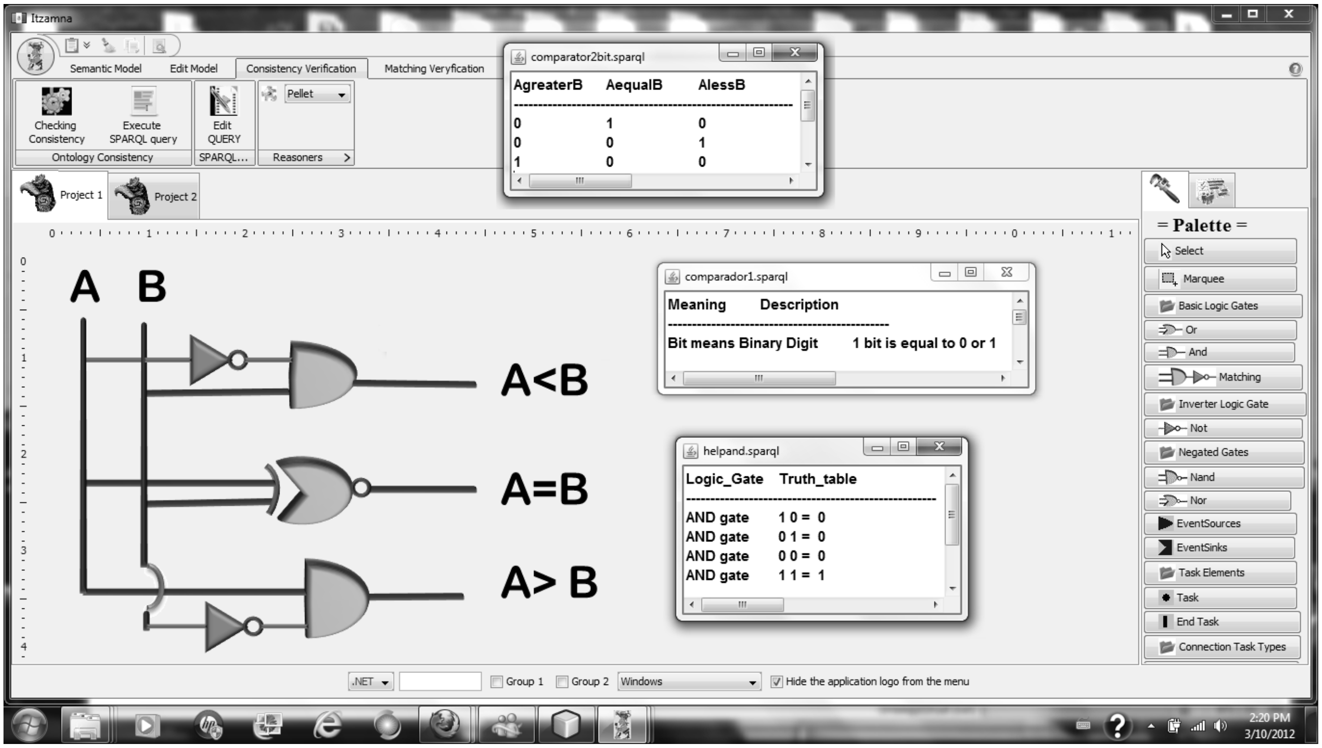


Figure 2. 1-bit Comparator circuit diagram and SPARQL queries

Table 1. 1-Bit Comparator Truth Table based on Instances

A	B	AB	A<B	A = B	A>B
:0	:0	:0_0	:0	:1	:0
:0	:1	:0_1	:1	:0	:0
:1	:0	:1_0	:0	:0	:1
:1	:1	:1_1	:0	:1	:0

This circuit has 2 bit binary inputs (A, B) and three single bit binary outputs (A $>$ B, A=B, A $<$ B). This kind of circuit can be extended for **2,3,...,n-bits** comparator circuit. For that reason, we have chosen this circuit for learning. The truth table using the instance notation is showed in Table 1. This circuit is built with 5 logic gates (2 not, 2 and, 1 xnor), as showed in Figure 2. The logic circuit model used for describe a 1-bit Comparator circuit was made in Itzamna Framework using its graphical interface of logic gates (the text in the image was adding for clarifying the circuit information), and is shown in figure 2. The input model (logic gates) is created by the user who selects classes and relation among concepts and he creates the logic gates instances (*:and1*, *:and2*, *:xnor1*, *:not1* and *:not2*). In this case the input model only has 5 logic gates and we can create its instances and relations among them using the Itzamna's menus (create instances vocabulary).

5.1 Assembling Verfication using The Pellet Reasoner

The Core Ontology written in OWL-DL, allow us to define restrictions which Pellet can verify during the consistency checking process. This action can be performed by menues in Itzamna framework, see Figure 3. For instance, the following code establishes that the *and* gate has only 1 output, because a FunctionalProperty is defined for *:andOutput* Object Property.

```
:xnorOutput a owl:ObjectProperty ;
    rdfs:domain   :Gate ;
    rdfs:range    :Bits ;
    rdf:type  owl:FunctionalProperty .
```

An interesting property of the ontology used in this work is a blank node. It is a node in an RDF graph representing a resource without URI or literal. We used it as variable. If we put the same blank node, the result for this node has to be the same. In our example below, _:c1, _:c2 and _:c3 are blank nodes (working as variables). The example shows how to *:xor1* and *:and2* gates are forced to have the same input (_:c2).

```
   # :xnor1 is a member of xnor gates
:xnor1 :isTypeGate  _:c1 .
   # :xnor1 requires 2 input values
_:c1   :hasInput2   _:c2 .
   # :not1 is a member of not gates
:not1  :isTypeGate  _:c3 .
   # :not1 requires only 1 input value
_:c3   :hasInput1   _:c2 .
```

A difference with Logic Programming Paradigm, we can check our types using ontologies. In particular when we create a new logic gate, for example *:and2*, we do not have to introduce all input and output values. In this case, it is only neccesary to establish the property relation *:and2* **:isTypeGate** *:and* . Besides, the ontology allow us to see circuits and gates saving in the ontology at the same time because the *Gate* class is a subclass of *Circuit*.

```
:Circuit         a owl:Class .
:Gate rdfs:subClassOf :Circuit .

:isTypeGate a owl:ObjectProperty ;
             rdfs:domain   :Gate ;
             rdfs:range    :Gate .
```

The *disjointWith* property allow to verify restrictions in the input model. For example a logic gate is not a bit, these two classes are different. Defining disjoint classes is also possible [1].

```
 :Gate rdfs:subClassOf   :Thing ;
        owl:disjointWith :Bits .
```

All instances created, properties (object and datatype) established among instances, and blank nodes in the Ontology are checked by the reasoner Pellet during the consistency verification process.

5.2 Output Validation using a SPARQL Query

The last step after the reasoner have checked the ontology circuit consistency is to apply a SPARQL query for validating the correct output of 1-bit comparator circuit. In our case, we have defined a query which describes the circuit and obtain the output for given input values. We can think that SPARQL is the version of SQL for ontologies. Besides, we can use variables in the queries, constraints, filtering information, logic operators, if statements and more. Each triples (each line after) are linking by variables which begin with a question mark. In this code *?type1* and *?AB* are examples of variables. The same name of variable imply the same value to look for in the query. We can execute and edit queries in Itzamna framework because the Jena API allowed us to use SPARQL queries in our framework programmed in Java language. The following example shows the SPARQL query used in this work for validating the output values in our 1-bit Comparator circuit.

```
PREFIX   : <http://www.ejemplo.org/#>
PREFIX fn:
<http://www.w3.org/2005/xpath-functions#>

SELECT DISTINCT
        ?AgreaterB ?AequalB ?AlessB
WHERE
{
    :xnor1  :isTypeGate   ?type1 .
    ?type1 :hasName       ?xnorName .
    ?type1 :hasInput2     ?AB .
    ?AB      :xnorOutput   ?AequalB .

    BIND( if(?AB = :0_0, :0_1,
          if(?AB = :0_1, :0_0,
          if(?AB = :1_0, :1_1 ,:1_0)))
          AS ?ABneg )
    BIND( if(?AB = :0_0, :1_0,
          if(?AB = :0_1, :1_1,
          if(?AB = :1_0, :0_0 ,:0_1)))
          AS ?AnegB )

    :and1    :isTypeGate   ?type2 .
    ?type2  :hasName       ?and1Name .
    ?ABneg :andOutput   ?AgreaterB .

     :and2    :isTypeGate   ?type3 .
    ?type3  :hasName       ?and2Name .
    ?AnegB :andOutput   ?AlessB .
}
```

If the user wants to give an specific input values, only needs to change the variables *?AB* and *?Cin* for instances of the *Bits* class. For example: *:0_1 :xorOutput ?XorOutput* .

An optional step, when the logic circuit has been verified and validated, consists on storing the project independent of the ontology or include it in the core ontology. It is important to note that these challenges increase the reuse of this ontology and decrease the time in the development of future circuits. Benefiting the economy of companies (Knowledge Capitalization [30][23]). In our example, the code included in the core ontology was showed in Figure 3.

6. Conclusions

Teaching Logic Circuits by means of Semantic Web Techniques is possible with core ontologies, reasoners, and SPARQL queries. Ontologies are usually expressed in a logic-based language (Description-Logic), enabling detailed, sound, meaningful distinctions to be made among

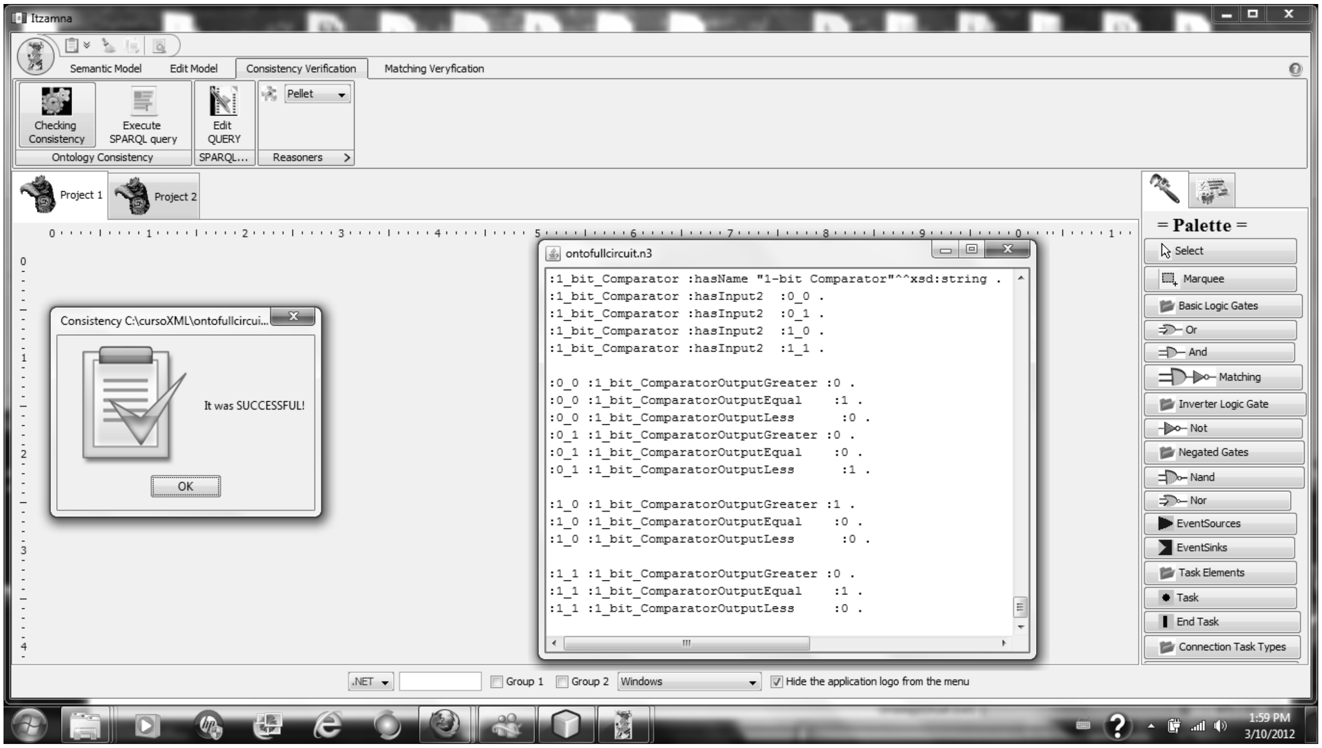


Figure 3. Consistency checking of the new circuit

the classes, properties and relations. Core Ontologies give more expressive meaning, maintains computability, do not require the validation of experts or apply a complex methodology for its construction. This core ontology for logic circuits increase the reuse of it and decrease the time in the development of future circuits. The use of an core ontology of logic circuits allowed us to validate the output of the 1-bit Comparator and verify the correct assembling of its gates using the Pellet reasoner and a SPARQL query with semantics in comparison with a classic SQL query. The queries on the ontology are simple and easy to do for all students whereas a classic SQL query in a database requires computational knowledge. In this paper we have presented a Semantic Web framework called Itzamna and described Semantic Web Techniques used for learning and teaching logic circuits.

7 Acknowledgments

This material is partly based upon work supported by the National Science Foundation under Grant No. OISE-0730065.

References

[1] F. E. Antoniou Grigoris and V. H. Frank. Introduction to semantic web ontology languages. 2005.

[2] E. Armstrong, J. Ball, S. Bodoff, D. B. Carson, I. Evans, K. Ganfield, D. Green, K. Haase, E. Jendrock, J. Jullion-ceccarelli, and G. Wielenga. The j2ee TM(tm) 1.4 tutorial for netbeans TM(tm) ide 4.1 for sun java system application server platform edition 8.1.

[3] K. Breitman, M. A. Casanova, and W. Truszkowski. *Semantic Web: Concepts, Technologies and Applications (NASA Monographs in Systems and Software Engineering)*. Springer-Verlag London, 2006.

[4] W. P. Davies John, Stunder Rudi. Semantic web technologies trens and research in ontology-based systems. 2006.

[5] S. Decker, M. Erdmann, D. Fensel, and R. Studer. *Ontobroker: Ontology based access to distributed and semi-structured information*. Citeseer, 1998.

[6] M. Doerr, J. Hunter, and C. Lagoze. Towards a core ontology for information integration. *Journal of Digital information*, 4(1), 2011.

[7] W. M. K. B. Duineveld A.J., Stoter R. and B. V.R. Wondertools? a comparative study of ontological engineering tools. 2000.

[8] N. F.Noy and D. L.McGuinness. Ontology development 101:a guide to creating your first ontology. March 2006.

[9] A. Gómez-Pérez, M. Fernández-López, and O. Corcho. Ontological engineering with examples from the areas of

knowledge management,e-commerce and the semantic web. 2003.

[10] T. Gruber. Ontolingua: A mechanism to support portable ontologies. pages KSL 91–66. 1992.

[11] T. Gruber. Toward principles for the design of ontologies used for knowledge sharing. pages 907–928. 1995.

[12] M. Gruninger and M. S. Fox. The role of competency questions in enterprise engineering. In *PROCEEDINGS OF THE IFIP WG5.7 WORKSHOP ON BENCHMARKING - THEORY AND PRACTICE*, 1994.

[13] N. Guarino. *Formal ontology in information systems: proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, volume 46. Ios Pr Inc, 1998.

[14] V. Haarslev and R. Müller. Racer system description. *Automated Reasoning*, pages 701–705, 2001.

[15] I. Horrocks, P. Patel-Schneider, and F. Van Harmelen. From shiq and rdf to owl: The making of a web ontology language. *Web semantics: science, services and agents on the World Wide Web*, 1(1):7–26, 2003.

[16] M. Hristozova and L. Sterling. An extreme method for developing lightweight ontologies. In *In Workshop on Ontologies in Agent Systems, 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2002.

[17] M. Hwang, H. Kong, and P. Kim. The design of the ontology retrieval system on the web. volume 3, pages 1815 –1818, feb. 2006.

[18] Java.net. Flamingo. http://java.net/projects/flamingo/, 2010.

[19] I. Jurisica, J. Mylopoulos, and E. Yu. Ontologies for knowledge management: an information systems perspective. volume 6, pages 380–401. Springer, 2004.

[20] A. Kiryakov, D. Ognyanov, and D. Manov. Owlim–a pragmatic semantic repository for owl. In *Web Information Systems Engineering–WISE 2005 Workshops*, pages 182–192. Springer, 2005.

[21] S. L. and M. B. Ontology evolution within ontology editors. volume 62, pages 53–62. September 2002.

[22] O. Lassila and R. Swick. Resource description framework (rdf) model and syntax. *World Wide Web Consortium, http://www. w3. org/TR/WD-rdf-syntax*, 1999.

[23] F.-M. Lesaffre and V. Pelletier. A business case of the use of ontologies for knowledge capitalization and exploitation.

[24] K. T. S. Michael C. Daconta, Leo J. Obrst. *The Semantic Web: A guide to the future of XML, Web Services and Knowledge Management*. Wiley Computer Publishing, Inc., 111 River Street Hoboken, NJ, jun. 2003.

[25] R. Mishra and S. Kumar. Semantic web reasoners and languages. *Artificial Intelligence Review*, pages 1–30, 2011.

[26] B. Motik and R. Studer. Kaon2–a scalable reasoning tool for the semantic web. In *Proceedings of the 2nd European Semantic Web Conference (ESWC05), Heraklion, Greece*, 2005.

[27] B. Parsia and E. Sirin. Pellet: An owl dl reasoner. In *Third International Semantic Web Conference-Poster*, 2004.

[28] J. Pérez, M. Arenas, and C. Gutierrez. Semantics and complexity of sparql. *The Semantic Web-ISWC 2006*, pages 30–43, 2006.

[29] T. Robal, T. Kann, and A. Kalja. An ontology-based intelligent learning object for teaching the basics of digital logic. In *Microelectronic Systems Education (MSE), 2011 IEEE International Conference on*, pages 106–107. IEEE, 2011.

[30] B. D. Rodriguez-Rocha, F. E. Castillo-Barrera, and H. Lopez-Padilla. Knowledge capitalization in the automotive industry using an ontology based on the iso/ts 16949 standard. volume 0, pages 100–106. IEEE Computer Society, Los Alamitos, CA, USA, sep. 2009.

[31] S. S., G.-P. A., D. W., R. M.-L., and N. N.F. Why evaluate ontology technologies? because it works! volume 19, pages 74–81. Jul-Aug 2004.

[32] S. S., S. R., S. H., and S. Y. Knowledge processes and ontologies. volume 16, pages 26–34. Jan-Feb 2001.

[33] S. Sosnovsky and T. Gavrilova. Development of educational ontology for c-programming. 2006.

[34] SWRL. Swrl:a semantic web rule language.

[35] B.-L. T., H. J., L. O., and Others. The semantic web. *Scientific American*, 284(5):34–43, 2001.

[36] N. L. J. Tolksdorf Robert, Bontas Paslaru Elena. A coordination model for the semantic web. pages 419–423. 2006.

[37] D. Tsarkov and I. Horrocks. Fact++ description logic reasoner: System description. *Automated Reasoning*, pages 292–297, 2006.

[38] W3C. http://www.w3.org/consortium/. 1994.

[39] W3C. Owl web ontology language, 1994.

[40] W3C. Semantic web activity: Semantic web activity state. 2002.

Experiential Learning: Applying Software Engineering Principles and Mobile Programming in Undergraduate Capstone Projects

C.E. Otero[1], D. Luce-Edwards[2], J. Smith[2], S. Clark[2], M. Whitt[2], B. Hines[2],and A. Qureshi[2]
[1]Department of Information Technology, University of South Florida, Lakeland, FL, USA
[2]Department of Math and Computer Science, University of Virginia, College at Wise, Wise, VA, USA

Abstract - *Capstone projects provide students the opportunity for applying concepts and knowledge acquired throughout students' academic curriculum. Although much can be gained through typical academic courses, experiential learning is necessary to fill gaps and provide students with understanding of why it is important to master concepts presented during typical course-work. To this end, an experiential software engineering capstone project was conducted during the academic year 2010-2011 at the University of Virginia, College at Wise. The capstone project provided students with significant exposure to real-life problems encountered when creating a high-quality software product that meets customer demands. This paper presents the work produced by students and can serve as documented experience of the types of work that can be accomplished by undergraduate students in a year-long capstone project. It describes the processes the students used to complete the software lifecycle (requirements, design, construction, testing, and deployment) of the MCQA system with emphasis on delivered artifacts and the lessons learned and experiences gained.*

Keywords: Software Engineering, Software Requirements, Software Design, Unified Modeling Language (UML), Graphical User Interface (GUI), Human Computer Interaction (HCI), Software Test, Capstone Projects

1 Introduction

Typical undergraduate Software Engineering courses (e.g., software requirements, design, etc.) provide a focused approach centered on topics particular to one phase of the software engineering life-cycle. Students in those classes are expected to gain knowledge about the concepts of the particular life-cycle phase, with little context, since appreciation of most software engineering topics is enhanced through the analysis, design, construction, testing, and delivery of a working product that meets some customer's demand for quality. In this sense, individual software engineering courses provide limited learning opportunities, since it can be hard for students to make meaning without experience. To this end, year-long capstone projects provide students the avenue for employing experiential learning to make meaning of fundamental concepts through direct experience; allowing students to combine an entire software engineering curriculum into a single project, for customers with real demands. Experiential learning provide students the opportunity to enhance their education by applying acquired skills within context and enhance their academic experience by acquiring new skills on-demand—without formal course-work—to meet the project needs.

This paper presents the 2010-2011 software engineering capstone project conducted at the University of Virginia, College at Wise. The project was conducted using experiential learning which relied on academic-industry partnership, where a (FL-based) major defense contractor served as customer for the year-long project. Through collaborative online tools, weekly meetings between customer and capstone team were conducted to set a framework for accountability and direct involvement of customers throughout the software development life-cycle. The paper focuses on the work produced by students and their documented experience throughout the capstone project. It describes the processes they used to complete the software lifecycle (requirements, design, construction, testing, and deployment) of a real system with emphasis on delivered artifacts and the lessons learned and experiences gained. The work can serve as documented experience of the type of project appropriate for undergraduate seniors working on a year-long capstone project.

2 Project Description and Requirements

By definition, a capstone "is a course or project near the end of a program which integrates knowledge, concept and skills associated with an entire sequence of study"[1]; therefore, students were required to follow the complete software engineering life-cycle to develop the Mobile Container Query Application (MCQA). The MCQA project followed an agile model of software engineering that incorporated the following phases: requirements, design, construction, and testing.

The project began with a brief statement of work (SOW) that included an overview of the expectations of the project and prioritized requirements. Upon receiving the SOW, the team began the requirements phase of the project by following the guide to the Software Engineering Body of

Knowledge (SWEBOK) for requirements engineering [2]; specifically, this phase included the following activities: elicitation, analysis, specification, and validation. As part of requirements elicitation, the team derived a list of requirements based on the SOW and soon after held a kickoff meeting with a representative of the contracting company. During that meeting, project requirements were gathered from the customer and the list of project requirements was updated to reflect the changes from the meeting. Upon completion of the meeting, the team began the analysis phase by having brainstorming sessions to examine the requirements they had already gathered for redundancy and conflicts. In addition, several use cases were created to capture knowledge acquired, elicit new requirements, and handle issues undefined by the customer, such as non-functional (or quality requirements). For the specification activity, the team developed the Software Requirements Specification (SRS) by following IEEE standard 830-1998 [3]. The SRS included the following sections: the project's purpose, scope, functional requirements, nonfunctional requirements, and an indexed list of all requirements. To validate the requirements, the team sent the SRS to the customer and after feedback, revised it based on that feedback and the customer accepted the SRS. After the requirements phase, the project was sufficiently defined to begin the design phase. A sample set of functional requirements are provided as follows:

- The software shall search for items using item number.
- The software shall display the item's current and previous location using geographical maps.
- The software shall receive and process unsolicited notification / messages from the server.
- The software shall provide capability for updating items.

From the requirements, a sample use case diagram is presented in Figure 1.

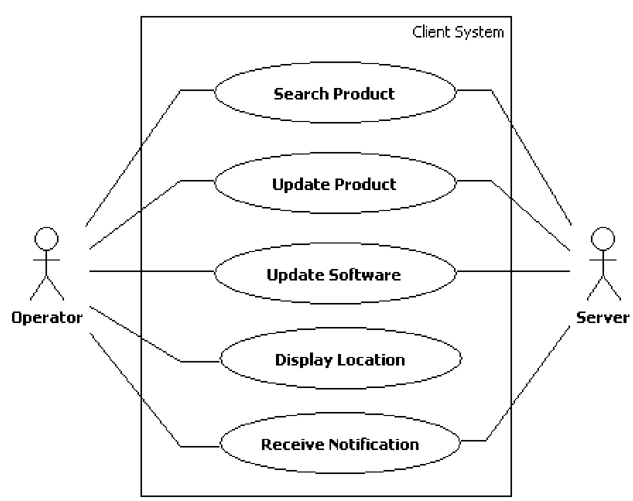


Figure 1. Sample UML Use Case diagram for the mobile client software.

For each use case identified for the system, one or more scenarios were created to document the expected system's behavior and identify non-functional requirements. Sample steps from the *Search Product* use case scenario are presented in Table 1.

Step	Operator Action	System Action
1	Operator enters valid product ID and clicks the search button.	Validates the data. Retrieves server's communication information from configuration file.
2		Establishes a connection with the server system and sends product request data to the server.
3		Waits a maximum of 3 seconds for a server response.
4		Response received and product information is displayed.
5		Save response data in file system and ask user to search for another product.
6	Operator clicks the cancel button to finish searching for products.	

3 Software Design Approach

The software design phase involved translating the contents of the SRS into a comprehensive design for the system before beginning the construction phase. The team performed three main design activities: the software architecture, the detailed design, and the HCI design. Each design activity is described in detail below. Upon completion of each design activity, design efforts, analysis, and evaluation were documented in the Software Design Document (SDD) for the project following IEEE standard 1016-2009 [4]. The SDD includes each design complete with the rationale behind that design as well as a requirements matrix that maps the requirements to their representation in the design.

3.1 Software Architecture

The software architecture was carefully designed to meet the product's functional and non-functional requirements. Since the product scope and schedule demanded a quick development life-cycle, most of the software architecture effort focused on the logical aspects of the software; however, limited design activities were conducted to evaluate the system's deployment and development perspectives. The logical architecture of the system—including mobile device and server PC—employs the client-server architectural style, where client subsystems send and receive data to and from a server subsystem via XML. Both client and server components (individually) follow the layered architectural style, as presented in Figures 2 and 3.

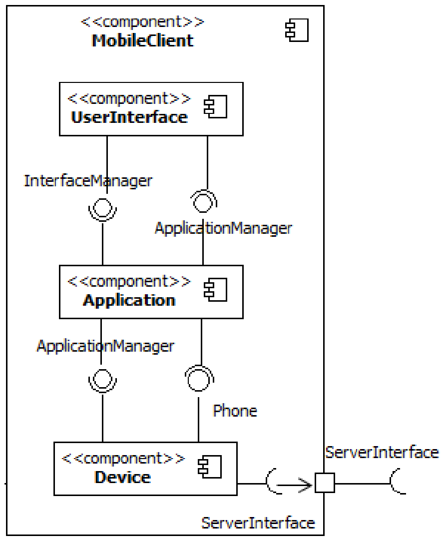


Figure 2. Mobile Client Subsystem Architecture

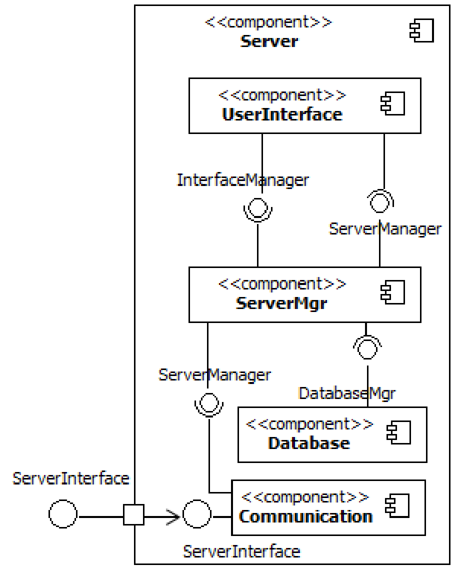


Figure 3. Server Subsystem Architecture

As seen, the client subsystem employs three different layers: the User Interface Layer, the Application Layer, and the Device Layer. The User Interface Layer contains the interaction elements of the client subsystem. The user will see and interact with elements of the UI Layer and all code that is specific to graphical interfaces is contained in this layer; for example, the UI screens for the Android-based device are written in XML and that code is only found in this layer. The UI Layer also contains elements to facilitate updates to the UI Layer when the Application Layer changes.

The application processing elements of the client subsystem is contained in the Application Layer. This layer handles the messages from the server, stores, and manages all of the data for the client subsystem. The Application Layer contains elements that update the UI Layer when it changes, after the UI Layer registers with the Application Layer. Finally, the Device Layer handles all of the features the hardware of the device supports. In this case, all of the features that come standard on an Android phone, for example, phone vibration, Wi-Fi, GPS etc.

Each layer provides an interface which serves as access point and application-specific messages are defined and used to transmit data between the layers. For example, the UI Layer can generate and send a Search Message to the Application Layer via the Application Layer Interface; the message is interpreted, converted to XML, and sent to the Device Layer to be sent to the server.

The server subsystem's logical architecture includes the following layers: user interface layer, server manager layer, database layer, and the communication layer, as presented in Figure 3.

The User Interface Layer for the server subsystem abstracts user interface capabilities provided through a web client; a program that is used to contact and obtain data or services from a server computer, represented as a Java desktop application. The Server Manager Layer handles the messages received by the communication layer and web client. It relies on concurrency to handle message so that it can process multiple database requests at the same time. The Database Layer contains the elements necessary to interface with the MySQL database that the system uses. Finally, the Communication Layer provides all the functionality to send and receive UDP and TCP data to and from the server.

Similar to the client subsystem, the layers in the server subsystem communicate via custom messages defined for the application. Messages sent/received to/from the client subsystem are transmitted using the UDP protocol and processed using the ServerInterface. For example, once an XML message is received (from the client) through the Communication Layer, the message is forwarded to the Server Manager component, which reads the XML contents and creates an internal message that can be passed on to the UI layer or processed by generating SQL queries and passing them on to the Database component. In case of the latter, the data received as result from the database query is converted to XML format and sent to the Communication Layer, which forwards the message to the Client System.

The driving forces behind these design choices were grounded on the customer's functional and non-functional requirements. Since portability and extensibility were of great concern, the client-server architectural pattern with XML messaging was chosen, since any mobile device that supports XML—which almost all modern devices do—can connect to the server. The deployment architecture for the system is presented in Figure 4.

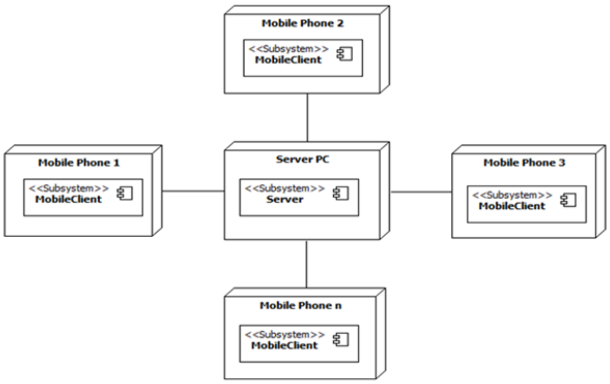


Figure 4. UML Deployment Diagram for the System

3.2 Detailed Design

Each component has its own detailed design which breaks the component down into classes, complete with attributes and functions. Detailed designs for both client and server subsystems employed several well-known design patterns, including the singleton, observer, and facade. For example, the Application Layer of the client subsystem and the Server Manager Layer of the server subsystem both relied on the observer design pattern for updating information in their respective subsystem's graphical user interfaces. Detailed design activities followed a strict, model-driven approach, where UML models were created and code was forward engineer. Changes to the code that led to changes on a given UML model were reversed engineered to incorporate the changes back into the detailed design. These capabilities were efficiently done using the Rational Rose UML tool. A sample detailed design for the Application Layer of the client subsystem is presented in Figure 5.

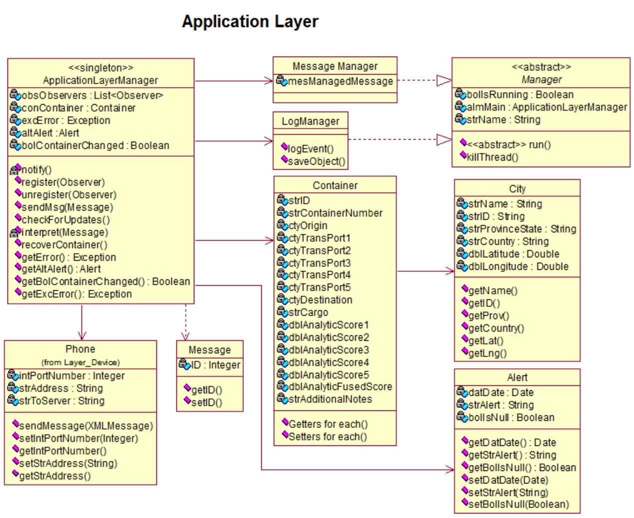


Figure 5. Class Diagram for the Application Layer UML Design

In order to increase the level of abstraction, a standardized message system was developed to handle the passing of data from one component to another. These messages—specified by the Message class—have a unique ID based on what type of message it is and each component supports the handling of those messages. By using the messaging system, components can be modified, replaced, or added to the system with little effort as long as they support the messaging interface. Also, new messages can be added easily, since they are designed as an independent class, so if a component requires a new message, all of the components would be able to see that new message immediately. All of these features and benefits of using the standardized messaging system support the customer's requirements for maximum portability and extensibility.

3.3 HCI Design

Upon establishing the requirements, software architecture, and detailed designs, the team began working on the Graphical User Interface (GUI) for the application. First, the team created low-fidelity prototypes using the Microsoft PowerPoint software. These prototype layouts provided low-complexity and interactive models of the proposed GUI; it supported clickable buttons so that navigation among screens could be presented to customers. This allowed customers to experience the user interface before committing to a particular GUI and allow them to provide feedback on what they thought the product should actually look and feel like. Following several modifications and approval from our client the team began coding the User Interface in XML.

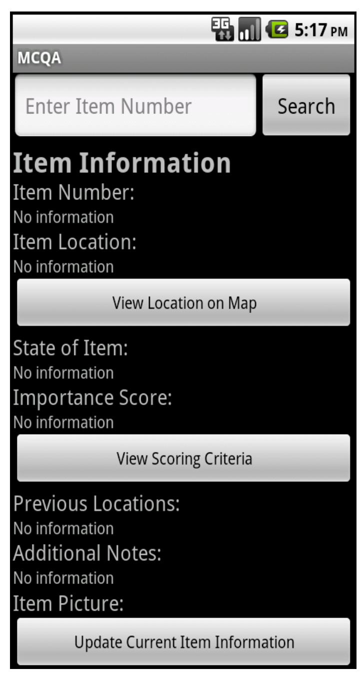


Figure 6. Item Information Screen which lets client see any information regarding the item of interest.

Droid Draw, an open source tool which provides a drag and drop interface to create GUI layouts and auto generated code for the Android devices, was the first step to the Android GUI. Once the team gained familiarity with the Android based XML, all coding was done strictly in the Android/Eclipse environment. After all design requirements were incorporated, screen shots of the UI were taken and sent to the client for validation. Through a series of requests and changes from the client, the team finalized the GUI. A sample GUI for the client application is presented in Figure 6.

4 Software Testing and Quality

Because of time constraints, most of the software testing focused on unit-testing, leaving little testing time for the systems as a whole. The unit tests were derived from the scenarios created for each use case. By reusing scenarios and elaborating on them, the creation of unit tests required little effort (for example, refer to Table I). A standard unit test template was developed by the team leads, executed by individuals responsible for the tests, and audited by the team's project managers. When bugs were found, they were assigned to individuals and results gathered and maintained by the testing group using the *ClearQuest* software tool.

Releasing a high quality product was very important to the team; quality validation began with the requirement phase where discussions dealt with various nonfunctional requirements present in the SRS. These Software Quality Factors were evaluated and translated to one or more nonfunctional requirement. For example, under the nonfunctional performance requirements (in the SRS) it states, SQM-1: The software modules shall be limited to 30 statements. This statement was based off the maintainability software quality factor. Other examples of supporting maintainability include the creation of a non-functional requirement to adhere to the Java Coding Style Guide [5]. This established several styles for construction, including styles for naming conventions, white spaces, bracket placement, etc., all intended to increase the software's maintainability. A list of other quality factors considered for the project is presented in Figure 7.

Software Quality Factors
• Efficiency
• Correctness
• Reliability
• Reliability
• Usability
• Maintainability
• Flexibility
• Testability
• Portability
• Interoperability

Figure 7. Quality Factors Considered for the System.

5 Configuration Management

A configuration management process was established to maintain the software and all supporting documents and information for the project. To support the configuration management process, a Basecamp [6] account was used for team collaboration. In addition, the open source tool Subversion [7] was employed to manage files and changes in the software. A change control process was devised in order to evaluate necessary and unnecessary changes and additions to the project. Controlling and managing changes was the primary tool for conducting peer reviews. The change process started with a request of a team member or from the client, from there the team evaluated the request and decided whether it would be implemented or not. Following the implemented change we validated the changes and incorporated the new changes to the project. This process is presented in Figure 8.

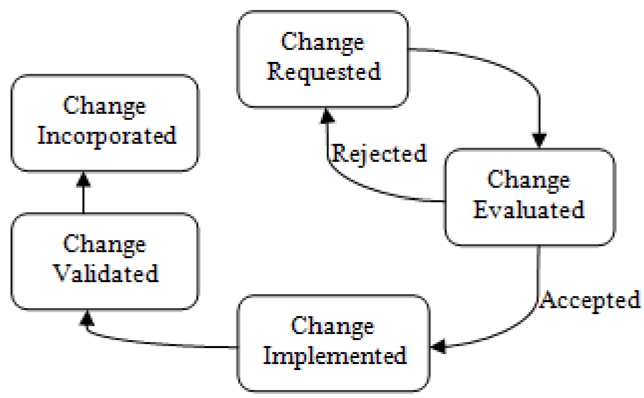


Figure 8. Software Change Control Process.

Change management and control was essential for choosing what the team wanted to add or change in the project since the client requested that the product would maximize the available technology, including geo locations to find positions of the items on the map via GPS and Optical Character Recognition (OCR) by utilizing the camera on the devices. This process helped us to decide what we had time for and what was realistic to finish before the deadline:

6 Lessons Learned and Conclusion

Throughout the year the team learned many lessons and expanded their knowledge of the software engineering profession. One of the primary experiences gained was the importance of working in a team environment and being held to a strict schedule. Before the Capstone Project, the team had limited experience working within a team, which is essential to the software engineering field. The team was very fortunate to have an opportunity to work with a client and to experience the complete development life-cycle to produce a working product. The team did not just learn about the basics during this course; many obstacles had to be overcome throughout the project, for example, transforming quality goals into requirements, learning and establishing a configuration management process, and learning about mobile programming.

For the project managers it was a first chance to lead a team and project. This was the first time that the team had an opportunity to create our own schedules and decide what the requirements would be. The team was required to spend much extra time, outside of the designated class time, in order to get some of the phases done so that the deadlines could be met. There was a lot of time spent learning the required languages that were needed to complete the project. The team had not experienced any mobile development before the start of the project; the team was required to learn java, XML, and Android. The team spent a majority of time researching all of the necessary information that was needed in order to develop this product. Most of the team needed to learn the tools and developing environments as well, this was the most in depth project that we had been required to do and required the team to learn a lot of very important tools.

Perhaps the most important lesson learned is the effects of changing requirements. Throughout the project the team needed to recognize that some of the requirements in the beginning did not provide sufficient specification for actual coding or developing the software system. Scheduling conflicts also existed which required scheduling changes, either a deadline being moved to a later date or to an earlier date, which were unforeseeable when the schedule was initially made. Overall, the team learned much more about how the real process of software engineering works and what it is to be in the job field. It has been a priceless experience and the team has learned more this year from this class than any of the members would have thought possible.

7 Acknowledgment

The students would like to express a sincere appreciation to The University of Virginia's College at Wise and the MCS department for support with this project. Also, we would like to thank the industry partner, which prefers to remain in anonymity given the nature of the work. Finally, the students would like to thank Dr. Otero for providing the industry contacts and overseeing the students' work.

8 References

[1] Australian Education International Glossary of Terms, www.aei.gov.au/AEI/CEP/UAE/Glossary/Default.htm, Last accessed on March, 5, 2011.

[2] Introduction to Software Engineering Body of Knowledge, http://www.computer.org/portal/web/swebok/html/ch2#Intro, Last accessed on March 15, 2011.

[3] IEEE Recommended Practice for Software Requirement Specifications, http://standards.ieee.org/findstds/standard/830-1998.html, Last Accessed on March 12, 2012.

[4] IEEE Standard for Information Technology - Systems Design - Software Design Descriptions, http://ieeexplore.ieee.org/servlet/opac?punumber=5167253, Last accessed on March, 12, 2012.

[5] Java Programming Style Guide, http://geosoft.no/development/javastyle.html, Last accessed on March 12, 2012.

[6] Basecamp Open Source, http://www.basecamphq.com/, Last accessed on March 12, 2012

[7] Subversion Open Source, http://www.subversion.tigris.org /, Last accessed on March 12, 2012.

A Peer-Supported Integrated Model to Promote High School Girls Learning Programming

Yi Fen Chen[1,2] **and Cheng-Chih Wu**[1]
[1]Graduate Institute of Information and Computer Education
National Taiwan Normal University, Taipei, Taiwan
[2]Taipei Municipal First Girls' Senior High School, Taipei, Taiwan

Abstract - *This paper is a report of our implementation of an integrated model to promote high school girls learning programming. Girls are believed having a social style of learning and learning better when working together and interacting with peers. In our peer-supported integrated model, we adopted the cooperative learning and pair programming approaches. Our goals are to have students gaining experience in problem solving activities, learning basic programming concept, and ultimately being interested in learning computer science. The evaluation results showed that the integrated model had positive effects on girls in learning programming. The students demonstrated higher motivation and initiative in learning, improved both thinking skills and cooperative skills, achieved quality projects, and increased task completion rates.*

Keywords: cooperative learning; pair programming; project-based learning; solo programming; peer-supported learning

1 Introduction

Programming is an essential component of computer science curriculum. Many studies have indicated that learning to program can have positive effects on students' thinking skills [1]. Thus, learning programming is not only to learn about computer science in its own right but also to learn about problem solving and about critical thinking. On the other hand, learning to program is considered difficult to most students because it is a complex activity involving multiple skills such as problem solving, IT usage, and a hierarchy of knowledge and abilities [2]. Although lots of instructional strategies have been proposed in the last four decades, teaching programming is still considered a major challenge [3].

In the senior high school computer curriculum of Taiwan [4], the required module outlines five key themes for students to learn: (1) computer hardware, (2) computer software, (3) computer network, (4) problem solving, and (5) computer and society. Three additional modules: "Basic Programming", "Advanced Programming", and "Topics in Computer Science" are offered as elective courses. One of the most important objectives of this curriculum is to teach students to learn about the problem-solving process. Problem solving is the recurring theme emphasized throughout the curriculum. Programming plays a major role in both the required module and the elective modules in the curriculum. It is considered as a mean to learn about computer science and about problem solving.

This paper is a report of our implementation of an integrated model to promote high school girls learning programming. Our goals are to have students gaining experience in problem solving activities, learning basic programming concept, and ultimately being interested in learning computer science. Computer science is traditionally a male dominated field. Woman entering to college and choosing computer science major has even dropped in recent years. In high schools, girls are not interested in computer classes and consider themselves as "not suitable" or "incapable" to learn computer science, specifically to learn programming. The reasons may be due to the disliked stereotypes of programmers (bearded, long hair, overweight, and dirty), the unfriendly working environment towards females, the discouraging support from the parents, and etc. However, one of the major factors may be due to the experience when they first encounter with the computer science subject-- they are not supported with proper instructional strategies.

Girls are believed having a social style of learning and learning better when working together and interacting with peers. Peer learning is an educational process where peers interact with other peers interested in the same topic. This instructional method helps the student in better understanding the programming concepts [5]. In our integrated model, we adopted the cooperative learning and pair programming approaches. Cooperative learning is an instructional strategy where students work in small teams, each with students of different levels of ability, using variety of learning activities to improve their understanding of a subject [6]. Pair programming is a variation of cooperative learning, in which two students work together at one computer. One (the driver) types in code while the other (the observer) reviews each line of code as it is typed in, and they switch roles frequently. Pair programming has been found to be very beneficial to student learning, particularly for women [7, 8, 9]. It can improve programming design quality, reduce defects, enhance technical skills and improve team communications. Pairing students are more confident in their work and enjoy it more [8].

Cooperative learning is often incorporated with Project Based Learning (PBL), in which, students work in small teams, each with different levels of ability, engaging in a variety of learning activities to improve their understanding of a subject [6]. PBL is widely used in computer science education. It is an instructional approach that engages student interest and motivation. Motivation could help students improve the academic skills they need to overcome the difficulties they encountered [10]. Project-based approach makes students achieve better and understand course objectives well [11].

In the following sections, we describe our integrated model in details, followed by the field implementation of the model and its evaluation results.

2 The integrated model

We designed our instructional model for teaching programming in four phases as depicted in Figure 1.

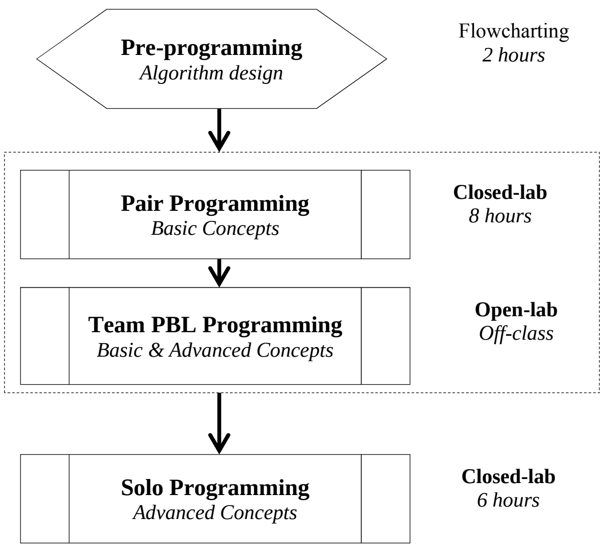


Figure 1. The Integrated model of teaching programming

The model starts with a pre-programming phase to teach flowcharting techniques. Since the transfer from algorithm design to writing codes in programming is a difficult step, teaching problem-solving techniques and algorithm development prior to programming is beneficial for learning in students' first encountering with programming [12].

The pair programming approach is conducted in the second phase to learn basic programming concepts, where two students collaborating in solving small programs is appropriate. When doing programming tasks, students sit side-by-side at one computer to complete a task together, take turns driving and navigating. The pair works together to achieve a common goal. Pair programming allows girls participating in group activities in a comfortable environment. The phase is designed for closed-lab activities.

In the third phase, a team, with three or four students, worked together to design and solve a real life programming project. The team PBL programming is performed in the off-class time (winter vacation in our case) to engage student interest and motivation. Students could find their own real world problems and solutions, set up their own learning goals, and search for solution. They can implement the project with the learned basic programming concepts, or with their self-learned advanced concepts. Students engage in design, problem solving, and decision making in the project.

Integrating peer-supported activities with project-based programming is a scaffolding strategy for enhancing student learning. We design the programming activities to involve different levels of cognitive activities as shown in Table 1. Cognitive skills learned revolve around knowledge, comprehension, and critical thinking of a particular topic [13].

Table 1. Cognitive domains involved in peer-supported activities

Phases	Activities	Cognitive Domains
Pair Programming	*Closed-Lab* T: Lecture, modeling S: Completing worksheets in pair.	Knowledge Comprehension
Team Programming	*Open-Lab* S: Completing a software project in team in off-class time.	Application Analysis Synthesis Evaluation

Note: T-teacher, S-students

The peer supporting was fading gradually till the fourth phase, where each student solves programs independently. During this phase, the instructor provides lectures and sample programs to explain the advanced concepts, and then the students complete their tasks individually in class. At this final stage, we expect students to experience the individual's efforts on developing programs as a typical programmer does.

3 Implementation of the model

In this section, we describe our implementation of the model in a senior high girls' school in Taiwan. More than 350 students who enrolled in a computer course participated in the study. The computer course was conducted hourly per week and lasted for a semester year, with a total of 36 hours. The instructor (the first author) allotted 16 hours for teaching programming by using the integrated model.

3.1 Pre-programming– Learning flowcharting

Flowcharting is a diagram technique to help students design and represent algorithm in problem solving process. Two exercises were provided for students to practice the technique. The first exercise is in real-life context which asks students to draw a flowchart to describe their after school agenda; the second is a mathematics problem about quadrants. Students are asked to describe the solution to measure which quadrant the specific point (x, y) will be in. They have already understood the rules as followed: The axes of a two-dimensional Cartesian system divide the plane into four infinite regions, called *quadrants*, each bounded by two half-axes. These are numbered from 1st to 4th and denoted by Roman numerals: I (where the signs of the two coordinates are I (+,+), II (−,+), III (−,−), and IV (+,−). It is interesting that everyone had different ways of thinking.

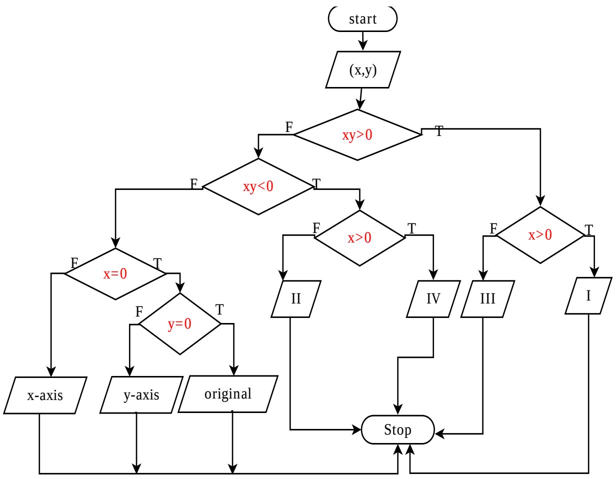


Figure 2. A flowchart of quadrant measurement by xy>0

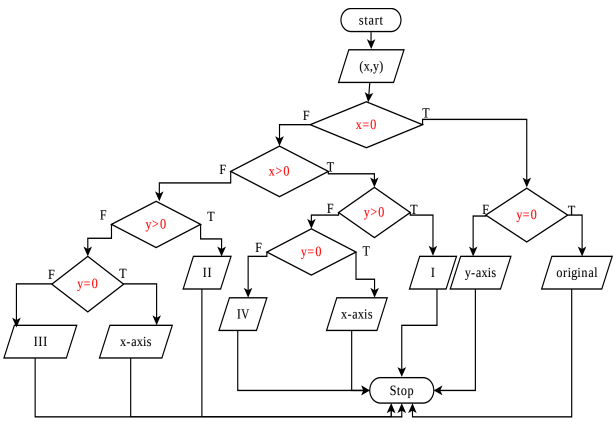


Figure 3. A flowchart of quadrant measurement by x value

Some students tried to draw their solutions by xy>0 (as Figure 2); some students used x value as the first condition (as Figure 3). Students realized that there is not only one flowchart (algorithm) for solving a problem.

3.2 Pair-programming– Learning basic concepts

The basic programming concepts taught in this phase are simple data types, variables, arithmetic expressions, basic input /output, and selection structures.

The initial teaching strategy is lecturing and modeling. The instructor first introduced the programming concept and then demonstrated the software development process by solving sample programs. The students observed the instructor's demonstration and built their own conceptual model of the process required for their assignments. Students worked in pair to practice programming by mimicking the process and modifying the sample programs. Learning occurs as students collaboratively engaging in these constructive processes. When doing the pair programming assignments, students were asked to record their results on the programming worksheets. In this stage, students gradually learned to transfer their algorithm thinking into programming codes. The implementation details of the pair programming phase are shown in Table 2.

Table 2. Pair programming for learning basic concepts

Basic Concepts	Teaching / Learning	Programming Problems
Data types, Variables, Arithmetic, Input/output	Lecture, Modeling	Calculate the circumference of a circle Calculate the average of two integer Convert Celsius to Fahrenheit Calculate the area of a rectangle with given base and length.
	Pair programming Laboratory	Calculate the volume of a ball Calculate the product of two integers Convert Fahrenheit to Celsius Calculate the area of a triangle, trapezium, square
Selection	Lecture, Modeling	Calculate Body Mass Index and provide indicators health experts recommend
	Pair programming Laboratory	Measure the triangle type Quadrant Measurement: Telling the specific point (x, y) will be in

3.3 Team PBL programming– Implementing real-world projects

The PBL activity was implemented during the winter break. Teams with three or four students were grouped randomly. Each member of a team maintains a particular role throughout the implementation of the project. The project leader should propose meeting agendas, suggest division of labor, and develop the project plan; the team members should come up with ideas, collect information, participate in discussion, and review resource materials.

3.3.1 Project assignments

The project is to build a small real-world system. This project is to be implemented with the basic skills (basic input/output; conditional structure) that students have learned and may involve aspects of iterative structure.

Teams are responsible for scheduling their own activities and deciding how to use their time to solve the problem and master the learning objectives. In terms of programming project, there should be a clear demonstration of how a project is derived, starting with a clearly stated problem statement, finding some possible solutions and going through the steps of problem analysis, algorithm design, coding, testing and debugging. Finally, students must write final reports into a required format which require placing the algorithm, flowchart, program codes, testing and correction. Following the project assignments, students clearly stated problem statement, found solutions and went through the steps of problem analysis, algorithm design, drew the flowchart, coding, testing, and debugging.

3.3.2 Supporting resources

To realize the benefits of peer learning, instructors must provide ‘intellectual scaffolding’.

a) Instructor’s support and external help: In our PBL, the instructor plays the role of facilitator, coaching both knowledge development and social skills, and carefully assessing what students have learned from the previous experience. Moreover, the instructor provides guidance, advanced programming concept access, and web resources to assist scaffolding. Frequently, the instructor acts as a mentor or tutor to the group, gets involved duly to navigate their thinking. In other words, the instructor is most active in providing immediate feedback via face-to-face discussions and online discussions. In order to achieve the goal of the project, students need to learn advanced skills such as repetition control structure, array data type, randomize and modules. These skills are almost never taught. Sometimes it is difficult to translate from their algorithm to programming code, the team members are allowed to seek help from computer science experts who are their instructors, parents or relatives.

b) Group Discussions: Group discussions provide for greater interaction between teammates. Students can meet face to face or meet online. Students are expected to help each other learn the material or solve the problem. Although students may have different abilities and background experiences, they are not given specific roles in peer tutoring, nor are students given information not shared by their teammates.

c) Project data bank: On PBL activity, instructor lists lots of projects to assist scaffolding. Students could acquire the experiences from projects. There were over 100 projects listed on the computer programming course support website as projects data bank.

3.3.3 Project quality measurement

The achievement of the project was measured according to the degree of effort in: (a) Problem solving strategies; (b) Programming skills; (c) Coding Style; (d) User Interface ; and (e) Team work.

We divided the grading system into five categories: (a) Excellent: Outstanding achievement; (b) Good: Above average achievement; (c) Satisfactory: Average achievement. The satisfactory student demonstrates more than minimum requirements of the course; (d) Pass: Below average achievement; and (e) Failure: Indicates that students’ work fell below a level of acceptance for the projects and was unsatisfactory.

3.4 Solo programming– Learning advanced concepts

In this stage, instructor provides lectures and sample programs to explain the advanced concepts, students need to finish their tasks individually in the class time. The courses are implemented in the second semester for six class hours. The examples of solo programming implementation of the introductory advanced concepts are shown in Table 3.

Table 3. Solo programming for learning advanced concepts

Advanced Concepts	Teaching / Learning	Programming problems
Repetition	Lecture Modeling	Print a serial numbers (for loop)
	Solo programming Laboratory	Simple game design: Guess a random number
Array	Lecture Modeling	Find the maximum score of the class
	Solo programming Laboratory	Are you there? Sequential search

4 Evaluation

All data were collected during the implementation of integrated programming course. We analyzed the final reports, programming codes, and a questionnaire with self-evaluation rating and attitude survey questions.

4.1 Quality of the projects

The performance of students in the projects was shown in Table 4.

Table 4. Performance of students in the projects

Grade scales	Achievement	No. of teams (N=100)
Excellent	Outstanding	34
Good	Above average	26
Satisfactory	Average	33
Pass	Below average	7
Fail	(No output)	0

More than half (60%) of the teams performed beyond our expectation. Although many teams referenced the system structure and code patterns from the project data bank, they displayed the originality in their system design.

As can be seen in Table 5, more than half of the teams used self-learned advanced concepts to complete the project. With asking for external help and referencing the project data bank, students learned advanced knowledge by themselves.

Table 5. The concepts applied in the projects

Learning modes	Concepts	No. of teams/Total teams
Taught	Selection	100/100
Self-learned	Repetition	56/100
	Array	46/100
	Randomize	29/100
	Modules	8/100

4.2 Initiative and motivation

As shown in Table 6, students chose their projects differently by interests. The most popular category was "Quiz system". More than one-quarter (27%) of teams worked in entertainment quizzes such as the questions of lyrics or the plots of a play (e.g., Movie, TV series, Cartoon, and Comics). Twenty-seven teams were interested in the quizzes of school subjects from textbooks (e.g., Chinese literature, English, History, and Chemistry) or quizzing about the behaviors or attitudes of their classmates or teachers.

Table 6. The categories of student projects

Categories	Projects of the study	Projects in the data bank
Games	13%	5%
Query Systems	15%	27%
Quiz system	72%	68%

The results also showed that students became more interested in adventure and scenario games as compared to the projects in the data bank. In addition to the time on individual work, students spent an average of 9.5 team working hours to finish their projects. Students contended that they worked very intensively because they are highly motivated to complete the task at hand during the session.

Because students were allowed to select the project of doing with their own interests, they were more willing to spend time on the task. It revealed that interesting projects would enhance students' initiative on programming. Students commented that:

"Programming is so interesting. I love it more and more especially when the program was done."

"I gained a deeper understanding of the computer via the interesting programming experience."

"The achievement gave me a great sense of accomplishment after we surmounted the difficulties. The PBL activities not only gave us the opportunity to learn programming skills, but also enhanced our ability to surmount the difficulties."

4.3 Thinking skills

In order to complete the project, students needed to overcome the difficulty they encountered. The teammates helped each other to resolve problems and provided emotional support to each other. Peer learning was a mediated tool to help students develop high order thinking skills, strengthen meta-cognitive skills, and apply programming skills in real life problem solving. Students commented that:

"This project assignment is a training designed for logical thinking. We must verify our thinking by testing again and again..."

"The programming course between my middle school and high school are quite different. In the middle school, we just typed in the sample program codes, we had no chance to develop independent thinking; on the contrary, we learned to think from the real life projects in high school programming course. Now, I love programming more and more."

4.4 Cooperative skills

In addition to the programming skills, students felt that they learned working in team, planning their work, and communicating with peers. Students commented that:

"This activity made us understand the importance of team cooperation".

"It is a big challenge that we had never encountered. It needed originality, brainstorming and pitching in. With the heart-warming cooperation with teammates, we were very glad that we completed the mission impossible!"

4.5 Programming tasks completion rates

The programming tasks completion rate of pair programming at the basic-concepts phase reached 89%, higher than the previous years. Furthermore, with the addition of the team PBL project activities in the winter break, the task completion rate of solo programming at advanced-concepts phase was raised to 94%. The results indicated that the peer-supported pair programming and team PBL programming had a positive effect on students' solo programming. Students had high confidence on accomplishing their task individually. When students begin to believe in their learning and work, they become motivated to learn more.

5 Conclusions

This paper outlines an integrated model to promote high school girls learning programming. As can be seen in the evaluation results, integrating pair programming and team project-based model had positive effects on girls in programming learning activities. The students demonstrated high motivation and initiative in learning, improved both thinking skills and cooperative skills, achieved quality projects, and increased task completion rates. Students were enthusiastic at the experience of pair programming, team project, and solo programming. They learned working in groups, planning of work, and cooperating with others. At the time of writing this paper, we surveyed the students who participated in the study two years ago and were about to entering college, sixteen percent of them admitted that their choosing of the majors in university were affected by the experience in this computer course.

6 References

[1] Richard E. Mayer, Jennifer L. Dyck and William Vilberg. "Learning to program and learning to think: what's the connection?" Communications of the ACM, vol. 29, no. 7, 605-610, Jul 1986.

[2] Tony Jenkins. "On the Difficulty of Learning to Program"; 3rd Annual Conference of the LTSN Centre for Information & Computer Sciences, Loughborough, LTSN-ICS, 2002.

[3] Andrew McGettrick, Roland Ibbett, John Lloyd, Gillian Lovegrove and Keith Mander. "Grand Challenges in Computing: Education– A Summary"; The Computer Journal, vol. 48, no. 1, 42-48, 2005.

[4] Janet Mei-Chuen Lin, Cheng-Chih Wu and Guey-Fa Chiou. "National policies and practices on ICT in Education--Chinese Taipei". In T. Plomp, R. Anderson, N. Law, & A. Quale (Eds.) Cross-National Information and Communication Technology: Policies and Practices in Education, 2nd edition, Greenwich, CT: Information Age Publishing Inc, 2009.

[5] Toni R. Black. "Helping novice programming students succeed"; Journal of Computing Sciences in Colleges, vol. 22, no. 2, 109-114, Dec 2006.

[6] Cooperative Learning Instruction Strategies, The online Teacher Resource, teAchnology, [online]. http://www.teach-nology.com/currenttrends/cooperative_learning/. (Accessed: 12 Nov 2011).

[7] Alistair Cockburn and Laurie Williams. "The costs and benefits of pair programming"; Proceedings of the First International Conference on Extreme Programming and Flexible Processes in Software Engineering (XP2000), 2000.

[8] Charlie McDowell, Linda Werner, Heather E. Bullock and Julian Fernald. "Pair programming improves student retention, confidence, and program quality"; Communications of the ACM, vol. 49, no. 8, 90-95, 2006.

[9] Linda L. Werner, Brian Hanks and Charlie McDowell. " Pair-Programming Helps Female Computer Science Students". ACM Journal of Educational Resources in Computing, vol. 4, no. 1, Mar 2004.

[10] Scheila Wesley Martins, Antonio Jose Mendes and Antonio Dias Figueiredo. "A Strategy to Improve Student's Motivation Levels in programming courses"; 40th ASEE/IEEE Frontiers in Education Conference, F4F-1 – F4F-7, Oct 2010.

[11] David Davenport. "Experience using a project-based approach in an introductory programming course"; IEEE Transactions on Education, vol. 43, no. 4, 443-448, Nov 2000.

[12] Robert Faux. "Impact of Preprogramming Course Curriculum on Learning in the First Programming Course"; IEEE Transactions on Education, vol. 49, no. 1, 11-15, Feb 2006.

[13] Benjamin S. Bloom. "Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain". New York: David McKay Company, 1956.

Learning the Concepts of Classes and Objects in a Game

Jinghua Zhang Elvira R. Caldwell Mustafa Atay Elva J. Jones
Department of Computer Science, Winston-Salem State University, Winston-Salem, NC, USA

Abstract— *Learning Object-Oriented Programming (OOP) is challenging for students who are new to programming. Many students have difficulty understanding the concepts of classes and objects, which are two main aspects of OOP. We developed a game-like instructional course module titled "Bionic Weasel". This module was designed to enhance students' understanding about classes and objects through game playing. This paper presents the design and implementation of the module and reports our experiences and results of using it in the lab. This module was used in CSC1310 Computer Programming I class in the Department of Computer Science (CS) at Winston-Salem State University (WSSU) in Fall 2011. To evaluate the impact of this game-like module, a pre-test, a post-test, and a survey were conducted. The pre-test and post-test comparison indicates that the learning game has significantly enhanced student learning on the OOP topics. Survey results and comments from students were also very positive.*

Keywords: Game-Based Learnnig, OOP, Classes and Objects, Programming, Computer Science Education

1. Introduction

Research shows that the fear of programming is one of the reasons why students do not want to major in CS. The traditional method of teaching Computer Programming focuses too much on the syntax of a programming language, which can be extremely difficult for students who are completely new to programming. Many innovative educational approaches have been proposed to motivate and engage students while achieving the desired student learning outcomes [1-4]. The survey conducted in [5] lists the most difficult topics in CS1. Sung et al. implemented some game-based lab assignments to enhance student learning on difficult programming topics like `arrays` [6-7]. Research show that learning difficult programming concepts in the gaming context will create high levels of enthusiasm and excitement about programming which will ultimately help students do better [8-9]. For example, UNC Charlotte Game2Learn project has been very successful in leveraging games in improving introductory computing courses [8-10]. Using games has become a very promising and effective approach to energize computing curriculum and engage students in the computing field. Some institutions incorporate gaming through computing curriculum [11] and some institutions create computer science game degree programs [12]

Our students like many others struggle with programming. When programming assignments are distributed, the frustrated cries often heard such as "Can you help me understand this? This problem is too hard to code. Please help me. How am I supposed to do this?" Visual learners struggle with the abstract concept of programming. Computer programming is seen by a vast majority of students as more difficult and time consuming, as well as less interesting than other courses of study. Learning to program is a time consuming task, as a very large number of hours must be spent at computer writing and debugging code (even if the student is already familiar with programming languages which, for the majority of our students, is not the case). Concepts such as structuring code, reusability, ease of maintenance, meaningful naming and user interface design need to be considered for all but the most basic programs. Unfortunately, this gives students a sense of information overload as well as a seemingly unstructured set of concepts to link together. The complexity of OOP adds a number of difficulties in addition to the problems of any first programming courses such as variables, identifiers, methods, and classes, and objects. Most freshmen entering the computer science major have no previous programming experience and some have only limited experience working with computers. Most introductory programming courses have a very high attrition rate. Introductory programming classes typically lose more than 50% of the students that enroll by the end of the course.

The test results of students in Computer Programming courses and the faculty observation urge us to try and implement an effective approach to transform the way the programming courses are taught and the way students learn at WSSU. Game-based learning has been very successful in the literature. We have not seen any games designed to help students learn the concepts of classes and objects, which our students struggle with most. We decided to develop a game-like instructional course module to help student have better understanding of this topic and improve their scores on the tests.

In section 2, we describe the Bionic Weasel game in detail.

Section 3 shows the results and reports our experiences using the module in the classroom. Conclusions and future works are described in section 4.

2. Bionic Weasel Game

Game development team in the CS department at WSSU consists of faculty, three undergraduate students and one graduate student. The advisory board oversees the implementation of the game-like instructional modules. The advisory board consists of faculty from department of computer science and department of education and professionals from game industry. The goal for this game is to enhance student learning of the concepts of classes and objects. After brainstorming the game ideas and producing the storyboard, the team decided to use the GameMaker 8.1 to implement the game and decided to the name the game as "Bionic Weasel". It took the team about three months to fully develop the game.

The core gameplay mechanism for this game is to let the player create objects using different classes to help the weasel avoid dangers with the goal of helping it exit the chamber safely. The player controls the bionic weasel created by the scientist. The scientist guides the player through each level of the gameplay by providing a tutorial or an example for each task.

The main menu of the game provides three options: *Start Game*, *How to Play* and *Exit Game*. Once the player selects the *Start Game* option, the player can choose to start a new game or load a previously saved game. The "*save game*" and "*load game*" features allow the player to save the game and continue to play at a later time. At the beginning of the game play, the player is asked to enter the banner ID number as shown in Fig. 1, which is a nine digit unique number at school. This number is used to generate a unique confirmation code at the end of the game to prove that the player has finished the game. This feature gives instructors an option to use the game as homework or an extra credit assignment outside the classroom. After the player cleared each level, the score summary will be displayed as shown in Fig. 2. The scoring is based on the number of objects the player has created and correctness of the statements. The player begins the game with three lives. The game consists of four levels and each level is designed to achieve specific learning goals.

2.1 Level One

The students are expected to gain the following skills after completing the level one of the game:

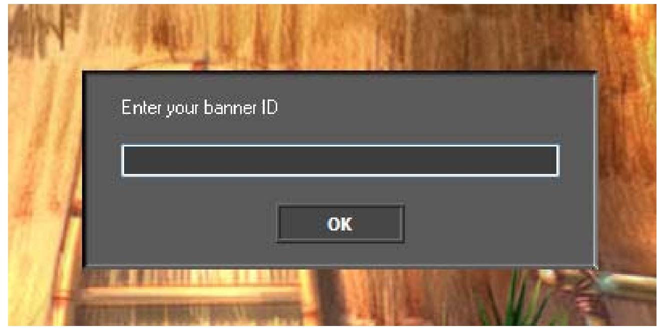


Figure 1. Enter a unique number to start the game.

Figure 2. Level Summary

1. Be able to create objects using different classes;
2. Be able to use constructors that accept one argument;
3. Be able to use the methods defined in the class;

In order to pass this level, the player needs to create objects to protect the weasel from falling spikes. The scientist provides step-by-step guidance and the player is expected to follow the guidance and learn from the scientist. There are two predefined classes: `Pipe` and `Cube`. The player needs to write a correct statement to create a `Pipe` or a `Cube` to protect the weasel. The player is presented a coding panel shown in Fig.3 to enter the statements. The statements must follow the syntax rules. A parser has been implemented to compile the statements. Immediate feedback will be provided to show the syntax errors. If there are no compile errors and the "RUN" button is clicked, the statements will be executed. For instance, in order to create a `Cube` object and make it float, the player needs to enter the statements, which are highlighted in gray as shown in Fig. 4. The `Cube` object is created as shown in Fig. 5 if the statements are correct. If the player enters a wrong statement as shown in Fig. 6 by calling a no-arg constructor, the game will provide an immediate feedback as shown in Fig. 7.

Figure 3. Coding Panel

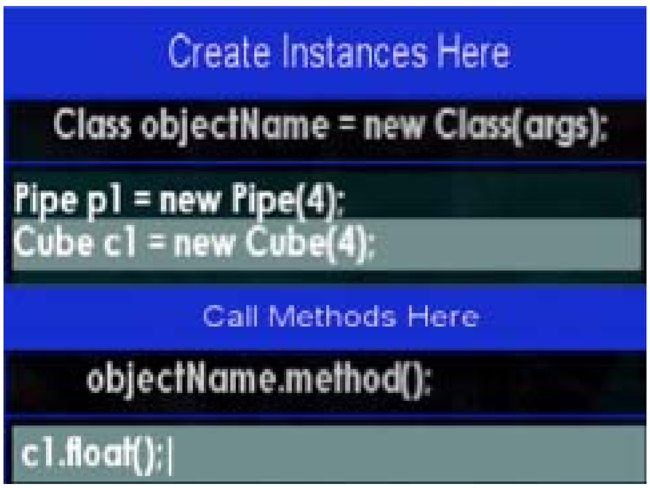


Figure 4. Statements to create a `cube` object and make it float

Figure 5. A `Cube` object is created.

Figure 6. Wrong Statement to create a `Cube`

Figure 7. Error Message

2.2 Level Two

The learning goal of this level for students is to gain proficiency of the concepts learned in level one. The difficulty of the gameplay increases by adding hot lava. The player has to create a `Cube` object to cover that in order to move on. This level also provides a checkpoint as shown in Fig. 9, which allows the player restarts at this point if the player loses all the lives instead of restarting from level one. The player can use the methods defined in the class to manipulate the objects such as rotating a pipe by calling `Rotate()` method in `Pipe` class as shown in Fig. 8. As a result, the pipe will be rotated as shown in Fig. 9.

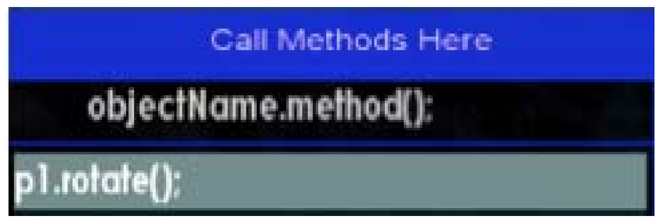


Figure 8. Calling `Rotate()` method in `Pipe` class.

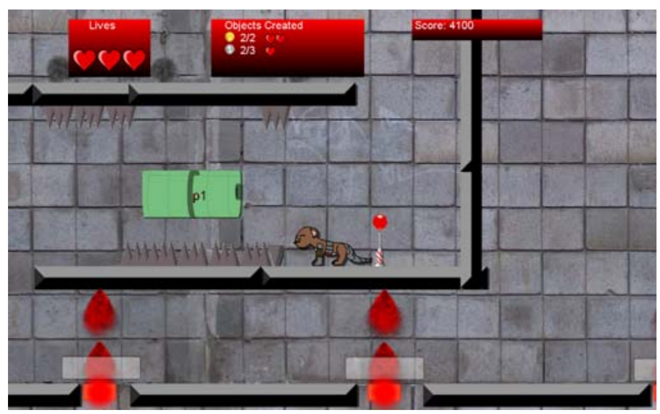


Figure 9. Rotated `pipe`

2.3 Level Three

The students are expected to gain the following skills after completing the level three of the game:

1. Understand the parameter passing rules;
2. Be able to use constructors that accept multiple arguments;

In level three, the player has to use the constructor with multiple parameters to create objects. The constructor with a single parameter is disabled in this level. The syntax of calling constructors with multiple parameters is a common source of error. The player will learn to provide arguments, which must be given in the same order as their respective parameters in the constructor. For example, the player needs to create a red `Cube` object at the right side of the weasel. The correct statements are shown in Fig. 10 and the result is shown in Fig.11.

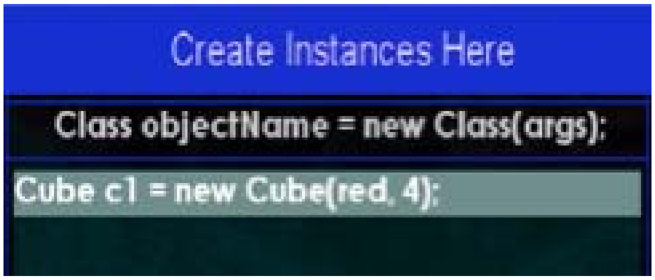


Figure 10. Using the constructor with multiple parameters.

Figure 11. A red `Cube` object is created.

2.4 Level Four

The learning goal of this level for students is to gain proficiency on the concepts presented in the previous three levels. The gameplay becomes more challenging in this level by adding additional spikes moving from left to right, which requires the player enter the correct statements faster.

3. Classroom Experience

3.1 Evaluation Results

This game-like instructional module was used in CSC1310 Computer Programming class at WSSU in Fall 2011. Eighteen students participated in this study during the lab session. 94% participants were African American students and 6% were Hispanic. The participants included five (28%) females and thirteen (72%) males. Most participants are traditional students, sixteen CS/IT majors, and two Math majors.

We do not have a controlled group in this study. All the participants had the lecture on the concepts of classes and objects in class by the same instructor before they came to the lab session. They took the pre-test, played the game and then took the post-test and the survey in one lab session (1 hour 45 minutes). Some students finished it early and some students used the entire session to finish the tasks.

Pre-test and post-test were similar as shown in Table 1. The questions were listed in blue. In both tests, students were expected to know the output of a given code segment which creates objects and uses methods defined in class. In the pre-test, students were tested on adding a constructor with two arguments, creating objects using the constructor with multiple parameters and using the methods defined in class. In the post-test, students need to be able to modify the existing constructor by adding an additional parameter and add an additional constructor. The post-test and pre-test results comparison of all participants is shown in Table 2.

The class average increased from 45.44 to 71.28. Table 3 shows the comparison results by gender. We found that 78% of participants showed improvements on the post-test and 60% of female participants got better grades on the post-test.

The survey questionnaire included seven questions regarding the features of the game. The choices on the questionnaire were Strongly Agree, Agree, Disagree, and Strongly Disagree. The Table 4 shows the percentage of students who marked Strongly Agree/Agree for each question. It is worthy to note that 100% of participants agree that the game helped them to understand the concepts of classes and objects. We are glad to find out that about 90% of students enjoyed playing the game. Participants also reported some bugs in the game and provided many useful comments in terms of how to improve this module.

3.2 Lessons Learned

Several lessons have been learned in this process. The gamers finished the module really fast and non-gamers had hard time manipulating the controls. The game interface needs to be simplified so that it will not become the barrier for non-gamers. The playtime should be no longer than twenty minutes. We also noticed that students who skipped the tutorial step showed poor results. The game should have allowed students to repeat the tutorials if students skipped them or didn't fully understand them.

4. Conclusions and Future Work

Since "Bionic Weasel" is the first game-like instructional module we have developed, we are very encouraged by the results. The comparison of pre-test and post-test results suggested that this module had a significant impact on student learning. This game module will be refined based on the feedback from the students, faculty and the advisory board. We plan to add more OOP features and concepts in the future. The refined module will be used as a learning activity in the Computer Programming courses in the upcoming semesters. It allows us to continuously assess the impact of this game-like instructional module on learning. It will be made available on our project website for instructors and students at other institutions.

5. Acknowledgements

This work was supported by NSF-HRD-1137548 grant and by WSSU RIP grant. Many thanks go to Amos M. Baker, Emanuel Smith and Matthew Perkins for their contributions to this project.

Table 1. Pre-Test vs. Post-Test

Pre-Test

```
/* The Pet class stores data for a pet. */
public class Pet{
   private String name; // A pet's name
   private String type; // A pet's type
   private int age;     // A pet's age

   public void setName(String n){
      name = n;
   }
   public void setType(String t){
      type = t;
   }
   public String getName(){
      return name;
   }
   public String getType(){
      return type;
   }
}
```

Q1. What is the output after the following code is executed? Use Scooby for the pet's name and dog for the pet type.

```
public class PetDemo{
 public static void main(String[] args){
  // Create a Scanner object for keyboard input.
  Scanner keyboard = new Scanner(System.in);

  // Create a Pet object.
  Pet myPet = new Pet();

  // Get the pet's name.
  System.out.print("Enter the name of your pet: ");
  myPet.setName(keyboard.nextLine());

  // Get the pet's type.
  System.out.print("Enter the type of your pet: ");
  myPet.setType(keyboard.nextLine());

  // Get the pet's age.
  System.out.print("Enter the age of your pet: ");
  myPet.setAge(keyboard.nextInt());

  // Display pet's information.
  System.out.println();
  System.out.println("Here is the information you provided:");
  System.out.println("Pet Name:"+ myPet.getName());
  System.out.println("Pet Type:"+ myPet.getType());
 }
}
```

Q2. Add a constructor that accepts the pet's name, type as arguments.
Q3. Fill in the blanks so that the constructor from Q2 is executed.

```
public class PetDemo{
  public static void main(String[] args){
   // Create a Scanner object for keyboard input.
   Scanner keyboard = new Scanner(System.in);
   // Create a Pet object.
   1. ___________________________________________

   System.out.println();

   // Display pet's name
   2. ___________________________________________

   // Display pet's type
   3.____________________________________________
 }
}
```

Post-Test

```
/*Circle class */
public class Circle {
  private final double PI = 3.14159; // Constant for pi
  private double radius;     // The circle's radius

   public Circle(double r){
     radius = r;
  }
  public void setRadius(double r){
     radius = r;
  }
  public double getRadius(){
    return radius;
  }
  public double  getArea(){
    return PI * radius * radius;
  }
  public double getDiameter(){
    return radius * 2;
  }
  public double getCircumference(){
    return 2 * PI * radius;
  }
}
```

Q1: Given the following demo program what would be the output?

```
import java.util.Scanner; // Needed for the Scanner class
public class CircleDemo{
  public static void main(String[] args){
   double radius;

   // Create a Scanner object for keyboard input.
   Scanner keyboard = new Scanner(System.in);

   Circle c = new Circle(2.0);  // Create a Circle object.

   // Display geometric info about the circle.
   System.out.println("The circle's area is"+c.getArea());
 }
}
```

Q2: Given the following demo program what is the output? The user enters 3 from the keyboard.

```
import java.util.Scanner; // Needed for the Scanner class
public class CircleDemo{
  public static void main(String[] args){
    double radius;

    // Create a Scanner object for keyboard input.
    Scanner keyboard = new Scanner(System.in);

    // Create a Circle object.
    Circle c = new Circle(0);

    // Get the radius of a circle.
    System.out.print("Enter the radius of a circle: ");
    radius = keyboard.nextDouble();
    c.setRadius(radius);

    // Display geometric info about the circle.
    System.out.println("The circle's diameter is "+
                     c.getDiameter());
 }
}
```

Q3. Declare an instance of the Circle class called ball.
Q4. How would you change the constructor to include an argument called color along with the radius?
Q5. How could you initialize color to blue without including it as an argument with radius in the constructor? Write the new constructor.

Table 2. Post-Test vs. Pre-Test Results Comparison

	Pre-Test	Post-Test
Mean	45.44	71.28
N	18	18
Std. Deviation	22.77	25.27

Table 3. Post-Test vs. Pre-Test Results Comparison by Gender

Gender		Pre-Test	Post-Test
Female	**Mean**	38	77.6
	N	5	5
	Std. Deviation	30.33	12.99
Male	**Mean**	48.31	68.84
	N	13	13
	Std. Deviation	19.90	28.74

Table 4. Survey Results

Features of the Game	Strongly Agree/Agree
1. The directions for the game are clear and easy to follow.	95%
2. The game is enjoyable to play.	89%
3. The game moves to the next level based on my correct responses.	89%
4. The game helped me to understand about classes and objects.	100%
5. The game functions well and is free from bugs (errors).	33%
6. It required a lot of effort for me to concentrate on classes and objects while playing the game.	45%
7. It was easy for me to navigate the game.	83%

6. References

[1] S. Yang, T. Breedlove, R. Carestia, Teaching introductory classes using LEGO robotics, *Journal of Computing Sciences in Colleges*, Vol. 19, Issue 2, pg 113-114, 2003.

[2] J. D. Bayliss and S. Strout, Games as a “Flavor” of CS1, *Proceedings of the 37th ACM technical symposium on Computer science education (SIGCSE'06)*, Houston, Texas, USA, 2006.

[3] J. D. Bayliss, Using games in introductory courses: tips from the trenches, *Proceedings of the 40th ACM technical symposium on Computer science education (SIGCSE’09)*, March 04-07, 2009, Chattanooga, TN, USA

[4] S. Leutenegger, J. Edgington, A Games First Approach to Teaching Introductory Programming. *In Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education (SIGCSE’ 07)*,Covington, Kentucky, March, 2007.

[5] Dale, N. B. Most difficult topics in CS1: results of an online survey of educators. SIGCSE Bull. 38, 2 (2006), 49-53.

[6] K. Sung, M. Panitz, R. Rosenberg, R. Anderson, Assessing Game-Themed Programming Assignments for CS1/2 Courses, *Proceedings of the Third International Conference on Game Development in Computer Science Education (GDCSE’08)*, Feb 2008.

[7] K. Sung, M. Panitz, R. Rosenberg, R. Anderson, CS1/2 Game-Themed Programming Assignments for Faculty, *Journal of Game Development*, Vol. 3, Issue 2, March 2008, PP. 27-47

[8] EAGLE M., BARNES T. Evaluation of a game-based lab assignment. *Proceedings of FDG 2009 - Fourth International Conference on the Foundations of Digital Games*, 26-30 April, Port Canaveral, FL, pp. 64-70.

[9] Michael Eagle, Tiffany Barnes, Experimental evaluation of an educational game for improved learning in introductory computing, *Proceedings of the 40th ACM technical symposium on Computer science education*, March 04-07, 2009, Chattanooga, TN, USA.

[10] Michael Eagle, Tiffany Barnes, Wu's castle: teaching arrays and loops in a game, *Proceedings of the 13th annual conference on Innovation and technology in computer science education*, June 30-July 02, 2008, Madrid, Spain.

[11] B. B. Morrison, J. A. Preston, Engagement: gaming throughout the curriculum, *Proceedings of the 40th ACM technical symposium on Computer science education (SIGCSE’09)*, Chattanooga, TN, USA, March 04-07, 2009.

[12] M. Zyda, V. Lacour, C. Swain, Operating a computer science game degree program, *Proceedings of the 3rd international conference on Game development in computer science education*, p.71-75, February 27-March 03, 2008, Miami, Florida.

Collaborative Learning Framework: A case study of an undergraduate database course

Vargas-Cerdán M. Dolores[1], Rojano-Cáceres J. Rafael[1], and Ramos-Quintana, Fernando[2]
[1]Facultad de Estadística e Informática, Universidad Veracruzana,Xalapa, México
[2]División de Ingeniería y Ciencias, ITESM Campus Cuernavaca, Cuernavaca, México

Abstract - *Teaching databases is one of the most important domains of all major curricula in undergraduate computer science programs. In order to have a good level of knowledge, the students need a solid background acquired in previous courses. Thereby, it is necessary to develop learning processes that aid students to retain by reinforcing and homogenizing the knowledge acquired. The Collaborative learning framework presented in this work aims to develop abilities to retain knowledge by developing cognitive skills, such as analysis, abstraction, synthesis and construction of structures, which are essentials in the database domain. In addition, the learning process is carried-out by building the knowledge collaboratively, which is synthetically expressed by a graph. This collaborative learning framework is being successfully applied since six semesters ago. A qualitative and quantitative oriented analysis was conducted, focusing on the retention of knowledge using this collaborative learning framework. Results showed that the students' knowledge reinforcement was improved***.**

Keywords: Collaborative learning; Retain Knowledge; cognitive skills.

1 Introduction

Databases curricula stress the development of different skills in order to offer the best solution to any information problem; the students need to analyze a problem, identify the data elements, place these elements into logical groups and establish the relationships between the resulting groups. In addition, the workplace demands professionals having abilities to encourage both the team-working and the development of essential cognitive skills aiming to solve real problems. Moreover, the search for information and the analysis of it are needed to reinforce the knowledge used for solving problems.

Collaborative learning has been defined by several authors as: a situation within which two or more people learn or attempt to learn something together [1] [2] [3]; in collaborative learning, the students work together to maximize their own learning [4]; collaborative group learning occurs when individuals interact through shared inquiry to construct their understanding of each other and their social worlds [5]. Collaborative learning, wherein students interact with peers to achieve certain objectives based on a set of activities, is a technique that has been shown to significantly improve the effectiveness of a learning experience [2]. Rather than working in isolation, students are organized in groups where certain interaction patterns are suggested. Tutors supervise this interaction and support students to guarantee progress towards the objectives. The research in [6] has shown that the learning in classrooms improves significantly when students participate in learning activities with small groups of peers [7].

The benefits of collaborative learning however are only achieved by active and well functioning teams [8]. One of the main challenges when designing a learning experience with collaboration is that of structuring the overall learning process as to trigger productive argumentation among students [2]. Therefore, an adequate learning framework that could take care of the learning process is needed. Edgar Faure, chairman of the UNESCO committee that produced the book Learning to Be [9], remarks that education must combine practical experience with academic studies, and must do it in a way that promotes self-education and thus prepares people for life-long learning. Proserpio & Gioia, 2007 in [10] mention than traditional collaborative learning provides this sharing of knowledge by learning through discussion, conversation, and comparison. The use of Information and Communication Technology (ICT) can contribute to universal access to education, equity in education, the delivery of quality learning and teaching, teachers' professional development and more efficient education management, governance and administration [11].

Collaboration is a coordinated and synchronous activity that is the result of a continued attempt to construct and maintain a shared conception of a problem [14]. Roschelle sees collaboration as the mutual engagement of participants in a coordinated effort to solve the problem together [12]. CSCL researchers believe that the focus of learning is on learning through collaboration with other students rather than directly from the teacher. [13]; Therefore, the role of the computer shifts from providing instruction to supporting collaboration by providing the medium of communication and scaffolding for productive student interaction [14]. Collaborative learning models are widely used in educational institutions. These models require a high interaction level among students and are mainly oriented towards class scenarios.

Nowadays, the collaborative tools may be clustered in the following main categories: application sharing, audio conferencing, chat and instant messaging, forum and online discussions, email, video conferencing, voting and surveying, web tour, whiteboard [15]. To build a collaborative process, such tools should be arranged and orchestrated in some way during a learning experience.

Several tools for working collaboratively have been proposed: [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27] and [28]. In this work the Collaborative Learning Framework is supported by a Collaborative Computer Tool called Collaborative Distributed Tool (CDT) developed by the research group *Computer Supported Collaborative Learning (CS for SDL)* from the *Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Cuernavaca and the Universidad Veracruzana,* specially for this Learning Framework. This paper reports on a quantitative and qualitative analysis that aims to improve the retention of knowledge, which is shown in the case study presented in this work.

The remainder of this paper is organized as follows: In section II we explain our case of study; Section III describes the data analysis and discussion of results; finally, in section IV, the conclusions and future work are exposed.

2 Case Study

The case study presented in this paper was included as part of a regular undergraduate program. The experience took place in three one-hour sessions during March 2010, in the Advanced Databases course of the undergraduate program of computer science at the Universidad Veracruzana.

The goal of the learning activity was to build collaboratively a network of concepts in the topic of Relational Model. The teacher's role was to analyze the built graph representing the network of in order to verify whether the retention of knowledge was improved using this collaboratively learning framework. The network of concepts is built starting by the extraction of underlying concepts, then pairs of such concepts are linking and finally the whole network is assembled. Thus, during this process, students develop skills such as analysis, synthesis and construction of structures. This last skill, construction of structures, reveals the capacity of students, if the structure is correct, to build semantic pieces of knowledge, where we can highlight qualitative aspects of the knowledge. We have to point out that the minimal unit of semantic knowledge is represented by the link of two pairs of concepts; in figure 1 we present an example, the concept of Attribute. The action of *grouping* is applied to this concept to obtain the concept *Relation*.

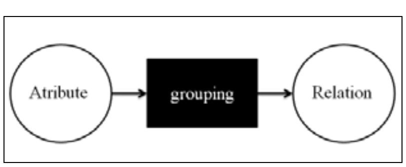


Figure 1. The minimal unit of semantic knowledge

The overall process of the retention of knowledge is illustrated in figure 2. As we can observe at the beginning of the process, the teacher provides the students with a topic to be reinforced. After carrying out an analysis (Rectangle I), the students make a self-identification of the main concepts, then in dyads they establish relations between pairs of concepts (Rectangle II). In the next step they start the construction of the network of concepts (Rectangle III) (The network of concepts has a form of a Petri net, this a graph is used as a representation of the knowledge reinforced); the output of this step is a network to be assessed by the instructor, he proposes tips or suggestions to update the network (Rectangle IV).The students analyze, synthesize, make abstractions and reorganize the structure to update the network (Rectangle V). The process is repeated until the end of the sessions or until the students get a coherent network of concepts.

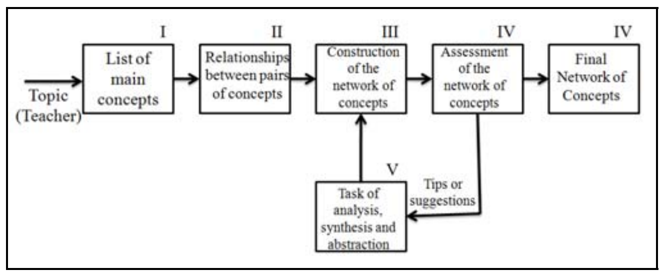


Figure 2. The learning process to reinforce knowledge previously learned

2.1 Students Profiles

A total of 13 participants took part in the learning process: 12 students and 1 tutor. The advanced database course belongs to the fundamental curses of the curricula of the computer science program.

The learning structure was then defined by dyads of students; the role of the tutor is to validate the coherence of the network that they have built. His participation is also oriented to assist students in the use of the computer tool and give them hints (at most two) about the construction of semantic structures. This last fact encourages autonomous attitudes to be assumed during the learning process.

Since all participants are computer science students, then we assumed that they had computer skills. We take care not to put together friends working as a team, the motivation for this organization is that collaboration with not close friends may increase the discussion effectiveness. Furthermore, the pairs were situated in different locations during the process.

2.2 Collaborative Learning Model

In this section we present the Learning model used to conduct our investigation.

At the beginning of the process, the teacher provides the student with the topic to be reinforced, after carrying out an analysis the students make a list of the most important concepts; then, students build basic relations between two concepts, such as, the concept of “tuples” to be related with the concept of “relation”, and finally, they connect all the concepts, in this way the network of concepts is built. The collaborative learning model is supported by the aforementioned collaborative computer tool CDT and it is Rich Internet Application (RIA) that can be executed for any web browser that have the Flash© plugging [29], at present it is available in the URL *collaborativelearningframework.net.* It is integrated by three elements: 1) A chat tool, 2) A shared working space, and 3) a bottom-up methodology. [30]. The knowledge being learned is represented by a network of concepts built by the students through a sequence of exchanged messages. The network of concepts represents a structure formed by a set of relationships between the main concepts of the topic being learned. In figure 3 are represented the elements of the process of Collaborative Learning. A coherent network of concepts is the tangible proof that the collaborative process of retention has been correctly achieved. In next sections we will explain the entire learning model.

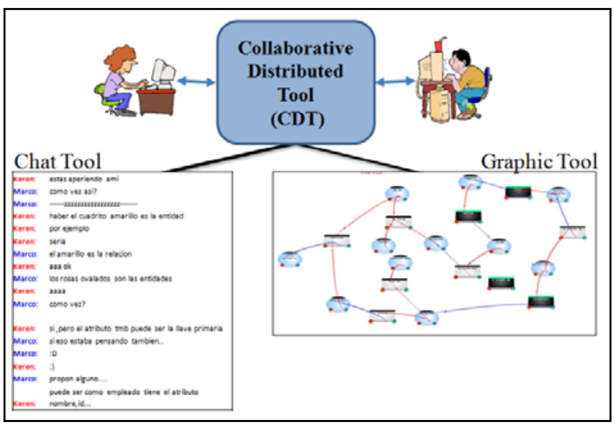


Figure 3. Collaborative Learning Elements

2.3 Design and procedure

The first session, of a total of three, the students learn how to build a Network of Concepts and how to use the Collaborative Learning Tool. The teacher provides the students with the topic to be reinforced and they individually make a list of the basic concepts that they remember about such topic. In the second session the students start the collaborative process building basic relations between two concepts. During the third collaborative sessions, the students built the network by linking all the concepts until a complete network is achieved. Directed links represent desirable correct relations between concepts. The network is built dynamically as the messages are being exchanged, because messages contain actions. A guide or mentor assists the students in order to verify the coherence of the network. The students could consult the mentor one time per session. Each session took 60 min. During three sessions, one month, the network was built.

3 Data analysis and discussion of results

In order to investigate whether there was a retention improvement of the students with respect to their previous learning, both a quantitative and qualitative analyses have been carried-out. The tangible results that we can get from the application of the Collaborative Learning Model are mainly three: 1) the initial list of underlying concepts that each student make individually; 2) The construction of pairs of concepts, resulting from the collaborative work; 3) The assembly of the whole Network of Concepts.

3.1 Quantitative analysis

3.1.1 Analysis of the concepts listed individually VS the concepts depicted collaboratively in the Network of Concepts.

In this case our materials were: the list of underlying concepts and the Network of Concepts. An example of a list of underlying concepts is illustrated in table 1. In Figure 4 the Network of Concepts is presented. The concepts have been translated from Spanish to English in order to facilitate the reading.

Table 1.An example of a dyads list of concepts

Student A	Student B
Relational Model	Attributes
Attributes	Relationships
Schema	Tuples
Domain	Primary Key
Cardinalities	Foreign Key
	Domain
	Cardinalities

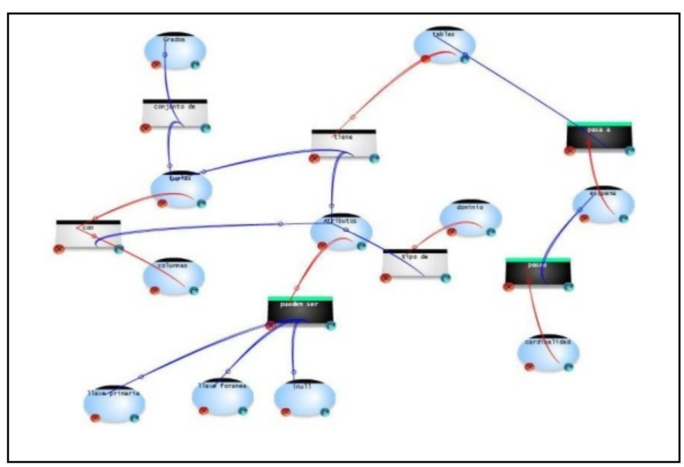


Figure 4. The final Network of Concepts

For this case, the final Network of Concepts presented in Figure 4 contains even more concepts than the list made individually. The concepts depicted in the network of concept are listed in table 2.

In table 3 are represented the information of the six couples, in the second and in the third column are listed the number on concepts that the students listed individually at the beginning of the learning process, and in the fourth column are the number of concepts presented in the network of concepts.

Table 2. The list of concepts depicted in the final Network of concepts

Concepts listed in the Network of Concepts
Attributes
Tuples
Primary Key
Foreign Key
Table
Domain
Cardinalities
Schema
Grade
Row

Table 3. The number of concepts listed by the students individually and after the collaborative work

Couple	Student A	Student B	Concepts listed in the Network of Concepts
1	5	8	9
2	7	5	16
3	6	7	10
4	14	12	14
5	5	8	9
6	7	9	10

Analyzing this information, we can say that there was an acceptable collaborative work, considering that they improved the list compared with that one they did in the beginning. In Figure 5 is presented graphically this case.

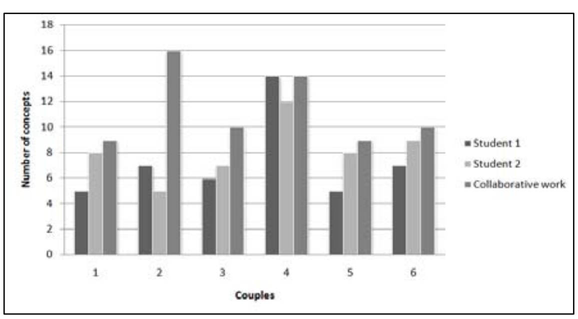


Figure 5. A graphic representation of the incremental number of concepts in the collaborative work

3.1.2 Analysis of the number of messages interchanged and the graphic elements that the students used to construct the network of concepts.

In this case our material for this analysis was taken from the dialogs by recording the messages and the movements performed in the working area. Table 4 shows this information. As we can observe, most of the teams has an equitable participation in the dialog as well as in the graphic work. The biggest difference belonged, in the case of the messages, to the couple 1. Meanwhile, the biggest difference belonged to couple 6, in the case of the construction of the Network of Concepts.

Table 4. Number of messages and graphic elements inserted during the collaborative work.

Couple	Number of messages intechanged		Graphic elements inserted in the Network of Concepts	
	Student A	Student B	Student A	Student B
1	82	49	18	13
2	25	21	38	23
3	30	28	19	16
4	72	57	26	19
5	13	16	24	9
6	42	36	2	26

3.2 Qualitative Analysis

In this case the material was the final Networks of Concepts, mainly we determined whether the students covered some relevant characteristics as following: 1) a list of the most important concepts of the subject; 2) the correct use of the graphic elements; 3) the relations between concepts were correctly established.

For the first case we got a list with the concepts that are part of the Relational Model, and from this list, we got the essential concepts, these were 8 concepts. According to this, we determine whether the Networks of Concepts is correct as a whole. Five of the six networks of concepts got the 86% of the essential concepts and just 1 got the 73%. In the second case, we got that the graphic elements were used properly, in just two cases the arrow direction was not correct, and in the third case, five of the networks presented one relation incorrect, one network didn´t present any wrong relation.

4 Discussion of Results

Comparison of qualitative and quantitative analysis indicates that the students' retention of knowledge was improved and reinforced. It is clear in the quantitative analysis that the students got better results after working collaboratively, they could remember more concepts. The qualitative analysis indicates that the Network of Concepts was constructed properly and completely, in this case we can affirm that the students developed cognitive skills, such as analysis, abstraction, synthesis and construction of structures.

5 Conclusions and future work

We have shown in this paper a case study of a Collaborative Learning Framework that is currently used in several courses of different undergraduate programs belonging to several universities of Mexico. The specific case study presented in this work was related with a course of databases. In this work, we argued that the retention of concepts is essential in the learning process. The combination of collaborative learning and the encouragement to develop skills such as analysis, synthesis and the construction of structures, among the most important, are essential aspects to improve the retention of concepts. During the bottom-up construction of the network of concepts the analysis skill is developed by students when they make an effort to extract the most relevant concepts of the topic under study. When the students relate pairs of concepts, they continue to develop the analysis skill in order to give a coherent relationship. At this level, the skill of synthesis starts to be developed because an abstract semantic meaning is composed of two underlying concepts linked correctly. This level needs already the development of structures. Finally, the analysis, synthesis and the construction of structures are strongly demanded when the whole network of concepts is assembled.

The quantitative and qualitative analysis, used to show how the collaborative attitudes improve the retention of concepts, has been based on the actions performed by the students to build the network of concepts. As future work, we aim to analyze the dialogs to find potential patterns that could reveal collaborative behaviors capable to assess the performance of the collaboration.

6 References

[1] Dillenburg, P., et al., eds. The evolution of Research on Collaborative Learning. Learning in Humans and Machines. 1996, Spada & Reimann (eds).

[2] Dillenburg, P., ed. Collaborative Learning, Cognitive and Computacional Approaches. Advances in Learning and Instruction Series, ed. E.D. Neville Bennet, Stella Vosniaou and Heinz Mandl. 1999, Van Someren, Reimann, Boshuizen & De Jong Elsevier Science Ltd.: Oxford, UK.

[3] Collazos, C.A., et al., Evaluating Collaborative Learning Process, in Department of Computer Science. 2001, Universidad de Chile: Santiago, Chile.

[4] P. Jermann, P. Dillengourg, "Elaborating new arguments through a CSCL scenario". Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning Environments, 2003, p. 205-226 Amsterdam: Kluwer, 2003.

[5] Ziegler, M.F., T.M. Paulus, and M. Woodside, Making Meaning Through Dialogue in a Blended Environment. Journal of Transformative Education, 2006(4): p. 302De-la-Fuente, L, Pardo, A., Asensio, J., Dimitriadis, Y. Delgado C., "Collaborative Learning Models on Distance Scenarios with Learning Design: A Case Study", International Conference on Advanced Learning Technologies, ICALT 2008, Santander, Spain, June 2008.

[6] Brown, A. & Palincsar, A. (1989). Guided, cooperative learning and individual knowledge acquisition. L. Resnick (Ed.), Knowledge, learning and instruction (pp. 307-336), Lawrence Erlbaum Associates.

[7] Soller, A., "Supporting Social Interaction in an Intelligent Collaborative Learning System," Int'l J. Artificial Intelligence in Education, vol. 12, no. 1, pp. 40-62, 2001.

[8] FAURE, E. et al. Learning to be: The world of education today and tomorrow. Paris: UNESCO, 1972.

[9] Albers, E., Braack, M., & Verseput, N.M. (2008). Computer supported collaborative learning. Retrieved from:
Retrieved in January 2011 from: http://www.eefjealbers.nl/CIMA/Pedagogies/Assignment_3_Albers_Braack_Verseput.pdf

[10] UNESCO, ICT in education, Retrieved in March 2012 from http://www.unesco.org/new/en/unesco/themes/icts/

[11] T. Matthews, S. Whittaker, T. Moran, and M. Yang. Collaboration Personas: A Framework for Understanding and Designing Collaborative Workplace Tools. In Workshop Collective Intelligence in Organizations: Toward a Research Agenda at Computer Supported Cooperative Work (CSCW), 2010, pp. 1-8.

[12] Chiu-Pin L., Lung-Hsiang, W., Yinjuan, S., Jitti, N., Chih-Jong, T. A 1:1 learning technology to support collaborative concept mapping: a case study of social studies lesson in elementary school. International Journal of Mobile Learning and Organization Volume 5 Issue 1, February 2011

[13] Roschelle, J. & Teasley S.D. (1995) The construction of shared knowledge in collaborative problem solving. In C.E. O'Malley (Ed), Computer-Supported Collaborative Learning. (pp.69-197). Berlin: Springer-Verlag

[14] Capuano, N.; Miranda, S.; Pierri, A.; Mangione, G.R.; Ritrovato, P.; , "Design and Execution of Dynamic Collaborative Learning Experiences," Intelligent Networking and Collaborative Systems (INCOS), 2010 2nd International Conference on , vol., no., pp.40-45, 24-26 Nov. 2010doi: 10.1109/INCOS.2010.29 URL: http://0-ieeexplore.ieee.org.millenium.itesm.mx/stamp/stamp.jsp?tp=&arnumber=5702075&isnumber=5702069

[15] Soller, A. (2004). Computational Modeling and Analysis of Knowledge Sharing in Collaborative Distance Learning. User Modeling and User-Adapted Interaction: The Journal of Personalization Research, 14 (4), 351-381.

[16] Soller, A., & Lesgold, A. (2003). A Computational Approach to Analyzing Online Knowledge Sharing

Interaction. Proceedings of Artificial Intelligence in Education 2003, Sydney, Australia, 253-260.

[17] Barros, B., and Verdejo, M.F. (2000). Analyzing student interaction processes in order to improve collaboration. The DEGREE approach. International Journal of Artificial Intelligence in Education, 11, 221-241.

[18] Barros, B., and Verdejo, M.F. (2000). Analyzing student interaction processes in order to improve collaboration. The DEGREE approach. International Journal of Artificial Intelligence in Education, 11, 221-241.

[19] Delium C. (2003), OSCAR: a framework for structuring mediated communication by speech acts. In: IEEE International Conference on Advanced Learning Technologies (ICALT'2003), Athens Greece.

[20] Baker, M.J. and K. Lund, Promoting reflective interactions in a computer supported collaborative learning environment. Journal of Computer Assisted Learning., 1997(13): p. pp. 175-193.

[21] Scardamalia, M. (2004). CSILE/Knowledge Forum®. In Education and technology: An encyclopedia (pp. 183-192). Santa Barbara: ABC-CLIO.

[22] Wang, D. & Johnson, P. M. 1994. Experiences with CLARE: A Computer-supported Collaborative Learning Environment. International Journal of Human Computer Studies 41(6), 851-879.

[23] Lesgold, A. Weiner, A., & Suthers, D. (1995, August). Tools for thinking about complex issues. Paper presented at the 6th European Conference for Research on Learning and Instruction, Nijmegen.

[24] Constantino-González, M. A. and Suthers, D. D. (2000). A coached collaborative learning environment for entity-relationship modeling. In Intelligent Tutoring Systems, Proceedings of the 5th International Conference (ITS 2000), G. Gauthier, C. Frasson, & K. VanLehn (Eds.), pp. 325-333. Berlin: Springer-Verlag.

[25] Miller, L., Eck, A., Soh, L., Hong, J. Statistics and Analysis Tools for a Computer-Supported Collaborative Learning System. 37th ASEE/IEEE Frontiers in Education Conference. October 10 – 13, 2007, Milwaukee, WI.

[26] Voyiatzaki, E., Margaritis, M., Avouris, N., "Collaborative Interaction Analysis: The Teachers' Perspective", Proc. 6th IEEE Int. Conf. Advanced Learning Technologies, July 2006, pp. 345-349.

[27] Hoppe, U. 2006. COLLIDE. Faculty of Engineering, University of Duisburg-Essen, http://www.collide.info/background

[28] Rojano, R., Ramos, F., García, G. Proposal for a Collaborative Distributed Tool for Building Network of Concepts. Puerto Vallarta : Alfa-Omega, 2010. XXIII Congreso Nacional y IX Congreso Internacional de Informática y Computación. pp. 383-388. 978-607-707-097-9.

[29] Ramos, Quintana, F., Sámano, Galindo, J. and Zárate, Silva, Víctor. Collaborative Environments through Dialogues and PBL to Encourage the Self-directed Learning in Computational Sciences. Marian Bubak, et al. Computational Science – ICCS 2008. s.l. : Springer Berlin / Heidelberg, 2008, pp. 725-734.

Learning Style and Students' Perception of Satisfaction on Web-based Learning Environment at Arabian Gulf University- Kingdom of Bahrain

Jefferson Gante Sanidad, MCP, DBM[1], Bernard Gante Sanidad, MIT[2], Sheryl Gante Sanidad[3]
[1] Computer Engineering Department, Gokongwei College of Engineering, De La Salle University- Manila, Philippines
[2] (Doctor in Information Technology Candidate) AMA Computer University,Quezon City Philippines
[3] (Master of Arts in Education Candidate) Manuel Luis Quezon Elementary School, Quezon City Philippines

***Abstract*— Many educators and the public are interested in online distance education, particularly web-based schooling. The underlying assumption is that this is a new and developing alternative modality of delivering distance education to accommodate various students who cannot attend the traditional method of learning where physical interaction is required.**

This study focused on the overarching question: "What are the associations between learning styles and students' perceptions of satisfaction among students enrolled in web-based learning environments?" To address this question, the researcher explored the associations between learning styles and students' perceptions of satisfaction in Arabian Gulf University- Kingdom of Bahrain.

***Keywords*— Distance Learning, Electronic Learning, Learning Style, Supervised Learning, Web-based Learning**

I. Introduction

The effectiveness of distance learning versus traditional classroom-based learning has been debated since the first generation of distance learning. As the use of technology to facilitate and deliver distance learning courses has increased, new challenges have emerged for the administration, faculty, staff and students of post-secondary institutions developing and implementing distance learning programs (Drazdowski, Holodick, & Scappaticci, 1998; Fulford & Zhang, 1993). Many professors fear distance learning is a means wherein administrators can justify the reduction of faculty positions to solve budget shortfalls. According to Novek (1996), many critics of distance learning fear the dehumanization and alienation of students as well as loss of critical thinking and social skills. Farrington (1999) posits that he does not accept that distant learning can be the equivalent of the face-to-face instruction. On the other hand, many other researchers have reported no significant difference between traditional classroom instruction and web-based instruction and that students feel closer to their faculty and peer students in web-based courses (Navarro & Shoemaker, 1999; Wade, 1999). Bloom and Hough (2003) postulate that a number of studies have indicated that web-based instruction decreases student learning levels. Even if there are no differences between distance learning and traditional learning environments, there is a need for additional research for the effectiveness of web-based instruction.

The Kingdom of Bahrain (in Arabic: مملكة البحرين , Mamlakat al-Bah-rayn, literally Kingdom of the Two Seas) is an island country in the Persian Gulf. Saudi Arabia lies to the west and is connected to Bahrain by the King Fahd Causeway (officially opened on November 25, 1986). Qatar is to the south across the Gulf of Bahrain. The planned Qatar–Bahrain Friendship Bridge will link Bahrain to Qatar as the longest fixed link in the world.

Bahrain has a total area of 665 km² (266 mi²), which is slightly larger than the Isle of Man, though it is smaller than the nearby King Fahd International Airport near Dammam, Saudi Arabia (780 km² or 301 mi²). As an archipelago of thirty-three islands, Bahrain does not share a land boundary with another country but does have a 161-kilometre (100 mi) coastline and claims a further twelve nautical miles (22 km) of territorial sea and a twenty-four nautical mile (44 km) contiguous zone. Bahrain's largest islands are Bahrain Island, Muharraq, Umm an Nasan, and Sitrah. Bahrain has mild winters and very hot, humid summers.

At the beginning of the twentieth century, Quranic schools (Kuttab) were the only form of education in Bahrain. They were traditional schools aimed at teaching children and youth the reading of the Qur'an. After the First World War, Bahrain became open to western influences and a demand for modern educational institutions appeared.

The year 1919 marked the beginning of modern public school system in Bahrain when Al-Hidaya Al-Khalifia School for boys was opened in Muharraq. In 1926, the Education Committee opened the second public school for boys in Manama and in 1928 the first public school for girls was opened in Muharraq.

In 2004 King Hamad bin Isa Al-Khalifa introduced a project that uses information communication technology (ICT) to support K–12 education in Bahrain. This project is named King Hamad Schools of the Future. The objective of this project is to connect and link all schools within the kingdom with the internet.

In addition to British intermediate schools, the island is served by the Bahrain School (BS). The BS is a United States Department of Defense school that provides a K-12 curriculum including International Baccalaureate offerings.

Bahrain has been at the vanguard of educational progress within the region, establishing the Gulf's first public education system in 1919 and giving its female population equal access to the Kingdom's educational resources. Schooling is free and compulsory for all Bahrainis and the Kingdom boasts an 87% adult literacy rate - among the highest in the region with almost 11% of all government expenditure directed towards education.

Education spending accounted for 10.8% of all government spending in 2006. Recent government reforms have included a teacher training program, a new polytechnic college, improvement of the upper-secondary vocational program and a quality assurance initiative which will raise accreditation standards and inspections for the education system and rank the performance of the system through regular national exams. The said reforms have been put in place to avoid a skills crunch further down the line. Over the next decade an additional 100,000 Bahrainis are expected to enter the job market.

A new Quality Assurance Authority for Education and Training (QAA) was launched in February 2009. The QAA will review and assess schools, universities and training institutes, as well as conduct national exams. The body will raise Educational standards within Bahrain. The QAA operates four monitoring units:

- Schools Review Unit
- Vocational Review Unit
- Higher Education Review Unit
- National Examinations Unit

Higher Education

As the platform for developing professional capacities and improving the career choices of Bahrainis, higher education takes the lead in providing cutting-edge skills necessary to thrive in an increasing global competitive environment. There are a vast number of specialist colleges offering courses in subjects such as finance, medicine, and IT.

- University of Bahrain
- Bahrain Institute of Banking and Finance
- DePaul University
- New York Institute of Technology
- RCSI-Medical University of Bahrain
- Arab Open University
- Royal University for Women
- Arabian Gulf University

The study will focus on the institution offering web-based learning in the Kingdom of Bahrain namely: Arabian Gulf University- Kingdom of Bahrain.

II. THEORETICAL FRAMEWORK OF THE STUDY

The first block represents the independent variable or the input. It serves as stimulus variable used to determine the relationship to an observed phenomenon. Here, all the data are gathered in preparation for the intervening variable. The independent variable in the study is the analysis and evaluation of the existing system. The second block represents the intervening variable where all the data are analyzed, evaluated and processed. The third block represents the dependent variable , which is the study itself "Learning Style and Students' Perception on Web-based Learning Environments at Arabian Gulf University- Kingdom of Bahrain". Overall outcome of the study are the evaluation that the university can get out of the study and most importantly the implications to the Philippine universities' web-based learning environment.

The study aimed to determine the "**Learning Style and Students' Perception of Satisfaction on Web-based Learning Environments at the Arabian Gulf University- Kingdom of Bahrain**".

Specifically, it sought to answer the following questions:

1. What is the demographic profile of the following respondents in terms of:
   a. gender
   b. age
   c. number of web-based courses taken
   d. number of hours dedicated to the web-based courses weekly
2. What are the learning styles of the student-respondents?

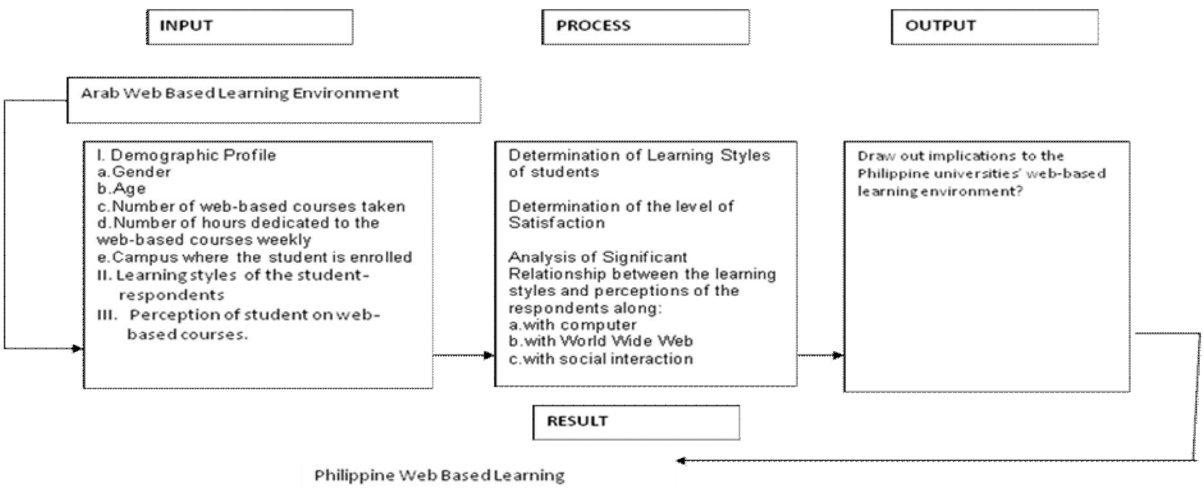


Figure I.
Paradigm of the Study

3. What is the level of satisfaction on the web-based learning environment as perceived by the student-respondents?
4. Is there a significant relationship between learning styles and the perceptions of satisfaction with computers among students enrolled in web-based courses?
5. Is there a significant relationship between learning styles and the perceptions of satisfaction with the World Wide Web among students enrolled in web-based courses?
6. Is there a significant relationship between learning styles and the perceptions of satisfaction with social interaction among students enrolled in web-based courses?
7. Is there a significant relationship between learning styles and the perceptions of satisfaction with instructor feedback among students enrolled in web-based courses?
8. Is there a significant relationship between learning styles and the perceptions of overall satisfaction among students enrolled in web-based courses?
9. What implications may be drawn for the Philippine universitiies' web-based learning environment?

Hypotheses

The following hypotheses were tested in the course of conducting this study:

1. The tests of association between learning styles and the perceptions of satisfaction with computers among students enrolled in web-based courses:
2. Tests of associations between learning styles and the perceptions of satisfaction with the World Wide Web among students enrolled in web-based courses.
3. Tests of associations between learning styles and the perceptions of satisfaction with social interaction among students enrolled in web-based courses.
4. Tests of associations between learning styles and the perceptions of satisfaction with instructor feedback among students enrolled in web-based courses.
5. Test of associations between learning styles and the perceptions of overall satisfaction among students enrolled in web-based courses.

III. RESEARCH DESIGN

The researcher used of the descriptive correlational method to explore the association between learning styles and students' perceptions of satisfaction in web-based learning environments. The components of the study consisted of four different parts of description and analysis. The first analysis was based on the demographics of the students enrolled in web-based courses at the Arabian Gulf University – Kingdom of Bahrain. The second analysis was on the computed values of variables related to the individual learning styles of the participants. The third analysis was an analysis of the computed values of variables related to students' perceptions of satisfaction in web-based courses. The final analysis used the Chi-square test of independence to test the association between learning styles and students' perceptions of satisfaction in web-based courses.

The researcher used three sets of variables that focused on the students' demographic profile, the students' preferred learning style, and students' satisfaction with web-based courses. The first set of variables focused on the students' demographic profile. These variables consist of information concerning gender, age, number of web-based courses taken, the number of hours dedicated to the web-based course weekly, and the university in which the student was enrolled. The second set of variables focused on the participant's preferred learning style. These variables were analyzed based on responses of the participants to Felder and Silverman's (1988) Index of Learning Styles (ILS) Inventory. The third set of variables focused on the participant's satisfaction with web-based learning environments. These variables were analyzed based on the participant's responses to Ham's (2002) web-based version of Hiltz's Distance Learning Perception Survey. Finally, the learning style variables and the satisfaction with web-based courses variables were analyzed to determine if a statistically significant association exists between learning styles and students' perceptions of satisfaction in web-based learning environments.

The Research Site

Arabian Gulf University- Kingdom of Bahrain was used as the research site for this study, the institution was chosen because the students enrolled are easy to reach by the researcher.

Arabian Gulf University also known as AGU is a university in Al-Manāmah, Bahrain, accredited by the Ministry of Education, Bahrain, and governed by Gulf Cooperative Countries, and is a member of Federation of the Universities of the Islamic World

Arabian Gulf University, a regional university that was founded to be a symbol for joint Gulf cooperation and an environment for fostering unity in the Gulf Cooperation Council. One of the main objectives of this university is that this academic institution, with its various colleges and programs, will participate in every way that it can in the enhancement of education and research, and the comprehensive development of the member states.

Materials for web learning are composed of in-house United Kingdom Open University (UKOU) and external academic contributors, and from third-party materials. For most courses, students are supported by tutors ('associate lecturers') who can provide feedback and tutorials on their work via face-to-face, by telephone, and/or the internet.

Respondents

The target population of this study includes students who enrolled in one or more web-based courses during the second semester of School Year 2011- 2012 under the graduate studies department of Arabian Gulf University-Kingdom of Bahrain taking up Distance Teaching and Training. Respondents were chosen at random.

Measures to ensure the confidentiality of the students were established. As a result, students' records and responses were kept strictly confidential and stored securely in a locked filing cabinet. Accesses were restricted to persons conducting the study. Upon completion of the study, the student data records and responses was destroyed to safeguard the identities of the respondents.

Instrumentation

The researcher used the following tools in order to gather data necessary for the proposed study:

Questionnaire and Evaluation Sheet. It is a written list of questions that the respondents will be made to answer. This is the main tool for the information gathering. The questions are partly fill-in the blanks and partly multiple-choice format.

A. Index of Learning Styles

Felder and Silverman's Index of Learning Styles (ILS) instrument was used to identify the preferred learning style of each participant. The ILS instrument was developed from Felder and Silverman's (1988) learning style model. The ILS instrument contains 44 questions that are designed to measure the participant's preferred learning style based on the Processing (Active-Reflective), Perception (Sensing-Intuitive), Input (Visual-Verbal), and Understanding (Sequential-Global) dimensions discussed in Chapter Two of this study.

The range of data associated with each learning style dimension is from 0 to 11 (e.g.,(A-R) Active-Reflective, (S-N) Sensing-Intuitive, (VS-VB) Visual-Verbal, and (SQ-G) Sequential-Global), resulting in 16 possible learning style preferences for each respondent. For statistical analyses, a scoring method that counts each dimension's corresponding value, (a) or (b), is used to determine the learning style preference for each dimension. The ILS instrument learning style dimensions are categorical data, meaning that a learner's preference on a given scale can be strong, moderate, or mild. For example, if a learner responds to the 11 questions relative to the Processing dimension at a rate of 9a (Active learning) to 2b (reflective learning), then it would be determined that the respondent has a strong preference towards the Active learning style versus the reflective learning style.

B. Index of Learning Styles Instrument Validity

Zwyno (2003), Livesay et al. (2002), and Van Zwanenberg, Wilkinson, and Anderson (2000) have tested the construct validity of the Felder-Silverman ILS instrument's individual dimensions using Cronbach's alpha test. A summary of the reported alpha levels from each study are listed in Table 3.2. With the exception of the .41 alpha value for the Sequential-Global dimension reported by Van Zwanenberg, Wilkinson, and Anderson, the ILS instrument yields alpha values that range between .53 and .70 for every learning style dimension.

Zwyno (2003) posits that the widely accepted social science cutoff value for alpha tests is $\alpha \geq .70$. However, Tuckman (1999) reports that it is acceptable for attitudinal tests to yield an alpha value that is greater than .50. As a result, Zwyno and Livesay et al. suggest that the ILS instrument is a suitable psychometric tool to assess the learning styles of individuals for the purpose of providing effective learning environments. As a result, this researcher accepts that the construct validity of the instrument is valid for the purpose of assessing the learning styles for the students enrolled in one or more web-based courses at the four community colleges.

Table I.
Cronbach's alpha values for ILS

A-R	S-N	VS-VB	SQ-G	SOURCE
0.60	0.70	0.63	0.53	Zwyno
0.56	0.72	0.60	0.54	Livesay et al.
0.51	0.65	0.56	0.41	Zwanenberg et al.

Adapted from "Applications, reliability and validity of the index of learning styles," by R. M. Felder and J. E. Spurlin, 2005, International Journal of Engineering Education, 21(1), p. 108.

C. Perceptions of Web-based Courses Survey

Hiltz's Distance Learning Perception Survey (HDLPS) instrument was developed in the late 1980s by Hiltz (1994) to evaluate the effectiveness of distance learning programs delivered in a closed network system. Hiltz (1994) designed the survey to measure student subjective assessment of computer-based courses based on Centra's (1982) literature review of teaching effectively and Paulhus and Christie's (1981) scales for measuring personal self-efficacy and interpersonal control. Hiltz postulated that conceptualized effectiveness occurs along course content, characteristics of teaching, course outcomes and comparison of the process in technology-based formats. While the Internet is a newer delivery method for collegiate programs and courses of study, the same principles addressed by Hiltz are equally pertinent today as when the computer-based distance learning courses were delivered on closed systems (Ham, 2002).

For the purpose of this study, the researcher chose to use Ham's (2002) revised version of the HDLPS to capture the data required to reflect the changes that have occurred in today's web-based courses of study The researcher asked permission to use Ham's web-based distance learning version of the HDLPS. The HDLPS used in this study focused on five sets of variables to gather data concerning students' perceptions of satisfaction in web-based courses including: (Two Sections) current feelings about computers (CEFFICACY), current feelings about the World Wide Web (WEFFICACY), social interaction (SINTERACT) (items 1-12 and 14 on survey), instructor feedback (IRESPONSE) (items 13 and 15-20 on the survey), and overall satisfaction (SATISFY) with web-based course environments (items 21-34 on the survey). Additionally, the HDLPS asked open-ended questions that addressed the "likes" and "dislikes" of the web-based environment. The open-ended questions were coded using frequency of response analyses.

HDLPS Validity

Ham's (2002) revised version of the HDLPS was reviewed by three distance learning professionals who analyzed the face validity of the survey. Based on their feedback, the decision was made to decrease the potential for response bias by negatively phrasing questions in the Likert-scale (items 22, 23, 24, 27, 28, 30 on the survey). Zwyno (2003) posits that the widely accepted social science cutoff value for alpha tests is **a >=.70**. The HDLPS yields alpha values that range between .78 and .89 for every component tested in this study. The reported alpha levels for the HDLPS are listed in Table 3.3.

Table II
Construct Validity of the HDLPS

Composite Scale	Cronbach's Alpha
Current feelings about computers	0.81
Current feelings about the World Wide Web	0.83
Social Interaction	0.78
Instructor Feedback	0.81
Overall Satisfaction	0.89

Target Population

The target population of this study was 162 students enrolled in one or more web-based courses at Arabian Gulf University-Kingdom of Bahrain. The researcher received 100 completed surveys from the target population, a cumulative response rate of 61.73%.

IV. SUMMARY OF FINDINGS

Hereunder are the highlights of the significant findings of this study:

1. Majority of the participants were male, 25 – 34 years old, enrolled in two or more web-based courses and spent 13 or more hours per week completing their web-based course assignments.
2. Half of the participants use computers and World Wide Web in their professions.
3. The most frequently used web-based learning activities were bulletin board discussions and group project.
4. In terms of the learning styles, majority of the participants fell between the following:
   a. Mild-Active
   b. Strong-Sensing
   c. moderate-Visual
   d. mild-Sequential
5. Majority of the participants' feelings about computers were positive: computers are stimulating (84%); computers are fun (77%), easy (76%), personal (78%), helpful (72%), efficient (75%), and reliable (78%).
6. Overall, the participants' feelings about the World Wide Web were positive. They felt that the World Wide Web is stimulating (90%); that it is fun (87%); easy (71%); personal (75%); helpful (59%); efficient (86%); and, reliable (76%).
7. The participants were asked to list what they "liked best" and "liked least" about their web-based courses. Majority of the participants, 70% (of 81 web-based courses), liked best ***group and team work***, and liked least ***too much busy work*** which is characteristic of web-based courses.
8. The tests of association between learning styles and the perceptions of satisfaction with computers among students enrolled in web-based courses:
   a. H1a-i: There is no significant association between the Active-Reflective learning style dimension and the perceptions of computers among students enrolled in web-based courses.
   b. H1j-r: There is a limited association between the Sensing-Intuitive learning style dimension and the perceptions of computers as being stimulating among students enrolled in web-based courses.
   c. H1s-aa: There is no association between the Visual-Verbal learning style dimension and the perceptions of computers among students enrolled in web-based courses.
   d. H1bb-jj: There is no association between the Sequential Global learning style dimension and the perceptions of computers among students enrolled in web-based courses.
9. Tests of associations between learning styles and the perceptions of satisfaction with the World Wide Web among students enrolled in web-based courses.
   e. **H2a-i**: There is no association between the Active-Reflective learning style dimension and the perceptions of the World Wide Web among students enrolled in web-based courses.
   f. **H2j-r**: There is no association between the Sensing-Intuitive learning style dimension and the perceptions of the World Wide Web among students enrolled in web-based courses.

g. **H2s-aa**: There is no association between the Visual-Verbal (VS-VB) learning style dimension and the perceptions of the World Wide Web (WEFFICACY) among students enrolled in web-based courses.

h. **H2bb-jj**: There is no association between the Sequential-Global (SQ-G) learning style dimension and the perceptions of the World Wide Web (WEFFICACY) among students enrolled in web-based courses.

10. Tests of associations between learning styles and the perceptions of satisfaction with social interaction among students enrolled in web-based courses.

i. **H3a-m**: There is limited association between the Active-Reflective (A-R) learning style dimension and the perceptions of social interaction (SINTERACT) among students enrolled in web-based courses.

j. **H3n-z**: There is no association between the Sensing-Intuitive (S-N) learning style dimension and the perceptions of social interaction (SINTERACT) among students enrolled in web-based courses.

k. **H3aa-mm**: There is no association between the Visual-Verbal (VS-VB) learning style dimension and the perceptions of social interaction (SINTERACT) among students enrolled in web-based courses.

l. **H3nn-zz**: There is a limited association between the Sequential-Global (SQ-G) learning style dimension and the perceptions of social interaction (SINTERACT) among students enrolled in web-based courses.

11. Tests of associations between learning styles and the perceptions of satisfaction with instructor feedback among students enrolled in web-based courses.

m. **H4a-g**: There is no association between the Active-Reflective (A-R) learning style dimension and the perceptions of instructor feedback (IRESPONSE) among students enrolled in web-based courses.

n. **H4h-n**: There is no association between the Sensing-Intuitive (S-N) learning style dimension and the perceptions of instructor feedback (IRESPONSE) among students enrolled in web-based courses.

o. H4o-u: There is no association between the Visual-Verbal (VS-VB) learning style dimension and the perceptions of instructor feedback (IRESPONSE) among students enrolled in web-based courses. .

p. **H4v-bb**: There is no association between the Sequential-Global (SQ-G) learning style dimension and the perceptions of instructor feedback (IRESPONSE) among students enrolled in web-based courses. .

12. Test of associations between learning styles and the perceptions of overall satisfaction among students enrolled in web-based courses.

q. **H5a-n**: There is a limited association between the Active-Reflective (A-R) learning style dimension and the perceptions of overall satisfaction (SATISFY) among students enrolled in web-based courses.

r. **H5o-bb**: There is no association between the Sensing-Intuitive (S-N) learning style dimension and the perceptions of overall satisfaction (SATISFY) among students enrolled in web-based courses..

s. **H5cc-pp**: There is no association between the Visual-Verbal (VS-VB) learning style dimension and the perceptions of overall satisfaction (SATISFY) among students enrolled in web-based courses.

t. **H5qq-ddd**: There is no association between the Sequential-Global (SQ-G) learning style dimension and the perceptions of overall satisfaction (SATISFY) among students enrolled in web-based courses.

V. CONCLUSIONS AND RECOMMENDATIONS

Distance learning is not a panacea, but as web-based courses continue to evolve, their potential for broadening access to higher education and training and development programs is compelling (Pantazis, 2002). Understanding web-based course delivery technologies in a world that demands opportunities for continuous learning, flexibility, convenience, and accessibility is increasingly important to higher education (Pantazis, 2002). While some teaching/learning models are very much like those used in a traditional classroom, electronic delivery makes the methods different enough to create new opportunities for evaluation (Offir & Lev, 1999; Pantazis, 2002). Consideration of this issue is based on the perspective that students should be taught using methods that maximize learning effectiveness. As a result, there is a need for research that examines which learning styles are effective in a web-based learning environment, as well as the association between students' preferred learning styles and satisfaction with web-based learning environments.

Recommendations

Additional research is needed with different populations to confirm the findings based on the association of learning styles and students' satisfaction of web-based learning environments.

1. Colleges must evaluate and assess the effectiveness of web-based course offerings in order to identify how technology impacts learning.
2. Future research could explore a variety of dependent and independent variables that impact the students' satisfaction of web-based environments.

3. Qualitative studies should be conducted to reveal "thick descriptions" about learning styles and students' satisfaction in web-based learning environments.
4. The exploration of the use of other instruments or tools could provide another dimension of the relationship between learning styles and web-based learning environments.
5. Additional research is needed to explore social interaction and instructor feedback of web-based learning environments.
6. Student evaluation on the effectiveness be included as part of the web-based course.

Implications to Philippine Universities and Colleges

Web-based learning is not yet a commonly accepted modality of education in most Philippine universities and colleges. Mainly because, before lessons can be delivered through the web, E-learning materials should be first developed. Many universities and colleges in the Philippines are ill-equipped with personnel who can develop e-learning materials.

There is therefore a need to strengthen first the capability of Philippine universities and colleges in making e-learning materials before this modality will be used to deliver distance learning. Another reason why web-based learning has difficulty being accepted in Philippine universities and colleges as a modality to deliver education is the poor ICT infrastructure with which to deliver web-based learning. While there are internet service providers availability of internet connectivity is unstable and missing in some remote places. Provision for mobile internet connection by internet service providers is available but quite expensive that only quite a few could afford.

Decisions about web based learning courseware must begin with an understanding of how the mind works during learning and of what research data tell us about what factors lead to learning. This is where decisions must begin. Naturally factors other than psychological effectiveness come into play in your multimedia learning decisions. For example, instructional strategies will be shaped by parameters of the technology like bandwidth and hardware, and by environmental factors such as budget, time, and organizational culture.

It may take some time for web-based learning courses to be adopted in Philippine universities and colleges but it is the future of education which will eventually become borderless and can be accessed anytime anywhere.

ACKNOWLEDGMENT

The researcher wishes to extend his sincere and profound gratitude, a genuine feeling of admiration to the persons who extended their untiring efforts to work together and made valuable contribution to the success of this study as well as institutions which serve as the fountain of knowledge like De La Salle University-Manila, Philippines, University of La Salette-Santiago City Philippines, Arabian Gulf University-Kingdom of Bahrain and AMA International University-Kingdom of Bahrain.

Heartfelt thanks to Dr. David R. Shockley, Executive Vice President Caldwell Community College and Technical Institute, for his invaluable guidance and help in the researchers' pursuit of this study. His willingness to share substantial ideas for the improvement of this study will never be forgotten.

To Dr. Fathi Abdulghader, for granting the researcher's request to conduct the study at the Arabian Gulf University- Kingdom of Bahrain.

To Mr. Adnan Ali Moh'd Alzawayed, I.T. Director of Arabian Gulf University-Kingdom of Bahrain for providing all the necessary data required for the realization of this study.

To the late Rev. Fr. Romeo B. Gonzales, MS, Ph D, former President of University of La Salette- Philippines for his valuable suggestions and support of this study.

To the researcher's parents, brother and sister, who were the researcher's source of strength and inspirations which brought about the success of this piece of work and for always being supportive in all his endeavors.

To our Almighty God, for being the source of wisdom, unconditional love and blessings.

REFERENCES

Bloom, K. C., & Hough, M. C. (2003). Student satisfaction with technology-Enhanced learning. *CIN: Computers, Informatics, Nursing, 21,* 241-246.

Ham, M. K. (2002). Students' perceptions of web-based distance learning courses. *Dissertations Abstracts International, 63*(04A), 3050298

Hiltz, S. (1994). *The virtual classroom: Learning without limits via computer networks*. Norwood, NJ: Ablex.

Livesay, G., Dee, K., Felder, R., Hites, L., Nauman, E., & O´Neal, E. (2002). Statistical evaluation of the Index of Learning Styles and Honey and Mumford's Learning Styles Questionnaires: How they compare and do they predict academic performance? *Psychology, 20*(3), 364-381.

Navarro, P., & Shoemaker, J. (1999). The power of cyberlearning: An empirical test. *Journal of Computing in Higher Education, 11,* 29-54.

Novek, E. M. (1996). Do professors dream of electronic sheep? Academic anxiety about the information age. (ERIC Clearinghouse on Resources ED399594).

Shockley, David Russel. (2005). *Learning Styles and Students' Perceptions of Satisfaction in Community College Web-based Learning Environments*

Zwyno, M. S. (2003). A contribution to the validation of score meaning for Felder-Soloman's Index of Learning Styles. Presented at the Proceedings of the 2003 American Society for Engineering Education Annual Conference & Exposition, Session 2351. Retrieved October 15, 2004, from http://ncsu.edu/felderpublic/ILSdir/zwyno_validation_study_.pdf

URI Online Judge: a New Classroom Tool for Interactive Learning

TONIN. Neilor A. [1], **BEZ. Jean Luca**[2]
[1]Computer Science and Engineering Department, Universidade Regional Integrada, Erechim, RS, Brazil
[2]Computer Science and Engineering Department, Universidade Regional Integrada, Erechim, RS, Brazil

Abstract - *The URI Online Judge is a new online tool created with the main purpose of making programming practice more dynamic, interesting and stimulating for those who have just entered into the art of programming. The URI Online Judge allows problem corrections in real time, interactivity between users, besides it allows flexibility in the choice of the programming language and it makes some supporting materials available. During the short time in which the tool has being used we have observed that it is a very good tool for self-study. As users of programming portals, the authors noticed some details that would be important to be implemented in a new tool, such as the separation of problems by categories. Another fundamental detail is the fact that this tool is available in two languages (Portuguese and English). This might facilitate the learning process for beginners, both locally and globally.*

Keywords: classroom tool, programming practice, online judge, informatics

1 Introduction

The major problem in the algorithms and programming teaching for new students is to efficiently meet the wide diversity of students and their different ways and pace of learning. One way to allow each student to learn on its own pace and speed is through the use of online tools for self-study/self-learning with high availability.

2 The Problem

In the traditional method of teaching algorithms, the teacher has to analyze logically, lexically and syntactically a solution developed by an undergraduate student. This is a time consuming process and may be flawed, since some errors or details may go unnoticed. In the Computer Science Course of Universidade Regional Integrada do Alto Uruguai e das Missões[1], some online tools have been used to enrich learning algorithms and programming languages. Such tools stimulate a healthy competition between students, by encouraging them to solve as many problems as possible, and helping the teacher in checking the correctness of a solution to a previously presented problem, since the correction becomes automated and in real time. Two of these tools are:

- BOCA Online Contest Administrator[2]
- UVa Online Judge[3]

The “BOCA” is an online judge for competitions. It is mainly used in the selective process of the Programming Marathon. It was used with Computer Science students of URI with learning purpose for a while. The students used it to solve problems about the contents developed in class. But every semester a clean out of the site was necessary and new problems, depending on the subject taught, had to be registered. So it is not feasible to work with that amount of problems because the system was not created for this purpose, and whenever the system was cleaned, both the ranks of the problems and the rank of the users were lost.

Another online judge was also used by our students, the UVa Online Judge[5] from University of Valladolid, Spain. This judge contains a huge variety of problems in all subjects of algorithms. However, its use was not feasible in class because the maintenance period in which the server was offline usually coincided with the class schedules, rendering the dependence on such a tool impossible. Sometimes, it is still used by students as a resource for programming competitions training.

3 The Integration Process

Our first idea was to create a website that had an automatic judge, able to meet the needs of our students. Besides, we also had to think in a way to improve the organization of the problems and to offer additional features, such as the categorization of subjects, availability of materials, tutorials on algorithms and programming, and eventually creating a forum that would allow collaborative learning.

In order to meet the diversity of students and their different ways and pace of learning, the URI Online Judge was built, mainly targeting interactivity, flexibility, new sources of information and usability. The portal can be accessed through one of the following addresses: *http://urionlinejudge.com.br/* or *http://urionlinejudge.edu.br/*. The main screen of the portal is shown in Figure 1, which summarizes the features found in it.

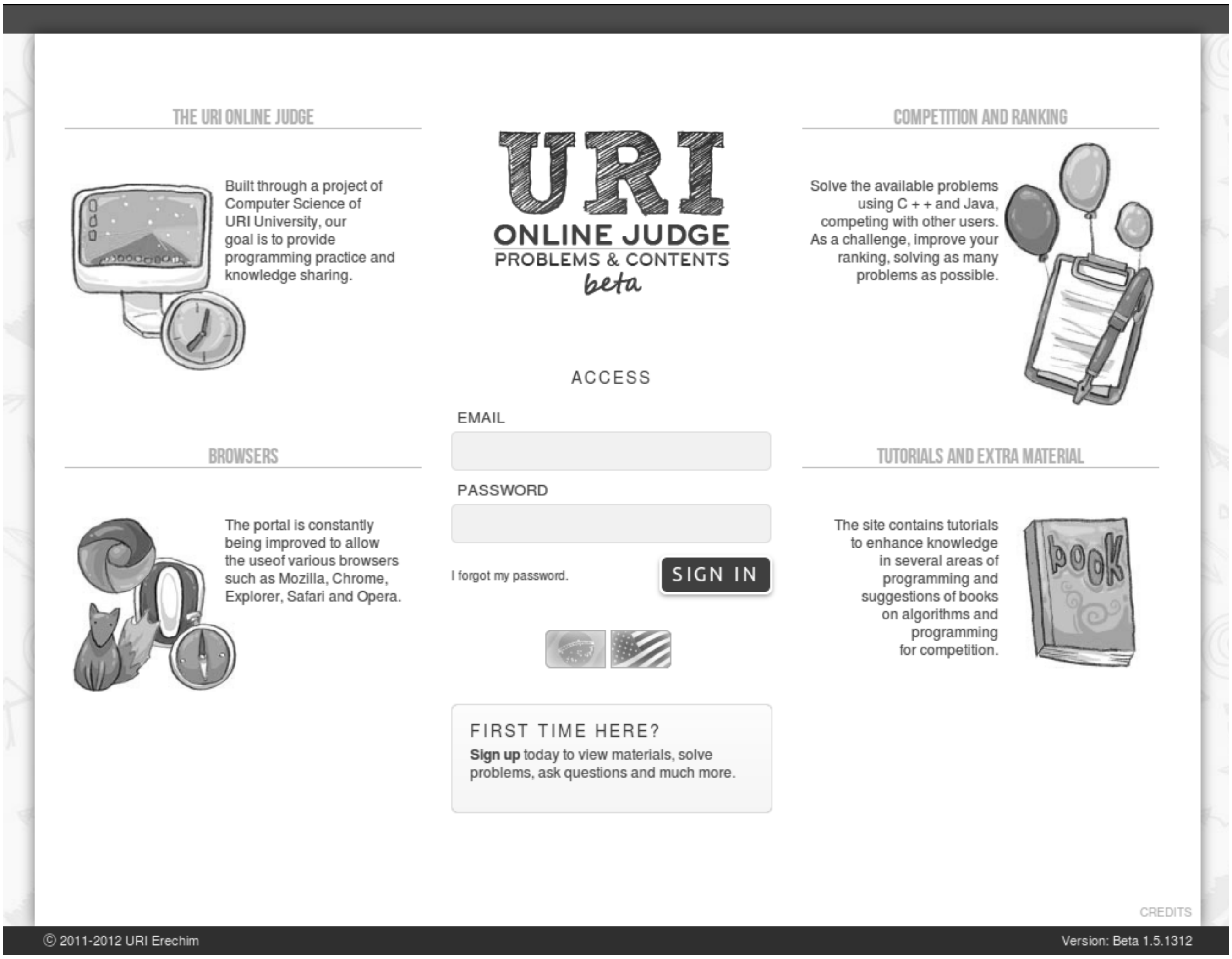


Figure 1 – URI Online Judge Login Screen

The portal, in its current version (Beta), is able to perform the following functions:

- display a list of problems divided into categories, with their respective subject and difficulties;
- receive new submissions and judge them;
- send source codes of submitted solutions to user;
- display problem rank and general rank;
- display user and real-time submissions;
- display user statistics;
- display user profile

The list of problems is divided into eight major categories (Figure 2), which facilitates navigation, guidance and focus on the student, in order to improve and develop their skills in specific subjects.

Besides the main category, it is important to present the topic related to the specific problem, especially for beginners, to familiarize them with the possible uses and applications of certain algorithms. It is also possible to hide the topics for advanced users.

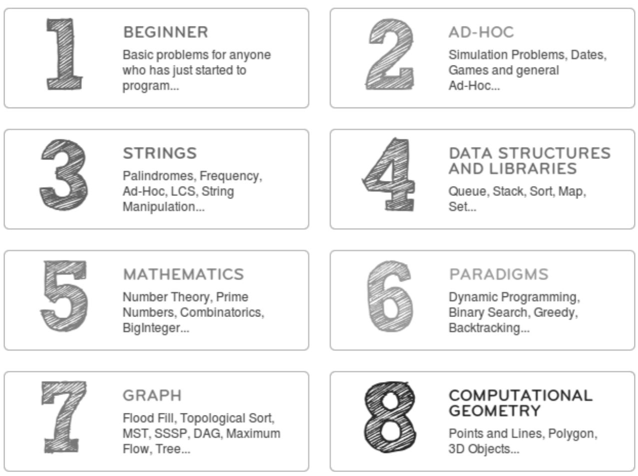


Figure 2 – Major categories

The level of difficulty of each problem is expressed in different colors, as seen in Figure 3. This allows students to focus on possible problems to be solved, taking into consideration their level of knowledge, without the risk of getting frustrated dealing more complex problems than those they are technically prepared to solve.

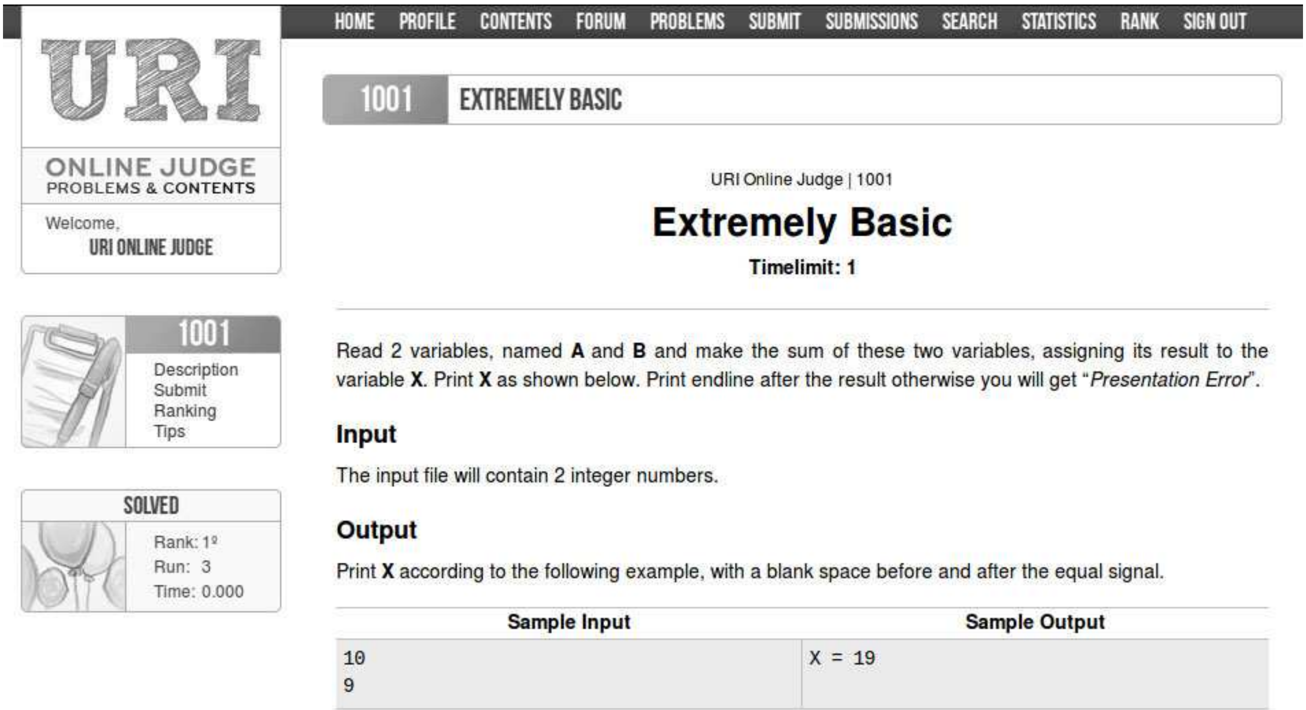


Figure 3 – Easy Problem (Level 1) – Pink Color

The portal allows to submit code in C + + and Java, by sending the file or pasting the source code directly in the field of the page, as shown in Figure 4.

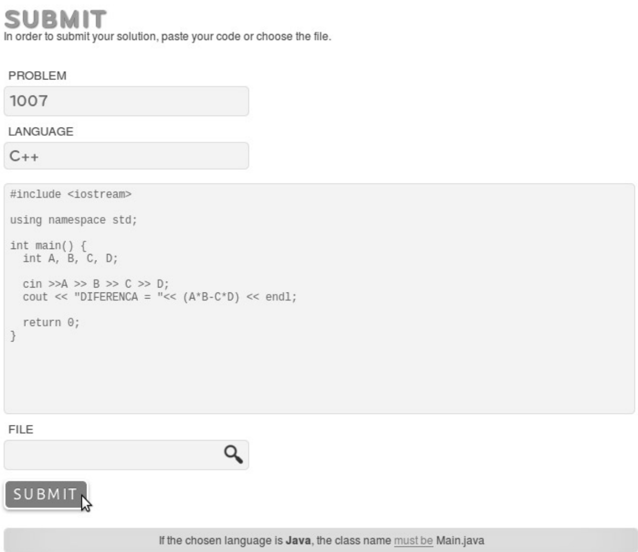
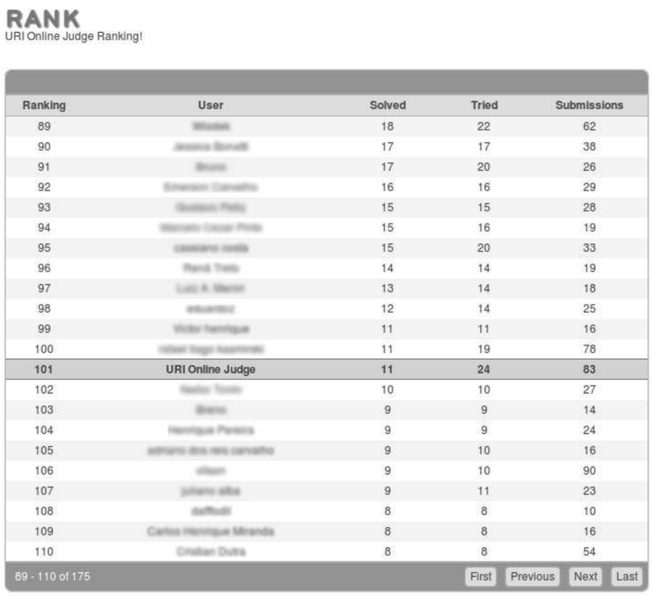


Figure 4 – Submission Screen for a C++ source code

Healthy competition takes place on the site through two ranks. The first one is a rank by problem and the second one is the general rank. Students try to solve the greatest number of problems to rise in the overall ranking (Figure 5), which counts the number of problems solved, using the amount of problems tried and overall submissions as tie-breaker.



RANK
URI Online Judge Ranking!

Ranking	User	Solved	Tried	Submissions
89	[illegible]	18	22	62
90	[illegible]	17	17	38
91	[illegible]	17	20	26
92	[illegible]	16	16	29
93	[illegible]	15	15	28
94	[illegible]	15	16	19
95	[illegible]	15	20	33
96	[illegible]	14	14	19
97	[illegible]	13	14	18
98	[illegible]	12	14	25
99	[illegible]	11	11	16
100	[illegible]	11	19	78
101	**URI Online Judge**	**11**	**24**	**83**
102	[illegible]	10	10	27
103	[illegible]	9	9	14
104	[illegible]	9	9	24
105	[illegible]	9	10	16
106	[illegible]	9	10	90
107	[illegible]	9	11	23
108	[illegible]	8	8	10
109	[illegible]	8	8	16
110	[illegible]	8	8	54

89 - 110 of 175 First Previous Next Last

Figure 5 – URI Online Judge Ranking

Students are also encouraged to improve their algorithms in order to make their solution more efficient, which is to run it in less time. Thus, it is possible to improve their position in the rank of each problem (Figure 6).

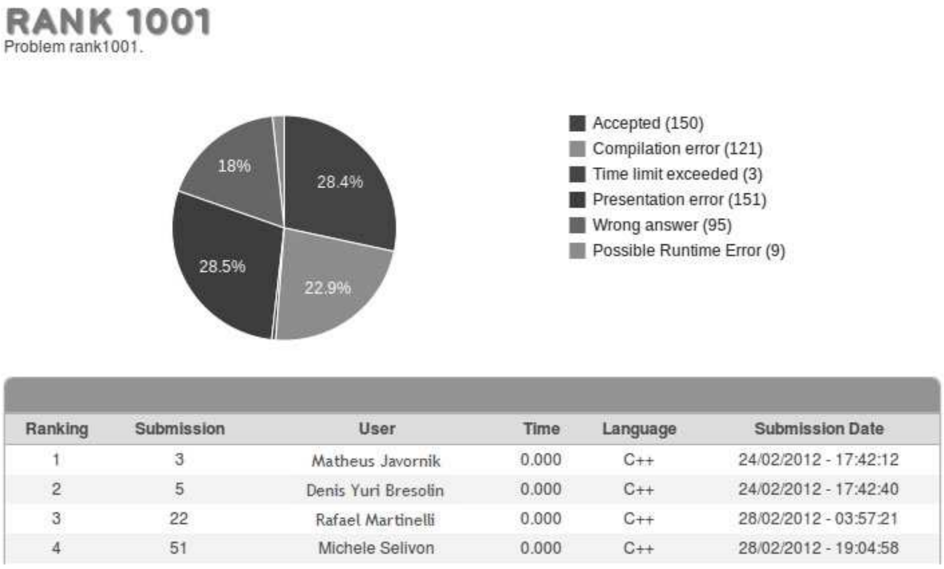


Figure 6 – Problem 1002 Rank

The statistics page presents two graphics, the first indicates the percentage of submissions in each answer, and the second indicates the percentage of submissions in each language available (Figure 7). The possible answers are: Accepted, Presentation Error, Runtime Error, Compilation Error and Wrong Answer. This screen also presents the statistics of solved problems in each of the categories of the website.

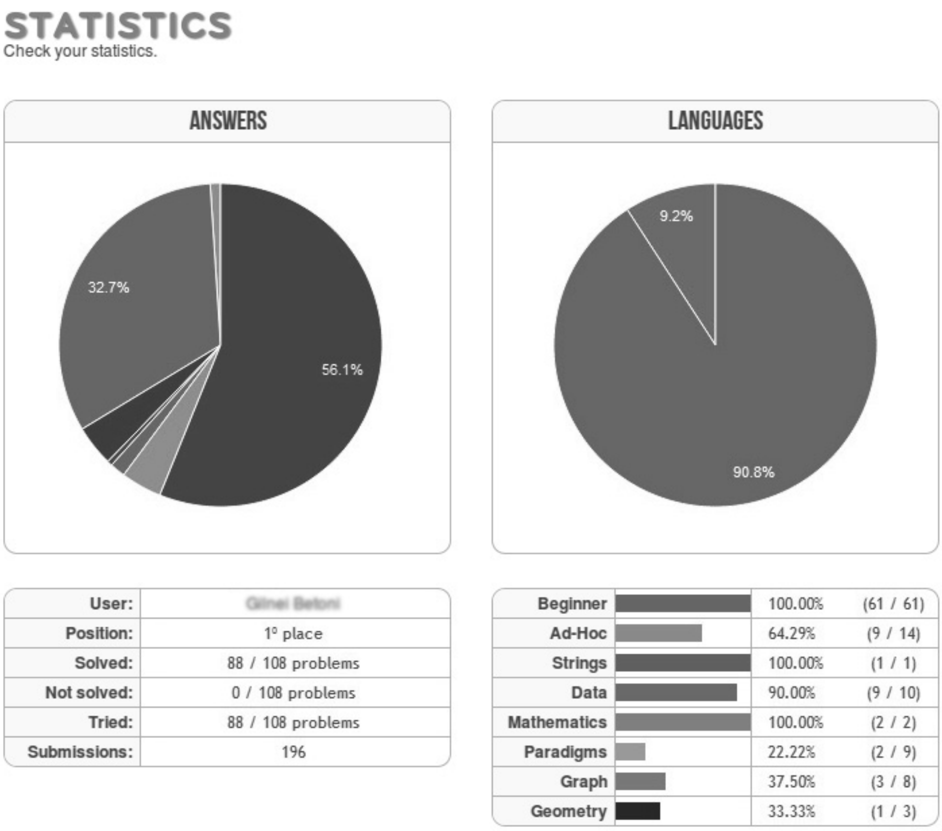


Figure 7 – Statistics related to user submissions

On the user's profile (Figure 8), it is possible to have a greater integration between the members of the portal, where it is possible to view some user's data, such as name, position in the overall site rank, number of solved problem, tried but not solved problems, the total number of submissions for that user and a listing of the problems solved with their respective run time.

As the portal is a web based tool, it is modified and improved every day, to better suit the needs of all users. The site is available in two languages, English and Portuguese, with its default language in English. This provides that students can practice and improve the language widely used in the field of computing, and make the online environment more accessible to users worldwide.

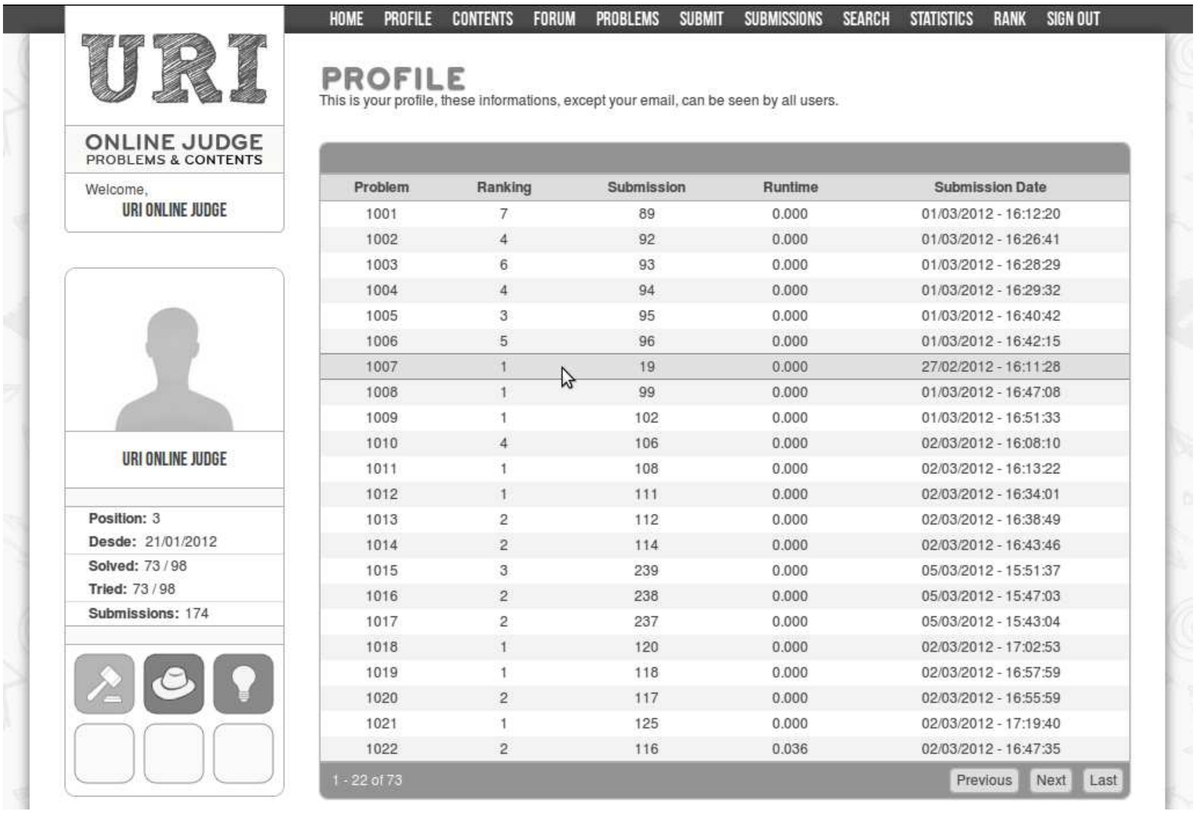


Figure 8 – User profile

4 Conclusions

During some tests performed in the classroom, with the new system, it was possible to observe the reaction of students, which was very enthusiastic. Their involvement in solving more problems to go upwards in rank, even though some of the problems had already been solved by them in previous subjects attended, was clearly noticed.

Every effort is being taken to assure that the portal will evolve substantially, so it has been upgraded continuously, with more information, more features and functionality enabling greater interactivity between users, contributing to their learning. It is crucial to mention that the portal was built having in mind the needs and difficulties of the students. We believe that the portal might contribute for the student to become a better professional in Computing Science field.

5 References

[1] Universidade Regional Integrada do Alto Uruguai e das Missões, Campus de Erechim, Erechim, Rio Grande do Sul, Brazil, http://www.uricer.edu.br/.

[2] Campos, Cassio P. de; Ferreira, Carlos E. BOCA: um sistema de apoio a competições de programação, 2004.

[3] UVa Online Judge, Online Judge and Contest system developed by the University of Valladolid (Spain), http://uva.onlinejudge.org/.

[4] Maratona de Programação da SBC, Maratona Home Page, http://maratona.ime.usp.br.

[5] Miguel A. REVILLA, Shahriar MANZOOR, Rujia LIU; Competitive Learning in Informatics: The UVa Online Judge Experience. International Olympiad in Informatics (IOI) Conference. El Cairo, Egypt. 16-23 August 2008.

Supporting active learning in large classrooms using pen-enabled computers

Sam Kamin   Wade Fagen
Department of Computer Science
University of Illinois
Urbana, IL 61801 USA
{kamin, wfagen2}@illinois.edu

ABSTRACT

Employing active learning methods in a large college class presents a variety of challenges. One of the most important is that it is difficult for the lecturer to properly calibrate the learning activities. We present our experience in such a class – an advanced, required class in Computer Science, with an enrollment of 150 students – where we employed a computer system based on pen-enabled computers to provide real-time feedback to the professor. The system helped to keep the students engaged, while having them engage in realistic, homework-like problems in the classroom. We describe the system and our experience with it.

KEYWORDS

Software Framework, Active Learning Tool, Collaborative Learning Tool

1. INTRODUCTION

The implementation of active learning in large college classes poses special challenges [1]. Some of these can be addressed with technology. Electronic voting systems (``clickers'') are becoming quite popular; these involve students in solving multiple-choice problems in the class, usually collaboratively. This keeps students more engaged in the class --- which in the modern college classroom has become a difficult task for all but the most entertaining instructors. However, in some circumstances, clickers are difficult to employ. In this paper, we describe one such circumstance, and discuss how we employed more advanced technology, with richer feedback, to enhance an active learning course.

The first author teaches a senior-level course in Computer Science at the University of Illinois, CS 421, Programming Languages and Compilers. The course is required for CS majors, and enrolls about 150 students. We wanted to employ an active learning approach, in which a significant portion of the class time would be devoted to student problem-solving. The class exercises should be of the same kind that the students would do for homework and on exams (including programming). Furthermore, we wanted the discussion of these exercises to be driven by the difficulties the students actually experienced. This raised another problem: Courses at this level --- perhaps especially in Computer Science --- are constantly evolving. We have no reliable history telling us what material will be most difficult for the students, or in what way. So, we wanted to assigned realistic, long-form exercises in the class, and *see* what the students were doing. These were not multiple choice questions; clickers would not solve our problem.

The traditional solution is to assign the exercises, and then walk around the class during the few minutes the students were engaged in them, looking over the students' shoulders. However, this is simply impractical in a large lecture hall, for many reasons. It is impossible to reach students who are not sitting on the aisles without disrupting other students; time constraints preclude reaching any students at the back of the class. The students' natural writing posture makes it difficult to see what they are writing without making them stop their work. More fundamentally, students are often shy about showing their work; having a professor look over their shoulder seriously impedes their ability to do the exercise. And besides, this would lead to a mode of interaction in which the professor offers help to one or maybe two students, rather than obtaining an overall sense of the classroom; this is fine in a small classroom, where every student will eventually get that kind of attention, but in a large class, the professor needs to focus on the class as a whole.

We solved our problem by developing a system that runs on wirelessly-networked, pen-enabled devices, on which some subset of the students solve the exercises (while the rest work on paper). The instructor can view the work of that subset (currently four students, changing constantly during the class), and display it to the class. We developed this system, which runs on Windows-based Tablet PCs [2] and Android devices [3], in our SLICE development framework [4,5], which allows us to develop applications for both platforms at once. We have used the system in every class in the current semester.

The benefits of this approach are:

(1) The exercises are not confined to multiple choice, or even short answer. They are, as we intended, simple versions of the problems students will confront in their homework.

(2) The students answer the questions in the most natural way, by writing out the solutions by hand.[1]

(3) Students have a sense of participating because, when they are using the tablets --- and every students gets the opportunity, because the tablets change hands after every exercise --- their work is often shown to the entire class.

(4) Because the professor is viewing the students' work remotely and anonymously, the intimidation factor is reduced.

(5) As part of showing a student solution to the class, the professor may write additional details or correct the student work. Starting a discussion with the students, based on the work of a peer, reinforces the sense of participation.

(6) Above all, the original goal is achieved: the professor can see exactly what the students are doing and what difficulties they are encountering, by watching them solve actual problems in real time.

In this paper, we describe the class and the computer system we developed in detail, and relate our experiences in the classroom.

2. THE COURSE

CS 421, Programming Languages and Compilers, is a required course in our CS curriculum, normally taken by students in their senior year. It is taught in twice-weekly, 75-minute lectures, in a large lecture hall; enrollment is approximately 150 students. The course material is generally considered quite challenging. Homework is given weekly and usually involves programming.

The goal of each lecture is mainly to teach the students the new skills needed to complete that weeks' homework. Accordingly, the in-class exercises are versions of the same kinds of problems as in the homework, often building up from much simpler versions of

[1] We have heard it said that students no longer write by hand, but instead type; and we have even heard that this especially true of Computer Science students. In our experience, this claim, which is made mainly by non-CS faculty, is completely divorced from reality, as one could see by visiting any CS classroom. Anecdotally, the first author has long banned laptops from his classes; out of hundreds of students, only two have ever claimed that this disrupts their note-taking

those problems, which can be solved completely in the class, to fuller versions that can only be solved partially. We hope to leave students in a good position to complete the homework. For example, the second assignment of the semester is to write functions over linked lists using recursion, in the programming language OCaml [6]. The in-class exercises that week built up from simple *non*-recursive programs on lists, to complicated recursion. Later in the semester, we talk about ``compilation schemes" – rules for generating machine language code for various high-level language constructs – and the exercises involve writing these rules; we start with simple examples that involve little more than copying examples given by the instructor, to creating rules from scratch.

The exercises are included within the class lecture slides. The slides containing exercises are printed before class and handed out to all students. (The lecture slides themselves are posted online before class, but very few students print those ahead of time.) Examples of exercises, and student answers, are given in section 4.

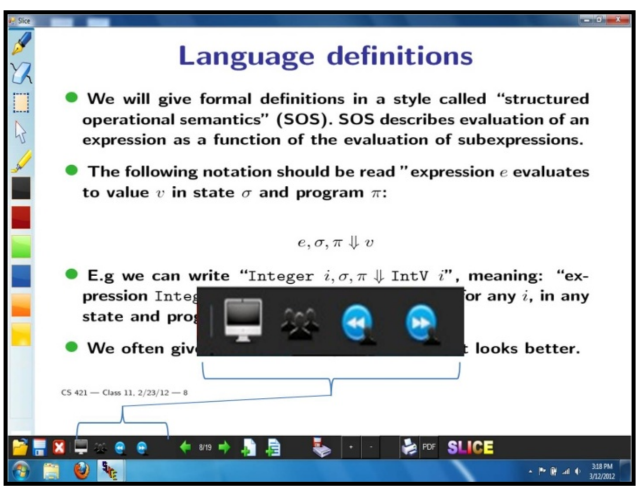


Figure 1: Lecturer's View

3. A MONITORING APP

To allow the instructor to monitor students' work on the in-class exercises, we used our SLICE [4,5] framework to build an application for pen-enabled computers. The application runs on Tablet PCs and on Android devices; in SLICE, the application code is exactly the same for both kinds of devices. The professor has a Tablet PC on which the lecture is given, and can view the work of students on other Tablet PCs and Android tablets.

At the start of each class, we distribute a set of computers (four, at present) to students randomly.[2] All the professor's notes, including the exercise slides, are transmitted to those machines wirelessly. We ask the students who received the tablets to solve the first exercise on them, and then pass them along to someone else. Classes usually have at least half a dozen exercises, so a significant portion of the class will use the tablets at some time during the class.

Figure 1 shows the instructor's version of the app. It is an ordinary Tablet PC-based presentation app, with buttons to change pen colors, erase pen strokes, move to the next slide, and so on. Most interesting is the set of buttons near the bottom left, highlighted in the figure. The ``heads" button switches the instructor's machine to monitoring mode, displaying the first student's tablet (at the same page as the professor's tablet, that is, the current exercise page). The arrow buttons next to it move from student to students. The display icon on the left displays, on the room display, the page of whatever student the professor is currently viewing; this allows the instructor to either show a good solution or point out an error that he suspects is common. The student's version of the app is similar, but with fewer buttons. It lacks the monitoring buttons, of course, but also lacks the erase button; students can only erase by crossing out. This was intentional, so that the instructor could see the students' false starts.

---

[2] We chose the number four for the most mundane of reasons: it is the number we can comfortably carry from our lab to the class. The app can handle any number of student machines, although there is a real question of how many response the professor can realistically monitor during the exercise periods.

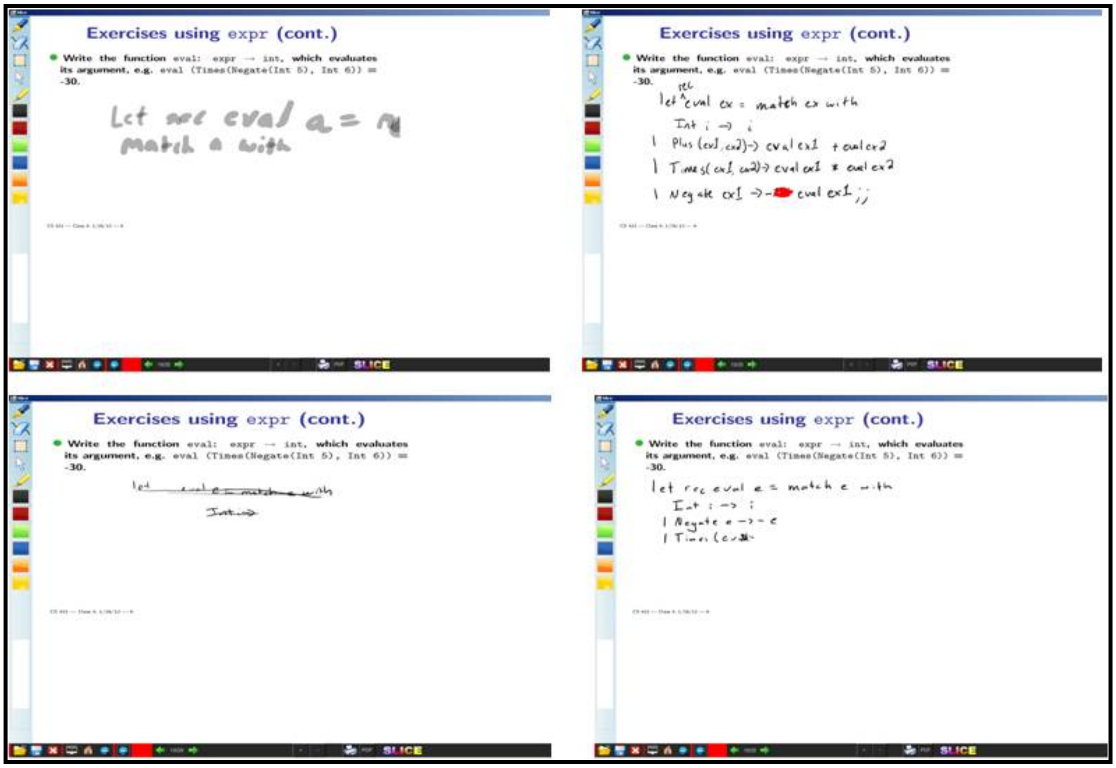


Figure 2: Four students doing the same exercise, viewed by the professor.

4. EXPERIENCE

The application is effective in giving the lecturer a window on the students' thought processes. The students seem to enjoy using it, and will sometimes draw pictures, or write their answers in a rainbow of colors. (The machines are very simple and natural to use; we have had no need to give the students any training on how to use them.)

Sometimes, the best information comes from the *lack* of student responses to an exercise; if none of the four students is working on an exercise after, say, one minute of thinking, that strongly suggests that those students --- and most likely, *almost all* the students --- are confused by the problem. But one of the benefits of getting this feedback constantly is that the professor learns over time how to pace the class, so that he gives fewer and fewer exercises like that. Or, to put it differently, he learns how to be so clear and concrete that students always understand what he asking of them. Nearly all the time, the students respond, although at very different speeds.

More typical are responses like those shown in Figure 2, where most of the students are completing the questions. One student will get the first part of the problem, and the professor will use that as the starting point for a discussion, and so on. (These screenshots show the students' final responses, after they have benefited from the entire discussion.) In programming classes, it is worthwhile to point out even trivial mistakes, because these can cost a lot of time when they sit down to do their homework.

Another way the system supports active learning is that it helps the teacher *calibrate* the pace of the exercises and, by extension, of the course material itself. The best example comes from the very beginning of the course. Lecture 2 covers basic material that the students have seen before: writing recursive functions. Either because they hadn't done this in some time, or because they were using a new programming language, they struggled with these exercises.

Because the material is so fundamental to the course, the professor changed the schedule of the course, and continued working on these problems in class 3. Absent the monitoring system, it may have seemed that the students were confused or disengaged, but the system provided concrete, unignorable evidence that he needed to spend more time on this topic.

Having said that, we should add that, despite the concern voiced in [1], we did not, in the end, cover less material than in previous semesters. We are certain of this because we can compare the exams. What we did was to jettison some “filler” material, leaving only material that represents concrete skills that can be tested. We view this, in itself, as a positive contribution of the monitoring system.

5. CONCLUSIONS

The essential thing in employing active learning is that the activities in which students engage should be *challenging* but *doable*. Careful calibration of the exercises is achievable if either the topic is well-studied and the way that students learn it are well understood, or if there is a tight feedback loop between students and instructor in the classroom. Neither of these conditions obtain in large college classes.

We have presented a system in which an instructor can closely monitor the work of a subset of the students in a class, and thereby create a facsimile of this tight feedback loop. This system uses wirelessly-connected, pen-enabled computers that are distributed to students randomly at the start of class, and move around during the class. The instructor, employing a networked tablet computer, can switch from lecturing mode to monitoring mode at will. When the students might benefit from seeing the work of another student, the instructor can display it on the room display. The system was developed using the SLICE framework, which greatly simplified development of applications like this one.

In the end, perhaps the greatest benefit of the system is that the professor gets a very clear picture of what the students are getting from the lecture. It forces the professor to focus less on *what he or she is teaching* and more on *what the students are learning*. And that is good for everyone.

REFERENCES

[1] Charles Bonwell and James Eison. Active Learning: Creating Excitement in the Classroom (J-B ASHE Higher Education Report Series (AEHE)). Jossey-Bass. 1991.

[2] Microsoft Tablet PCs, Wikipedia entry: wikipedia.org/wiki/Microsoft_Tablet_PC

[3] Android, Wikipedia entry: wikipedia.org/wiki/Android_(operating_system)

[4] S. Kamin, M. Hines, C. Peiper and B. Capitanu. "A System for Developing Tablet PC Applications for Education." in *Proceedings of the 39th SIGCSE technical symposium on Computer science education*. 2008.

[5] Wade Fagen, Sam Kamin. “A Cross-Platform Framework for Educational Application Development on Phones, Tablets, and Tablet PCs.” in *Proceedings of the 2012 International Conference on Frontiers in Education: Computer Science & Computer Engineering (FECS 2012)*. Las Vegas, Navada, July 2012.

[6] OCaml website: caml.inria.fr/ocaml/

Ethical Concerns in the Virtual World

Suhair H. Amer
Department of Computer Science
Southeast Missouri State University
Cape Girardeau, MO

Abstract— Advancement in Virtual Reality (VR) had made it possible to provide cost effective, easy and reliable solutions to healthcare, government agencies, classroom and personal needs. However, ethical concerns have become part of the VR controversy. This paper will discuss the uses of VR and the physical, psychological and social risks associated with the use of VR. Finally, summary of perceptions of college students on VR will be mentioned.

Keywords— **Ethics, Virtual reality, social implications of virtual reality, VR and physical, VR and psychological, VR and social.**

1. Introduction

A. Definition

"Virtualization is defined as the process by which a human viewer interprets a patterned sensory impression to represent an extended object in an environment other than that in which is physically exists" [5]. There are three main elements of virtual reality that are manipulation, navigation, and immersion. Manipulation is being able to touch and move objects in the virtual world. Navigation allows a person to walk around and explore the virtual world. Immersion uses head mounted displays and audio to immerse the user.

Virtual Reality (VR) can be defined as "...a way for humans to visualize, manipulate, and interact with computers and extremely complex data" [14]. This interaction can be achieved by integrating real-time computer graphics and a variety of sensory input devices. Head-mounted displays, tracking systems, earphones, gesture-sensing gloves, interaction/navigational devices and sometimes haptic-feedback, data gloves, joysticks, 3D mice, treadmills devices allows the user to believe that he/she is experiencing and sensing real situations. In addition, other methods may incorporate projection walls and rooms, as well as basic flat screen computer systems. VR allows users a total immersion and interaction into a three dimensional computer generated world; however, it can still include a level of computer generated worlds that are not necessarily totally immersive. For example, one can simply engage in a game such as Second Life while sitting in front of his/ her computer. In such as case, VR allows us to interact with other people, by means of digital representation (avatars), or computer controlled environments and simulated people (bots). The area of interaction resembles a world like environment, which may contain parks, skies, homes, shops, road ways, and sometimes places as complex as cities. VR can be used as training aids by the military and medical professions. They can also be used to reconstruct crimes by the Law enforcement agencies. Engineers can use them to bring their designs to life. Virtual reality simulation is, also, used as a tool for easing the pain of burn victims, teaching future doctors, and curing people from phobias and anxiety. In general, VR technologies are extremely expensive which is, currently, limiting their access to commercial users.

B. Origin of Virtual Reality and where it is Heading

The first VR technology was introduced by Ivan Sutherland of Stanford University when he experimented with computer graphics and wrote the software program called SketchPad which was used to bring engineering designs to life. One of the first forms of virtual reality date back to 1910 with the early flight simulators that was used to train new pilots in a quick, cheap and safe manner. By using controlled movements these simulators would roll and move, even though it did not resemble the exact airplane's movements, it gave the new pilots an idea of what it would be like in the air in a safer environment. In addition, military projects include flight simulators, tank operation simulators, and even combat simulators that are designed to help train a soldier for combat without the need to use actual vehicles and placed in dangerous scenarios [15].

Logging into a virtual world or their avatar in Second Life is becoming a common activity nowadays. More households are using game consoles utilizing virtual environments. More companies are becoming web integrated, and are beginning to advertise in the virtual world. It is expected that different agencies and ordinary people will get more engaged in activities that involve virtual capabilities [13].

2. Healthcare

Healthcare training can be expensive and time consuming; however, it is important to make sure that surgeons are adequately trained to perform well in the real world operating room setting. VR simulation training can be an effective and easy to perform activity for surgical residents that can aid them to perform different surgeries. For example, VR was examined as a solution for surgeons practicing in minimally invasive surgery (MIS) [2] which is difficult to teach and hard to integrate into a surgical training program. Using VR has shown to be highly successful especially.

In addition to using VR as a way for long term training, it can also be used for short term memorization. Medical residents can engage in a real time and realistic environment as if they were performing an actual surgery procedure. The environment can be interactive and guide them with selecting what steps to complete first and correcting their mistakes in a real time fashion [3]. In addition, VR simulations can teach medics to be able to handle the intense amount of stress by using relaxation training [12].

Treating anxiety and specific phobias can be done successfully by VR exposure therapy [7]. Anxiety and phobias negatively affect those who suffer from them by making it difficult for them to function normally in everyday life. The simulation adjusts to the patient’s feelings and emotions. Research has shown that VR has a great potential for treating specific phobias, especially acrophobia (the fear of driving), social phobia, panic disorder, claustrophobia (the fear of having no escape), aviophobia (the fear of flying), post-traumatic stress disorder, and arachnophobia (the fear of spiders). For example, a patient with Acrophobia (fear of heights) can undergo treatments such as entering a virtual environment of heights. Some patients, after completing several successful sessions, reported having substantially lower levels of anxiety in flying, and using elevators [13]. VR had shown tremendous success with patients who suffer from burns and excruciating pain during burn wound debridement [9]. By projecting calming images, such as a peaceful winter environment, patients reported considerably less pain when VR was used because it provides a distraction for the patient’s mind.

3. Government Agencies

The Centre for Disease Control and Prevention is experimenting with the use of VR technology for training emergency responders who face a variety of physically, emotionally, and mentally challenging situations that they may not be ready for. Immersive VR platform were designed to expose trainees to the sights, sounds, and smells that they may encounter as emergency responders. Simulations may incorporate smells of human waste, death, and decomposition and sounds of indigenous wildlife, native languages, coughing, and crying. These devices can help to ease the impact of post traumatic stress disorders for emergency responders [4].

The military uses VR simulation to train pilots for combat missions. Recently they have begun to use VR immersive simulation to prepare Humvee drivers for roadside bombs. The simulation helps the trainee to experience actual combat and teaches safety response to military personal. Therapists are using virtual simulation of war zones to help aid soldiers dealing with Post-traumatic stress disorder [4].

Law enforcement agencies are able to use VR platform programs to help re-create crime scenes to aid with their investigations. Crime scenes can be re-constructed using information such as photographs, measurements, and crime scene data. Some programs are powerful enough to reveal dust on surfaces, blood spatter patterns, and body position and injury projections [4].

4. Classrooms

Although, in the classroom setting, VR experience may not be fully immersive, there is a lot to be gained from VR platform software for computer systems. For example, Economical learning systems can be used with college students to help give the students hands on approach to see how buying and selling affects the economy. Such Games and applications help keep students attention and aid them in visually seeing the results and impacts in a lifelike manner. VR can be used to better educate individuals about other locations in the world or how something works by allowing them to gain first hand experience by immersing them into a situation or scenario that allows them to interact with a another person or a machine [13].

5. Games

The research on VR and gaming has increased rapidly, especially in enhancing interactivity and immersiveness which is very useful in order to perform virtual reality gaming [11]. Games offer entertainment such as Adventure, Competing Tasks, Specific Target, and other game elements. Serious gaming which consists of tasks that include specific knowledge can be attached to game modules. Additional information can be presented using messages, tutorials and experience like in real life [6]. The impression of interactivity and immersiveness in virtual reality gaming can be achieved by using good visualization and animation of characters. In addition, researchers have analysed human emotion through haptic devices [1].

6. Risks Associated with the use of Virtual Reality

A. Physical

Physical risks are present with the use of VR applications. The majority of these risks are minor side effects such as nausea and eye strain. For example flight simulations may leave the user feeling disoriented when walking. Similar to cell phones warnings, VR electronics can emit small amounts of radiation which contributes to biological breakdown and can lead to cancers. Flickering of the visual screen can cause eye strain but it can even cause seizures [10].

B. Psychological

Psychological impact can be profound because creating a Utopia type world may allow the user to become immersed and become dangerous. For example, a user who has a psychological disorder may become confused about reality and the virtual world. Addiction to VR could directly cause other forms of psychological disorder such as becoming aggressive as a result of playing aggressive games [10].

C. Social

Many people are connecting socially online, buying online, and creating communities online. Many worry that this will create a disconnect among people. To start with, some argues that VR gives the ability to manipulate a person's perception, their self control, and trust in a virtual level which raises the question of whether it is acceptable for this to continue. For example, some behaviour is not regarded as acceptable in the real world so why should it be acceptable in the virtual world. Some researchers suggest that the ethical guidelines should be re-designed to fit the VR spectrum, and bring into account individuals rights and respects [10].

Several crimes have been reported that occurred in the VR realm. For example, a Japanese lady "killed" her online "husband" she met while playing Maplestory. A 16year old boy in Tokyo defrauded people out of $360,000 in an online game. In addition, cyber crimes such as online theft and identity theft have increased because of virtual reality [8].

One of the ethical concerns addressed when using VR to train surgeons, questions how technically advanced is the equipment used for the simulation.

7. Perception of College Students on Virtual Reality

When interviewing students regarding their perception of VR, their opinion is mainly affected by their knowledge about the subject. Their initial reaction was related, for example, to the movies that describe the use of VR. Few had any knowledge regarding the uses of VR outside the movie industry or gaming. When students were asked to discuss ethical issues related to the use of VR, they started to develop more systematic conclusions.

One student indicated that with regard to VR and healthcare, doctors should be the ones consulted. Another student indicated that it is "extremely smart to spend more money to advance VR" in the "scientific forefront" and those individuals who have concerns regarding technology and the use of VR should get past their concerns as VR can advance healthcare.

Others believe that similar to the non-virtual world, morally good people will act the same in both non-virtual and virtual world and immoral people will act the same in both worlds. One student is "not confident that the human race is responsible enough to have the power of that type of technology with customized virtual worlds at their disposal." Some students would like to have this technology used by the military, emergency, medical training and educational.

Students agree that virtual reality continues to grow and create opportunities that were never thought possible. Some argue that virtual reality should be monitored by professionals but without full control from the government. This group argues that if users are being ethical and not breaking any laws, there is no need for total government control. One student indicated that there is a need for professional monitoring. Users should enjoy the benefits of virtual reality, but be careful regarding the information they provide to others.

Students understand that virtual societies are much like their real world counterparts, and how some actions that are acceptable in one may not be in the other. One student indicated that the decisions made in the virtual world may affect real life and world's influences may affect the virtual as well.

8. Conclusions

Advancement in VR had made it possible to provide cost effective, easy and reliable solutions to different agencies. It allows surgeons to get adequately trained to perform well in the real world operating room setting in a cost and time effective manner. It facilitates training emergency responders who face a variety of physically, emotionally, and mentally challenging situations. It provides educational games and environment to classroom students. It engages people in virtually based games and worlds. As the use of VR technologies evolve and increase, physical, psychological and social issues and risks have become part of the VR controversy. Physical risks may include nausea, eye strain, feeling disoriented and cancer. Psychological risks are highly associated with users that are already having psychological disorders; or it may increase aggressiveness. Finally, many are worried that being involved in cyber worlds may disconnect people who will loose the feeling of being actually connected to other people.

References

[1] A. Basori, D. Daman, A. Bade, M. Sunar, "The Potential of Human Haptic Emotion as a Technique for Virtual Human Characters Movement to Augment Interactivity in Virtual Reality Game", The International Journal of Virtual Reality,2008.pp.27-32

[2] A. Gallagher, E. Ritter, H. Champion, G. Higgins, M. Fried, G. Moses, C. Smith, and R. Satava. "Virtual Reality Simulation for the Operating Room". Annuals of Surgery. 2005 February; 241 (2): 364-372.

[3] B. Cowan, H. Sabri, B. Kapralos, M. Porte, D. Backstein, S. Cristancho and A. Dubrowski. "A serious game for total knee arthroplasty procedure, education and training." Journal of Cyber Therapy and Rehabilitation 3.3 (2010): 285.

[4] B. Pimentel. "SGI tries its hand at forensics / Italian police using computer graphics for crime scene analysis ". San Francisco Chronicle. 2005.

[5] D. Harrison and M. Jaques. Experiments in Virtual Reality. Oxford: Butterworth-Heinemann, 1996. Print.

[6] D. Michael, S. Chen, “Serious Game, game that educate, train, and Inform “, Thomson Johnson technology, Thomson Course Technology PTR, Boston, USA, 2006.

[7] D. Parson and A. Rizzo. “Affective outcomes of virtual reality exposure therapy for anxiety and specific phobias”. Journal of Behavior Therapy and Experimental Psychiatry. 24 October 2006; 39 (2008): 250–261.

[8] D. Rushkoff. "Internet is easy prey for governments." CNN Website. 2 February 2011. Web. 9 February 2011.

[9] H. Hoffman, D. Patterson, E. Seibel, M. Soltani, L. Jewett-Leahy and S. Sharar. “Virtual Reality Pain Control During Burn Wound Debridement in the Hydrotank”. Clinical Journal of Pain. 2008 May; 24 (4): 299-304.

[10] K. Behr, A. Nosper, C. Klimmt, and T. Hartmann. "Some Practical Considerations of Ethical Issues in VR Research." Presence: Teleoperators & Virtual Environments (2005): 668-676.

[11] M. Rossou, “Learning by Doing and Learning through Play: An Exploration of interactivity in Virtual Environments for Children”, ACM Journal of Computers in Entertainment, Vol. 2, No.1, 2004,pp.1-23.

[12] M. Stetz, C. Long, B. Wiederhold, and D. Turner "Combat scenarios and relaxation training to harden medics against stress." Journal of CyberTherapy and Rehabilitation 1.3 (2008): 239+. Academic OneFile. Web. 26 Mar. 2011.

[13] P. Anderson, B. Rothbaum, and L. Hodges. "Virtual Reality: Using the virtual world to improve quality of life in the real world." Menninger Winter Psychiatry Conference. Park City Utah, 2000.

[14] S. Aukstakalnis, and D. Blatner. Silicon mirage: The art and science of virtual reality. Berkeley, CA: Peachpit Press. 1992.

[15] S. Tate. Virtual Reality. 1996. 18 3 2011 <http://ei.cs.vt.edu/~history/Tate.VR.html>.

INTERNATIONAL STUDENT ATTITUDES CONCERNING ACADEMIC DISHONESTY SITUATIONS

A. James Frost, PhD [1], B. Alan Hamlin, PhD[2]
[1] Informatics Research Institute, Idaho State University, Pocatello, Idaho, USA
[2]Management & Marketing, Southern Utah University, Cedar City, Utah, USA

Abstract - *Incidents of academic dishonesty continue to affect every college and university in the nation, at undergraduate and graduate levels. This is also true at colleges and universities around the world. At some point during their academic careers, estimates are that 50-70% of all college students engage in cheating, plagiarism and other forms of dishonesty. The need for action to minimize this problem is evident, especially given the need of employers for highly-skilled and ethical workers in a global economy, and the recent spate of business scandals related to ethical misconduct in many nations. This paper describes the perceptions of business students from 20 different nations on 5 continents regarding what specifically they think constitutes academic dishonesty, and what they perceive should be done when infractions occur. The results could provide guidance to college professors and administrators as they evaluate incidents of dishonesty involving students from different cultures and backgrounds.*

Keywords: Academic integrity, international, cheating, student attitudes, instructor actions

1 Introduction

Frederick Douglass [1](April 1885) viewed integrity highly and stated, "The life of the nation is secure only while the nation is honest, truthful, and virtuous." The authors of this article embrace this concept and have extensive, long-term experience as both college professors and management consultants. Over the past several years, they have collected information from business students attending both domestic and foreign colleges and universities on their attitudes toward academic dishonesty and what they do when infractions occur [2](Frost, Hamlin & Barczyk, 2007). This paper provides a review of the literature about existing student attitudes towards academic integrity, and an analysis of a survey given to over 200 students in 20 nations about their perceptions of acceptable and unacceptable behavior in an academic setting. These perceptions are based on different scenarios given to the students on the survey instrument, and also provides input regarding whether the students themselves have engaged in unethical behavior. It is hoped that, with the results of this paper, faculty and administrators who are involved in adjudicating cases of academic dishonesty will be provided helpful information regarding cultural differences which might impact their decision about how best to discipline those who break the rules.

This paper is organized into four parts. The first describes why the problem of academic dishonesty is important, examining the extent of the problem and describing approaches to control it. The second is a review of the literature, covering current research and findings about how colleges are dealing with the problem in a multi-national setting. The third is an analysis of our primary research and the tool used to conduct it. The last section provides concluding remarks based on the research, and assesses the implications for further study in the field.

2 WHY THIS PROBLEM IS IMPORTANT

While the root cause of academic dishonesty is subject to much debate, anecdotal evidence suggests multiple factors, including media influence, lack of family training, peer pressure, and changing societal norms. Many undergraduate students in colleges and universities either engage in dishonest behavior themselves; refuse to turn in fellow students who they see cheating; think it is permissible to cheat if the rewards are high enough; or have some other type of unhealthy or unrealistic attitude. These attitudes result in more dishonest behavior, which in the long run, hurts the cheater and honest students that do not engage in dishonest acts [3](ibid).

When considered in tandem with the public perception of increased corporate dishonesty (which has evolved over the past decade as a result of lax ethical practices) and employers' requirements for educated business graduates with a thorough grounding in integrity, the need has never been greater for a solution to the problem of academic dishonesty. Six points highlight the urgency of this issue. First, academic dishonesty occurs frequently in every discipline, as discussed in the next section. Second, there is often no uniform method for dealing with the problem even within the same department, much less between different universities in different countries. Further, administrators are often more concerned with increasing enrollment than with reducing unethical behavior. Thus, individual faculty members can be left to fend for themselves, and most instructors, regardless of tenure status, do not wish to increase their workload by becoming "enforcement officers" in the classroom. Third, non-tenured faculty members have even less incentive to deal with this problem, since student retaliation on end-of-semester evaluations can interfere with the instructor's goal to attain tenure. Fourth, discrepancies in either policy or implementation can result in legal problems. Fifth, honest students are disadvantaged when dishonest students are not charged and punished, especially if the instructor grades on a curve. Sixth, how the issue is handled is of paramount importance in getting a positive outcome from this very negative experience. Academic instructors must foster the perception for the benefit of faculty and students, that integrity policies and enforcement mechanisms are fair and consistently applied. Even if these points are addressed, dishonesty will remain a problem for colleges and universities. The scope of the issue is so massive that the authors strongly believe that it is their responsibility to at least make an effort to minimize it [4](Hamlin & Powell, 2008).

Most research projects and studies of academic dishonesty in the past compare student behavior and/or attitudes from two or three different countries. This report seeks to expand the scope of the comparison, by using the same survey instrument to compare student attitudes in many nations about the same academic scenarios. Given the fact that many college classes today contain students from many different nations, such information might help faculty and administrators in their efforts to both communicate expectations, and handle with empathy and fairness any infractions in the classroom.

2.1 REVIEW OF THE LITERATURE

In the U.S., academic dishonesty permeates all levels of the educational system. A study by Bushway and Nash [5] (1977) reported that American students cheat as early as the first grade. Similar studies show that 56% of middle school students and 70% of high school students have cheated in the course of their studies [6](Decoo, 2002). The first scholarly studies of academic dishonesty at the college level were conducted in the 1960s [7](Bowers, 1964). This researcher found that in US colleges and universities, 50-70% of students had cheated at least once. In a major study in 1990, rates of cheating remained stable, but differed between institutions, depending on their size, selectivity, and anti-cheating policies[8] (LaBeff, et al., 1990). Generally, smaller and more selective schools had less cheating. Small, elite liberal arts colleges had cheating rates of 15-20%, while large public universities had rates as high as 75% [9](LaBeff, 1990). Klein and others [10](2007) surveyed 268 professional students and found that the business students did not report cheating more than the other students. However they were more lenient in their attitude toward cheating.

In Europe, the Middle East, Asia, and Africa, dishonesty is also prevalent at all levels. Not only that, but perceptions about what actually constitutes dishonesty also differ. In one study, significant differences were found between American and Polish students regarding attitudes, perceptions and tendencies toward academic dishonesty [11](Lupton, et.al., 2011). Donald McCabe, a very well-known authority on academic dishonesty in the U.S., did a study comparing student attitudes and norms from the Middle East (specifically Lebanon) to those of Americans. His results support the view that Lebanese university students are strongly influenced by the norms of the collectivist society in which they were raised, and therefore differ in their attitudes about what constitutes academic dishonesty from their American counterparts, who were raised in a more individualistic society [12](McCabe, et. al., 2008).

The impact of culture on a student's perception of what constitutes dishonesty is illustrated in a paper that appeared in the College Student Journal in 1998. This research compared cheating trends of American versus Japanese students, and also what determinants, techniques and deterrents contributed to these trends [13](Miller, et. al, 1998). Another study by Hajime Yasukawa analyzed how cross-cultural differences affected both the quantity of cheating, and the attitudes about whether cheating was actually dishonest. He compared U.S. and Japanese students over time, and found that Japanese students reported a higher incidence rate of cheating on exams, a greater tendency to justify the cheating, and also greater passivity in their reactions to observing other students who cheat [14](Yasukawa, et.al., 1999).

In Russia, there is a heavy focus on group assignments in education from a young age. This "muddies the water" about what is or is not permissible when students are expected to do their own work. One study of college business students in the U.S. and Russia found major differences in attitudes, perceptions and tendencies towards academic behavior and dishonesty [15](Lupton and Chapman, 2002). Similarly, research about cheating patterns between college students in India and the U.S. showed not only that the impact of growing up in a collectivist culture affects perceptions about what constitutes dishonesty, but even illustrates the differences between the sexes of such perceptions. For example, the data revealed that Indian and U.S. men were more likely to cheat than women of both cultures [16](Taylor-Bianco, 2007).

It is also important to note the motivators for cheating. Simkin and McLeod [17](2010) noted six cheating motivators in business students. First, they noted the issue of new opportunities that did not exist twenty years ago. The ability to quickly copy materials verbatim from the internet is very tempting to time-strapped students. This is often coupled with a "winning is everything' attitude in some cultures that can justify doing anything that will give one a competitive advantage. There is also the issue linked to the previous motivator that the reward for excellence may exceed the punishment if caught breaking an academic integrity rule. In fact, these are sometimes only guidelines and these are open for personal interpretation. There is also a major concern for the faculty member's career and/or the classroom environment when noting an issue of academic integrity. Some schools foster an environment that accepts issues in academic integrity and any faculty member that takes a student to task on integrity issues may find their career sidetracked. Not many schools include vigilance in academic integrity in their promotion and tenure guideline. This links to the attitude in some schools to redefine what is cheating. Again, there is often an opportunity to apply personal interpretation. Finally it is interesting that students may operate under a different "moral code" and not view cheating as a serious problem at all, especially in collectivist countries when the culture embeds in its citizens an attitude that "one cannot let one's countryman fail."

2.2 ANALYSIS OF PRIMARY RESEARCH

2.2.1 DATA ANALYSIS

Forty three international undergraduate business students participated in a study concerning academic integrity. The survey was conducted in hard copy with the students circling their correct response. The data was then entered into an Excel worksheet for summary and analysis. Pivot tables were constructed to compare responses for each question posed. Data was also collected to address nationality and gender issues when a larger sample is aggregated. The students were posed fourteen questions in this survey involving mild to serious issues of academic integrity. After the academic integrity question was posed it was followed up with concerning the student's personal attitude to the situation and a question on what should the instructor do in response to the situation.

The international composition of the students is shown in Table One with a 63% representation from the female gender. We did not analyze individual countries due to the limited sample size from each country.

Table One: Student Survey Composition

Country	Female	Male	Total
China	4	1	5
Columbia	1	4	5
DR Congo		1	1
France	3	1	4
Germany	1	1	2
Indonesia	2	1	3
Italy	1	1	2
Japan	2	1	3
Lithuania	3		3
Mexico	1		1
Portugal		1	1
Slovakia		1	1
South Korea	2	1	3
Spain	4		4
Sweden	1		1
Turkey	1		1
USA	1	2	3
Grand Total	27	16	43

The first question dealt with copying another student's homework. 58% of the students had copied another students homework NO more than a couple of times. There was not a difference between the genders in resisting the temptation to copy another student's homework. The first sub-question dealt with the student's personal viewpoint on the "acceptability" of this action. Three of the students (all male) viewed it as acceptable to copy another student's homework. 72% of the students viewed copying homework as rarely acceptable or unacceptable. The second sub-question on the topic of copying another student's homework had to do with the instructor's response. Nearly one-fifth of the students (19%) felt the instructor should ignore the activity. At the other end of the spectrum, two students (both female) felt the student should be expelled from school for copying homework.

The second question asked if the student being surveyed ever provided homework to another student. Forty-two percent of the students reported that they had shared homework many times (more than six-ten times). Even though 42% of the students have shared homework many times, 47% of the students view this as rarely okay or unacceptable. This may be surprising since 28% of the students reported that they had shared homework less than twice. There appears to be different viewpoints on what they have done in the past and what they now view as acceptable. Further, in the final sub-question, 30% of the students feel the instructor should ignore the issue of sharing homework.

The third question looked at students that would collaborate with another student on an individual assignment. This was an assignment to be completed individually; however the student received assistance from another student. Nearly half of the students reported that they participated in this dishonest activity rarely (either never or a couple of times). Nine percent reported participating in this action more than six times (probably frequently). Although the majority did not participate in this action frequently, 70% of those surveyed thought it was acceptable behavior or okay occasionally. 50% of the students reported that the instructor should ignore the action and another 35% felt that the instructor could lower the grade on the assignment and require a redo of the assignment.

The following questions dealt with more "serious" violations of academic integrity. Instead of looking at homework issues that constitute a small portion of the student's grade, an exam usually represents much more of the performance and measure for the class. When asked if they worked with another student on a take-home exam instead of working alone, half said never and another 28% reported only once or twice. The students appear serious about their take-home exams. As we often see from personal experience, half of the group viewed this as a student activity that should not be conducted or participated in for gain. However, 18% thought this was an acceptable behavior. So, we see at least one in five that do not view collaboration on a take-home exam as a problem. This is paralleled in the response as 19% had done this action three to many times. The question did identify that this exam was to be completed individually. Three-fourths of the class indicated the instructor should ignore the activity or require a re-take of the exam beginning with a lower grade. This promotes an attitude of “worth the risk” for the students considering academic dishonesty.

When asked if the student had used a cheat sheet hidden in an ink pen, or on my body, etc., during an exam, nearly 70% responded that they did this less than twice (42% never). However, 30% used cheat sheets more than three times (5% many times). Seventy seven percent of those responding viewed this as unacceptable behavior. The students viewed this action as a serious infraction as 35% selected an F for the assignment and another 33% felt the student should be failed in the class for using a cheat sheet. Again though, 7% reported no action should be taken.

The sixth question explored the international student’s exposure to using technology to gain an advantage by storing exam answers electronically via a cell phone or similar device. Eighty four percent had never used technology like this; however 15% received exam questions via a cell phone. Ten percent of the students viewed this as acceptable behavior while 77% viewed it as unacceptable. Also, ten percent felt nothing should be done to the offender while 40% felt an F on the assignment was appropriate and another 35% selected an F in the class.

The seventh question of the set dealt with the students collaborating with another individual to receive answers on the exam. Nearly two-thirds reported they had never done this action. However, seven percent reported they had done this action more than six times. Three fourths of the students viewed this as an unacceptable behavior however seven percent viewed it as acceptable. Two thirds of the students felt the student should be given an F in the class or for the assignment (equally split on the consequences).

When asked if the student had ever reviewed an actual copy of the exam before test time 63% indicated they had not, however the remainder had looked at an exam at least once. Two students reported seeing exams prior to test time over six times in their academic history. Interestingly, three students viewed this as acceptable behavior; however two of these students did not report they had participated previously. So, perhaps, some had access and participated however they did not believe it to be appropriate behavior while others would have participated if they would have had a chance as they did not view it as anything they should not do. Nearly half of the students indicated that the students should be failed in the class. Ten percent indicated expulsion from school was appropriate and fourteen percent reported nothing should be done.

The ninth question asked if the student had ever provided test questions to another student before that student took the exam. Although forty percent said never, the remainder had provided test questions to other students, 25% reported six different occasions or more. A little more than forty percent viewed this activity as unacceptable, however one third thought it was okay to do occasionally. Only one of the students thought the student should be expelled while one third thought it should be ignored.

The following question had the students addressing if they had ever written a mnemonic on an area behind the instructor to recall answers. Two of three students indicated they had never used hidden mnemonics to assist there work during an exam. However, one in four had used the technique many times. Also, half of the students viewed this as an unacceptable action; however 15% viewed it as acceptable behavior. The two predominate responses were to ignore the action or fail the student on the assignment.

When queried about the issue of copying articles from the internet without any citation, forty percent indicated they had never done this action and another 38% selected only once or twice. However, ten percent indicated they had done this many times. Further, fifteen percent viewed this as acceptable behavior although 42 percent viewed it as unacceptable. Two thirds of the students felt an appropriate action was to lower the grade on the assignment or award an F in the class. Fifteen percent felt it should be ignored and two felt expulsion from school was appropriate.

The next question dealt with the student obtaining a term/research paper from a paper mill on the internet and turning it in for a class assignment. Interestingly, there were two answers to this question with this group. Eighty percent had never done this while the remainder did it once or twice. Sixty five percent viewed it as unacceptable behavior and another fifteen percent thought it was rarely acceptable. Seven percent felt it was an acceptable action. Based on this response, we believe even if they had only done one or two research papers, they regarded it as unacceptable behavior. Slightly over forty percent indicated they felt an F for the assignment was appropriate while twenty five percent opted for retaking the assignment beginning at a lower maximum grade. Only one student selected expulsion from school.

College and high school students often turn to youtube.com services for assistance in many activities or a giggle as humorous stunts are acted out. The viral actions can provide instructions dealing with how to tie a tie to a proper technique to disassemble a firearm. We posed the question on students using youtube.com to discover new techniques for cheating. Only one student admitted to seeing cheating techniques on youtube.com. The linked question on using these techniques for assistance on exams or reports was equal with almost all of the students indicating they didn't use the techniques from youtube, probably because they didn't look at them. There is an interesting video done by a young lady showing her cheating techniques by hiding notes in her skirt. When asked about the implications of what she was showing, she didn't see a problem with it. One student did write on the survey, "Thanks for the hints" with a smiley face.

2.3 REVIEW

2.3.1 Academic Dishonesty

We see a small fraction of the students involved in activities that would be considered some form of academic dishonesty. Our previous research confirms this in both of our schools as well. Generally few students participating in this survey were involved in issues of dishonesty and the size of this group decreased as the severity of the offense increased. Still, there is always someone that affects the integrity of the group by cheating. The following list identifies the thirteen question areas posed to the students:

1=copy homework
2=Provide homework
3=Worked with another student
4=Collaborate on take-home exam
5=Used a cheat sheet
6=Stored answers electronically
7=Collaborated during exam
8=Received copy of exam
9=Provided exam answers
10=Received answers electronically
11=Wrote mnemonic behind instructor
12=Copied material with no citation
13=Used paper from paper mill (19% used a paper mill 1 - 2 times)

There are two extremes of participation in academic dishonesty and several items are noteworthy. The second question had a high number of students providing their homework to other students. Also, in questions one, three, nine and twelve, more students participated in a specific action than in other questions (greater than ten percent). In questions two and nine, the student was the expert providing expertise to other students, perhaps in a sharing effort of community. It is difficult to not extend the hand of assistance when called on by someone that is struggling. However, the blame is equally shared when acts of academic dishonesty are conducted.

Also, question six indicated very few students stored exam answers in electronic devices (cell phones, calculators, etc.). Finally, question 13 could be interpreted in several fashions. First, 79% of the students have never participated and 19% have turned in a research paper from an internet source only once or twice while one individual has done it many times. We are also concerned about the one in five students that purchased a research paper from the internet once or twice. How many research papers are they required to submit in their undergraduate career? Is this the sum total of all research papers they have done in their undergraduate career? This definitely is a cause for concern as we review the results of our survey.

Another point is that in almost all of the questions, there was someone heavily involved in the action of academic dishonesty. It is interesting that only the question on using an electronic device to store answers received 96% of the students reporting they had never done that action. Perhaps calculators are not as universally available to international students in business classes to store text messages with answers during exams.

2.3.2 Student's View on Action

The questions on the student's personal view on issues of academic dishonesty displayed a variance from their actions on the same issues. A large percentage indicated it was acceptable behavior to provide homework, collaborate on take-home exams, and provide exam answers. Each of these responses approached the 50% of those surveyed answering acceptable or acceptable occasionally. Interestingly, more than ten percent of the students indicated it was acceptable to receive answers via electronic devices (14%), write a mnemonic assistance behind the instructor (14%) or copy material from the internet without citation (16%).

2.3.3 Instructor Actions

The final subset of the survey involved the student's view of what action was appropriate for the instructor to invoke. They ranged from do nothing to expulsion from school. The greatest forgiveness from the students was when the action was working with another student on homework and collaborating on a take-home exam. This was followed by one-third of the students suggesting light punishment for providing exam answers and writing mnemonics behind the instructor. Half of the students selected a harsh punishment (fail the class or expulsion from school) when a student receives a copy of the exam.

3 Conclusions

Our initial research suggests that instructors in university and college classrooms should be aware that international students are also susceptible to the lure of academic dishonesty. Most international students resist the temptation to gain advantage during exams or when assigned research papers. However, a small percentage (at least one in ten) will use various techniques to gain an advantage on other students. Also, as a counterpoint, they indicate they like to assist others, even if they potentially could suffer the consequences.

We feel it is important to level the playing field in academia and not provide an advantage to a student attempting to use auxiliary methods to score higher in a class. It is not an issue of being a cop and policing the exam and homework. To fairly judge the performance of each student, a small set of students should not be rewarded for their deceit. This group of international students as a whole did not participate in rampant academic dishonesty. However, it is a small set that does not see a problem with specific actions or indicate they would take advantage given certain conditions.

Our data gathering efforts are continuing as the sample set is now nearly tripled in size with this summer's data collection in France by one of the authors. We are encouraged by these initial indications and intend to begin cross-gender and cross-culture analysis with larger sample sets. Also, we plan to conduct a longitudinal study if we are fortunate enough to have the ability to collect international data annually.

4 References

Number in square brackets ("[ ]") should cite references to the literature in the main text. List the cited references in numerical order at the very end of your paper (under the heading `References'). Start each referenced paper on a new line (by its number in square brackets).

[1] Cambridge, MA: MIT Press, p. 23. Douglas, Frederick, Speech on the twenty-third anniversary of Emancipation in the District of Columbia, Washington, D.C. (April 1885), found on http://en.wikiquote.org/wiki/Frederick_Douglass, March 8, 2012.

[2] Frost, J., Hamlin, A., & Barczyk, C. (2007). A Survey of Professor Reactions to Scenarios of Academic Dishonesty in American Universities. The Journal of Business Inquiry: Research, Education, and Application, 6(1), p. 11-18.

[3] Ibid

[4] Frost, J., Hamlin, A., & Barczyk, C. (2007). A Survey of Professor Reactions to Scenarios of Academic Dishonesty in American Universities. The Journal of Business Inquiry: Research, Education, and Application, 6(1), p. 11-18.

[5] Bushway, Ann & Nash, William R, (1977). School Cheating Behavior. Review of Educational Research, 47(4), p. 623.

[6] Decoo, Wilfried (2002). Crisis on Campus: Confronting Academic Misconduct. Cambridge, MA: MIT Press, p. 23.

[7] Bowers, William J. (1964). Student Dishonesty and its Control in Colleges. New York: Bureau of Applied Social Research, Columbia University, p. 155.

[8] LaBeff, Emily E., et. al., (1990). Situational Ethics and College Student Cheating. Sociological Inquiry, 60(2), p. 192.

[9] Ibid

[10] Klein, Helen A., Levenburg, Nancy M., McKendall, Marie, Mothersell, William, Cheating During the College Years: How do Business School Students Compare? Journal of Business Ethics (2007) 72:197–206.

[11] Lupton, Robert A., and Kenneth J. Chaqman. Russian and American college students' attitudes, perceptions and tendencies towards cheating. Educational Research 44.1 (2002): 17-27.

[12] McCabe, Donald, Tony Feghali, and Hanin Abdallah. Academic Dishonesty in the Middle East: Individual and Contextual Factors. Research in Higher Education 49.5 (2008): 451-467.

[13] Miller, Richard L., et al. Academic dishonesty: A Delineation of cross-cultural patterns. College Student Journal 32.4 (1998): 590. Miller, 1998.

[14] Miller, Richard L., et al. Academic dishonesty: A Delineation of cross-cultural patterns. College Student Journal 32.4 (1998): 590. Miller, 1998.

[15] Lupton, Robert A., and Kenneth J. Chaqman. Russian and American college students' attitudes, perceptions and tendencies towards cheating. Educational Research 44.1 (2002): 17-27.

[16] Taylor-Bianco, Amy, and Dawn Deeter-Schmelz. An Exploration of Gender and Cultural Differences in MBA Students' Cheating Behavior: Implications for the Classroom. Journal of Teaching in International Business 18.4 (2007): 81-99.

[17] Simkin, Mark G., McLeod, Alexander. Why Do College Students Cheat? Journal of Business Ethics (2010) 94:441–453, Springer.

Ethical Perceptions of College Students on Emerging and Converging Technologies

Suhair H. Amer
Department of Computer Science
Southeast Missouri State University
Cape Girardeau, MO

Abstract—this paper will discuss several emerging and converging technologies and college students perceive some of the ethical issues related to the emerging and converging technologies such as biometrics technology, nanotechnology, nanocomputing, expert systems, pervasive computing, and intelligent user interfaces.

Keywords-emerging technology , converging technology, ethical concerns

1. Introduction

Emerging and converging technologies are terms used to cover various cutting-edge developments in the emergence and convergence of technology. Emerging technologies describe significant development in technologies in a new territory in their field. Converging technologies cover areas where different disciplines are merging towards a common direction. In this paper several emerging and converging technologies will be briefly discussed and perception of college students of the ethical concerns related to such technologies will be mentioned.

2. Biometric Technology

A. Definition and Ethical Concerns

Biometrics compromises methods that can uniquely recognize humans based on one or more physical or behavioral traits [1]. In Biometric technology, first, the user presents his/her physical trait. The characteristic is then captured and submitted for preprocessing. A template is created for the extracted information and then stored. In addition to fingerprint identification, other biometric identifications include facial or retina patterns, voice recognition, hand prints, signature identification and writing patterns. Although, biometric identification proves to be important security tool, there are issues related to storage of such data [2]. Other ethical concerns are related to privacy and how the government will or may choose to make biometrics part of everyday life. The public implementation of such technologies poses the issue of invasiveness and raises issues regarding rights to personal privacy. This is because individuals would like to make sure that their stored personal information is secure [3].

B. Student Perception

Students perceive Biometrics as an ever growing market, and that it will continue to grow in the future. Some believe that recognition processors will become faster, and will be able to recognize persons from father distances. They indicate that biometrics can be a useful technology if implemented correctly and not as a way to keep tabs on every person. The main purpose of the technology is to ensure privacy and automate the identification process. It can be useful in keeping databases of criminals but should not be used to record how frequent will a regular person visit a store. They are concerned that society may become too dependent on computer security and lose trust in manual security. They are, also, concerned because technology is not entirely reliable. Finally, after students have learnt about the different techniques and issues especially identity theft, they feel more comfortable with the existence of such techniques.

3. Nanotechnology and Nanocomputing

A. Definition and Ethical Concerns

The United States' National Nanotechnology Initiative website defines Nanotechnology as "the understanding and control of matter at dimensions of roughly 1 to 100 nanometers, where unique phenomena enable novel applications [4].

Nanotechnology can give "god-like" powers [5] and can eliminate other issues such as assembling beef instead of slaughtering cows. It is important to develop nanotechnology using guidelines to insure that the technology does not become too potentially harmful. As with any new technology, it is impossible to control development; however, ethical related guidelines have been developed with regard to Molecular Nanotechnology [5].

Other ethical concerns are related to privacy and control, longevity and runaway nanobots. With regard to privacy, small devices can be easily inserted into one's clothing without them being aware of it. Although longevity may appear like a good idea, some are concerned that it will result in lack of new ideas and new blood. With regard to nanobots and the possibility creating what any one wants, they can operate incorrectly and become uncontrolled [6].

B. Student Perception

Students found it difficult to either support or not to support the technology. They believe that it is important for the government to invest into such technology because it is very dangerous to allow other agencies take control or to advance in such field. They believe that this new technology can really help our world, but if used wrongly it could really hurt us at the same time.

4. Ubiquitous Communication

A. Definition and Ethical Concerns

Ubiquitous communication is everywhere at the same time which was made easier with the use of mobile phones and wireless networks. “Ubiquitous computing enhances computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user.” One important feature about ubiquitous communication is that users should be able to use ubiquitous computers without being conscious of doing so. It allows communication between interlinked communication devices that can be stationed in various locations [7].

One of the major ethical concerns relate to privacy which is a major issue regarding universal form of contact where information may be intercepted while flowing from one communication device to another [8].

B. Student Perception

Students argue that ubiquitous computing and communication is the new trend in wireless communication, networking, and computing. Most people use this invisible form of communication nearly every day. They consider it an essential part of their lives. They think that any ethical issues that may arise with regard to this technology can be dealt with as it occurs.

5. Expert Systems

A. Definition and Ethical Concerns

There are many uses for expert systems. First, an appropriate domain is identified and then an expert's problem-solving expertise is captured, and his/her knowledge and problem-solving rules are represented and then validated. An easy-to-use interface then allows users to use the system for a problem-solving task at hand. Expert systems enable experts to perform problem-solving activities more quickly and consistently. In general, both the human expert and the expert systems share similar concepts, training, and experience with domain matters. Logic programming presents us with an excellent tool to develop a variety of intelligent systems. Systems such as these can span a wide range of types of systems, ranging from simple tools to assist the decision-maker to fully automated, autonomous reasoning systems [9].

B. Student Perception

The students indicated with regard to expert system technologies, that the development process is very important as without this process the probability of software succeeding is very unlikely. They also indicated that training people on the new software is a very important to get them familiar with the new technology and to address any issues regarding their use.

6. Pervasive Computing

A. Definition and Ethical Concerns

Pervasive computing, also known as ubiquitous computing, is the growing trend towards embedding microprocessors in everyday objects so they can communicate information [10]. Pervasive computing devices are completely and constantly connected. Pervasive computing relies on the convergence of wireless technologies, advanced electronics and the Internet. All models of pervasive computing share a vision of small, inexpensive , robust networked processing devices, distributed at all scales throughout everyday life and generally turned to distinctly common-place ends [11].

Wireless technology and especially handheld devices have achieved significant market penetration. Today, wireless communication is relied upon extensively in the workplace, home, classrooms, and even for entertainment [12].

There are both social and ethical issues related to the intelligent aspects of computing. Ambient intelligence, for example, is a technology that enables people to live and work in environments that respond in an intelligent way. One concern regarding the use of such technology includes worries about the loss of freedom and autonomy, and whether humans will become overly dependent on technology [6].

B. Student Perception

Students argue that despite ethical and social issues that may be related to the use of pervasive technologies they are a worldwide phenomenon that continues to grow. The students indicated that with the tremendous enhancement capabilities, we must realize that predators may use this technology. However, we should not prevent the advancement of such technology. They help them find things that can be posted on social networks, or through the use of their phones.

7. Intellegent User Interfaces

A. Definition and Ethical Concerns

Many of today’s modern conveniences rely on the use of intelligent user interfaces. Intelligent user interfaces are basically machines or devices that engage in computationally calculated intelligent actions that are used by humans. The focus of such interfaces is to simulate human actions and thinking [13].

Intelligent user interfaces mimic human behavior and may be considered harmful. For example, with regard to communication, words can be taken so differently through text than through word of mouth which could be a problem. This requires the computer to be predictable and also be able to tell the difference in dialog. Another issue is concerned with storing vast amounts of data that is necessary to produce better results [14].

B. Student Perception

Students are excited about the technology that are related to artificial intelligence especially with regard to correcting spelling mistakes while texting. They are also fascinated with videogames and animation. They indicated that advances in such field will affect their regular lives, work and education. Some indicated that instead of the user adapting to an interface, an interface can adapt to the user. The intelligent user interface tries to determine the needs of an individual user and attempts to maximize the efficiency of the communication.

Some indicated that intelligent user interfaces have proven themselves to be quite beneficial. They have already been used to assist humans. Despite their limitations, they have been quite advantageous for those in need of aid. Research is starting to be done on expanding the social relationship between robots and humans.

Some of their ethical concerns were related to the question whether these interfaces will become more social in an effort to expand their intelligence or will they continue to be focused on completing tasks. Few were concerned about how some may use such interfaces in their evil plans. They are worried about machines that can be programmed and used by different type of individuals.

8. Conclusion

In this paper, college student perceptions of ethical issues related to converging and emerging technologies were mentioned. Almost all students indicated that their perception changed as they got more acquainted with the technology and especially when they got to learn about the ethical issues related to the use of such technology. They were mainly concerned with technologies that may affect their security and privacy. They were in support to technologies that are related to their every day's activities such as using cell phones.

References

[1] http://www.biometrics.gov/documents/biointro.pdf,accessed [2/25/2011]

[2] Alterman, Anton. "``A Piece of Yourself": Ethical Issues in Biometric Identification." *Ethics and Information Technology* 5.3 (2006): 139-50. Print.

[3] Maguire, M. (2009). The birth of biometric security. *Anthropology Today*, 25(2), 9-14. doi:10.1111/j.1467-8322.2009.00654.x http://library.semo.edu:2343/ehost/detail?hid=104&sid=1aca5d9c-5276-4455-95a5-069f239df4b1%40sessionmgr113&vid=5&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#db=aph&AN=37208823

[4] "The History of Nanotechnology." nanogloss June 11th, 2009 : n. pag. Web. 26 Mar 2011. <http://nanogloss.com/nanotechnology/the-history-of-nanotechnology/>.

[5] "The Ethics of Nanotechnology." actionbioscience 2000-2011: n. pag. Web. <http://www.actionbioscience.org/newfrontiers/chen.html>.

[6] Tavani, Herman. Ethics and Technology, Controversies, Questions, and Strategies for Ethical Computing. 3rd. Hoboken, NJ: John Wiley & Sons Inc, 2011. 375-385.

[7] "UbiCC Journal - Volume 4 Number 3 - Ubiquitous Computing and Communication Journal." *Scribd*. Ed. Thanos Vasilakos. 15 Aug. 2009. Web. 11 Feb. 2011. <http://www.scribd.com/doc/24121966/UbiCC-Journal-Volume-4-Number-3-Ubiquitous-Computing-and-Communication-Journal>.

[8] Reed, G M and Sanders, J W. "An Ethical Principle for Ubiquitous Communication." May 2007. Web. 25 Mar. 2011. <http://www.iist.unu.edu/www/docs/techreports/reports/report373. pdf>.

[9] Jones, James D. A different paradigm for expert systems: an introduction to logic programming and related knowledge representation issues. Expert Systems; Nov2006, Vol. 23 Issue 5, p342-355

[10] www.Pervasivecomputing. Net

[11] "Integrating Technology and Digital Media in the Classroom". Shelly, Cashman, Gunter and Gunter.

[12] "Pervasive Computing". Henn, Hepper, Rindtorff, and Schack.

[13] Dautenhahn, Kerstin. (2007). "Socially Intelligent Robots: Dimensions of Human-Robot Interaction." Philosophical Transactions: Biological Sciences, 362, 1480, 679-704.

[14] Wærn, Annika. "What Is an Intelligent Interface?" *SICS*. Web. 09 Feb. 2011. <http://www.sics.se/~annika/papers/intint.html>.

Data Warehouse Implementation of Examination Databases

Er. Majid Zaman[1], Dr. S. M. K. Quadri[2], Er. Muheet Ahmed Butt[3]
Directorate of Information Technology & Support Systems, University of Kashmir, Srinagar, J&K, India
Head & Director, PG Department of Computer Science, University of Kashmir, Srinagar, Srinagar, J&K, India
Directorate of Information Technology & Support Systems, University of Kashmir, Srinagar, J&K, India

Abstract: *A data warehouse is a relational database that is designed for query and analysis rather than for transaction processing. It usually contains historical data derived from transaction data, but it can include data from other sources. In addition to a relational database, a data warehouse environment includes an extraction, transportation, transformation, and loading (ETL) solution, an online analytical processing (OLAP) engine, client analysis tools, and other applications that manage the process of gathering data and delivering it to business users [2][10]. This was the case at University of Kashmir (UOK) Examination Department, where such a project brought together various attributes of Examination System which included Conduct, Secrecy, Transit, Tabulation, Accounts and other related data sources in to an integrated data warehouse.*

Keywords: Data Warehouse, OLAP, ETL, Examinations.

1.0 Introduction

To remain competitive in today's academic climate, any University needs a foundation of quality data that too in examination system where the data need to be more precise and consistent. Organizations of higher education need this capability as much as Fortune 500 companies. Organizations, both governmental and business, have to manage large amount of information stored in some form of databases or files [1][9]. One of the main problems to deal with information managing is the weak interoperability between various databases and information systems. Especially this problem is serious when we want to organize collaboration between the information systems of various departments within the organization. Solution is data-warehousing, the idea of drawing data from several different (heterogeneous) data sources on different platforms and computers. Building a data warehouse is extremely complex and takes commitment from both the information technology department and the business analysts of the organization. It takes planning hard work, dedication, and time to create a relational database that delivers the right data to the right user. University of Kashmir's Examinations data warehouse (UOKEDW) is not a panacea for every data problem, but it is a very good start toward a permanent solution.

2.0 Towards Computerization

University of Kashmir Examination Department declares 450 evaluation and 450 re-evaluation results of around 3.5 Lakh enrolled students appearing in both under and Post Graduate Examinations of around 200 affiliated colleges. University of Kashmir, like any academic organization in India started realizing need and importance of computerization in late 1990 and early 2002, however it was not those days that we were provided with RAD tools. In house computerization was done in FoxBASE, dbase and SCO Unix as operating system, three teams were set up initially to computerize following systems,

- Tabulation
- Transit
- Secrecy
- Certificates
- Accounts
- And Conduct

As year passed on, information explosion within the department was at its peak, it was not much long before university administration realized that in order to manage such a crucial data of the Examination Department and provide friendly access to information, new teams needs to be setup with enhanced budget be provided in order to store/manage data to meet the changing demands.

Newly constituted teams were given decent budget along with freedom to choose the tools required for development, within no time university was riding on with the success of Information Technology and almost all the areas of the university administration were computerized-n autonomous solutions within single organization each working fine but with no possibility of integration.

3.0 UOK Examinations Data Warehouse Development

Development of UOK Examination data warehouse started in the in around 2004 as a client/server project. The tools which were mainly used in the first phase of development were Microsoft SQL Server 2000 and Windows Server 2005 workstation. Every software development carried out on these platforms was done in home. While getting the warehouse server in place, the software development team provided data access software for efficient usage of the warehouse in implementation of both read and write operations [3]. Although many of the access tools were in their adolescence at the time, accessing data was much easier with these graphical user interface (GUI) tools than with the fourth-generation tools then in use.

University of Kashmir formed a development team of two Project leaders with ten Master of Computer Applications final year students and Computing Assistants from the data administration and Examination Automation Centre to build the data warehouse. The team selected a representative group of business analysts to serve as pilot users to test the warehouse and access software. During the next few months, the team built a “student” warehouse model based on over 150 questions, which the pilot users considered difficult or critical to answer using current information resources [10][11].

During 2006, many of the original data warehouse team members shifted back to their regular duties, leaving a core of six fulltime equivalent employees working on the project. That core has remained intact, receiving additional help from UOK’s institutional research office and many of the business analysts who are regular users of the warehouse. Also, the data administration department initiated a formal program to train users on the warehouse. To date, there are over 250 trained warehouse users which have been trained while carrying out training programs for various employees. The major goal is to train 400 odd employees, approximately 10 percent of UOK’s employees.

4.0 Examination Warehouse Architecture

UOK’s Examination data warehouse resides in a client/ server environment. UOKEDW extracts data from loaded on to the Unix Server Majorly kept for Secure Data Entry Process for marks entry and loads it into a Microsoft Windows server running an MS SQL as RDBMS. UOKEDW server is a IBM Xenon Server with 8 GB of memory and two processors, running the Windows 2008 Server operating system. Users connect through Ethernet to the warehouse over UOK Campus Area Network backbone via Transmission Control Protocol/Internet Protocol (TCP/IP). The suggested GUI data access has was first implemented in Visual Basic 6.0 now transformed to Visual Basic .NET 2008., which runs identically on the Windows. Microsoft Access® is another tool used mostly for data migration from one database Server to other. The process of using GUI tools to build structured query language (SQL) requests and bring the results back to a client machine. With client/server architecture, once the data are in the workstation, users “own” the data, cutting and pasting at will into their favorite software (e.g., spreadsheet, word processor, graphic tools) [4].

5.0 Examination Data Warehouse implementation Phases

The following steps were under taken for designing of UOKEDW

5.1 **Requirement Gathering:** The first thing that the constituted project team was engaged in gathering requirements from the various employees working in the examination system. Because end users were typically not familiar with the data warehousing process or concept, requirement gathering was implemented using one-to-one meetings or as Joint Application Development (JAD) sessions, where multiple stake holders in the examination wing were interacted with so that the requirement analysis done in a proper manner.

5.2 **Physical Environment Setup:** Once the requirements gathering were somewhat clear, it became necessary to set up the physical servers and databases. At a minimum, it was necessary to set up a development environment and a production environment. It was not enough to simply have different physical environments set up [5]. The different processes many data warehousing projects where there were three environments: Development, Testing, and Production (such as ETL, OLAP Cube, and reporting) also need to be set up properly for each environment. The primary goal of this phase was to identify what constitutes as a success for this particular phase of the data warehouse project.

5.3 **Data Modeling:** This was a very important step in the data warehousing project. Indeed, it was fair to say that the foundation of the data warehousing system is the data model [7]. A good data model will allow the data warehousing system to grow easily, as well as allowing for good performance. In UOKEDW project, the logical data model was built based on user requirements, and then it is translated into the physical data model.

5.4 **ETL** (Extraction, Transformation, Loading) process typically took the longest to develop the UOKE’s data warehouse implementation cycle as there were many heterogeneous data bases involved in extraction transformation and loading process. The reason for this was that it took time to get the source data, understand the necessary columns, understand the business rules, and understand the logical and physical data models before ETL would have been successfully carried out.

5.5 **OLAP Tube Design:** The OLAP cube was derived from the Requirement Gathering phase [6]. The users working in the Examination Wing had some idea on what they want, but it was difficult for them to specify the exact report / analysis they wanted to see and anlyse. When this was the case, it is usually a good idea to include enough information so that they feel like they have gained something through the data warehouse, but not so much that it stretched the data warehouse scope by a mile. Hence front end development became an important part of a data warehousing initiative of UOK.

5.6 Front End Options: The front-end options ranged from an internal front-end development using scripting languages such as VB, VB .NET, ASP, PHP, to off-the-shelf products such as Crystal Reports, to the more high-level products such as Actuate. When choosing vendor tools, it was made sure that it could be easily customized to suit the business of examination, especially the possible changes to the reporting requirements of the Examination System. Possible changes included not just the difference in report layout and report content, but also included possible changes in the back-end structure.

5.7 Report Specification: Report specification typically came directly from the requirements phase [8]. To the end user/employee working in the examination system, the only direct touch point he or she had with the data warehousing system is the reports they see and analyse. So, report development, although not as time consuming as some of the other steps such as ETL and data modeling, nevertheless play a very important role in determining the success of the data warehousing project.

5.8 Query Processing – In this the OLAP reports or reports were made to run directly against the RDBMS often exceeded the time limit, and it was hence ideal for the data warehousing team to invest some time to tune the query, especially the most popularly ones.

6.0 Security and privacy

Security and safeguarding privacy are major concerns when building a data warehouse and when it came to Examinations sensitive data the applicability of security and privacy was at prime focus. Security in a database means protecting data against unauthorized disclosure, alteration, or destruction. Granting select (authorization to read only) access to tables or views achieves a certain level of security in a warehouse [12]. At UOKEDW, read-only access to the data warehouse was provided at the database level. This procedure followed an open access policy for employees approved by UOKEDW administration in 2008. This policy was based on the notion that giving employees access to data and holding them accountable is better for the organization than withholding the data.

Although many RDBMSs support column level security [13][14], UOKEDW has not implemented this feature, primarily due to the high cost of administering user access. In traditional operating systems, tasks or screens control access, meaning users only have access to a single record or instance of data (e.g., verifying admission status of a student). In a data warehouse, users have access to a table or set of tables in a subject area, which means access goes beyond retrieving single records to retrieving groups of records.

At UOKEDW, the Assistant Registrar is the trustee of the student Registration database, Assistant Controller Secrecy is the trustee of Secrecy Database, Assistant Controller Conduct is the Trustee of Conduct Database, and so forth. In these databases the write/read access is given at various levels to the employees depending on the nature of work they have in examination system. To obtain name and address information, the data trustee grants access to the person database. The user's business need determines whether access is granted. Given the large number of records in UOKE's data warehouse, placing name and address in a separate database achieves a certain level of privacy.

7.0 Conclusion

The future of UOKE's data warehouse is becoming more clear. Initially, the warehouse served as a resource for accessing information from legacy systems. Eventually, the warehouse will serve as a telescope into UOKDW's distributed data stores. Some of these data will reside in the data warehouse, while other elements will be "viewed" from the RDBMSs where the data reside. UOKEDW foresees a time when the telescope extends beyond UOKEDW to other organizations with common goals, such as the neighboring Maricopa County Community College District. The real power of the warehouse will be actualized in years to come. The data warehouse fills an important data administration role in a client/server environment. As distributed application developers move further away from the central computing core, the data elements in the warehouse ensure the integrity of the organization's enterprise data.

The bottom line is that data warehousing is here to stay. Warehousing gives organizations the opportunity to "get their feet wet" in client/ server technology, distributed solutions, and RDBMS. This is essential for any future mission critical application, making the data warehouse a low-risk, high-return investment.

8.0 References

[1] Jorge Bernardino, Pedro Furtado, Henrique Madeira," A Cost Effective Approach for Very Large Data Warehouses", Proceedings of the International Database Engineering and Applications Symposium, 2002

[2] Kimball, Ralph, “The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses”, John Wiley & Sons, Inc, 1996.

[3] Krishna. “Principles of Curriculum Design and Revision: A Case Study in Implementing Computing Curricula CC2001”. ITiCSE’05, June 27–29, 2005

[4] Larry, Greenfield, LGI Systems Inc., "The Data Warehousing Information Center," 1997 pp http://pwp.starnetinc.com/larryg/index.html.

[5] M. A. R. Kortnik, D. L. Moody. “From Entities to Stars, Snowflakes, Clusters, Constellations and Galaxies: A Methodology for Data Warehouse Design.” 18th. International Conference on Conceptual Modelling. Industrial Track Proceedings. ER’99.

[6] W. J. Labio, Y. Zhuge, J. N. Wiener, H. Gupta, H. Garcia-Molina, J. Widom. Stanford University. “The WHIPS Prototype for Data Warehouse Creation and Maintenance”. SIGMOD 1997

[7] Marotta. “A transformations based approach for designing Data Warehouses Internal Report.” InCo. Universidad de la República, Montevideo, Uruguay. 1999.

[8] D. Theodoratos, T. Sellis (DWQ project). “Designing Data Warehouses.” DKE ‘99

[9] F. Bancilhon, N. Spyratos, “Update Semantics of Relational Views,” ACM TODS 6, 1981, 557–575.

[10] A. Bauer, H. G¨unzel, eds., “Data Warehouse Systeme — Architektur, Entwicklung, Anwendung, dpunkt.verlag”, 2001.

[11] J. Lechtenb¨orger, G. Vossen, “On the Computation of Relational View Complements,” Proc. 21st PODS 2002, 142–149.

[12] J. Lechtenb¨orger, G. Vossen, “Multidimensional Normal Forms for Data Warehouse Design,” 2002, to appear in Information Systems, Elsevier Science.

[13] W. Lehner, J. Albrecht, H. Wedekind, “Normal Forms for Multidimensional Databases,” Proc. 10th SSDBM 1998, 63–72.

[14] H. Lenz, A. Shoshani, “Summarizability in OLAP and statistical databases,” Proc. 9th SSDBM 1997, 132–143.

New Active Learning Tools

ATTORNEY MARCELLA COCCANARI
Coccanari & Partners Law Firm –
You Man Right –
City Council of Rome - Schools Commission
Roma, Italy
MR. ROCCO LANATA'
BKC Business Company –
Milano, Italy

Abstract
We leave in an technological age, but training and teaching do not uniformly guarantee high standards of results. The decline of education and widespread illiteracy creates serious consequences across society, but also has clear implications in terms of productivity.
The method that we are going to develop in a software, is not a mnemonic technique, a speed reading program or one of many "approaches" to the study: is a complete set of functional techniques.
This is an autoimproving software that helps in teaching and learning and "learns" from student results and from students and teachers' feedbacks.

Keywords
Teaching
Learning
Effectivness
Barriers
Interacting
Self-improvement

I. SITUATION

We leave in a technological age, but training and teaching do not uniformly guarantee high standards of results.

Even if more than 300 billions of dollars are spent in education, the capability to read, to write and to calculate of students in American schools is crashing.

More that 13% of American teenagers are not able to read a newspaper.

More than 44 millions of American citizens are not able to read the safety instruction on chemical tools.

Out of 4 soldier, one is not able to understand the safety instruction on firearms.

Overfilled classrooms, underachieving students and "learning disorders" are today all too common. Teachers struggle with classroom management problems while principals ponder the correct levels of discipline to control unruly students.

At the same time, adults worry about holding their jobs as technology outpaces the skills they worked so hard to learn.

Such problems have generated libraries full of educational theories. But the problems persist: students who drop out of school only to end up on the street or in gangs, the student who is so restless and disruptive in class that he is given drugs to "help him focus" or to at least keep him quiet, adults who struggle to hide their non-comprehension of essential written material such as prescription labels, or adults who last read a book in high school when their teacher forced them to do so and who now struggle to comprehend or properly apply the tasks before them on their job, and many others.

Teachers try to apply different methods of teaching, but they don't work uniformly with everybody,

Teachers cannot fully assist a single student in a classroom,

Students even if they pass their exams, later don't remember the given information and they are not able to apply them. Besides student's most frequent protest regarding school, is that it is boring, they cannot keep attention more than few hours.

In connection with many business consultants we have proof that in the workplaces the new workers are enable to show and applied what they were supposed to learn.

II. SOLUTION. THE TECHNOLOGY AND ITS RESULTS

The Study Technology developed by American author and educator L. Ron Hubbard.

Study Technology consists of tools and techniques teachers can use to improve the learning rates of students. These same tools and techniques can be used by students themselves to improve their ability to understand and to use the materials they read and study.

There had never before existed a true technology of study that can guarantee high standard of results.

The Study Technology is a breakthrough in self-paced, individualized instructional methodology and in education, that emphasizes, as the name states, a practical technology that is in fact a system of learning how to learn.

Study Technology undercuts the reasons people are illiterate or cannot apply materials they have studied. It is not just another system, it is a workable methodology that makes it possible for a person to recognize and handle the barriers to a successful study.

Purpose of this Study Technology: Converting traditional group lesson plans into personalized learning modules and creating effective and adaptable checksheets for each and every subject..

Several comparative studies and case studies have been conducted on Study Technology programs, documenting their substantial and measurable results. More than 28 million people have now been aided by Study Technology, across some 70 nations, with real emphasys on capability to apply the learned data.

We can demonstrate that teaching the same subject to two different classes, with Study Technology it can be faster and students are able to understand and apply!

Study Technology is not a collection of study tips or memory tricks. It is not speed-reading or the latest method of taking better notes. Rather, in its full range, it is a powerful educational philosophy with clear techniques of application. It provides a system of learning how to learn every and each subject.

III. THE THREE BARRIERS

A. Absence of mass (physical object) of what is being studied

If one is attempting to understand the function and operation of a car or a computer or a solar system, the printed page and spoken word are no substitute for the object itself. It would be difficult to understand how to use a computer for the first time if you did not have the computer there in front of you. In fact, lacking the object associated with a word can inhibit all understanding. If the mass of a subject is absent, you can actually feel squashed**.** It can make you feel bent, sort of spinny, sort of dead, or bored. A person studying a subject without the objects related to that subject will experience these and several other specific reactions. Knowing how to identify and handle these reactions is vital to a student's ability to grasp and use a subject—and more than vital to a teacher's ability to get a student to learn the subject.

B. Skipped gradient

A gradient is a way of learning or doing something step by step. A gradient can be easy where each step can be done easily, or it can be hard where each step is difficult to do.Too steep a gradient consists of not having mastered prior skills before going on to more complicated or detailed steps.A student who has skipped a gradient may feel a sort of confusion or a feeling of reeling (i.e., moving or swaying like you might fall).These are two reactions a person will have when they have missed a step or hit too steep a gradient in the subject they're studying.This is often referred to as "missed basic skills" or "insufficient basic skills."

C. A word not understood or Wrongly understood

The third and most important barrier is the misunderstood word. "Mis" means *not* or *wrongly*. "Misunderstood" means *not understood* or *wrongly understood*. A misunderstood word is a word which is *not understood* or a word which is *wrongly* understood. Have you ever been reading a book or a report, gotten to the end of the page and couldn't remember what you read? Therein lies the phenomena of the misunderstood word—all becomes distinctly blank beyond a word not understood or wrongly understood. It can make you feel blank or washed out. It can make you feel "not there" and a sort of a nervous upset feeling can follow after that. The matter is far more critical than one might surmise and of the three barriers it is the misunderstood that bears most upon human relations, the mind and understanding. It is the misunderstood word that establishes aptitude—or lack of it. It produces a vast panorama of reactions and is the prime factor involved with stupidity. It also determines whether or not one can actually perform a learned skill, and to what degree of proficiency. All of these are the result of one or more words or symbols not understood or wrongly understood. The misunderstood word can stop a student in his tracks completely. Knowing how to determine when there is a misunderstood word or symbol, how to find it and how to handle it are critical to the success of any student.

IV. APPLICATION TO BUSINESS

With Study Technology programs has been proved that employees become true professionals achieving new levels of skill at their jobs with remarkable positive effects on their efficiency and productivity.

When workers are not able to fully learn their jobs, the result is not just a bad grade. Yet in the corporate world, the retention rate for materials studied by workers is a constant challenge for those who are supposed to train them. The same holds true for job proficiency rates. And opportunities are increasingly rare for professional growth so workers remain current in their field or get retraining if necessary.

Some previous programs have been implemented at a major computer hardware firm: training time was reduced by as much as 70 percent—with recent graduates performing as well or better than veteran employees. As a result, the company is now using this system to introduce all new information system processes. At another computer firm, assembly productivity went up 28 percent after a specific training and quality improved 80 percent with quality levels remaining above 99.5 percent each month.

Study Technology programs has been delivered to employees and management personnel from Fortune 500 companies with widely varied products, medical centers and government agencies. When vehicle safety inspectors in California were trained using Study Technology, computer input errors were reduced from 20 percent to 0.5 percent.

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

V. THE NEW SOFTWARE

We are proposing the developing of a very specific software addressed to teachers and students, based on a method that is changing the entire vision of teaching and learning. The key word is: proven effectiveness.

In its first section, our software will guide teachers and educators, with a step by step approach, into the workout of effective educational programs on every subject and the students will have a practical tool to learn successfully.

The software at the beginning is programmed to show different subjects that is far away from his/her understanding. The student, following the method, learn something about the chosen subject, that he or she first did not like or wasn't able to understand. Going through the instructions he can realize that every subject can easily be learned, faster than ever.

Some test are available for each subject. Students need to proof their competence before entering into the final software section which allows them to learn the given subjects part of the school program.

The software can verify if the students is doing well or it can bring the person back on the correct gradient (in a particular case the teacher can intervene and assist the student on the method).

For the practical application, the core of the method, we studied different options that bring the student to study the theory and after proof the real understanding with specifics actions which demonstrates the effective of the learning path.

With the advanced section of the software, the user will be helped mostly in identification and resolution of symptoms coming from the three barriers to study.

Software includes: tools to keep alive in students the purpose and reason why for study. This cut the potential threat of "I already know about it".

Tools to completely grasp the meaning of words , the basics elements to build the understanding.

Tools to demonstrate one's understanding and ability to apply what has been studied.

Methods to work out a checksheet as a guideline for study and study's instructions.

The description of different ways a word can be misunderstood and consequent methods to arrive at a full conceptual understanding of a word.

The software performs periodical checks on: level of attention, interest, application, purpose and motivation, grade of withdrawal from study, physiological phenomena related to the three barriers to study, personal development and understanding of key words and concepts.

VI. CONCLUSION: FUTURE GENERATIONS AND EDUCATION

The real challenge for Man in the new millennium is not the construction of orbital space stations. The real challenge is to create a truly effective education through this innovative software to increase the productivity and bring out the potentialities and abilities. This challenge is definitely the biggest adventure, that can carry Man beyond the stars!

COPYRIGHT

ACKNOWLEDGEMENT

Applied Scholastics Italia & Mediterraneo
Applied Scholastics International
(www.appliedscholastics.org)
EURYDICE - Network on Education System and Policies in Europe (www.eurydice.org)

REFERENCES

[1]"Barriers to Study" based on the works of L. Ron Hubbard (Applied Scholastics International).

Teaching Ethics Across Disciplines

Suhair H. Amer
Department of Computer Science
Southeast Missouri State University
Cape Girardeau, MO

Abstract—This paper briefly describes the techniques and some of the attempts to teach ethics across different disciplines. Each discipline faces some challenges while attempting to incorporate ethical components in their curriculum. Across all disciplines, techniques such as group discussion, where case studies, professional and personal examples are used, is one of the common techniques used.

Keywords- teaching ethics, ethical concerns, ethical education.

I. Introduction

College students are expected to understand ethical principles related to their disciplines and act accordingly. It is important to engage students in ethics during their early college educational years to shape their behaviors and to continue to reinforce such information during their years of study. However, with the required coursework and time constraints, it is difficult, sometimes, to teach ethical issues to students in an effective way. Ethical components need to be incorporated in a course to assure that students appreciate the need for ethics and be able to analyze and debate ethical issues.

Several studies attempted to determine college students' perception of ethics. In [1], five factors were studied that were "(1) the impact of education and faculty/instructors on ethics; (2) students' attitude towards cheating; (3) the impact of technology; (4) the importance of ethics; and (5) the ethical campus environment." Their findings assured that it is important to incorporate ethics training in the classroom, enforce and carry out ethical standards and that instructors need to be ethical themselves before asking that from their students.

Before teaching and incorporating ethics in the class room it is important to understand the students' attitude towards ethical issues. For example, cheating in college is compared to students' perception of cheating in "high schools, colleges, and non-major classes" [1]. According to [1] students tend to cheat less in colleges and especially in non-major classes and they do not report cheaters and that cheating increases with the use of technology. With offering more blended and online courses, more students are using such technology to cheat which means that their overall learning decreases. However, the good news is that the study of [1] indicated that the majority of students consider ethics to be very important and that they not only hold others to these ethical standards but hold themselves accountable. Indeed incorporating ethical components in education is producing ethical aware students.

The following sections will briefly describe how ethics is incorporated in different disciplines

II. Computeing and Engineering

In computing disciplines it is essential to apply ethics to the computing curricula. For example, the criteria for Accreditation of Computer Science and Information Systems programs require that computer ethics be included in the computing curriculum to give programs credibility and enhance the decision making skills of students with regard to the use of computer technology. It is important to introduce ethics into the curriculum at various stages; however, when introduced early, students are able to benefit from learning computer ethics and respect topics such as copyright and intellectual property issues. In the computer science curriculum it is not easy to incorporate ethics modules. This is because computer scientists are used to working on problem-solving situations where answers are straight forward. It is also challenging to teach ethical issues especially those that do not have definite solutions [2]. Different techniques can be utilized to teach computer ethics such as content-based and application based models [3]. In general, students tend to solve or address ethical dilemmas by stating their opinion before being taught ethics which is not usually analysis-based. After being introduced to ethical modules, students increase their knowledge and are able to identify ethical concerns, reference similar examples and apply ethical theories to answer ethical related questions. In [4] a case-based analysis technique was used to teach ethics which uses a bottom-up methodology. Deciding on whether an action is right or wrong is determined by comparing the action to closely related cases using multitude of attributes to reach effective ethical decision-making.

Engineers are usually placed in the position of public trust and Engineering ranks high in being a trustworthy profession in public surveys. As a result, members of this profession must be "guided by externally imposed rules and also engage in significant self-regulation"[5]. Although, engineering ethics can be considered another example of professional ethics, it has some distinct characteristics [5].

It is important to mention that engineers may play a part in the decision-making process within organizations; however, most of the time, they don't have the final authority on technologies and they neither finance nor authorize. They are concerned with addressing "how" but not "ought" issues which are not usually addressed in the design process. In addition, Engineering raises several group related ethical issues since it involves teamwork and shared responsibility for decision-making. This is because technical knowledge is not, usually, associated with the hierarchy within the organization which may cause inappropriate assessment of risk [5].

A need to integrate ethics in the courses of Engineering degree has been identified by the documentation of the accrediting professional bodies, both in Australia and elsewhere. Students need to be aware of ethical concerns and to be to be introduced to "non-technical patterns of thought and so improve their capacity for interactions in the community where they serve" [5]. Teaching may involve traditional lectures and students' debates.

III. Medical

Medical ethics is one of the most important components of medical education. Medical students are required to demonstrate awareness of the main professional obligations of doctors. They also need to be familiar with the statutory requirements and codes of conduct for medical practice in the country they are working at. They should become aware of the doctor patient relationship and respect patients through telling them the truth and respect their privacy and maintain confidentiality. They should also be aware of how to apply ethical principles dealing with end-of-life decisions and how to manage options with regard to terminally ill patients. They also need to develop sensitivity to ethical issues while dealing with the pharmaceutical industry and other providers of medical technology. They should become familiar with dealing with HIV/AIDS patients. They will learn how to provide genetic services and ethical counseling in organ transplantation in an ethical manner and understand the ethical issues involved in transplantation and organ donation practices. Student gets familiar with the ethical issues in providing mental health services. In addition, they get familiar with the different ethical issues related to providing health care to children and in research involving children [6].

IV. Science

In this age, scientific experimentation and discovery is ethically debated by the society such as genetic information gathering and technology, animal dissection and experimentation and assessment of environmental threats. Although biologist may argue that "science is the search for objective truth, and therefore ethically neutral" [7], socially reflective scientists recognize that there are ethical responsibilities to those who make scientific discoveries. Educators need to be professionally responsible and instill a framework to making ethical decisions to their students as they will be responsible for making decisions about how society will control and/or support biology. For example, educating students about bioethics is challenging because it is interdisciplinary in scope and does not fit into a standard science curriculum. Sustainable environmental policy can be introduced to students of biology, chemistry, etc; however, graduate education in biology is focused and biologists are hesitant to speak outside their specific area of expertise and have limited training in ethics. Furthermore, the National Institutes of Health (NIH) requires education in science ethics for all graduate students that are supported by federal grants; however, it concentrates on the ethics related to conduct of research. In general, students are required to learn about ethical implications of their work before college and during their undergraduate degrees [7].

The ABCDE Method [7] is an example of a method of ethical decision making which allows an individual to reach a final decision on an ethical conflict. Students are asked to think about opposing arguments, costs and benefits, and to reach a final decision based on personal fairness. In the model, Argument (A) stage asks the student to list arguments for and against the ethical conflict. In the Both Sides (B) stage, students make sure that an argument has two or more sides having consequences to the side. In the Costs and Benefits (C) stage, the students use the information they have gathered to generate more statements describing the costs and benefits of each argument. In the Decision (D) stage, the student will use open discussion and debate to reach a conclusion or decision. In the Evaluate (E) stage, the student will decide if the process is fair given all the arguments, costs and benefits, and final decision. With this model, [7] had success in teaching ethics to undergraduate students.

Some colleges, teaching ethics in science to undergraduates, integrate ethics into a capstone experience, a safety course, discipline curriculum or the entire university curriculum [8]. Usually, undergraduate students majoring in science have little work experience in their field which requires that a professional ethics course describes professional context. Courses that discuss ethical issues, for science majors, should cover ethical principles, and students' experiences with ethics in education and laboratory data. It should then demonstrate the importance of accurate, unbiased data while teaching how research, development, and quality-control work is done, disseminated, and used. They should also get aware of professional relationships with collaborators, mentors, reviewers, those who they review, employers, and the public. Additional topics can be discussed as appropriate. In addition, professional ethics courses are approached differently from science courses. An ethics course requires examination and discussion of personal and professional values. It is also not supported with experimental data since usually values of the instructor and students may be different. Finally, although students may be affected by the course, they still don't develop enough assessment and responses to the different ethical challenges they may meet [8].

V. Environmental Science and Forestry

Undergraduate programs in natural resources education usually have only taught professional codes of ethics as an obligatory review in the first-year freshmen natural resource introductory course and junior/senior policy courses. Although the natural resource programs have begun to teach professional codes of ethics, faculty members find it difficult to decide how to teach these codes to students. Many teaching methods are utilized such as the case study approach which may use, for example, bottomland hardwood forests that addresses undergraduate forestry and wildlife students. As the eastern United States supports a tremendous hardwood resource, many forests are exploited by diameter-limit cutting harvesting under selective management. This had both short- and long-term negative impacts on the hardwood resource and the potential for landowners to sustain the critical values that hardwood forests can provide for future generations. This is why, it is important that university administrators and faculty instill in their students the importance of professional ethics early and at different stages of their student's academic life. It should be formally incorporated into every course of a professional nature in natural resources curricula and not only be a component of the course but something that students should continuously deal with. One technique can be to periodically review of applicable codes of ethics and should involve case studies and discussions of recent situations [9].

VI. Conclusion and Future Work

Literature has shown that different strategies are employed to incorporate and teach ethics across various disciplines. This paper describes the importance of teaching ethics across different disciplines. Different teaching techniques, information delivery strategies, ethical knowledge-based inquiries, and case-based modules can be used to teach ethics. It is relevant to incorporate teaching ethics early enough so that students may benefit from it. It also should be covered at different stages with different emphasis and direction. Examples targeting the specific discipline make it easy for the student to relate and understand the importance of applying ethics in different aspects of their personal and professional life. To teach ethics it is important to act ethically as well and to understand and tolerate the different backgrounds and beliefs of others.

References

[1] Linda K. Lau, Brandon Caracciolo, Stephanie Roddenberry, Abbie Scroggins. College students' perception of ethics. Journal of Academic and Business Ethics. Volume 5 - March 2012.

[2] Dudley-Sponaugle, and D. Lidtke, "Preparing to teach ethics in a computer science curriculum." Technology and. Society, 2002. (ISTAS"02)

[3] Marsha Zaidman. Students in Computer Ethics-Speak Up. Journal of Computing Sciences in Colleges, Volume 26 Issue 2. December 2010

[4] Michael Quinn. Case-based Analysis: A Practical Tool for Teaching Computer Ethics. SIGCSE 2006, March 1-5, 2006.

[5] Iain Skinner, Iain MacGill, Hugh Outhred. Some Lessons from a Decade of Teaching Ethics to Undergraduate Engineering Students.

[6] Facilitators' Guide for teaching medical ethics to undergraduate students in medical colleges in the South-East Asia Region. http://www.searo.who.int/en/showdetailsnew.asp?code=B4477 accessed Februayr 1012

[7] Carol K. Johansen David E. Harris. Teaching the Ethics of Biology. THE AMERICAN BIOLOGY TEACHER, VOLUME 62, NO. 5, MAY 2000

[8] Linda M. Sweeting. Ethics in Science for Undergraduate Students. Journal of Chemical Education. Vol. 76 No. 3 March 1999.

[9] Brian Roy Lockhart and Ralph D. Nyland. Teaching professional codes of ethics to forestry and wildlife students: A case study using diameterlimit harvesting in a bottomland hardwood stand. Natural Resources and Environmental Issues. Vol. 12 Issue 1. Article 27.

SESSION

SOFTWARE ENGINEERING AND SYSTEM DEVELOPMENT + PROGRAMMING/LANGUAGE ISSUES

Chair(s)

TBA

Characterization of CS1 Student Programming

N. Bosch, D. Towell, and J. Homer
School of IT and Computing, Abilene Christian University, Abilene, TX, USA

Abstract— *This paper reports on a quantitative evaluation of data collected from student programming assignments in an introductory programming course, using 44,548 submissions by 192 students over six semesters. Student programs are gathered and automatically scored for correctness; students are permitted to repeatedly submit possible solutions until a specific deadline, and each submission is immediately scored and feedback given about correctness or any errors. We report on the process followed by students (total time spent per problem, number of submissions, rate of completion,* etc.*) as well as the results (program length, complexity, state space). Our study reveals some common characteristics in "typical" CS1 student solutions and provides some reflection on the assignment framework.*

Keywords: Computer Science education, CS1, Learning patterns

1. Introduction

Computer science educators have a significant interest in improving the general understanding of how novice programmers acquire and improve upon basic computer programming skills. Introductory programming (CS1) courses apply varied approaches in the design and assessment of programming assignments, often differing in the number, size, submission policies, and degree of feedback given for assignments. In this paper, we present an evaluation of one programming assignment framework for a CS1 course and discuss the advantages and disadvantages of this approach.

This experience report describes the behavior of students in a CS1 course at Abilene Christian University. Utilizing metrics based on submitted source code, frequency of submission and other factors, we quantify and evaluate some student performance properties.

In our introductory programming class, the programming assignment framework requires students to submit their programs to an online system for automated scoring. The system uses several techniques to score the submission including direct inspection for required features (such as functions), preprocessing and other source modifications, compiling, linking with both standard and specialized libraries, and execution with controlled input. Any step in the scoring process which produces an error or unexpected result generates a report to the student with the details. Students receiving an error message are permitted to iteratively correct and resubmit their solutions until a correct solution is achieved or a predetermined deadline time is reached. The notable characteristics of this approach are the presentation of immediate feedback on either the success or cause of failure and the allowance for frequent and repeated iteration (bounded only by an end-time).

The intention of this programming assignment framework is to encourage students to complete each assignment, overcoming early errors and eventually reaching a final working solution. The CS1 course presents a large number (approximately 75) of assignments in which students are expected to write complete working C++ programs. In this way, students are expected to repeatedly practice and refine their programming skills, gradually increasing in experience and ability throughout the semester.

This paper reports on the analysis of the gathered data. We will address the success rate (eventual completion of programs by students), average number of attempts per problem, time delay between student submissions for the same problem, and the quality of programs produced by students in this approach. Related works are discussed in Section 2. Our data gathering and evaluation is described in Section 3; the results of our analysis are presented in Section 4 and some interpretation is given in Section 5. The paper concludes in Section 6 with a summary of our evaluation of this data and some thoughts about future study.

2. Related Works

Many educators share a common interest in understanding how students learns and what can be done to improve their learning and retention. The study of novice programmers (notably how they think about and create computer programs) has attracted much attention [1], [2], [3], [4]. In recent years, efforts have been made to automatically record and evaluate student-created solutions for programming assignments, and to analyze the behavior exhibited and stages of progress encountered by student programmers [5], [6], [7], [8], [9], [10], [11].

Jadud [7] studied the behavior of students in an introductory course. By logging compilations of student work, he was able to identify the error types most commonly encountered by students and analyze their responses when presented with error messages. He found that many (51%) student compilations were performed less than 30 seconds after the previous attempt; our data corroborates this rapid-response trend in student behavior and we present some analysis of the success rate for such behavior.

Various toolsets have been developed to automatically log and/or score student programs. Spacco, *et al.* introduced

Marmoset [11], an automated system for student submission, and used this system to study the development process followed by students in programming assignments. Edwards and Perez-Quinones introduced Web-CAT [6] as a system that "grades students on how well they test their own code." Norris, *et al.* introduced ClockIt [10] to monitor student development practices, with the goal of identifying and encouraging successful practices. Murphy, *et al.* introduced Retina [8] to log and aggregate data about student progress, providing high-level reports (to both the instructor and the students) and specific feedback to each student.

Some previous work has focused on the value of feedback in assisting students toward a working solution. Hattie and Timperley [12] reviewed the general use of feedback in education, addressing the varying effects of specificity, timing, and frequency of feedback, and considering positive vs. negative feedback. Regarding timing of feedback, they note some indication of beneficial effects in delaying task-level feedback, but providing immediate process-level feedback. Our data shows that students make relatively poor use of immediate feedback on program correctness; although as educators we intend for programming assignments to help students to develop process-level skills, this behavior suggests that they see the assignments simply as tasks to be completed and respond to feedback accordingly.

Nordquist [9] introduces a submission system much like the one presented in this paper, with immediate feedback on correctness and an opportunity for repeated submissions; Nordquist's system also allows for students, having completed their own solution, to request and receive the instructor's solution. Edwards and Perez-Quinones [5] discuss the value of automated grading and address strategies for providing student feedback.

3. Methodology

We gathered data from and about programs that students created and submitted for CS1 courses offered over several semesters. Each course section is given approximately 75-80 programming assignments (in C/C++) to complete during the semester. Each assignment, or group of assignments, increases in difficulty, as new concepts are introduced by lecture, reading and examples; our curriculum emphasizes procedural skills early and introduces classes and objects in the third semester. The CS1 programming assignments are designed to be completed sequentially through the semester, utilizing skills and occasionally whole functions developed through previous work.

Students attempt to solve each problem by submitting source code to a system that timestamps, stores, compiles and runs their code to verify the correctness of their solution. Correctness is primarily determined by verifying that correct output is generated for specific test cases and carefully generated random test cases. If an error is discovered, either run time or logical, students receive immediate feedback about the failed test case: the given input, the expected output, and the actual output of their program. Students are allowed to repeatedly submit attempts for each assignment, as often and as many as they desire, until the specified deadline.

3.1 Process Metrics

Our policy permitting iteration allows us to investigate some aspects of the problem-solving process followed by students, such as number of attempts and time spent between attempts while working on a specific problem. In combination with computed scores for each submission we can measure the following aspects of each student's progress.

Work Sessions

A work session is the span of time in which the student is actively working on their solution to a specific problem. We assume that an elapsed time between submissions of 30 minutes or less indicates that the student was actively working on their solution during that time. Thus a work session begins at some point before the first submission and continues as long as the previous submission was less than 30 minutes before. A gap of more than 30 minutes is taken to indicate that the student took a break, and the next submission was made after the start of a new session of work.

Iteration Interval

The iteration interval is the span of time between consecutive submissions within a work session. Iteration interval provides a quantitative measure of the step size employed by the student during problem solving. For our purposes here, individual submissions are grouped as occurring (relative to the student's previous submission) within one minute, between one and two minutes, between two and five minutes, between five and fifteen minutes, between fifteen and thirty minutes, or as the initial submission for a new session (more than thirty minutes after previous).

Work Time

Work time is the total number of minutes a student worked on a specific problem. An estimate of total work time for a student can be computed as the total of iteration intervals plus the time before the first submission of each work session. Under the session assumptions described above, total work time includes the iteration interval if the time is 30 minutes or less. For the initial work period and intervals of more than 30 minutes, we rely on adjusted self-reported data from the students to estimate these time spans. The mechanism for collecting this information is explained in the next section.

3.2 Survey

In addition to the computed metrics above, a survey was conducted to gather information about student perceptions and to help estimate total work time. As noted above, the initial work interval at the start of a session cannot be determined automatically and yet may be the bulk of the total work time. The survey asked the following questions:

```
1. Typically, how many minutes did you
work on a problem before making the
first submission?
2. Typically, how many minutes did you
work on a problem during one sitting?
3. Typically, after leaving a problem
for a time and coming back to it again
later, how many minutes did you work
before making another submission?
4. Typically, how many total minutes did
you spend on each problem?
5. Typically, how many submissions did
you make for each problem?
```

Questions 1 and 3 above are the core of this instrument, seeking to learn the "start up" time in a working session before submitting an attempted solution. Questions 2 and 4 can be used to check consistency with 1 and 3, since the iteration interval times computed automatically can be deducted, leaving time before the first submission and time before other sessions. Question 5, whose answer is known from gathered data, allows us to crudely measure the accuracy of student responses.

3.3 Source Code Metrics

The solution to each problem presented is source code of a working program able to complete the task described in the problem statement. Another avenue of evaluation for student submissions considers properties of the source code using common metrics. Solutions for each problem have been created by instructors and teaching assistants, although it is important to note that, while the "correct" solutions were created by knowledgable programmers, they do not necessarily represent the best possible implementation. These programs were primarily written to be clear and straightforward implementations. Additionally, they are written to reflect the solutions that can be reasonably expected to be within reach of the students, given their experience level and the course material covered prior to that point. Thus, these solutions provide a baseline against which student solutions can be measured and quantified. Since students are allowed and expected to iterate toward a working solution, we will consider only those submissions which successfully solve the problem.

Since our CS1 curriculum is primarily concerned with procedural concepts, we concentrate on "size" metrics as a surrogate for overall solution difficulty. We have chosen to utilize these measurements: source lines of code, McCabe cyclomatic complexity, and number of declared variables. Together, these metrics provide some indication of the difficulty level for a specific program.

Source Lines of Code

Source Lines of Code (SLOC) is the number of lines of code in a program after removing comments, character strings, and blank lines. Removing comments, character strings and blank lines is an attempt to reduce discrepancies between authors; intuitively, this provides a more accurate measurement of the meaningful source code. SLOC has traditionally been used as a rough measure of size and effort; it is included here because it is easy to measure and has been widely used in the past. In addition (and anecdotally), SLOC appears to be used by the students themselves to assess the relative difficulty of a program.

Cyclomatic (McCabe) Complexity

Cyclomatic complexity [13] is a measurement of the number of linearly independent paths of execution in a program, a quantitative measurement of the logical complexity of a program. The easiest way to compute cyclomatic complexity is as the sum of the number of flow-control branches plus the number of separate functions in source code. Since student submissions are limited to C++, the only branches to take into account are conditional statements (`if`, `case`, and the `?:` tertiary operator) and looping constructs (`for`, `while`, and `do`). Cyclomatic complexity correlates with control-flow complexity and is used here as a measure of the cognitive load or difficulty of the problem in that sense.

Declared Variables

Declared variables count the variables of types common to CS1 programs (`int`, `float`, `double`, `bool`, `long`, `unsigned`, `char` and `string`) which are declared in the source code. This metric considers all declared variables in global scope as well as within the scope of functions, however it excludes function parameters. Declared variables roughly captures the state space of the program, representing another dimension of program complexity.

Fig. 1: Source code metrics plotted against time in semester

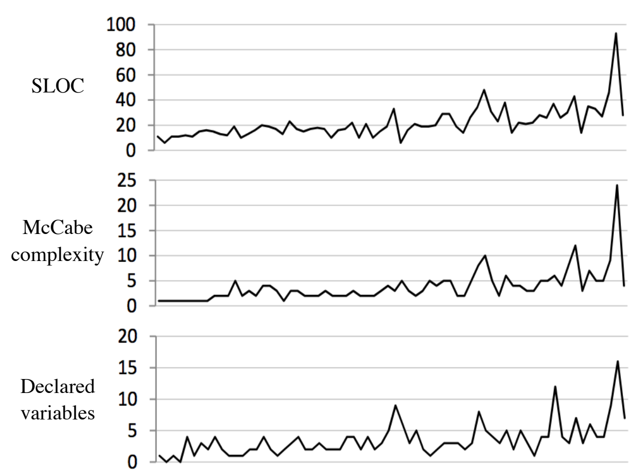


Figure 1 depicts (respectively) the expected SLOC, McCabe complexity measurement, and number of declared variables for each solution throughout the course of a semester. Each measurement reflects a gradual increase in difficulty over time. There are small peaks later in the semester, indicating more difficult assignments that often combine several concepts, but generally the difficulty level rises steadily as new concepts are introduced and students are expected to demonstrate greater proficiency in programming.

Although a number of different metrics could be applied to source code, we believe that these metrics provide a reliable measurement of the effort required to create CS1-level programs.

Figure 2 shows one possible solution to a typical assignment from the curriculum (approximately halfway through a semester): displaying Pascal's Triangle. Students are encouraged to use a previously-created implementation of the `Factorial` function and create the `Combinations` function to calculate binomial coefficients. In this implementation, SLOC is 28, McCabe complexity is seven (three functions plus four loops), and declared variables is six.

Fig. 2: Pascal's Triangle

```
#include <iostream>
#include <iomanip>
using namespace std;

int Factorial(int n)
{
    int f = 1;
    for (int i=2; i<=n; i++)
        f *= i;
    return f;
}

int Combinations(int n,int k)
{
    return Factorial(n) /
           (Factorial(k) * Factorial(n-k));
}

int main()
{
    int n;
    cout << "How many rows? ";
    cin >> n;

    for (int i=0; i<=n; i++) {
        for (int j = 0; j < n-i; j++)
            cout << "   ";
        for (int j = 0; j <= i; j++)
            cout << setw(6) << Combinations(i,j);
        cout << endl;
    }
}
```

4. Results

In this section, we present the results of collecting the information described in Section 3. In the next section we provide our interpretation of these results.

Data was collected from programming assignments in six semesters of a CS1 course, Spring 2009 through Fall 2011, in which 192 students were each expected to complete 75-80 problems. Students were allowed (and expected) to submit multiple attempts while working toward a solution; nearly all students who attempted a given problem eventually completed it. Table 1 shows the distribution of final results, after iteration, for all student-created solutions where the student submitted at least one attempt.

Table 1: Final Results for Student Solutions

Best result reached	Count	Percent
Submitted, but did not compile	284	2.8%
Compiled, but not fully complete	322	3.2%
Successfully completed	9532	94.0%

4.1 Process

The 9,532 correct student solutions required 44,548 submissions, or an average of approximately 4.7 submissions and 1.3 sessions per problem to reach a solution. If only problems with "correct" McCabe complexity of seven or higher are considered the average number of sessions grows to 1.5.

Table 2 displays student submission counts for successfully completed problems, grouped by iteration interval. The rightmost column of Table 2 shows how many (and overall percentages) of the submissions from each time "bucket" resulted in an increase over the previous highest grade in that assignment for the student. The average interval between attempts within the same session is 2.6 minutes.

Table 2: Submissions by Iteration Interval

Since Last Submission	Count	New Highest Score
First submission	9532	
< 1 min	14628	5332 (36%)
1-2 min	7652	2429 (32%)
2-5 min	5729	1753 (31%)
5-15 min	3079	936 (30%)
15-30 min	946	279 (29%)
New session	2982	919 (31%)
Total	**44548**	**11648 (26%)**

4.2 Survey

We sent 384 surveys to current (57) and recent (327) students and received 92 apparently valid responses, a 24% response rate. Invalid responses (*e.g.*, answering 240 to every question) have been removed. Still, a wide variation in responses was seen, as indicated by the standard deviations for each question. Student responses are characterized in Table 3.

Table 3: Survey of Time Required

	Minutes before first submission	Total minutes in typical session	Minutes on a new session before submission	Minutes spent on typical problem	Avg submissions for a problem
Average	34	52	18	67	6
Std. Deviation	40	35	15	64	6

4.3 Source Metrics

For our problem set, the average SLOC in the “correct" solutions is 23.8; over all student solutions, the average SLOC is 28. Table 4 displays the McCabe complexity of student programs in comparison to “correct" (or expected) solutions. Table 5 displays the number of declared variables in student programs in comparison to “correct" solutions.

5. Interpretation

From the data shown here, it is clear that most student submissions fail in both the overall goal of solving the stated problem and the more immediate goal of simply improving their score. Students submissions tend to occur in spurts of several rapid attempts interspersed with longer intervals. Half of all student submissions (50.0%) occurred within two minutes of the previous submission; approximately one-third (32.8%) of all submissions occurred within one minute of a previous submission.

An examination of a random sampling of these quick-turn-around submissions reveals several different types of changes being made. As might be guessed, many include trivial corrections for typographical errors such as simple compile errors (*e.g.,* missing semi-colon) or simple output mistakes (*e.g.,* printing the wrong variable or a misspelled input prompt). However, another category of submissions exist which contain minor changes made hurriedly and seemingly somewhat at random. In many cases the change is unrelated to the feedback received on the previous attempt. It may be that the policy allowing unlimited attempts is being abused in these cases. However, the number of successful completions of student problems, given that any attempt was made, leads us to believe that allowing students to iteratively submit, test, and alter their programs until a solution is reached encourages learning and problem-solving.

We see that students, on average, write their programs with a couple more lines than are really needed to solve the problem. Such a small difference in length, however, may indicate nothing more than small variances in stylistic preferences, rather than significant differences in logic and implementation.

5.1 Survey

The total amount of time T that students work on a typical problem is the sum of the time spent in all sessions working on that problem, or the sum of the lead-in time in the first session plus the lead-in time in later sessions plus the average working time after the first submission of each session.

Table 4: Flow Complexity of Student Solution

McCabe Complexity of “Correct" Solution	5 or more below	2 - 4 below	correct $\pm$ 1	2 - 4 above	5 or more above	Average McCabe Complexity	Number of Student Solutions	Problems Given
1			.986	.013	.001	1.07	1177	77
2			.912	.077	.011	2.41	2167	163
3		.001	.896	.078	.025	3.66	1984	151
4		.020	.871	.083	.026	4.55	870	63
5	.002	.061	.689	.140	.109	6.79	1244	103
6		.023	.713	.179	.085	7.41	613	64
7	.051	.145	.589	.145	.069	7.22	331	29
8	.053	.220	.542	.163	.022	7.82	227	23
9		.279	.311	.200	.211	10.95	190	19
10		.014	.771	.129	.086	11.11	70	9
11	.101	.242	.450	.087	.121	11.67	149	15
12	.050		.660	.160	.130	13.15	100	10
13	.019	.028	.556	.194	.204	14.94	108	12
14	.273	.045	.205	.023	.455	18.39	44	5
15			.154	.231	.615	20.62	13	3
All	**.022**	**.032**	**.804**	**.093**	**.048**	4.95	9532	770

Table 5: State Space in Student Solutions

Number of Variables in “Correct" Solution	5 or more below	2 - 4 below	correct $\pm$ 1	2 - 4 above	5 or more above	Average Number of Variables	Number of Student Solutions	Problems Given
0			.285	.649	.066	2.24	847	50
1			.508	.448	.044	2.48	1527	107
2		.033	.850	.105	.012	2.42	1938	152
3		.064	.790	.114	.032	3.44	1934	152
4		.090	.705	.157	.049	4.65	1518	124
5	.005	.154	.636	.143	.062	5.30	629	59
6	.022	.255	.484	.122	.117	6.37	368	41
7	.015	.145	.512	.275	.052	7.66	324	33
8	.007	.132	.523	.245	.093	8.92	151	17
9	.163	.297	.302	.151	.087	7.83	172	16
10	.050	.250	.550	.100	.050	9.45	40	8
11	.500	.063	.063	.313	.063	10.06	16	2
12	.364		.636			9.73	11	3
14	.714	.036	.143	.107		6.25	28	2
All	**.008**	**.068**	**.648**	**.231**	**.045**	3.82	9532	770

So, clearly we have:

$$T = [F] + [B * (S - 1)] + [(M - S) * N] \quad (1)$$

F is the average lead-in time in the first session
B is the average lead-in time in any later session
M is the average number of submissions per session
N is the average iteration interval (between submits)
S is the average number of sessions per problem.

The typical iteration interval, number of submissions per session, and number of sessions per problem can be calculated from student submission data; the values S, M, N used below are calculated for just the survey respondents and so may vary slightly from the overall averages.

The values F, B are not measurable by automatically gathered data and so rely on student-reported data. By survey results, students estimated an average $F = 34$ minutes spent on the problem before their initial submission (question #1) and $B = 18$ minutes at the beginning of each later session (question #3).

$$\begin{aligned} \text{Then, } T &= F + B * (S - 1) + (M - S) * N \\ &= 34 + 18 * (1.3 - 1) + (4.6 - 1.3) * 2.6 \\ &= 48 \end{aligned}$$

This conclusion is lower than the student-reported average of 67 minutes per problem. We believe that the difference is attributable to the imprecision of student estimates. Students also tended to overestimate the number of submissions for problems, as compared with known data. Despite differences, the rough similarity of known and student-reported data supports an assumption that, on average, students do spend about an hour per problem. Given approximately 75 problems in the course, this implies that students spend about 75 hours working on programming assignments or approximately 5 hours per week (in a 15-week semester).

5.2 Source Metrics

The average McCabe complexity for the expected solutions is is 3.40, as compared to an average McCabe complexity value of 4.95 for student solutions. It may be possible to craft a solution with a lower than expected McCabe complexity measurement, by employing optimizations not yet introduced in class or by discovering and exploiting gaps in the automated scoring of the problem. However, despite the fact that expected solutions place more emphasis on clarity than optimal McCabe complexity, 14.1% of all student submissions were two or more above the McCabe complexity for the solutions, as opposed to only 5.4% of submissions that were two or more below the McCabe complexity for the expected solution. While most students were able to complete assignments within one McCabe complexity of the "correct" solution, the average difference of one McCabe complexity reflects students' overall tendency to solve problems with higher complexity than required.

The larger average complexity is likely caused in part by an incomplete grasp of the assignment, and thus a tendency to introduce additional, unnecessary logic to handle cases not required by the problem statement. Students may also occasionally use an unnecessarily complicated and circuitous algorithm to solve an assignment, but even for those submissions which use the same algorithmic strategy as the expected solutions, additional McCabe complexity may arise from an imperfect understanding of Boolean algebra used to reduce logical expressions and control flow to a more minimal level.

Again, it may easily be possible to use fewer variables; the "correct" solutions were not intended to be fully optimized by any standard, but simply to reflect expected student behavior at this point in the course. The average number of variables per "correct" solution is 2.91, as compared to an average variable count of 3.82 for student solutions. Student submissions once again exhibit higher complexity than the expected solutions, in the form of increased state space, with 27.6% of student submissions using two or more variables beyond the number appearing in the "correct" solutions. This is to be expected, as students are still learning the basic concepts of programming and may not fully grasp when it is necessary or even helpful to add an additional variable to the program.

Having examined metrics for the complexity, state space, and length of student solutions, we see that students tend to craft solutions that are slightly larger (in every dimension) than is required. This excess likely stems from their iterative approach to problem-solving, identifying and correcting one problem at a time and persisting until no problems are apparent. It may also be indicative of the average student's incomplete understanding of concepts like state space and Boolean algebra.

6. Future Work and Conclusion

In this paper, we have presented data gathered from CS1 sections over the past few years and reported on measurements of both the process and results of student programming when using an iterative assignment framework. This data suggests that students that the advantage of very high completion rates may be offset by more complicated solutions when given instant feedback and unlimited retries, possibly because the framework lessens the need for (and perceived importance of) full understanding. When encountering issues, students "fix" the problem by introducing additional control flow structures, thus increasing the complexity of the solution.

In future work, we will observe the effect induced by making some part of the student's score dependent on the McCabe (or similar) complexity measurement of their submission, relative to the expected value. Similarly, we will investigate the effects of "throttling" submissions by introducing either a maximum number of attempts, or minimum submission interval. We also plan to introduce measurements of student confidence in the correctness of their program. Given the high failure rate of student submissions before reaching a solution, we think it would be valuable to assess the confidence students feel in the correctness of their programs prior to submitting them for scoring.

References

[1] S. Edwards, "Improving student performance by evaluating how well students test their own programs," *Journal on Educational Resources in Computing*, vol. 3, no. 3, pp. 1–24, Sept. 2003.

[2] S. Edwards, J. Snyder, M. Perez-Quinones, A. Allevato, D. Kim, and B. Tretola, "Comparing effective and ineffective behaviors of student programmers," in *Proceedings of the fifth international workshop on computing education research workshop (ICER'09)*. ACM, 2009, pp. 3–14.

[3] T. VanDeGrift, T. Caruso, N. Hill, and B. Simon, "Experience report: getting novice programmers to think about improving their software development process," in *Proceedings of the 42nd ACM technical symposium on Computer science education (SIGCSE'11)*. ACM, 2011, pp. 493–498.

[4] A. Venables, G. Tan, and R. Lister, "A closer look at tracing, explaining and code writing skills in the novice programmer," in *Proceedings of the fifth international workshop on computing education research workshop (ICER'09)*. ACM, 2009, pp. 117–128.

[5] S. Edwards, "Experiences using test-driven development with an automated grader," *Journal of Computing Sciences in Colleges*, vol. 22, no. 3, pp. 44–50, Jan. 2007.

[6] S. Edwards and M. Perez-Quinones, "Web-cat: automatically grading programming assignments," in *Proceedings of the 13th annual conference on Innovation and technology in computer science education (ITICSE'08)*. ACM, 2008, pp. 328–328.

[7] M. Jadud, "A first look at novice compilation behaviour using bluej," *Computer Science Education*, vol. 15, no. 1, pp. 25–40, March 2005.

[8] C. Murphy, G. Kaiser, K. Loveland, and S. Hasan, "Retina: helping students and instructors based on observed programming activities," in *Proceedings of the 40th ACM technical symposium on Computer science education (SIGCSE'09)*. ACM, 2009, pp. 178–182.

[9] P. Nordquist, "Providing accurate and timely feedback by automatically grading student programming labs," *Journal of Computing Sciences in Colleges*, vol. 23, no. 2, pp. 16–23, Dec. 2007.

[10] C. Norris, F. Barry, J. F. Jr., K. Reid, and J. Rountree, "Clockit: collecting quantitative data on how beginning software developers really work," in *Proceedings of the 13th annual conference on Innovation and technology in computer science education (ITICSE'08)*. ACM, 2008, pp. 37–41.

[11] J. Spacco, D. Hovemeyer, W. Pugh, F. Emad, J. Hollingsworth, and N. Padua-Perez, "Experiences with marmoset: designing and using an advanced submission and testing system for programming courses," in *Proceedings of the 11th annual conference on Innovation and technology in computer science education (ITICSE'06)*. ACM, 2006, pp. 13–17.

[12] J. Hattie and H. Timperley, "The power of feedback," *Review of Educational Research*, vol. 77, no. 1, pp. 81–112, March 2007.

[13] T. McCabe, "A complexity measure," *IEEE Transactions on Software Engineering*, vol. SE 2, no. 4, pp. 308–320, December 1976.

IPLP: A Method to Deliver Programming Techniques by Making Analogy

Hari Mohan Pandey[1], **Dr. Andrew Turner**[2]
[1]Department of Computing, Middle East College of Information Technology, Muscat, Oman
[2]Human Resource, Coventry University, Priory Street, Coventry, CV1 5FB, U.K.

Abstract: *- This paper mainly explores some of the fundamental problems encountered by faculties and students on computer programming courses at Middle East College of Information Technology (MECIT). The paper is based on the author's experience during teaching "Programming Logic Development" module. During the lecturing authors analyzed that students faced many difficulties in understanding the programming logic specially the concepts such as: data types, conditional statements, applying loops and many more. The objective of the proposed research is to apply analogy based approach known as "Intelligent Practice in Learning Programming (IPLP)". A structure called as analogy table, games, and a song have been built/written and implemented as a learning tool to make students understand the difficult programming concepts.*

Keywords: *Analogy, Analogy Tree, Analogy Table, Intelligent Learning, Learning Tools, Learning Programming, etc.*

1 Introduction

Fundamentals of computer programming is an integral part of computer science curriculum and a major stumbling block for many computer science/information technology/computer engineering students, particularly in the initial phase of the study.

In [1] Dunican explained that programming courses are difficult to understand for the beginners and he gave the idea of the analogy tree to make the programming understand to the students.

Jenkins [2] also explores the same problem. In his paper [2] he said "*Programming is a complicated business*". He gave some idea to learn basic programming techniques. If we are saying that learning programming logics are difficult for beginners this means that at the same time we are talking about to teach too.

In other word [3] we can say that the difficult to learn, programming skills are difficult to teach too. Computer Programming is a mandatory subject in many college or Universities education programs. Students faced difficulties in understanding this subject because they lack in analytical and logical concepts.

In [4] it was proved that the tertiary level students all over the world have difficulties in mastering the programming subjects or in other words problem in understanding the logic to develop the programs.

Kolling, M. and Rosenberg, J said [5] that the problem becomes more challenging when the teachers and students deals not only with the logics (algorithms and flowcharts) but also with programming languages such as C, C++, Java or any others.

I***ssues at MECIT:*** A similar issue related to programming is faced at *Middle East College of Information Technology*, at *Level-0. Students study a* module known as "*Programming Logic Development*", given to make students understand the upcoming programming languages. But students struggle a lot in grasping programming and logic development. Various teaching methods have been implemented by instructors over the years such as the use of module guide, use of different text and references, implementation of algorithms and flowcharts using raptors (tool for designing flowcharts) and others, still remains a challenge for students as well as teachers.

Aim of the paper: The aim of this paper is to introduce a new approach known as: "*IPLP"* a combined form of analogy tree, analogy table, games and songs to make students understand the computer programming specially data types, conditional statements, loops and many more concepts related to computer programming.

Outline of the paper: Section-2 is explores the problem statement in detail. Section-3 covers the proposed solution given by the author to make students understand basic programming logic. Section-4, 5, 6 and 7 explore the idea about the practices used in order to deliver the programming concepts effectively. These sections cover the approaches such as: game, an analogy, a story and a song used by faculty. Section-8 supports analogy based teaching and learning process. Section-9 shows the observations and analysis. Section-10 presents the conclusions and future enhancements planned for the present work. Lastly but not the least references have been given.

2 Problem Statements

Authors analyzed the problems and came up with some specific problems statements such as:

- Each programming language has some set of rules called syntax or signature and students who are tyro in the field of computer programming, find it very difficult to grasp the syntax and use it for writing the computer program.

- Difficulty in understating the programming notion because the learners have no counter parts in day to day life.
- Lack of analytical and logical abilities to grasp the programming logics.
- Lack of motivation in implementing the logic in the form of programming.

On the basis of above mentioned problem statements we have identified varying levels of students in the class as shown in Figure 1.

Level-1: *Students who are not having the logical abilities:* We observed that in a batch of 35 students approximately 10% - 12% students are not having the logical understanding. Even the students are not in a position to formulate the logic in textual forms. Hence, for such category of students writing program is hard nut to crack. But the interesting thing is that they have good aptitude abilities.

Level-2: *Students who are not having aptitude abilities:* Similar to level-1 approximately 10%-15% are not good in aptitude. Hence, they are not able to develop and understand the programming concepts.

Level-3: *Students who are having logical and aptitude abilities but not having the analytical abilities:* Students who are having logical and aptitude understanding but not having the analytical understanding comes in this category. For example suppose you are asking a simple question to multiply two numbers then they can develop the program but if you are asking the same question like without using multiplication symbol (*) multiply two numbers. According to our experience approximately 50%-60% population falls in this category.

Level-4: *Students fully comfortable*: This is the last level; in this category we kept those students who are comfortable in all aspects. Typically, an only 10%-20% student's lie in this category.

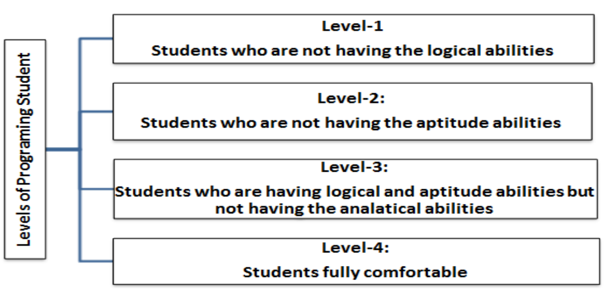


Figure 1: Different Levels of Programming Students

3 Proposed Solution

In order to address the problem identified in the *section-2* we must look in a pragmatic way that is we have to identify and describe a teaching and learning method where students can extract theory from practice and then applied back to develop the logical, aptitude and analytical abilities. We have proposed a method called as "*Intelligent Practice of Learning Programming"* (IPLP). Since students read and learn at different speeds that is a topic might be easy for some students while difficult for others because we all have different levels of intelligence (Howard Gardner, 1943).

Students grasp the logical concepts given in any programming module at varying pace depending on the type of problems. Hence, it is clear that one mode of delivery using some particular set of examples may not be good enough to understand the concepts in a better way.

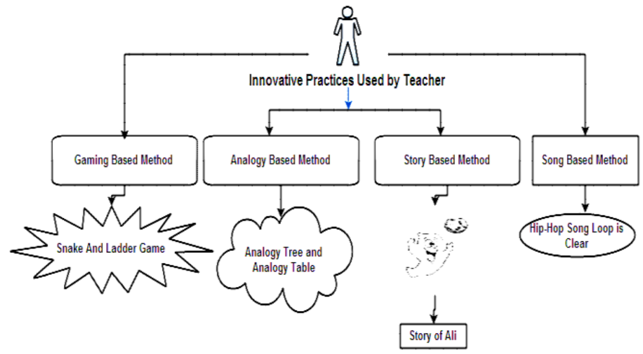


Figure 2: Innovative Practices used to deliver Programming Concepts

The problem we faced while teaching the "*Programming Logic Development*" at MECIT is that adequate material is not provided for all types of students (Figure-1) to understand the course materials. Even most online learning material do not give the idea on how to connect theoretical concepts with their implementations. Therefore a student suffers from the malady of programming concepts. And if you are not able to cope up with the programming concepts then you have to start from the scratch again and again iteratively. The example shown in Table-1 justifies the above mentioned statements.

4 Game Based Method: A Revision Tool

We used this method as a revision tool to revise previous lectures in an interesting and understandable manner. Our objective is to encourage students to work in team. Another objective of this method is to motivate students to communicate their idea freely. As far as game based teaching is concerned we used a popular game known as "*SNAKE AND LADDER*".

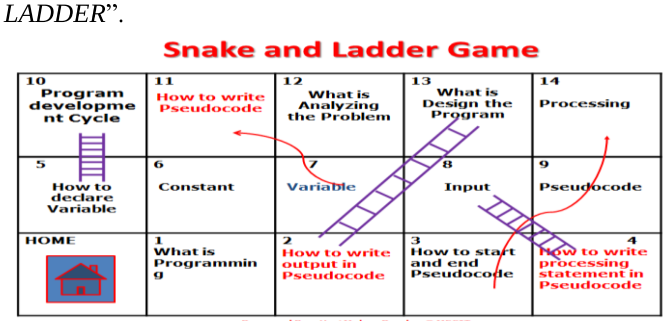


Figure 3: Snake and Ladder Game to asses' students' understand ability

The game with rules is given below:

- We will follow all the rules of normal Snake and Ladder game.
- Number is given in each box.
- Students have to pick a number based on the draw.

- S/he has to answer the topic given in the box to the whole class using a suitable exemples.
- Same student will get the chance to ask the question to any one in the class after giving answer of the question.
- This type of learning process encourages active participation of learners in the learning process.

5 Analogy to Deliver Programming

The techniques that have been investigated in this research are the use of analogy to make difficult concepts easy. Authors used this technique to explain data types, conditional statements, loops and other programming concepts.

We have given enough evidence to prove that using analogy for teaching computer programming course is an effective and efficient method of teaching and learning for an instructor and students. The structure of the analogy tree is simple to understand and ease to implement as shown in Figure 4.

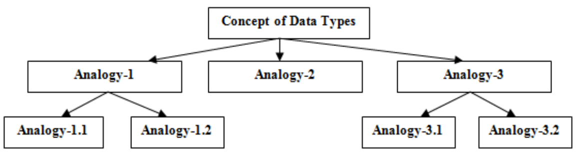


Figure 4: Structure of Analogy Tree to remember the analogy

The structure of analogy table is also simple as shown in Table-2 below:

Table-2: The Structure of Analogy Table

S.No.	Identifying Input	Implementation condition if any	Result/Output

5.1 Analogy for Data Types w.r.t Size

The analogy of data types is divided into two parts: first to teach the size of the data types and second for explaining the types of values which comes under any particular data types. Authors have used box based analogy (Table-3) to teach data types, which is most of the time hard to explain.

In author's experience, the most common problem faced by students in computer programming is understanding the fundamental data types especially in terms of their size and types of values any particular data types can store.

Table-3: Analogy for the Data Types based on size

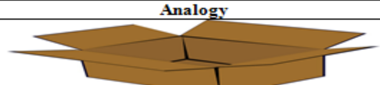
Analogy	Descriptions
	Real Data Type: Double
	Real Data Type: Float
	Integer Data Type
	Character Data Type
	String: Sequence of Characters

Table-3 explains the data types in terms of size using boxes. After looking at the box size it is easy to understand that the size of double data type is bigger than float, similarly the size of float is bigger than integer, integer is bigger than characters, and string is the sequence of characters that is the reason we used many boxes to explain string data types.

5.2 Analogy for Data Types w.r.t Values

In order to explain the types of values any particular data type can store we have used another analogy shown in Table 4. The data shown in Table 4 have been divided into two columns. The first column shows the analogy and the second column shows the description of an analogy.

5.3 Analogy for Conditional Statements

To explain the conditional statements and make the students grasp the concepts of conditional statements we used simple and conceptual analogy based on conditions. Some examples are given below:

Example-1: If-Then-Else Conditional Statement

A scenario based example has been discussed using analogy such as: *"Write pseudocode for the person whose name is Hari; who takes "warm coffee" if time is between 7A.M. to 10: A.M. Otherwise he takes "cold coffee"*.

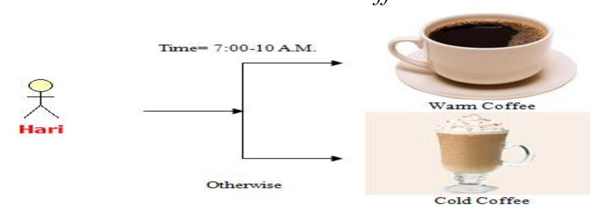


Figure 5: Analogy to understand If-Then-Else If Statement

Now, writing pseudo code (Figure-6) is an easy job as given below:

S.No.	Identifying Input	Implementation condition (if any)	Result/Output
Example-1	Input: time	time>=7 && time <=10	True: Warm Coffee False: Cold Coffee

```
Begin
Declare time As float
Declare Msg As String
Write "Enter the time"
Input time
If (time>=7 && time <=10) Then
    Set Msg = "Warm Coffee"
Else
    Set Msg ="Cold Coffee"
End If
Write Msg
End
```

Figure 6: Mapping from Analogy Table to Pseudocode.

Example-2: If-Then-Else If Conditional Statement

To explain the concepts of If-Then Else If statements we tried another analogy as given below: *"Write pseudocode for finding maximum of three numbers using If-Then-Else-If statements; take the input from the user"*.

Analogy: Suppose there are three persons (Hari, Jitendra and Anjum), each one is saying that he is having more money than the other two Table 5.

Now, after explaining the analogy shown in Table 5, questions were asked to the students to write the pseudocode and we were surprised all the students wrote and showed and the following:

Pseudocode

```
Begin
Declare n1 As Integer
Declare n2 As Integer
Declare n3 As Integer
Declare Msg As String
Write “Enter three numbers”
Input n1, n2, n3
If (n1>n2 && n1>n2) Then
    Set Msg = “n1 is maximum”
Else If (n2>n1 && n2>n3) Then
    Set Msg = “n2 is maximum”
Else If (n3>n1 && n3>n1) Then
   Set Msg = “n3 is maximum”
End If
Write Msg
End
```

Example-3: Select-Case Statement

Select case statement is one of the crucial concepts given in programming to deal with many conditions/cases in an easier and efficient way. But it is hard to understand for a student who has just started programming courses. Therefore, once again we used analogy to explain select case statement to the learners.

“Write pseudocodes to enter code and display messages Table-6 using Select-Case Statement accept the “code” as input from the user”.

Table-6: Display Messages w.r.t the code

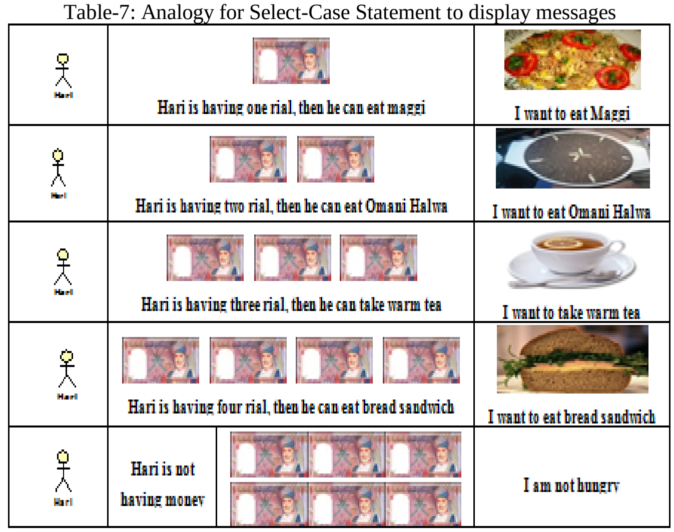
Code	Message
1	“ I want to eat Maggi”
2	“I want to eat Omani Halwa”
3	“I want to take warm tea”
4	“I want to eat bread sandwich”
Any other	“I am not hungry”

Next, the author asked students to write the analogy of Table-7 in the form of pseudocode. We found that about 95% students implemented pseudocode correctly as:

Pseudocode: To display message given in Table-6 w.r.t the code given

```
Begin
Declare code As Integer
Declare Msg As String
Write “Enter the code”
Input code
Select Case Code
Case 1:    Set Msg = “I want to eat Maggi”
Case 2:    Set Msg = “I want to eat Omani Halva”
Case 3:    Set Msg = “I want to take warm Tea”
Case 4:    Set Msg = “I want to eat bread Sandwich”
Default:   Set Msg = “I am not hungry”
End Case
Write Msg
End
```

Table-7: Analogy for Select-Case Statement to display messages



5.4 Effectiveness of Analogy Based Method

Analogy Table is effective enough to understand the key things required in writing the pseudocode and designing the flowchart for the beginners. It is very effective because of the following reasons:

- A learner can easily map the developed analogy in the form of pseudocode and flowcharts.
- A learner can investigate the conditions using analogy (second column).
- Learner can identify the outcome of any conditions using the last column.
- After using the analogy table learner will be in a position to develop his or her own analogy and map in the form of pseudocode and flowchart.

6 Story Based Method

We used the story based method to explain *loops*, *Types of Loops* and *ASCII* code. Here, is the example given to demonstrate the effectiveness of story based method.

- ***Title of the Story***: A Story of a little boy “Ali”
- ***Objective***: To understand the working of the loops in programming languages.
- ***Method Used***: A series of slides with animation used to explain the working of loop
- ***Story Starts***: Once upon a time there was a little boy in Muscat whose name was Ali.
- **What Ali likes?** Ali likes to *play with balls* when the *sun is shining*.

The diagrammatical representations of the story have been shown to the students in a stage wise manner as shown in the Figure 7 below:

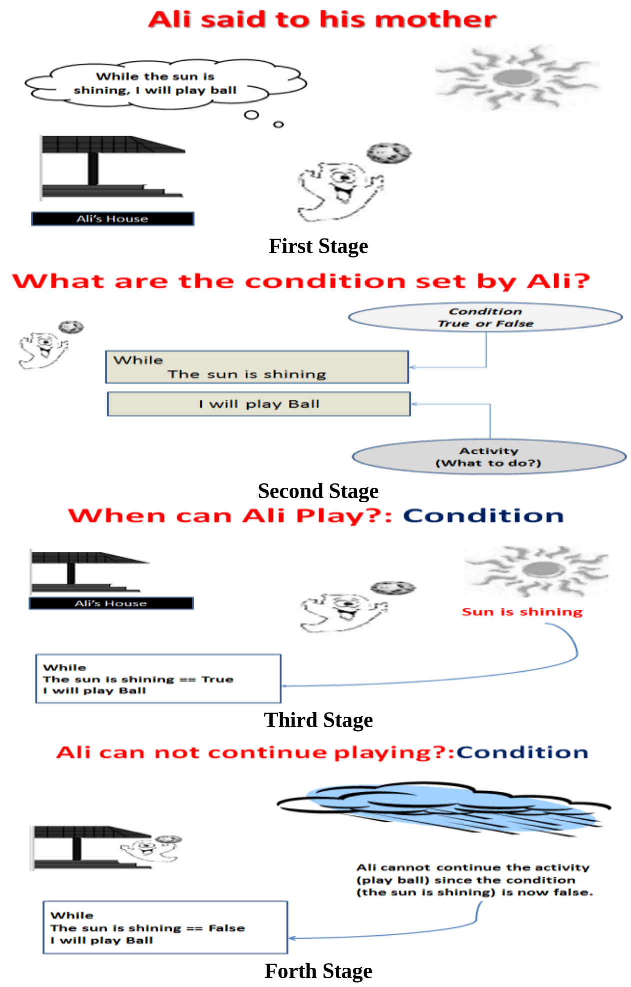


Figure 7: Story of Ali in a stage wise manner from first stage to forth stage.

6.1 Effectiveness of Story Based Method

Story based method was found very effective in the class when instructor used this method in one class then instructor found that the grasping speed was faster in comparisons to explaining the same concept in an ordinary way. By using this method instructor explained the following concepts related to loop:

- Repetition of set code means loop.
- If the given condition is true (Ali can play if sun is shining), loop will run.
- If given condition is false (Ali cannot play if rain will start) loop will not run.

7 Song Based Method

In order to explain loop in an effective manner authors of this paper wrote a song. The title of the song is "*Loop is clear*" which is given below:

"Loop is clear"

What is loop, what is loop, repeat the code that is loop, repeat the code that is loop, clear clear clear clear.
Types of loop, Types of loop, pretest loop, post-test loop clear clear clear clear.
While loop, and for loop are pretest loop pretest loop clear clear clear clear.
Three parts for "for" loop, initialize, condition, increment or decrement clear clear clear clear
Increment then add and decrement then subtract clear clear clear clear.
Loop is in the loop, nested loop, nested loop, clear clear clear clear.
Larger loop outer loop, another one is inner loop clear clear clear clear.

8 Developing Analogy Based Teaching

In the previous section we have given enough evidence to prove that using analogy for teaching computer programming course is an effective and efficient method of teaching and learning for an instructor and students respectively. The current research work is nothing but the development of two teaching methods first tree based analogy and second table based analogy that is analogy tree [1] and analogy table respectively. The structure of the analogy tree is simple to understand and ease to implement as shown in Figure 4.

The implementation of Table-2 (analogy table) for example-1, example-2, and example-3 given in section 5.3 are shown as in Table-8.

Table-8: The Structure of Analogy Table

S.No.	Identifying Input	Implementation condition	Result/Output
Example-1	Input: time	time>=7 && time <=10	True: Warm Coffee
			False: Cold Coffee
Example-2	Input: n1, n2, n3	n1>n2 && n1>n2	True: n1 is maximum
		n2>n1 && n2>n3	True: n2 is maximum
		n3>n1 && n3>n2	True: n3 is maximum
Example-3	Input: Code	Case1	True: I want to eat Maggi
		Case2	True: I want to eat Omani halva
		Case3	True: I want to take warm tea
		Case4	True: I want to eat bread sandwich
		default	True: I am not hungry

The Table-2 is still in consideration or in other words it is in its design stage based on the key aspects having yet to be investigated:

- How relevant feedback given to the learners.
- How to map developed analogy to human cognition.
- Developing an effective and easy analogy.
- How to map the analogy into analogy table.
- How to investigate condition using analogy.

9 Observations and Analysis

Author found that the impact of using analogy on students brought a drafting change among the students. In the academic year 2011-12 author taught "*Programming Logic Development*" module in four different sessions. In each session there are 35 students. We tested the above explained method in each session and found the following results given in Table-9.

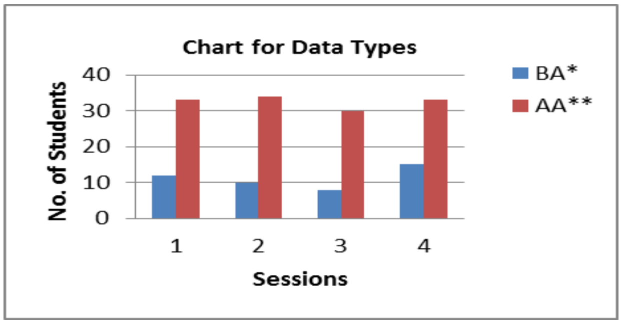


Figure 8: Graph shows the analysis on the data given in Table-9 for Data Types

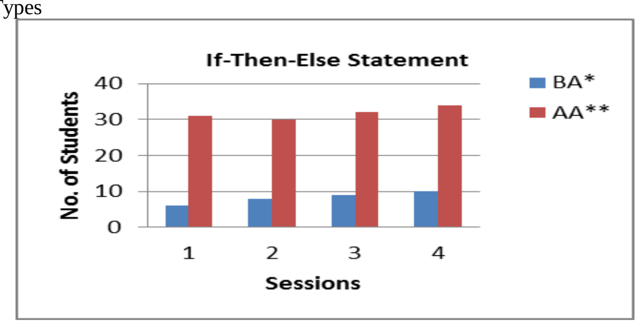


Figure 9: Graph shows the analysis on the data given in Table-9 for If-Then-Else statement

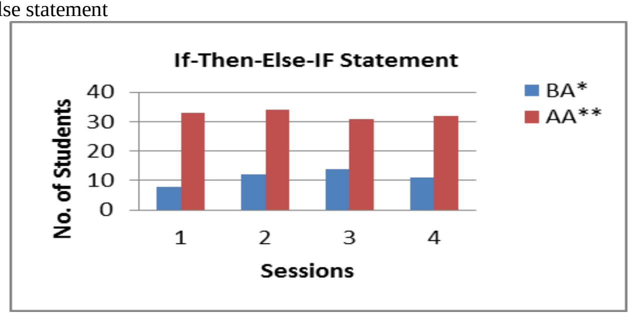


Figure 10: Graph shows the analysis on the data given in Table-9 for If-Then-Else-If statement

Figure 11: Graph shows the analysis on the data given in Table-9 for Select-Case statement

10 Conclusions and Future Works

In this paper we addressed the practical problems faced by an instructor most of the time during the delivery of programming logics to students, and at the same time learner face the problem in grasping the concepts. The presented research work shows a teaching method *Intelligent Practice of Leaning for Programming"* (*IPLP*) which is combined form of analogy tree and analogy table.

The authors tested the proposed method in different session of "Programming Logic Development" module at the Middle East College of Information Technology during the academic year 2011-12 and collected data as shown in Table-9. The results show that after using analogy with the delivery the behavior of students changed completely. Author gave many complicated problems to test the students understand ability in the course and found that after using analogy students was more comfortable in solving problem and developing logics.

The presented research work is still at the initial stage, a large amount of work has yet to be conducted to give a proper shape to the presented skeleton.

As far as the future work is concerned the authors primary goal is to structuring analogy tree and analogy table in such a way by which students can understand, identify and categorize the assigned programming problems in easy, effective, and efficient manner and map the used analogy into the core programming concepts. In future author main goal is to develop a system to identify the following:

- ➢Identifying Input, Output, and processing
- ➢Basic concepts to understand programming
- ➢Problem solving strategies

11 Acknowledgement

The authors sincerely extends their acknowledgements to Dr. T.R. Narayan, Dean, MECIT, Mr. Kiran G. R. Director, Academic Advancement, MECIT, Mr. Arun Shankarappa, Head Department of Computing, MECIT, for guidance and supports. Authors are also thankful to Ms. Samia Naqvi, Language Centre, Mr. Jitendra Pandey, Mr. Anjum and all the faculty of the Department of Computing for supporting in the presented work.

12 References

[1] Enda Dunican, Making the analogy: Alternative delivery techniques for first year programming courses. In J. Kuljis, L. Baldwin & R. Scoble (Eds), Proceedings from the 14th Workshop of the Psychology of Programming Interest Group, Brunel University, June 2002, 89-99.

[2] Jenkins, T. (2002). On the difficulty of learning to program. Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences, 53-58. Retrieved November, 2006 from http://www.psy.gla.ac.uk/~steve/localed/jenkins.html

[3] Allison, I., Orton, P., & Powell, H. A virtual learning environment for introductory programming. Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences, 48-52, 2002.

[4] McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y.B.-D., Laxer, C., Thomas, L., Utting, I., & Wilusz, T. A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. *ACM SIGCSE Bulletin*, *33*(4), 125-140, 2002.

[5] Kölling, M., & Rosenberg, J. Guidelines for teaching object orientation with Java. *ACM SIGCSE Bulletin, Proceedings of the 6th Annual Conference on Innovation and Technology in Computer Science Education, 33*(3), 33-36, 2001.

Table-1: Example with Pseudocode, flowchart, questions asked by students, analysis done by authors as an instructor, proposed solutions.

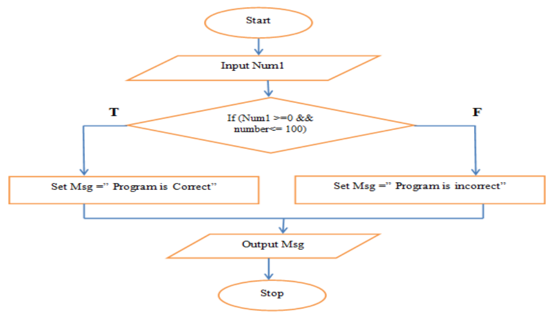
<table>
<tr><td colspan="2">Example Explained in Class: Write a pseudocode and design flowchart to display “Program is Correct” if the entered number is between 0 to 50 otherwise “Program in incorrect” using If-Then-Else statement.</td></tr>
<tr><td>Pseudocode:
Begin
Declare Num1 As Integer
Declare Msg As String
Write “Enter a number”
Input Num1
If (Num1>=0 && Num1<=100) Then
  Set Msg=“ Program is Correct”
Else
  Set Msg=”Program is incorrect”
End If
Write Msg
End</td><td>Flowchart:

</td></tr>
<tr><td colspan="2">Question Raised by Students:
• What is all about the pseudocode? It is very fancy name why we are writing pseudocode.
• Explain Integer.
• Why you are using two messages?
• What is the need to write the pseudocode, if anyone can tell the message after getting the number, then what is the need to do this much work?
• Why output is only one message at the end, if we don’t know number is according to given condition or not?</td></tr>
<tr><td colspan="2">Analysis done by Authors: As an instructor I answered all the questions asked by students at that moment of time, but after coming from class I analysed why students asked those questions, because I already finished those topics like data types, declaring variables, conditional statements, and writing output. Then I came to a conclusion that the concepts I explained to write pseudocode and flowcharts are not clear to most of the students. Hence, they all are perplexed completely.</td></tr>
<tr><td colspan="2">Solution Developed by Author: As an instructor I tried two new techniques to explain the difficult topics and both techniques are based on the analogy.
➢ Tree Based Analogy/ Analogy Tree and Table Based Analogy/ Analogy Table
Author gave this technique a name called as “Intelligent Practice of Leaning Programming” (IPLP).</td></tr>
</table>

Table-4: Analogy for the Data Types based on values

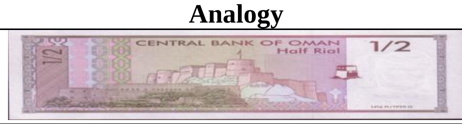
Analogy	Descriptions
	**Real Data Types Example: Money = RO ½ = 0.5.** That is any number with decimal point known as real data types.
	**Integer Data Type Example: Money = $ 1** That is any whole number (+ve, -ve or 0) known as integer number
	**Character Data Type Example:** Students were asked to identify the images then they said: **Image1= U, Iamge2 = X, Image3 = H and Image4 = T.** That is any keys we press from the keyboard except the function key called as character data type in single quote.
 Image-1  Image-2  Image-3  Image-4	**String Data Type Example:** Students were asked to write the name of all the images: Image1 = “………..” Image2 = “………..” Image3 =”………..” Image4 = “……....” **In reply they said:** Image1 = “Aero plane”, Iamge2 =” Middle East College of Information Technology”, Image3 =”Google”, Image4 =”Laptop”. All are the example of string that is sequence of characters.

Table-5: Analogy for If-Then-Else If statement for finding maximum of three numbers

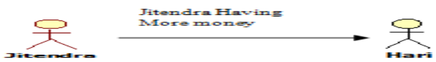
Jitendra is saying I am having more money

Hari is saying I am having more money

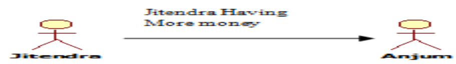
Anjum is saying I am having more money.

Instructor to Students: How to decide who is having more money?

Student's Answer: **Condition-1**: If Jitendra is having money more than Hari and more than Anjum than he is having more money



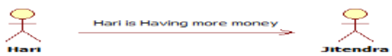
Case-1



Case-2

If case-1 and case-2 both are true than Jitendra is having more money

Condition-2: If Hari is having money more than Jitendra and Anjum than Hari is having more money



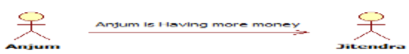
Case-1

Case-2

If case-1 and case-2 both are true than Hari is having more money

Condition-3: If Anjum is having money more than Jitendra and Hari than He is having more money or we can simply say that if Jitendra is not having maximum money (condition-1 is not true) and same for Hari (condition-2 is not true) than Anjum is having more money.

Version-1: If case-1 and case-2 both are true than Anjum is having more money



Case-1

Case-2

Version-2: If Condition-1 is false and Condition-2 is also false than Anjum is having more money

Anjum is having more money

Table-9: Test Result of Proposed Method

S.N.	Total Students in Sessions	Topics Covered							
		Data Types		If-Then-Else		If-Then-Else-If		Select-Case	
		BA*	AA**	BA*	AA**	BA*	AA**	BA*	AA**
1.	Session-L (35)	12	33	06	31	08	33	15	35
2.	Session-M(35)	10	34	08	30	12	34	14	34
3.	Session-B(35)	08	30	09	32	14	31	16	35
4.	Session-D (35)	15	33	10	34	11	32	13	32

BA*: Before Using Analogy **AA*:** After Using Analogy

Internal Conferences as a Constructivism Based Learning Arrangement for Research Master Students in Software Engineering

J.Mottok[1], I. Schroll-Decker[1], M.Niemetz[1], and G.Scharfenberg[1], B. G.Hagel[2]
[1] Faculty of Electrical Engineering and Information Technology,
University of Applied Sciences, Regensburg, Germany
[2] Faculty Computer Science,
University of Applied Sciences, Kempten, Germany

Abstract - *This summary describes the idea and the realization of "Internal" conferences as a constructivism based element in the education of Research Master students in Software Engineering. The conception is dealing with research based learning, activated learning, and constructivism based methods to acquire knowledge in an ideal way: performing research work, by preparation of posters, papers and presentations and by attending the conference as a presenter.*

Keywords: research based learning, activated learning, constructivism methods, internal conferences

1 Introduction

This paper deals with university internal conferences as an experimental laboratory for young academic research master students in Software Engineering to achieve competencies in presenting and expressing their own research results. Constructivism based methods are used in the seminar "Scientific Presentation" which is followed by the internal research master conference at the Regensburg University of Applied Sciences as a partner of the cooperative master's degree program of four Bavarian Universities of Applied Sciences. Because Software Engineering is a very volatile subject constructivism based learning methods will be recommended.

1.1 Cooperative master's degree

The cooperative master's degree program started at the Regensburg University of Applied Sciences in 2009. It is a cooperation program between four Bavarian Universities of Applied Sciences (Deggendorf, Nürnberg, Regensburg and Ingolstadt).

The aim of the program is qualification for the implementation of scientifically based application-oriented research and development work in the fields of Electrical Engineering and Information Technology, Software Engineering, Mechatronics and related disciplines. The goal is to educate students in analytical, creative and design skills. Technical, methodological and personal competencies are trained.

The teaching of these skills is based on related projects, applied in the research and development activities of the participating universities.

Thus, the relevance of topics is secured by the specific qualities of the involved faculties. Systematic work in the scientific research fields is managed by succeeding project phases. Both appropriate teaching modules, and the project and master thesis work, relevant for scientific publications, are an integral part of the course. The final thesis has the character of an independent and original paper which highlights the student's problem solving and method skills.

Professors supervise the research master students intensively in all research phases and in the corresponding seminars.

The advantages of a long term involvement of students in research master projects during a three semester education are that, in addition to the technical and methodological skills, the adaption of personal skills are promoted in practical training: teamwork, communication, language, and presentation skills.

The implementation of training objectives in connection with application-oriented research projects must involve appropriate scientific parameters. Criteria for this are:

- Sufficient experience of the faculty in research and development activities, e.g. demonstrated by relevant publications.
- Existence of an infrastructure in the participating laboratories.
- Interdisciplinary cooperation of colleagues in the field of work.

Research Master Students are involved in research projects from different faculties and universities of applied sciences. Therefore there is no strict subject canon as given in normal master examination regulations. In consequence every research master student can select subjects from existing master programs of the cooperating universities. In an assortment colloquy the student's subjects decision is checked against the demands of the research topics. The final subject selection must match the module structure given in Table I. The possible subjects are given by the master programs of the cooperating universities.

The German abbreviation SWS means Semester week hour (1 SWS = ¾ hour per semester week), whereas ECTS is the abbreviation for European credit transfer system. One academic year corresponds to 60 ECTS-credits that are equivalent to 1500–1800 hours of study.

The research master students use this module structure in the first and second semester of the cooperative master's degree program.

TABLE I. MODULE STRUCTURE – TEACHING MODULES

No.	*Module*	*SWS*	*ECTS Credits*
1	Mathematical, physical science, and natur sciences module	8	10
2	Technological module	4	5
3	Interdisciplinary Module	4	5

The module Research Methodology (see Table II) is offered by the involved universities as a network contribution.

TABLE II. MODULE STRUCTURE – RESEARCH METHODOLOGY

Semester	*Module*	*SWS*	*ECTS Credits*
1	Research Methods and Strategies, Part 1	6	6
2	Research Methods and Strategies, Part 2	6	6

In detail the following subjects can be selected by the research master students:

- Literature and patent search.
- Standards and standardization.
- Funding opportunities for national and EU projects.
- Academic writing and presentation.
- Scientific conference in combination with a master seminar.
- Rhetoric.
- Project methodology.

The high portion of research oriented work for research master students is summarized by the project work 1 and 2, and also by the final master thesis project as given in Table III. Overall there are 58 ECTS directly assigned for research related work. If possible, research master students work in cooperating research teams with PhD students.

TABLE III. MODULE STRUCTURE – PROJECT AND MASTERTHESIS

Semester	*Module*	*SWS*	*ECTS Credits*
1	Project 1	10	12
1	Seminar to Project 1	2	2
2	Project 2	10	12
2	Seminar to Project 2	10	2
3	Master project	-	28
3	Masterseminar	2	2

.

1.2 Background

Constructivism is a theory about the acquisition of knowledge, about learning and teaching [1, 2, 3, 4]. The core statement is that every person constructs his or her own personal reality by communicating with the environment. Thus, this perception of reality differs from the ones of other people. Learning is seen as the construction of meaning and therefore as dynamical evolution of the individual reality.

The educational approach of constructivism contradicts the thesis that knowledge can be transferred from the teacher to the learners. Constructivism emphasizes the origination of ideal conditions for learners so that they are able to construct their knowledge themselves. It prefers to support surroundings which enable the learner to acquire knowledge by researching and exploring of so-called authentic questions individually or in groups instead of providing information and skills to the learner as the traditional approach would suggest.

A constructivist learning environment [5] is characterized by giving the learner a chance to think, explore and discuss in collective cooperation with other learners. This constructivism approach is applied in the seminar "Scientific Presentation" and the corresponding internal research master conference.

2 Internal Conferences

2.1 Preparing Internal Conferences

The seminar "Academic Writing and Presentation" is placed in the beginning of the semester, whereas the scientific conference is placed in the final weeks of the semester cycle.

Both the seminar and the conference contribute to the module “Research Methods and Strategies”.

In order to attain our main objective of improving student involvement, motivation, and self-directed learning, we wanted to create a learning arrangement where the students could enjoy a sense of freedom. Those learning arrangements should include collaborative tasks that are of some personal interest to each student and promote individual and collective reflection about the seminar and research topics involved [6].

Therefore the research master seminar “Academic Writing and Presentation” is covered by the following elements:

- Basic information communicated by dialogue based lecturing of the professor.
- Self-organized learning phase [7] (students perform literature search by themselves).
- The student is able to design his or her own role using the learning environment and applying problem-solving-strategies.
- Preparation of a poster and a paper comprising of learning or research topics.
- Usage of a learning journal for documenting learning success and discovered problems with the learning issues.
- Group Puzzle (also called Jigsaw Puzzle [8]) – The students are self-directed and able to organize their learning process actively.
- Open Space [9].
- Usage of the computer based learning platform Moodle (blended learning approach).
- Retrospective – a feedback loop for collecting students’ learning experiences.

The seminar “Academic Writing and Presentation” is placed in the beginning of the semester, whereas the scientific conference is placed in the final weeks of the semester cycle. Both the seminar and the conference contribute to the module “Research Methods and Strategies”.

In order to attain our main objective of improving student involvement, motivation, and self-directed learning, we wanted to create a learning arrangement where the students could enjoy a sense of freedom. Those learning arrangements should include collaborative tasks that are of some personal interest to each student and promote individual and collective reflection about the seminar and research topics involved [6].

Therefore the research master seminar “Academic Writing and Presentation” is covered by the following elements:

- Basic information communicated by dialogue based lecturing of the professor.
- Self-organized learning phase [7] (students perform literature search by themselves).
- The student is able to design his or her own role using the learning environment and applying problem-solving-strategies.
- Preparation of a poster and a paper comprising of learning or research topics.
- Usage of a learning journal for documenting learning success and discovered problems with the learning issues.
- Group Puzzle (also called Jigsaw Puzzle [8]) – The students are self-directed and able to organize their learning process actively.
- Open Space [9].
- Usage of the computer based learning platform Moodle (blended learning approach).
- Retrospective – a feedback loop for collecting students’ learning experiences.

2.2 Preparation of a poster

Poster sessions provide an excellent opportunity to present research results. They are an approved technology since their introduction in 1970 [10, 11]. You can find a short discussion from the beginning of poster sessions in [12]. Until the present day you will find information about poster sessions in various articles [e.g. 13, 14]. Because it is one of the keystones of our internal conferences we will study them in detail under the viewpoint of constructivist didactic.

Constructivist didactic consists of three major elements applied in individual learning phases [1]:

- Construction: we construct our reality.
- Rekonstruction: we are the discoverers of our reality.
- Deconstruction: we unmask our reality.

We will see that all three elements are part of the preparation process of a good poster. A good poster will be one which transports all the information you want and captures the audience’s attention. So it may be a challenge for the students to prepare a good poster.

First we have to define the content of our poster. That’s an easy decision in the process of preparing the poster because the master students want to present the results of their research. Secondly we have to examine the audience of the poster session. That's not so easy and needs the help of the professor. In our case the audience consists of other students, professors and interested people from local industry and organisations. We are especially interested in the knowledge the audience will have in our subject of research. Normally they will be no experts in our research subject so we have to simplify our information.

When we know our audience and their knowledge, the students have to think about how they learnt the subjects of their research, more precisely how they constructed their reality. This method of self-reflection is not normally learnt by the students. So it is up to the professor to develop this method in earlier sessions. By thinking about their own learning process the students are able to put themselves in the position of the audience. Now they may have an idea how the audience will construct their reality. This knowledge will determine the design of the poster: which graphics do they need? Which information will they present as pure text? How will they guide the audience through all of their information so the audience will be able to construct their own reality? In the literature and the internet we find helpful information on how to design posters [10, 15, 16].

When the first version of the poster is ready the students can show it to the professor and to people comparative to the audience. In [15] we will find a nice process to check readability and clarity of the posters. A tester should read the sample layout aloud while the students listen. They notice every problem the reader will have with their poster. And the students will create new versions until every tester understands the information on the poster. Now the students have a poster which will be understood by the audience. For the students this is difficult because they have to reconstruct and deconstruct their reality from version to version. But they learn how the audience will think. In [17] Berkun describes that we need this process to get a good design. He quoted Einstein who said "the physicist's greatest tool is his wastebasket". So let's think about our wastebasket on the desktop as a great tool!

In addition to the poster the students should prepare a handout which they can give to the audience if they need more information. In that handout students may describe their results in greater detail.

Students should also prepare a small oral presentation of their poster for the people who will not read everything but are interested in their work. This presentation has to be consistent with what the audience can see on the poster, so it should be well prepared.

Now we will present a checklist for the professor:

- Define the content of the poster with your students.
- Tell the students about the audience.
- Tell the students about the audience's knowledge in the research subject.
- Tell the students about design of posters and show them links to designing guides.
- Be one of the first reviewers of the posters.
- Give hints to reviewers with the audience's knowledge.
- Help the students in preparing handouts.
- Advise the students of preparing a presentation.
- Advise the students of preparing a discussion about their poster.

Figure 1. Poster session

Let's write about our experiences with our students who never prepared a poster: The student's first question is often "Why can't we use slides for our presentation?" followed by "What is the difference between posters and slides? We only have to rearrange our slides to get the poster!" The students need time to understand that slides only maintain an oral presentation and that that is completely different from posters which were only read by the audience. Here the teacher has a big challenge to explain the differences and motivate the students to prepare good posters.

2.3 Usage of learning journal

As a flight recorder of the learning process each research master student writes a learning journal. A learning journal comprises the cooperating aspects in the learning process, the content and the learning process itself and also the working technique used.

This reflection about the personal working approach is carved out by the categories and questions in Table IV. The learning journal is a well verified memory tool for the final retrospective session of the seminar.

TABLE IV. QUESTIONAIRE – LEARNING JOURNAL

Category	*Leading Question*
Cooperation	How was the cooperation? How were the tasks distributed? How do we handle conflicts? Did I notice anything?
Content and learning process	What have I done? How have I done it? What was my result? Was I confident? What did I understand? What did I not understand? What was missing? In which topics will I continue?
Working technique	How effective is my method? Is my method adequate for the problem solution? For which cases is my method applicable? Pros and Cons of my method.

Each student will come back to his or her project group and will try to present a well-organized report to the group. Therefore each individual is essential; and that is precisely what makes this strategy so effective.

2.4 Group Puzzle (Jigsaw Puzzle)

Group Puzzle or Jigsaw Puzzle is a cooperative learning technique which increases positive educational outcomes. Each student's part is essential for the completion and full understanding of the given learning issues. This method can be found in the constructivism method pool [1].

The research master students of the seminar “Academic Writing and Presentation” are divided into small project groups (see Figure 2). The project groups delegate a team member to each working group. In the working group the students discuss and learn in a self-directed manner, so that each member of the working group can explain the complete learning issue.

Each student will return to his or her project group and will try to present a well-organized report to the group. Therefore each student is necessary for the project group; and that is precisely what makes this strategy so effective.

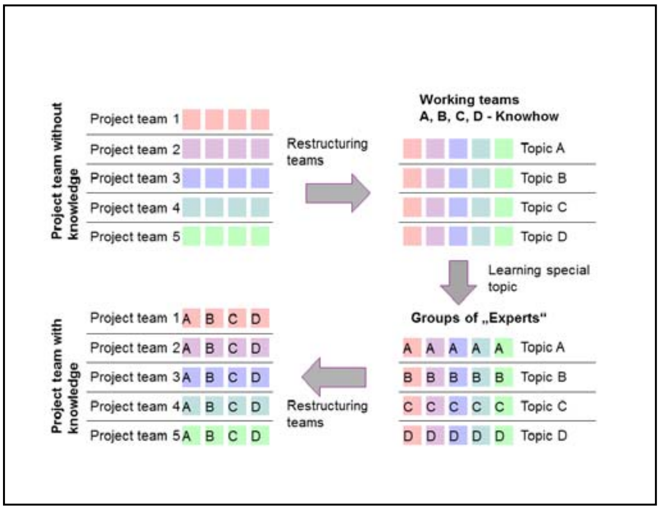


Figure 1. Group Puzzle used with the research master students in the seminar “Academic writing and presentation” .

During the seminar the Group Puzzle method is used for the following learning issues:

- How to create a poster presentation.
- How to write a scientific paper.
- How to perform a conference talk.

2.5 Open Space

Open Space is originally detailed in the introduction from Harrison Owen [18].

Open Space is a highly participative learning method in which participants self organize around an agenda. Participants generate issues and topics, e.g. poster presentations before or during the Open Space session. The discussion groups are self-selected by the students. Each group decides in which poster they are interested.

The process is particularly effective in uniting students with diverse competencies, in solving complex and contentious issues.

The poster session of the Applied Research Conference is performed in Open Space structure.

2.6 Retrospective

The retrospective as feedback instrument for quality improvement is used in software engineering process models, like SCRUM [20]. In our context of a constructivism learning arrangement the retrospective is applied as a kind of feedback loop about the learning process to finalize the seminar.

The leading questions for preparing a learning journal as given in Table IV are a good starting point for the retrospective session.

As a retrospective facilitator the supervising professor follows the content. A retrospective leader focuses on the process and structure of the retrospective. He attends to the needs and dynamics of the group and helps the group reach a goal. Retrospective leaders remain neutral in discussions, even when students have strong opinions [20].

When the content involves your own team, it is easy to get caught up in the discussion. It is tempting to jump into an engaging conversation, especially when you care about the topic, but you must refrain from doing so!

3 Experience

The proposed learning arrangement of internal conferences facilitated the self-organization of the students and created a context where they could work autonomously and felt pleased with what and how they were learning.

The first research master conference at Regensburg University of Applied Sciences was performed in April 2011. The students were highly motivated by the opportunity to present their research work to their peer students. They were consequently well prepared and practiced in the learned techniques for their research result presentation. Several students were encouraged to contribute their results to external conferences and scientific journals. An outlook is given on how to use the proposed learning arrangement in bachelor education courses.

3.1 Conference structure

The Applied Research Conference is placed in the 3rd semester of the research master education. The Applied Research Conference is prepared by the seminar "Academic Writing and Presentation" in the 2nd semester (see Figure 3).

The workflow of a real conference is used by our internal conference. We schedule a call for paper phase and a review process. Thereby the conference proceedings consist of only fully reviewed papers. Also the conference proceedings are printed with a citable ISBN. Students with highly scored papers will present their research results during the conference. In addition a poster session is established in the Applied Research Conference for student paper contributions with lower scores. The poster session is organized following the Open Space approach as given in [18].

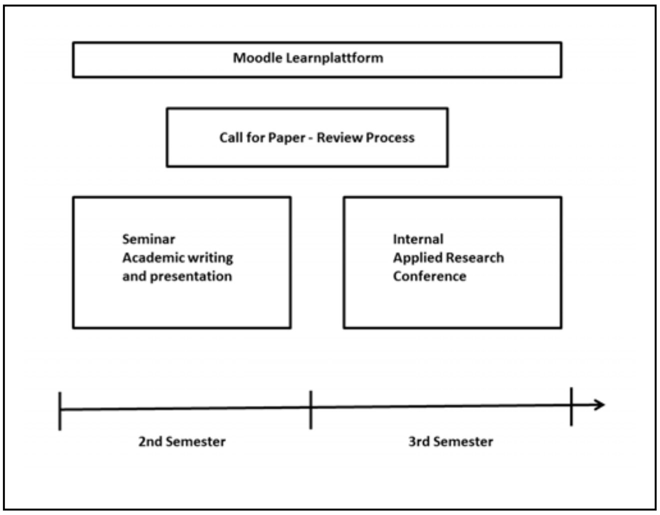


Figure 3. Sequential arrangement of the seminar "Academic writing and presentation" and the Applied Research Conference as interanl conference with review process. The learn platform Moodle is used for exchange between supervisors and research master students

3.2 Lessons learned

Students and the professor shared responsibility for all that happened. Our experiments in 2011 in a real setting, involving courses with many students, also showed that internal conferences in higher education, supported by appropriate strategies, is not only possible, but an interesting option to create valuable educational contexts that maximize students' participation and learning.

The process required effort, but this effort was rewarding and promoted participation and learning.

University teachers and students realize the potential impact Moodle may bring to their daily instruction and management.

The learner is to be encouraged to acquire knowledge and skills in a self-regulated manner. Learning as a process is, according to the catalogue of Arnold [21], also the development of self-competence. In this way the student develops the ability to motivate his or herself for life-long learning.

The sustainability of the students` competence development is proved by the following criteria [7]:

- The student is able to design his or her own role using the learning environment and to apply problem-solving-strategies.
- The students are self-directed and able to organise the learning process actively. Knowledge is created by the individual learner (by teamwork exercises,

individual research work, interview by the instructor, project work in cooperation with other learners).

- The students create the objects of learning constructively in their minds. Practice tasks and sample solutions are designed (see chapter 1, Background).

- The process of learning is situative. During the seminar “Academic Writing and Presentation” the learning students stay in a realistic situation by solving concrete problems. Applicable knowledge is created in a situation close to the reality of professional life so that inert knowledge is avoided (see Figure 3).

- Reflection and in-depth study of the theoretical essentials follows the practical task (in this specific case: an applied research task).

This way of proceeding is by John Dewey [22] the “father” of project learning: Human experiences originate from the interaction of experiencing and producing actions. Hence knowledge is created by action. The learner shows interactive behavior experimenting curiously. This means interactive pedagogy.

The five steps of interactive pedagogy are:

1. Practical action and primary experiences.
2. Problem related and reflectional thinking.
3. Rendering information material accessible.
4. Construction of working hypothesis.
5. Testing and validating the hypothesis by practical action.

- Learning through social interaction: brain research results can be described in an integrative interpretation [23]. In this respect, the holistic approach of processing information cerebrally confirms the constructivist viewpoint of didactics [24]. Caine depicts twelve principles of learning-teaching in [25]. Principle two is: “The brain is social.” i.e. a learning process which includes social interaction is effective. The cooperation of the learners in small-sized groups as study community is superior to individual learning.

- Learning is action-oriented. This means physical motion, the use of as many senses as possible and the being involved of both body and mind while learning.

This S-A-V-I-learning is characterized:

- o Somatically: learning by action and motion.
- o Auditively: learning by speaking and listening.
- o Visually: learning by observing and imagination.
- o Intellectually: learning by problem solving and monitoring the process.

All four forms have to be realized if optimal conditions for learning are required. A supervised project work appertains especially to S-A-V-I-learners.

The spider chart in Figure 3 shows the different criteria of sustainability in competence development. The distribution is the result of a survey among instructors and research master students.

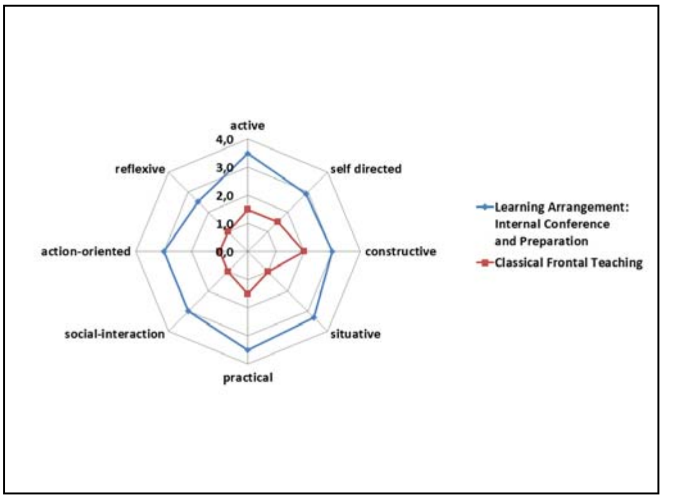


Figure 2. Criteria of competence development in the seminar “Academic writing and presentation” and the following internal conference (data from a survey among teaching staff and students)

4 Outlook

After the first Applied Research Conference, the described learning arrangement proves that it is a useful experimental laboratory to introduce young academic research master students to “Academic Writing and Presentation” and corresponding internal conferences.

Students emphasize the advantage of self-directed and active learning during the seminar. This was justified by a survey among instructors and research master students.

The integration of the learn platform Moodle into the seminar concept and the conference preparation has provided a more engaging learning environment.

The success lies in the awareness and willingness of the professor and the students.

5 References

[1] K. Reich, „Konstruktivistische Didaktik – Lehr- und Studienbuch mit Methodenpool“, 4. Ed, Beltz Verlag, 2008. Methodenpool: http://methodenpool.uni-koeln.de

[2] H. Siebert, „Didaktisches Handeln in der Erwachsenenbildung“, 5. Ed., ZIEL Verlag, 2006.

[3] E. von Glaserfeld, “An introduction to radical constructivism”, in: P. Watzlawick (ed.) (1984) “The invented reality”. New York: Norton, pp. 17–40. English translation of: E. von Glasersfeld, (1981) „Einführung in den Radikalen Konstruktivismus“, in: P. Watzlawick (ed.) „Die Erfundene Wirklichkeit“, Munich: Piper, pp. 16–38.

[4] J. Piaget, “La construction du réel chez l’enfant”, Neuchâtel: Delachaux et Niestlé (p. 311).

[5] J. Mottok, G. Hagel, M. Utesch, F. Waldherr, „Konstruktivistische Didaktik - Ein Rezept für eine bessere Softwareengineering Ausbildung?“ in Tagungsband des Embedded Software Engineeing Kongress' 2009, S. 601-610, 2009.

[6] J. Hofmann, “Blended Learning case study”, in The ASTD b-learning handbook: Best practice, strategies and cases studies for an emerging field, A. Rossett, Ed. New York: McGraw-Hill, 2002, pp. 516-519.

[7] J. Mottok, A. Gardeia, “The Regensburg concept of P-seminars, Global Engineering Education Conference (EDUCON), IEEE. 2011.

[8] G. Macke, U. Hanke, P. Viehmann, „Hochschuldidaktik, Lehren – vortragen - prüfen“, Beltz Verlag, Weinheim, 2009.

[9] I. Pereira, A. D. Figueiredo, “Extending open space technology
for blended learning”, 10th IEEE International Conference on Advanced Learning Technologies, Sousse, Tunisia, 2010.

[10] H.-J. Talbott, “Effective poster presentations for SAS® User Group Conference: Recommendations and techniques, NorthEast SAS Users Group Seventh Annual Conference, Philadelphia, Pennsylvania, USA October 2-4, 1994.

[11] T. A. Fretz, “Poster sessions: A new means of communication”, HortScience, 14(2), p. 106-107.

[12] R. C. Reba, “Relative advantages and disadvantages of poster presentations at a scientific meeting”, J. Nucl Med 1979 20(6), p. 582.

[13] C. R. Harris, G. L. Maricle and B. Birkenholz, “Poster presentation: the key to communication of ideas,” paper presented at the Annual Meeting of the American Association for Agricultural Education/American Vocational Association, Cincinnati, OH, USA, December 6th, 1990

[14] R. Montúfar-Chaveznava, M. Ali Yousuf, I. Caldelas, “Poster sessions as a strategy to motivate engineering learning” in Proc. Of the International Conference on Engineering Education ICEE 2008, 27-31 July 2008 in Pécs-Budapest, Hungary

[15] writing@CSU, “Writing Guide: Poster sessions”, Retrieved from the World Wide Web on Nov. 20, 2011: http://writing.colostate.edu/guides/speaking/poster/

[16] L. V. Harris, “Design tips for creating your scientific poster and communicating your message”, Retrieved from the World Wide Web on Nov.20,2011:http://www2.tech.purdue.edu/cg/courses/cgt411/profDev/ScientificPosterTips-DrH.pdf

[17] S. Berkun, “Why good design comes from bad design”, March/April 2000; Microsoft Developer Network. Retrieved from the World Wide Web on Nov. 20, 2011: http://msdn2.microsoft.com/en-us/library/ms993290.aspx

[18] H. Owen, “Open Space Technology: A User's Guide”, Mcgraw-Hill Professional, 1997.

[19] L. Zekanovi-Korona, B.K. Miovic, K. Fucko, “Moodle-application
in educations of students at the Unversity of Zadar”, MIPRO 2010, Opatija, Croatia, 2010.

[20] E. Derby, “Agile retrospectives: Making good teams great”, Pragmatic Programmers, 2006.

[21] R. Arnold, H. Siebert, „Konstruktivistische Erwachsenenbildung - Von der Deutung zur Konstruktion von Wirklichkeit“, Schneider Verlag: Hohengehren, 2003.

[22] J. Dewey, „How we think“, 1910, German translation, Zürich 1951.

[23] U. Herrmann, „Neurodidaktik“, Beltz Verlag, Weinheim, 2009

[24] M. Spitzer, „Geist im Netz, Modelle für Lernen, Denken und Handeln“, Spektrum Verlag, Heidelberg, 2008

[25] R. Caine, G. Caine, C. McClintic and K. Klimek, 2nd. Ed. “The 12 Brain/Mind Learning Principles in Action”, Corwin Press 2008.

A Process for Collecting and Analyzing Chronological Data for CS1 Programming Assignments

Raymond Pettit[1], Ryan Clements[1], Susan Mengel[2]
[1]School of IT and Computing, Abilene Christian University, Abilene, TX, USA
[2]Computer Science, Texas Tech University, Lubbock, TX, USA

Abstract - *In this paper, the use of chronological data is explored in analyzing students' submitted programming assignments. By capturing intermediate versions of a student's program during development, the progression of steps can be seen that the student took in attempting to solve the problem. This chronological data can be used to give instructors additional meaningful information as to a student's understanding of programming concepts and yet does not add any additional burden to the student while completing the assignment.*

Keywords: CS1, program assignments, computational thinking, assessment

1 Introduction

It is a challenge to access the mastery of concepts in college-level entry programming courses. Students come to a CS1 course from a diverse set of backgrounds. Some students have had much exposure to programming in high school, while others have had none.

The failure rate in college-level introductory computer science courses may be 30% or higher [1][2]. This failure rate found in a college's CS1 course may be one of the higher failure rates across campus. One of the contributing factors is that it is often difficult for an instructor to make an accurate assessment of what a student really understands at any given point in the course [5].

As programming skills are usually a goal of a CS1 course, instructors typically assign work in the form of programming assignments in which students must write a small, but fully functioning program that meets certain requirements. For a thorough discussion of ways to assess mastery of CS1 concepts, see [6].

For instructors, often only the end product of the student's efforts is examined; i.e., the completed program. From this completed program, correctly working or not, an assessment of the student's mastery over the given material is made.

One problem with only looking at the end product is missing the student's journey to get to the end, which can show quite a bit about the student's level of understanding, especially with regards to higher-level concepts and problem solving. An instructor who observed a student throughout the assignment would have more information from which to make an assessment of that student's mastery.

In order to capture the data of the student's journey to complete a programming assignment, the authors propose a process described in this paper as an experiment to capture some of that data and to see if it could be beneficial in assessing students. If the process is proved feasible and beneficial, it could be incorporated into many assignments in the future.

2 Background

In the CS1 course in which the experiment took place, the programming assignments were divided into two groups, in-class assignments and out-of-class assignments. Control could not be exercised over the resources that students used to complete an out-of-class assignment. Resource availability for in-class assignments, however, was strictly controlled. The in-class programming assignments consisted of both the non-computer-based write-out-the-program-on-paper style and the computer-based write-and-run assignments. Both of these styles were used for exams, quizzes, and in-class practice.

Part of the motivation of this work besides looking more into the student's problem solving process is to begin to develop a set of instruments to measure the student's ability to perform computational thinking [7] as introduced by Jeannette Wing. She defined computational thinking as embodying the skills of abstraction and automation. Subsequent work [8] has expanded upon the definition to include algorithms and design. To gain insight into the students algorithmic and design processes would be facilitated by observing how they solve problems.

Another part of the motivation of this work comes from observing students in lab who haven't truly mastered the material, but who can often produce a correctly working program, either through sheer luck or trying a multitude of approaches until something works. It is because of these motivations that the authors do not want to rely entirely on the finished product (the correctly working competed program) to assess the skill of the student.

3 Related Work

Work has been done to evaluate student progress within a single programming assignment by capturing and examining each compilation attempt [3]. Information captured this way can aid in analyzing the most difficult sections of the program and also point to areas of frustration. The authors automated the process of examining compiled submissions, but needed a substantial amount of training data for reasonable accuracy. Data examined included average number of consecutive compilations with the same edit location, average number of consecutive pairs with the same error, the average time between compilations, and average number of errors.

Intention-based scoring is a similar strategy that attempts to use students' intermediate submitted work (in the form of compile attempts) [4]. The goal of the process is to assess how close a student's initial attempt is to the correct solution. Examination of the intermediate forms was done manually and resulted in the classification of the bugs found.

4 Methodology

4.1 Participants

This experiment was performed during the Fall 2011 and Spring 2012 semesters in a required class, CS115: Introduction to Programming Using Scripting, for students in the following majors: Information Technology, Information Systems, Digital Entertainment Technology, and Math Education. The class consisted primarily of students from one of these four majors. The programming language used in this course was Python 3. No prior experience in programming was required to enroll in this course.

In the Fall 2012 semester, 26 students were given the assignment. There were 23 students who were given this assignment in the Spring 2012 semester. In both cases, this assignment was given as an in-class quiz which had a small (1%-2%) impact on the student's final grade. All students who were in attendance on the scheduled day took part in the assignment as a standard required quiz.

4.2 Procedure

Several methods for collecting chronological data during a programming assignment were explored. One method was to create a custom editor that the students would be required to use while completing the assignment. The custom editor would store data as the user entered it. This custom editor would give a great deal of flexibility and power to record all of the data that was wanted, but would require a significant amount of effort to build and deploy.

Another possibility was to do direct keystroke logging during the quiz period. The data could then be reconstructed and analyzed later offline. While keystroke logging programs are available, there would have been a significant amount of manual effort in implementation and processing the data received.

Fortunately, it was not necessary to capture every keystroke to obtain the information that was desired. The main concern was with the ordering of steps while completing the assignment. All that was needed was a lower resolution "playback" of the programming session. To this end, the authors experimented with the built-in archiving capabilities of the Google Documents editor. The Google Documents editor saves a new copy of a file after any 10 seconds of inactivity. Each of these intermediate versions can be easily accessed at a later time. By using Google Documents, the authors would also have the ability to collect assignments easily from the students when they finished and be able to store a permanent archive that could be accessed in the future.

In order to prepare for the experiment, a shared Google Document was created for each student. As the campus where the trial took place is a Google Apps Campus, all student user accounts are already connected to Google Documents. Creating a shared Google Document for each student, therefore, was a straightforward effort and took about one minute per student to accomplish. The authors are exploring ways to make shared document creation more of an automated process in the future. Even if students do not already have a Google account, they can create one for free.

On the day of the quiz, students simply looked at their Google Documents folder and found the shared document bearing the chosen name (i.e., CS115.01_Quiz_4.txt). This document opened as an empty document within Google's web-based text editor. Students were then given the problem description and directed to use the editor to write the solution.

This assignment was very similar to a paper and pencil- based programming test in that students did not have access to an integrated development environment (IDE), compiler, or interpreter. They were not allowed to do trial runs of the program and iterate on the feedback given in order to capture a more accurate picture of what the student fully understood.

The only benefit provided by the editor was the ability to manipulate text in standard ways such as deleting or moving previously typed lines. Although students can delete previously input text, the deleted versions are still saved as part of the Google Documents archiving process.

4.3 Data Collection

Data was collected in the form of intermediate versions of the modified text file. A researcher then stepped through the intermediate forms of a student's completed coding assignment while recording the order in which designated milestones were accomplished. Within Google Documents, a document owner can choose to view a detailed revision history of a given file. Once the beginning version of the file is selected, then it is a simple matter of stepping through the various revisions until the final version is reached. For example, typically one to two lines of new code per revision was seen.

Before the submitted programs were opened, the researcher analyzed the assignment and designated certain required milestones that must be present in a correctly working program. Milestones included: collecting input, printing output, declaring a function, calling a function, and performing a calculation.

CS115, Quiz 4, Castle Defense

You're in charge of the castle's perimeter defenses. Your first round of defense is catapults, but they will only kill a certain percentage of the enemies. After the enemies get past the catapults, your second line of defense is archers, but they will only kill a certain percentage of the enemies that had gotten past the catapults.

Write a program that will help you to predict the number of enemies that will get past both defenses (so you can plan for the boiling oil). The program should ask the user for the number of enemies, the expected catapult kill percentage, and the expected archer kill percentage.

Print out the total number of enemies expected to make it past both defenses and through to the castle (round the answer to the closest whole number). For values of 1000 or more, insert commas in your output.
Ex. 1,000 or 15,324 etc.

Define a main function exactly as shown here- **def main():**
From within the main function, ask for data from the user and do the printing. There should not be any user input or printing in any function besides **main().** Do not use global variables in the program. Any call to the **left_standing** function should be done from within the **main** function.

Create a function that, given the number of people in a group and a kill rate, will return the number of people that survived. Include all mathematical calculations in this function. Do not do any mathematical calculations outside of this function. Define this function exactly as shown here- **def left_standing(group, kill_rate):**

Sample Runs

Castle defense
Enter number of enemies: 100
Enter catapult kill percentage: 40
Enter archer kill percentage: 50
Total enemies remaining: 30

Castle defense
Enter number of enemies: 10
Enter catapult kill percentage: 50
Enter archer kill percentage: 50
Total enemies remaining: 2

Castle defense
Enter number of enemies: 12345
Enter catapult kill percentage: 10
Enter archer kill percentage: 20
Total enemies remaining: 8,888

Figure 1: The Castle Defense Assignment

For the assignment later described in this paper (about 15 lines of code and 8 milestones), it took a student researcher about one minute to step through a single student's assignment and record the ordering of the milestones. In the future, it may be possible to automate this process by specifying variable and/or function names and using a program that analyzes text.

At the end of this data collection process, a spreadsheet containing a milestone ordering for each student is filled out. From there, the data can be analyzed in a variety of ways as given below.

5 The Sample Problem

The process described in this paper has been used for two semesters in a CS1 course at Abilene Christian University. The first assignment that was given in this way is titled "Castle Defense." This assignment was given in the Fall 2011 semester and the Spring 2012 semester. The Castle Defense assignment as it was given to the students is shown in Figure 1.

This assignment was created to test understanding of functions and their use. By restricting the student to only use the function definitions given in the problem statement, the student must see the calculation required as composed of two similar calculations that can be performed to get the correct answer. Many students only see functions as a means of taking code from one part of a program and putting it in another section. They may have difficulty in understanding the value of having a function which can receive values, process them, and return a resultant value. They would not initially see a function as something that could be called multiple times. It is because of this misperception of functions that this assignment is of value in helping instructors understand deficiencies in students' perceptions of functions.

In order to assess a student's completed assignment as objectively as possible, a rubric was created for grading the assignment. This rubric was used strictly to analyze the final version of the program that the students completed. Each student earned weighted points for each of the 22 conditions that were accomplished successfully. The rubric is shown in Table 1.

In order to record the chronological data for this assignment, milestone markers were created. These milestones are required accomplishments along the way to completing the program correctly. However, these milestones could be achieved in any order and still produce a correct end product. It is through this analysis of the ordering that instructors can gain insights into the level of conceptual understanding of a student and their ability to design. These milestones are:

1.) Student opened file
2.) Function main() is defined
3.) Function left_standing() is defined
4.) Function left_standing() is called the 1st time
5.) Function left_standing() is called the 2nd time
6.) Input is requested of the user
7.) Print statement shows final result
8.) Body of left_standing() started

Table 1: Grading Rubric

	Castle Defense Grading Criteria CS115.01 Quiz 4	weighting
1.)	Both functions declared as stated	4
2.)	Mathematical calculations are only in left_standing()	4
3.)	Input calls are only done from main() function	4
4.)	Ouput is only done from within main()	4
5.)	left_standing() is not called outside of main()	4
6.)	main() is called	2
7.)	Proper output for header text ('Castle defense')	2
8.)	Values returned from input statements are stored or used	6
9.)	Input values are converted to number types	5
10.)	An attempt is made to print the final result	5
11.)	The final result is formated properly (text and commas)	3
12.)	left_standing is called at least once	7
13.)	left_standing() is called properly the first time	4
14.)	The value returned from the first call to left_standing() is stored or used	7
15.)	left_standing is called twice	7
16.)	left_standing() is called properly the second time	4
17.)	The value returned from the second call to left_standing() is stored or printed	7
18.)	left_standing() contains correct mathematical calculation	4
19.)	left_standing() returns a calculated value	5
20.)	All referenced variables are in scope (not including undeclared variables) (B)	5
21.)	All variables used have been declared (B)	5
22.)	All variables declared are used (B)	2
	Total	100

The first milestone simply gives a way to record the time at which the programmer opened the file for editing. Including 1) as a milestone allows for keeping elapsed times between milestones. The second milestone is fulfilled when the programmer completes the official function definition for the main() function. The third milestone is fulfilled when the left_standing() function is officially defined. In order to complete this programming quiz successfully, the left_standing() function must be called twice. Milestones 4 and 5 are fulfilled when these functions are officially called. Milestone 6 is fulfilled when the programmer uses the input() function to request input of the user. Milestone 7 is fulfilled when the program attempts to print out the final answer. The result may not be calculated yet, but the call to print the answer is made. The final milestone, 8, is fulfilled when the body of the left_standing() function has been started. The body included any work on the left_standing() function besides its definition or a blank return statement.

6 Results

6.1 Summary Data

For each student, an analysis was made as to the order in which the milestones were accomplished. These individual orderings were summed and the average taken. The summary information for the assignment given in the Fall 2011 semester and the Spring 2012 semester is shown in Tables 2 and 3.

The ordering summary data indicates that the two different groups of students, on average, completed the problem in similar progressions. As well, the standard deviation for a given milestone is not significantly different from one semester to the next.

Elapsed time ranged from a few minutes to a little over 10 minutes on each milestone. Grades ranged from as low as 20 to as high as 100 using the grading rubric in Table 2.

Table 2: Ordering Summary for Fall 2011 (n=26)

Milestone	Average	Standard Deviation
Student Opened File	1.00	0.000
main() defined	2.88	0.927
left_standing() defined	3.22	1.808
left_standing() 1st call	5.85	0.988
left_standing() 2nd call	7.08	0.793
Input gathered	3.80	0.957
Print final result	5.40	1.414
Body of left_standing() started	6.05	1.588

Table 3: Ordering Summary for Spring 2012 (n=23)

Milestone	Average	Standard Deviation
Student Opened File	1.00	0.000
main() defined	2.87	1.014
left_standing() defined	3.57	1.619
left_standing() 1st call	6.10	1.044
left_standing() 2nd call	7.36	0.809
Input gathered	3.74	1.096
Print final result	5.56	1.562
Body of left_standing() started	5.36	1.590

6.2 Comparison to Expected Orderings

There are many different orderings of these milestones that show a logical progression through the program. The authors created two straightforward orderings, to see if these were indeed the most popular. The first ordering suggestion is based on a novice following the flow of control of a program and basing their ordering of code completion on the program's flow. The second suggestion is based on what the more experienced programmers indicated as their preferred progression through the code. This ordering involved creating and finishing the main function, including calling the auxiliary function, before creating the auxiliary function. These two orderings are shown in Table 4.

Table 4: Suggested Orderings

Milestone	Flow of Control	Experienced
Student Opened File	1	1
main() defined	2	2
left_standing() defined	5	7
left_standing() 1st call	4	4
left_standing() 2nd call	7	5
Input gathered	3	3
Print final result	8	6
Body of left_standing() started	6	8

The summary results show that after defining the main() function, the next step (on average) was to define the left_standing() function. This is likely due to the student's beginning the assignment by including the problem constraints first. As the ordering of the defining of the left_standing() function may not be an indicator of the student's thought process, the authors later added the eighth milestone which better records when a student was giving their attention to the left_standing() function.

6.3 Exploring One Student's Data

As the authors began analyzing the collected ordering data and comparing it to grades earned in the assignment, a few surprises were uncovered. One of these surprises involved a student that scored a much higher grade on this assignment than his course average would have predicted. The student in question had a course grade that placed him as the 19th out of 23 students in the course. The instructor's subjective assessment of this student's performance concurred with this student's low ranking. The surprise was that this student was one of only eight students to score a grade of 'A' on this assignment (according to the grading rubric shown in Table 1). The ordering for this student's assignment is shown in Table 5.

This ordering reflects that the student had not fully thought out his approach to the problem beforehand as main was not completely defined before work began on the left_standing() function. It also raises questions as to his overall approach since he went back and forth between main and the left_standing() function. When the particular student learned of his 'A' grade on this assignment, he expressed surprise and stated that he was very unsure of how to create a program to achieve the desired goal.

Table 5: An Individual Student's Curious Orderings

Milestone	Flow of Control
Student Opened File	1
main() defined	2
left_standing() defined	4
left_standing() 1st call	7
left_standing() 2nd call	8
Input gathered	3
Print final result	6
Body of left_standing() started	5

Additional analysis showed that 16% of the students in Fall 2011 and 21% of the students in Spring 2012 addressed the body of the function left_standing() at steps four or below. The average grade of the Fall students was 61 and the average grade of the Spring students was 58.2 out of 100. In Fall, 60% of the students addressed the body of the function at steps 6 to 8, and in Spring, 43%. The average grade was 79 in the Fall and 78 in the Spring for those students. Clearly, students who follow the ordering suggested in Table 4 tend to do better than those who might follow a different ordering.

There are other interesting cases that have given additional insight into the reasoning of the students as they completed this assignment. The authors would like to create some template ordering patterns that reflect a good approach to this assignment and a poor approach to this assignment. If accomplished, the instructor could receive an automatically generated assessment of the likelihood that a student showed mastery of the concepts in question. This template could be used by the instructor to look deeper into the work of certain students and help identify deficiencies in learning, such as using a more random pattern of orderings in accomplishing assignments and insufficient numbers of students able to complete portions of an assignment.

7 Future Work

Outlined in this paper is an early implementation of a technique for recording additional data that students produce while completing a programming assignment. Currently, it can be a time intensive process to set up an assignment for capture using Google Docs, especially with large numbers of students. Additional work needs to be done to have a more automated means of setting up and collecting data from larger numbers of students.

Analysis of the data is currently done manually and requires a researcher to step though the intermediate versions of a student's code. The process could be improved by feeding these intermediate versions into a program that could assign the orderings automatically.

Work needs to be done to produce template patterns for more or less desirable orderings. To create templates, several assignments will need to be given to students and a comparison of orderings of experienced to novice students made to gain more insight into how orderings should be done. Certainly, much care must be taken when creating the assignment and selecting the milestones as there likely will be just a small subset of the milestones that are of interest in testing the mastery of a given concept.

8 References

[1] Theresa Beaubouef and John Mason. 2005. "Why the high attrition rate for computer science students: some thoughts and observations". *SIGCSE Bulletin* 37, 2 (June 2005), 103-106.

[2] Dawn McKinney and Leo F. Denton. 2004. "Houston, we have a problem: there's a leak in the CS1 affective oxygen tank". *SIGCSE Bulletin.* 36, 1 (March 2004), 236-239.

[3] Ma. Mercedes, T. Rodrigo, and Ryan S.J.D. Baker. 2009. "Coarse-grained detection of student frustration in an introductory programming course". In *Proceedings of the Fifth International Workshop on Computing Education Research Workshop* (ICER '09). ACM, New York, NY, USA, 75-80.

[4] H. Chad Lane and Kurt VanLehn. 2005. "Intention-based scoring: an approach to measuring success at solving the composition problem". *SIGCSE Bulletin* 37, 1 (February 2005), 373-377.

[5] João Paulo Barros. 2010. “Assessment and grading for CS1: towards a complete toolbox of criteria and techniques. In *Proceedings of the 10th Koli Calling International Conference on Computing Education Research* (Koli Calling '10). ACM, New York, NY, USA, 106-111.

[6] Anne Venables, Grace Tan, and Raymond Lister. 2009. “A closer look at tracing, explaining and code writing skills in the novice programmer”. In *Proceedings of the Fifth International Workshop on Computing Education Research Workshop* (ICER '09). ACM, New York, NY, USA, 117-128.

[7] Jeannette Wing. 2008. “Computational thinking and thinking about computing”. *Philosophical Transactions of the Royal Society,* 366, 3717-3725.

[8] Ljubomir Perković, Amber Settle, Sungsoon Hwang, and Joshua Jones. 2010. “A Framework for Computational Thinking Across the Curriculum”. In *Proceedings of the 15th Annual Conference on Innovation and Technology in Computer Science Education* (ITiCSE 2010). ACM, New York, NY, USA, 123-127.

A Study of Misconceptions and Missing Conceptions of Novice Java Programmers

Chiu-Liang Chen[1], Shun-Yin Cheng[2], and Janet Mei-Chuen Lin[3]
[1]Department of Information Management, National Taipei College of Business, Taipei, Taiwan
[2, 3]Graduate Institute of Information and Computer Education, National Taiwan Normal University, Taipei, Taiwan

ABSTRACT - *To enhance learning effectiveness in object-oriented programming courses, many researchers have tried to identify difficulties faced by students and the misconceptions they may harbor. As a further attempt along this line, we conducted one-on-one clinical interviews with 22 freshmen who were taking their first Java programming courses. The objective was to investigate not only which programming concepts/constructs most students had difficulties with, but why they found them difficult. We focused on fundamental OO concepts and Java programming constructs such as: (1) classes vs. objects, (2) static data members vs. constant data members, (3) constructors, (4) access modifiers, (5) syntax for method calls (e.g., for calling static vs. non-static methods, for calling methods in other classes vs. within the same class), (6) parameter passing, (7) method overloading, (8) inheritance, (9) polymorphism, (10) for-each vs. for loops, and (11) the use of standard Java libraries. Students' misconceptions and missing conceptions in each of these concepts/constructs are described in detail in this paper.*

Keywords: Java Programming, Missing Conceptions, Misconceptions, Greenfoot, Clinical Interview.

1. INTRODUCTION

Java programming is a required course for many computer science majors, but it is often found difficult by students. To better understand the difficulties encountered by novice Java programmers, many studies have tried to identify their misconceptions [3, 6, 11]. Whether Java instructors should adopt an object-early or object-late approach has also been widely discussed [e.g., 2, 11]. To support the object-early approach so that students can engage in learning OO concepts from the beginning, various pedagogical tools, such as DrJava [1], BlueJ [9], and Greenfoot [7], have been developed.

One common approach to identifying difficulties or misconceptions that students may have typically asks what they do or do not know. However, as Perkins and Martin [10] have pointed out, it is inappropriate to oversimplify knowledge as simply knowing or not knowing something. Students may have *fragile* knowledge about a specific programming concept, which results in programming errors rather than not knowing anything about it at all. Students' fragile knowledge of BASIC statements was then analyzed by Perkins and Martin [10].

One-on-one clinical interviews were conducted in this study to probe students' understating of Java programming constructs and fundamental object-oriented concepts. Students were observed closely as they solved a set of Java programming problems, each centered on a specific programming concept/construct. For example, a problem may require students to modify a Java program by declaring some of the data member(s) of a class to be *static*. Through close observation and interviews we intended to determine different levels of students' understanding about a concept/construct. For example, does a student know what a data member is? Can s/he describe the properties of a *static* data member? Can s/he correctly identify which data member(s) should be made *static* in a specific problem?

The remainder of this paper is divided into four sections. Section 2 discusses related work; Section 3 describes the research method and procedure; Section 4 presents our findings; and Section 5 is the conclusion.

2. RELATED WORK

2.1 Misconceptions of Novice Java Programmers

Eckerdal and Thuné [3] performed a phenomenographic analysis to identify different understandings of object-oriented concepts exhibited by students. They then applied variation theory to the results to pinpoint what students need to be able to discern in order to gain a "rich" understanding of those concepts. For the purpose of building a concept inventory of programming fundamentals, Kaczmarczyk, Petrick, East and Herman investigated students' misconceptions in a series of core CS1 topics [6]. Their investigation revealed four themes, with each theme containing several misconceptions. Three of those misconceptions were described in detail, including two misconceptions about memory models, and a misconception involving data assignment when primitives were declared. Ragonis and Ben-Ari [11] investigated difficulties encountered and conceptions built by novice object-oriented programmers. Their study identified 58 conceptions and difficulties, which were grouped into four primary categories: class vs. object, instantiation and constructors, simple vs. composed classes, and program flow.

2.2 The Object-Early Approach and the Greenfoot Programming Environment

Many previous studies have focused on how to teach introductory programming courses using Java, in particular whether an object-early approach should be adopted [e.g., 2, 11]. As opposed to an object-late approach, an object-early approach introduces classes and objects very early in the course and emphasizes core OO concepts from the beginning. Bruce [2] claimed that using an object-late approach to teach Java would do a disservice to students because students would learn a programming style that does not fit the language itself. He further suggested that instructors should use pedagogical tools such as BlueJ and DrJava to support object-first instruction. After teaching Java to high school students for an academic year, Ragonis and Ben-Ari also concluded that classes and objects should be introduced first, but they suggested using diagrams to enhance students' understanding of OO concepts [11].

The Greenfoot programming environment used in this study is yet another pedagogical environment for Java, like BlueJ and DrJava. In fact, it was created by the same research team that developed BlueJ. The design goals of Greenfoot were to make programming engaging, creative and satisfying for the students, and to actively help teachers in teaching important, universal programming concepts [7]. With Greenfoot, students can easily visualize important object-oriented concepts.

2.3 The Clinical Interview Research Method

Clinical interviews were originally used by Jean Piaget to reveal children's thinking processes [4]. Perkins & Martin employed the method to investigate students' difficulties in learning BASIC programming [10]. In their clinical interviews, an experimenter interacted with students as they worked, systematically providing help as needed progressing from general strategic *prompts* to specific *provides*. Based on the data collected from clinical interviews, students' fragile knowledge was classified into four types: (1) missing knowledge: knowledge that a student had either forgotten or never learned; (2) inert knowledge: knowledge that a student failed to retrieve but in fact possessed; (3) misplaced knowledge: knowledge suitable for some roles invaded occasions where it did not fit; and (4) conglomerated knowledge: students produced code that jammed together several disparate elements in a syntactically or semantically anomalous way.

3. METHOD

3.1 Procedure

A total of 84 students enrolled in two introductory Java programming courses offered by the Department of Information Management at National Taipei College of Business in Fall semester of 2010, which lasted 18 weeks from September 2010 to January 2011. There were three 50-minute class periods per week for each course. Students were offered a mixture of lectures and hands-on exercises that took place in a computer lab. Both courses were taught by the first author of this paper.

The midterm exam, which included a written test and a programming test, was administered in the 9th week, whereas the written test and the programming test of the final exam were given in the 16th and the 17th week respectively. Based on students' test scores, the instructor selected 22 low-achievers to participate in the one-on-one clinical interviews, which were conducted in the 18th week. The first two authors served as the interviewers.

During each interview session, the interviewee was closely observed as s/he worked on a set of assigned problems. S/he was allowed to take as much time as needed to solve each problem. Whenever the interviewee encountered an impasse and did not know how to proceed, the interviewer would intervene by providing a predetermined series of guidance, ranging progressively from asking questions, to providing general strategic prompts, and finally to giving specific syntax for a statement. Each interview session lasted from 45 minutes to 2 hours, depending on how much time a student needed to solve the assigned problems.

The content taught in the two courses was the same. It was adapted from [8] and covered three major parts:

- The Greenfoot programming environment: the class diagram for examining the classes used in the scenario and the inheritance relationship among them, the `Act` button and the `Run` button for execution control, the Greenfoot API, and how objects can be created and methods invoked interactively from the pop-up menu.

- Basic programming constructs: variables and constants, arithmetic, relational and logical operators, conditional statements, loops, arrays, method invocation, and packages.

- Object-oriented concepts: classes and objects, access modifiers, constructors, static class members, inheritance, polymorphism, method overloading, the `super` keyword, the this keyword, collections vs. the for each statement, and Java standard class library.

3.2 Data Collection

The data-collection instruments used in this study included the programming problems for students to solve during one-on-one clinical interviews, the guiding questions given by the interviewer to the students for each programming problem, and a screen-capture software tool. Detailed contents of these instruments are elaborated below.

3.2.1 The Programming Questions

The scenario used in this study was a simulation of number representation in binary using cards. It was derived from the Binary Numbers scenario created by Lenton and Brown [5]. Figure 1 and figure 2 show the main window and the class diagram of our revised Binary Number scenario respectively. As shown in Fig. 1, there are 5 cards on upper side of the scenario's world. Each card has two states: either shown or not shown. When a card is shown, its face up image is displayed and the binary digit direct below it should be 1. On the other hand, when a card is not shown, its back image is displayed and the binary digit right below it should be 0. Theses cards share a common back image, but each has its own unique face up image. The 5 cards starting from left to right contain 2^4, 2^3, 2^2, 2^1, 2^0 spades respectively on their face up images, and the number of spades on each card's face up image represents the associated value of that card. Total value of the face-up cards is shown to the right of the equal sign. For example, the total value of the three shown cards in Fig. 1 is 19, and its binary representation is 10011.

When the scenario is running, the user can click any card or binary digit to flip it. When a card switches its state, the associated binary digit changes correspondingly (and vice versa). Simultaneously, the total value for the new set of shown cards updates immediately.

The students were provided with a partially implemented version of our revised Binary Number scenario and required to work on 9 programming questions for the clinical interview. The questions as well as their related programming concepts are summarized in Table 1. As shown in Table 1, only question 6 asks for adding new functionality to the existing scenario, the other 8 questions ask for modifying code to more properly implement the scenario. For example, in question 1, students had to find out one data member in PokerCard/BNCard class that could be shared among all PokerCard/BNCard objects and declare it as static. Figure 3 shows a code segment of the PokerCard class, since all PokerCard objects share the same back image, the backImg data member (see line 6) should be declared as:

```
private static String backImg="card_back.png";
```

3.2.2 Guiding questions

The guidance series for all programming questions was proposed by the interviewers in advance. For each particular programming question, guidance was planned according to the sub goals (Table 1) that students should achieve for answering that question. There were two types of guidance: questioning and giving explanations. When students encountered problems, the interviewer first questioned to make sure if they understood related concepts. If they didn't, the interviewer then explained those concepts to the students. For example, guidance for the first sub goal of programming question 1 was:

Can you tell me where the data members are?

If the student had no idea how to answer the question, the interviewer then gave some further explanations, such as:

Generally speaking, a class is composed of method members and data members, and these are method members. Can you find out where the data members are?

If the student still had no idea, the interviewer then gave the exact answer:

These are the data members.

Figure 1. Main window of our revised Binary Number scenario for this study.

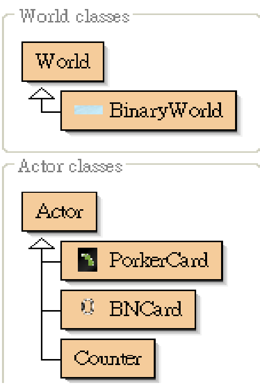


Figure 2. Class diagram of the partially implemented version of our revised Binary Number scenario

Table 1. Programming questions for the clinical interview.

Programming questions	Related Concepts	Sub goals for each question
Q1: Find a data member that is proper for declaring as static from the PokerCard or the BNCard class and declare it as static.	Static data members.	● Figure out where the data members are. ● Figure out properties of a static data member. ● Determine which data member is appropriate for declaring as static. ● Declare a data member as static.
Q2: Find a data member that is proper for declaring as constant from the PokerCard or the BNCard class and declare it as constant.	Constant data members.	● Figure out properties of a constant data member. ● Determine which data member is appropriate for declaring as constant. ● Declare a data member as constant.
Q3: The BNCard class already contains a constructor with two parameters in its method signature as follows: (int value, boolean isShown) ● Define another constructor with one parameter: (int value) ● Call the original constructor from the newly defined constructor. ● Modify the BinaryWorld class so that when instantiating the BNCard objects, the newly defined constructor will be invoked automatically instead of the original constructor.	● Properties of a constructor. ● Method overloading.	● Define a new constructor in a class. ● Call the original constructor from the newly defined constructor by using the this keyword. ● Modify the way of an argument passing in new BNCard statements of the BirnaryWorld class
Q4: Modify the BinaryWorld class: (1) Define a new method named populateWorld. (2) Remove the code defined in the constructor for instantiating all actor objects to the newly defined method. (3) Call the populateWorld method within the constructor.	Divide the program task into different modules.	● Define a new method. ● Identify related statements existing in the BinaryWorld class constructor and remove them into this newly defined method. ● Call this newly defined method from the BinaryWorld class constructor.
Q5: Modify the Counter class: When the space key is pressed: (1) Switch all cards' states to not shown. (2) Set all binary digits to 0. (3) Reset the total to 0.	● Get a collection (set of specific objects) ● Use for-each loop to process this collection.	● Determine which method this functionality should be defined in. ● Use if-else conditional statement to determine if the space key is pressed. ● Import a class from the Java class library. ● Read all PokerCard Objects. ● Use for-each loop to process all elements of a collection.
Q6: Apply inheritance concept to simplify code defined in both PokerCard class and BNCard class: (1) Define a class named Card that extends the Actor class. (2) Remove appropriate class members from both PokerCard class and BNCard class to the Card class. (3) Let both PokerCard class and BNCard class inherit the Card class.	● Apply the inheritance concept to promote code reuse. ● Define subclasses.	● Define a direct subclass of the Actor class named Card. ● Identify common functionalities among both PokerCard class and BNCard class. ● Define those common functionalities in Card class instead of in both PokerCard class and BNCard class. ● Let both PokerCard class and BNCard class extend the Card class.
Q7: Modify the BinaryWorld class: use only one superclass variable to reference two subclass objects successively.	Apply the polymorphism concept to promote program extensibility.	● Figure out meaning of the polymorphism concept. ● Declare a Card variable to substitute for both PokerCard variable and BNCard variable. ● Use the Card variable to reference PokerCard object and BNCard object successively.
Q8: Find a method that is proper for declaring as private from the PokerCard or BNCard class and declare it as private.	● Data hiding ● Access modifiers of class members	● Figure out what the access modifiers are. ● Figure out differences between public methods and private methods. ● Find out the exact method that could be declared as private.
Q9: The addObjects has been repeatedly called for 5 times, try to simplify the code by placing these method calls in a for loop or a while loop.	Use the loop constructs.	● Figure out syntax of while loop or for loop. ● Figure out how the value of the loop index can be used as arguments in the addObjects method call to determine objects' positions in the scenario world.

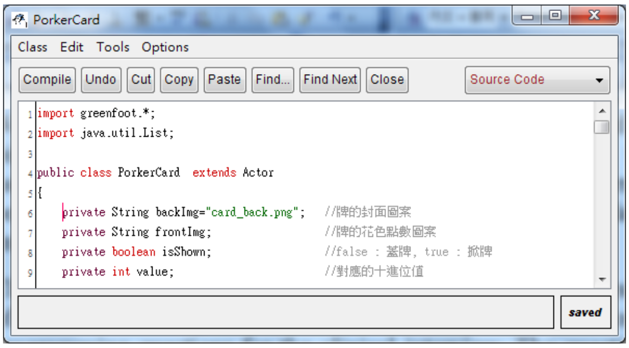


Figure 3. Code segment of the PokerCard class.

3.2.3 The screen-capture software package

A commercially available screen-capture software package and a microphone were installed to capture students' programming processes as well as all discussions between student and the experimenter. Both the programming processes and the audio data were transcribed for further analysis.

4. RESULTS

4.1 Misconceptions harbored by students

This study has unveiled the following misconceptions harbored by many students:

1. Static data members vs. constant data members: Confused the properties of static data members with constant data members (3/22, 14%).

2. Classes vs. Objects: Failed to recognize that classes were actually user-defined data types that could be used to define variables just like built-in data types such as integer and Boolean (8/22, 36%).

3. Constructors: (1) Erroneously thought that constructors of all classes defined in a project were invoked automatically when the project was opened (3/22, 14%); (2) Defined the constructor(s) of a class as its private member(s) when they should be public members instead (8/17, 47%). (3) Did not realize that constructors, as a special type of method, did not require a return type (11/22, 50%).

4. Arguments passing for method calls: Confused formal parameters with actual parameters (7/22, 31%). This is demonstrated by the code segment shown in Figure 4: code at line 16 is right for calling the constructor located at line 11 to line 14 while code at line 17 is not.

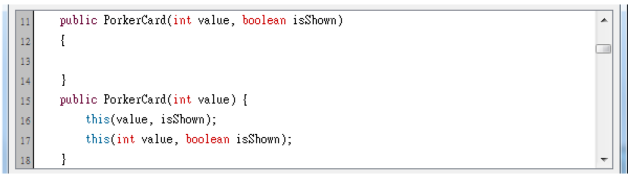


Figure 4. Mixed the use of formal parameters with actual parameters.

5. Polymorphism: (1) Mistakenly considered that declaring one superclass variable instead of two subclass variables means declaring two variables at the same line (3/22, 14%); (2) Mistakenly considered that referencing two subclass objects successively by the same superclass variable means substituting two superclass objects for two subclass objects (10/22, 45%). For example, the 7th question asked for applying polymorphism to modify related code defined in BinaryWorld class (as shown in Figure 5) such that both PokerCard object and BNCard object were referenced by the same Card variable (as shown in Figure 6). Some students mistakenly considered that just one superclass variable is defined in the following code:

```
private Card pkCard, bnCard;
```

Additionally, some mistakenly considered that polymorphism meant substituting superclass objects for subclass objects, with their code demonstrating as follows:

```
private Card pkCard;
private Card bnCard;
pkCard = new Card(…);
bnCard = new Card(…);
```

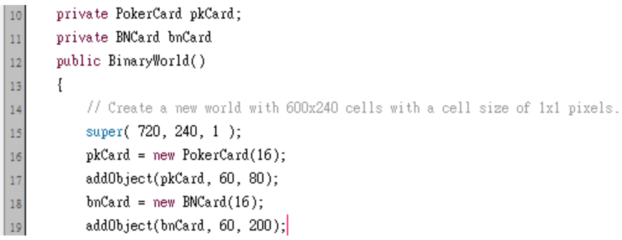


Figure 5. Reference subclass objects with their own variable types.

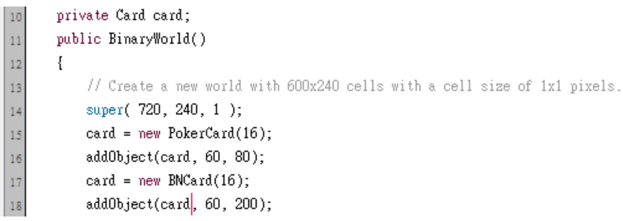


Figure 6. Reference subclass objects polymorphically.

6. Access modifiers of methods: Mistakenly considered that a private method could not invoke methods defined in other classes (2/17, 12%).

4.2 Missing Conceptions

Missing conceptions refer to knowledge that students have either not retained or never learned. In this study, students' missing conceptions have been identified as follows:

1. Could not recall the correct syntax of certain Java Programming constructs: (1) Static data members (12/21, 57%): Did not know the keyword for defining a static data member and/or the position for this keyword; (2) Constants (12/22, 55%): Did not know the keyword for defining a constant and/or the position for this keyword; (3) Instantiate Objects (12/22, 55%): Did not know the purposes of the new keyword were to instantiate an object and to initialize this object by passing arguments to the constructor; (4) Loop constructs (6/13, 46%): Did not how to control the execution flow by writing proper initialization expression, termination expression as well as increment expression for the for loop head; (5) Constructors (6/22, 27%): Did not know that a constructor is a special kind of method with the same name as the class it's defined in; (6) Method calls: Did not know syntax for calling non-static methods of other classes (12/17, 71%), syntax for calling static methods of other classes (10/19, 53%) and syntax for calling methods within the same class (4/22, 18%); (7) Inheritance (3/21, 14%): Did not know the syntax, i.e., using the extends keyword, for defining a class that inherit another class.

2. Did not understand how certain Java programming constructs were executed internally: (1) The for-each construct (17/17, 100%): Did not know the execution flow of using the for each construct to process all elements of a collection; (2) The for-loop construct (8/13, 62%): Did not know the control flow of the for loop construct; (3) Arguments vs. return value (11/22, 50%): Did not know how to provide additional information for a method to execute via arguments as well as how to get execution results from a method call via return values.

3. Did not realize certain object-oriented concepts: (1) Polymorphism (17/19, 89%): Did not understand the key meaning for polymorphism is that a super class variable can reference all its subclass objects; (2) Static data members (18/21, 86%): Did not know that whenever a data member is declared as static, only one copy of the data is maintained and shared for all objects of the class; (3) Method overloading (17/22, 77%): Did not know that multiple methods with the same name but with different formal parameters can be defined within the same class. Furthermore, some did not know how to pass proper arguments to call the exact overloaded method; (4) Constant (16/22, 73%): Did not know that the value of a constant data member cannot be changed after its initial value is given; (5) Constructors (15/22, 68%): Did not know that a constructor is always automatically executed whenever an object of this class is created, and that the constructor is usually used to initialize the states of that object; (6) Inheritance (11/19, 58%): Many students realized that a class can inherit and access functionalities and properties of its super classes, but still could not apply this concept to define proper class inheritance hierarchy; (7) Method's access modifiers (8/17, 47%): Did not know differences between both public methods and private methods; (8) Class data members (9/22, 41%): Could not figure out which data members are in a certain class.

4. Could point out certain features of specific concepts, but still could not apply them in problem solving: (1) Static data members: Could correctly describe static data members as data members that can be shared for all objects, but still were not able to find the exact data member that is proper for declaring as static; (2) Method's access modifiers: Could give descriptions that are somewhat correct for public methods vs. private methods such as: public methods are out of a class while private methods are inside a class, but still were not able to figure out the exact differences between both.

5. Not familiar with the application of standard Java libraries, such as collection (java.util.List).

5. CONCLUSIONS

In this study, we conducted one-on-one clinical interviews with 22 college freshmen to explore the difficulties encountered by novice Java programmers. In particular, typical missing conceptions and misconceptions that students may have about fundamental object- oriented concepts and certain programming constructs were identified. These findings pointed out specific topics that Java instructors may want to pay special attention to in order to minimize students' learning difficulties.

6. ACKNOWLEDGMENTS

This research has been funded by the National Science Council of Taiwan, the Republic of China, under the grant numbers NSC 97-2511-S-003-014-MY3.

7. REFERENCES

[1] Allen, E., Cartwright, R., and Stoler, B. DrJava: a lightweight pedagogic environment for Java. SIGCSE '02: Proceedings of the 33rd SIGCSE technical symposium on Computer science education

[2] Bruce, K. B. (2005). Controversy on how to teach CS 1: A discussion on the SIGCSE-members mailing list. ACM SIGCSE Bulletin, v.37 n.2, June 2005 [doi>10.1145/1083431.1083477]Greenroom - The Greenfoot Educators Community. Retrieved December 28, 2010 from http://greenroom.greenfoot.org/door.

[3] Eckerdal A., Thuné, M. (2005). Novice Java programmers' conceptions of "object" and "class", and variation theory, Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education, June 27-29, 2005, Caparica, Portugal [doi>10.1145/1067445.1067473]

[4] Ginsburg, H. P., Entering the child's mind : the clinical interview in psychological research and practice. November, 1997. Cambridge: Cambridge University Press

[5] Greenroom - The Greenfoot Educators Community. Retrieved December 28, 2010 from http://greenroom.greenfoot.org/door

[6] Kaczmarczyk L. C., Petrick E. R. , East J. P. , Herman G. L. (2010). Identifying student misconceptions of programming, Proceedings of the 41st ACM technical symposium on Computer science education, March 10-13, 2010, Milwaukee, Wisconsin, USA [doi>10.1145/1734263.1734299]

[7] Kölling, M. (2010). The Greenfoot Programming Environment. ACM Transactions on Computing Education, Vol. 10, No. 4, Article 14.

[8] Kölling, M. (2009). Introduction to Programming with Greenfoot. Upper Saddle River, New Jersey, USA: Pearson Education.

[9] Kölling, M., Quig, B., Patterson, A., and Rosenberg, J. 2003. The BlueJ system and its pedagogy. J. Comput. Sci. Educ. 13, 4.

[10] Perkins, D. N. and Martin, F. (1986). Fragile knowledge and neglected strategies in novice programmers. In E. Soloway & S. Lyengar (Eds.), Empirical Studies of Programmers, 1986, pp. 213-229.

[11] Ragonis, N., & Ben-Ari, M. (2005). A Long-Term Investigation of the Comprehension of OOP Concepts by Novices. Computer Science Education, 15(3), 203-221.

A Fast and Effective Design and Implementation of Online Programming Drills

Zhuhan Jiang, and Jiansheng Huang
School of Computing, Engineering and Mathematics, University of Western Sydney, Australia
z.jiang {j.huang}@uws.edu.au

Abstract - *We present a student drill system for the C++ programming language, supporting the online delivery and automatic marking. We propose to mainly use sufficient number of random numbers to evaluate symbolic expressions, statements, or functions, while applying sparingly pattern templates to match certain specific syntax or programming structures. A systematic scheme of emulation vs user code, along with the introduction of the "essential equality" of the numerical values, is shown to allow program marking to be readily formulated. By effectively detecting and barring potentially dangerous code segments in the student programs, the auto-marking can be safely achieved on a regular web server. Our online drill system will also monitor the execution of the student code so that it can be forced to terminate under undesirable circumstances like an infinite loop. The aim is to provide an effective and fast implementable C++ drill system that is sufficiently robust for the general students of programming fundamentals.*

Keywords: Programming Drills; Secure Code Execution; C++; Online Delivery

1 Introduction

With the advent of the new technologies in education, including online deliveries and knowledge-based intelligent training, work has already been done to automate as much student assessments as possible. For instance, different approaches have been developed to auto-assess short free text answers [1] or summary writing [2], or to achieve fairer assessment marks [3]. Similar efforts have also been made towards automatically marking computer programs in various different aspects, including checking the student program against the expected output [4], converting the program into an Equivalent Transformation functional language and having the program somewhat tested there [5], evaluating syntax structure for SQL [6] or decision tree [7] or graph similarity [8] for Java, and comparing statistically some of the design characteristics of the target Java programs [9]. However, these works typically try to mark the assignments against some selected *features* under often the program restriction or syntax limitation, without considering the full scaled security threats, and often not real-time. Also almost all the implemented cases are on the Java virtual machine, and little if any is done for the C++ language. Another major draw back is that the program correctness is largely based on an estimate without absolute certainty over a number of programming features. This is obviously due to the complexity of a full-fledged programming language, and the variety of variant ways to achieve the same programming goal. This paper thus concentrates on the basic parts of C++, the trustworthy marking, the fast and effective implementation, as well as the dependable security enforcement that can comfortably support the real-time online drills to any external users. In order to effectively and efficiently put such a programming drill in place, it is almost a must to have the drill or assessment designed and implemented with the auto-marking capability. A drill system with the automatic marking support will allow students to repeat their drill exercises or optionally do some additional ones should the need arise for the individual students, and may also assist students in best pacing their studies and progress.

How does one go about designing and implementing such a drill system for the programming? The most orthodox approach would probably be to write a new compiler for the targeted programming language so that it supports verification of a single statement or declaration. However, first of all this undertaking would in general be beyond the capacity of a typical programming instructor. Secondly, this new compiler would not be so much in the traditional sense as it now needs to acquire the capacity to process statements symbolically. This in a way also explains why there are no such facilities available in any of the compilers for the mainstream programming languages. In this work, we will concentrate on the C++ programming language as it has been used as the programming language for the programming subjects in our computing related courses. Although C++ is a language of large number of complex features, only the rudimentary features are required to illustrate the fundamentals of programming. And this is consistent with our design purpose for the corresponding drill system in that the drills will only make sense for the elementary features such as simple assignment statements, selection and loop structure, simple function design and calling, as well as the arrays and record structure. Although the drill can in principle be designed at any level, the rudimentary level is where the drills will benefit the students most and is what we shall concentrate on most.

This paper is organized as follows. We will first in section 2 propose a broad design for an online programming drill system for C++. We then in section 3 address a number of technical strategies for the security and other design aspects. Section 4 then puts everything together, develops the implemental interface, and illustrates the system with examples. Finally section 5 gives the conclusions.

2 Design for online programming drills

How to design a programming drill system that supports automatic marking and issues pertinent error messages or other coding tips? In this section, we will first describe the broad purpose and strategy, and then analyze the difficulties and challenges, and then find the corresponding solutions.

2.1 The broad methodology

In a typical case of asking a student to write a code segment, the student may be posed a question which describes what requirements are to be fulfilled. A simplistic example question could read as follows.

Sample Question: Suppose a rectangle has its width and height represented respectively by the int variable **width** *and* **height***. Write an expression in C++ to represent the area of the rectangle.*

The basic strategy is to build a complete program by suitably combining the header, the user input, and the footer. If the header is kept in *header.cpp*, the footer in *footer.cpp* and the user input in *user.cpp*, then the complete program *combined.cpp* is exactly of the form shown in Fig. 1. More precisely, the general strategy is to design the header and footer so that the compilation of *combined.cpp* by a compiler, and the subsequent execution, will be sufficient to determine if the code in *user.cpp* provided by a student fulfills the coding requirement. Such a determination will typically rest upon the compilation errors, the execution exit code, as well as the screen output. For our sample question as an illustration, we can put into *header.cpp*

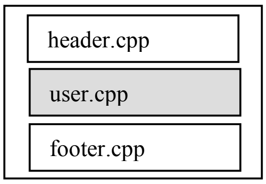


Fig. 1: Combined.cpp

```
#include<iostream>
using namespace std;
double randd( ) {  // a typical support function
  // here is the code to generate a random number
}
int user(int width, int height) {
  return ( //  user code starts below
```

and put into *footer.cpp*

```
        ); // user code ends above
}
int main( ) {
    const int N=10;
    for(int n=0; n<N; n++) {
      doulbe w=randd( ), h=randd( ), a=w*h;
      if(a != user(w,h)) return 1;  // this line is improvable
    }
    return 0;
}
```

This way, if the user code contains an expression that is equivalent to (or exactly the same as) **width*height** then the combined source code constructed via Fig. 1 can be compiled and executed successfully with exit code 0 and without any output to the *stdout* device. If the user-entered expression doesn't evaluate to the expected value, then the exit code will be 1. The output to *stdout* can also be collected, and the drill system may decide whether the unexpected output should be flagged as an error or it will be simply ignored. Since the programming language and the compiler for C++ are not designed for symbolic evaluations, we utilize "enough" random samples to test the validity of the "symbolic" expression.

It is important to note here that the user code is always enclosed into a separate function **user**(), and because we will disable the use of **exit**() function for the user code, the combined program will never terminate directly from the user code. Also it can be tempting at times to use template to symbolically match the expression pattern. However this is in general less ideal as there are often infinitely many forms of expression that evaluate logically to the same value. For robustness against rounding errors, we will also eventually replace the equality (**==** or **!=**) between 2 values of *double* type by a function **equal**() in subsection 3.3.

2.2 Flagging compiler errors and logical errors

When there are syntax errors in the user's code for a drill, the system should list and highlight them, and should also allow the erroneous code to be tested or debugged in a normal compiler IDE. However, the code supplied back to the student for fixing should in general not take the form engineered for the *combined.cpp* as in the above. Instead, we will use a plainer version of header and footer to construct a simpler combined program, *combined_pub.cpp*, to indicate the syntax or logical errors. For the above given sample question, the *combined_pub.cpp* can for instance take the form

```
#include<iostream>
using namespace std;
int main( ) {
    double width=2.3, height=4.5;
    double area = ( // user code starts below .
        //  e.g.   width * height
                    ); // user code ends above
   cout  << area << endl;
```

```
    return 0;
}
```

where the shaded section of code represents the code entered by the user. This is to ensure that the source program given back to a student to debug is as simple as possible, containing no programming structure or features other than those that are absolutely necessary. Hence it is the compiler's error messages from processing this *combined_pub.cpp* that are actually returned back to the student user. Not releasing the original *complete.cpp* to the users also makes it hardly possible for anyone to "cheat" the answers.

If the user code is syntactically correct but doesn't complete the assigned task properly, then it is considered to have logical errors. Logical errors can be detected by checking if certain variables carry the correct values at the appropriate points. If not, the program will abort with a non-zero exit code. Alternatively or concurrently, the program can generate certain output to the *stdout* device. The drill system will trap the output for *stdout* and then compare it with the anticipated output. If they differ beyond the tolerable level depending on the individual drill questions, then a logical error is also discovered. For instance, if a drill question expects the student code to output a phrase verbatim, then the student code is flawed if its output doesn't exactly match the expected.

2.3 Main security considerations

Because the drill system is expected to be delivered on the Internet and students' risky or erroneous code is to be executed as it is, security constitutes a crucial part for the design of the drill system. The main security issues are as follows:

i) a poor or logically incorrect code could leave a program executing forever without termination, and could generate massive or infinite amount of screen output;

ii) the student code could activate other programs on the server side, leading to information theft, resource waste or even data sabotage;

iii) concurrency problems such as the potential concurrent executions for different questions for the same student user;

iv) authentication of the student users and their access status for different drills, as certain status may allow them to repeat the drills many times or view the possible solutions.

These security issues will be considered in full in the next section. Broadly speaking, however, for i) we will monitor the execution of the user program and terminate the execution if it takes an unusually long period of time, and we will only capture the initial portion of the screen output up to a reasonable amount. For ii), we will disable some language features that would allow programs to do potentially damaging things. For iii), we will allocate dedicated benchmark area for each student user, and each user will only be allowed just a single session at any time. Finally for iv) we will enforce a password checking on the users for the drill system, and will schedule the access mode according to the user status and the access date and time.

2.4 Workflow of the drill system

Having had an overview of the broad methodology in the previous subsection, we are now ready to explore the full structure of the proposed drill system. The most essential procedure consists of displaying a question, getting a user code segment, processing and executing the code, analysing syntax and logical errors, supplying back options to view or debug the errors if any, updating the mark if pertinent, and returning control back to the original page. This workflow can be summarised in the following Fig. 2.

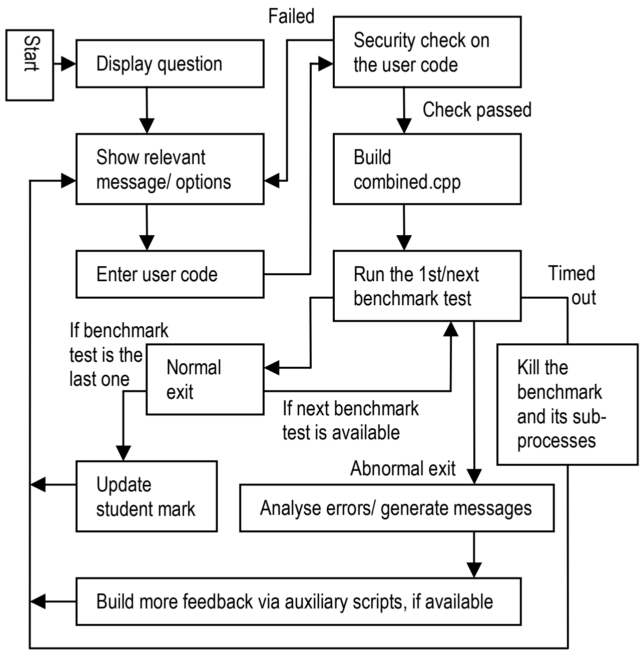


Fig. 2: Workflow for the drill system.

We note that in Fig. 2, the step to run a benchmark test is repeatedly monitored by another process for its termination. In other words, a benchmark test is typically written in the form of a script file (in PERL or SH), and the execution of this script will be regularly checked to see if it has terminated or not. If it is not terminated within a given period of time, say 4 seconds, it will be deemed that the user code contains an illegal infinite loop and the execution of the benchmark script will be terminated forcefully, implying also that the benchmark has failed. The exit code of the benchmark script will indicate success (normal exit) for 0 and different form of failure for a nonzero value. For a different drill question, there may be different number of benchmark tests associated with it, typically about 2-4 tests, and usually the first benchmark test is to see if the program can compile successfully. If one by one all the benchmark tests have been successfully passed, then the user code has fulfilled the requirements set out in the question, and the procedure will update the student mark and complete the task. If the *k*-th benchmark test is the first test that timed out or exited with a non-zero exit code, then the control returns to the question page for the timeout case, and otherwise the corresponding *k*-th analyzer will be executed to generate a suitable error message or provide other form of feedback. Since the 1st benchmark test is almost exclusively dedicated to compiling the program, the corresponding analyzer will produce webpage buttons for the user to either view the full highlighted error messages generated by the standard compiler or to activate a compiler IDE with the complete program loaded automatically so that the user can debug it within the IDE.

Hence for each drill question, there are *N* groups of scripts *benachmark-k.sh* and *analyzer-k.sh* for *k*=1, .., *N*, where *N* may vary from question to question. There are optional scripts *auxiliary-k.sh* which, if exist for a particular *k*, will be additionally executed to build more varieties of feedback to the user. There are furthermore files *header.cpp* and *footer.cpp* for the construction of the file *combined.cpp*, and also the files *header-pub.cpp* and *footer-pub.cpp* to construct the file *combined-pub.cpp* which is to be returned back to the user to illustrate the syntax errors.

3 Individual system design strategies

The benchmark strategies and the follow-up analyzers are very much question-specific. In this section, therefore, we will look into a number of typical techniques to support the drill questions, to fine-tune the system design, and to develop the corresponding benchmark strategies.

3.1 Prevention of dangerous code

Since a user-entered code can practically contain anything, it can be very dangerous to execute such a program after compiling it. Even though it's almost impossible to fully understand the semantics of a foreign code, we can nonetheless identify areas of code vulnerability so that precautions can be taken to prevent the execution of the potentially dangerous code. Let us first enumerate the possible main security breaches when compiling and executing an unknown piece of C++ code.

i) the **system**() function in the source code will allow external programs to be executed;

ii) library functions such as **execv**() and **execve**() can also be called to execute external programs;

iii) *#define* directive may be used to mask functions like **system**(); likewise the line continuing character (the trailing "\") may also have this effect;

iv) some system libraries may give the programs much more power than those the students actually need;

v) reading or writing a file by the user program can also secretly access or modify the data stored on the server;

vi) infinite loops, and/or massive screen output due to them;

vii) the system configuration for the web server, and the concurrency control on the use of the testing venue;

For the first case (i), we search the user code for the pattern of "<b>**system**<w>**(**", where <b> and <w> denote the word boundary and 0 or more white spaces respectively. In order to make the template search not to miss out any camouflaged instances of **system**( ), we disable the *#define* and *#undef* directives within the user-entered code and disallow all the header files except for <**iostream**>, <**iomanip**> and <**cmath**>. This is done by searching the portion of the user code for the *#define* and *#undef* directives and for the *#include* directive for any of the above mentioned "risky" header files, and the program will be flagged as "errors" if any of these dangerous constructs are found. The safety is thus ensured because such programs will not be executed. We hasten to add that the small feature reduction of the programming language in this way will not affect the drill purpose as the latter is meant to be relatively elementary, and those advanced libraries disallowed by the capacity reduction of the language are not expected to be learned through such drills. In order to apply the template matching more effectively for this purpose, we first pre-process the user code via "**g++ -E –xc++**" so that all the comments are removed, line breaks are joined, and statements are standardized or "cleaned up". We can also easily filter out all the string literals so that the **system**() that appears in string literals won't be mistaken for the dangerous code. For easier implementation, we also disable the use of **exit**() in a similar manner as **system**(), although this is not necessary as the testing results such as the exit code can be passed back via a dedicated file as well.

For case (ii), we simply disallow the use of the header files that prototype the library functions **execv** and **execve** and the like. Cases (iii) and (iv) are essentially already covered in the discussion for case (i). As for case (v), even though it may be feasible to set files access permissions more accurately and stringently, it is much easier to just disallow the header file for the file processing. This again will not much impact on the scope of the drills, as for instance our first programming fundamental unit doesn't include the explicit use of files as the learning outcome. Another very effective way of reducing the risks of executing the dangerous code is to simply limit the size of the user code to a reasonably small amount so that there is not much capacity for anyone to engineer a piece of sophisticated security-breaching code.

For case vi), each benchmark test, involving potentially the execution of the student program, will be given a fixed amount of time to complete, as we already mentioned briefly in subsection 2.4. If the test doesn't complete itself within this period due to a potential infinite loop, then a simplified way to terminate it in *bash* can be

```
sh ${benchmarkProgram}.sh "${executionDirPath}" &
PID=$! ; sleep ${amount} ;  kill -9 $PID ;  wait $PID
```

and the exit code immediately after this can be used to determined if the test **${benchmarkProgram}.sh** is actually killed, and if yes, further cleaning-up of the sub-processes created within the test will also be conducted. As for controlling the output to the *stdout*, the output is simply put through a text filter which only lets pass the initial portion up to a specified given amount. This will thus stop a student user from accidentally wasting a large storage space on the server.

For case vii), we propose to use Linux for the operating system, Apache for the web server, and *cgiwrap* for the execution of the web scripts. This allows any single *standard* user on a Linux machine to serve all the drills, making such an implementation fast, effective and hassle-free, and easily developed and maintained by the unit instructor alone. We note that if one wishes to enable file IO on top of *stdin* and *stdout*, one can still essentially do so within this system configuration. A simple way to achieve this is to add into the *header.cpp*

```
char* validate(char* f) {
 if(strstr(f,"..")!=NULL
   // or matches any of the designated files like header.cpp
   ) exit (9); else return f;
}
#define IFS(x,y) ifstream x(validate(y))
#define OFS(x,y) ofstream x(validate(y))
```

and then use **ISF**(*handle*, *filename*) to set up the file input, and likewise for the file output. The student code will first be searched for the explicit appearance of *ifstream* or *ofstream*. The code won't be executed if any of them is found, and the student will be prompted to use **IFS** etc instead. The macros like **IFS** will ensure that all the files opened for read or write are within the current directory or its subdirectory, and are not any of those dedicated files used by the drill system itself. Hence no security breaches will occur via the file IO. The exit code 9 here is allocated to flag the insecure files.

If another standard user account is also available on the same Linux machine to support the drill system, then one may choose to relax somewhat the restrictions on the student code and leave the code of some "risks" to be executed by this user, reducing but not completely removing the potential impact of the illegal program operations. If the web server sits on a dedicated Linux machine, then it is possible to create a separate account for each registered student, and the code will be executed there. However, these two alternatives are still not completely impact-free if one allows file IO to be directly available rather than via the safer macros like **ISF** in the above.

For the concurrency control on the use of the testing venues, we allocate a different subdirectory to each different student user, at which temporary files or testing results will be stored. To avoid the same user from using the same subdirectory concurrently for two different drill questions, potentially overriding each other's files, we will in principle allow only one valid session for each user at any time. Since it is rare for anyone to actually run multiple drill sessions concurrently, our design is inclined more towards the ease of use. When a drill is started, it will generate a random (drill) session ID and have it stored as an ownership label for the current session, overriding the original value there. In other words, a new session always obliterates the older one. When an old session proceeds further, it will detect that its session ID has been superseded by another session because it no longer matches with the stored ownership label, and will issue a warning to indicate this to the user. At this stage, the user may choose to accept the expiry of this old session by closing the browser or doing nothing, or may choose to be treated as a newer session to supersede all the others by simply reloading the drill page again.

3.2 Using built-in support of the language

C++ has included some support to identify the data type of a given variable, and this is done through the *typeid* function in the <**typeinfo**> header file. For example, if a student is expected to enter just **int count;** to illustrate he understands the declaration of an *int* variable, then we can declare a dummy variable *count_v* beforehand and then just execute

```
if(typeid(count)!=typeid(count_v)) return 1;
```

to indicate if *count* has the expected data type. Similarly, if one defines beforehand **char x[5];** then **if(typeid(x))!=typeid(y)) return 1;** will indicate that **y** has not been declared as a same type as **x** including the implied array size.

To ensure that the user-entered code can be repeatedly utilized for several different tests and that it will be kept independent from the rest of the support code as much as possible, we typically enclose the user code inside a user function **user**() as is the example case in subsection 2.1. To check if the user code is correct, one can set up a parallel function **emulate**() which is typically prototyped in exact the same way as the **user**() function so that for each set of (randomly or not) generated test data it will be verified if **user**() and **emulate**() return the same result. If not, logical computational or design errors will be flagged for the user code. However, if the outcome of the user code involves outputting to the *stdout*, then **emulate**() will do it similarly except that its output will be written to a designated file. After the output of **user**() is piped into a different file, these two files can be compared to see if they contain the same content, identically or up to white spaces and blank lines depending on the individual drill questions. For instance, if the student code is expected to achieve the effect of **cout << count;** , then the corresponding program component could read

```
ofstream targetfile;
void emulate(int count) { targetfile << count; }
int user(int count) {
  int count_passed=count;        // user code starts below
  cout << count;                 // user code
  return count==count_passed; // user code ends above line
}
int main() {
  const int N=10;            // number of random tests
  targetfile.open("target.out");
  for(int n=0; n<N; n++) {
    int m=rand();
    emulate(m);             // output to the target file
    if(!user(m)) return 1;          // output to stdout
  }
 return 0;
}
```

Here we optionally added an additional requirement that variable *count* doesn't get its value changed by the user code. We note that one of the *benchmark-k.sh* will pipe the *stdout* of the above program and compare it with the content of the *target.out* generated dynamically.

If the student code is expected to read from *stdin*, then the input for the *k*-th benchmark will be statically stored in the file *input-k.txt*, or dynamically generated into it. When the benchmark controller is to ready to execute *benchmark-k.sh*, it first checks if a generator program *generator-k.sh* exists, if yes, it will be executed and its output goes into the file *input-k.txt*. Then the controller checks if *input-k.txt* exists, if yes, its content will be piped into *benchmark-k.sh*. Otherwise the *k*-th benchmark program will be executed without any *stdin* input.

3.3 Rounding errors and structure templates

When comparing the numerical results of the user code with that of expected emulated results, rounding errors generic to the computer hardware or compiler can mislead people when interpreting the results. Hence we need a mechanism that can equate two values within sensible range of errors. For this purpose, we first introduce a function **dIncrement**() which will increase a given value of *double* type by the smallest possible amount that is still able to actually change the value. One such function can be implemented as

```
double dIncrement(double v) {
  int exponent;
  double mantissa = frexp(v, &exponent);
  if(!mantissa) return DBL_MIN;
  mantissa += DBL_EPSILON/2.0f;
  return ldexp(mantissa, exponent);
}
```

for the value of *double* type while it can be similarly implemented for the *float* type as well. We will then use the function **equal**() via

```
bool equal(double x, double y) {
  double xplus=dIncrement(x), yplus=dIncrement(y);
  return abs(x-y)<=abs(xplus-x)+abs(yplus-y);
}
```

to check if two values *x* and *y* of *double* type are to be considered "equal" or not within our benchmark tests. Suppose a user function is to emulate

```
double emulate(double v) { return v>10.0? 1.0 : 0; }
```

then the benchmarking could just test if **equal(emulate(s), user(s))** for a number of random values, and also test for those close to the boundary values as in

```
double v=10.0, plus=dIncrement(v), minus=v-(v-plus);
if ( emulate(d)!=user(d) || emulate(plus)!=user(plus) ||
   emulate(minu)!=user(minus) )
  return 1; // logical error encountered
```

Although for most simple statements in a program there is not much for a student to circumvent, for statements like a loop based pure numerical calculation it is possible for a user to simply directly enter the resulting value without utilizing the intended programming structure like a *while* or *for* loop. Hence in the later benchmark tests for a given drill question, we also make use of the template matching to ensure that the intended structure is indeed present in the user's code. For instance, if a question asks a student to code a *do-while* loop in C++ to sum up say $1^2+2^2+\ldots+50^2$, then the following (single line) benchmark script

```
g++ -E user.cpp 2>/dev/null | grep -v -e "^#"   \
  | perl -e 'undef $/;$_=<>;exit !(/\bdo\b.*\bwhile\b/s);'
```

first removes C++ comments and then checks if a *do-while* structure exists within the valid statements in *user.cpp*. An error will thus be flagged if the exit code is not 0. If for the same exercise we wish to enforce a *for* loop, then we just need to replace the last part in the above script by

```
perl -e 'undef $/;$_=<>;exit !(/\bfor\s*\([^;]*;[^;]*;[^;]*\)/s);'
```

4 Implementation and user interface

The proposed Programming Drill System is implemented for the first year first semester undergraduate unit called programming fundamentals. It allows a certain amount of marks to be associated with such drills. Once passing the deadline for a particular group of drill questions, then the drills can be re-opened again for optional additional practice, and this time, the sample solutions will be made available online real-time as well.

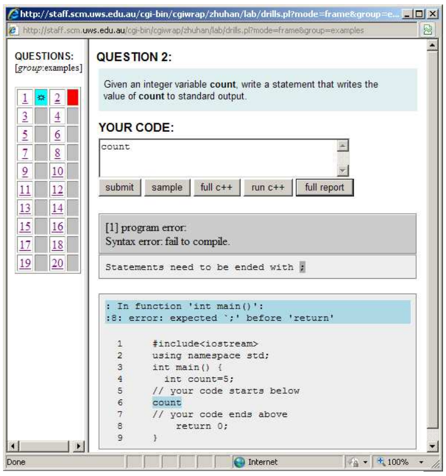


Fig. 3: Web interface for the drill.

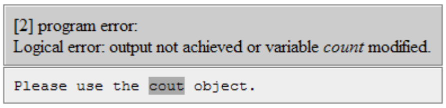


Fig. 4: Logical error is flagged.

For each drill question, each student is only recorded for whether the question has been attempted and whether one of the attempts has given a correct answer. In other words, once a question has been answered correctly, a student can still experiment with different answers without ever losing any marks there. If a student's code causes syntactical errors, then he has the option to see the full error report given by the compiler, on top of the drill system's own short error message. The student can also choose to view a full C++ program, which includes not only the student's own code but also those necessary header and footer to make a complete C++ program. The "run C++" button will even allow the student to fire up a C++ IDE with the full program being loaded into it automatically so that the student can debug his program within the IDE immediately. Fig. 3 shows the interface for drilling the exercises of group *examples*. Based on the use AJAX technology, the 1st shaded box below the buttons contains the error messages generated by *benchmark-k.sh*, the next box contains additional tip generated by *analysis-k.sh*, and the last box contains the full error report by the compiler (after the "full report" button is clicked). The highlighted text is for the actual error messages and the user code. For illustrational purpose, we deliberately left all the buttons on for the Fig. 3. In practice we typically only show the relevant buttons immediately after the user code is submitted, and some of the buttons such as "full report" will be hidden once it's clicked, until a new submission is made. If the student code entered for this particular exercise is "**count;**", then this program actually has no syntactical errors, and thus the button "full report" won't appear. Instead, the logical error will be displayed, see Fig. 4. For the privileged users, clicking "sample" will display the complete combined C++ program rather than just the sample portion for *user.cpp*.

5 Conclusions

We have proposed a fast and effective design and implementation of an automatic programming training system for the programming fundamentals in C++. We have investigated a number of thorny security issues and methodological strategies to resolve the design and implementation challenges. Our proposed and developed drill system will greatly facilitate the teaching and learning of the basic programming concepts and techniques, and will be particularly helpful to those relatively weaker students who would otherwise struggle to find enough "semi-supervised" exercises to gain better understanding of the programming principles. It can also be an indispensable tool to vocational students or students on a self-study program.

6 References

[1] S. G. Pulman, and J. Z. Sukkarieh, "Automatic short answer marking", Proceedings of the 2nd Workshop on Building Educational Applications Using NLP, June 2005, pp. 9-16.

[2] Y. He, S. C. Hui, and T. T. Quan, "Automatic summary assessment for intelligent tutoring systems", Computer & Education, vol. 53, 2009, pp. 890-899.

[3] Z. Jiang, and J. Huang, "The mathematical structure of a bias-minimized assessment algorithm", WSEAS Transactions on Systems, vol. 10, no. 11, 2011, pp. 387-397.

[4] H. Suleman, "Automatic marking with Sakai", SAICSIT '08: Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology, Oct. 2008, pp. 229-236.

[5] H. Koike, K. Akama, C. Ishikawa, N. Nakamura, and S. Watanabe, "Towards a software development model for automatic marking software", SIGUCCS'07: Proceedings of the 35th annual ACM SIGUCCS fall conference, Oct 2007, pp. 190-193.

[6] Y. Chen, and H. Chen, "Research of automatic marking on SQL server skill assessment based on XML", International Conference on Web Information Systems and Mining, 2010, pp. 8-12.

[7] D. Kalles, and A. Pagagelis, "Programming drills with a decision trees workbench", Adapting Information and Communication Technologies for Effective Education, by L. A. Tomei (edr), IGI, 2007, pp. 108-120.

[8] K. A. Naude, J. H. Greyling, and D. Vogts, "Marking student programs using graph similarity", Computer & and Education, vol. 54, 2010, pp. 545-561.

[9] M. Ahmadzadeh, S. Namvar, and M. Soltani, "JavaMarker: a marking system for Java programs", International Journal of Computer Applications, vol. 20, no. 2, 2011, pp. 15-20.

A hybrid structured methodology for developing computer-based industrial computer systems

J. Albaladejo, C. Domínguez, L. Escolar, A. Perles, J.M. Martínez, H. Hassan, J.V. Capella
Department Of Computer Engineering
Universidad Politécnica de Valencia, Valencia Spain

Abstract - *Industrial informatics systems are usually complex systems. To build them it is necessary to follow a software developing methodology. The methodologies available nowadays come from the experience of the software developer's community over the last 50 years. New programming languages, programming paradigms, models and representations have been showing up and evolving over that time. In this article we present the methodology that we recommend electrical and electronic engineers to develop computer based control systems. The methods and techniques that we select and recommend here come from our own experience with software developing teams in the last 15 years. Those teams comprise electrical or electronic engineers who needs solving control problems in their specific areas instead of software engineers, so the methodology is adapted to their actual skills in programming.*

Keywords: Software development, informatics systems, top-down and bottom-up methodologies, problem-based methodology, data acquisition.

1. Introduction

Industrial informatics systems are usually complex systems. Its complexity is due, firstly, cause of the nature of the problems that they want to solve "complex systems" with interrelated variables. This leads to the necessity of representation models and control processes that aren't trivial. Additionally, the technological elements necessary to execute the control processes are also complex systems ([1],[2]). To develop an industrial computer system with sufficient guarantees of success is important to follow a validated methodology to keep bounded the complexity of the problem in each of the phases of the project: planning, design, development, validation, deployment and exploitation. The experience of the software developers' community over the past 50 years has identified important issues to be taken into account in industrial informatics projects. The recommendations are similar to those proposed for problem solving in other areas, although adapted to the particularities of industrial informatics systems.

Mainly, the basic and permanent recommendation consists of dividing the complicated problems in smaller or simpler ones that can be addressed separately, based on the basics principles of the structured programming ([3],[4],[5]]). Since people are still the key element in the project organization and their attention capacities are limited, it is necessary to make an abstraction process, keeping apart the irrelevant parts of a problem and focusing on the important things.

We present in this article some design methods and programming techniques that allow this abstraction. We explain how that methods and techniques contribute to the success of the project. The design and development recommendations proposed in this article are based on different concepts such as: top/down and bottom/up design approaches, modeling the problem before start coding, modular decomposition of the program, and minimizing interfaces between the modules ([6],[7],[8]). These all proposals are of interest to ensure the success of the project:

(a) To mix different design approaches has advantages if tailored specifically to each part of the project. For example, top-down approach is recommended in the design of the tasks. The main control task can be decomposed in a set of more specific collaborating tasks: "observe the actual situation", "desire a new situation", "decide the action", "act" and "report to user". The bottom-up approach is recommended however for developing some of the basic utilities and services, such as the input / output operations to sense and manipulate the external process, or the reading / writing operations on the blackboard system.

(b) Formal modeling during the design phase has advantages, because it allows a systematic codification from the model and then it is also easy to verify the program using the model. For example to formally design a state machine and then to easily code it using a "switch" sentence in C language.

(c) The modular decomposition based on divide and conquer algorithms has advantages too, because it allows address a complex problem into simpler parts by abstraction and concealment. You can delegate the development of each part to different team members, and it accelerates the edit-compile cycle cause of the smaller translation units.

(d) The minimization of interfaces between the modules, through the "common data module", has advantages, because it improves the consistency of information, avoiding duplication of variables, and simplifies the use of the variables by using a common interface. Additionally a centralized policy management and protection of the variables can be developed with the use of the blackboard access functions.

In order to illustrate the recommendations, we use an example based on the control problem of a greenhouse. In the following paragraphs we show the used methodology. In the section 2 we explain the requirements and the specifications for the project. In section 3, the methodology based in a modular decomposition and in section 4 the conclusions.

2. Requirements and specifications of the example guide

To address a hybrid hierarchical design methodology, we use the approach "Problem Based Learning", using a guided example. The various stages of designing an industrial software project are explained using this example. The objective is to plan how to achieve the proposed industrial project. The process in a synthetic way is as follows.

The project is analyzed from two points of view (or roles): The role of the "client" (or employer) is assumed by the project manager; and the role of designers group is taken by the engineers. After the presentation, the manager establishes and sets the particular requirements for the project.

The project is based on a general microprocessor. Considering this platform as a reference system, we start analyzing (or identifying):

- The inputs (I) and outputs (O) of the industrial process. This is done by using sensors that report about the actual situation. And the actuators, which can modify the status of these magnitudes, being activated by digital and/or analog signals.
- The inputs (I) and outputs (O) of the user. The inputs objectives to control the process. And all the information that the user need to observe. That is, defining the user interface.

Finally, the project manager defines the necessary tasks to control the process and the control strategy. In the example ("the greenhouse") a sampled control system is developed. The project should be documented with a set of technical reports (problem perception and requirements, design and implementation, and validation). Additionally the dead line for the project is established as an important requirement.

The requirements for the greenhouse are set in table 1.

In this article, for the implementation of the project we will briefly discuss the guide lines, analyzing just only the temperature control loop. Then, in the following sections you will see how we apply the methodology through a modular decomposition, and then, we will show the conclusions of this methodology.

3. The Proposed Hybrid Methodology

The divide and conquer methodology splits a problem in several sub-problems of the same type, until these become simple enough to be solved directly. This is known as modular decomposition. Every project has a particular modular structure. In order to explain our methodology, in this article we attempt to abstract to the minimum number of modules that we need in a project. In consequence, we synthesize and analyzes in the following sections the modules: common data, process interface, the task sequencing, the graphics user interface (GUI), and the control module. In later stages of the design, it's possible to add other different sub-modules, to accomplish more specific functions.

For ease reading and coding the program, we concatenate the name of each function and the name of the module. For example "DataWriteVariable" will correspond to the "WriteVariable" function defined in the "Data" module, and so on.

Table 1: The greenhouse requirements and specifications
A computer system must control a greenhouse process so that the relative humidity (RHint) and the temperature inside (Tint) remains as close as possible to the desired values, all times. To do this, inside the greenhouse is equipped with: a heating system, a drip system, a ventilation system, and an aeration system (windows). To measure the temperature there are two temperature sensors which measures the external (Text), and internal (Tint) temperatures. To measure the relative humidity (HR) inside greenhouse there is a humidity sensor that measures it (RHint). *The specifications of the sensors that are available for this application are:* *Temperature sensors (thermocouples) are analog, with a linear behavior in the range of [-15.0 to 60.0]°C. A voltage of 0.0V corresponds to a temperature of -10.0 °C, and 3.5V correspond to 50.0 ºC.* *The humidity sensor is analog with a linear behavior. This sensor reading corresponds to the following voltage values -2.5V voltage equivalent to 0.0% of the relative humidity (RH); while a voltage of 3.0V corresponds to 100.0% RH.* *The ventilation system (fan) is analog, with the following characteristics. When the fan is powered with 1.5V is turned off, and when 3.5V is 100% of its power. It can be considered the fan power behaves linearly.* *For the actuators systems, we have the following characteristics. The heating system is digital TTL compatible: 0 V, switches ON, and 5V OFF.* *The aeration system (windows) is digital. This system is driven by a step motor with position control that has three inputs (M2M1M0), operating in inverse logic. The value "$M_2M_1M_0$ = 000", fully open (30 degrees), and the value "$M_2M_1M_0$ = 111" completely closed (0 degrees). It's possible to program all the intermediate steps for the window, between [0 to 30] degrees. And finally, the irrigation system is digital TTL compatible, with 0V the sprinkler is closed, and 5 V opens.* *The irrigation system can be stopped manually by the user pressing an emergency button. A TTL digital input is used for this purpose. The desired operating conditions should be chosen by the user. The values of temperature and relative humidity should be within optimum range.*

3.1 Design of data module

The role of the data module, serves to minimize the interfaces (cross-dependences) between the different modules of the system that need to share data. The data module it's like a blackboard where the program read and writes the variables that other modules use.

From the standpoint of coding, to implement the data model, we need extract and define the variables that represent the controlled system behavior. The data module allows us represent the abstraction and concealment of information between the modules. This module publishes access functions (read and write) to hidden variables that other modules can use. With these functions it's impossible to change the variables values from the other modules. The access functions protect the variables, e.g. controlling the value range. (See table 2).

Analyzing the external system process, we can determine all relevant variables associated with, in order to model all the features and specifications of the application. They are classified as the inputs and outputs for the external system controlled by personal computer (PC): analog inputs (AI), analog outputs (AO), digital inputs (DI), and digital outputs (DO).

3.1.1 Variables and access functions for reading and writing

The data module defines the problem variables, for example temperature, humidity, etc.

```
<type> Variable;
```

The C data type "int, float, bool", etc. must be chose depending on the physical magnitude represented by the variable. To update the variables, there are defined two types of access functions:

- Access functions for writing, which stores the current values of the variables. The prototype of these functions is:

```
void DataWriteVariable(<type> var);.
```

This function needs a parameter of the same type that the variable that is going to be updated. It returns an empty value ("void"). A minimal implementation for this function would be:

```
void DataWriteVariable (<type> value){
        Variable = value;}
```

- Access functions for reading, are those that provide the values of the variables to other modules. The prototype of these functions is:

```
<type> DataReadVariable(void);
```

This function doesn't need any parameters. It returns a value of same type the accessed variables. An implementation of these functions is:

```
<type> DataReadVariable (void){
        return(Variable);}
```

The prototypes of the functions are included in the header file "`DataModule.h`" in order to export these interfaces and publicity them to other modules, so they can use the variables. (See code 1).

In the "DataModule" source the variables and the access

```
void DataWriteVariable(<type> value);
<type> DataReadVariable(void);
// others access functions declarations
```

Code 1: The data header file.

functions are defined. In the code 2, we can see a pseudo code for a writing access function. Observe that this function allows us to control the range of the variable (minValue, maxValue). This is the main purpose of the access function, to protect the variables.

```
#include <stdio.h>
#include "DataModule.h"
// Declaration of variables in the model
<type> Variable;
// others variables
// Implementation of the access
//----functions for writing ----void
DataWriteVariable(<type> value){
  if(minValue<=value&&value<=maxValue)
    Variable = value;
  else
  printf("error reading Variable");}
// exception handling
//----function access for reading----
<type> DataReadVariable(void){
    return Variable;}
```

Code 2: The data implementation

All the access function implemented in the data module are basic services (resources) that will be used later in the other modules. This would be an example of a bottom-up design approach. In the table 2 we see the variable set used on the guided project.

In order to be not exhaustive, we will only use three variables in our explanation: the heater (Heater), the internal temperature (TempInt), and the desired temperature (TempDes). The use of the remaining variables of the guided project example can be deduced from these. The access functions for these variables are (See table 3):

3.2 The process module

This module, as the data module, is useful to explain how to make a design with a hierarchical bottom-up approach. There exist two types of signals present in any industrial application: The digital and analog signals. These signals will be encoded as variables with the different C types.

This module defines and implements the functions that establish communication between the external process and the computer. This communication means reading the sensors,

and writing physical actuators, using a data acquisition card (DAQ). This module also defines the interface with other modules, being the only module that includes the libraries of the DAQ card.

Table 2: Example guide variables

Type (I/O)	Name	Variable type	Min & Max
Digital Inputs (DI)	- Irrigation alarm	bool Alarm;	OFF ON
Digital Outputs (DO)	- Heating system - Drip system - Aeration system	bool Heater; bool Sprinkler; int Windows;	OFF-ON
Analog Inputs (AI)	- Relative humidity actual and desired	double HumiRel; double HumiDes;	[0.0,100.0]%RH
	- Temperature internal, external and desired	double TempInt; double TempExt; double TempDes;	[0.0, 40.0]ºC
Analog Outputs (AO)	- Ventilation system	double Fan;	[0.0, 100.0]%Power

The implementation of a modular architecture allows the designers to change to a different DAQ with a low impact in the program. It's only needed to rewrite the process module file (ProcessModule). The DAQ functions are provided by the manufacturer, and the process interface functions are those that must be implemented by the designers.

Table 3: Example of access functions

Access Function	Name

writing	void DataWriteTempInt(double value);
	void DataWriteTempDes (double value);
	void DataWriteHeater(bool value);

reading	double DataReadTempInt(void);
	double DataReadTempDes(void);
	bool DataReadHeater(void);

For each measurement with the DAQ, it's necessary to create a task for the signals being observed. It's also necessary to create a task in order to modify the behavior of the process, taking into account the DAQ card technical characteristics provided by the manufacturer. This can be done in a function like this (See code 3):

```
void ProcessIOInitDAQ(void){}
```

The designers define and implement two types of functions that make the interface with other modules:

- Functions that measure the state of the process, which are associated with two types of sensors: analog/digital. It's also necessary to link the DAQ ports to the sensor inputs.

The prototype of these functions is:

```
<type> ProcessObserveSensor(void){}
```

The code 3 remarks the necessary steps to implement the data acquisition. An important thing is to model the transfer function of the sensor system in order to obtain the physical magnitude from the voltage present in the DAQ input port.

- Functions that write on the actuators, generating output signals (digital/analog). Each actuator must be connected to a physical output port.

```
void ProcessHandleActuator(<type>value){}
```

In the case of digital sensors and actuators, the process functions can use bite-wise C operators and masks to read/modify only those bits that are necessary.

```
#include "ProcessModule.h"
// include vendor DAQ library //
// Initialization of DAQ card //
// void ProcessIOInitDAQ(void){…}
// sensors&actuators tasks management//
// Create tasks, physical channels,
// signals levels, etc //
// initialization of digital masks//
//functions for reading sensors //
<type>ProcessObserveSensor(void){
// function that handle a sensor //
// local variables //
// <type> MeasureUnits;  //
// units in the MKS system //
// <type> Variable;
//start read  with the DAQ functions //
// DAQ handling errors  //
// transference function  //
// Variable = f(MeasureUnits); //
// return the same <type> value //
   return(Variable);  //
}
//functions for writing actuators //
void ProcessHandleActuator(<type>value){
// function to handle an actuator //
// local variables  //
// <type> ActionUnits;  //
//units in the MKS system //
//transference function  //
// ActionUnits = f (value); //
// write values with DAQ functions //
// DAQ handling errors  //
}
```

Code 3: Process implementation

3.3 The sequence module

The sequence module organizes the tasks execution in order to obtain a consistent behavior's program. If not, the system fails. This module is a good example to explain the proposed hybrid methodology, and also the versatility of C functions. The main sequence is implemented by the SequenceTasks function, and it's decomposed in other subtasks. Each of these subtasks can be implemented in this module, or may be in other modules. For example, we use the top-down approach for the implementation of the subtasks: observe, decide, and actuate. All of them are implemented in this module using functions made in others modules. And the bottom-up design approach is used for the implementation of the subtasks: Decide and inform, that are implemented directly in other modules.

Each of the tasks in the sequence module is executed in the same order that is showed in the figure 1, and are repeated every sampling period or interval (T). To do this, it's necessary to add in the sequence module a resource named “Timer”.

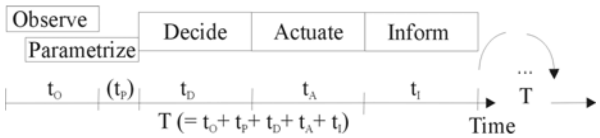


Figure 1: Task sequence

Following our guided example, we are going to use the temperature inside the greenhouse, the desired temperature introduced by the user, and the heater actuator to explain a simple sequence of tasks.

- Observe task. Is implemented by SecuenceObserveProcess (See code 4). This task acquires the sensor measures from the physical process and update all the variables with the current state (see figure 6). This is done using two subtasks: DataWriteVariable which stores values in the common data module, and the ProcessObserveSensor that returns the actual value from the sensors. The parameter of DataWriteVariable must match with the returned value from ProcessObserveSensor.

For the example, the indoor temperature measure of the greenhouse:

```
void SequenceProcessObserve (void){
DataWriteTempInt(ProcessObserveTempInt())
;}
```

In this case the C data type is double:
ProcessObserveTempInt returns a double value (the indoor temperature in Celsius degrees). And DataWriteTempInt updates the TempInt double variable in the data module.

- Parameterize task. Is made by the function SequenceVariablesDesired. (See code 4). This task allows the user to setup the system goals. This task is also divided into subtasks: DataWriteVariableDes which stores values in the common data module, and GUIReadSpecificVarDes that returns the desired value from the user. The parameter of DataWriteVariableDes must match the returned value from GUIReadSpecificVarDes.

For the example, the desired temperature:

```
void SequenceVariablesDesired (void){
DataWriteTempDes(GUIReadTempDes());}
```

In this case the C data type is also double.

- Decide task. This task is carried out by the function ControlDecideAction. (See code 4). It analyzes the disagreement (or the difference) between the values of the variables, the measured by sensors, and those desired by the user. We can obtain both values from the data module with the corresponding “DataReadVariableXX” functions. This task is defined and implemented in the control module, and establishes the decision to be taken into account using a control strategy. In this explanation we focus on the programming aspects in stead of on the control algorithms.

For the example, the temperature control loop, in the ControlDecideHeater function we analyze the difference between DataReadTempDes and DataReadTempInt. The difference belongs to different ranges, which are associated with different states outputs in a finite state machine. Each of them leads to a different decided output on the Heater actuator, witch is stored in common data by DataWriteHeater.

- Actuate task. Made by the function SequenceHandleProcess. This function performs the effective output on the actuators, depending on the decision taken in the previous task. It’s implemented by two subtasks: ProcessHandleActuator, implemented in the process module, and DataReadActuator implemented in the common data module. The argument in DataReadActuator must match the returned value from ProcessHandleActuator

For the example, the actuator heater we use:

```
void SecuenceHandleProcess(void) {
ProcessHeater(DataReadHeater());}
```

- Inform task. Isn’t a critical task. Is made by the function GUIInformUser. Is used to inform the user about the current state of the system. It shows all the variables and it’s possible to print and report historical values. This task can be performed at user demand, at any sampling process. This function is implemented in the GUI module.

In our example:

```
void GUIInformUser (void){}
```

Shows on the screen the actual values (TempInt, TempDes, Heater) reading them with the functions DataReadXX. See code 5.

The TimerControl function activates SequenceTask (see figure 2). The length of the sampling period T is given in the specifications of the system. In all cases, the sampling period should never be less than the maximum expected computation time for all the tasks (figure 1). This is valid only in non-critical applications. In other cases, if the tasks affect to the safety or property of persons, the duration of the tasks, will be accomplished under the control of a task scheduler in a real-time operating system, to avoid the system degradation. That case is not covered in this project.

```
#include <stdio.h>
#include "SequenceModule.h"
#include "GUIModule.h"
#include "DataModule.h"
#include "ControlModule.h"
#include "ProcessModule.h"
void SequenceProcessObserve(void){
DataWriteActualVariable(ProcessObserveSensor());}
void SecuenceVariablesDesired(void){
DataWriteVariableDes(GUIReadSpecificVarDes());}
void SecuenceHandleProcess(void) {
ProcessHandleActuator(DataReadActuator());}
```

```
// Timer Enabled
// TimerControl(SequenceTasks());
// Executed every timer period T
void SequenceTasks(void){
     SequenceObserveProcess();
     SequenceVariablesDesired();
     ControlDecideAction();
// Implemented in ControlModule
     SecuenceHandleProcess();
     GUIInformUser();
// Implemented in GUIModule
}
//- functions for writing actuators -//
 void ProcessHandleActuator(<type>value){
// function to handle an actuator
// local variables
// <type> ActionUnits;
// units in the MKS system
// transference function
// ActionUnits = f (value);
// write the value using the DAQ functions
// DAQ handling errors
}
```

Code 4: Sequence implementation

```
#include <vcl.h>
#include "GUIModule.h"
#include "DataModule.h"
#include "ProcessModule.h"
#include "ControlModule.h"
// Initial. with GUI methods
// Initial. DAQ card, for e.g.,
ProcessIOInitDAQ();
// read functions for desired
// parameters from GUI, e.g. TempDes
void GUIInformUser (void){
// show the actual values
// using GUI functions , like:
Form1->Edit1->Text=DataReadTempInt();
Form1->Edit2->Text=DataReadTempDes();
Form1->Edit3->Text=DataReadHeater();
// the remaining model variables
}
```

Code 5: Graphical User Interface (GUI)

For example, in the guided project a button click event validate the desired temperature introduced by the user in an edit box. See figure 3.

3.4 Control module

In the case of sampling systems, the control module manages the appropriate decisions in the system. In this module we define the functions that control the external process, based on the current states, and the desired objectives of the user. Both, the current states and the desired values of the user are queried from the data module, via access functions. Depending on the discrepancies, the control module decides the action to be taken.

In this project, we propose a simple control. Therefore, we propose an algorithm that performs a finite state machine. Each of the system states must be well defined. The control is based on the discrepancy between the actual and desired variables. The actuator outputs from each state, is given in terms of these differences. We assume that each response of the process is independent to any others, and is given by the action of an actuator on the process.

The code 6, shows two different functions in the control module. The function ControlDecideActionAnalog, is an example for a proportional response used for the analog actuators.

And the prototype function ControlDecideActionDigital is an example for the digital actuators.

```
#include "ControlModule.h"
#include "DataModule.h"
void ControlDecideActionAnalog(void){
if(DataReadActualVariable()<DataReadVariableDes()){
 if(DataReadActuator()<ACTION_MAX)
 DataWriteActuator(DataReadActuator()+ ACTION_INC);
 }
  else
 if(DataReadActuator()>ACTION_MIN)
       DataWriteActuator(DataReadActuator()-
     ACTION_INC);
void ControlDecideActionDigital(void){ }

  if(DataReadActualVariable()<=DataReadVariableDes(
 ))
      DataWriteActuator(Actuator_OFF);
    else
      DataWriteActuator (Actuator_ON);
}
```

Code 6: Control module

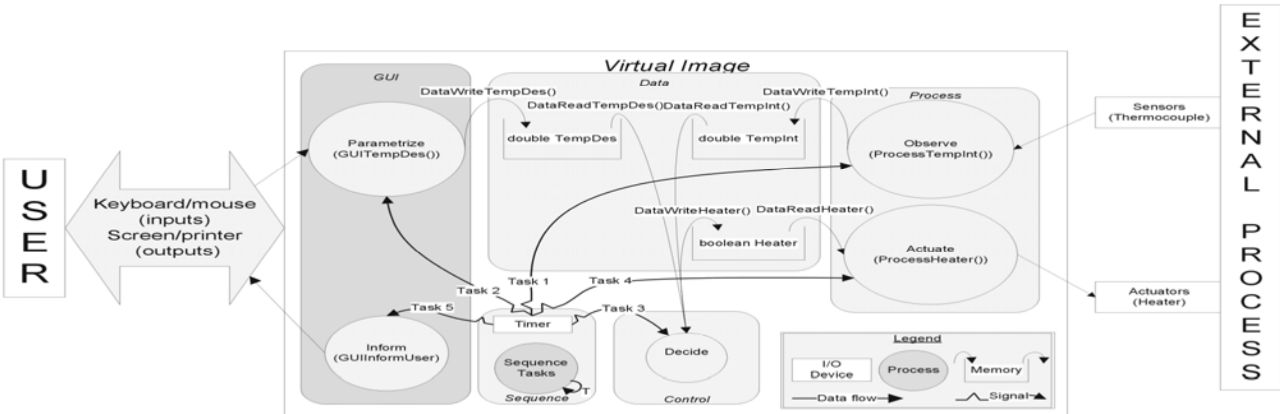


Figure 2: System data flow

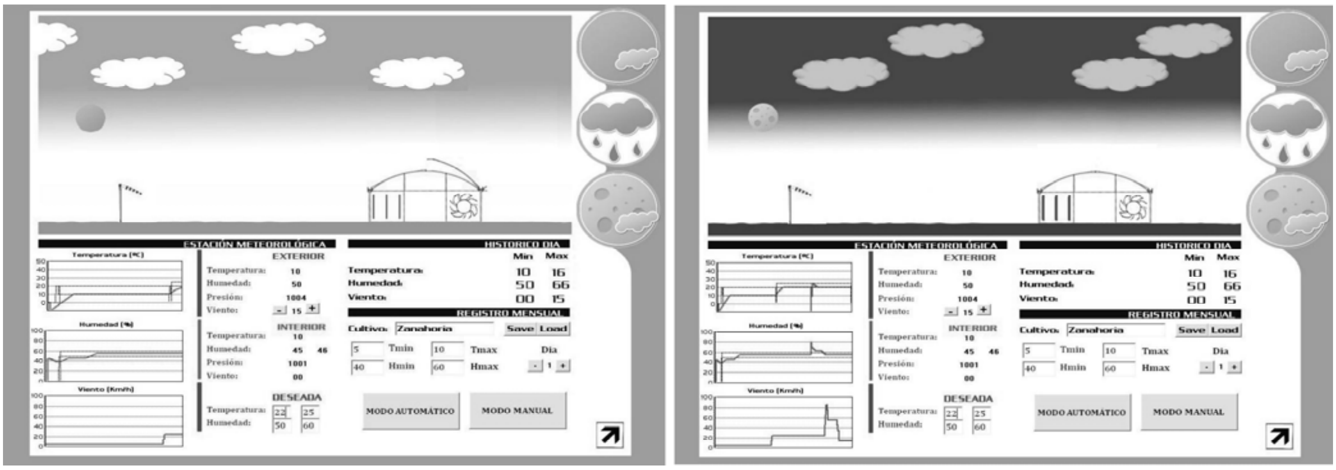


Figure 3: Examples of GUI in the guided project

4. Conclusions

The methodology has been presented to designers teams. The conclusions about this experience are the following. In general, the recommendations about the methodology are perceived as an extra burden, because they have to make separate compilation and manage the set of project's modules. To do that, the designers need to understand the development tool and the compile and link processes. Additionally, they have to deal with the variables scope and the control of their visibility ("#include" directive, "extern or static" key words, etc.). The advantages that are evident in complex real projects are not so evident in a simple project like this. There is no discernible workflow acceleration, e.g. edit-compile cycle if it is not a project with thousands, or hundreds of thousands lines. It is actually very soon perceived the advantage of the decomposition of the problem, due to the project is developed by a team of developers and is useful to delegate parts of the work in other people.

It must be considered, however, the necessary initial phase of coordination and establishment of a mechanism for communication among team members.

In the choice of a design method downward or upward, there are always times when the designers "*does not see where are they going, no matter what the method is used*".

If it comes to solving some downward (top / down) it is convenient to try to reassure the designers telling him/her "*this part is going to be detailed later*". If it comes to solving some upward (bottom / up) is important to note that the effort is being devoted to implementing these low-level services will be amortized, since these are services that will be used, although is not very clear now how or why.

The assessment of the proposed design methodology causes different reactions to the designers. In general, the majority group tries to follow the recommendations. Another group feels serious doubts and openly raises questions about it. Finally, a minority group, which could be labeled as "the rebellion group", continues with the development using different not evaluated approaches, usually based on some personal scheme of the programming process.

The recommendation adopted is usually to rethink the concepts again and drive the situation if it happens at an early stage of development process and the team is still receptive. Otherwise, if the project is at an advanced phase or the team is not receptive, it is also interesting to let things flow and try to exploit the adverse situation to make a comparison of advantages / disadvantages of each development approaches. Learning process is often the best when the difficulties have been experienced in a personal way. It sometimes happens, however, that due to the small size of the training project is not always clear the difficulty of the development without formal methods, difficulty that would be clearly evident in real complex projects.

2 References

[1] S. Cook (1983). An overview of computational complexity. Communications of the ACM, 26(6), 400-408.

[2] Huges, J.K. & Michtom J.I. (1977). A Structured Apporach to Programming. Prentice-Hall.

[3] Kernighan; B.W. & Plauger, P.J. (1979). Programming style: examples and counter examples. Yourdon Press. Upper Saddle River, NJ, USA.

[4] Liskov, B. & Guttag, J. (1986). Abstraction and Specification in Program Development. The MIT Press.

[5] McFarland, G. (1986). The benefits of bottom-up design. ACM SIGSOFT Software Engineering Notes. 11, 43–51.

[6] Davis, A.M. (1994). Fifteen principles of software engineering. Software, IEEE, 11(6); 94 - 96

[7] Nandigam, J.; Gudivada, V.N. & Hamou-Lhadj, A. (2008); Learning software engineering principles using open source software. FIE 2008. S3H-18 - S3H-23.

[8] Al-Abdulkarim L., & Al-Yahya M. (2009). Teaching Software Engineering Principles Using Micro Project with Eclipse (MPE) Activity: An Evaluation. Fourth International Conference on Software Engineering Advances.

This work was partially supported by the School of Computer Engineering (ETSINF), and the Department of Computer Systems (DISCA) at the Technical University of Valencia

Testing Students' Knowledge of C Language Using Bonus Questions

M. Korenblit
Department of Computer Science, Holon Institute of Technology, Holon, Israel

Abstract **-** *The paper deals with special questions which are asked students in exams in C programming language. A student receives a very short program in C and has to answer either a question on output of the program or on a function return. For some students, at first glance, the program seems tangled and strange, and the question looks unsolvable. In fact, however, the problem is immediately solved if a student understands some details of C.*

Keywords: C Programming Language, Bonus Question, ASCII Code, Character, Function, Operator

1 Introduction

C [3, 6] is a general-purpose high-level computer language. It is one of the most widely used programming languages and there are very few computer architectures for which a C compiler does not exist. C has greatly influenced many other popular programming languages, such as C++, C#, and others.

C was designed to provide low-level access to memory with language constructs that map efficiently to machine instructions. Despite its low-level capabilities, the language encourages cross-platform programming. A standards-compliant and portably written C program can be compiled for a very wide variety of computer platforms and operating systems with few changes to its source code. The language is available on a very wide range of platforms, from embedded microcontrollers to supercomputers [11].

For this reason, C is widely used in academia, industry and commerce [4]. It is taught in many institutions of higher learning as the first (or basic) programming language for students of computer specialties and as an example of a programming language for students of non-computer specialties. For a computer specialty, a course in C is not only a prerequisite for other programming courses, it is required for all advanced studies in computer science. In addition, for students for whom computers are not a specialty, a course in C language can inspire the ability of logic thinking, provide advantaged computer method and application tools for later specialty theory studying [9].

At the same time, the experience with teaching C programming has shown that students have a problem understanding a number of its concepts and dealing with the syntax of the language [2]. C's syntax, specifically, the syntax of input/output functions, is not elegant. The C programming language can be mysterious and gives special meaning to many punctuation characters, such as asterisks, plus sign, braces, and angle brackets [8]. The condensed fashion of coding in C is not always clear for beginners. At the same time, in comparison to a MATLAB program, the C corresponding code is longer and carries a substantial overhead [4].

Because of these facts, many papers are devoted to teaching C programming. In some of these publications [2, 7, 9] the authors describe the specific courses in C language which are taught in their institutions. A number of articles [2, 4, 5, 7] compare and contrast C with other programming languages. Some of these concern the teaching of C (successively and concurrently) with other computer languages and platforms. Effective methods for teaching the C programming language are discussed in [8] and [10].

2 Exams in C and use of bonus questions

In this paper, we describe some special questions asked on exams in C programming at the Holon Institute of Technology. The titles of courses in which these exams are given are: "Programming in C" and "Advanced Programming Workshop". The first course is intended for the first year students of electrical and electronics engineering. The second one is given to the first year students of the computer science department. The students take this second course after taking "Introduction to Compute Science." In "Advanced Programming Workshop" students acquire skill in programming on the base of advanced tools in C.

The following are examples of the kinds of questions asked on an exam in C programming:

1. Write a function that implements an algorithm to solve any problem.
2. Write a complete program that uses the above function and standard functions.
3. Fill in the blanks in the code of the following function (program).
4. Identify and correct the errors in the following code.
5. What does the following program (function) do?
6. What is the output of the following program?

The questions concerning writing a code are the most important. They constitute a significant part of the exam and have the most weight on the final grade.

A question like number 6 (shown above) is asked as a bonus question. A student receives a very short program and has to show its output. At first glance, the program seems tangled and strange, and the question looks unsolvable. In fact, however, the problem is solved immediately if a student understands some details of C.

A student gets additional points for the correct answer to this question. In addition, the question is very easy in checking exams.

Although these questions are relatively short, by answering them, students demonstrate their understanding (or lack of understanding) of the material. The credit for developing this kind of question belongs to my colleague, Alexander Abramovich. He proposed the questions presented in Examples 1 and 2 (see below) which were included in the final exams of the course "Programming in C" in 2004 [1].

3 Examples of bonus questions

Example 1

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '-'-'-');
}
```

This program prints the numerical value (specification `%d` indicates it) of the expression `'-'-'-'` which may seem strange. What is it? Let us analyze.

Solution. The minus sign denotes the subtraction operator. However, the same sign in single quotes is a character. Each character has its numeric representation in the computer (ASCII code) and an arithmetic operation on characters is actually the operation on their codes. The first and the third minus signs (but not the second one) in our expression are in single quotes. Therefore, we have here the difference of codes of the same character, i.e., the difference of two equal numbers. This difference equals 0 and thus, the output of the program is 0.

What must a student know to answer this question? First, he has to know the standard function `printf` and its specifications. Second, he must understand that it is possibly to perform arithmetic operations on characters as on numbers. Third, he has to possess the skills to read the sentence `printf("%d", '-'-'-')` and to distinguish its corresponding parts. At last, quick-wittedness is an important factor as well.

Sometimes, a student asks during the exam: "What is the ASCII code of `'-'`?" That is, despite understanding the essence of the problem, he fails in the last step and does not see that the information on the ASCII code value is redundant.

Example 2

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '/'/'/');
}
```

Solution. The idea is the same. The slash denotes the division operator. Therefore, we have the quotient of two equal numbers and thus, the output of the program is 1.

Example 3

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '%'%'%');
}
```

Solution. The idea is the same. The percent sign denotes the remainder operator. The remainder after division of two equal numbers equals 0 and thus, the output of the program is 0.

Example 4

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '&'&&'&');
}
```

Solution. The sign `&&` denotes the logical operator AND. The ASCII code of each ordinary character (including `'&'`) is a positive number. Any nonzero value in C is interpreted as TRUE. TRUE AND TRUE gives TRUE and the numerical value of TRUE is 1. Thus, the output of the program is 1.

Here, in addition to demands of the previous examples, a student must understand that there is no essential difference between arithmetic and logical expressions in C. C evaluates all expressions that include relational, equality, and/or logical operations to 0 or 1 which represent FALSE or TRUE, respectively. Moreover, C accepts any nonzero value as TRUE and zero is treated as FALSE.

Example 5

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '<'<'<');
}
```

Solution. The idea is similar. The sign < denotes the relational operator “less”. Two equal numbers are compared here and the statement that one of them is less than another one is false. That is, the logical expression equals 0 and thus, the output of the program is 0.

Example 6

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '<'<'<'<'<');
}
```

The program seems very similar to the previous one but its output is different.

Solution. The second and the forth signs < in the expression are actual “less” operators. The others are characters. As known from the previous example, the subexpression `'<'<'<'` equals 0. Therefore, zero is compared here with the ASCII code of `'<'` which is a positive number. The statement that zero is less than a positive number is true and thus, the output of the program is 1.

Example 7

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", !'='!=!'=');
}
```

Solution. The sign `!=` denotes the equality operator “not equal” and the sign `!` denotes the logical operator NOT. The ASCII code of `'='` which is a positive number is interpreted as TRUE and, therefore, `!'='` gives FALSE, i.e., 0. The statement that zero is not equal to zero is false and thus, the output of the program is 0.

When answering this question a student should remember that the unary operator `!` has precedence compared to the binary operator `!=`.

Example 8

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", '^'^'^');
}
```

Solution. The sign `^` denotes the bitwise operator XOR. Operands in this example are two bit sequences each of which is a binary representation of the ASCII code of `'^'`. The XOR operation is performed on pairs of bits located in the same positions in these sequences. Since the sequences are equal, bits in each pair are equal as well. The XOR operation on equal bits (0 XOR 0, 1 XOR 1) gives 0. Therefore, the result is a sequence of 0 bits that is a binary representation of 0. Thus, the output of the program is 0.

This question may present some additional difficulties for a student due to incorrect interpretation of the sign `^`. Sometimes a student forgets the meaning of this sign in C and thinks that it denotes the operation of raising to a power as in some other languages, such as BASIC, MATLAB. In this case a student asks during the exam: “Does the power operator really exist in C?” or “How to raise to a power without the ASCII code of `'^'`?”

Remark. The shortcoming of all the above questions is that their answers are only 0 or 1. After studying a number of previous exams, a student can guess a correct answer without a deep understanding of a topic. The possible solution of this problem is requiring to explain the answer.

Example 9

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%c", '&'&'&');
}
```

Solution. The sign `&` denotes the bitwise operator AND in this context. Operands in the example are two bit sequences each of which is a binary representation of the ASCII code of `'&'`. The AND operation is performed on pairs of bits located in the same positions in these sequences. Since the sequences are equal, bits in each pair are equal as well. The AND operation on equal bits gives the value of these bits (0 AND 0 is 0, 1 AND 1 is 1). Therefore, the result is equal to each of the operands, i.e., it is a binary representation of the ASCII code of `'&'`. Specification `%c` indicates that the result is printed as a character and thus, the output of the program is &.

The possible shortcoming of this question is that the output of the program is a character while `&` is the only character appearing in the parameter list of `printf`. That is, a student can as in the previous examples guess a correct answer without understanding. Requiring to explain the answer may solve this problem as above.

4 Bonus questions with strings

Example 10

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", sizeof "sizeof");
}
```

Solution. The `sizeof` operator returns the size in bytes of its operand. The string `"sizeof"` that is the operand contains six letters (characters) and a special string termination character (null character) represented by the character constant `'\0'`. Therefore, our string actually contains seven characters. The size of a character is one byte and thus, the output of the program is 7.

What must a student know to answer this question? First, he has to know the `sizeof` operator and to remember that if its operand is not a type name the parentheses may be omitted (as in the example). Second, he must understand that a series of characters written in double quotes is a string and its contents are not an operator even it is a keyword of C. Third, he should not forget that each string ends in the null character.

The typical mistakes in the question are: “The output is 6” (a student forgets the null character) and “The output is 8” (a student thinks that `'\0'` is two characters).

Example 11

What is the output of the following program?

```
#include <stdio.h>

void main()
{
   printf("%d", sizeof"+"+sizeof"++");
}
```

Solution. The idea is similar. The strings `"+"` and `"++"` contain two and three characters, respectively (with null characters). Hence, the program prints the value of the expression 2+3 and thus, the output of the program is 5.

A student must understand that there is no connection between the increment operator, `++` and the pairs of plus signs in the program.

Example 12

What is the output of the following program?

```
#include <stdio.h>
#include <string.h>

void main()
{
   char *str = "str";
   printf("%d",
    strcmp("strcpy(str,str)","strlen(str)"));
}
```

Ostensibly, the program contains three basic standard functions on strings: `strlen` (returns the number of characters preceding the null character in the string), `strcpy` (copies the second string into the first one), and `strcmp` (lexicographically compares two strings). However, let us examine closely the program.

Solution. The program, indeed, contains function `strcmp`. This function returns 0 if the compared strings are equal, -1 if the first string is less than the second one and 1 if the first string is greater than the second one. The strings to be compared are `"strcpy(str,str)"` and `"strlen(str)"` (they are in double quotes). Hence, there are no functions `strlen` and `strcpy` in the program. Function `strcmp` compares the strings character-by-character. The first three characters in the first string (`'s'`, `'t'`, `'r'`) are equal to the first three characters in the second string. The forth characters in the strings are `'c'` and `'l'`, respectively. Character `'c'` is less than character `'l'` since it is nearer to the beginning of the alphabet. For this reason, the first string is less than the second one and function `strcmp` returns -1. Thus, the output of the program is -1.

Let us consider specific mistakes committed by students in the solution of this problem.

Some students understand the essence of the problem until the end but do not remember when `strcpy` returns 1 and when it returns -1. According to my opinion, in this case, if the erroneous answer (the output is 1) is accompanied by the correct explanation, the student deserves additional points.

Another common mistake is the following. A student explains that the output is 1 “because the first string is longer than the second one”. Despite revealing two compared strings by the student, he does not deserve additional points in this case.

Some students accompanied the correct answer by the following explanation. Function `strlen` returns 3 (it is the length of the string `"str"`) and function `strcpy` returns an address (it is correct) that is less than 3 (why?). For this reason, function `strcmp` returns -1. Although the answer was correct these students did not get additional points because their explanation shows misunderstanding of the problem.

5 Bonus questions with strings and loops

Example 13

Here is function `func`:

```
int func (char *str)
{
  int sum=0, i;
  for (i=0; str[i] != '\0'; i++)
    sum += str[i];
  return sum - *str*strlen(str);
}
```

What does function `func` return if `str` points to the string `"abcde"`?

Solution. Characters are actually represented as one-byte integers in the computer. Hence, the loop in function `func` summarizes the ASCII codes of characters `'a','b','c','d','e'`. Since pointer `str` points to the first character in the string `"abcde"`, the notation `*str` is equivalent to `'a'`. The second sign `*` denotes the multiplication operator. Function `strlen` returns the length of the string that is equal to 5. Therefore, function `func` returns the value of the following expression: `'a'+'b'+'c'+'d'+'e'-5'a'` (characters denote their ASCII codes). We use the fact that the letter ASCII codes are successive numbers (`'b'` is `'a'+1`, `'c'` is `'a'+2`, etc.) and obtain:

```
'a'+'b'+'c'+'d'+'e'-5'a'=
'a'+'a'+1+'a'+2+'a'+3+'a'+4-5'a'=
1+2+3+4=10.
```

Thus, function `func` returns 10.

The only fact that students need to know about ASCII codes to answer this question is that the ASCII codes of letters are successive numbers. They do not always understand it and ask sometimes: "What are the codes of letters (at least, of `'a'`)?"

Besides, the question checks how a student understands the meaning of the sign `*` according to the context. If a student asks during the exam: "What is the sign `*`? Is it multiplication or a pointer?" the lecture answers: "It is an asterisk".

Example 14

Here is function `func`:

```
int func (char *str)
{
  int sum=0, i;
  for (i=0; str[i] != '\0'; i++)
    sum += *str/str[i];
  return sum;
}
```

What does function `func` return if `str` points to the string `"abcde"`?

Solution. The idea is similar. The loop in function `func` summarizes the quotients of ASCII codes of `'a'` (it is pointed by `str`) and characters `'a','b','c','d','e'`. That is, function `func` returns the value of the following expression: `'a'/'a'+'a'/'b'+'a'/'c'+'a'/'d'+'a'/'e'`. Integer division yields an integer result. For this reason, division of a less integer by a greater integer gives zero. Character `'a'` is the first in the alphabet and its ASCII code is the smallest. Therefore, we have:

```
'a'/'a'+'a'/'b'+'a'/'c'+'a'/'d'+'a'/'e'=
1+0+0+0+0=1
```

Thus, function `func` returns 1.

This question checks how a student understands division rules in C. The following absurd situation takes place sometimes in the exam. A student remembers the letter ASCII codes and tries to estimate the following expression:

```
97/97+97/98+97/99+97/100+97/101.
```

Even the correct result of this calculation will be incorrect concerning the question and the student will not get additional points.

6 Summary

In this paper, we have discussed the special bonus questions proposed in C programming exams and analyzed a number of students' answers to these questions. We believe that despite their small size, these questions are valid indicators of students' knowledge in various topics related to learning C (arithmetic and logical operations, characters and ASCII codes, standard functions, loops, strings, pointers, bitwise operations, etc.) and demonstrate the students' ability to grasp a C code. The questions allow to students to get additional points and, at the same time, they do not significantly increase the work of the examiner.

7 Acknowledgments

The author thanks B. Haberman, Y. Kempner, L. Kugel and V. E. Levit for helpful advises.

8 References

[1] Abramovich, A. (2003). "C Programming Language". http://www.hit.ac.il/staff/javiator/C_Electrical_Engineering_Course/index.html

[2] Budny, D., Lund, L., Vipperman, J., & Patzer, J. L. "Four Steps to Teaching C Programming". Proceedings of 32nd ASEE/IEEE Frontiers in Education Conference, Boston, MA, November 2002, pp. F1G-18 – F1G-22.

[3] Deitel, H. M. & Deitel, P. J. (2007). "C How to Program" (5th ed.). Pearson Prentice Hall, NJ.

[4] Fangohr, H. "A Comparison of C, MATLAB, and Python as Teaching Languages in Engineering". M. Bubak et al. (Eds.): ICCS 2004, LNCS 3039, Springer, 2004, pp. 1210 – 1217.

[5] Jablonowski, J. "Some Remarks on Teaching of Programming". Proceedings of International Conference on Computer Systems and Technologies (CompSysTech'2004), Rousse, Bulgaria, June 2004, pp. IV.10-1 – IV.10-6.

[6] Kernighan, B. W. & Ritchie, D. M. (1988). "The C Programming Language" (2nd ed.). Prentice Hall PTR, NJ.

[7] Puhan, J., Bürmen, Á., Tuma T., & Fajfar, I. (2010). "Teaching Assembly and C Languagy Concurrently". International Journal of Electrical Engineering Education, 47(2), 120 – 131.

[8] Richardson, J. J. "Effectively Teaching C in an Introductory Embedded Microcontroller Course". Proceedings of Il/IN Sectional Conference, Illinois, April 2005, 8 pages.

[9] Wang, Y. "Study on the Experiment Teaching of C Language". Proceedings of 1st International Conference on Information Science and Engineering (ICISE2009), Nanjing, China, December 2009, pp. 3463 – 3465.

[10] Xue, H. & Shen., Y. "An Analysis of C Language Programming Teaching Reform Based on CDIO". Proceedings of 1st International Conference on Information Science and Engineering (ICISE2009), Nanjing, China, December 2009, pp. 3200 – 3202.

[11] Wikipedia. "C (Programming Language)". http://en.wikipedia.org/wiki/C_(programming_language)

A C Language Compiler Design in Comprehensive Experiment Methodology for Computer Science and Technology

Tian Wang, Yongtao Hao
College of Electronics Information & Engineering, Tongji University, Shanghai, China

Abstract - *The problems that exist in computer science today is too much attention has been paid to validate experiment, experiment content is mainly on a single aspect of technology and other issues. It is necessary to propose an experiment reform program for the computer science professional training. This program should cover the computer science foundation and core courses such as digital logic and digital IC, principle of computer organization, principle of microcomputer and interface technology, assembly programming, compiler theory and operating system. In this paper, a theory of C-compiler for the MIPS I instruction set processor is raised as an experiment reform program. The program is based on the LLVM project, and it is more innovative and practical compared to traditional compile-principle experiment.*

Keywords: Comprehensive experiment methodology, C language, Compiler porting, LLVM, GCC

1 Introduction

Low Level Virtual Machine (LLVM) is an open source project undertaken by the University of Illinois [1]. It provides support on the compiler, and can be used as a background for a variety of language compiler and used for compiler optimization, connection optimization, online compiler optimization and code generation. LLVM reuses the GCC as the frontend process tool of high-level language and provide its unique backend porting infrastructure to avoid the heavy workload of GCC backend porting [2].

2 LLVM Infrastructure

The architecture of LLVM compiler follows the logic compiler phasing rules[3], divided into the frontend of the high-level language processing, optimization of intermediate representation, backend code generator which is related to the target processor[4] (as shown in Figure 1).

LLVM uses GCC to analysis the source code written in high-level language and parse it into LLVM Intermediate Representation (IR). The latest version has supported for C, C + +, FORTRAN, JAVA. Moreover you can add new language support through frontend porting interface. The IR optimizer which is established on the LLVM virtual instruction set will process the IR generated in the previous step on standard scalar optimization, loop optimization, as well as inter-procedural optimization. This will generate optimized LLVM IR. The backend code generator contains instruction selection, register allocation, machine code optimization, code output and so on. The three parts above are relatively independent and optimization of one part does not affect the other two parts, so that we can improve the reuse of the module and reduce the unnecessary duplication of work [5].

3 LLVM Backend porting

Backend code generator will translate the LLVM IR, which is generated by optimizer, to assembly code for a specific processor. It contains two parts: the target code generator which is processor independent and backend transfer interface. Actual code is generated by the former process and to deal with it differently according to the description made by the latter part. This kind of division reduces the cost of the development of backend code generator greatly and makes the porting work reduced to the implementation of backend transplant interface for target processor.

LLVM backend porting interface is made up of abstract classes which describe the target processor architecture. Some of the classes are listed in table 1. For each specific processor, the first thing need to do is inheriting these classes, and then the corresponding feature attribute function should be implemented based on the new processor architecture. In this way, support for the new processor should be gained.

Table 1 Examples of the LLVM backend transplant interface

CLASS	USE
TargetMachine	Global description
TargetLowering	Description of IR conversion description
MRegisterInfo	Description of register
TargetInstrInfo	Description of instruction set
TargetFrameInfo	Description of frame stack layout
TargetSubtarget	Support of processor sub-series
TargetJITInfo	Support of processor JIT
……	……

TableGen In some cases, there may be a large number of records of description for the target processor needed to be

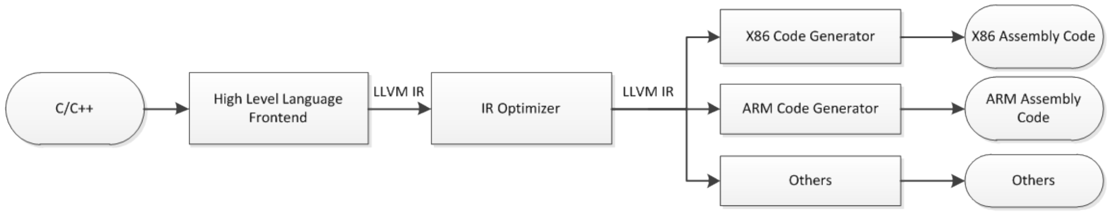


Figure 1 LLVM infrastructure

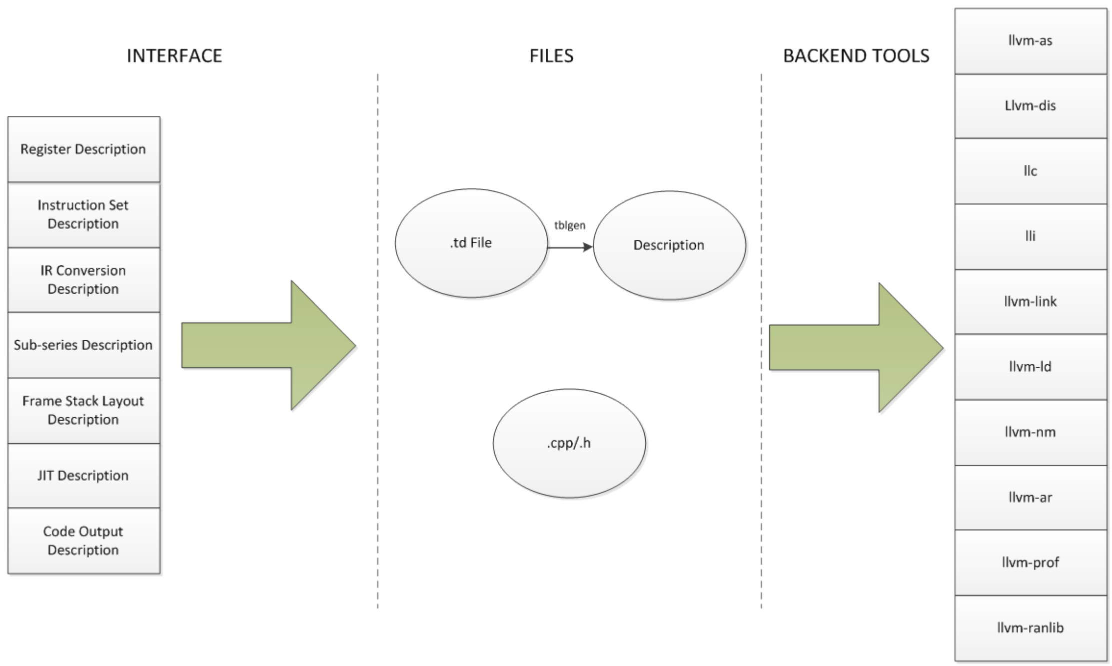


Figure 2 the LLVM backend transplantation structure

maintained, and they may have a lot in common. If these records are prepared artificially, it will spend a lot of time and will be very prone to error. To this end, LLVM provides a tool “TableGen” to reduce the workload of description. As long as users use .td file fits TableGen grammar rules to describe processor, tool tblgen can resolve it into C++ code. In this way, transplantation can be divided into two parts, using TableGen to describe the target processor and writing C++ code to complement it. Figure 2 above describes the structure of the LLVM backend porting [6].

Register description There are two aspects contained in the register description, TableGen description on the target processor and achieving class MRegisterInfo. That is to implementation these three files: XXXRegisterInfo.td 、XXXRegisterInfo.h and XXXRegisterInfo.cpp. The first file describes properties of each register in processor, alias relationship between registers and register allocation scheme when programs are running. This is achieved by using the four records provided by TableGen. XXXRegisterInfo.h and XXXRegisterInfo.cpp need to inherit class MRegisterInfo and implement virtual function contained within the class. These virtual functions include providing the instruction that transmit the value in register to stack slot, providing the instruction that take the value from stack slot to register, providing instruction that copy register and so on.

Instruction set description The instruction set description also includes two aspects: TableGen description on the target processor’s instruction set and inheriting class TargetInstrInfo. The file XXXInstrInfo.td describes instruction set of the target processor, instruction features, addressing method of instructions, instruction operand, the instruction encoding method, the output format and the relation between instructions and virtual instruction set and so on. Document XXXInstrInfo.h and XXXInstrInfo.cpp need to inherit class TargetInstrInfo and implement the virtual functions in it. These function interfaces include the judgment of an instruction is a move instruction between registers[7], to determine whether an instruction is to read or write the stack slot command and so on.

IR conversion description IR conversion describes the way how LLVM IR converts to the assembly code of target processor. This can be taken down into three parts: the legalization of the operands, instruction matching options and other conversion description. As a matter of fact these can be achieved by fulfill documents XXXLowering.h and XXXLowering.cpp.

When LLVM IR's type and the type of target processor system are inconsistent, type conversion should be done. This progress is called legalization of the operands. Instruction set used by the LLVM IR is its virtual instruction set and this may not match the instruction used by target processor [8], so the conversion of instruction should be taken, too. Other conversion should be set for the shift size of the shift instruction type and target processor scheduling optimization options.

4 Conclusion

Currently, there has existed successful LLVM transplantation in the processor ARM and processor NiosII, thus porting LLVM to the processor based on MIPS I is feasible. Compared with traditional compile-principle experiment program, this program does not focus on the lexical analysis, syntax analysis and IR generation process, but on the assembly code generation. You will not be able to describe the target processor and accomplish porting of LLVM backend if you have not understood its structure and instruction set. This arrangement complements the traditional compiler theory experiment which is mainly on lexical analysis and syntax analysis and linking the relatively isolated compiler theory experiment to digital logic device design experiment, assembly language program design experiment, CPU design experiment which has been finished in the experiment reform program. This will help students to see these courses as a whole and to form their overall knowledge of computer science and technology.

5 References

[1] Terei A, Chakravarty T. An LLVM backend for GHC[C]. Proceedings of the Third ACM Haskell Symposium on Haskell. 2010:109-120.

[2] Xiaoxi Ren, Renfa Li, Kehuan Zhang, Yuanni Guo. A compiler technology based on retargetable method for embedded system[J]. Journal of Computer Applications 2004, 24(2):165-167.

[3] Lattner C, Adve V. Architecture for a next-generation GCC[C]. First Annual GCC Developers. 2003:121-132.

[4] S. Ren, N. Lu, W. Zhang, and Z. Pan. LLVM-infrastructure-based NIOS Ⅱ backend fast porting. Computer Applications and Software, vol. 28, 2011.

[5] F. Dong, Analysis of LLVM Compiler Infrastructure and Backend Porting For ARM[D]. vol. master: Shanghai Jiao Tong University, 2007

[6] F. Dong, Y. Fu. Backend porting for ARM based on LLVM infrastructure[J]. Information Technology. vol. 7,2007.

[7] G. Hadjiyiannis, S. Hanono, S. Devadas. ISDL: An instruction set description language for retargetability[C], In Proceedings of Design Automation Conference, pages 299-302, June 1997.

[8] Vikram Advd, Chris Lattner, Micheal Brukman, LLVA: A low-level virtual instruction set architecture[C]. Proc. of the 36th Annual ACM/IEEE International Symposium on Microarchitecture (MICRO-36), San Diego, CA, December 2003.

SESSION

TEACHING METHODS AND ENVIRONMENTS, TEAM TEACHING, OUTREACH METHODS AND DIVERSITY CHALLENGES

Chair(s)

TBA

Teaching Computational Science and Simulations using Interactive Depth-of-Field Technologies

M.G.B. Johnson and K.A. Hawick
Computer Science, Institute for Information and Mathematical Sciences,
Massey University, North Shore 102-904, Auckland, New Zealand
email: { m.johnson, k.a.hawick }@massey.ac.nz
Tel: +64 9 414 0800   Fax: +64 9 441 8181

February 2011

ABSTRACT

Simulation and modelling are key techniques used by computational scientists in many disciplines. Teaching students how to best use these methods is assisted by highly interactive, visual and motivational technology. Recent commodity pricing of depth field cameras such as the Kinect makes an integrated approach to real time interactive teaching and learning of simulation and modelling feasible and exciting to students. We describe some of the techniques, software prototypes we have developed for teaching interactive simulation methods using devices such as the MS Kinect. We discuss examples including a discrete lattice model like the Ising model of a magnet, and continuous equation models such as fluid flow and speculate on the implications of ubiquitous depth-of-field devices for highly interactive simulations for learning.

KEY WORDS

depth-of-field; Kinect; teaching technologies; interative simulation; visualisation technology; computational science education.

1 Introduction

Students learn about the use and development of computer simulations in many different subjects or disciplines. Some will come to develop their own simulation systems with backgrounds in programming and software. Others will simply use simulation systems and applications without becoming involved in direct software development [13, 20].

There is great power in allowing educational users to directly interact with simulations. This mode of interaction is usually known as computational steering [5, 24, 25] and it can help them explore parameter spaces and establish some degree of intuitive understanding of the models and real systems being studied. A great many tools and techniques have been developed in the field of Human Computer Interaction [1, 12] over the last fifty years and in particular many good graphical user interface widgets are read-

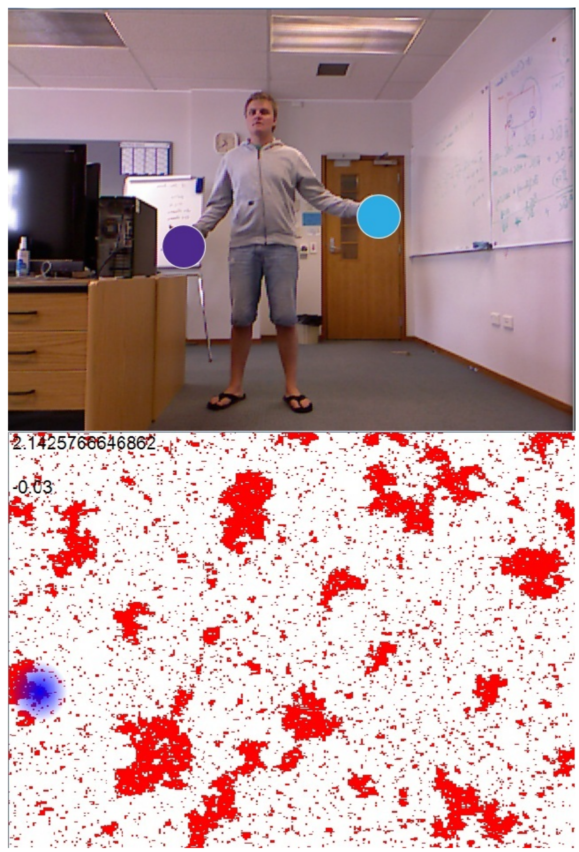


Figure 1: A user manipulating the temperature and magnetism of a two dimensional Ising model using the position of their hands in the x,y-Plane

ily incorporated into simulation software. For example widgets such as sliders, spinners, text-fields, navigation bars and so forth are now commonplace and make good use of the ubiquity of a mouse or similar spatial two dimensional tracking device on essentially all modern desktop and laptop computers.

Devices that allow three dimensional interaction have been experimented upon in recent years but only within the last year has an economically priced such device – the Microsoft Kinect (Kinect) – appeared with the potential to be genuinely ubiquitous. As often happens in computing technology, technical possibilities do not really become accepted and widely used until they are economically viable.

In this article we consider the use and impact of driving computational simulations using the Kinect [23, 42] in an educational environment. We discuss how a well known simulation model – the Ising model of a magnetic material - can be readily linked into the software framework of a device like the Kinect so as to support human gestural and also vocal interaction with the running model. The ease with which this can be done opens up many educational possibilities to allow non-programmer students easy interactive access to a range of simulation models.

The Kinect [29] has attracted a lot of attention as an enabler of innovative HCI modes for gaming applications [31, 35, 40], but also for applications including geospatial navigation [4], handicap and visual impairment support [6, 7, 44], and also interactive learning [8].

Our research group has developed a number of simulations that can use this learning and interaction approach, including equation-based simulations such as the Cahn-Hilliard [18] or Ginzburg Landau systems [15]; agent-based [19] and particle-based [17] models; phase transitional models such as the Kawasaki [16] and Potts systems [14]. In this present article we focus on use of Kinect-like devices to interlink to and drive models like the Ising system and fluid dynamic system.

Commodity-prices devices like the Kinect make possible a range of human interaction possibilities The Kinect itself is well integrated set of sensors [28] including cameras, orientation devices and sound capabilities. Devices like the Kinect and its software frameworks support detection of specific devices like paddles [32] or wands, but more interesting - and indeed natural, is for the user to use gestures [9, 10, 34, 39, 41] to interact with a simulation model.

Much attention to date has focused on interacting with

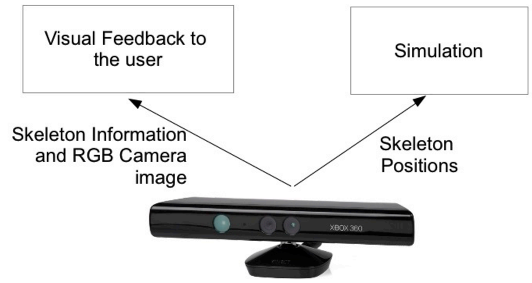


Figure 2: this is a possible picture that we could use to show the architecture

3D objects [22, 36]. Our present paper focuses on the human user as a 3D entity [33] that must be recognized and the detected [26] human activities [38] used as feedback into any interesting running simulation.

The necessary human detection [43] requires tracking the whole skeletal body [21, 37] as well as specifics such as head and hands [11]. There appears to be a wealth of work to be done in the field of HCI in appropriately categorising and naming appropriate 3D gestures so that suitable simulation driving libraries can be formulated. As we discuss in this paper, it is not necessarily feasible or indeed desirable to solely use gestures and a hybrid approach using some mix of gestural and speech/sounds [30] may be more natural for a simulation user.

In Section 2 we describe some of the typical parameter interactions of simulation models; in Section 3 we describe how we linked our model to the Kinect software framework/libraries. We present some simulation implementations showing the system and discuss our results in Section 5 and offer some concludions and summary of the educational experiences and uses of this device and its likely successors in Section 6.

2 Interactive Simulations

Integrating the Kinect into a teaching pedagogy can function as both a tool and a toy [27]. As a tool it enables students to explore the simulations in a new and exciting way. Although, most simulations are not classified as a game in the classical sense [27], they can be used as such to encourage and inspire students to study computer science by exposing them to the science of computing rather than just using a computer as a tool.

Two distinct types of control methodologies must be considered when implementing Kinect control into a simulation. The first is where the user controls variables the in the simulations with their body gestures.

Examples of this is the Ising Model shown in figure 1 where the user manipulates the temperature with their left hand and the magnetism with their right. In the second, a user directly manipulates the simulation dataset in real time directly interacting with the on-screen simulation. An example of this is a fluid simulation as shown in figure 3 where the user is directly modifying the problem space rather than manipulation of variables that influence the model. In an the Ising model the user could use the Kinect to individually break links between lattice points as in [14] and observe how that changes the properties.

Figure 3 shows the Nvidia Compute Unified Device Architecture (CUDA) GPGPU OpenGL fluids demonstration program being controlled using the Kinect. To integrate the Kinect control with the CUDA simulation, we use C++ .NET to initialize and get the skeleton information from the Kinect which we then pass directly to the CUDA kernel. In this situation the Kinect functions as a mouse replacement, but we see in figure 1 the Kinect taking the place of a graphical slider or button widget.

For any interactive simulation user feedback if vital to confirm that you are changing the model in the intended way. For the Ising model shown in figure 1 we first implemented only a blue semi-transparent blue circle that would follow the users hand as they moved in in the y-plane. We discovered that this feedback was insufficient and users frequently stepped out of the view of the camera and/or felt that the system was tracking their hands incorrectly. To remedy this we added another window that displayed the RGB camera output to the screen and also drew two colored circles around their hands on the screen to show them that they were being tracked correctly. This solution fixed all problems with users positioning themselves incorrectly in the camera frame and reassure them that there hands were being tracked correctly.

Figure 2 shows the simulations architecture we have implemented this allows us to interchange any simulations into the software without having to re-implement the user visual feedback discussed above. This is a variation of the agent-driven architecture presented in [2] where the simulation defines a set of rules and parameters about how the users within it may act but allows the user to change their state.

3 Kinect & Related Technologies

The Kinect represents a new paradigm in human computer interaction. It utilizes two depth sensors and a single RGB camera to detect up to six people simultaneously and create a virtual skeleton to map to two of those users. It also includes four microphones for voice recognition and voice triangulation to distinguish between users.

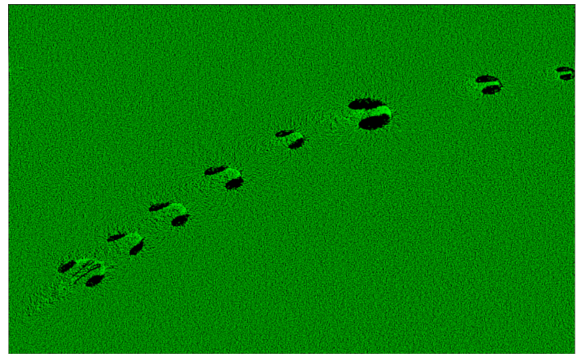


Figure 3: A user 'stiring' the fluid simulation running in realtime using GPGPU and CUDA

The initial version of the Kinect made for the Xbox 360 which we used in our simulations has a minimum depth sensing capacity of 0.7 meters which works very well when detecting a full body skeleton. A new version of the Kinect made for windows systems allows for a 'near' mode which reduces the minimum distance to 0.4m. This makes it more practical for desktop uses where the user remains sitting at the desk rather than standing.

The Kinect has limitations in its depth sensor mechanism which uses an infrared laser project and a monochrome CMOS sensor [42] is only capable of detecting 2048 levels of depth in the range of 0.7 to 6m. It is possible [43] to distinguish a hand from the depth input at close range. However, it is insufficient to distinguish specific hand gestures at a distance where the whole user is visible by the camera. This limitation restricts the number of recognizable actions that the user can articulate with their body.

Outside of the official SDK It has been used in [10,39] to track a single hand and table tennis paddles in [32] but is unable to do so at a meaningful distance away from the device where the whole skeleton is visible we explore this limitation in section 4.

Using the skeleton detection is simple on the Kinect. A majority of the work is done for you on the device processor and skeleton detection in code is trivial. We show in figure 4 the 20 point skeleton data relayed from the device overlaid onto the RGB camera stream, which is then displayed back to them to show the current tracking status. The official Microsoft SDK currently supports C#,C++ and Visual Basic. We found that C# was the easiest to use due to the managed code. Simple initialization of the Kinect is shown in pseudo code in Figure 1

Algorithm 1 Pseudo-code for Initialization of the Kinect for skeleton tracking

declare $Kinect\ kinect\_device$
$kinect\_device \rightarrow enableSkeletonTracking$
$kinect\_device \rightarrow Start()$
$kinect\_device \leftarrow$
$new\ event\ handler\ for frame\ ready\ event$
$frameReadyEventFunction()\{...\}$

Currently Windows is the only platform supported by the official Microsoft SDK which we have used in this paper. In December 2010 an Open Source SDK for skeleton tracking was released by PrimeSense the company behind the underlying Kinect technology. This open SDK provides similar functionality in a multi platform environment, but is out of the scope of this paper.

In December 2011 computer maker ASUS and PrimeSence released a depth sensing skeletal tracking device called the Xtion Pro and the Xtion Pro Live. These devices are very similar to the Kinect in almost every way. The Xtion Pro has a infrared emitter and a monochrome CMOS sensor, while the Xtion Pro Live has the same specs and adds two microphones and a RGB camera for colored video [3]. More detailed specifications such as the depth sensitivity are not available for these devices.

4 Gestures

The basis of most modern computer input devices is the hand and fingers (mouse, keyboard, touchscreen), this is because they are one of the most dexterous parts of the body. The Kinect is unable to simultaneously track a users whole body and their finger/hand gesture movements at the same time. This reduces the number of useful (practical) gestures a user can articulate with their arms and legs.

There is no current standardized gesture vocabulary for Depth of Field computer input devices such as the Kinect. This makes it difficult for developers to implement consistent user interface experiences. The most difficult action to articulate is the mouse 'click'. Currently this is typically performed by holding a users hand over the target on the screen and holding it in the same position for a fixed amount of time. This signals the users 'intent' to select the option rather than a unintentional hand movement. This method was impractical for our simulations as once tracking

Figure 4: A user feedback visualization showing the 20 point skeleton overlaid onto the user

was begun any movement of the hands would alter the simulation parameters, thus moving the hands to a specific place on the screen was not possible. The Ising model simulation uses the x,y,z positions of the users hands in 3d space. The x and y position for each hand is used as direct inputs into the simulation, while the depth plane was used for the user to signal their intent to manipulate the model parameters. For use in a dynamic teaching environment we allow only the closest user to interact with the simulations.

Utilizing the voice recognition features of the .NET toolkit were also trialled. We tested the accuracy of two voice commands at a range of distances from 0.5 - 4 meters away from the Kinect sensor. The voice keywords used where "stop" and "go" to indicate the start and stop of hand tracking. The testing was performed in a closed lab with no background noise.

5 Results and Discussions

The skeleton tracking from the Kinect is performed at an average of 30 frames per second which is more than sufficient for fluid skeleton tracking. As a majority of the processing is done on the device there is minimal host system overhead in processing the data returned from the device. When implementing the Ising model Kinect interaction was the the display overhead in printing the RGB camera output from the device to the screen was slowed significantly when it was performed in the same thread as the main simulation. This was remedied by moving the simulation to another thread, this approach allows the simulation to run without negatively affecting the feedback to the users.

When controlling the simulations we found that there was no intuitive way to initiate or end the tracking while maintaining the current setting in the simulation. We solved this by ending the tracking when the users hands were behind their body in the depth plane. Other methods trialled were clapping to trigger an audio switch, this method resulted in accidental altering in the simulation when the hands were moving. Audio commands were also tried but were to imprecise even in a lab setting. The voice commands were correctly recognized only 60% of the time in a lab testing environment with no background noise. We speculate that this could have been due to the system not correctly recognizing New Zealand (or Scottish) accents. While The voice recognition accuracy could be improved to allow for differing accents we believe it would be impractical in a dynamic teaching environment where there may be significant background noise and/or movement.

The Kinect enables complex simulation spaces to be explored with the use of body gestures and sound. We found that student interaction with the models too be intuitive and allowed them to see in real time the effects temperature and magnetism changes had on the Ising simulation without having to completely understand the model or programming. This tool can be used as both a teaching resource and a recruiting mechanism we have successfully used these demos to recruit new students at university open days where the Kinect worked well in the busy environment.

The lack of industry standardized gestures to interact with the screen (like the mouse and keyboard) is not necessarily a problem. We believe that the technology in a general sense is much to dynamic to impose such a set of standards to cover all uses. However the 'click' action still remains frustrating unintuitive and constrained by hardware limitations in the current generation of Kinect and other similar devices such as the Xtion Pro Live. Using two Kinect's in a stereo vision arrangement presents a future research opportunity where the skeleton detection on the Kinect is used for course depth detection and the stereo vision could be used to get fine depth measurements for accurate hand gestures.

To extend the Kinect interaction from the toy space into the tool space [27] more advanced interaction parameters must be implemented. This presents a fundamental problem with the Kinect in its current form. If we use the Kinect as an evolution of the mouse then there is no equivalent keyboard facility. The lack of the ability to select different functions of the mouse presents a significant obstacle. On screen widgets could replace the keyboard but without an intuitive click mechanism that is not obviously viable. Simple speech commands might be suitable substitutes for clicks and so forth.

6 Conclusions

We have developed and experimented with some prototype educational learning simulations using depth-of-field devices such as the Microsoft Kinect. We have shown that the Kinect is a powerful educational tool that allows users to interact with simulations, both directly and in-directly.

Our software experiments focussed on replacing conventional sliders and form interaces with hand gestures to continuously change model parameters within a running simulation. The present generation Kinect technology is well suited to this.

This allows students to interact with these simulations in an intuitive manner and gain an understanding of the model without having to know/learn programming. This ability to expose potential students to the concepts of advance simulation programming in a fun and engrossing way is invaluable. For current students it allows a three dimensional navigation within the simulation space using intuitive body gestures.

The current generation of the Kinect works very well as a simulation exploration tool. However, it has some significant limitations in its depth sensitivity as it is only able to process 2048 levels of depth in the range 0.7 to 6 meters. This sensitivity is insufficient for accurate hand gesture recognition an a distance where the users whole body is visible by the device.

We believe that devices like the Kinect or Xtion Pro will likely become ubiquitous and thus available to standard user applications software. We have indicated some uses that such devices can be put to in support of interactive simulations, and learning support tools.

There is still not inconsiderable work to be done however in establishing a suitable software vocabulary for describing the human gestures and interactions that can and will be used in such simulation learning tools. Without a standard such vocabulary it will be difficult to develop reusable support libraries to support interfacing to simulations software and other learning tools. There are a number of other interesting directions for depth-of-field hardware and related novel HCI devices to be developed. Hybrid devices that support a mix of gestural and speech interactions seem to hold particular promise.

Acknowledgments

Thanks to V Du Preez and B Pearce for their assistance in testing our system.

References

[1] Antle, A.N., Marshall, P., van den Hoven, E. (eds.): Workshop Proceedings Embodied Interaction: Theory and Practice in HCI. 29th Int. Conf on Human Factors in Computing Systems, ACM, Vancouver, BC, Canada (7-12 May 2011), iSBN 978-1-4503-0268-5/11/05

[2] A.P.Gerdelan, Hawick, K., A.Leist, Playne, D.: Simulation frameworks for virtual environments. In: Proc. International Conference on Internet Computing (ICOMP'11). No. ICM4087, Las Vegas, USA (July 2011)

[3] Asus: Xtion pro live. http://www.asus.com/Multimedia/Motion_Sensor/Xtion_PRO_LIVE/ (March 2012)

[4] Boulos, M.N.K., Blanchard, B.J., Walker, C., Montero, J., Tripathy, A., Gutierrez-Osuna, R.: Web gis in practice x: a microsoft kinect natural user interface for google earth navigation. Int. J. Health Geograpics 10, 45 (2011)

[5] Brooke, J.M., Coveney, P.V., Harting, J., Jha, S., Pickles, S.M., Pinning, R.L., Porter, A.R.: Computational steering in realitygrid. In: Proc. UK eScience All Hands Meeting. vol. 2, pp. 2–4 (2003)

[6] Cameirao, M.S., Badia, S.B., Oller, E.D., Verschure, P.F.M.J.: Virtual reality based rehabilitation and game technology. J. NeuroEngineering and Rehabilitation 7, 48 (September 2010)

[7] Chang, Y.J., Chen, S.F., Huang, J.D.: A kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. Research in Developmental Disabilities 32(6), 2566–2570 (2011)

[8] DePriest, D., Barilovits, K.: Live: Xbox kinect virtul realities to learning games. In: Proc. 16th Annual Technology, Colleges and Community Online Conference (TCC'11). pp. 48–54. No. ISSN 1937-1659, University of Hawaii (2011), http://etec.hawaii.edu/proceedings/2011/

[9] Deshayes, R., Mens, T.: Statechart modelling of interactive gesture-based applications. In: Proc. First International Workshop on Combining Design and Engineering of Interactive Systems through Models and Tools (ComDeis- Moto),. Lisbon, Portugal (September 2011), iNTERACT 2011, 13th IFIP TC13 Conf. on HCI

[10] Doliotis, P., Stefan, A., McMurrough, C., Eckhard, D., Athitsos, V.: Comparing gesture recognition accuracy using color and depth information. In: Proc. Conference on Pervasive Technologies Related to Assistive Environments (PETRA). Crete, Greece (25-27 May 2011)

[11] Garstka, J., Peters, G.: View-dependent 3d projection using depth-image-based head tracking. In: Proc. 8th IEEE Int. Workshop on Projector-Camera Sytems. pp. 52–58. Colorado Springs, USA. (24 June 2011)

[12] Gouin, D., Lavigue, V.: Trends in human-computer interaction to support future intelligence analysis capabilities. In: Proc. 16th Int. Command and Control Research and Technology Symposium. No. Paper 130, Quebec City, Canada (21-23 June 2010)

[13] Hawick, K.A., James, H.A., James, C.J., James, C.L.: On summer schools, building student teams and linking undergraduate education to research. In: Arabnia, H.R. (ed.) The 2006 International Conference on Frontiers in Education: Computer Science & Computer Engineering. p. CD. CSTN-023, Las Vegas, Nevada, USA (June 2006), cSREA Press, ISBN 1-932415-99-8

[14] Hawick, K.A., Johnson, M.G.B.: Bit-packed damaged lattice potts model simulations with cuda and gpus. Tech. Rep. CSTN-134, Computer Science, Massey University, Albany, Auckland, New Zealand (July 2011), www.massey.ac.nz/~kahawick/cstn/135/cstn-135.html

[15] Hawick, K.A., Playne, D.P.: Numerical Simulation of the Complex Ginzburg-Landau Equation on GPUs with CUDA. In: Proc. IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN). pp. 39–45. No. CSTN-070, IASTED, Innsbruck, Austria (15-17 February 2011)

[16] Hawick, K.: Visualising multi-phase lattice gas fluid layering simulations. In: Proc. International Conference on Modeling, Simulation and Visualization Methods (MSV'11). Las Vegas, USA (July 2011)

[17] Hawick, K., Playne, D., Johnson, M.: Numerical precision and benchmarking very-high-order integration of particle dynamics on gpu accelerators. In: Proc. International Conference on Computer Design (CDES'11). No. CDE4469, Las Vegas, USA (July 2011)

[18] Hawick, K.A., Playne, D.P.: Modelling and visualizing the Cahn-Hilliard-Cook equation. In: Proceedings of 2008 International Conference on Modeling, Simulation and Visualization Methods (MSV'08). Las Vegas, Nevada (July 2008)

[19] Husselmann, A., Hawick, K.: Simulating species interactions and complex emergence in multiple flocks of boids with gpus. In: Proc. IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2011). pp. 100–107. Dallas, USA (14-16 Dec 2011)

[20] James, H.A., Hawick, K.A., James, C.J.: Teaching students how to be computer scientists through student projects. In: Young, A., Tolhurst, D. (eds.) Proc. Seventh Australasian Computing Education Conference (ACE2005). pp. 259–267. CSTN-011, Conferences in Research and Practice in Information Technology, Australian Computer Science Communications, Vol 42, Number 5, Newcastle, Australia (February 2005)

[21] Kar, A.: Skeletal tracking using microsoft kinect.

Methodology 1, 1–11 (2010)

[22] Krainin, M., Henry, P., Ren, X., Fox, D.: Manipulator and object tracking for in-hand 3d object modeling. Int. J. Robotics Research Online, 1–17 (2011)

[23] Kuhn, T.: The kinect sensor platform. In: Proc. Advances in Media Technology. pp. 1–4. Technische Universitat Munchen (21 June 2011), iSSN 2191-8015

[24] van Liere, R., Mulder, J.D., van Wijk, J.J.: Computational steering. Future Generation Computer Systems 12, 441–450 (1997)

[25] Linxweiler, J., Krafczyk, M., Tolke, J.: Highly interactive computational steering for coupled 3d flow problems utilizing multiple gpus. Comput. Visual Sci. 13, 299–314 (2010)

[26] Luber, M., Spinello, L., Arras, K.O.: Learning to detect and track people in rgbd data. In: Proc. Workshop on Advanced Reasoning with Depth Cameras, Robotics Science and Systems (RSS). University of Souther California (27 June 2011)

[27] Malone, T.W., Lepper, M.R.: Making learning fun: A taxonomy of intrinsic motivations for learning. In: Aptitude, Learning and Instruction Vol 3 - Conative and Affective Process Analysis, pp. 223–253. Hillsdale (1987)

[28] Martynov, I., Kamarainen, J.K., Lensu, L.: Projector calibration by inverse camera calibration. In: Scandinavian Conf. on Image Analysis (SCIA2011). pp. 536–544. No. 6688 in LNCS, Springer, Ystad Saltsjobad, Sweden (2011)

[29] McEwan, T.: Being kinected. ITNow 53(4), 6–7 (2011)

[30] Molenaar, G.: Sonic Gesture. Master's thesis, University of Amsterdam (October 2010)

[31] Nacke, L.E., Kalyn, M., Lough, C., Mandryk, R.L.: Biofeedback game design: Using direct and indirect physiological control to enhance game interaction. In: Proc. Computer Human Interaction (CHI 2011). pp. 103–112. Vancouver, BC, Canada (7-12 May 2011)

[32] Rambone, F.: Table Tennis Paddle Detection Library for Microsoft Kinect. Master's thesis, Universita della Svizzeira Italiana, Faculty of Informatics (June 2011)

[33] Schouten, B.A.M., Tieben, R., Ven, A., Schouten, D.W.: Computer Anaylsis of Human Behavior, chap. Human Behavior Analysis in Ambient Gaming and Playful Interaction, pp. 387–403. Springer (2011)

[34] Schwaller, M., Lalanne, D., Khaled, O.A.: Pygmi?creation and evaluation of a portable gestural interface. In: Proc. 6th Nordic Conf. on Himan Computer Interaction (NordiCHI 2010). pp. 773–776. Reykjavik, Iceland (16-20 October 2010)

[35] Staiano, A.E., Calvert, S.L.: The promise of exergames as tools to measure physical health. Entertainment Computing 2, 17–21 (2011)

[36] Steinicke, F., Benko, H., Daiber, F., Keefe, D., de la Riviere, J.B. (eds.): Proc. Special Interest Group Touching the 3rd Dimension (T3D). CHI 2011. ACM, Vancouver, BC, Canada (7-12 May 2011), iSBN 978-1-4503-0268-5/11/05

[37] Suma, E.A., Lange, B., Rizzo, A.S., Krum, D.M., Bola, M.: Faast: The flexible action and articulated skeleton toolkit. In: Proc. IEEE Virtual Reality. pp. 247–248. Singapore (19-23 March 2011), iSBN 978-1-4577-0038-5/11

[38] Sung, J., Ponce, C., Selman, B., Saxena, A.: Human activity detection from rgbd images. In: Proc. AAAI Workshop on Plan, Activity and Intent Recognition (PAIR 2011). pp. 47–55. San Francisco, USA (7-8 August 2011)

[39] Tang, M.: Recognizing hand gestures with microsoft?s kinect. Tech. rep., Stanford University, Dept. Electrical Engineering (2011)

[40] Turner, J., Browning, D.: Workshop on hci and game interfaces: A long romance. In: Proc. OZCHI 2010 : Design ? Interaction ? Participation. Queensland University of Technology, Brisbane, Queensland, Australia (22-26 November 2010 2010)

[41] Ulribe, A., Alves, S., Rosario, J.M., Fiho, H.F., Perez-Gutierrez, B.: Mobile robotic teleoperation using gesture-based human interfaces. In: Robotics Symposium, 2011 IEEE IX Latin American and IEEE Colombian Conference on Automatic Control and Industry Applications (LARC). pp. 1–6. Bogota, Volombia (1-4 October 2011)

[42] Wilson, A.: Using a depth camera as a touch sensor. In: Proc. Tabletops and Surfaces. pp. 69–72. Saarbrucken, Germany (7-10 November 2010)

[43] Xia, L., Chen, C.C., Aggarwal, J.K.: Human detection using depth information by kinect. In: Proc. 2011 IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 15–22. Colorado Springs, Colorado, USA (20-25 June 2011)

[44] Zollner, M., Huber, S., Jetter, H.C., Reiterer, H.: Navi - a proof-of-concept of a mobile navigational aid for visually impaired based on the microsoft kinect. In: Proc. 13th IFIP TC 13 Int. Conf. on Human-Computer Interaction (INTERACT'11). Lisbon, Portugal (5-9 September 2011)

RoboSoccer Project: Teaching Complex Concepts through Undergraduate Research

Andrzej Bieszczad
California State University Channel Islands, One University Drive, Camarillo, CA 93012, USA

Abstract - *RoboSoccer was an undergraduate research project funded by mini grants established by California State University Channel Island to allow faculty and students to engage in shared projects. The system that involved a number of students in a span of several semesters working on a variety of aspects such as robot control, visual reconnaissance, and game AI. The robots were controlled by laptop-based automated managers that were obtaining the status of the game from a visual analyzer of images taken by a camera hanging over the soccer field. The project not only provided an excellent vehicle to educate students in a variety of areas not covered by the regular curriculum, but also exposed them to the "real world" development environment. Demonstrations of the system were shown at several venues, including the CSUCI Research Fair and Business and Technology Partnership (B&TP) Annual showcase. The project made a very positive and long-lasting impression on the local community for both the Computer Science Program and for the university.*

Keywords: robotics, robots, undergraduate research

1. Introduction

The inspiration for the RoboSoccer project was RoboCup [1], a highly-recognized tournament in which teams of robots from the best research institution in the world compete in several categories. While the means of a CSU campus, a primarily educational institution, are far from the levels needed to compete at the world stage, with a bit of ingenuity in obtaining modest funds for the equipment, and to pay for the assigned time for the instructor (the author of this paper), it was possible to set up a project that was almost as exciting for the CSUCI students as participating in the RoboCup.

Most of the students were seniors, but the majority was not exposed to any of the technologies needed to implement the system. Therefore, the project involved a lot of learning, and the excitement was the single most important component that kept the students involved and committed.

2. Overview

The equipment included several Mindstorm NXT-based Lego robots, three laptops, a camera, two ladders with a bar to hold the camera, overhead lights, a modular foam soccer field, and a projector with a screen (see Figure 1). It was quite an undertaking to move, assemble, and then de-assemble all of the necessary pieces. As shown in Figure 2, the ladders were holding a bar with an attached camera that was taking pictures of the soccer field. The real-time video showing the current situation on the field was projected on the screen through the projector attached to the image analyzing laptop. The strong lights minimized the variance in lighting

Figure 1: Part of the RoboSoccer team with the equipment.

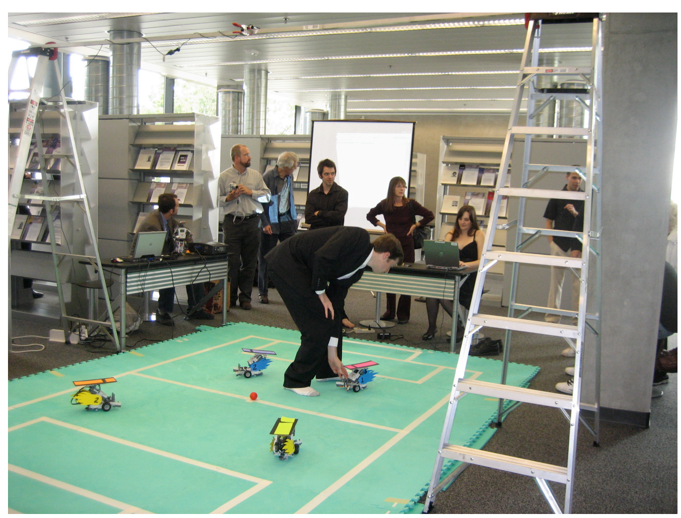


Figure 2: RoboSoccer setup.

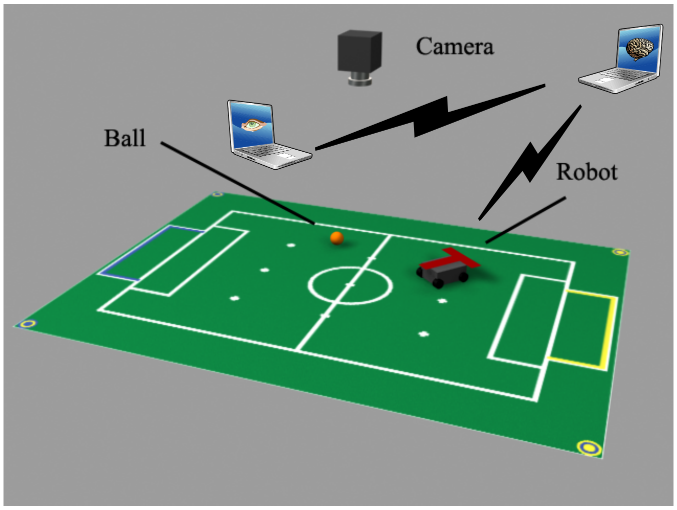


Figure 3: RoboSoccer functional architecture.

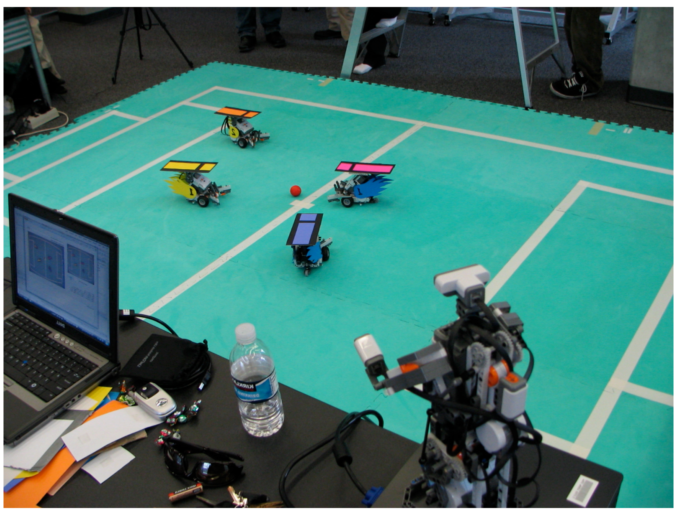


Figure 4: RoboSoccer soccer field.

conditions between various venues (e.g., eliminating shadows).

3. System Configuration

Two laptops running team management software were behind the goals, while the third getting the image feed from the camera was on a side. The role of the former was to receive image analysis data from the latter, and transform that knowledge into commands for the Lego players (Figure 3). The camera was connected through a USB cable to the image analyzing laptop. Wifi was used to communicate between the laptops. The communication with the Lego robots was carried over a Bluetooth link. For identification purposes, the Lego players were color-coded with 'hats' (Figure 4), as were the goals.

4. System Architecture

The RoboSoccer system was structured as four sub-projects:

- The first part, image sensing, determined the current state of the game (positions of the players and ball).
- The second, the communication module, handled the communication between the sensing unit and two team control units, and then each team control unit and the robot players.
- The third, the the control unit, was responsible for determining what moves the robots should take based upon the information from the sensing unit.
- The last piece was the individual skills of the robots themselves, such as walking, turning and kicking.

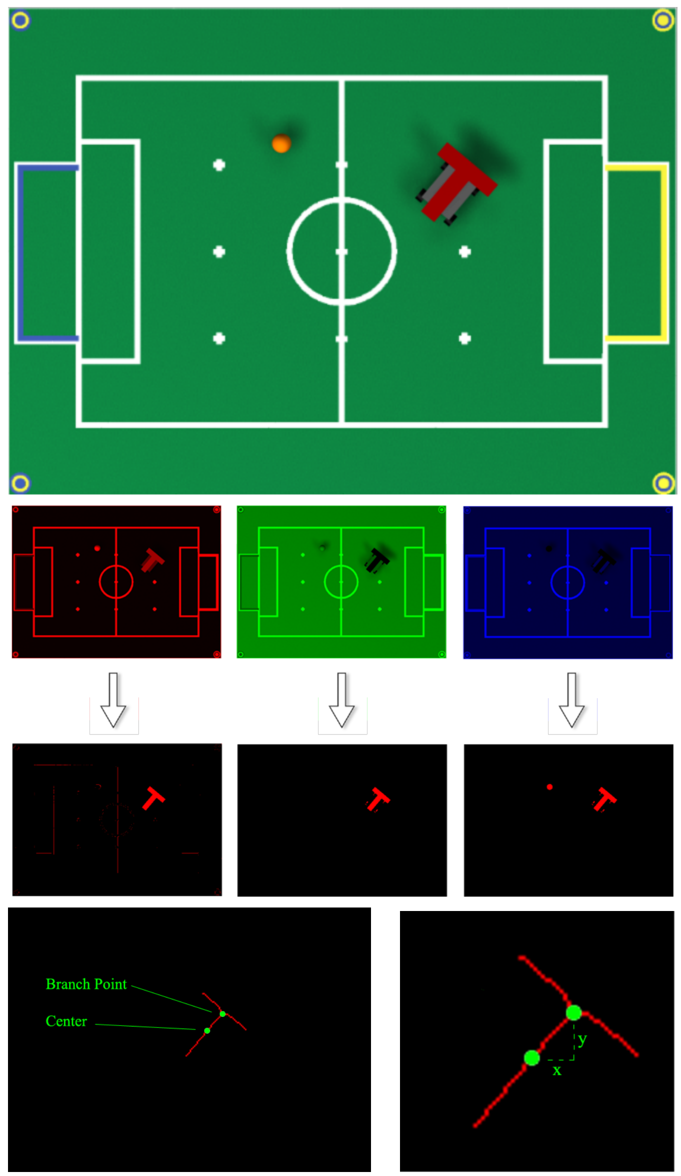


Figure 5: RoboSoccer Image Sensing.

4.1. Image Sensing

One of the critical elements of creating a robot soccer team was designing a vision system for the robots. The RoboCup system assumes a centralized view of the game, so for this project we relied on a single camera serving as the eye for both teams, and subsequently for every robot on the soccer field.

An overhead camera (Figure 2) captured the view of the soccer field (top image in Figure 5). In the figure, the soccer ball is orange, and the robot is wearing a red 'hat'. Other robots wore hats of different colors. Each object's distinguishing color was known ahead of time and kept in a file for reference for the image analysis tools.

Each image component is binarized as a result of a process called thresholding [2]. In the thresholding operation, pixels that are below and above a specified range of intensity are set to a background color (in this case black), while pixels that are within the specified range are set to the foreground color (in this case red). The specified range has an upper and a lower bounds (threshold levels) that are based on the reference file that contains each object's color. The thresholding process was repeated for each object that we needed to look for.

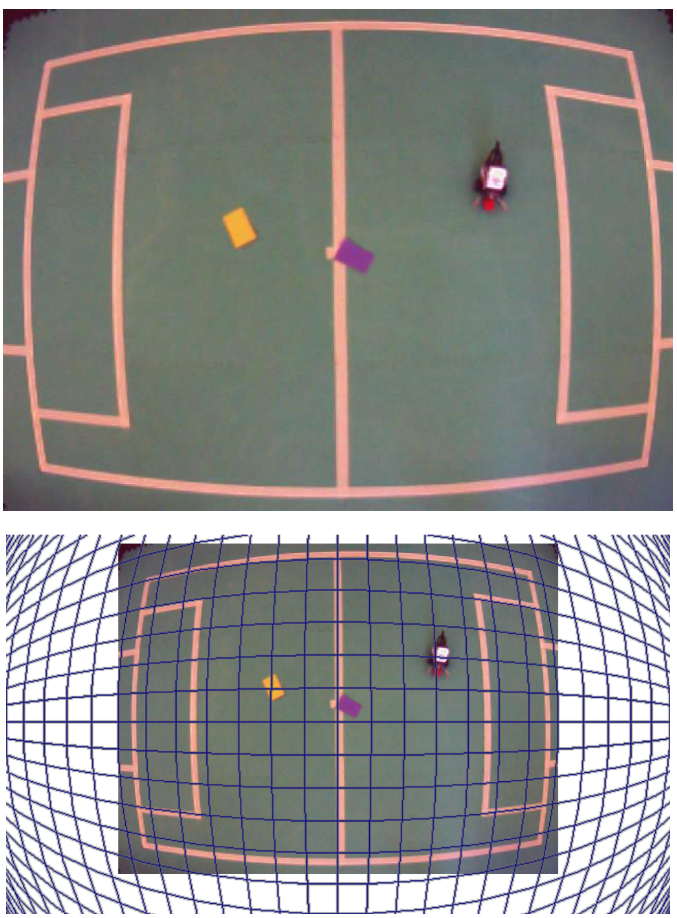


Figure 6: MindSoccer image distortion by wide-angle lens.

The three thresholded images were combined using a bitwise AND operation on the pixels. This results in a single binary image that contained the object that we were looking for.

The image shown in the lower part of Figure 5 is a reduction of the original image; it only consists of the robot's identifying 'hat'.

Further image analysis on the above image measured the single object (contiguous pixels), and returned its center location in image coordinates. That was the position of the robot.

The images underwent a second image analysis technique known as skeletonization [3]. This process reduces groups of contiguous pixels to a single width spine that maintains the general shape of the pattern. Depending on the initial shape of the pattern, there could be several branches. The shape of our hats reduced the number of branches to three. The image analysis tools allowed us to locate the coordinates of the branch intersection of the skeleton. By knowing the location of the branch point and the location of the center of the object, we could determine the other important element of the game: the orientation of the robot. The following formula gave us the orientation of the robot:

$$\theta = \arctan \frac{y}{x} \qquad (1)$$

4.2. Dealing with Image Quality

The actual images that we received from the camera were distorted because we were using a wide angle lens (to capture the whole field from a relatively low height). This type of distortion is known as a barrel distortion.

The potential problem with this type of distortion is that if we told the robots to move along a line according to the pixel coordinates, then they would actually end up traveling along a curved path.

In order to compensate for the distortion, the pixel coordinates can be mapped onto a sphere. By placing the origin at the center of the image, an x and y coordinate can be converted as follows:

$$x' = \theta = \arcsin \frac{x}{\sqrt{r^2 - y^2}} \qquad (2)$$
$$y' = \theta = \arcsin \frac{y}{\sqrt{r^2 - x^2}}$$

This put the coordinates into sphere space, which in turn could be linearly mapped to soccer field space. From the

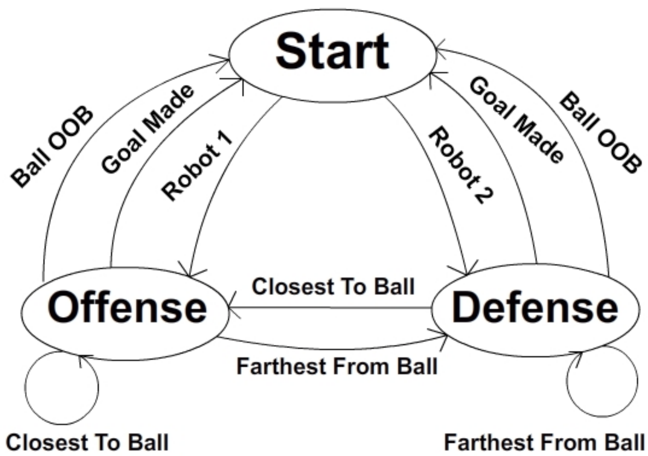


Figure 6: RoboSoccer team control manager's logic.

sensing module, the coordinates and angles of the robots and coordinates of the ball were then passed to the logic module.

5. Logic Module

As the information about the current game state was received from the vision system, each team control manager processed the data and determined the optimal strategy and movements to make for its two robot players. Once these movements were calculated, the results were communicated, via a Bluetooth connection, to the individual robots.

The main algorithm of the control unit can be described by a Finite State Machine (FSM) with three states (Figure 6): Start, Defense, and Offense. The team play was organized around the following principle: Since there were two robots per team, either both robots on a team were in the Start state, or one was in the Defense state while the other was in the Offense state.

At the beginning of the game, assuming that both robot players started at equal distances from the ball, one of the robots ("Robot 1" in figure 6) was arbitrarily set in the Offense state. 4 4If not in the Start state of the game, the Offense state was assigned to whichever of the team's robots was closer to the ball. By default, the other robot ("Robot 2" in Figure 6) not assigned Offense would assume Defense. Subsequently, the robot followed the transitions of the FSM show in Figure 6. As long as a robot was closer to the ball than the partner, it stayed in the Offense state; if that changed and the partner got closer to the ball, the roles were exchanged. After a goal was scored, both robots went back to the Start state. The same action was undertaken if the ball was out of bounds (OOB). When the ball was placed back in the field and a starting signal was issued, the cycle would repeat.

5.1. Defense System

Upon receiving the state from the sensing mechanisms and deciding which robot would act as the offensive player and which would be the defensive player, the control unit then applied the following strategy:

1. Calculate the distance from the closest enemy robot to the ball. It is the red line in Figures 7 and 8.
2. Using this distance as a radius, create a circle around the Defense robot (as shown in Figures 7 and 8).

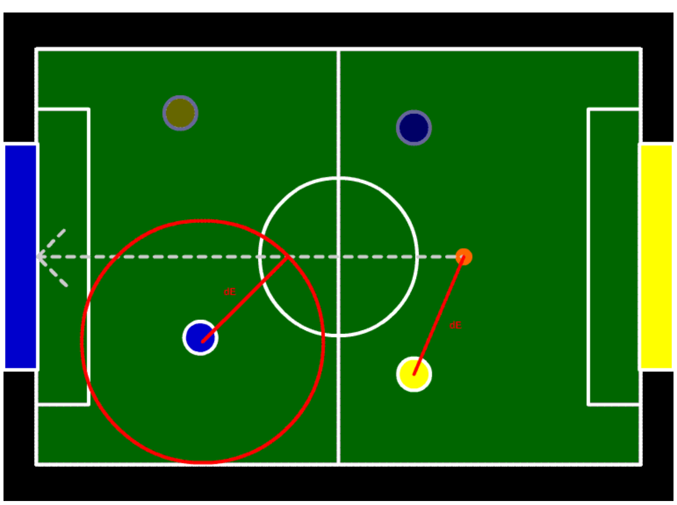


Figure 7: Branch A/B.

The robot is not in a position to jeopardize the offensive robot. Here the distance to the enemy robot is calculated and the closest point is chosen as the point to block at.

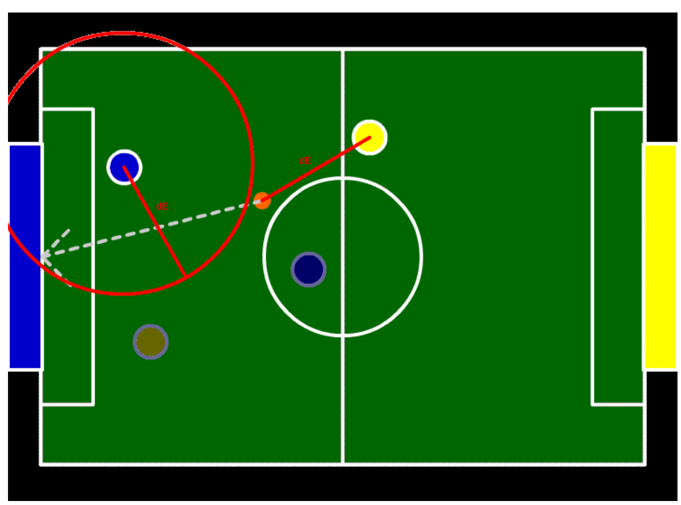


Figure 8: Branch B.

The robot is in a position to jeopardize the offensive robot: Here the desired spot calculated by the intersection points would cause the robot to stand where the offensive robot needs to go. Instead, he takes the spot on the mid point of the enemy's projected shot to block it and back up his offensive player.

3. Create a "predicted shot" line which estimates the shot the enemy robot will try to make (dashed line in Figures 7, 8, and 9). This ‘shot’ was a line connecting the center of the ball with the center of the friendly goal (the goal which the robot was trying to defend).

4. Figure out where the circle and line intersect. This could be at 0, 1, or 2 points. These three results created three branches of actions the robot might take.

Branch A: The circle and line intersect at one point.

- **Response:** The robot moves to that point.

Branch B: The circle and line intersect at two points.

- **Response:** The robot moves to the point of intersection which is closer to the ball.

In both branches A and B, one more check was made to ensure the robot would not get in the way of the offensive player. If the distance from the desired destination to the ball was less than a fixed amount (equal to the measured length of the robot), the robot would instead move to the mid point of the projected path of the enemy shot. That was to avoid fouls.

Branch C: The circle and line never intersect.

- **Response:** The robot moves towards the ball in the direction that will get it closest in the shortest amount of time.

The images in Figure 7, 8, and 9 show what happens during each branch of the routine. In the images, the darker blue robot is the robot on the control unit's team which is currently

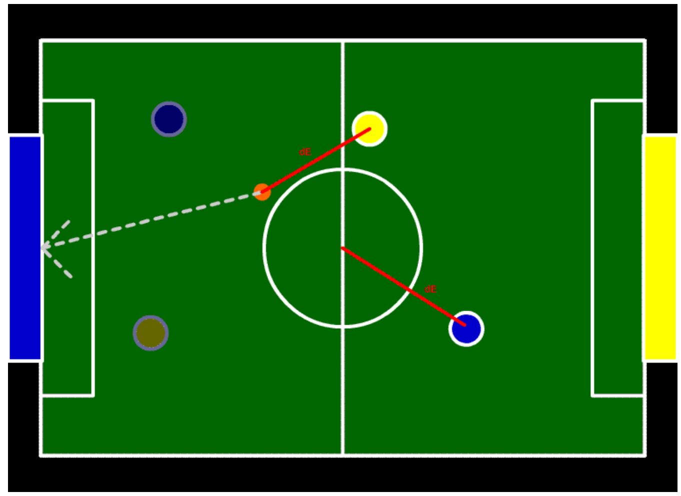


Figure 9: Branch C.

The robot cannot get in a position to block in time. Here the robot tries to get as close to the ball as it can.

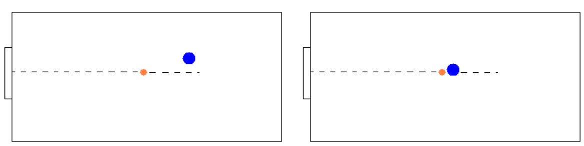
in offensive mode. The darker yellow robot is the robot which is in the opposing teams defensive mode. The red line is the distance from the closest enemy robot (the one in the offensive mode) to the ball. The red circle is the circle created with radius of the distance from the robot to the ball. The hashed grey line is the predicted trajectory of the shot the enemy robot will try to make.

4.1. Offense System

At any given time one robot on a team should be assigned to the Offense state. That role was always assumed by the robot closest to the ball. As seen in the FSM in Figure 6, that condition was checked continuously, and the roles were swapped as necessary.

The following was the strategy for the offence. First, a line was drawn through the target on the enemy goal and the ball. In the examples in Figures 10, 11, 12, and 13 it is assumed the target is the center of the goal, but potentially it could be another point (for example, a point to pass the ball to another player).

If the robot is above the line (in image geometry; as shown in Figure 10), the robot approaches the ball from above and from behind. The parameters for the exact position were determined experimentally by measuring the physical



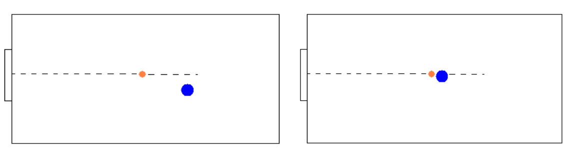


Figure 11: Robot is below the line.

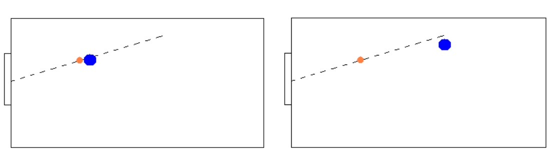


Figure 12: Getting into a shooting position at an angle.

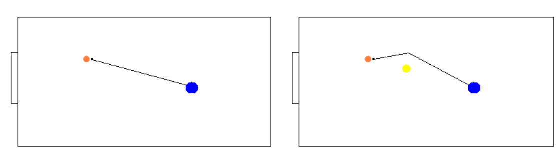


Figure 13: Testing for obstacles.

distance that the robot must maintain from the ball to properly kick it. That depended on the robot physical appearance. A number of designs were used in an effort to get best shooting properties; accordingly, the computation parameters were modified.

If the robot is below the line (Figure 11), the robot moves from behind and from below the ball. The purpose of the line is made clearer when a different ball position is used (Figure 12). In this case the robot's y coordinate is higher than the ball's y coordinate, but since the shot is at this angle the robot moves below the ball (with respect to the ball-target line) to shoot.

If the robot is on the wrong side of the ball, it must go around the ball without hitting it. Otherwise it might accidentally push the ball towards its own goal. Again, the physique of the robot needs to be accounted for.

There is also the issue of other robots blocking the path. To avoid these obstacles some more geometrical reasoning must be done. A line is drawn from the robot to the point where the robot intends to travel. Both players on the opposing team are checked to see if they are on the path. The peer player needs not to, since if it was there, then it would be it who was in the Offense state.

There might be two cases. If an opponent is on the line but not blocking the robot by being too close to the ball, the robot needs to construct a new path to the point that goes around the obstacle (Figure 13). If the opponent is standing directly at the point that the robot needs to move to (and shoot from), the Offense robot selects a new target point just behind the Defense robot (again, avoiding fouls). There is a good chance that the defending robot will switch the role to Offense, and as such will try to move on the opposite side of the ball to be in a position to attack. That may create an opportunity for the currently attacking robot to approach the ball and shoot.

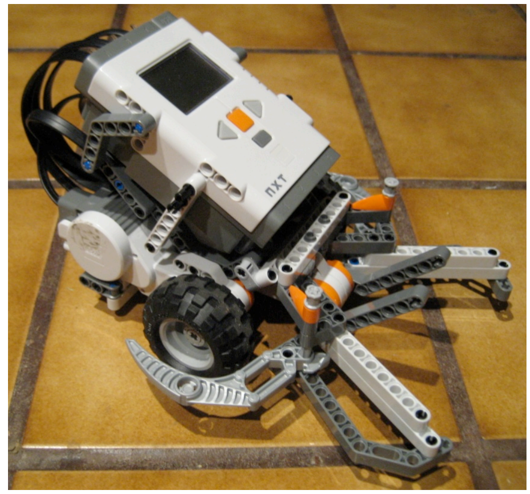


Figure 14: RoboSoccer player.

Once the robot had a clear path, it walked forward along the trajectory for a certain distance recomputing in real-time the next situation.

The important thing to emphasize here is that this computation was done in real-time. At each stage, a new state of the game was received, analyzed, and than translated into commands for the robot. In that way, the robot continuously corrected its position, so it was always trying to orient its front towards the goal. Eventually it reached a point close enough to the ball to shoot. Some final correction to robot orientation could be made just before shooting to ensure a clear shot at the goal.

5. Robot Software

Lego’s NXT computer was used to control the robot players (Figure 14). The embedded software developed using RobotC platform [4] included implementation of the protocol to

Figure 15: Tight collaboration on the project.

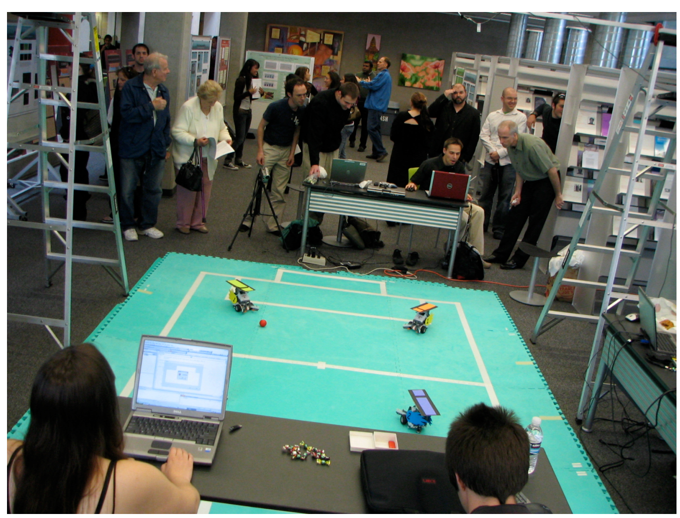


Figure 15: RoboSoccer showcase.

receive commands from the team manager over a Bluetooth link, as well as a number of robot actions realized by sequences of control settings to the robot engines. Two engines controlled robot wheels, and one the shooting lever. The robot could turn by applying various speed to the engines powering the wheels. The arms in the front of the robot (white in the figure) helped with "scooping" the ball that was then pushed by the lever (gray) going down with certain speed related to the distance to the target.

6. Communication

There were two communication links implemented. The first, over an ad-hoc WiFi network, allowed distribution of the game state (positions and orientation of the players of both teams and the position of the ball) to team managers. Both teams received the same information. The communication was maintained over a TCP session.

The other link, between the team manager and the players was implemented using the RFCOMM profile of Bluetooth. That profile simulates a serial port, so from the perspective of a team manager, it was just communicating over the serial port. The messages included both low-level and high-level content. Low-level content included direct values for the engines that ultimately controlled the actions of the robots. High-level messages were used to activate more complex actions implemented in the robots; e.g., "kick".

7. Educational Benefits

Computer Science curriculum is one of the most challenging. In addition to the core courses in the field it includes plenty of required Math and Science courses. At CSUCI, CS students need to take 123 units to graduate. Out of that number, almost half are in non-major subjects (GE and other areas). That requirement leaves room for only a handful of electives in the program. That is not sufficient to teach students everything that is required to implement such complex systems as RoboSoccer. Therefore, using other educational vehicles is of paramount importance in graduating professionals ready to take on challenges of the "real world". As we described elsewhere [5], capstone projects constitute one such mechanism. They tend to be smaller and individualized usually, however. The project reported in this paper was run under the auspice of "undergraduate research" and "student-faculty research", two vehicle that CSUCI put in place to accommodate such needs.

The project required a level of collaboration (as seen in Figure 15) that is unmatched in any regular course offering. The time spent on the project handsomely exceeded the time usually expected in similarly weighted offerings. Just setting up and tearing down the system was a 2-3 hours endeavor. The same level of commitment was required from the instructor. In addition to the educational supervision and guidance, the limited means translated into trips to stores and warehouses to purchase the equipment as well as using a personal car to transport the equipment.

The roles in the team were assigned along non-competitive lines; namely, we did not have a team A and a team B to compete against each other. Instead, the team was divided along the functional lines. Therefore, we had an image processing team, a communication team, a defense team, an offense team, and the robot embedded software team. That allowed the students to focus on collaborative perfection of the system components.

8. Conclusions

It is safe to state that the experience for the involved students was just great. While every student specialized in a particular area, they shared the knowledge and experiences with others. They all saw the power of collaboration and the importance of every one of them in the ultimate success of the project.

The system was demonstrated at a number of venues where enthusiastic audience was very glad to see that "such things can be done in Ventura County" (see Figure 16). The presentations were done mostly by the students, so there was yet another educational element of the project: practicing communication skills.

9. Acknowledgments

The author would like to acknowledge the efforts of the student members of the RoboSoccer team: Jennifer Morrison, Dawud Davies, Glenn Deschmidt, Brian Gilreath, Robert Kiffee, Shahab Lashkari, Cole Rieger, Nick Winters , and Charles Zilm as well as professor Anna Bieszczad.

It is worth mentioning that after graduating none of the students had any problem with finding a well-compensated creating job as a computer professional.

10. References

[1] http://www.robocup.org/

[2] Shapiro, Linda G. & Stockman, George C. (2002). "Computer Vision", Prentice Hall.

[3] Rafael C. Gonzalez, Richard E. Woods (2007). "Digital Image Processing", Prentice Hall.

[4] http://www.robotc.net/

[5] Bieszczad, A., (2011) "A Microcosm for Effective Undergraduate Education of Computer Science", Proceedings of the 2011 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'11), Las Vegas NV, July 2011, CSREA Press, pp. 194-200.

Research and Teaching with Remo:

Student research projects and teaching for and by undergraduate students

Dagmar Monett*
Computer Science Division
Department of Cooperative Studies, HWR Berlin
Berlin, Germany
Dagmar.Monett-Diaz@hwr-berlin.de

Mario Sänger
Student trainee at ITSO GmbH and
Computer Science Division, HWR Berlin
Berlin, Germany
Mario@itso-berlin.de

Abstract—**In this paper we give insights about our experiences in both undergraduate research and supervision of Computer Science students. We focus on how undergraduate students support teaching when this is included in their student research projects. Concrete settings for an Artificial Intelligence course give challenging ideas for further action.**

Keywords-undergraduate teaching; student research; sandwich principle; supervision

I. INTRODUCTION

Delegating responsibilities to the students and making them a part of the decision-making process seems to be a common practice in both student research and teaching. Even so, combining both of these aspects in a successful way is not as simple as expected when considering research and teaching together.

Berrett comments in [3] that teaching is no longer a distraction from research, since a recent study demonstrates that students involved in both have better improvements in research skills than those conducting only research. Although these findings were analyzed for graduate students teaching in undergraduate courses, we believe the same applies for their undergraduate mates. Certainly, no few skills, attitudes, and methodologies are important for both research and teaching, disrespecting the awarded level.

Would the relationship between research and teaching be strengthened when students incorporate in their research teaching those contents they research in? What happens when subjects for research proposals are discussed with undergraduate students in advance and they include developing computer-based software for improving teaching? How do co-teaching models [5, 9] for instruction can be applied considering student research projects and teaching for and by undergraduate students?

We have answers to these questions regarding our experiences in an Artificial Intelligence (AI) course at the Berlin School of Economics and Law (BSEL).[1]

* Contact author.

[1] The BSEL is a University of Applied Sciences. In German: HWR Berlin.

II. COMPUTER SCIENCE AT THE BSEL

The Computer Science Division from the Department of Cooperative Studies (DCS), former Faculty of Company-Linked Programs, at the BSEL, offers since 1993 a solid education in the field of Computer Science, from software development to systems support [1]. A clear focus is put on applied skills, since students are part of a special cooperative program that combines full-time classroom study with regular practical on-the-job trainings at business enterprises. Upon successful completion of their studies, students graduate with 210 ECTS-credits[2] have access to all the classic professions of a computer scientist, with both deep theoretical business knowledge and company-based practical working experience.[3]

A. *Study Research Projects*

Undergraduate students from Computer Science at the DCS have to accomplish two study research projects (SRP) as part of their curricula. Such projects allow students to research in a specific topic that can be proposed by educators from the BSEL, or by their direct supervisors at the training companies, or, in the ideal case, by both. With no doubt, the last constellation encourages applied research in a stronger way.

SRP duration takes a maximum of six months in the 4th and other six months in the 5th study semester. Each semester comprises both a 3-month theoretical and a 3-month practical phase. During the theory periods, students profit from an intensive, on demand coaching from their DCS academic supervisors because they are almost always at the university. During the practical phases, the contact is kept mainly through email, since students are at their training companies. There, they profit much more from their company advisors and real-live practice.

Students get 14 ECTS-credits with each SRP. They should learn how to work on a subject-specific or cross-disciplinary task, independently. Although research in a new topic or field is encouraged for the second SRP, an in-depth analysis of a

[2] European Credit Transfer and Accumulation System. In Germany, one credit point is equivalent to 30 hours of study.

[3] See more about the Department of Cooperative Studies at http://www.hwr-berlin.de/en/department-of-cooperative-studies/. The enterprises spectrum comprises more than 650 companies from Berlin and Germany.

complex topic, as natural continuation of the first SRP, is also possible.

Partial results should be presented in the context of a short talk and in a comprehensible, stylistically convincing form. It takes place on the *projects' week*, where a deep discussion is expected and other classmates or interested guests can be part of the audience. At the end of the practical phase, students should submit a written report (like a small thesis of about 4000 words or 25 pages long) for evaluation by a BSEL professor.

For evaluating a SRP, BSEL professors weight the following criteria:

a) Theoretical considerations (8 points): Literature review, decision making, discussion of implementation approaches, and argumentation.

b) Practical realization (32 points): Design, practical application of technical knowledge, selection and use of tools, and implementation.

c) Initiative and commitment (8 points): Original ideas, ability to comment and to criticize, and independence.

d) Working style (8 points): Work organization and style, cooperation within the team and with other persons.

e) Subject-specific content (16 points): Definition of constraints and termini, evaluation and analysis of methods, description of goals, time management in reaching these goals, and discussion of results.

f) Work structure and organization (8 points): Outline, structure, style, and bibliography.

g) Oral presentation (20 points): Selection of materials and information density, use of media, verbal expression, and time management.

Each criterion deals with several aspects but is not limited to them; they serve as an orientation to the BSEL professors. Criteria related to the written report, i.e., from *a* to *f*, represent 80% of the SRP grade. The last criterion completes the final evaluation on 20%.

In the following sections, we give insights from our experiences supervising SRP and how undergraduate students strengthen both research and teaching, based on an Artificial Intelligence course at the DCS.

III. The AI Course

The AI undergraduate course is an optional 7-credit course and is part of the curricula in the 5th study semester. Students learn there the focus, history, termini, and applications in AI. They develop theoretical and practical basics on modeling and representing knowledge, as well as on processing it using both problem solving and learning strategies, from knowledge to agent-based systems as well. This is why the AI course comprises two main areas: autonomous agents and multi-agent systems, and knowledge-based systems. We have located subjects for SRP mainly in the former area since 2008. Topics for SRP in the latter have been centered on the inference in expert systems.

The book from Beierle and Kern-Isberner [2] on knowledge-based systems is followed for introducing the theory and practice on expert systems in our AI course. Special attention is put on the inference mechanisms using forward and backward chaining algorithms. The need to better explain these reasoning algorithms to the students gives us the idea for a SRP's subject: how to visualize their functioning, as well as studying the impact of visualization in the classroom. We even think beyond these limits and ask ourselves, what happens when the student self teaches these contents, too! This way, we define the tasks, goals, and general requirements for a SRP, including the implementation of a software program.

The second version of the resulting software product is Remo (Rule-based expert system modeler), a program for modeling rule-based knowledge and for analyzing reasoning algorithms' functioning. We will describe Remo further in the next section. Its main features comprise the definition of condition-action rules, the graphical representation of rule-based knowledge, the visualization of inference processes, and a step by step execution of both data- and goal-driven algorithms. We used it successfully in the AI course, which we will also explain in the following section.

IV. Remo

Remo was entirely developed by Sänger[4] as part of two student research projects [10, 11]. It was of special importance for the AI course (and as SRP, too) because of the lack of adequate software support for both visual modeling knowledge and for analyzing inference algorithms for rule-based expert systems. To our knowledge, there is no computer-based software tool available for teaching these contents in depth.

The first SRP [10] was concerned with the development of basic functionalities for Remo, like a graphical editor for modeling rule-based knowledge, the implementation of algorithms for backward and forward chaining inferences, and their debug visualization. During this SRP, the student had to run through the four major phases of any software development process: analysis, design, implementation, and testing. The analysis phase was devoted to the collection of requirements for the application. In the design phase, a concept for the implementation of the requirements was conceived, i.e., relevant solution methods were selected and evaluated. A model-driven approach using the Eclipse Modeling Framework (EMF) [7, 12] and the Graphical Modeling Framework (GMF) [6] was used. Afterwards, the concept was implemented and tested. During the implementation phase, there was much attention paid to a modular component structure and a good extensibility of the application. Therefore, each main feature, like the graphical editor or the inference algorithms, was implemented as a separate plug-in [4]. Moreover, the extension point mechanism of the Eclipse platform was used to provide an easy way to extend the application, e.g., for adding new inference algorithms.

[4] Mario Sänger currently attends the 6th and last semester in the Computer Science career at the Computer Science Division, DCS.

In the second SRP [11] additional functionalities were developed. The application was extended with different export features, which allow the user to transform the graphical model of the application into other (textual) representations. Two transformation options were provided: the user can transform the model into simple "IF … THEN …" rules or into Prolog syntax. The exported Prolog model works with various Prolog implementations like SWI-Prolog [13]. It is also possible to validate the created models, e.g., to check whether each element has a unique name and whether it is connected to other elements or not. Resulting error and warning messages are shown at the top left corner of the graphical elements. Furthermore, an event log console was developed; it increases transparency of algorithms functioning to the user. Moreover, the use of Remo in the AI course was also a major activity in the second SRP.

Fig. 1 shows an excerpt of a knowledge base created with the graphical editor of Remo. Ellipses represent facts and rectangles are conclusions or further conditions in rules. Check marks on facts indicate that they are true, i.e., that they are evidential facts used for reasoning. Cross marks specify that a fact is false. If a fact node has neither a tick nor a cross, then the state of the element is unknown, and it will be determined during the inference process according to user information. Rules relate conditions to actions. A conjunction symbol in a rule node means that all connected nodes from the left should be satisfied as preconditions in order for the action part of the rule to be executed. If a node has a disjunction symbol, only one precondition needs to be satisfied for the rule to be executed.

The given excerpt is taken from an exercise which was created by Mario Sänger and that deals with the evaluation of visiting requests in a museum. It shows that, for a valid visiting request (VVR), both a full list of participants (PL) and a guide by a trustworthy group leader (GL) should be satisfied. A trustworthy group leader is either a teacher from a local school (LT) or a person who has already guided a group in the museum (AG).

Fig. 2 shows the debug console of Remo. With its help, the user can better follow the inference algorithms' functioning. This allows a gradual walk through the inference process. The console works together with the graphical editor, i.e., the currently focused node in the debug console is simultaneously highlighted in the editor by filling its background to gray (see also Fig. 3.)

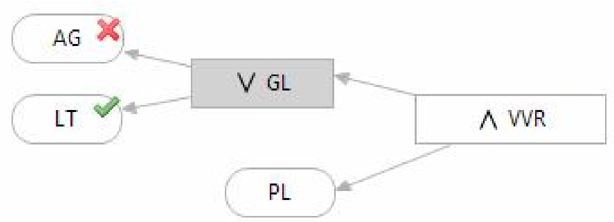


Figure 1. Example of rule-based knowledge in a graphical representation using Remo.

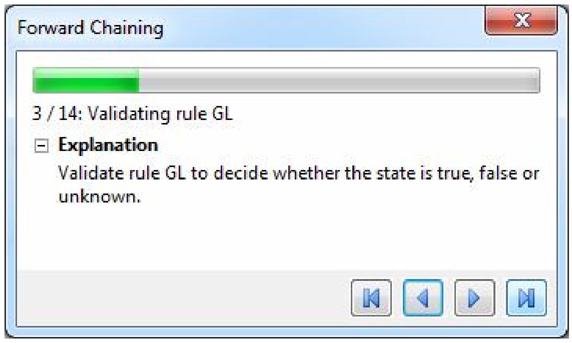


Figure 2. Debug console for the forward chaining inference algorithm.

Fig. 3 shows the complete user interface of Remo when the execution mode is activated. Further views, like a properties view of the selected node or an outline view, are located at the bottom in the user interface.

During both research projects the student gained valuable experiences regarding the software development process and the presentation of project results in suitable form. These experiences are also very useful in a variety of tasks and projects at both the BSEL and the business enterprise students are enrolled at during their dual studies. The following section discusses one of these applications, i.e., the use of Remo in teaching.

A. *Remo in the classroom*

AI contents related to expert systems were prepared and taught in a collaborative way, profiting from the benefits of co-teaching. The supervisor and the undergraduate student divided responsibilities for planning, for teaching, and for moderating learners' work on exercises. A combination of co-teaching approaches like "one teach, one observes" and "team teaching" was followed.

Furthermore, we applied a sandwich principle consequently in our lecture on expert systems, i.e., we combined theory units of about 20 minutes with practical examples and exercises with Remo in order both to motivate and to increase student attention. For this, separate meetings and email communication between the undergraduate student and the supervisor were planned and carried out. Such discussions strengthen the student's preparation on the contents to teach (Remo related, but also about inference in expert systems). They helped in designing course slides, examples, and exercises for the lecture as well, the latter especially conceived to support active learning in the classroom.

The student considered the following schedule when teaching new contents to the other students:

1) Remo I: Introductory part with focus on the SRP subject, on the main features of Remo, and on how to work with it. Duration: 10 min. Media and materials: lecture slides, data projector, Remo.

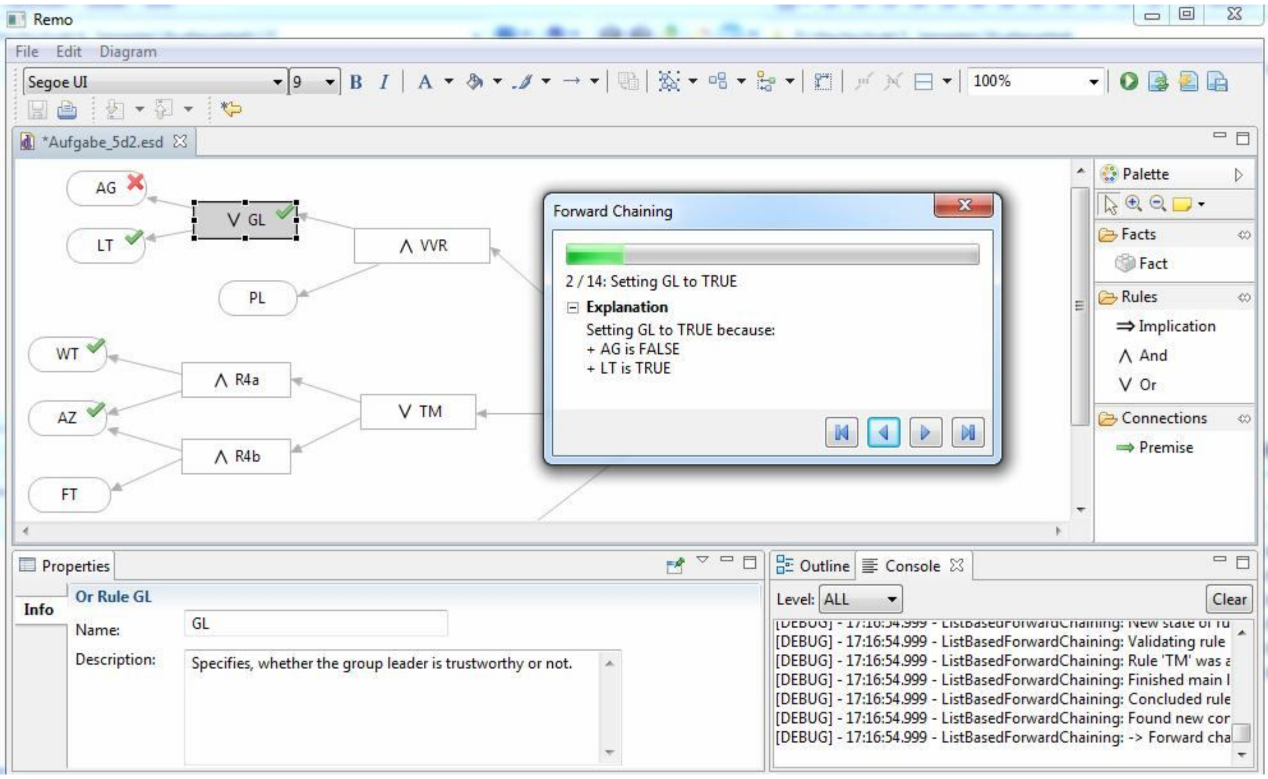


Figure 3. Remo's user interface in execution mode.

2) Forward chaining: Theory part with focus on the forward chaining reasoning algorithm and on its functioning. Duration: 10 min. Media and materials: lecture slides, data projector.

3) Exercise forward chaining: Practical part with exercise on forward chaining, first manually and then with Remo. Duration: 30 min. (20 min. solving; 10 min. discussing). Presentation of solutions and discussion: in plenum. Media and materials: individual materials, blackboard, Remo, data projector.

4) Remo II: Theory part with focus on implementation aspects in Remo and on related technical details. Duration: 5 min. Media and materials: lecture slides, data projector, Remo.

5) Backward chaining: Theory part with focus on the backward chaining reasoning algorithm and on its functioning. Duration: 15 min. Media and materials: lecture slides, data projector.

6) Exercise backward chaining: Practical part with exercise on backward chaining, first manually and then with Remo. Duration: 25 min. (15 min. solving; 10 min. discussing). Presentation of solutions and discussion: in plenum. Media and materials: individual materials, blackboard, data projector, Remo.

7) Integrated exercise: Practical part with complex exercise integrating all contents together, in the form of a mock exam. Duration: 80 min. (30 min. solving; 50 min. discussing). Presentation of solutions and discussion: in plenum. Media and materials: individual materials, blackboard, data projector, Remo.

8) Questionnaire to evaluate Remo and teaching: Duration: 10 min. (see next Section for details.)

Schedule points one to six were accomplished in the first day the undergraduate student taught. The last two were accomplished a day after, at the beginning of the next lecture. We prepared also several exercises for individual learning and distributed them in the classroom, together with other teaching materials. All of them were additionally uploaded to the BSEL's E-learning platform, i.e., to the Moodle[5] course site.

Short discussions and deep reflection about the schedule completion and other organizational issues were carried out before, during (in the breaks), and after the lectures. Through them, a dynamic adaption of the already planned course's guide was possible, depending on the current development and on the invested time. Feedback and coaching was also offered to the undergraduate student for improving and refining didactic methods to use and in order to reduce uncertain feelings about own teaching techniques.

Several examples and exercises were carefully selected from [2]. The undergraduate student also had the possibility to design new examples and exercises. This was included in his evaluation, i.e., the ability to create new, adequate content to teach. There also evaluated his teaching methods, his answers to questions, his preparation in the field (AI contents to teach were new to him), as well as additional teaching materials prepared by him.

[5] Modular Object-Oriented Dynamic Learning Environment, a free source course management system.

V. Evaluating both Teaching and Remo

At the end of the AI course part taught by Sänger, we distributed a short survey for evaluating both teaching by students and Remo as well. The questionnaire's primary objective was to get feedback from students mainly about the impact and use of Remo, as well as about using undergraduate students for teaching contents in the AI course.

The survey questions were proposed by the student and they were discussed in advance with the supervisor. Conceiving, applying, and analyzing survey results were also included as part of his evaluation in the AI course.

A. Content

Students were asked to answer up to eleven questions divided in two major areas: teaching and Remo. Table I shows the main contents and the questions' scope, as well as the answer types that were considered.

B. Results

Although the survey was applied to a relatively small group of students (13 attended the AI course), its results were very inspiring to us.

Testing the application with the hope of getting valuable feedback for Remo improvements was, as expected, positively welcomed by the students. Their opinion and suggestions (during the lecture and after processing the survey results) helped us to further complete a to-do list with new features, changes, and necessary modifications to Remo. For example, while visualizing reasoning algorithms with Remo was rated from "good" to "very good" by 9 of 12 students, the usability of the user interface still needs to be improved: only 4 students rated the Remo usage as "good" or "very good"; 7 evaluated it as "satisfactory", instead.

TABLE I. Questionnaire Content and Scopes

Nr.	Main content	Type[a]	Focus on
1	Teaching by undergraduate students.	Rating scale	Teaching
2	Quality of the lecture when students teach.	Rating scale	Teaching
3	Sandwich principle and lecture's structure supporting the learning process.	Rating scale	Teaching
4	Using sandwich principle and lecture's structure in other courses.	Yes/No	Teaching
5	Suggestions for teaching improvement.	Open-ended	Teaching
6	Remo supporting the learning process.	Rating scale	Remo
7	Remo usage.	Rating scale	Remo
8	Suggestions for usage improvement.	Open-ended	Remo
9	Visualizing reasoning algorithms with Remo.	Rating scale	Remo
10	Suggestions for visualization improvement.	Open-ended	Remo
11	Suggestions for Remo improvement.	Open-ended	Remo

a. All ratings based upon a six-point rating scale.

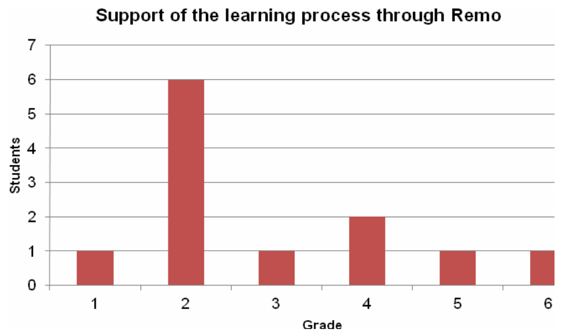


Figure 4. Results to the question "To what extent Remo supported your learning process in the subject field?" from 1 (very much) to 6 (no support).

Suggestions in this concern were introducing shortcuts for the creation of facts and rules and drag-and-drop functionalities, for instance. Nevertheless, more than half of the students (7 of 12) felt that the support of the learning process by the application was "good" to "very good" (see the bar chart in Fig. 4 for detailed results).

Regarding teaching, 11 of 12 students reported that teaching by undergraduate students made the lecture more interesting. That both the sandwich principle and the lecture's structure support the learning process was also in corroboration with the respondents. Furthermore, also 11 of 12 students wish such a course design in other modules of the curricula.

VI. Lessons Learned and Discussion

When students teach students, all they share is a common language that helps them to understand new contents "differently," often even better. On the other hand, explaining others any content, preparing it in advance, and asking questions from others is a very powerful way for individual learning. Furthermore, classmates' questions flow more openly and unbiased because students mostly know each other and there already exists a relationship between them. Thus, active participation and questions in such a relaxed atmosphere is encouraged, as well as friendly discussions that enrich effective understanding, even when a student in front of the class is being evaluated, too. In general, we could perceive a positive group reaction to questions, comments, and exercises. We believe this contributes positively to both education and learning quality.

A bidirectional support is very motivating for all students: SRP students that teach support others when learning how to work with the new software tool; they answer questions in the classroom and beyond the course hours, they broadcast their own positive experiences and enthusiasm proudly, feeling they are like real experts (they are in what they have developed and researched on for months!). But they also receive ideas and suggestions for further work in their SRP.

Interestingly, classmates are part of the software development willingly because they contribute to the testing process as end users while they solve exercises in the new contents. This kind of *beta* testing, i.e., testing in different environments and computer constellations, as well as usability testing concerning the user interface and Remo usage, contribute enormously to software validation. Moreover, bugs or problems were welcomed in order to improve Remo, some of them being rapidly fixed or modified as appropriate.

Students are in general more motivated in such a lecture for another three reasons. First, some of them usually compare the current work to their own in a different SRP and often times with different supervisors. Second, this internal competition stimulates favorably the emergence of new ideas and discussions between students, too. Finally, understanding new contents by visualizing the way some algorithms work contributes to the learning process in the field and is welcomed when students assist.

A well-planned sandwich structure helps supervisors to find a better balance between frontal lecture and students' needs. While a student is teaching, the supervisor could invest more time in observing other students' behaviors, in completing ideas with parallel comments or alternative explanations when needed, in helping to moderate discussions, in annotating successful or critical lecture development for further analysis, in assessing student's achievements and performance, as well as in simply thinking on new challenges that can improve teaching in the future. Exploiting combinations of co-teaching models and approaches is also a key to success for lecturers when working with undergraduate students.

We suggest planning a differentiated evaluation schema in advance. This is of special importance when the SRP student should also attend the course he or she is working in his/her SRP for. In what extend which contributions and related work belong to the SRP or not, as well as what will be part of the course evaluation, should be well-defined and discussed with the student before both the SRP and the course start. Seeking a balance between course work and research as part of a SRP should appropriately be decided upon by the supervisor. We could consider the role as teaching assistants and the active development in the class as part of the course evaluation for those students working in SRP for the AI course. Thus, we did not include questions related to expert systems in their final exam, for instance, because such students are "almost experts" in that area.

Critic discussing and planning together SRP subject, tasks, goals, and applications in the classroom have several benefits for lecturers but also for students doing SRP. Giving the students enough freedom to propose new features, to suggest didactic methods according to their learning types, to decide on what to teach and how, and even to break traditional teaching schemas, enrich their willingness to both teach and research, as well as their skills, talent, and disposition. Depending on their abilities and needs, for example, re-engineering an existing software program in order to include new functionalities or enhancing existing features to be applied in teaching, could be well part of SRP, too. We believe delegating such responsibilities to the students welcomes decision-making and decentralizes the lecturer's tasks and efforts in an innovative way. In addition, such involvement will better prepare undergraduate students for academia.

We suggest seeking topics and subjects for SRP with enough time in advance to a course start. This contributes to better planning of resources and content to be taught. On the other hand, it leaves necessary time for the conception, design, and implementation of a software tool (if this is the case in the SRP). When giving a subject of SRP, we also recommend having prepared materials and literature about the topic, past documents, or developments when needed, as well as useful contact information to give to the students by the first meeting.

We also recommend, if it is a case of further development for a project, concerting meetings with former developers and teaching assistants in order to exchange ideas in new discussions, as source of inspiration for further work. In such meetings, former students could also inform widely about the state of the art of their work. With this respect, we suggest documenting and registering essential data like date, duration, main topics, and to-do list, for instance, for generating interesting statistics about it. This will be a good estimator to better plan both resources and time to dedicate to SRP and to supervision in general. In particular, Monett invested 2011 almost 6 hours per student in the first SRP, in average. In the first SRP subject of this paper, a total of 8 hours and 40 min. were invested in the whole supervision, including discussions, written report revision, and preparation of contents, to name a few.

Combining research and teaching together in student projects gives undergraduate students the opportunity to be coauthors of future research papers and, even better, to be oral presenters at international events. This was already the case after the 2009/10 edition of our AI course. Such an experience as in [8] was the first of its type at the DCS. Furthermore, it received very positive feedback, also from the AQAS e.V., a German agency for the accreditation of study programs. This paper is the second educator-student attempt in this sense, where we also reflect on the aspects that should characterize a successful combination of undergraduate student research, course work, and teaching.

We should mention some initial fear and resistance to accomplish these tasks. In other words, writing "real" papers or parts of them and presenting research results at international conferences are often rejected, because the benefits of scientific contributions for the future professional life are not well known to undergraduate students yet. However, they turn in a positive stimulus when good supervision, constant support, and co-work are offered to the students.

This is why we suggest not including this higher phase as part of an undergraduate course evaluation, but to give an incentive, from the first discussion and assignment of the research topic on, about the possibilities, conditions, and advantages of a scientific publication on the field. To our opinion, even when a solid research is available and mature for publication, approaching deadlines often gives stress to the students, which could negatively influence their scores in other course topics or even in other parallel courses. Thus, sending contributions to conferences should take place after the course finishes, when possible. For a successful project continuation it is very important to maintain the contact with the students in case they are not part of the department staff. Any kind of further collaboration with them would be as positive as effective if communication, interest, and support exist. The topics and content are of course of utter importance as well.

VII. Conclusions

Bringing together student research projects and teaching gives decisive, explicit input to outstanding teaching in Computer Science. We presented, so far, our experiences on these topics at the Computer Science Division, Department of Cooperative Studies, at the BSEL. We focused on a particular software application, Remo, for modeling rule-based knowledge and for reasoning about it, subject of SRP. We successfully used Remo in an AI course for teaching inference in expert systems. It was challenging for us to involve undergraduate students, authors of SRP and partially training at business enterprises, in teaching.

When a SRP is combined with teaching activities, it is especially attractive for the following reasons: (1) it supports the academic staff, not only in research interests, but also in their lectures, (2) it incorporates and enhances students' soft skills like the ability to teach, (3) it prepares undergraduate students for academia and further steps in this field, (4) it stimulates research essentially focused in current applied topics, and (5) it encourages early interest in publishing research results, amongst many other benefits. We could verify all of them in our concrete settings and hope several recommendations for concrete actions are useful to others.

Further work will be devoted to the application of other co-teaching models and approaches as well as to their evaluation in the classroom. For this, and for the further development of Remo, we already have a new student working in a successive SRP who will enhance and complete both Remo and its documentation, already started online at http://code.google.com/p/hwr-remo/. We also plan to extend our experiences to other curricula modules, as positively perceived and signalized by the students in the survey.

Acknowledgment

We thank Alexander Müller and Markus Beckmann. They programmed in 2010 a first software prototype, predecessor of Remo.

References

[1] -, Bachelor of Science in Computer Science. BSEL's flyer (in German), September 2011. Available online at http://www.hwr-berlin.de/fileadmin/downloads_internet/studiengaenge/flyer_fb2/BRO_FB2_Informatik.pdf.

[2] Ch. Beierle and G. Kern-Isberner, Methods of knowledge-based systems: Foundations, algorithms, applications (Methoden wissensbasierter Systeme: Grundlagen, Algorithmen, Anwendungen), 4th ed., Vieweg+Teubner Verlag, 2008.

[3] D. Berrett, "Want to Be a Good Researcher? Try Teaching," The Chronicle of Higher Education, August 18, 2011. Available online at http://chronicle.com/article/Want-to-Be-a-Good-Researcher-/128753/.

[4] A. Bolour, "Notes on the Eclipse Plug-in Architecture," Eclipse corner article, Bolour Computing, July 2003. Available online at http://www.eclipse.org/articles/Article-Plug-in-architecture/plugin_architecture.html.

[5] M. Friend and L. Cook, Interactions: Collaboration Skills for School Professionals, 6th Edition, Pearson, 2010.

[6] A. Hunter, Graphical Modeling Project. The Eclipse Foundation, 2012. Available online at http://www.eclipse.org/modeling/gmp/.

[7] E. Merks, Eclipse Modeling Framework. The Eclipse Foundation, 2012. Available online at http://www.eclipse.org/modeling/emf/.

[8] D. Monett, R. Janisch, and S. Starroske, "NL-Analyzer: Enhancing Simulation Tools to Assist Multiagent Systems' Teaching," in Proceedings of the Workshop Multi-Agent Systems for Education and Interactive Entertainment (MASEIE), 9th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2010, Toronto, Canada, May 10-14, 2010.

[9] O. Raia, "Co-Teaching," Keys to inclusion, vol. 3, issue 2, NJCIE, East Brunswick, NJ, May 2010. Online at http://njcie.net/pdf/resources/co-teaching.pdf.

[10] M. Sänger, "Inference in Expert Systems: Further development of an existing expert system," 1st student research project, Computer Science Division, DCS, BSEL, 2011, unpublished.

[11] M. Sänger, "Inference in Expert Systems: Further development of an existing expert system – II," 2nd student research project, Computer Science Division, DCS, BSEL, 2012, unpublished.

[12] D. Steinberg, F. Budinsky, M. Paternostro, and E. Merks, EMF: Eclipse Modeling Framework, 2nd Edition, Addison-Wesley Professional, 2008.

[13] J. Wielemaker, SWI-Prolog 6.0: Reference Manual, University of Amsterdam, The Netherland, 2012. Available online at http://www.swi-prolog.org/.

Task-Specific Strategies in Teaching Programming I for First Year Undergraduate Students

Xiaoyuan Suo, Kun Shao
Department of Math and Computer Science, Webster University, Saint Louis, MO, USA

Abstract—*In this paper, we report an experimental study of a task-specific teaching strategy used in teaching C++ programming for the first year undergraduate students in a small liberal arts university. The paper describes and empirically evaluates a combined design and coding model to teach students who have minimal programming and computing skills. We started the teaching process by conducting a background survey from the students who enrolled in programming I classes. The combined design and coding model is constructed for teaching this group based on the survey result. We analyze the survey elements, report our design framework and course structure, and discuss our teaching process and experience. The proposed teaching strategies are proved to be effective in increasing students' ability in logic thinking and performance in programming. The task-specific-strategy teaching model appears to be beneficial and effective, especially for teaching students with limited background in computing and programming.*

Keyword: Task-Specific, Design, Teaching Programming, Teaching Undergraduate Student, Teaching Strategies

1. Introduction

Traditional approaches to teaching computer science programming courses, especially in the lower-level division, tend to focus on developing programming skills through logic theory and cognition-based assignments [1]. This emphasis may not sufficiently prepare students for the upper level CS courses and computer-related employments, especially in the institute where we teach and career-oriented programs consist of the essential contents of its curricular. New jobs in the computing and technological field increasingly demand designing, coding and, testing side by side to achieve programming efficiency. This paper describes and examines the outcomes of a teaching experience in programming I with C++ using a task specific approach which is specially designed for students who have minimal background in computing and programming.

The Department of Mathematics and Computer Science at our university has recently revised its programs to include an interdisciplinary and interactive approach to the study of computer science. All four majors (Information System, Computer Science, Computer Science with Emphasis in Information Technology, and Information Management) mandate Programming I (COSC 1550 C++). In addition, Programming I is cross-listed in the general education requirement for critical thinking, making it a significant and popular course for many other majors. A modified approach to reaching to all students of various academic fields is therefore clearly needed.

Teaching programming design to assist problem solving has been widely recognized as an effective approach in traditional programming classes. Such approach enables students to develop a good understanding in programming and helps them gain programming techniques systematically [2, 3] However, in our department, the contents of programming design and analysis are traditionally offered in the upper level courses such as system analysis and software development. Students in programming I have, so far very limited exposure to the concept and practice of program design.

The model of task-specific design is a simplified approach to teaching programming. Each problem presented in the lecture session is identified as a general task. This general task is divided into several sub problems. Each step of solving a sub problem is defined as a sub task. Each sub task is discussed in detail in the respect of design, implementation, debugging and testing. The problem solving process does not continue until the previous task is completed and tested to work.

2. Related Studies

Many studies have emphasized the importance of design-first model in teaching programming. Deek et al. [4] describes a first year computer science course based on a problem-solving model, in which students focus on formulating problems and devise their own approaches in finding solutions. Language features are taught in the context of solutions.

In our experimental classes we introduce language features along with coding practice to demonstrate and enhance understanding of language syntax and sentence structures.

Linn and Dalbey [6] proposed a concept of “chain of cognitive accomplishments”. The chain starts with the features of the language to be taught. The second link is focused on the design skills, including templates and procedural skills of planning, testing and reformulating code. The third link involves problem-solving skills, knowledge and design strategies.

The task-specific approach we applied in our programming classes follows the concept of the accomplishment chain, but focuses on introducing more design strategies in the early stage of teaching to make learning easier to access.

As Winslow pointed out [7], a good pedagogy requires the instructor to keep initial facts, models and rules simple, and only expand and refine them as the student gains more experience.

In our teaching experience of using task- specific strategies, tasks are divided and solved one at a time, making problem-solving process less complicated and easier to achieve. More sophisticated tasks will be attempted only after students gain sufficient experiences.

Robins et al. [5] also recommended that instructions should focus not only on learning the new language features, but also on combining and use of the features. His study also suggested that programming strategies should receive more and more explicit attention in introductory programming courses. One way of doing so is to introduce many examples of programs as they are being taught.

Similar study conducted by Vihavainen [8] proposed a teaching methodology in which “small goals” are discussed as part of the teaching approach. “Small goals” are defined as the small parts that clearly set intermediate goals.

In our experimental teaching we aimed to use more systematically defined goals or tasks. Specifically, each task must be clearly identified, an algorithm for solving the task must be carefully designed and implemented, and finally a through test must be conducted to guarantee a correct execution. All tasks must be performed in sequence; and each individual task contains steps such as: design, implementation, debugging, and testing.

3 Student Background and the Purpose of the Special Design

3.1 Student Background

We started this study by conducting a background survey from students enrolled in Programming I classes. The purpose of the survey was to better understand students’ background before developing an effective teaching strategy.

Thirty freshman students participated in the survey. 21/30 students reported as computer science related major, while 23% students reported that they do not have any knowledge in logics or programming prior attending the classes. Figure 1 shows students’ background in computer science and related subjects.

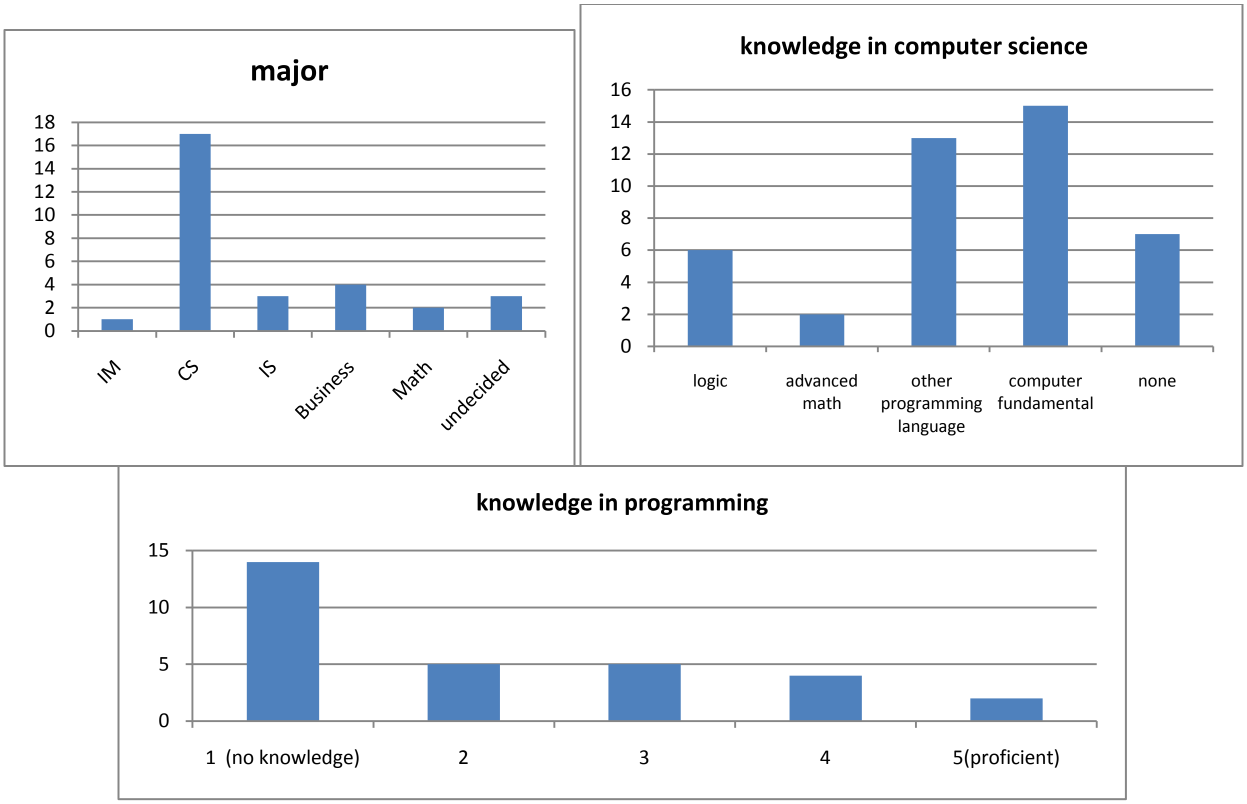


Figure 1.

Top left: Students' major. IM: information management, CS: computer science; IS: information system. Business majors include accounting, international business, marketing etc.

Top right: Students' knowledge in computer science prior to attending the class. Multiple choices are given. In this table, advanced math represents Calculus I or above.

Bottom left: knowledge in programming (including other language) prior attending the class

3.2 Purpose of Design in Task Specific Strategies

From the background survey, students expressed a greater interest in learning programming techniques that directly related to problem solving. We, as instructors, also believed that a foundation course such as Programming I should focus on appropriate strategies in solving a problem along with hands-on lab exercises in practice. The strategic task-specific design was therefore initiated and constructed to enhance our curriculum, and was experimented in our programming I classes.

In the traditional programming classes, lessons are typically started in the way of having students implement the entire program with little or no design. Students tend to debug the whole program after implementation. The disadvantage of this approach is that students will have to deal with quantities of errors at a time. Many students get confused and frustrated. Students commented debugging as a negative experience where "programming gets ugly". In the task-specific design, students were able to segment a problem into smaller and more solvable tasks, dealing one task at a time. In this way, students were able to reduce the number of errors to correct at compilation. The debugging process is therefore simplified considerably.

4. Teaching Methodology

4.1 Course Structure

The Programming I course in our department is a traditional three-credit course. It runs for sixteen weeks. The course is offered three sessions per week; each session is 50-minute long. This course covers the basic C++ language structures, evaluation on mathematical expressions, output/input, control statements, loops, functions and arrays.

Using the task-specific design, we structure weekly classes in the following manor: We focus on introducing new language features in the first session of the class. We enhance students' understanding by demonstrating the logics and usage of each feature in a set of programming segments.

In the second session of the week, we teach how to combine multiple code segments for solving the larger and more complicated problem. In this session a typical task-specific design model is introduced and discussed in detail. This model specifies the relationship between each programming segments developed in the previous session and incorporates each step in the problem solving process.

In the last session of the week, a lab session is constructed to test and enhance students' understanding on the language features and problem solving strategies have been developed in the week. Typically, in this session, we ask students to apply task-specific strategies to work on an extended program.

4.2 The Task-Specific Design

As shown in Figure 2, our model of task-specific design is illustrated in a flow chart with each problem being broken down into mini-tasks. Each mini task should be achievable with no more than 10 lines of code. Tasks are formatted as a flow of checklists, each task should be solved sequentially, that is, no further tasks should be attempted unless the current task is accomplished, tested to be error free and executed correctly.

To enhance what we teach in the class sessions, we developed a unique set of programming assignments directly related to each weekly topic. Each programming assignment contains approximately 100 lines of C++ code; students need to attend in-class discussion and lab sessions in order to complete the assignment successfully.

The framework of the task-specific design is composed of three steps. All tasks in a problem need to be solved in the sequence of design, implementation/debugging, and testing. Language features are identified in the design phase firstly. Algorithmic solution and critical thinking skills to solve the task problem is discussed and specified secondly. Debugging involves multiple testing cases in both syntax and output. Upon each task problem solved correctly, we require students to combine the error-free segments and debug for the entire program before they submit the final solution.

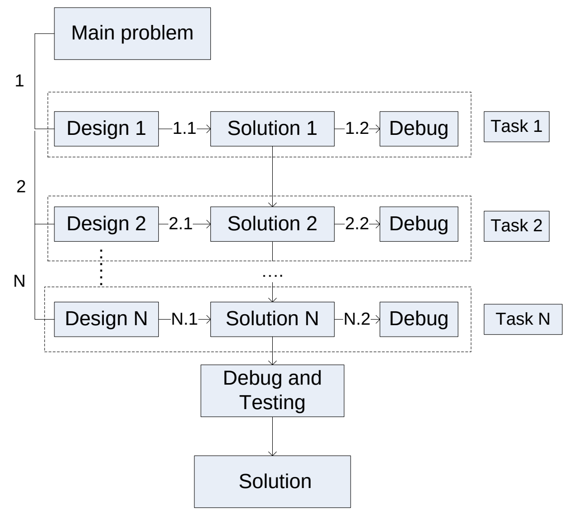


Figure 2: Task-specific strategy flow chart.

4.3 A Sample Lab Assignment

Figure 3 provides a conceptual framework for a sample lab assignment that is practiced using task-specific design when loops are introduced. This assignment asks students to construct a subtraction table on which both row and column numbers are entered by user from the keyboard.

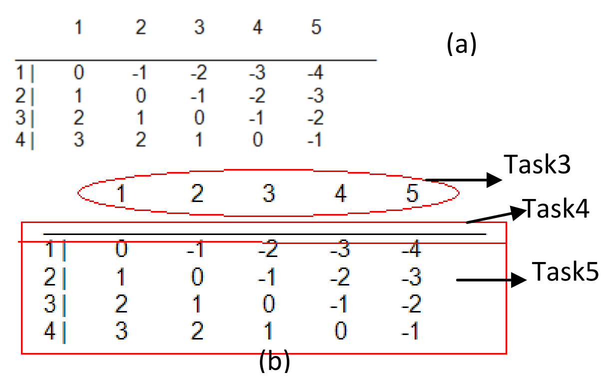


Figure 3. (a) A sample output; (b) three tasks identified for the output design.

Figure 4 is an example of the solution worked out by a group of students and practiced in a class section.

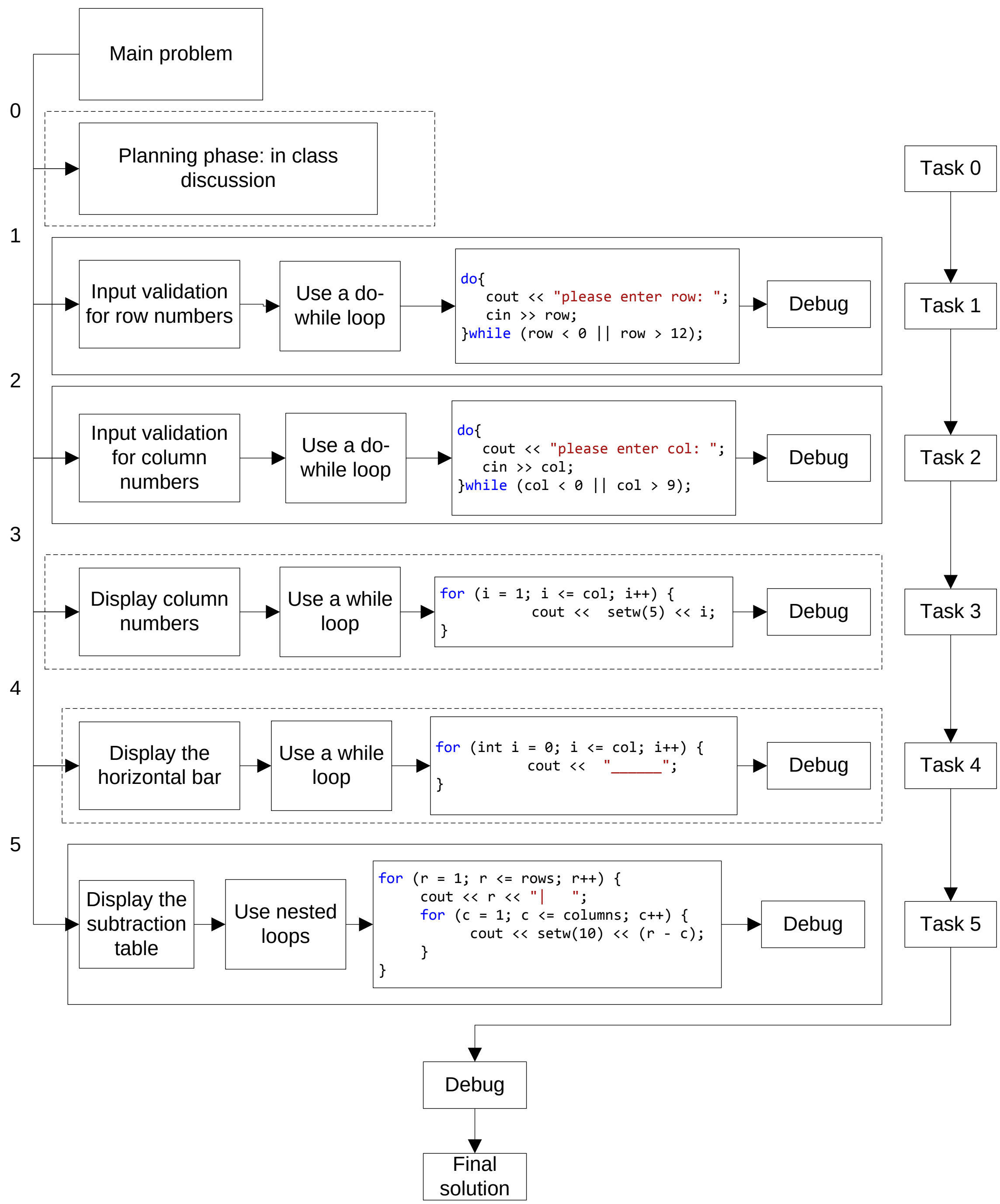


Figure 4. A sample lab assignment. Both the design and the solution are supplied by students.

5. Students Feedback

The outcome of teaching programming I using task-specific strategies is clearly positive. Students reported that they were able to solve a problem within the limited class time (50 minutes). Grades from the assignments indicate that the majority of the students acquired knowledge effectively with approximately 95% of them achieved A- or above. We are very satisfied with this outcome.

To follow up with this experimental study, we conducted an informal in-class discussion and evaluation. The result is described as following:

Typical comments from students at the beginning of the class were: “I have no idea where to start a program”; “I don’t know what to do when I have a problem to solve.” After task-specific strategies were introduced, students reported: "Programming became more interesting and approachable”.

Teaching through task-specific strategies gives students a lot of confidence in problem solving and programming. Programming is made easier to many of student with very limited CS background. Students feel more motivated and interested in trying to solve more complex problems with alternative solutions.

Debugging process became much more manageable. After applying task-specific strategies, students reported that they were no longer experiencing the panic of a long list of error message, and “Errors seemed easier to handle.”

The effectiveness of task-specific design also reflected in students' overall tests scores. With a set of similar questions, the average score was 74/100 in a previous class. The average score reached to 86/100 after the task-specific model was introduced and applied.

Figure 5 summarizes students' feedback on their future studies in computer science. It positively indicates that students are enthusiastically “looking forward” to the next programming course and expecting a career in the IT field.

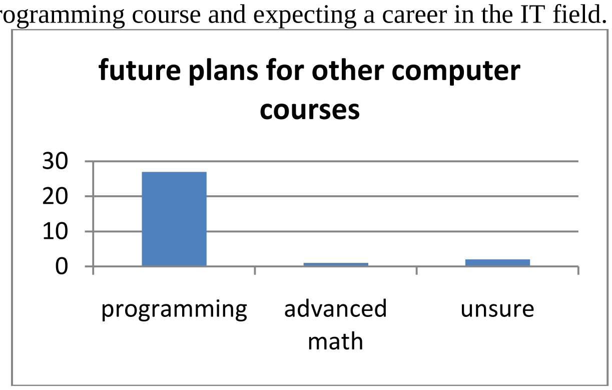


Figure 5: Students’ responses for attending other courses we offer in the same department.

6. Conclusion

The proposed teaching model, task-specific design consists of three main components: design, implementation and debugging/testing. This teaching-approach is proved to be effective in facilitating student learning. It provided special benefits to those students who have limited background in computing. The effectiveness of this approach is reflected in: 1) improved overall students’ performance in assignments and tests; 2) increased confidence of students in programming and 3) raised motivation of students in computer science studies. Currently, the scope of this experimental study is restricted to classes in which the task-specific model is applied and examined. Class time is limited for more comprehensive demonstrations and discussions. Future studies will focus on expanding the number of classes for further evaluation and making attempts to use task specific design in other subjects of our computer science program.

7. References

[1] Hundhausen, C., et al., *Integrating Pedagogical Code Review into a CS1 Course: An Empirical Study*, in *SIGCSE*. 2009: Chattanooga, Tennessee, USA.

[2] Ghafarian, A. *Teaching Design Effectively in the Introductory Programming Courses*. in *Consortium for Computing in Small Colleges*. 2001.

[3] Jose, G. *Object-Oriented Simulation: A Modleing and Programming Perspective*. in *49th ACM Southeast Conference*. 2011. Kennesaw, GA.

[4] Deek, F., H. Kimmel, and J. McHugh, *Pedagogical changes in the delivery of the first-course in computer science: Problem solving, then programming.* Journal of engineering education, 1998. **87**: p. 313.

[5] Robins, A., J. Rountree, and N. Rountree, *Learning and Teaching Programming: A Review and Discussion.* Computer Science Education, 2003. **13**(2): p. 137.

[6] Linn, M.C. and J. Dalbey, *Cognitive consequences of Programming Instruction: Instruction, Access, and Ability.* Educational Psychologist 1985. **20**(4).

[7] Winslow, L.E., *Programming pedagogy—a psychological overview.* SIGCSE bulletin, 1996. **28**(3): p. 17.

[8] 8. Vihavainen, A., M. Paksula, and M. Luukkainen. *Extreme Apprenticeship Method in Teaching Programming for Beginners*. in *SIGCSE'11* 2011. Dallas, Texas, USA

Alternative Teaching Methodologies of Transformations in Computer Graphics

Hassan Farhat
University of Nebraska at Omaha

Abstract- *In computer graphics, objects representation, motion and attributes are modeled using mathematical concepts; among them matrix algebra and vector space. Objects are described in different coordinate systems. The rendering process applies mathematical transformations to the objects as they move down the rendering pipeline.*

In earlier work we discussed the use of vector spaces in the transformations between the different coordinate systems. The application of vector spaces applies as well to the three object transformations in computer graphics. For the most complex, a 3D rotation about arbitrary axis, one of the common methods to obtain the rotation matrix is by applying a sequence of transformations to align the axis of rotation with one of the principle axis.

In this paper we discuss other alternative methods. The paper surveys two additional powerful methods: 1) a method based on using the vector spaces concepts discussed in earlier work, and 2) by applying the quaternion mathematical concept. A quaternion is a generalized representation of a complex number in 3D space. We discuss how the two methods are related by looking at the matrix formed using the vector space method.

1 Introduction

In computer graphics, a scene is composed (of one or several instances) of objects described in their own (modeling) coordinates. Each instance of an object is placed, and is transformed trough animation, to what is called world coordinates by a sequence of primitive transformations [4, 7]. Transformations are then applied to the objects described in world to describe them in alternative coordinate system, the viewing coordinates. The projection coordinates, defined in viewing coordinates, establish a view volume where only those parts of a scene that are within the view volume are eventually displayed on the screen. Additional transformation steps include mapping the view volume to a normalized volume (cube). This transformation is followed with clipping algorithms, applying the proper shading algorithms, and mapping the resulting object description to a viewport for display.

In [1,5,6], given a 3D object in world coordinates, rotation of the object about an arbitrary axis is discussed. One traditional method of obtaining the composite transformation matrix is to: a) translate the axis so as one of the points passes through the origin, b) rotating the axis to align with one of the major axis (two rotations), c) rotate about the proper major axis by the desired angle, c) apply the inverse rotation of step (b), and d) apply the inverse transformation of step (a). Another less common method is to form an orthonormal bases with one of the axis serving as one of the bases elements. The obtained bases can then be used to replace step (b) above. A third alternative approach is to use quaternion algebra where a quaternion rotation matrix is obtained.

In this paper, we survey the two additional methods and discuss how the vector method results and quaternion results are similar. Both methods result in identical rotation matrix. Vector algebra is included in any minimal set of mathematical concepts in the coverage of computer graphics. Sample topics where vector algebra is discussed include ray tracing, lighting models, determining the polygon nature (convex or concave), determining the orientation of a polygon face (front and back), and describing viewing coordinates.

The paper is organized as follows. In section 2 we review the 3D primitive transformation. In section 3, we look at the composite transformations of the 3 primitive transformations and discuss how earlier work [3] can be used in the formation of a general rotation matrix. In section 4 we discuss the derivation of the vector algebra based rotation matrix and its similarity to quaternion rotation matrix. The conclusion is given in section 5.

2 The Three Fundamental Object Transformations

Over the same coordinate system, transforming an object means modifying the object in terms of shape and location (coordinates) with respect to the given coordinate system. In computer graphics, a special type of transformations is used, affine transformations. Affine transformations include the three primitive transformations: translation, rotation, and scaling. Of these transformations, rotation and translation are rigid-body transformations.

Affine transformations make it easier to transform an object by transforming the endpoints of the object description only. The transformations preserve linearity; a line is mapped to a line. They also preserve line intersections and parallelism of lines. Rigid-body affine transformations preserve angles as well.

In addition to the geometric descriptions of an object, the endpoint descriptions may include attribute descriptions, such as the color of the object at endpoints. Linear or bilinear interpolation may then be used to determine the color of the object at other points on the object.

Fundamental to computer graphics is the fact that a sequence of geometric transformations (object movement) is equivalent to performing a sequence of matrix multiplications. In OpenGL the sequence of matrix multiplications is saved in what is called current transformation matrix (CT). The coordinates of objects are normally represented using homogeneous coordinates, an n dimensional point is represented as n + 1 dimensional point. This makes it feasible to apply matrix product when choosing a sequence of the primitive transformations that include translation for example.

Given a 3D point P = (x, y, z), the point is represented in homogeneous coordinates as P = (x, y, z, 1). When we apply geometric transformations such as translation, scaling or rotation, we obtain a new point P_n = (x_n, y_n, z_n, 1). The coordinates of the new point are obtained based on the primitive transformation applied.

The translation transformation, where a point P = (x, y, z) is transformed by t_x, t_y and t_z along the x-axis, y-axis, and z-axis, results in a new point with coordinates x_n = x + t_x, y_n = y + t_y, and z_n = z + t_z. This can be represented using matrix algebra as

$$\begin{pmatrix} x_n \\ y_n \\ z_n \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \\ z + t_z \\ 1 \end{pmatrix}$$

where the 4 by 4 matrix represents the translation matrix T(t_x, t_y, t_z). Scaling transformations are applied when we scale along the three axis by s_x, s_y and s_z. This results in a new transformed point with coordinates x_n = s_x.x, y_n = s_y.y, and z_n = s_z.z. The scaling matrix, S(s_x,s_y, s_z), is the 4 by 4 matrix shown below.

$$\begin{pmatrix} x_n \\ y_n \\ z_n \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} s_x.x \\ s_y.y \\ s_z.z \\ 1 \end{pmatrix}$$

The rotation transformation is applied to a point where the transformed point is obtained from the original point by rotation about an arbitrary axis by a desired angle, either clockwise or counterclockwise. When rotating about one of the principle axes we could get the three rotation matrices by obtaining the matrix of rotation about one of the primary axis. The matrices for rotations about the other two axes are formed by permutation of the computed rotation matrix. For example, the rotation matrix about the z-axis in a right-handed coordinate system, by a positive angle θ, results in a counterclockwise rotation (as seen by a viewer looking at the origin from a positive z value). The rotation matrix obtained, $R_z(\theta)$, is the 4 by 4 matrix shown below.

$$\begin{pmatrix} x_n \\ y_n \\ z_n \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta).x - \sin(\theta)y \\ \sin(\theta)x + \cos(\theta)y \\ z \\ 1 \end{pmatrix}$$

By performing a cyclic permutation where (xyz) is replaced by (zxy) we obtain a rotation about the y-axis. Hence, for the above matrix, x is replaced by z, y by x, and z by y to yield the transformation equations below.

$$\begin{pmatrix} z_n \\ x_n \\ y_n \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} z \\ x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta).z - \sin(\theta)x \\ \sin(\theta)z + \cos(\theta)x \\ y \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x_n \\ y_n \\ z_n \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \sin(\theta).z + \cos(\theta)x \\ y \\ \cos(\theta).z - \sin(\theta)x \\ 1 \end{pmatrix}$$

Repeating above results with another cyclic permutation results in

$$\begin{pmatrix} z_n \\ x_n \\ y_n \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} z \\ x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} \sin(\theta).y + \cos(\theta)z \\ x \\ \cos(\theta).y - \sin(\theta)z \\ 1 \end{pmatrix} .$$

The rotation matrices about one of the principle axis are normally written as $R_i(\theta)$ where i is one of the principle axis.

3 The Composite Transformations

When forming a sequence of transformations of a point or an object, the sequence of transformations can be combined into a single matrix, a composite matrix, that accomplishes the desired final result. As an example application, the scaling and rotation matrices given in section 2 leave the origin as the invariant point. To scale with (x_f, y_f, z_f, 1) as the invariant point, we form

$$\begin{pmatrix} x_n \\ y_n \\ z_n \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & x_f \\ 0 & 1 & 0 & y_f \\ 0 & 0 & 1 & z_f \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -x_f \\ 0 & 1 & 0 & -y_f \\ 0 & 0 & 1 & -z_f \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} s_x & 0 & 0 & x_f(1-s_x) \\ 0 & s_y & 0 & y_f(1-s_y) \\ 0 & 0 & s_z & z_f(1-s_z) \\ 0 & 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}.$$

Similarly a rotation about an arbitrary axis of rotation, from the rotation matrices above, requires forming a composite transformation matrix. The composite rotation matrix to a rotate a point is determined from: a) the coordinates of the point, b) the axis coordinates (given as two points), and c) the angle of rotation.

Assume that the angle of rotation is θ, the axis of rotation is given by two points (x_1, y_1, z_1) and (x_2,y_2, z_2), and the point to rotate is P = (x, y, z). The traditional method of obtaining the rotation matrix is obtained by the following sequence of transformation is to:

a) translate the axis of rotation (and with it the point) so that the initial axis point is at the origin,
b) rotate the axis of rotation so as to place on one of the primary axis,
c) rotate the point about the axis of rotation by desired angle,
d) apply the inverse of step (b), and
e) apply inverse of step (a).

Step (b) above can be accomplished by two rotations. The first places the axis of rotation in one of the xy, xz or yz planes. The second rotation aligns the axis with one of the primary axis.

An alternative, but less common method of the above sequence of steps, is to accomplish (b) by making use of forming an orthonormal bases of the xyz space with the axis of rotation as one of the bases vectors.

In [3] we discussed how we could compute the coordinates of a vector or a point in two different coordinate systems (world and viewing for example) using vector spaces. Each coordinate system forms an orthonormal bases. Assume the axis of rotation is the vector V_z[1] and the unit vector of V, v_z, forms the z axis of the orthonormal bases, $v_z = (v_{zx}, v_{zy}, v_{zz})$. Similarly assume the other two orthonormal vectors corresponding to x and y axis are, respectively, $v_x = (v_{xx}, v_{xy}, v_{xz})$, and $v_y = (v_{yx}, v_{yy}, v_{yz})$. The rotation matrix for step (b) is given by the orthonormal rotation matrix

$$\begin{pmatrix} v_{xx} & v_{xy} & v_{xz} & 0 \\ v_{yx} & v_{yy} & v_{yz} & 0 \\ v_{zx} & v_{zy} & v_{zz} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

In [3], we employed a similar matrix to give the coordinate of the same point in space in two different coordinate systems. The generalized equation computed is

$$\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ 1 \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} & 0 \\ x_{21} & x_{22} & \cdots & x_{2n} & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nn} & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix} . \begin{pmatrix} 1 & 0 & \cdots & 0 & -P_{01} \\ 0 & 1 & \cdots & 0 & -P_{02} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -P_{0n} \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix} . \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} & -\sum_{i=1}^{n} x_{1i}.P_{0i} \\ x_{21} & x_{22} & \cdots & x_{2n} & \sum_{i=1}^{n} x_{21i}.P_{0i} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nn} & \sum_{i=1}^{n} x_{ni}.P_{0i} \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix} . \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \\ 1 \end{pmatrix}$$

where the (c, c_2, …, c_n) are the coordinates of a point in the viewing system, the row vectors (x_{i1}, x_{i2}, …, x_{in}) form the viewing bases, and (x_1, x_2, …, x_n) are the coordinates of the point in world. The n by n upper-right matrix above has alternative interpretation. It can be thought of as a rotation matrix that transforms the bases vectors (viewing bases) so as to align them with the natural bases (world bases). A point described in world, when multiplied by the above matrix results in new coordinates in world as well. However, since the effect of the matrix is to align the bases of viewing with the natural bases, the new coordinates of the rotated point is that of the viewing coordinates. Orthonormal matrices have the property that the inverse of the matrix is the transpose of the matrix. Hence, from step (b) it is easy to obtain step (d).

In the third alternative method of obtaining the composite rotation matrix, the quaternion method is used. In this method it is assumed the axis of rotation passes through the origin. A quaternion is a generalized notation of a complex number. A 2D complex number has the two operations: multiplication and addition. The complex number is an order 2-tupple. Two of its alternative representations are as Cartesian coordinates point or as a polar coordinates point (length and angle). When polar representation is used, the effect of the multiplication is to obtain a rotation for the angle part. A quaternion is a generalization of complex numbers that extends to the 3D case. An ordered pair for case complex number is (x, y) extended to quaternion is represented as (s, v) where s is a

[1] We also represent V_z as $\overrightarrow{V_z}$

scalar and v is a vector. In the next section we derive the quaternion matrix but using vector algebra. The section is based on [8].

4 The Alternative Rotation Matrix Computation

The computation is based on: the given coordinates of a point to be rotated, P = (x, y, z, 1); the coordinates of the axis of rotation, assumed as a vector, V, passing through the origin, V = (x_v, y_v, z_v, 0); and an angle of rotation θ. The objective is to derive the rotation matrix, R, where $P_n = (x_n, y_n, z_n, 1)^T = R \cdot P^T$. The rotation matrix, R, will cause P to rotate counterclockwise about V by an angle θ.

Figure 1 below show the vector V, and the points P and P_n.

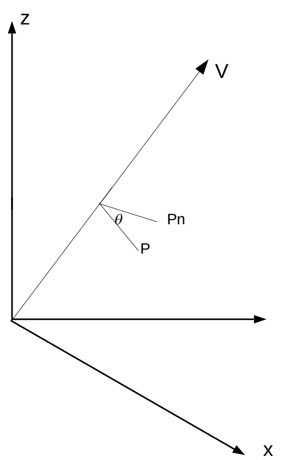


Figure 1

To derive the transformation matrix, R, we use vectors to arrive at the desired rotation matrix. In above we form the plane perpendicular to the vector V and containing the point P, Figure 2.

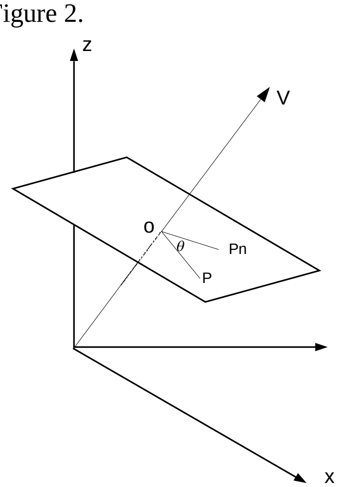


Figure 2

For simplicity we work with the unit vector, v, in the direction of V. We then form vector equalities based the vectors OP, the position vector O, the unit vector v, and the left-hand perpendicular vector to $\overrightarrow{OP}$, $\overrightarrow{OP}^{perp}$. We note that the position vector, P_n is the vector sum $\overrightarrow{P_n} = \overrightarrow{O} + \overrightarrow{OP_n}$. We also note that the vector $\overrightarrow{OP_n}$ can be written as a linear combination of the vector $\overrightarrow{OP}$ and the perpendicular vector to $\overrightarrow{OP}$, $\overrightarrow{OP}^{perp}$, Figure 3.

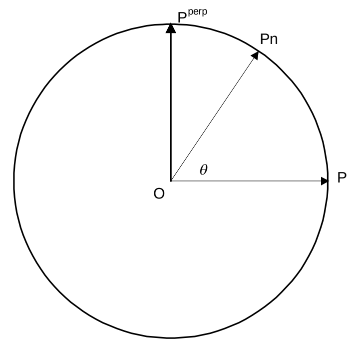


Figure 3

From the figure we have $\overrightarrow{OP_n} = \cos(\theta)\overrightarrow{OP} + \sin(\theta)\overrightarrow{OP}^{perp}$.
The position vector P_n is then obtained as

$$\overrightarrow{P_n} = \overrightarrow{O} + \cos(\theta)\overrightarrow{OP} + \sin(\theta)\overrightarrow{OP}^{perp}$$

Using cross product of the unit vector v and the vector OP we obtain the coordinates of the perpendicular vector OP^{perp}. Hence the above equation can be rewritten as

$$\overrightarrow{P_n} = \overrightarrow{O} + \cos(\theta)\overrightarrow{OP} + \sin(\theta)(v \times \overrightarrow{OP}),$$

where $\times$ is the cross product operator.

In order to find the desired P_n, it is remaining to find the coordinates of the vector O. Since O = kv for some scalar value k we have,

$$\begin{aligned}\overrightarrow{P_n} &= \overrightarrow{O} + \cos(\theta)\overrightarrow{OP} + \sin(\theta)(v \times \overrightarrow{OP}) \\ &= \overrightarrow{O} + \cos(\theta)(\overrightarrow{P} - \overrightarrow{kv}) + \sin(\theta)(v \times (\overrightarrow{P} - \overrightarrow{kv})) \quad (1) \\ &= \overrightarrow{O} + \cos(\theta)(\overrightarrow{P} - \overrightarrow{kv}) + \sin(\theta)(v \times \overrightarrow{P})\end{aligned}$$

k is found by noting that k is the projection of P on the axis of rotation,

$$k = \left\|\overrightarrow{P}\right\|\cos(\alpha)$$

where alpha is the subtended angle between P and V. With

$$\vec{v}.\overrightarrow{P} = \left\|\vec{v}\right\|\left\|\overrightarrow{P}\right\|\cos(\alpha) = \left\|\overrightarrow{P}\right\|\cos(\alpha) = k,$$

we have $\overrightarrow{O} = k\,\vec{v} = (\vec{v}.\overrightarrow{P})\,\vec{v}$. Substituting in equation (1) results in

$$\overrightarrow{P_n} = (\vec{v}.\vec{P})\vec{v} + \cos(\theta)(\vec{P} - (\vec{v}.\vec{P})\vec{v}) + \sin(\theta)(v \times \vec{P})$$
$$= \left((\vec{v}.\vec{P})(1 - \cos(\theta) \right)\vec{v} + \cos(\theta)\vec{P} + \sin(\theta)(v \times \vec{P}) \quad (2)$$

When the above vector equation is applied on each of the three-dimensional coordinates, on $P_n = (P_{nx}, P_{nx}, Pn_x)$, $P = (P_x, P_x, P_x)$, and $v = (v_x, v_x, v_x)$, where P_n is the coordinates of the point P after the rotation, we obtain

$$\begin{aligned}
P_{nx} &= (1 - \cos(\theta)(v_x p_x + v_y p_y + v_z p_z)v_x + \cos(\theta)p_x + \sin(\theta)(v_y p_z - v_z p_y) \\
&= \left((1 - \cos(\theta))v_x^2 + \cos(\theta)\right)P_x + \left((1 - \cos(\theta))v_x v_y - \sin(\theta)v_z\right)P_y + \left((1 - \cos(\theta))v_x v_z + \sin(\theta)v_y\right)P_z \\
P_{ny} &= (1 - \cos(\theta)(v_x p_x + v_y p_y + v_z p_z)v_y + \cos(\theta)p_y + \sin(\theta)(v_z p_x - v_x p_z) \\
&= \left((1 - \cos(\theta))v_x v_y - \sin(\theta)v_z\right)P_x + \left((1 - \cos(\theta))v_y^2 + \cos(\theta)\right)P_y + \left((1 - \cos(\theta))v_z v_y - \sin(\theta)v_x\right)P_z \\
P_{nz} &= (1 - \cos(\theta)(v_x p_x + v_y p_y + v_z p_z)v_y + \cos(\theta)p_z + \sin(\theta)(v_x p_y - v_y p_x) \\
&= \left((1 - \cos(\theta))v_x v_z - \sin(\theta)v_y\right)P_x + \left((1 - \cos(\theta))v_y v_z + \sin(\theta)v_x\right)P_y + \left((1 - \cos(\theta))v_z^2 + \cos(\theta)\right)P_z
\end{aligned}$$

When the rotation matrix is extracted from the above, with $\cos(\theta)$ replaced by c and $\sin(\theta)$ replaced by s, we obtain the equation

$$\begin{pmatrix} P_{nx} \\ P_{ny} \\ P_{nz} \\ 1 \end{pmatrix} = \begin{pmatrix} ((1-c)v_x^2 + c) & ((1-c)v_x v_y - sv_z) & ((1-c)v_x v_z + sv_y) & 0 \\ ((1-c)v_x v_y - sv_z) & ((1-c)v_y^2 + c) & ((1-c)v_y v_z - sv_x) & 0 \\ ((1-c)v_x v_z - sv_y) & ((1-c)v_y v_z + sv_x) & ((1-c)v_z^2 + c) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} P_x \\ P_y \\ P_z \\ 1 \end{pmatrix}$$

The above 4 by 4 matrix is same matrix obtained when using the quaternion method. In fact in [8] it is shown that the vector method of rotation yields equation 2. When the method is compared to the quaternion method, a similar equation is obtained for the vector component of the quaternion.

5 Conclusion

The field of computer graphics makes extensive use of transformations. The 3D rotation about arbitrary axis is normally formed using one of three methods. 1) Successive applications of transformation to align the axis of rotation with one of the primary axis where the rotation is performed followed by the inverse transformations. 2) A less common method that is based on the use of orthonormal bases in combination with method 1. And, 3) using the quaternion method.

In this paper we surveyed these methods and applied earlier work of vector spaces in the discussion. In the discussion we emphasized the method of rotation based on the use of vectors. Of the three methods, introductory computer graphics texts emphasize the composite transformation method and the quaternion method. Among the possible advantages of the vector algebra method: a) it provides the students with a simpler procedure than the quaternion method, and b) the students gain more solid understanding of vector algebra.

References

[1] E. Angel. “Interactive Computer Graphics, A Top-Down Approach Using OpenGL, 5th Edition”. Addison Wesley, 2009. E. Angel. “OpenGL, A Primer, 2nd Edition”. Addison Wesley, 2005.

[2] H. Farhat. “Transformations and Basis in the Computer Graphics Viewing Pipeline,” International Conference on Frontiers in Education, FECS 2009, pp. 267-271.

[3] Foley, van Dam, Feiner, Hughes. “Computer Graphics Principles and Practice, 2nd edition”. Addison Wesley, 1990.

[4] D. Hearn, M. Baker, W. Carithers. “Computer Graphics with OpenGL, 4th Edition”. Prentice Hall, 2011.

[5] F. Hill, S. Kelly. “Computer Graphics using OpenGL, 3rd Edition”. Prentice Hall, 2007.

[6] D. Rogers, J. Adams. “Mathematical Elements for Computer Graphics, 2nd edition”. McGraw Hill, 1990.

[7] A. Watt, M. Watt. “Advanced Animation and Rendering Techniques Theory and Practice”. Addison Wesley, 1992

Team-Teaching Scratch Programming and Voice Acting Art in Elementary School Classrooms

Ying-Fang Lin[1], Janet Mei-Chuen Lin[2], and Jin-Cheng Kang[3]
[1]Taipei Municipal Shilin Elementary School, Taipei, Taiwan
[2]Graduate Institute of Information and Computer Education, National Taiwan Normal University
[3]New Taipei Municipal Er-Chong Elementary School, New Taipei City, Taiwan
[1]evonlin67@gmail.com, [2]mjlin@ntnu.edu.tw, [3]nypd3@ms3.hinet.net

Abstract - *This study investigated the feasibility of interdisciplinary team teaching that combined Scratch programming taught in the computer class with voice acting taught in the arts and humanities class. One computer teacher and two art teachers conducted team-teaching in two sixth-grade classes for 8 weeks. A cycle of planning, acting, observing and reviewing was performed iteratively, which provided feedback for the instructors to modify their teaching strategies as they progressed through seven lesson units, each of which requiring students to develop a Scratch programming project involving the use of voice acting elements. Data were collected through classroom observation, student questionnaire survey, journals kept by the team teachers, and one-to-one interviews. It was found that interdisciplinary team teaching combining Scratch programming and voice acting art helped to enrich content learned by students, encourage innovations and experiments, increase peer interaction, enhance learning interest and bolster confidence in learning. It also promoted self-reflection and exchange of ideas among team teachers, thereby improving their teaching skills.*

Keywords: Scratch programming, Team teaching, Voice acting art

1 Background

As technology has become ever-present in our daily lives, it also has made its way into almost all aspects of education, including art education. Some universities have implemented courses that allow college students to explore the intersection between art and computer science. For example, Computer Science (CS) professors at University of Massachusetts Lowell teamed with Music, Theater, and Art professors to offer both introductory and advanced Performamatics courses. One of the courses, called Sound Thinking, aimed at providing Music majors the opportunities to learn something about technology and CS majors to learn something about music ([1]). To ensure there was true cross-over between the two disciplines, the researchers established a set of learning objectives for all students. For example, students were expected to be able to identify properties of sound and describe the organization of sound into music, distinguish between analog and digital audio, discuss the basic differences between various audio file formats and sound compression techniques, and create a web-based computer program that plays a music file. Scratch was selected as the development platform. It was found that students who took the course could gain musical and computational understanding through experimentation and exploration that was both educational and fun.

In another course called Artbotics ([2]), also offered by University of Massachusetts Lowell, students were given Super Crickets, servo and DC motors, and a variety of sensor types, including touch, infrared ranging, and light, as well as art materials to create interactive, tangible exhibits. The Cricket was programmed using Cricket Logo. The art-related objectives of the course were for students to learn principles of aesthetic and conceptual elements in visual art; whereas the computer-related objectives expected students to be able to formulate structured algorithms and program them. The researchers concluded that the combination of art and computer science was an engaging medium for students.

Northwestern University has also offered an Animate Arts program that sought to train students to cross the boundaries of art and technology through team teaching and a curriculum that drew from visual and sound arts, cultural and narrative theory, and computer science ([3]). The program succeeded in teaching programming to students who might otherwise find it inaccessible or unappealing.

As for integration of technology in art education in K-12 schools, what has been seen most often is the use of computer graphics software by art teachers to teach art concepts to youngsters (e.g., [4-6]); besides, some music teachers use music notation software, ear training program, or software to record student performances ([7]).

As an attempt to explore other possible ways of integrating technology in art education, particularly in teaching performing arts, we conducted an action research in sixth-grade classrooms to investigate if we could teach Scratch programming in combination with voice acting art through team teaching. Specifically, students would be required to apply the concepts they learned in the art classroom about voice acting to the Scratch projects they would be assigned to work on. Our objective was to examine how the team teaching could be implemented effectively and how this new endeavor would benefit teaching and learning.

2 Literature Review

2.1 Team Teaching

As opposed to the single-subject, single-course, single-teacher pattern of teaching, team teaching involves two or more teachers working cooperatively to help students to learn. Teams can be single-discipline, interdisciplinary, or school-within-school teams ([8, 9]). Team members set learning objectives together, determine the sequence of topics, design teaching materials, prepare individual lesson plan, teach students, and assess students' learning.

McKinley [10] examined student and faculty perceptions of two interdisciplinary team-taught courses. According to their findings, students felt that interdisciplinary team-teaching led to a broader and deeper understanding of the material; faculty also stated that their students exhibited a greater interest in the course material than in traditional classes; moreover, the learning environment provided by the course required students to think and write at levels not often present at traditional classes. Ediger [11] pointed out a few problems that teachers might encounter in team teaching situations. Other than the problem that team members needed to find time to plan the course together before, during, and/or after the class day, it was also difficult to provide for individual needs and supervising individual projects and activities in large group team-teaching instruction. Ediger also cautioned that human relations problem might surface in team teaching and it was important to allow for individual differences among instructors.

Researchers have had a continuing interest in exploring the boundaries between science and the arts. For example, Overton & Chatzichristodoulou ([12]) explored teaching of science concepts through the performing arts. Education students collaborated with students from the School of Arts and New Media to produce performances designed to teach difficult science concepts to elementary school children. It was found that when engaging with science concepts through performing arts, it involved assimilation of the concept, subsequent cognitive conflict and possible accommodation of the concept through having to interpret and present it through the performing arts, which was a combination of creative thinking in science and arts disciplines. Krug & Cohen-Evron ([13]) indicated that ideas and processes could be made concrete for students through an artist's presentation and enlarged the scope and deepened the breadth of an idea or theme. They reported a practice in which art, math, science, English, and social studies teachers at a Midwest high school in the US selected "Heroes" as a theme to study for a group of 70 ninth-grade students. The arts were used as a way to enhance and supplement curricular content from each field of specialization. Students and teachers worked together to exchange and enlarge the general knowledge and concepts of each other's subject areas.

2.2 Scratch Programming

Scratch (http://scratch.mit.edu/) was developed at the MIT Media Laboratory. It was intended primarily for 8- to 16-years old. Guided by three core design principles—more tinkerable, more meaningful, and more social, its developers aimed at providing a programming tool with which children and teens can learn mathematical and computational ideas, learn to think creatively and reason systematically, and learn to work collaboratively through creating computer games, interactive stories, graphic artwork and animation ([14-16]). Scratch is now available in nearly 50 languages, including the traditional Chinese version used in this study.

The Scratch programming environment (see Figure 1) is divided into four areas. On the upper right is the stage area where sprites can move. Below the stage is an area that shows all sprites in the project. The middle pane is the scripting area where a programmer creates a script to control a sprite's movement as well as edit its images and sounds. The left-most pane is the palette of command blocks. The palette is divided into eight color-coded categories: motion, control, looks, sensing, sound, numbers, pen, and variables. A Scratch script is created by dragging command blocks from a palette into the scripting pane and assembling them into stacks of blocks.

Maloney, Peppler, Kafai, Resnick, & Rusk ([17]) reported on the Scratch programming experiences of urban youth at an after school center over an 18-month period. Their analyses of 536 Scratch projects revealed that the youngsters were able to learn key programming concepts, including user interaction, loops, conditionals, communication and synchronization, even in the absence of formal instruction. However, Boolean operations, variables, and random numbers were not easily learned on their own. Furthermore, the urban youth showed a sustained engagement with Scratch programming.

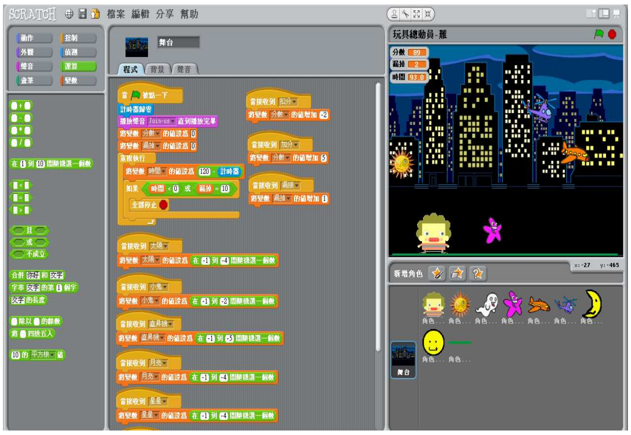


Figure 1. The Scratch programming environment

3 Method

3.1 Participants

Seventy-one sixth-graders from an elementary school in New Taipei City, Taiwan participated in this research. Thirty-seven students were from one class (CA) and 34 from the other (CB). Prior to this study the participants had learned in previous computer courses various application packages, including MS Word, MS PowerPoint, PhotoImpact, Swishmax, NBlog, and common Web applications. In addition, they had learned Scratch programming for 11 weeks before team teaching started.

Other participants of this study included two art teachers CT and RT (RT is the first author of this paper), a computer teacher KT (the third author of this paper), and two observers R1 and R2.

3.2 Lesson Planning

Table 1 outlines the seven lesson units planned by the team teachers (i.e., the two art teachers and the computer teacher) together. It shows the titles of each unit and its computer-related and art-related learning objectives respectively.

Table 1. The Lesson Units and Learning Objectives

Unit	Computer-related Objectives	Art-related Objectives
Unit 1: One-Man Chorus	● Record a sound file ● Differentiate between “play sound” and “play sound … until done” commands ● Use “play sound” and “wait *x* secs” ● Register a new accoun at the Scratch website	● Recognize the importance of voice acting in performing arts ● Understand the basic concepts of voice acting ● Explore one’s voice
Unit 2: The Sound of Lines	● Search for targeted projects on Scratch website and leave comments about them ● Produce desired sound effects with Scratch commands ● Use “play sound” and “play sound until done” in project	● Describe attributes of sounds suitable for representing line segments of different colors and shapes ● Produce sounds for various line segments ● Appreciate how sounds can be used to create artwork
Unit 3: A Day in the Zoo (see Figure 2)	● Predict the commands to be used to produce the actions required ● Move sprites around the screen with motion commands ● Edit the costumes and sounds owned by a sprite ● Use “repeat” command to repeat a sequence of commands certain times	● Imitate sounds of different animals ● Describe how an animal may make particular sounds under certain psychological or physiological conditions ● Produce sounds of an animal in different scenarios ● Draw different appearances of an animal
Unit 4: Acting Out Various Emotions (see Figure 3)	● Create a sequence of commands ● Use “set volume” to control the sound volume ● Understand the concept of using messages and broadcasts to control the application	● Record voices of three different emotions ● Identify properties of voices that express different emotions ● Critique each other’s work
Unit 5: Self-dialogue	● Use “play sound … until done” command in project ● Know how to change a sprite’s appearances with the “switch to costume” command	● Know how to write dialogue ● Express various emotions with voices ● Create suitable stage backgrounds for different emotions
Unit 6: The Dog-Cat Dialogue	● Understand the function of “broadcast” and “when I receive” commands ● Use messages and broadcasts to write the dialogue	● Act out the function of sending and receiving messages ● Understand how Scratch executes sequences of commands
Unit 7: Dialogue between Two People (see Figure 4)	● Use “broadcast” and “when I receive” commands in project ● Produce the dialogue with Scratch commands ● Critique each other’s work	● Understand the relationship between voice and aging ● Record a dialogue between two people of different ages

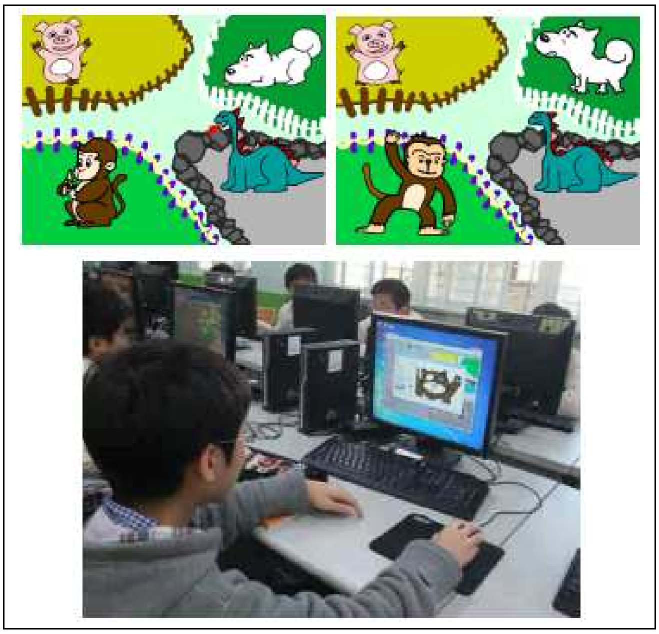


Figure 2: Students creating sprites for *A Day in the Zoo*

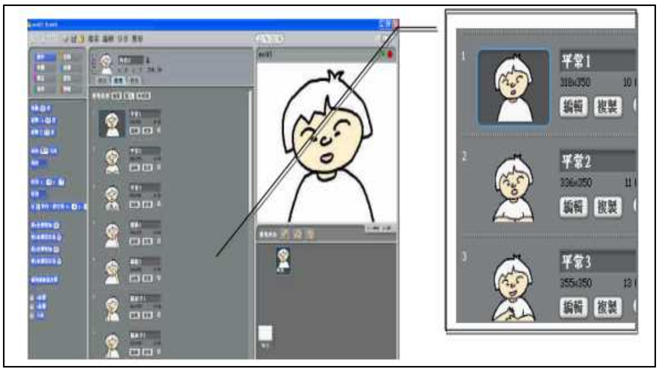


Figure 3: The project for *Acting out Various Emotions*

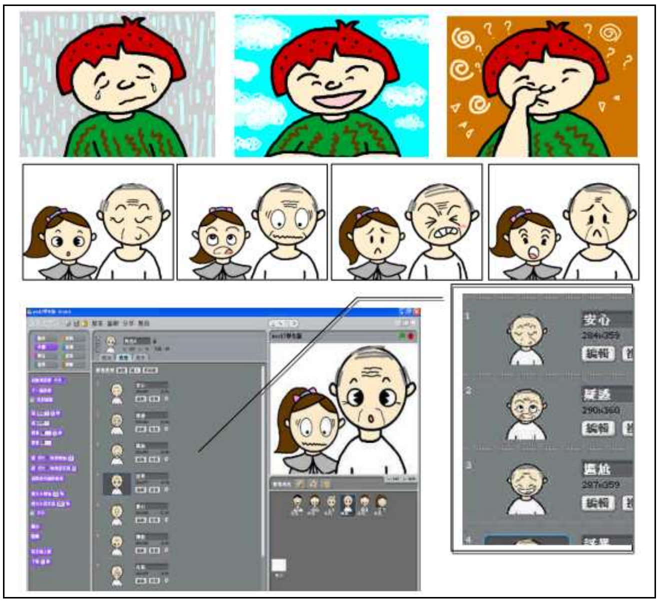


Figure 4: The project for *Dialogue between Two People*

3.3 Procedure

This research was conducted from October to December 2011 for 8 weeks. There were 2 periods, each of 40 minutes, for the art class and 1 period for the computer class per week. The computer class was taught by KT and the art class was taught by RT. When KT was teaching Scratch programming in his computer class, RT sat in his class as an assistant and evaluator; when RT was teaching voice acting in her art class, the other art teacher CT served as the assistant and evaluator. The evaluators used a pre-designed evaluation sheet to evaluate classroom teaching from three perspectives: content design, teaching strategies, and student performance. The two observers (R1 and R2) sat in both classes and used the observation sheets to record important events occurring in the classroom, including students' behavior, teacher-student interaction and student-student interaction. They also were responsible for suggesting possible adjustments to teaching practice based on their classroom observation.

The implementation of each lesson unit followed the same pattern. At the beginning of each unit, the three teachers (KT, CT, and RT) discussed and reviewed the unit plan together, then KT taught Scratch programming concepts and skills covered in that unit to class CB. After the class, KT adjusted his teaching practice based on the feedback provided by the evaluator RT and the two observers (R1 and R2); he then taught the same course material to class CA. The art part of the same unit was then taught by RT to class CB in her Art class, in which she explained the voice acting concepts and skills to students and guided them in implementing performance ideas in their Scratch projects. Similarly, RT would discuss with the evaluator CT and the two observers after class to see if any adjustments needed to be made to her teaching; she then went on to teach the same material to class CA. At the completion of each unit, the team teachers met and examined the entire process to decide if any major modification to the team-teaching implementation was needed. This pattern continued through all the seven lesson units.

A questionnaire survey was administered in the ninth week to collect students' feedback on their learning experience during team-teaching, as well as how they felt about Scratch. The survey was followed by one-to-one interviews with the participating teachers and students.

4 Results

This section presents our major findings. We describe students' cognitive, affective, and behavioral outcomes, how the team teachers benefited from their team-teaching experience and the difficulties they had encountered.

4.1 Cognitive Outcomes

According to the data collected from student questionnaire survey, 90% of students agreed that they were better able to comprehend the concepts taught in the classes because the two teachers, who had different areas of expertise, approached the same topic from different angles and inspired students from different perspectives, which helped to improve the quality of student projects. As one student mentioned in an interview:

S1: The computer class used to be boring because the teacher only taught us how to write codes. Now we have another teacher to teach us how to draw sprites and make interesting sound recordings. It was much more fun, and I was more satisfied with my own projects.

Four-fifths of the students also agreed that the combination of two content areas provided a richer learning environment that promoted creative thinking and idea generation. Through voice recording they were able to explore the characteristics and possibilities of their own voices and grew to love their own voices.

Moreover, Students also showed greater confidence in creating Scratch scripts. For example,

S10: I didn't like to help others, but now I do, because I feel more confident in my own programming ability.

The following excerpts were taken from CT's classroom observation notes:

CT: Teacher RT illustrated how to associate different sounds with line segments that had different colors, height, and width. The students were excited over the magic power of sound.

CT: The students were highly interested in creating new sprites and changing their costumes.

CT: The students were happily engaged in the activity. They imitated sounds of different animals with great imagination. The classroom was filled with laughter.

CT: Teacher RT explained the concepts with real-life scenarios that were closely related to students' experiences. For example, students were told to imagine how they had felt when they were sick, scolded by their parents, didn't do well in an exam, or lost their money. The examples enabled students to find ready themes to write script dialogues for their projects and record voices showing different feelings.

The computer teacher KT observed that interdisciplinary team teaching enriched learning experience because of the novelty provided by the new learning materials. Furthermore, team-teaching allowed more time for teachers to give an in-depth coverage of a topic and for students to work on each Scratch project. As more than 60% of students agreed, "We had more time to practice, and we learn more about each topic."

As for students' opinion on the usability of Scratch, the majority of students felt that the sound recorder and paint editor provided by Scratch were easy to use and allowed free exploration of ideas. The students also appreciated that the Scratch website made it easy for them to upload and share their Scratch applications. They felt proud and excited when they saw their Scratch applications displayed on the website. Students also found it meaningful to view and comment on applications uploaded by others. Students' comments had been rather vague at first, such as "Your costume is cool!" or "What a cute frog!", but most students were able to give more meaningful comments after they were instructed on how to write better comments.

With respect to the suitability of using Scratch to teach voice acting, the art teacher CT stated:

CT: Scratch is a very useful software package. Students can write a script, record their own voices and save it. The recorded sound files enable students to evaluate their own performance after class, which is not possible with the traditional teaching method. Although teachers can record individual students' voices with audio or video recorders, it is very time-consuming and impractical.

One thing that both the teachers and the students complained about Scratch was that it provided few ready-made sprites to choose from. Furthermore, the teachers were also annoyed by occasional classroom disruptions due to strange errors with Scratch that caused the applications to hang.

4.2 Affective Outcomes

Student questionnaire survey revealed that more than three-fifths of students liked the team-teaching arrangement and hoped that they could have similar learning experience for other topics of performing arts.

The survey also showed that students' favorite unit was Unit 7 (Dialogue between Two People). The reasons given were that it gave them more space for creativity and they liked the idea of playing two roles, even though they also agreed that the Scratch script they had to create for that project was much harder. The unit that students disliked most was Unit 1 (One-Man Chorus). The unit was found to be dull and the concepts too abstract to grasp.

4.3 Behavioral Outcomes

It was found that classroom dynamics in both classes changed gradually during the 8-week period when this research was conducted. In the beginning most students simply worked on their own projects, rarely did they pay attention to their peers or volunteer to help. Some students even showed reluctance when asked by the teacher to help other students. However, as the process went on, the classroom atmosphere became livelier and the interactions between students increased substantially. Those students who were able to finish the assigned work faster would look around and ask if anyone needed help. They also were enthusiastic in exchanging ideas and giving suggestions on how to improve each other's project. As RT noted in her evaluation sheets:

RT: S2 would ask, "Anyone need any help?" Then there would be 5 or 6 students raising their hands and yelling, "Me! Me!" It's a great improvement.

RT: S3 walked to S4 to help her. Later S4 herself also walked around the computer lab, checking if anyone needed her help.

Interdisciplinary team teaching also had the effects of changing teachers' stereotypical views of certain students because it enabled teachers to discover other talents of a student which was less likely in single-course situations. The computer teacher KT found that some students who had not been performing well in his class became very engaged and were able to turn in outstanding projects. As he noted:

KT: S5's past performance was unimpressive in my class, but he really surprised me this time. He was quick in grasping the concepts and his projects had many creative ideas.

It was also found that assignment completion rates of both classes greatly improved. Those students who used to be less disciplined and/or less motivated had become more engaged as team teaching proceeded. As one student said:

S6: I didn't like Scratch, but I like it now. It's fun to use Scratch to make sound recordings.

Teacher RT also noted:

RT: S7 never had behavior problems, but he had never turned in any assignment before. He came to me in class one day with his worksheet in hand and asked for my help. I spent the recess time helping him to write the script. I felt he was willing to learn and do his assignments as long as he can get some help. After a few projects, he eagerly volunteered to help me out with chores around the classroom, such as passing out and collecting the headsets.

The team teachers also noticed that changes happened in some passive students. They became more concerned about getting their projects done and were ready to seek help when they encountered difficulties. Two students' changes were most evident, as shown in the following observations made by RT:

RT: S8 kept coming to me, checking if I had received the file he sent through email. He seemed very eager to show me his work and get my approval. He also showed great curiosity about what others were doing with their projects.

As for student S9, he used to have some behavior issues and always relied on the help of others to do the projects, but RT observed the following change in S9:

RT: S9 tried to do the project on his own. He worked attentively with the code stacks and listened carefully to make sure that the dialogue produced by his program sounded like a reasonable conversation between two people.

CT also noticed that some students who used to be shy and subdued became more extraverted as they gained more and more confidence in their own programming skills.

4.4 Teacher Professional Growth

The three team teachers agreed that team teaching helped to improve the quality of teaching. As CT described:

CT: Brainstorming with other professionals is important. We tend to be isolated in a classroom with too little stimulation from peers. Team teaching enabled us to learn from each other. Regular discussion with team members and watching them teach helped to engaged ourselves in self-reflection on our shortcomings.

CT: Teachers with different expertise may approach the same topic from different angles, which will help us to learn new perspectives and insights, techniques and values from exchanging ideas with one another.

Both CT and KT suggested that more subject areas be involved in team teaching:

CT: We could have involved language art teachers. If we had, the dialogues for the projects could have been written in the language art class so students had had more time to think and polish their writing skills.

KT: It would be wonderful if a software package taught in my computer class could be used by math, language arts, or teachers of other subjects in their classes. It is a great way of connecting student learning and technology.

4.5 Difficulties Encountered

Team teaching makes more demands on time and energy, which presents a great challenge for future implementation of team teaching. As CT and KT pointed out:

CT: It took extra time and effort to rethink the courses to accommodate team-teaching and design new teaching materials; besides, we needed to sit in and observe what students were doing in each other's classes, which meant two extra class periods for me and four extra periods for KT. It was exhausting. I don't think there would be many teachers enthusiastic about doing this because most of us already have heavy workload.

KT: Ideally I should have attended my partner's art classes every time so I knew what she was teaching about voice acting, but since I also serve as technology coordinator in my school, the administrative duty did not allow me to do so. Discussion with RT and CT helped somewhat, but first-hand observation by myself would have helped me to teach my part better.

5 Conclusion

This study investigated the feasibility of interdisciplinary team teaching that combined Scratch programming taught in the computer class with voice acting taught in the arts and humanities class. It was found that the implementation was not only feasible but enhanced the quality of learning, including enriching content learned by students, encouraging innovations and experiments, increasing peer interaction, enhancing learning interest and bolstering confidence in learning. It also promoted self-reflection and exchange of ideas among team teachers, thereby improving their teaching skills. However, as team teaching tends to make more demands on teachers' time and energy, teachers may need more support and incentives before they would be willing to try this non-traditional teaching method.

6 References

[1] A. Ruthmann, , J.M. Heines, G.R. Greher, P. Laidler, C. Saulters II, "Teaching Computational Thinking through Musical Live Coding in Scratch", Proceedings of the 41st ACM technical Symposium on Computer Science Education SIGCSE. ACM Press, March 2010

[2] H.A. Yanco, H.J. Kim, F. Martin, and L. Silka, "Artbotics: Combining Art and Robotics to Broaden Participation in Computing", Proc. of the AAAI Spring Symposium on Robots Robot Venues: Resources for AI Education. Stanford Univ., CA. 2006.

[3] I. Horswill and M. Novak, "Evolving the Artist-Technologist", IEEE Computer Society, 2006, pp. 62-69.

[4] eHow.com, How to Integrate Technology in Art Education. Retrieved February 29, 2012 from http://www.ehow.com/way_5545261_lesson-plans-graphic-design.html.

[5] S. Meadors, Computer Graphics Lesson Plans. Retrieved February 29, 2012 from http://syrylynrainbowdragon.tripod.com/comlp.html.

[6] Lesson Planet, Computer Graphics Lesson Plans. Retrieved February 29, 2012 from http://www.lessonplanet.com/lesson-plans/computer-graphics.

[7] Kelly's Music & Computers: Using Technology in Music Education, Retrieved February 29, 2012 from http://kellysmusic.ca/education.asp.

[8] I.J. Singer, "What Team Teaching Really Is", Team Teaching: Bold New Venture. Indiana University Press, Bloomington, IN, 1965, pp. 13-28.

[9] Buckley, F. J., Team teaching: What, Why, and How? Sage Publication, Inc, Thousand Oaks, 2007.

[10] B. McKinley, "An Examination of Team-Taught Interdisciplinary Courses", 1996 (ERIC Document Reproduction Service No. ED397877).

[11] M. Ediger, "Assessing Five Sacred Cows in Education", 2000. (ERIC Document Reproduction Service No. ED442862)

[12] D.T. Overton and M. Chatzichristodoulou, "The Teaching of Science through the Performing Arts", Procedia Social and Behavioral Sciences, 2(2), 2010, pp. 3871-3875.

[13] D. Krug, & N, Cohen-Evron, "Curriculum integration, positions and practices in art education", Studies in Art Education, 4(3), 2000, pp. 258-275.

[14] J. Maloney, K. Peppler, Y.B. Kafai, M. Resnick, and N. Rusk, "Programming by Choice: Urban Youth Learning Programming with Scratch", ACM SIGCSE Bulletin, 40(1), ACM Press, 2008, pp. 367-371.

[15] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J.S. Silver, B. Silverman, and Y.B. Kafai, "Scratch: Programming for Everyone", Communications of the ACM, 52(11), ACM Press, 2009, pp. 60-67.

[16] J. Maloney, M. Resnick, N. Rusk, B. Silverman, and E. Eastmond, "The Scratch Programming Language and Environment", ACM Transactions on Computing Education, 10(4), ACM Press, 2010, pp. 16:1-16:15.

[17] M. Resnick. "All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten", Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition, 2007, pp.1-6.

Teaching with the Emerging GENI Network

James Griffioen, Zongming Fei, Hussamuddin Nasir, Xiongqi Wu, Jeremy Reed, and Charles Carpenter
Laboratory for Advanced Networking, University of Kentucky
301 Rose Street, Lexington, KY 40506, USA

Abstract— *Over the last few years the National Science Foundation (NSF) has been investing in and developing a new network called* GENI, *a wide-area testbed network for at-scale experimentation with future internet designs. The GENI network has recently become available for use and is beginning to attract users.*

In this paper, we take a closer look at GENI with a particular focus on how GENI can be used to enhance education in the areas of computer science and computer engineering. We describe what GENI is, the resources available in GENI, and how instructors might use GENI in their classes. Being early adopters, we describe our experience using GENI in our classes, and we point out various features and challenges of using GENI. Finally, we provide tips and pointers to instructors who are interested in incorporating GENI into their own classes.

Keywords: hands-on experiment, GENI, network testbed, network monitoring, network lab assignment

1. Introduction

Although the Internet has been extremely successful and has transformed essentially every aspect of our culture, there is wide-spread agreement that the existing Internet architecture suffers from various problems and limitations that are constraining the development of new and innovative services. To address this issue the National Science Foundation (NSF) has launched several efforts designed to investigate new network architectures capable of supporting the types of applications and services desired in a "future internet" (e.g., the Future Internet Design (FIND) program [1] and the Future Internet Architecture (FIA) program [2]). In addition, NSF noted that the only way to truly evaluate new network architectures and designs is to construct an at-scale (i.e., Internet-scale) testbed network on which researchers can experiment with their ideas. To that end, NSF created and charged the GENI Project Office (GPO) with the task of implementing a large-scale testbed network that could be used for the development of next generation internet architectures.

After several years of development, the *Global Environment for Network Innovations (GENI) [3]* has recently become available for use and is rapidly attracting researchers working on novel network architectures and applications. Although GENI is also available for educational use, the vast majority of the GENI effort to-date has been focused on, and involved, the network *research* community.

The goal of this paper is to present ways in which GENI can be used to enhance *education*, particularly in the areas of computer science and computer engineering. Section 2 begins by giving a brief overview of GENI and the types of resources available in GENI. Section 3 then provides an example of one of the possible ways in which a user might access and control/use the GENI infrastructure. Having provided an introduction to GENI, we then offer some potential educational uses for GENI in Section 4. Section 5 then describes projects that we have used in our networking classes at the University of Kentucky. We then briefly compare GENI to alternative approaches in Section 6 and offer some concluding remarks and tips to instructors in Section 7.

2. GENI Overview

The goal of GENI is to enable users to build and operate their own Internet-scale networks. To that end, GENI has *resources* (e.g., computers, routers, network links) all across the nation (and to some extent internationally via federation) that are available for users to reserve and use in their own (private) network. For example, a user working on new data center services might build a testbed network by reserving desktop style computers (physically located) in Utah, Wisconsin, Kentucky, and Georgia for use as "clients" in their network, server machines in Salt Lake City to mimic a "data center", and routers in Atlanta, Washington DC, Kansas City, and Salt Lake city, with links that interconnect the "clients" and "data center" into a complete network. In the past, building such a network would be impossible for the normal user, and would even be challenging, if not impossible, for ISPs that own and operate network routers and data centers. However, with GENI, it becomes relatively easy to create such a network; in fact, the above network could be created for the user in a matter of minutes.

Alternatively, consider a user that wanted to test out new services run out of a data center supporting mobile (wireless) users. In this case, the user might build a test network by reserving a set of mobile (wireless) nodes in New Jersey and in Wisconsin connected via routers in Washington DC and Atlanta to a set of "data center" machines in Georgia and Kentucky.

In short, GENI offers a wide range of resources all across the nation that users can quickly and easily compose into a network of their own design. Moreover, GENI can support multiple user networks simultaneously – each being independent of the others. In other words, each user gets their own network composed of a set of resources that have been reserved for that network. In GENI terminology, a user's network is called a *slice* of GENI (a subset of the GENI resources allocated for the user's network). Each individual resource in the network is called a *sliver* of the slice. The list of resources that comprise a slice are often represented by a *Resource Specification (RSPEC)*.

The set of resources that make up GENI are owned and operated by different entities. The set of resources that are operated by a particular entity is called an *Aggregate*. When a user reserves resources, GENI contacts the various aggregate operators where those resources are located to verify that the resource is available and that it can be reserved. Consequently, each aggregate operator has the ability to allow or deny requests for its resources.

Many resources are programmable, implying that the user is free to design, implement, deploy, and control the software that runs on the resource. In particular, users are able to develop their own network protocol stacks and are not required to use the standard TCP/IP protocol stack. Ideally users could build their own network from the ground up, starting with their own physical and data link layers and all the layers up to the application layer. However, for pragmatic reasons the current version of GENI only allows users to select from a set of conventional technologies at the physical and data link layers (e.g., Ethernet, 802.11, etc). However, beyond that users are free to build/use any network layers they desire. For example, if a user wanted to write and test their own network layer protocol (instead of using IP), GENI would support it. If they wanted to add new services or processing into routers in the middle of the network (things that might be viewed as layering violations in the current Internet architecture), GENI would support it. In short, GENI allows users to throw out the current Internet (TCP/IP) model and start all over should they want to.

2.1 GENI Resources

Unlike other wide-area testbed networks such as Planetlab [4] and Emulab [5] which offer one (or very few) types of resources (e.g., a raw PC or Vserver virtual machine), GENI was designed to offer a wide range of different types of resources ranging from PCs, to (truly) mobile nodes, to network processors, to high-speed packet capture devices, to high-end compute servers.

GENI resources can be allocated to a single user for exclusive use. For example, a raw PC can be included in a slice, and the creator of the slice can request that a specific version and distribution of Linux operating system (e.g., Ubuntu 10) be installed and loaded on the PC. The slice creator has complete control over the machine. On the other hand, GENI resources can be virtualized and shared by multiple users. The virtualization technology makes it possible for each user to get a piece of the resource, such as an OpenVZ container or a virtual router, without interferences from other users. The user can pretty much do whatever she/he want within her/his slice.

Three categories of resources are provided by GENI. The first category of GENI resources are end hosts, which include raw PCs, virtual PCs such as OpenVZ and Vserver, and virtual platforms such as python sandbox environments provided by the million node GENI [6]. As the GENI project installs GENI server racks on more campuses, server class computers with fast processors and large storage space will become available as high-performance end hosts to be included in the user slice.

The second category of GENI resources are routers. When setting up a network experiment, PCs and virtual PCs can be configured to act as routers. GENI also provides physical routers (such as Juniper routers installed at Atlanta, Washington DC, Kansas City and Salt Lake City connected to the Internet2 backbone). The user can request Juniper logical routers (virtualized routers) from the GENI ShadowNet aggregate [7]. In addition, wireless access points, open flow switches, network processors (NetFPGAs) and mobile nodes can be allocated to set up a customized network testbed for users.

The third category of GENI resources are links, both wired (such as copper and optical links) and wireless. GENI provides layer 2 connections between different aggregates. The user can request that a layer 2 physical link be reserved between routers or end hosts for a given slice. To study the behavior of residential users, one can even include cables to the home as a part of slice, in order to collect the usage data [8]. Virtual links can also be established among nodes in an experiment. They can be VLANs set up among nodes within an aggregate or GRE tunnels/TCP-UDP tunnels set up between nodes from different aggregates.

2.2 GENI Measurement Infrastructure

In addition to resources, GENI provides a rich set of instrumentation and measurement tools for monitoring user experiments. They enable users to collect measurement data easily and make sure that experiments behave as expected. For example, GENI instrumentation and monitoring tools (INSTOOLS) has been developing *slice-specific monitoring* infrastructure to capture, record, and display information about a user slice, based on the consideration that experimenters are primarily interested in the behavior and performance of their experiment [9]. After the user instructs the system to instrument a slice, INSTOOLS will automatically deploy, configure, and run monitoring software or services on the resources that comprise the slice. It will also set up servers to collect the information captured at these resources, process the information, and make it available to the user via a graphical user interface.

Three categories of data are collected. The first one is network tables such as ARP tables, IP address tables, and routing tables. The second one is traffic information, such as IP, ICMP, TCP, or UDP traffic over time. The last one is operating system information, such as CPU load, memory load, and loaded modules. All these data are presented via a portal, which provides a "one-stop shop" to access all the data collected from the experiment. The traffic and load information are presented as graphs showing how it changes over time, while other information is presented as tables. They are accessible from any web browser through the Internet. The user can quickly identify abnormal behavior by observing these graphs and tables.

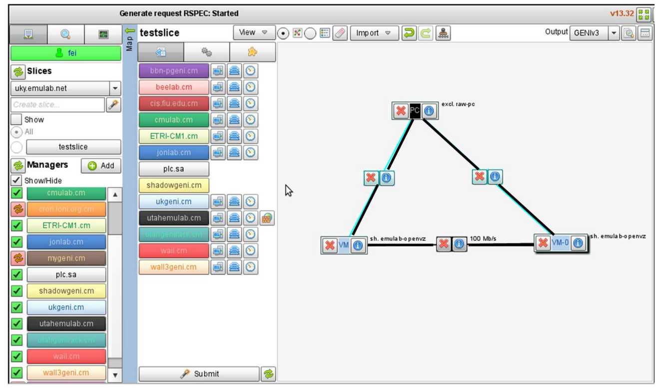


Fig. 1: Flack interface.

3. Using GENI

The first step to use GENI is to get an account from a GENI aggregate. After answering several questions to provide the identity of the applicant, the user's account will be approved by a GENI administrator. As GENI is a federated system, a user having an account with any aggregate can request resources from other aggregates to be included in her/his slice, subject to the policies of the aggregates involved.

GENI provides users with a graphical user interface called Flack, which is a Flash application accessible from a web browser [9]. It was developed as a part of the ProtoGENI project [10]. After logging into Flack, the user can find resources available across the GENI aggregates. To create a new experiment, the user only needs to drag the icons of the corresponding resources (raw or virtual nodes) from the desired aggregate onto the canvas, as shown in Figure 1. The user can specify which operating system image will be used in these nodes. Links can be set up between these nodes. Typically, a VLAN will be created for nodes from the same aggregate, while a GRE tunnel (IP-in-IP encapsulation) [11] will be created for nodes from different aggregates. The user can also drag a delay node from an aggregate to be used as a part of the link. The delay node will be used to emulate different characteristics, such as latency, bandwidth and loss rate, of the link.

After the "submit" button is clicked by the user, the GENI system will automatically allocate the resources from the corresponding aggregates and set up the experiment. This may take several minutes to finish because the machines will be booted and configured according to the specification provided by the user.

Once an experiment has been created, the slice can be instrumentized as shown in Figure 2. The first step is to click on the plugins tab. The "instrumentize" button will show up. The second step is to click the "Instrumentize" button. The instrumentation tool will distribute measurement software to experimental nodes and configure them to capture traffic and load information on these nodes. One measurement controller will be added for the slice at each aggregate. It will be used to collect and process data captured at experimental nodes, and act as a server to present the measurement data to the user. When the instrumentation process is finished, clicking on the "Go to portal" button (step 3 in the figure) opens a browser window to the INSTOOLS portal for the experiment.

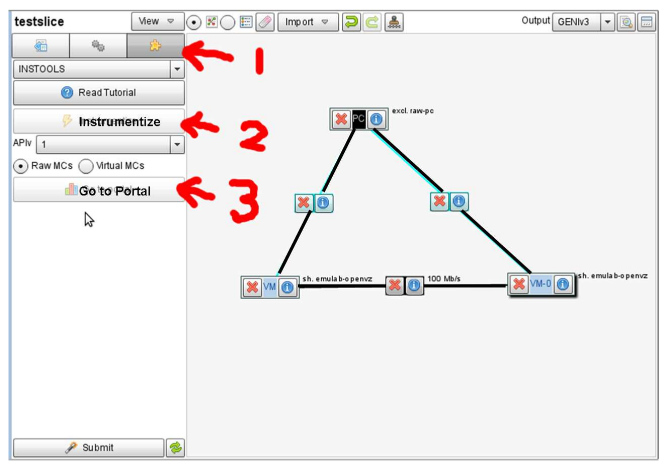


Fig. 2: Instrumentize an experiment

The portal presents the user with the collected information. Figure 3 shows the physical topology of the experiment to the user. It is a map view overlaid with the resources and network topology used in the slice. The user can also choose to look at the logical topology, which ignores the location information and can be read more easily. The user can click on any node or any link and a dialog box will appear, from which the user can choose what performance information to observe. Whatever is picked by the user will show up on the left-hand side of the interface.

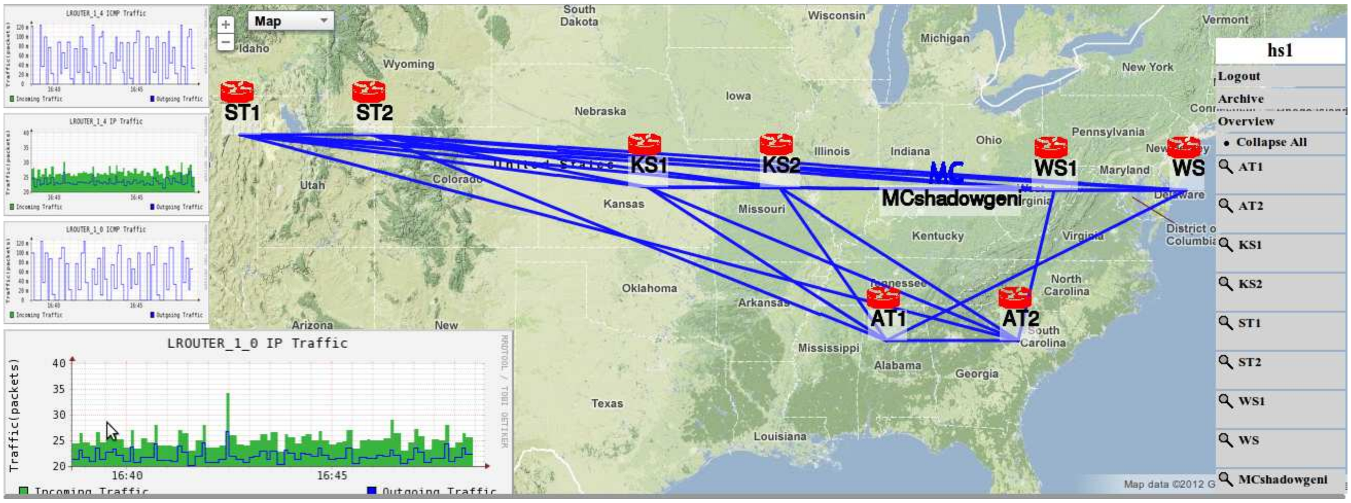


Fig. 3: The monitoring portal

4. Educational Uses for GENI

Networking classes have long used a variety of different "lab" environments to give students hands-on experience. Examples range from a single PC executing a network of virtual machines (e.g., VMware, Virtual Box, OpenVZ, etc.), to custom-built networking lab facilities [12], to emulation facilities like Emulab [5], to wide-area overlays like Planet-lab [4]. This raises the obvious question "What can GENI do that these others cannot?" or stated another way, "Why should I use GENI?"

The short answer to that question is that GENI is in many ways a superset of these other approaches. Because GENI is based on Emulab, Planetlab, etc, most anything that can be done in one of these other environments can also be done in GENI; *but* GENI also offers features not available in these other environments. In particular, GENI is designed to support "at-scale" networks that can scale up in terms of the number of resources offered, the types of resources offered, and the speed/performance of resources offered. As a result, a variety of new types of experiments and students projects are enabled by GENI–projects that are difficult, if not impossible to implement using past approaches. The following briefly outlines various types of projects that are enabled by the new GENI infrastructure.

4.1 Types of Projects Enabled by GENI

The following is not intended to be a comprehensive list of the types of projects possible on GENI, but rather is presented to give an idea of the potential uses for GENI in computer science and computer engineering courses. The first two types of projects mentioned below are possible using past systems and have been widely used in operating system and networking classes. GENI not only supports these, but enables a variety of new types of network experiments:

1) *Conventional OS/Networking Assignments*: These projects ask the student to make modifications to existing OS and networking code to create their own (routing) protocols or network services. Systems like Emulab, virtual machine networks, and custom lab facilities have long been used to support these types of assignments. Many GENI resources also allow students to have complete control over the software that runs on GENI resources.
2) *Network Monitoring Assignments:* These projects ask the student to write active (intrusive) and passive monitoring code to measure the performance of the Internet. GENI, like PlanetLab, offers geographically distributed resources that have been particularly useful as a basis for these types of assignments.
3) *Data Center/Cloud Assignments:* With the emergence of cloud computing, GENI, unlike past systems, offers high-performance clusters that can be used to implement "data center" services using custom or conventional (e.g., hadoop) data center software.
4) *Mobile Networking Assignments:* Because GENI includes a variety of (virtualized) mobile resources, it is an excellent platform on which to do assignments involving mobile users and code.
5) *Wireless Assignments:* GENI supports a variety of wireless network technologies including both mobile and fixed infrastructure that allows students to do assignments that must deal with the realities of wireless networks (e.g., variable loss rates).
6) *Home Networking Assignments:* While there are limited home network resources (e.g., cable modem/DSL) available, the ability to write code that utilizes the resources of opt-in home users is a unique feature of GENI.

7) *High-performance Networking Assignments:* Most student assignments are designed to build something that works, with performance as an afterthought. However, GENI offers high-performance servers, programmable network processors, and optical networks that enable assignments that test scalability of performance.
8) *Application-level Monitoring Assignments:* GENI includes some unique application-level resources such as low-power radar sensors and web cameras that are virtualized and accessible for use by users. Moreover, the high-performance network links available in GENI make it possible to move data off of these devices to network servers in real-time.
9) *Complete Network Assignments:* Because GENI supports so many different resources, it is possible for students to work on projects that involve every aspect of a complex/complete network ranging from (mobile) client nodes connected via wireless links to an optical backbone networks with advanced services built-into the network structure, as well as data center computing power offering cloud services.

5. Experience of Teaching with GENI

We have been using GENI to teach network courses at the University of Kentucky, including hands-on projects that use GENI. Students have enjoyed many features provided by GENI. The controlled environment makes it worry-free for unexpected changes to operating systems and network connections. The widely distributed GENI aggregates enable experiments to be deployed over machines distributed across the Internet. The easy setup and teardown let students make mistakes and modify and improve their experiments within minutes. In this section we describe in some detail three networking experiments that are helped by the availability of GENI. We have given the first two projects in our networking classes. The third project is one that we are considering giving in the future.

5.1 Reliability Protocols

Reliability is one of the most important concepts in computer networks. Traditional mechanisms for implementing reliability include Stop and Wait, Go Back N, and Selective Repeat. TCP uses a slightly different reliability protocol by using cumulative acknowledgment and buffering out-of-order packets at the receiver.

A programming assignment can let students implement one of the reliability mechanisms. The sender and receiver can be implemented as a UDP client and a UDP server. A timer must be implemented in order to deal with the cases when data or acknowledgment packets are lost. The goal is to observe the behavior of the programs in a wide variety of loss scenarios.

The problem with testing these programs in a general purpose computer lab is that we typically do not observe much loss in normal networking conditions. Therefore, all data packets to the receiver and all acknowledgment packets back to the sender get through without loss. Consequently the timeout and retransmission mechanisms will not be tested. Ideally, we would like to have a controlled environment in which we can get whatever loss rate we would like for any links. The scenarios should be repeatable. This is hard, if not impossible, to achieve in a general purpose lab.

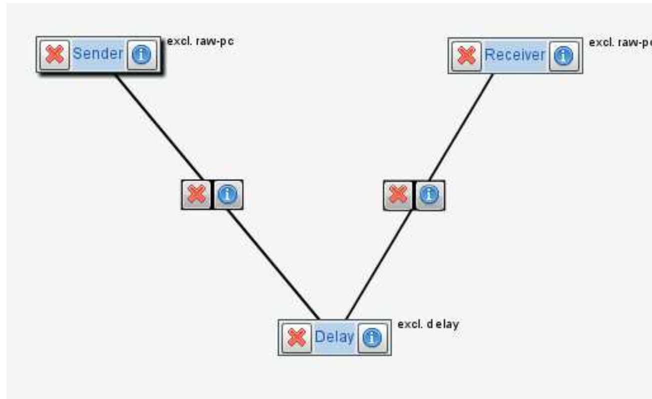


Fig. 4: Sender and receiver connected with a delay node as the link between them.

In contrast, GENI allows students to set up a network topology with a sender node, a receiver node and a link between them as in Figure 4. The link can be implemented by a delay node. GENI allows the delay node to control the link characteristics, especially the loss rate in this case. All these can be easily set up using the GUI provided by GENI and the experiment can be up running in a few minutes. Students can run their programs on the sender node and the receiver node, which typically are loaded with standard Linux OS. The loss rate can be specified to whatever values we want. We can do stress test of the protocol implemented to see whether it can handle all kinds of extreme scenarios.

5.2 Network Configuration and Automatic Route Control

In this project, students first create a network topology and then manually configure the network routing using conventional network administration software. In the second phase, students implement, deploy, and test a new/emerging type of router that separates routing from forwarding. In particular, they will write a Forward Information Base (FIB) controller process that allows remote control of the forwarding operation of a router. This creates the potential to run Routing Decision (RD) services remotely - for example, on a centralized controller node. By sending commands to a FIB Controller, an RD server is able to modify the Forwarding Information Base in a router.

The first step of this project is to create a network topology using the ProtoGENI Flack interface. For example, students can create a 4-node topology as shown in Figure 5, where the two nodes on the top are shared nodes from the utahemulab component manager and the two nodes at the bottom are from the ukgeni component manager.

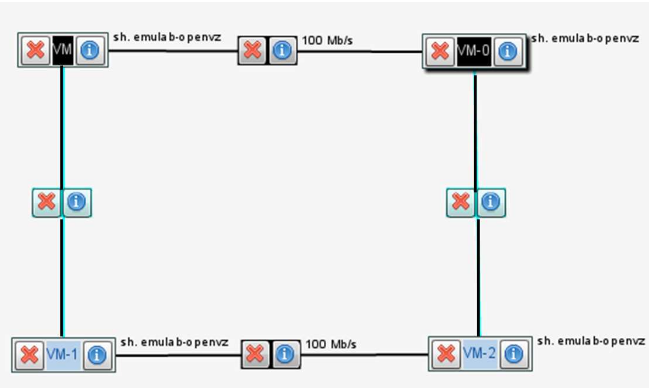


Fig. 5: A sample network topology.

Students login to these machines and use Unix commands such as *ifconfig*, *netstat*, *arp* and *route*, to observe and set up routing so that one can ping from any node to any other node. This manual configuration process helps students gain experience with the basics of IP, ARP and routing.

The second part involves writing programs to act as a FIB controller at each node and the RD server to control the routing through the FIB controllers.

The FIB controller is a process that runs on every router. It accepts commands from RD servers and then inserts or deletes the appropriate routes in the FIB (routing tables). To insert or delete entries in the FIB, the FIB Controller uses the Unix "system()" system call or the UNIX "exec()" system call to run the Unix *route* program. The FIB Controller and the RD Server speak a FIB Controller Protocol (FCP) that students design and implement. Students are free to design any FCP protocol. The only requirement is that the FCP be run over the UDP protocol (i.e., carried in the payload of UDP packets). The FIB Controller should receive UDP packets (from the RD servers) on a predefined UDP port number.

The RD server runs on a separate node - a node that is not acting like a router. The RD needs to be able to send IP packets to every one of the FIB Controllers it is controlling. This implies that there exists a direct link from the RD node to each router, or there exists an IP path from the RD node to every router. Students need to modify the above topology to meet these requirements.

As mentioned above, the RD server will speak the FCP protocol with the FIB controllers that it manages. The RD server will not make any routing decisions itself. Instead, the RD server will accept commands from a computer terminal (command-line interface), translate the commands into the FCP protocol and send them to the appropriate FIB Controller to be install/deleted. The command-line interface can include commands such as creating a path to a destination address, deleting a path, showing the routes to an IP address.

5.3 Path Characteristics of WANs

Most of our network classes use the Internet as a basis to teach principles of network design. To get a better understanding of the Internet, it is essential to understand the characteristics (such as delay, bandwidth, loss rate) of the links/paths in the Internet. We can ask the following common questions about the Internet. What is the typical delay of a path from the east coast to the west coast? How do delay, bandwidth, and loss rate differ between a local link and a wide area path, or among different wide area paths? Do they change a lot over time? The goal is to understand these characteristics and how they affect the design of Internet protocols. For example, we need to track round trip time in TCP because it changes over time. TCP congestion control protocol needs to dynamically adjust the congestion window because available bandwidth over a link will vary a lot depending on the competing traffic.

This project requires students to collect measurement data from the current Internet. They have to compare the characteristics of the different paths chosen, and then observe how the performance changes over time for each path. A simple tool such as ping can be used to measure the round trip time. We can send ping traffic to some popular websites, such as www.google.com. This has two problems. The first problem is related to the high frequency pings performed automatically by programs. Sometimes they may be misunderstood as denial of service attacks. We did get emails requesting explanation or termination of tests from those sites in the early days. The second problem is that these tests are one-way operations. We can send ping to these well-known websites, but the responses are pre-determined by the ICMP protocol. We can use other tools, but have similar problems that we cannot program the target site to reply in the way we want.

The alternative approach is to get guest accounts from friends in other universities or corporations. Typically, the number of these guest accounts is very limited, partially because various security concerns lead organizations to impose strict policies on guest accounts allowed.

GENI provides an easy way to have machines located around the country and even globally. We can define a topology consisting of machines from selected aggregates. We can send traffic between these nodes either through the dedicated links, or through the normal Internet paths via their public interfaces. We have control over both ends of the path we are interested in. With this capability, students can write/run their programs or use existing tools such as pathrate, pathchar, on the experimental machines allocated by GENI. These programs can collect path information over the wide area networks over time.

6. Related Work

Virtualization (vmware, virtualbox, etc) [13]–[15] has been widely used in Operating System/Networking courses for projects because it is a simple and cost effective way to give students complete access over a computer. Logical links can also be set up among VMs to establish the topology of a network experiment. However, VMs have limitations on the flexibility of communication with other machines and do not offer the performance of real hardware. More importantly, they do not offer the ability to control the delay/bandwidth of the network links, or the geographic location/distribution we get in GENI.

Emulab [5] is the original software the current ProtoGENI is based on. It offers many of the features of GENI, such as easy allocation of resources, easy setup of experiments and friendly graphical user interface. However, it is a single site facility and thus does not provide the geographic distribution of resources. Besides, it consists of a cluster of PCs and does not offer the wide range of resources (wireless/mobile, compute servers, juniper, openflow, optical routers, etc) as GENI does.

Planetlab allows users to set up experiments using Vserver of Planetlab nodes widely distributed across the Internet [4]. It offers geographic diversity, but does not offer the ability to control all layers of the network. The system does not set up links for communications between experimental nodes. Instead, the user has to create overlays links among nodes to set up the topology of an experiment. Using Vserver prohibits users from controlling the operating system in experimental nodes. It also results in poor performance compared to physical nodes. Although resources are distributed, there are very few resources at each location – typically 2 nodes at each location.

Special purpose network lab facilities (like the hands-on lab at Purdue, or the lab experiments by Jorg Liebeherr and Magda El Zarki) can be used for hands-on experiments [12], [16], [17]. However, these dedicated labs have cost and maintenance issues, are not easily (concurrently) shared by students, typically offer only a single configuration (e.g., network topology, operating system, application software) that can only be modified on the timescale of days or weeks (if at all possible), and can only be used when the lab is "open". It also requires a lab monitor to allocate lab resources to users. Moreover, because resources are limited, it is is impossible to experiment with large-scale systems consisting of many nodes separated by large geographic distances.

7. Conclusion

Hands-on experiments are an essential part of computer science courses for students to learn practical skills by doing. Many of these experiments are enabled by GENI because of its unique features, such as its easy-to-use graphical interface, quick setup and teardown of experiments, and a wide range of available resources. Though GENI can further improve the access methods of experimental nodes, it definitely makes a dramatic advance in the ways in which the hands-on experiments can be done. To explore further, we recommend the ProtoGENI website [10] and the GENI tutorial [9].

Acknowledgment

This material is based upon work supported in part by the National Science Foundation under grant CNS-0834243. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of BBN Technologies Corp, the GENI Project Office, or the National Science Foundation.

References

[1] "NSF Future Internet Design Project," http://www.nets-find.net/.
[2] "NSF Future Internet Architecture Project," http://www.nets-fia.net/.
[3] The GENI Project Office, "GENI System Overview," http://www.geni.net/docs/GENISysOvrvw092908.pdf.
[4] "Planetlab: An Open Platform for Developing, Deploying, and Accessing Planetary-scale Services." http://www.planet-lab.org.
[5] "The Emulab," http://www.emulab.net.
[6] "Million Node GENI," http://groups.geni.net/geni/wiki/ GeniAggregate/MillionNodeGeni.
[7] "GENI ShadowNet Project," http://groups.geni.net/geni/ wiki/Shadownet.
[8] "The HomeNet Project," http://homenet.hcii.cs.cmu.edu/.
[9] J. Duerig, R. Ricci, L. Stoller, M. Strum, G. Wong, C. Carpenter, Z. Fei, J. Griffioen, H. Nasir, and J. R. ans X. Wu, "Getting started with geni: A user tutorial," *ACM SIGCOMM Computer Communication Review (CCR)*, January 2012.
[10] "ProtoGENI." http://www.protogeni.net.
[11] D. Farinacci, T. Li, S. Hanks, D. Meyer, and P. Traina, "Gneric Routing Encapsulation (GRE)," RFC 2784, March 2000.
[12] D. E. Comer, *Hands-On Networking with Internet Technologies*. Upper Saddle River, New Jersey: Prentice Hall, 2004.
[13] "VMware." http://www.vmware.com.
[14] "Xen Source™." http://www.xensource.com.
[15] "The Linux Kernel-based Virtual Machine (KVM)," http://kvm.sourceforge.net.
[16] J. Liebeherr and M. E. Zarki, *Mastering Networks: An Internet Lab Manual*. Addison-Wesley, 2004.
[17] J. C. Adams and W. D. Laverell, "Configuring a multi-course lab for system-level projects," in *Proceedings of the 36th SIGCSE technical symposium on Computer Science Education*, February 2005, pp. 525–529.

The Partnership for Early Engagement in Computer Science (PEECS) Program: Teaching African-American Middle-School Students Computer Science

A. Nicki Washington, Ph.D., L. Burge, III, Ph.D., Marlon Mejias, and Ketly Jean-Pierre
Department of Systems and Computer Science, Howard University, Washington, DC, USA

Abstract **-** *To address the challenge of increasing the number of African-American K-12 students retained in the CS pipeline, and thereby prepared to matriculate in computer science university programs, the Partnership for Early Engagement in Computer Science (PEECS) Program was developed. PEECS is a unique partnership between Howard University's Department of Systems and Computer Science Department and Curriculum and Instruction, Howard University Middle School of Math and Science [(MS)^2], and Google, Inc. to create a culturally-relevant pedagogy that targets African-American 6th-8th grade students. PEECS presents a novel approach to feeding the CS pipeline, by creating a program that includes culturally-relevant pedagogy and experiences to teach middle-school students computer science fundamentals, such as algorithms, problem solving, programming, and computational thinking. Through this unique program, which includes parental engagement, students are able to understand the broad range of opportunities in CS, develop basic competencies in CS fundamentals, and maintain a positive perception of CS.*

Keywords: *Computer science education, cultural relevance, pedagogy*

1 Introduction

While computer science continues to permeate every aspect of society, the number of high-school students that are adequately prepared to enroll in university computer science programs is declining. According to an Association of Computing Machinery report, the percentage of high schools with rigorous computer science courses dropped from 40 % in 2005 to 27 % in 2009 [1].

To address the challenge of increasing the number of African-American K-12 students retained in the CS pipeline, and thereby prepared to matriculate in computer science university programs, the Partnership for Early Engagement in Computer Science (PEECS) Program was developed. PEECS is a unique partnership between Howard University's Department of Systems and Computer Science Department and Curriculum and Instruction, Howard University Middle School of Math and Science [(MS)^2], and Google, Inc. to create a culturally-relevant pedagogy that targets African-American 6th-8th grade students. PEECS presents a novel approach to feeding the CS pipeline, by creating a program that includes culturally-relevant pedagogy and experiences to teach middle-school students computer science fundamentals, such as algorithms, problem solving, programming, and computational thinking. PEECS is designed to not only expose, but also teach and retain students in CS, such that they are better prepared for introductory computer science courses upon entering high school. Through this unique program, which includes parental engagement, students are able to understand the broad range of opportunities in CS, develop basic competencies in CS fundamentals, and maintain a positive perception of CS. The PEECS curriculum was developed using Google's Computing and Programming Experience (CAPE) Curriculum and CS Unplugged Curriculum. *Google Computing and Programming Experience (CAPE) Expansion Program*

The Google Computing and Programming Experience (CAPE), is a four-week summer program located in Google offices across the U.S. The program is designed to excite and inspire rising 9th-grade students in computer science. With particular emphasis on student experiences, CAPE aims to inspire a future generation of creators in computing. Students experience daily life at Google and witness various aspects of CS, from its impact on society and culture to the role it plays in the work and personal lives of individuals. CAPE leverages the Computer Science Unplugged curriculum [14], and focuses on teaching students about algorithms, computational thinking, programming, and computing theory through interactive and collaborative workshops, courses, projects, guest speakers, and field trips. At the end of the four weeks, students have the opportunity to showcase their projects created during the program.

1.1 Importance of Cultural Relevance

Culturally-relevant pedagogy must leverage students' culture as a vehicle for learning to develop academic excellence, while maintaining cultural identity and engaging the world and others critically, [15].

Culturally-relevant pedagogy is based on three criteria [15]:

1. Students must experience academic success.
2. Students must develop and/or maintain cultural competence.

3. Students must develop a critical consciousness through which they challenge the status quo of the current social order.

With the onset of technologies such as Facebook, iPods, Xbox, smart phones, mobile applications, and more, African-American students are already actively engaged in utilizing computer technology. However, in order to transition them from consumers to creators of this technology, culturally-relevant curriculum must teach fundamental concepts such as algorithms, problem solving, and abstraction in the context of issues that affect their daily lives[7] [9].

Research indicates that successfully teaching underrepresented K-12 students requires being aware of and countering issues that prevent them from entering the CS pipeline [9]. Howard University is a Historically Black College and University (HBCU) that traditionally serves this underrepresented population, awarding the fourth highest number of engineering bachelor's degrees to African-American students in recent years [16]. These numbers underscore the significance of the university's role in educating America's future STEM workforce.

The presence of role models of the same gender and ethnicity are critical to engaging students in CS [17][18]. First, students are able to imagine themselves in that professional role. Second, role models can debunk stereotypes that students have about computer scientists, including what they look like and what they do. While the authors cute the difficulty in finding such role models, Howard University has the unique position of directly providing them, including African-American male and female computer science Ph.D.'s. In a field where it is especially challenging to identify such individuals, students participating in this program will be exposed to an entire team comprised of African-American male and female undergraduate and graduate students, and faculty members on a daily basis. This continuous exposure will help to not only illustrate the variety of individuals and activities in computer science, but also provide students a positive image of themselves throughout the course of the program. This group will help to not only dispel the myths about computer science and what a computer scientist looks like, but also further engage students in culturally-relevant pedagogy.

The review of the literature has identified several important points:

1. **There is a lack of diversity in computer science**. As the baby-boomer generation enters retirement, the large gap forming in the U.S. workforce, referred to as the "Quiet Crisis," describes the increasing demand for and limited availability of qualified U.S. citizens available to fill positions in science, technology, engineering, and math (STEM) [30]. Many of these positions cannot rely on off-shore options, as they require security clearances (and thereby U.S. citizenship). While the domestic enrollment in computer science is increasing, the enrollment of underrepresented minorities is still dismal. According to results of the most recent Taulbee Survey, two-thirds of all bachelor's degrees conferred in computer science were awarded to white, non-Hispanics, with only 4% awarded to African-Americans [4][16]. Over the next 50 years, it is projected that the minority population will increase at a much faster rate than the non-minority population, with African-Americans and Hispanics increasing from a combined 22% to 40% of the workforce[5] [6]. These projections illustrate the importance of exploiting this untapped resource to quell this "Quiet Crisis."
2. **Students find computer science boring and not applicable to them and their interests**. Students often associate computer science with programming, without much correlation to other technologies and everyday activities that they use and engage in. In addition, students in urban areas do not readily identify computer science as a field that they can successfully pursue. Due to a number of factors, including negative perception of being difficult, lack of understanding about and preparation for computer science courses, and a lack of African-Americans and Hispanics in computer science, many students feel this subject area is not an area that includes them or their interests [7] [3][8-10][27-31].
3. **There is no formal computer science curriculum implemented at the high-school level for many school districts**. Research has shown that attracting and engaging students in the computer science pipeline must occur as early as possible. Much of the effort and research on K-12 computer science education has focused on preparing students for AP Computer Science [2] [7-10] [26]. However, many schools lack a CS curriculum that incrementally exposes students to computer science and teaches core CS fundamentals leading to AP CS, if desired.
4. **High-school students are ill-prepared for AP Computer Science**. If offered, many high-schools do not offer introductory-level computer science (CS) courses to successfully prepare students for AP Computer Science [2]. In many urban areas, AP Computer Science is not even offered. Instead, the lowest-level introductory computer technology classes are designed to teach students basic computer literacy, including how to use applications such as Microsoft Excel and PowerPoint [3]. These courses do not teach computer science fundamentals such as computational thinking and problem solving. In addition, those schools offering AP Computer Science typically attract a very small and similar group of students to the course, due to each of the aforementioned reasons.

Like many school districts, there is no formal K-12 computer science standard in Washington, DC Public Schools (DCPS). The closest equivalent is the Embedded Technology Standard, which infuses technology into other core subject areas, such as reading, world literature, math, and science. This standard does not focus on student development of core computational competencies. Instead, it uses technology to better understand core subject areas, such as creating multimedia presentations and spreadsheets [13].

2 Methods

The goals of the PEECS program are to:

1. Increase the number of African-American middle-school students with computational competencies, consistent with the learning outcomes identified in the CSTA Level I Model Curriculum for grades 6-8.
2. Increase the number of African-American middle-school students that successfully complete introductory-level high-school computer science courses, in preparation for AP Computer Science.

To achieve these goals, the following objectives are outlined:

1. **Improving Computational Thinking Skills and Computing Competencies.** The Computer Science Teachers Association (CSTA) developed the K-12 Model Curriculum, to identify topics and learning outcomes for K-12 computer science education [29]. PEECS leveraged Google's CAPE curriculum as a platform for building a year-long curriculum for grades 6-8. The curriculum maps to the CSTA Model Curriculum Level I topics and learning outcomes.
2. **Improving Perception of Computer Science.** PEECS will improve this perception through the culturally-relevant content, implemented in a year-long program to include coursework, hands-on projects, field trips, guest speakers, and more. To assist with the development and implementation of culturally-relevant pedagogy, Uplift, Inc., a local nonprofit organization, will partner with the other PEECS stakeholders (Howard University, $(MS)^2$, and Google).

The PEECS curriculum is divided into 6 instructional units. Table 1 outlines the six instructional units developed, the CSTA topics and learning outcomes achieved, and an example of a culturally-relevant module/lesson and activity. We note that each unit includes several learning modules, and each unit also maps to multiple CSTA learning outcomes. This mapping illustrates how each unit continues to reinforce computer science fundamentals throughout the academic year.

In addition to the curriculum, students participated in field trips around Washington, DC that have some computer science relevance. These include Google (DC), and the National Air and Space Museum. Guest speakers also met with students to discuss their experience in CS, including Howard SYCS alum and supporters (including industry affiliates) who are in CS careers.

Table 1. PEECS Curriculum Outline

Unit	CSTA Topics and Learning Outcomes	Sample Lessons	Sample Activities
1. Intro to CS	1.a-c; 2.a,b; 3.a,b; 5.c; 9.a-d ; 10.a,b; 11.a,c; 12.a-d	You Did WHAT?!? Breaking and Building a Computer	Given a list of 15-African-Americans in CS, research information on each using Google, Bing, and Wolfram Alpha, and create a virtual collage.
2. Problem Solving	6. a-c; 7.a, b	The Sort Shuffle: Dancing Our Way to a Sorted List	Find and print lyrics to a favorite song. Develop a code for compressing & decompressing the song. Share with classmates.
3. Programming	2.a,b; 7.a,b; 8.a,b; 11.b	Tell Me When to Go: Methods and Functions	Create a Scratch animation that tells about you or something you like to do.
4. Networking, Internet, and Web Design	4.a,b; 5.a,b,d; 8.b,c; 9.a-d; 11.b; 12.a	The Way You Move: Routing Data in Networks	Based on time vs. distance, create a shortest path to get to various destinations using only public transportation (bus and metro).
5. Mobile Application Development	1.b; 3.a; 8.a,b; 10.b,c	LOL Smiley-Face: SmartPhones vs. Computers	Create a mobile application interface for a scavenger hunt, to be used to input clues on field trips.
6. Robotics	7.a,b; 8.a,b; 10.a-c	Man Down! Robots in Society	Create a robot that will help sick students who are home alone.

PEECS will be conducted using two cognitive learning models: *open learning environment* and *collaborative problem solving*. These theories significantly contrast direct instruction, which many African-American students receive in classroom settings. In the first academic year, the open learning environment will be used in each classroom. In the second academic year, collaborative problem solving will be used. In the third and final academic year, the stakeholders will identify what learning environment was most appropriate to facilitate learning and implement that theory.

- *Open Learning Environment (OLE)*-The OLE is designed to address difficult material with a wide variety of methods and greater deal of flexibility. In contrast to direct instruction, OLE encourages open-endedness as a means to pursue learning goals, through the promotion of divergent thinking and multiple perspectives. In the context of PEECS, the OLE supports the following values: personal inquiry, divergent thinking, and hands-on, concrete experiences with realistic, relevant problems.
- *Collaborative Problem Solving (CPS)*-The goals of CPS are to develop content knowledge in complex domains, problem-solving and critical thinking skills, and collaboration skills. In the context of PEECS, CPS supports the following values: critical thinking and problem-solving skills, rich social contexts and multiple perspectives for learning, authenticity, ownership, and relevance of the learning experience for students, and developing a desire for life-long learning and the skills to sustain it.

3 Results

Currently, PEECS is completing its first year of implementation. Pre-assessment data was collected by an external evaluation firm from approximately 113 6th, 7th, and 8th graders at the Howard University Middle School of Math and Science. The data was aggregated across all grades and classes. Figure 1 presents the prior experience students had in CS. We note that, all references to CAPE are referring to the PEECS program. The CAPE reference was not changed by the external evaluator. 62 students reported no prior CS experience, or none other than using computer applications such as Microsoft Word, Excel, etc.

Figure 2 presents student feelings towards CS careers. This figure illustrates how, prior to the course, majority of the students have no desire to participate in CS careers or understand how influential they are to their future careers. 39% of the students are unsure if CS will be a part of their careers, while 25% of the students do not want CS to be a part of their future work at all.

Students were also provided assessment questions on various CS topics, including conditional statements, recursion, iterations, and more. Figure 3 presents the results of an assessment question on loops. Only 19% of the students answered the question correctly.

Figure 4 presents the results of an assessment question on binary numbers. Only 5% of the students answered the question correctly.

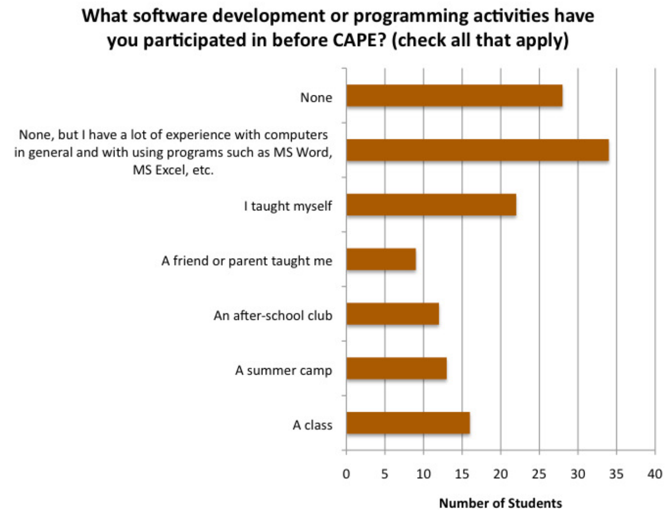


Figure 1. Prior CS Experience

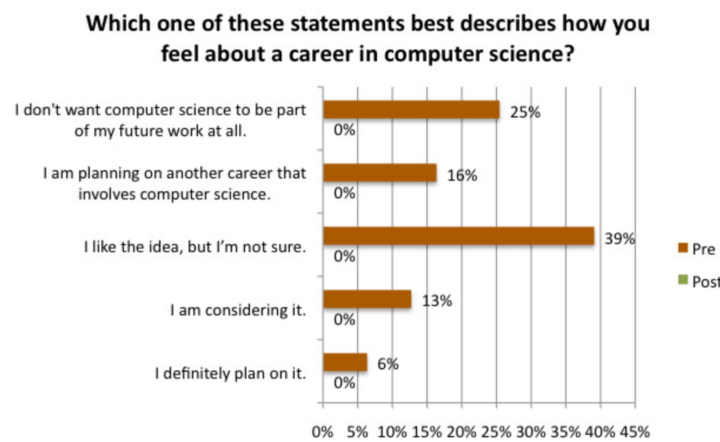


Figure 2. Attitudes towards CS Careers

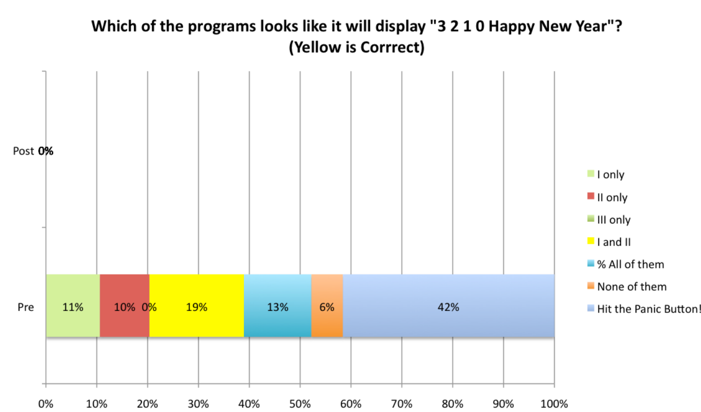


Figure 3. Loops

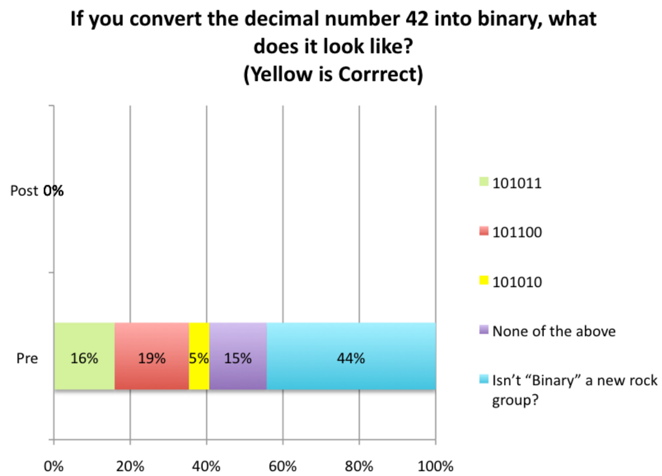


Figure 4. Decimal to Binary Conversion

The pre-assessment data illustrates that majority of the students lack an understanding of what CS is, as well as CS fundamentals, and how they apply to their daily lives. Throughout the 2011-2012 academic year, the PEECS program has taught a CS curriculum that aims to improve the understanding of CS fundamentals for all 113 students. Post-assessment data will be collected beginning May, 2012 to evaluate the effectiveness of the PEECS program and culturally relevant curriculum.

4 Conclusions

In order to not only engage students in computer science, but also develop students' computational thinking skills and computing competency, students must be adequately trained in computer science prior to high school. According to one study, 94% of 8th-grade students make course decisions related to preparing themselves for college and careers [24] [25]. Therefore, middle-school students who do not consider a STEM (and in this case, CS) major or career as a positive option, or maintain, at a minimum, a positive orientation towards STEM (CS) subjects may not enroll in the necessary high-school courses that would properly prepare them to enter those fields later.

PEECS will provide a blueprint for administrators as well as instructors on how to successfully teach African-American middle school students computer science using culturally-relevant pedagogy and experiences. It will serve to increase the number of middle-school students (and ultimately underclassmen high school students) prepared to pursue rigorous CS courses in high school and thereby bridging the "digital divide." This program will also train African-American CS undergraduates to pursue careers in CS education, which provides positive role models in CS and aligns with the goals of the CS 10K program.

5 References

[1] Computer Science Education Week. Data/Computer Science Facts. http://www.csedweek.org/resources/datacs-facts.

[2] Goode, J. (2010). "Connecting K-16 curriculum & policy: Making computer science engaging, accessible, & hospitable for underrepresented students." Proceedings of the 40th SIGCSE Technical Symposium on Computer Science Education.

[3] Margolis, J., Estrella, R., et.al. *Stuck in the Shallow End: Education, Race, and Computing*. Cambridge, MA: MIT Press.

[4] Taulbee Survey. Computing Degrees and Enrollment Trends. http://archive.cra.org/info/taulbee/CRATaulbeeReport-StudentEnrollment-07-08.pdf

[5] US Census Bureau (2004), "US Interim Projections by Age, Sex, Race, and Hispanic Origin". http://www.census.gov/ipc/www/usinterimproj/

[6] Toossi, M (2006), "A New Look at Long-term Labor Force Projections to 2050". Monthly Labor Review. 11/2006.

[7] Margolis, J., Goode, J., Bernier, D. "The Need for Computer Science." Education Leadership, vol. 68, no.5, Feb. 2011.

[8] Goode, J. (2008). Re-Programming high school computer science. Communications of the ACM, 51(11).

[9] Goode, J. (2008). Increasing diversity in K-12 computer science education: Lessons from the field. Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education, 40(1), 362-371.

[10] Goode, J., Estrella, R., & Margolis, J. (2006). Lost in translation: Gender and high school computer science. In W. Aspry & J. M. Cohoon (Eds.), Women and Information Technology: Research on Underrepresentation. Cambridge, MA: MIT Press, 89-113.

[11] CS 10K Initiative. http://www.computingportal.org/cs10k.

[12] Exploring Computer Science. http://www.exploringcs.org/

[13] OSSE Embedded Technology Standard. http://osse.dc.gov/seo/cwp/view,a,1274,q,561249.asp

[14] Computer Science Unplugged. http://csunplugged.org/

[15] Ladson-Billings, G. (1995). “But That's Just Good Teaching! The Case for Culturally Relevant Pedagogy,” *Theory into Practice,* 34, 159-165

[16] Gibbons, M. (2006), “Engineering by the Numbers”. American Society for Engineering Education. http://www.asee.org/publications/profiles/upload/2006ProfileEng.pdf

[17] Spencer, S.J., Steele, C.M., and Quinn, D.M. “Stereotype Threat and Women’s Math Performance,” Journal of Experimental Social Psychology, vol. 35, no.1, pp.4-28.

[18] Steele, C.M. and Aronson, J. “Stereotype Threat and the Intellectual Test Performance of African-Americans,” J. Pers. Soc. Psychol. Vol. 69, no.5, pp.797-811.

[19] Armor, D.; Conry-Oseguera, P.; Cox, M.; King, N.; McDonnell, L.; Pascal, A.; Pauly, E.; and Zellman, G. Analysis of the School Preferred Reading Program in Selected Los Angeles Minority Schools. Santa Monica, CA: Rand Corporation, 1976. (ED 130 243).

[20]Becher, R. M. Parent Involvement: A Review of Research and Principles of Successful Practice. Urbana, IL: ERIC Clearinghouse on Elementary and Early Childhood Education, 1984. (ED 247 032).

[21]Coleman, J. B., and Hoffer, T. Public and Private High Schools: The Impact of Communities. New York: Basic Books, Inc., 1987.

[22]Willis, C. L. "Strengthening the Bonds between the School and the Community." Journal of Educational Public Relations 9(1987): 28-31.

[23] “What Research Says About Parent Involvement in Children’s Education,” Michigan Department of Education.

[24] Akos, P., et. Al (2007). “Early Adolescenets’ Apsirations and Academic Tracking: An Exploratory Investigation. *Professional School Counseling,* vol. 11, no.1.

[25] Neathery, M. Faye (1997) “Elementary and Secondary Students’ Perceptions toward Science and the Correlation with Gender, Ethnicity, Ability, Grade, and Science Achievement. *Electronic Journal of Science Education,* vol.2, no. 1.

[26] Massachusettes Department of Education. “Increasing Student Interest in Science, Technology, Engineering, and Math (STEM).” http://digits.us.com/wp-content/themes/digits/Student-Interest-Programs-Using-Promising-Practices%28P3%29.pdf

[27] Mergel, B. Instructional Design and Learning Theory. http://www.usask.ca/education/coursework/802papers/mergel/brenda.htm#The%20Basics%20of%20Cognitivism

[28] Ormrod, J.E. (1999). Human learning (3rd ed.). Upper Saddle River, NJ: Prentice-Hall.

[29] Frost, D. et. al (2009). A Model Curriculum for K-12 Computer Science: Level I Objectives and Outlines. http://csta.acm.org/Curriculum/sub/ACMK12CSModel.html

[30] Jackson, S.A. “The Quiet Crisis: Falling Short in Producing American Scientific and Technical Talent.”Building Engineering and Science Talent. http://www.bestworkforce.org/ODFdocs/Quiet_Crisis.pdf.

Creativity, Modeling and Empathy: An Out-of-the-Box Methodology for Teaching Digital Design

John Hadjilogiou: jh@fit.edu, Samuel Kozaitis: kozaitis@fit.edu,
Bartel van der Veek: bvanderveek@my.fit.edu, Ashwin Acharya: aacharya@my.fit.edu,
Kenneth Shafer: kshafer2011@my.fit.edu, and Robert Sheffield: rsheffield2011@my.fit.edu

Department of Electrical and Computer Engineering, Florida Institute of Technology
150 W. University Blvd, Melbourne, FL 32901, USA

Abstract

This paper addresses the usefulness of a teaching methodology involving: creativity, modeling, and empathy (CME) for computer engineering/science students at any level. Through this methodology, first year students can master specialized and embedded system design approaches on a rudimentary level. It approaches digital design from a perspective that differs greatly from the approaches currently addressed in traditional textbooks. The models that emerge from this approach are: conceptualization/visualization/verbalization; block diagram; process searching; process documentation; process demonstration; enhanced block diagram, and finally the Process Description Implementation Diagram (PDID). The PDID is a special mathematical model which allows for a wide variety of possible implementations. It also hides the identity of the original specifications. This methodology seeks to promote creativity, empathy and engagement of students and faculty. This approach, used since 2009, is demonstrated for several applications, namely: specialized processors for arithmetic and non-arithmetic operations, and embedded systems with look-up tables.

Index of Terms- CME, Modeling, Creativity, Empathy, PDID, Out-of-the-box methodology

I. INTRODUCTION

In this paper creativity and design are the central focus of the classroom. This paper will show a new way of teaching first year students. It will show a method that will implement modeling and design in their everyday problem solving. Students using the creativity, modeling, and empathy (CME) approach may someday see new solutions to old problems, even faster, smaller solutions. This approach does not hold only for the Computer Engineering industry but for all engineering industries.

In order to teach a new way, a new environment is necessary. The ways to create a learning environment will be outlined and ethical considerations will be included.

This paper will address three applications of the CME approach. The first Application is that of a Specialized Arithmetic Processor. Three implementations, for the same application will be outlined using the CME approach. First, the design will be implemented by using a Finite State Machine (FSM) with one storage element per state. Next, the same application will be implemented as an FSM using the minimum number of storage elements. Finally the device will be designed by using the program approach (microprogramming). The second application will be a Specialized Non-Arithmetic Processor. The final application will be that of designing an embedded system for a Look-up table in both the Data and Program Memory of the PIC-18.

II. PREVAILING EDUCATIONAL ENVIRONMENT FOR CME

Student/ Faculty Interaction

The interaction between student and faculty is highly encouraged throughout, whether it is discussions during class or outside. The results of this healthy interaction in an open environment will be the emergence of novel designs and solutions. Creating this healthy student / faculty interaction is a vital component of CME.

Teaching and Experimentation

Students gain a substantial amount of knowledge by a hands-on approach. The outcomes are visible and results are more comprehensible, based on the designs and information gained from the enhanced block diagrams and PDID. The class and labs are designed to work parallel to each other in such a way that as the theoretical material is covered in class the actual physical, operation of that embedded system can be experimented with in the lab. For this reason the labs revolve around projects and experimentation that will train and guide students to a thorough understanding of the material. This allows the students to gain the big picture of how the system works as well as all the individual internal functionalities of the entire system.

Student Evaluation

As the semester progresses, assessments are periodically given to students covering topics that have recently been completed. This serves as a test to see what students have learned and understand, as well as a guide for the faculty to monitor the performance of the students. Faculty may also notice trends in the evaluations that display areas of concern and can therefore act upon those areas, before proceeding. This will make sure a majority of students are at par and no one is left behind. These evaluations also contribute towards the final grade and therefore encourage the students to be consistent and maintain a steady learning of the course material.

Adaptation

The models discussed in this paper are course independent and can easily be adapted for various other courses. Although the examples provide are directed towards the computer science/engineering fields, a broad spectrum of courses in a student's curriculum may use these methodologies advantageously.

Outcome

Through CME students will be able to creatively apply their skills to a wide variety of situations, applications, and designs.

III. ILLUSTRATIVE EXAMPLES

Example 1: Process of generating the mathematical model and three implementation designs for arithmetic specialized processors

The concept of circuit and logic designs has been taught in class using the Visualize, Verbalize, Demonstrate, and Enhance (VVDE) method, which allows students to learn from the system approach. The success of this method results in the generation of the Process Design Implementation Diagram (PDID). The PDID is an unconstrained representation that leads to the design of any possible implementation.

Some of these implementations include finite state machine with one storage element per state (one-hot code assignment), finite state machine with minimum number storage elements, and the controlled memory approach known as microprogramming.

In this paper, illustrative example $Z \leftarrow 2X_1+2X_0$ is used to describe the authors' new and unique approach. With this new approach all solutions will be implemented. The example chosen is simple enough to be addressed extensively in this paper and at the same time possesses the key feature to demonstrate the procedure.

A. Visualization / Verbalization Phase

The description of the above stated illustrative example is conveyed to the students in a variety of forms. After the description is fully explained and understood, the student spends time individually visualizing the big picture for their design. They then collectively verbalize the process in detail enough to be able to draw a simple block diagram. The student gets a firm understanding by verbalizing the process. Even if it appears to be a simple task, time needs to be allocated for this phase.

B. Block Diagram

The block diagram of our illustrative example is shown in Fig.1.

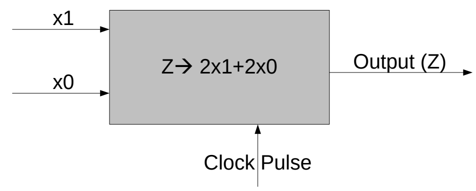


Figure 1: A SIMPLE BLOCK DIAGRAM FOR THE SYSTEM $Z \leftarrow 2X_1+2X_0$.

The X_1 and X_0 are inputs over which the designer does not have control. Z is an external output. Clock Pulse addresses the speed of the operations. The next step is o describe in timing fashion the process for its implementation.

C. Process Description

After some brainstorming, most of the students came up with the following process description for a given time i.

1. The sum of (two times X_1) plus (two times X_0) was added to the present carry and a number N was obtained.
2. The least significant part of N is the output Z and remaining part of N is the next carry.

Describing the process in words is the key and the most challenging part. After some elaboration, the students are asked to demonstrate and optimize the process with students as the elements. Optimizing in this context is interpreted as minimum number of students (elements) and each student can perform a dedicated task that is easy to remember.

D. Demonstration Phase

The described process was then demonstrated and optimized by the students as shown in enhanced block diagram.

E. Enhanced Block Diagram

The enhanced diagram is shown in (Fig.2.).

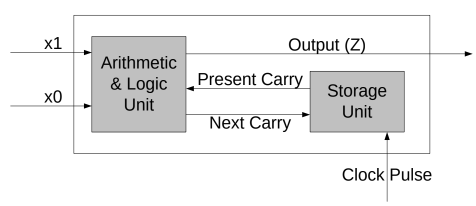


Figure 2: THE ENHANCED BLOCK DIAGRAM

The way this enhanced block diagram is conceptualized and demonstrated by the students' results in many distinct possible realizations. From this enhanced block diagram, the Process Description Implementation Diagram (PDID) is generated.

F. Process Description Implementation Diagram (PDID)

The above process description combined with the enhanced block diagram generates the PDID shown in (Fig.3.). The PDID acts as a road map to all possible conceivable implementations. This newly created/generated diagram facilitates the implementations of the various designs.

G. Various Implementation Designs

G.1 A Finite State Machine with a Single Storage Element per State

Students first use the PDID as a road map to create a finite state machine where a single state is represented by a single memory storage unit.

To create a finite state machine out of the PDID, the first step is to do the state assignment. In this case, the states are the storage elements of the carries and are shown in Table I.

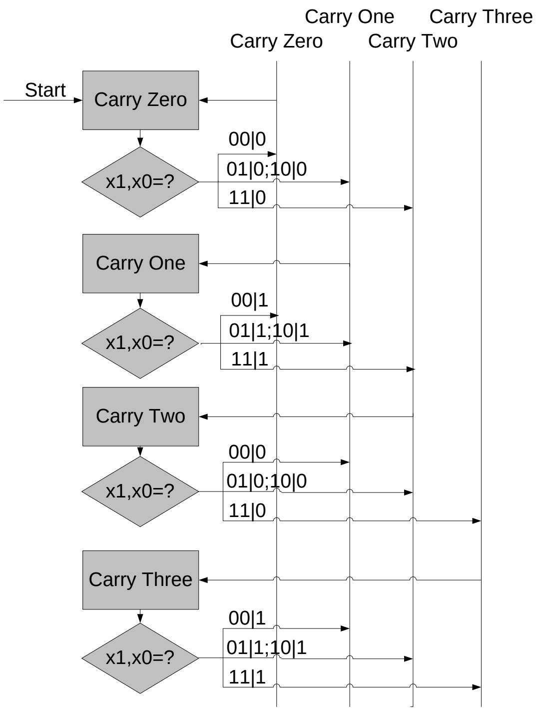


Figure 3: **PDID DRAWING FOR THE EXAMPLE SYSTEM $Z \leftarrow 2X_1 + 2X_0$**

TABLE I: ONE-HOT CODE ASSIGNMENT, ONE STORAGE ELEMENT PER STATE

	Y3	Y2	Y1	Y0
Carry 0	0	0	0	1
Carry 1	0	0	1	0
Carry 2	0	1	0	0
Carry 3	1	0	0	0

Next, based on the PDID from Fig. 3 and Table I, the input expressions for the storage elements (assuming the D-type is in use) are obtained by inspection and are the following

$$D_0 = Y_0\bar{X}_1\bar{X}_0 + Y_1\bar{X}_1\bar{X}_0 \quad (1)$$

$$D_1 = Y_0(\bar{X}_1X_0 + X_1\bar{X}_0) + Y_1(\bar{X}_1X_0 + X_1\bar{X}_0) + Y_2\bar{X}_1\bar{X}_0 + Y_3\bar{X}_1\bar{X}_0 \quad (2)$$

$$D_2 = Y_0X_1X_0 + Y_1X_1X_0 + Y_2(\bar{X}_1X_0 + X_1\bar{X}_0) + Y_3(\bar{X}_1X_0 + X_1\bar{X}_0) \quad (3)$$

$$D_3 = Y_2X_1X_0 + Y_3X_1X_0 \quad (4)$$

After the expressions for the D-storage elements the output Z expression can be generated using the output information from the PDID and is shown in the adjacency map (Table II).

TABLE II: ADJACENCY MAP

X_1X_0	00	01	11	10
Y_0	0	0	0	0
Y_1	1	1	1	1
Y_2	0	0	0	0
Y_3	1	1	1	1

From the adjacency map (Table II), the expression for Z is derived and shown in equation (5).

$$Z = Y_1 + Y_3 \quad (5)$$

The complete implementation is shown in Fig. 4.

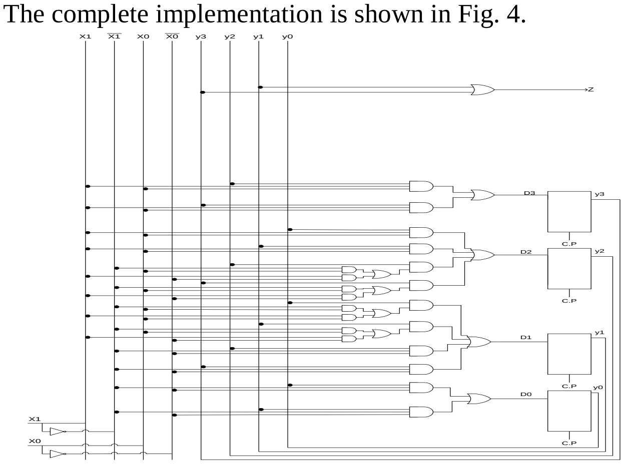


Figure 4: **THE FINITE STATE MACHINE OF THE SYSTEM $Z \leftarrow 2X_1 + 2X_0$, IT CONTAINS THE INPUT X_1 AND X_0, OUTPUT Z, AND THE INPUTS AND OUTPUTS FROM THE D-TYPE STORAGE ELEMENTS**

G.2 A Finite State Machine with Minimum Number of Storage Element

It is also possible to create a FSM using a minimum number of storage elements. For this example it requires four storage elements for the one-hot code assignment while requires only two for the minimum number of storage elements and shown in Table III.

TABLE III: MINIMUM NUMBER OF STATE VARIABLES

	Y1	Y0
Carry 0	0	0
Carry 1	0	1
Carry 2	1	0
Carry 3	1	1

The transition table can then be generated from the PDID. For the example system $Z \leftarrow 2X_1 + 2X_0$, the transition table with input for each storage elements is shown in Table IV.

TABLE IV: MINIMUM NUMBER OF STATE VARIABLE TRANSITION TABLE

Z	Y1	Y0	X1	X0	Y1	Y0	D1	D0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	1	0	1
0	0	0	1	0	0	1	0	1
0	0	0	1	1	1	0	1	0
1	0	1	0	0	0	0	0	0
1	0	1	0	1	0	1	0	1
1	0	1	1	0	0	1	0	1
1	0	1	1	1	1	0	1	0
0	1	0	0	0	0	1	0	1
0	1	0	0	1	1	0	1	0
0	1	0	1	0	1	0	1	0
0	1	0	1	1	1	1	1	1
1	1	1	0	0	0	1	0	1
1	1	1	0	1	1	0	1	0
1	1	1	1	0	1	0	1	0
1	1	1	1	1	1	1	1	1

From the table above, we obtain the following design equations (equations 6, 7, and 8); the complete design is shown in Fig. 5.

$$D_0 = Y_1 \oplus X_1 \oplus X_0 \quad (6)$$

$$D_1 = Y_1 X_0 + X_1 X_0 + Y_1 X_1 \quad (7)$$

$$Z = Y_1 \quad (8)$$

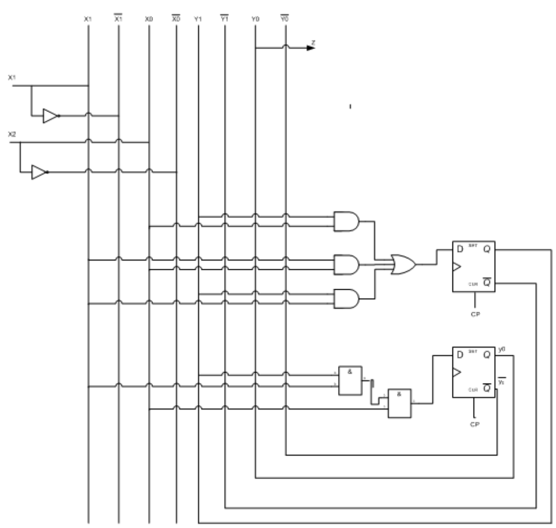


Figure 5: A FINITE STATE MACHINE USING MINIMUM STORAGE ELEMENTS.

G.3 Control Memory Implementation (Microprogramming)

The PDID also allows students to easily create a storage program machine implementation, also known as microprogramming. First, the content of the read-only memory (ROM) must be defined from the PDID, given in Table V.

TABLE V: CONTENT OF THE READ ONLY MEMORY (ROM)

Location	Output				Next State Location							
0	0	0	0	0	0	0	0	1	0	1	1	0
1	1	1	1	1	0	0	0	1	0	1	1	0
2	0	0	0	0	0	1	1	0	1	0	1	1
3	0	0	0	0	0	1	1	0	1	0	1	1

The first four columns are the output of every memory location, and the next eight are the memory location expressed in binary. The top location corresponding to carry zero is binary location 00 and the last one corresponding to carry three is binary location 11.

After that, the ROM implementation diagram can be drawn by inspection and is shown in Fig. 6.

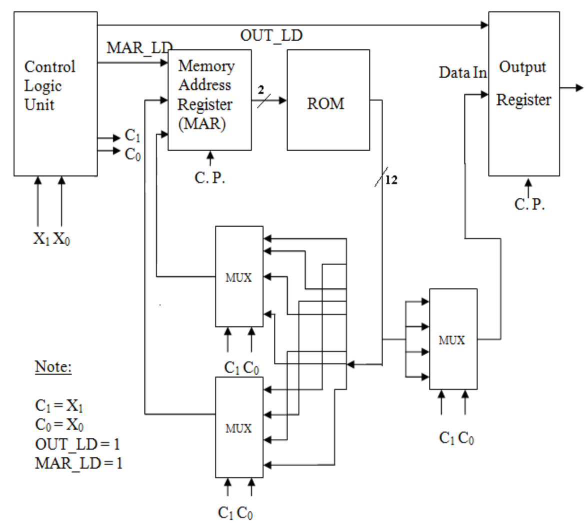


Figure 6: ROM IMPLEMENTATION-MICROPROGRAMMING

Example 2: Specialized Non Arithmetic Processor

It is important to realize that the top-down approach and PDID model are not some specialized trick for designing arithmetic processors, but rather allow the analysis and design of a wide variety of applications. The methods presented in this paper could, for example be used to design the control for the following dynamic telephone system under the conditions shown in Table VI.

TABLE VI: DESIGN SPECIFICATIONS

c1	c0	Function
0	0	Dial the entire ten digits
0	1	Dial the last eight digits
1	0	Dial the last six digits
1	1	Dial the last four digits

The sample number to be dialed is 3216055005.

The results should be output sequentially, decimal digit by decimal digit. That is to say that the correct output for an input of 11 is (101 then 000 then 000 then 101) NOT (1001110001101). Note that all digits of the given dialing number are less than 8 and thus only three bits of output are needed. In a more general system where the dialing number is unknown, a fourth bit would be needed. This is straightforward to implement but in the case of this example is superfluous.

The block diagram, enhanced block diagram, and PDID of this problem are shown in Fig. 7, Fig. 8, and Fig. 9 respectively.

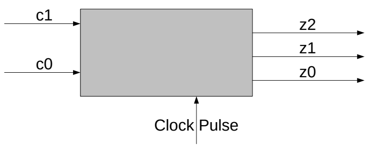


Figure 7: **BLOCK DIAGRAM**

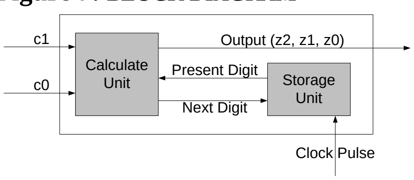


Figure 8: ENHANCED BLOCK DIAGRAM

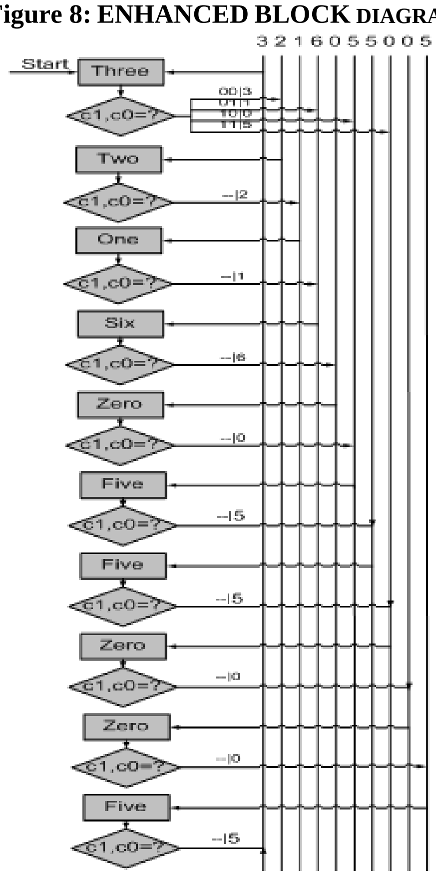


Figure 9: **THE PDID MODEL**

Note that by visual inspection of that PDID it becomes obvious that the Memory Control Approach (also known as microprogramming) is the implementation of choice. One possible final implementation is shown in Fig. 10, where the ROM table is shown in Table VII.

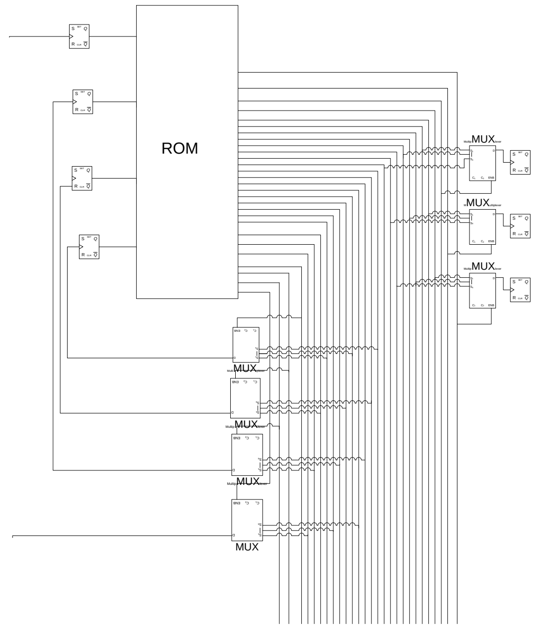


Figure 10: **SAMPLE FINAL IMPLEMENTATION OF SPECIALIZED NON ARITHMETIC PROCESSOR**

Table VII: **PRELUDE TO MICROPROGRAMMING**

c1 c0	Output				Next Location			
location	00	01	10	11	00	01	10	11
0000	011	001	000	101	0001	0011	0101	0111
0001	010	010	010	010	0010	0010	0010	0010
0010	001	001	001	001	0011	0011	0011	0011
0011	110	110	110	110	0100	0100	0100	0100
0100	000	000	000	000	0101	0101	0101	0101
0101	101	101	101	101	0110	0110	0110	0110
0110	101	101	101	101	0111	0111	0111	0111
0111	000	000	000	000	1000	1000	1000	1000
1000	000	000	000	000	1001	1001	1001	1001
1001	101	101	101	101	0000	0000	0000	0000

Example III: Embedded System- Look Up Table

Let us examine the following problem "Presuming that the look up table given in Table VIII was stored in program memory and that the lower three bits of PortC are connected to three switches, create a method to output the relevant stored ASCII character based on the status of the switches."

Table VIII: **LOOK UP TABLE**

Switches	000	001	010	011	100	101	110	111
ASCII	0	1	2	3	4	5	6	7

A. Look-up-Table in ROM

One possible method of solving this is to write a program in Assembly, another is to use the CME design method to create a microprogramming circuit. The resulting Assembly code could be as below if using a PIC18. (Mazidi [4])

```
ORG 0h
SETF TRISC
CLRF TRISD
B1 MOVF PORTC,W
ANDLW B'0000111'
CALL ASCI_TABLE
MOVWF PORTD
BRA B1

ASCI_TABLE
MULLW 0x2
MOVFF PRODL, WREG
ADDWF PCL
RETLW '0'
RETLW '1'
...
RETLW '7'
END
```

Another possible method is to use the PDID approach and implement the result via hardware is shown in Fig. 11.

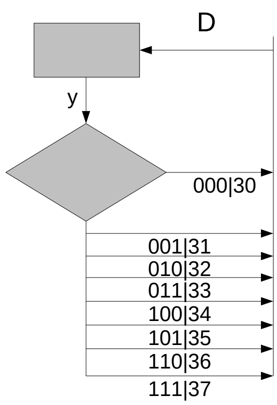


Figure 11: **PDID MODEL**

The logical equation for D, based off the PDID, is D=y. Table IX gives the output for this problem.

Table IX: **OUTPUT TABLE**

$x_2\ x_1\ x_0$	y
000	00110000
001	00110001
010	00110010
011	00110011
100	00110100
101	00110101
110	00110110
111	00110111

Now we use logic gates to directly implement this set of outputs occurring as the switches are changed. The complete design, as laid out in our logical equations is shown in Fig. 12.

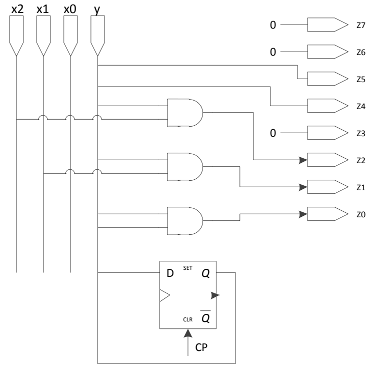


Figure 12: **MICRO-PROGRAMMED DESIGN**

Note that the storage element only outputs y=1, and thus it is possible to replace it with a 1.

B. Look-up-Table in RAM

Now suppose that the data were stored in the RAM starting at address 20h instead of the ROM. One possible way of creating the desired output would be the following program (Mazidi [4])

```
ORG 0h
SETF TRISC
CLRF TRISD
LFSR 2, 0x20
B1 MOVF PORTC, W
ANDLW B'01000111'
MOVFF PLUSW2, PORTC
BRA B1
END
```

However, the microprogramming method provides another way to retrieve the information stored at the given RAM locations: Create a simple circuit that output 0,0,1,0,0,x_2,x_1,x_0 and send that to the ram as the address, along with the instruction to read what is written at that address. Thus the resulting circuit is shown in Fig. 13.

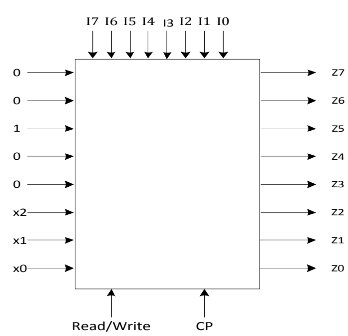


Figure 13: **RAM EQUIVALENT**

The advantage of using this style of modeling when teaching is that opens the minds of the students to creative approaches. The advantage of the PDID model in this case is that it does not represent a solution, but rather the solution set of whatever problem was modeled. That is extremely helpful for looking at the problem from new perspectives.

IV. FUTURE RESEARCH WORK

As this is an ongoing process of research with intentions to continuously improve upon previous methodologies. The goals set forth will focus specifically on the following areas:

Firstly, to achieve a concise summary of all the various feasible arithmetic and logical units implemented. To be able to weigh the pros and cons of each and as a result achieve improved designs.

Secondly, to elaborate on the instructions set of an embedded system, by performing each instruction within a fixed number of clock cycles.

Finally, to obtain an enhanced modeling approach for more elaborate systems.

V. CONCLUSION

Creativity and design methods are necessary for the future expansion of the industry. Using the "CME approach" as well as these modeling methodologies can lead to new creative designs for old and future applications. It is a mind opening approach with a variety of uses.

VI. Appendix A

Unleashing Our Inner Strengths and Bring Excellence in a Teaching Environment

The reading for this appendix can be accessed at the following website below.

https://sites.google.com/a/my.fit.edu/digitallogic/appendix/embrace-wholeness-in-teaching-methodology-for-computer-scientists-and-engineers/appendix-a

VII. Appendix B

Model Rules of Professional Conduct

The list of the Model Rules is in the following website.

https://sites.google.com/a/my.fit.edu/digitallogic/appendix/embrace-wholeness-in-teaching-methodology-for-computer-scientists-and-engineers/appendix-b

VIII. Acknowledgment

The authors would like to thank the undergraduate students who contributed to this paper and the current teaching methodology: **Robert Ritter III, Vincent Scotti, Jamie Bales, Tabitha Beavers, Baxter Brendan, Michael Wells, Brandon Baxter, Ranghou Ha, Mehrvash Ghorhaniha, Joseph Franzese, Nick Kerkorian, Ellen Lefton, Miles Davis, Jennifer Nguy, Jimmy Nguyen, Abdulrazzaq Meer, Michael Lindenmayer, Casey Santilli, Johnathan Dillon, Daniel Saint Pierre, Mohammad Shansah, Khalid Mujalled, Manuel Castro, Rod Mulligan, Alejandro Pena, Taylor Schluter, and Andy Chow** and the rest of the ECE1551/1552 students in fall 2011 and spring 2012 semesters.

REFERENCES

[1] M. Miller. *The Tyranny of Dead Ideas: Letting Go of the Old Ways of Thinking to Unleash a New Prosperity.* Times Books, 2009.

[2] "Change needed in engineering education" by James Plummer, dean of engineering and professor of electrical engineering at Stanford University, IEDM meeting , December 2008

[3] "Embrace Wholeness in Teaching Methodology for Computer Scientists and Engineers"John Hadjilogiou, Samuel Kozaitis, Pattarapong Rojanasthien, Abhijit Chakravarty and Ramakanth Subbagari, Florida Institute of Technology,2010.

[4] PIC Microcontroller and Embedded Systems: using Assembly and C for PIC18, Mohammad Ali Mazidi, Rollin D. McKinlay, Danny Causey Pearson by Prentice Hall, 2008

IX. Author Biography

John Hadjilogiou, Ph.D., P.E. is a professor of the Department of Electrical and Computer Engineering at Florida Institute of Technology and the chairperson of the Florida Engineers in Education (FEE) committee of the Florida Engineering Society (FES), State of Florida. He is the recipient of "Minuteman Plaque" from the Electronic System Division, Air Force System Command, Rome Air Development Center and the MITRE Corporation, in recognition of his leadership and vision for the continuous improvement in the quality of undergraduate engineering education nationwide. This prestigious plaque is normally awarded to high-ranking military personnel and very seldom awarded to members of academic institutions. He is recognized as a leader in CME approach and system approach in the designing of electrical/computer engineering systems.

Samuel Kozaitis, Ph. D is currently a professor and Department Head of Electrical and Computer Engineering where he supervises research in automated feature extraction of imagery. He has also worked for Digital Optics Inc., General Motors Research Laboratory, Wayne State University, and performed consulting for several government agencies and industry. His current emphasis involves the development of algorithms to reduce noise in imagery, and automatically detect important features. He is a member of IEEE, Sigma Xi, SPIE, and Tau Beta Pi and received the outstanding faculty service award at Wayne State University.

Bartel van der Veek is a Ph.D. candidate of the Department of Electrical and Computer Engineering at Florida Institute of Technology. He received his Master degree in Electrical Engineering from the Technical University in Eindhoven, the Netherlands in 2009. His specialization is in the field of control systems, but has done research in a variety of fields including modeling and control of flexible structures, fiber optic sensor systems, wireless communication systems, electromagnetic compatibility, and electromagnetic wave propagation. At Florida Institute of Technology he is the instructor for Signals and Systems, and assists Dr. Hadjilogiou with the Computer Design laboratory.

Ashwin A Acharya is a graduate student in Computer Engineering at Florida Institute of Technology. He is currently a teaching assistant for Dr. Hadjilogiou's Digital Logic and Computer Design laboratory. He takes an active role in generating evaluations/assessments and creating a unique website for each course. He performs logic circuit simulation and microcontroller programming through Xilinx and the MPLab respectively. He is a member of IEEE, Tau Beta Pi and Phi Eta Sigma honors Societies. His key research interests include speech recognition, image processing and embedded systems.

Robert A. Sheffield is an undergraduate Electrical Engineering student at the Florida Institute of Technology. Prior to Transfer to Florida Institute of Technology he attended Brevard Community College, where he graduated with honors with an Associate's Degree in Drafting and Design Technology. He is a member of Phi Theta Kappa (national junior and community college honor society), IEEE, SIAM, and has been elected chair of the student chapter of IEEE, FIT-IEEE, for the 2012-2013 Academic year.

Kenneth R. Shafer is an undergraduate Computer Engineering student at the Florida Institute of Technology. He graduated from Pine View School for the Gifted with honors. He is a licensed Ham Radio operator (call sign: KJ4IFI). He also is a National Merit Scholar and a member of the National Honor Society.

Using Scaffolded Activities to Teach Computational Thinking to Middle School Girls

Heidi Webb[1] and Mary Beth Rosson[2]
The Pennsylvania State University, College of IST, University Park, PA 16802
[1]hcw120@psu.edu, [2]mrosson@psu.edu

Abstract— *We describe a set of workshop activities designed around computational thinking concepts and skills for GirlsCreateIT!, an after-school program to provide middle school girls the opportunity to learn about computing. Each workshop activity was created to introduce the girls to computing tools and computational thinking while also being fun and motivating; examples include working with stories and games as well as simple web applications. We evaluated the effectiveness of the activities over six weeks, with two groups of 14-16 girls. Our evaluation confirmed that the activities were engaging to the girls, and that the scaffolding we designed into the activities allowed for quick start up in these relatively brief after-school sessions, while still meeting learning objectives. In this paper we discuss the design of the learning activities, findings from our preliminary evaluation, and implications for the future design and deployment of computational thinking activities for this age group.*

Keywords- computer science education, learning tools, instructional scaffolding

1 Introduction

According to U.S. Census data the percentage of children living in households that have computers rose from 75% in 1993 to 85% in 2003[1]. The number of students who have not seen or used a computer prior to the first day of school is decreasing each year. Depending on the grade level, computer usage skills range from literacy to fluency. However, mere use of computer technology does not equate to understanding *how* applications work, or recognizing the *role* that technology plays in everyday scenarios. Students at all levels of the K-12 pipeline need additional skills and understanding of technology applications to prepare them for an increasingly technology-centered future. For the educator, finding an effective tool or application for teaching computational concepts continues to be a challenge.

In 2006 Jeannette Wing began a discussion on the need to expose students of all grade levels to computational thinking (CT) skills and techniques, which include problem solving, pattern recognition and algorithm design [2]. Others have reinforced this message and there is now a shift occurring in U.S. high schools to convert computer science (CS) courses that used to focus on computer programming to a curriculum inspired by CT concepts, as well as integrating CT into other courses in the high school curriculum (e.g., journalism, general science) [3, 4].

One illustrative program that has been favorably received by high school educators is "Exploring Computer Science" (www.exploringcs.org). This curriculum consists of modules that can be interchanged and arranged to accommodate the many variations useful to schools with different curricula. Students learn about computing and its relationship to the world around them through projects that emphasize creativity in the building of computational artifacts. For example, one module has the students learn about computing components through interviewing a family member for details in purchasing a computer; from the specifications discovered during the interview they learn about how computing components meet functional requirements. Other modules have the students perform scavenger hunts on the internet, learning about search tools and online resources available for future school activities; through the use of games, stories and webpage creation students can also learn basic concepts of software design and programming. Programming is still a part of the coursework, but is introduced and practiced as a tool that can be used to aid in discovery of new ideas and concepts [5].

These changes occurring in the high school computing curricula are exciting, in that they point to an entirely different and more conceptual view of computing as a pervasive supporting technology in the world. To date however, little attention has been devoted to analogous options suitable for middle school students. This is surprising given that a middle school child can be viewed as a 'digital native, which implies knowledge and comfort with technology' [6]. That is, one might think that they are already exposed to a range of technologies and should be able to leverage that general use to explore more conceptual issues.

Middle school is a pivotal time for both boy and girls; they are generally in a new school, can begin to select elective courses and are interested in exploring different types of extra-curricular activities. This age is also the time at which the gap between boys and girls with respect to computing begins to widen, as girls (often under the influence of their peers) turn away from courses that involve technology learning. This general situation suggests that middle school might be a particularly important time to involve girls in CT learning activities and projects.

This paper discusses the design of computational thinking computing activities for the GirlsCreateIT! after-school program for middle school girls, a preliminary evaluation of the activities, and the implications for future deployment and use of such activities. Because CT is a complex and skill-based domain of learning, we incorporated learning scaffolds to help in managing the active learning process [7]. In this we were guided by two simple research questions:

- *What CT concepts and skills are comprehensible and usable by young women of middle school age?*
- *How can the educational strategy of scaffolded examples be used to engage middle school girls in computer activities, while also introducing and building an understanding of CT concepts and skills?*

The contributions of this work are to provide activity designs integrating CT concepts and the results from the initial workshop evaluations.

2 Related Work

2.1 CT and Computer Science Education

An emerging body of work is investigating CT concepts and computer science education for the K-12 community. Repenning [8] has developed curricula emphasizing CT tools that have been focused on teacher training and authoring. In addition, a group of researchers and practitioners have initiated discussions about CT as part of the K-12 curriculum objectives and teaching approaches. In the summer of 2009, the Computer Science Teachers Association (CSTA) and the International Society for Technology in Education (ISTE) began a multiphase project aimed at developing an operational definition of computational thinking for K-12 [9]. Table 1 maps the general CT concepts to these operational computer programming concepts.

Table 1 - CT Concept/Computer Activity Mapping

CT Concepts	**Computing Activities**
• Problem Solving ○ Analyze ○ Design solutions	Identifying important pieces of problem Brainstorming about computer solutions Organizing solution steps as an algorithm
Implement designs	• Use of specific software concepts ○ Expressions ○ Control Structures ○ I/O • Data Types and Structures ○ Searching ○ Sorting ○ Test and Debug
Abstractions	Use of variables and alternate solutions
Understand the Vocabulary	Communicate through written documentation the problem solving process of problem decomposition Using programming vocabulary (coding)

This concentration of concepts all have a foundation in computer programming, the standard mechanism used to implement an information-processing solution. Computing activities invariably involve some form of problem solving to generate ideas about solution approaches and to design implementations. The level of abstraction needed for the solving of specific problems can be tailored from simple concepts about the variables needed to store information, to the complex concepts of list processing using searching and sorting. A central concept in computing implementation is the testing and debugging of one's design. Vocabulary is always important for learners, particularly when new concepts are being introduced and the learner needs to communicate with teacher or peers to complete an activity. For these reasons we have focused on these concepts in the computing activities we designed to teach CT.

2.2 Middle School Girls and Computer Science Education

Young girls enjoy working with computers but they often expect there to be a purpose or goal that motivates the use of computing technology. Stories and games have been found to work well in providing activities for girls of middle school age. For example, research conducted with Storytelling Alice found that middle school girls not only enjoyed "programming" a story that was relevant to them but they also enjoyed turning it into a movie [10]. It is also important that girls are provided learning activities in computing environments which build their self-confidence and provide opportunities for a positive learning experience.

When girls are not comfortable in a learning context, their attitudes of inadequacy can transfer to their perceptions of the subject matter, especially if it involves computing technologies. It is important to a young woman entering the new learning environment to fit in and be successful. Working in isolation may not be the best motivator for a young girl who is working on fitting in and also trying to build her confidence in her new learning environment. For some young girls, self-confidence becomes a personal barrier that inhibits them from seeing the computer as nothing more than a tool they are unable to conquer or master [11]. In contrast, working in small groups or pairs has been successful in creating an environment which is positive and motivating for young women.

It is important that CT learning activities for middle school girls takes into account not only the learning activity but also the context of learning, so as to best engage and motivate them to learn about computing technologies.

3 The CreateIT! Workshops

A series of six workshops were designed for weekly 90 minute after-school sessions. The workshop activities concentrated on CT: problem solving and abstraction as it contributes to the design and implementation of information-processing solutions. Each workshop session was organized into activity units comprised of interactive stories, games, database/webpage and interactive structures. In all activities, the participants were given an activity workbook that scaffolded the use of the tool and associated activity; the workbook served a secondary purpose of gathering answers to conceptual question concerning the day's activities. The style and structure of these workbooks are a key element of the scaffolding we provided for the learners, modeled on prior work with "minimal manuals" for adults learning new computer skills [12]. By the end of each session the girls had not only been introduced to a particular computer application through an activity that included CT concepts but they also created a fun computing artifact. Though we conducted the workshops in a series, they were modular and could be held as stand-alone units.

3.1 General Workshop Format

Each workshop was designed to enable the middle school girls to complete an activity that introduced and exercised CT concepts; the result was a computer artifact built with a computing tool selected specifically to support that activity. Working in pairs the girls were provided a workbook and a scaffolded example for each week's activity. In earlier research studying novices learning object-oriented design, we demonstrated the benefits of *scaffolded examples* that introduce complex concepts and skills in a graduated fashion[7]; we adopted similar techniques for the CT workshops. Novices can become overwhelmed if asked to start from scratch when using a computing tool for the first time; starting from a working example offers an opportunity to explore and build confidence as new skills are acquired. If the example artifacts are carefully designed, they can also convey concepts that are implicit in their design. In recent work, we showed that high school girls benefited from several types of scaffolding – including examples – in learning about databases and simple web development [13].

All activities use a scaffolded example – a working artifact that introduce CT concepts through learners' exploration and subsequent extension or modification [7]. The scaffolded example provides information implicitly to students, but we also wanted to offer guidance aimed at students who are between the stages of concrete and abstract development stages. They may need additional scaffolding to grasp key concepts like abstraction or reasoning about alternate solutions. For this reason, we created workbooks to guide the girls through the activities and help them to learn more about the environment through exploration.


Activity 1: Exploring the FishFoodFun Game

The game **FishFoodFun** shows has our hungry fish trying to avoid eating the bad fish food. When bad food is eaten, the fish goes from being healthy (green color) to feeling sick (a pink color). When too many bad fish are eaten then the fish doesn't feel well enough to eat anymore and stops eating-- which means the game is lost. If you can get the fish to eat enough healthy fish before getting too sick then you have won the game!

1. It's important we understand what is working with the game before we begin to make changes. Look over the **FishFoodFun** game that is open in the Scratch application. You will be using the arrow keys to move your fish. Select the **green flag** and run the game.
2. Let's analyze the game by using the following table. Once you have played the game a few times you will want to record what you observed.

Game Elements	Write down what you noticed
Arrow keys	
Bad Food Count	
Good Food Count	
Win Message	
Lose Message	

Figure 1 - Activity Workbook Page

Each workbook was designed to correspond to the scaffolded example and the programming tool the girls would use to produce the computing artifact for that week (Figure 1). The workbooks were also designed as a "minimal manual", an instructional guide that offers goal-evoking questions and prompts to help learners think about relevant knowledge, but expects them to formulate and enact the specific steps needed [14]. Learning materials that provide opportunities and support for users to actively explore computing tools helps them to engage more actively in the learning domain and build richer understandings [15].

The CT concepts were organized into the high level constructs as proposed in previous discussions of CT, e.g. [9], as well as our operationalization of these abstract skills within specific computing technologies. Table 2 summarizes the example and related CT concepts for each session. For instance, the first activity uses a story that introduces design and implementation of solutions that solve a specific problem; the second uses a simple game that introduces further abstractions needed to improve the game, and so on.

Table 2 Activities and CT Concepts

Activities	Scaffolded Example	CT concepts
Interactive Stories	The activity involves a story about a Vet and a Mama wolf which needs the sequential sequence of the story fixed, dialogue corrected to match the character activities and use variables to control the action sequence based on users input.	• Design and implement solutions to problems • Use the vocabulary • Recognize abstractions and move between levels of abstractions • Test and Debug
Game	The activity involves working with a game which is considered "broken"; the game needs the right and left player controls fixed, has variable to control losing the game needs to be created and the win/lose messages are also missing. determining what needs to be fixed and then implementing their corrections and modifications	• Design and implement solutions to problems • Use the vocabulary • Recognize abstractions and move between levels of abstractions • Test and Debug
Database/Web page	The activity for this workshop is working with the GirlsCreateIT! Database creating queries to access stored data and then produce a webpage from the queries. Images are worked on with a pixel editor to work with alternate data sources for the webpages.	• Design and implement solutions to problems • Use the vocabulary • Recognize abstractions and move between levels of abstractions • Use innovation and creativity with technology
Interactive Structures	The activity for this workshop involves working with the Lego Robotic Mindstorm System. The participants learn about external sources of input by working with sensors (touch and light) and output by activating a motor by creating an interactive display for a park exhibit.	• Design and implement solutions to problems • Recognize abstractions and move between levels of abstractions • Use innovation and creativity with technology • Test and Debug

3.2 Workshop Activities

Each workshop activity was designed to be completed in 90 minutes, leaving time for orientation at the start and questions or personal work at the end. The girls worked in pairs sharing a computer on which the particular tool and example was installed (e.g., Alice or Scratch). Each activity included opportunities for the girls to add embellishments or creative additions to their projects, such as adding additional scenery, modifying existing objects by changing color or size or even importing new objects. We now describe each activity in more detail to provide its design rationale and procedural details.

1) *Interactive Stories (Alice 2.2):* Using a pre-existing story (Figure 2), participants work through a combination of additions to the storyline and events. The activity requires the girls to break down a problem into smaller units, design a workable solution to each problem piece, turn these solutions into steps (an algorithm), and implement the solution to enhance the story. Programming-related learning objectives of methods, variables and decision logic were emphasized in this activity. The artifact created from the activities in this workshop shop was an interactive story that could be controlled by the user's response to a question. The participants also had the opportunity to add personal embellishments to the story such as deciding which actions the characters would exhibit and new dialogue.

Figure 2 - Mama Wolf and Vet Story

2) *Game Activity (Scratch 2.0):* In the game activity participants work with an incomplete game (Figure 3) that needed to be corrected to work properly. The programming-related learning objectives in this workshop were more complex than those in the story editing, for example inclusion of more extensive decision logic, and the creation and setting of variables to control what the game displays. Analysis of the artifact prior to making changes *is* important as well, because understanding is required to do more testing and debugging with the game. To convey discoveries and observations as they work on the artifact the participants *also need* an understanding of basic *computing* vocabulary. The activities begin with a semi-functioning game that needs to be corrected. For instance, a game counter to track and decide about a win was missing, and the keyboard controls are not set up correctly for full game control (only up/down work). Based on an initial analysis of the example game, a design is envisioned; modifications are designed and then completed. At the end of the activity the participants have a working game where they have had a chance to add creative elements while learning more about if statements, loops and variables.

Figure 3 - Scratch FishFoodFun Game Scene

3) *Database Workshop Activity (GirlsCreateIT Database Appliction):* This activity involves the girls working with an appliction designed for this program which covers similar CT concepts as those covered in the story and game activity. Using concepts from algorithmic thinking the girls applied them to the creation of simple database queries (Figure 4) and again worked with abstraction in the form of

Figure 4 - GirlsCreateIT! Database

variables. In this case, the data is collected and analyzed from the queries for use in a webpage. Creating the webpage not only involves the use of the data but also the creation of icons; this adds elements of creativity as well as opportunities to learn file management techniques.

4) Interactive Structures Activity (Lego® RCX): This activity is the most complex of the activities due to the need for specialized software and hardware (Figure 5).

Figure 5 - RCX Unit

The Robolab programming environment was used with the Lego® RCX system; the girls used example programs to become acquainted with the programming environment, compiling and downloading the example, setting up the RCX unit for the program with the appropriate sensors and finally testing the program with their unit. Once finished modifying their version of the example program, additional sensors were introduced to extend the activity. The CT concepts from the previous weeks' sessions are again re-visited in the sense of analyzing a problem and designing an improvement to an existing situation; new concepts include novel input and output abstractions (sensors, motors), and the more complex solution process need to combine the different components into a single compuational artifact.

4 Workshop Experiences

A first version of the GirlsCreateIT! workshop series was created and delivered as part of the local School District 's Outreach Program, in Fall 2011. The workshops took place at a local middle school. The original plan was to offer one workshop per week for six weeks, with a maximum enrollment of 16; however due to high registration volume, a second (identical) set of workshops was delivered in parallel, resulting in a total of 30 participants. Research consent was obtained from 24 of the participants. Though the program was advertised to girls at all grades in middle school, the majority of attendees (67%) were in 6th grade; across the 24 girls the average age was 11.7. Though we also collected survey items assessing personal characteristics and attitudinal measures, this initial report focuses on qualitative data gathered from session observations and girls' comments.

4.1 Are girls this age able to grasp CT concepts and skills?

An important goal of the initial study was to design and evaluate a set of computing activities that illustrated and required the application of CT concepts, such as problem solving while also engaging and motivating girls of this age. Because we expected the girls to be between the ages of 11 to 13, we focused on problem solving, analysis and construction activities that have been shown to be motivating to this population – storytelling, games construction, and work with robotic components [16]. We included the activity on webpages and databases because our own earlier research had shown this to be motivating and accessible by slightly older girls (in 9th and 10th grades [13]). At the same time, we recognize that problem solving and abstraction in a computational setting can be complex. Thus at the highest level, *we needed to determine whether the activities were feasible* as learning problems for girls at this age.

We were pleased to see that in general the girls' reactions were quite positive – not only were most of them able to complete the suggested activities, but they clearly enjoyed themselves, and on the final day when they had extra time, many chose to go back and extend their previous projects (e.g. adding another dimension to their Fish game in Scratch or changing the storyline of the Vet and wolf).

In each activity the CT concepts of problem solving and abstraction were practiced in an interleaved fashion as the girls worked to understand and solve the problems that were posed. To guide these general activities, a computational problem solving process that included brainstorming ideas for potential solutions, choosing among alternatives, and designing and implementing a computer-based solution was introduced in the first week. For example, during the interactive story activity the vet needed to act scared because the example story illustrated her coming up to an angry wolf and showing no affect. To solve this problem the girls first brainstormed about real world reactions when a person is scared. They sketched out a sequence of events (the logic of an algorithm) for initiating a scared reaction, and finally explored features of the programming environment (method calls in Alice) to the computational elements they needed for their solution (e.g., they found the Vet character had the methods *move away from* so she could leave and also *pass out* if they had thought about fainting).

In one case a pair of girls had trouble articulating what might count as "acting scared" behaviors; for them, the problem related to the abstraction of working with a character in a story. It was too removed from their day to day experience; when instead we encouraged them to consider how this behavior related to them personally it became clearer what they needed to do. One suggestion for refining activities like this is to explicitly coach the young learners to put themselves into a story or game context when trying to decide what to do; although this may seem obvious to us (and many girls had no trouble jumping to that approach), the computational framing for the problem may be distancing enough that specific coaching will help. Analysis of similar problem solving episodes across the workshops is still underway, but our general observation of girls working on their own in the final week suggests that this framework for computation problem analysis and solution was learned and reused successfully. In future versions of the workshops, we will embed evaluation metrics that more precisely track and monitor the outcomes of these recurring episodes.

Programming terminology was introduced in each exercise when and if it applied to the computing activity. The story built in Alice introduced the notions of "variables" and "methods", and when the girls moved to Scratch for their work on games, these same terms were reused but now in a different setting. At this transition we did notice (based on workbook comments) some confusion about the term "methods" (in an object-based programming environment a method contains the code or script that is used to animate characters, manage variables, etc.). Alice refers directly to the term methods and we attempted to leverage this when moving to Scratch, even though its user interface refers to the same coding concept as a "script". We saw this as an opportunity for understanding the notion of "computer instructions" at a more abstract level, but this terminology change may need additional scaffolding to prevent confusion. At the same time, it did appear that the girls could understand that concept of a variable in both tool settings. We are still probing the workbook free-form comments for more evidence of their understanding of control logic (e.g., what one pair referred to as the “iffy thing”).

Understanding how data is stored and represented as variables in computing applications can become complex when students are learning new terminology and the interfaces. The database activity was designed for the girls to use a drag and drop interface so they would be successful with simple queries by minimizing syntax errors that can occur in programming. After the girls had written a few of the initial queries it became apparent that the activity could have moved on to more complex searches ***and/or*** logic. Several girls were disappointed when we told them they would need to run each query separately and that their alternate solution also would not work, writing a second query to search in the results already returned. One eager group of students knew you could search within returned results and wanted to know why the application did not support this feature. These classroom observations provided encouraging results that the CT concepts of designing and implementing solutions to problems, as well as moving between levels of abstraction were carrying over to workshops using different computing tools.

4.2 How did scaffolding help in learning CT concepts?

Our workshops included different types of scaffolding, and we wanted to determine whether and how the techniques could introduce the CT concepts in a way that encouraged the girls without overwhelming them. Because there was a two year age range among some participants, we expected that some girls might arrive with rather different levels of cognitive development, computing experience and general problem solving abilities. Thus it was important to provide customized assistance as needed during the workshops, to help encourage enough confidence and motivation for the girls to discover answers on their own. For this reason each workshop used scaffolded examples with an accompanying activity workbook where the girls would find instructions, tips and could record their answers to questions interwoven into the activity. Classroom assistants observed the girls’ work and offered advice as needed.

For the interactive story activity that used Alice 2.2, the scaffolded example included a camera object, a basic object used in all 3D stories in this environment; unfortunately the camera adds complexity to introductory projects, where rotation and movement in 3D space tend not to be a learning objective. The problem may not seem significant but it highlights issues that tend to shift the focus of the workshop to aesthetics. Instead of focusing on problem solving and the storyline, girls spent valuable time to get angles positioned correctly. Thus in this particular case, the scaffolded example raised complexities (i.e., exposing code that is not useful as part of the learning activity). If we continue to use Alice for future projects, we will explore ways to de-emphasize camera control and movement – this will be a challenge as it is a basic feature of most Alice programs.

In contrast, the *FishFoodFun* game built in Scratch allowed the girls to work in a simpler 2D environment; this in turn helped to keep the learning focus on key aspects of problem solving, use of appropriate vocabulary, and game testing and debugging. One measure of this is that the girls tended to write workbook comments that wondered about how to position characters in the Alice story, but in Scratch they spent most of their time solving the (deliberate) inadequacies of the example game. After observing the girls using both Alice and Scratch, we concluded that additional levels of scaffolding may be needed for use of Alice. While additional scaffolding of this sort may make it possible for them to use Alice, it may shift the focus too much away from the fundamental CT concepts that constitute our objectives.

For the database and web page activity the girls worked with an existing database using the GirlsCreateIT! tool; using this data, they used a drag and drop query builder and simple web page editor to extract structured information and display in on a web page. This tool and existing data store helped to place the focus of the workshop activity on learning how data is stored, accessing the stored data and problem solving when returned results were not the ones expected. Working with an established database of information on animals also reduced the complexity associated with data management (e. g. duplicate records, empty fields).

A key element in this activity was construction of queries to access the data in the database. Queries are normally constructed from specific programming instructions (e.g., in SQL, a SELECT statement with specific search criteria, for example ‘Select all animals > 1000’ where 1000 would be the value for pounds). In specifying such instructions, extra information in a query statement (e.g., ‘lbs.’) leads to a syntax error. This syntax requirement adds complexity to the learning activity and can overwhelm a novice as they learn how to work with structured data. However, with our custom web/database tool, the drag-and-drop query builder provides distinct shapes to help compose the search criteria, illustrating the basics of writing queries. With the operators only fitting into the distinct shape of a circle and the columns (attributes) using a rectangle; syntax errors were kept at a minimum.

Reducing the complexities of writing a query proved to be helpful during the workshops since some girls had difficulty in recognizing the greater than and less than operators; also they quickly discovered that spelling counted when working with search parameters.

The robotic activity creating interactive structures were the most complex, involving additional hardware (e.g. sensors, RCX brick) and a new software tool. The scaffolded example provided the girls the opportunity to work on testing and debugging while still producing a working personalized structure they could proudly display to their families during the final showcase presentations.

5 Discussion and Future Work

The initial study provided an opportunity for us to determine if the computing activities introducing CT concepts to middle school girls were engaging, motivating, and useful in a brief after-school format. Although we are still analyzing the data we collected, the preliminary qualitative findings reported here reveal that concepts such as problem decomposition and mapping onto computational solution steps were accessible and fun for this age group. In fact the girls were engaged enough that they would have been able and willing to consider additional complexity if we were to develop more advanced workshops (particularly with the Scratch and Legos tools).

By using various forms of scaffolding (e.g. examples, workbooks, classroom assistants, pairs of learners, etc.) we created a learning environment that engaged and motivated the girls to work with different computing tools and at times even had them strive to do more than what was expected. When the girls had extra time they prefered to turn back to the 2D more than the 3D programming environment. From our observations of the girls' overall engagment with the activities, we consider it important the learning environment provide some consistency in computing tools and will adjust the next offering of the workshops.

A second instance of the GirlsCreateIT! after-school program for middle school girls will take place in Spring 2012. The program will be offered again to girls in 6th to 8th grade, after school, one day per week. The general format for the workshops will stay the same; however the activities and measurements are being refined based on findings from the study reported here. The weekly sessions will continue to use scaffolded learning activities to introduce and build on CT concepts which require minimal instruction, and the girls will again work in pairs and use workbooks to help guide them through the self-paced activities. However, to provide a more seamless transition between activities across the workshops, the girls will use Scratch for interactive stories, games as well as a structured data activity. In the longer term we also hope to use an extended version of Scratch to control sensors and motors for the final activity. This should provide an opportunity for the girls to learn one computing environment as they are being introduced to foundations in computer programing and still build on the CT concepts throughout the workshop series.

This research contributed to the computer science education community by providing new examples and evaluations for computing activities that introduce CT concepts and skills to middle school girls. Similar activities that build on CT concepts are needed across the K-12 curricula, and the activities and lessons learned that we have described are already providing a set of new resources for teachers who might use them in their own teaching or after-school programs.

The second phase of the study will further refine the assessment materials and learning activities. This phase will also create materials that might be incorporated into different formats, for example a week-long summer camp. This will make the contribution more general as a scaffolded learning framework for introducing computing skills and concepts that increase in complexity as the learners become more comfortable with the technology.

6 Acknowledgment

We thank our workshop assistants, Dan Guzak and Neha Sharma and the local Middle School Outreach program for their support of this work. The project was partially supported by the National Science Foundation, CNS-0634337.

7 References

[1] J. Cheeseman Day, A. Janus, and J. Davis, "Computer and Internet Use in the United States: 2003 Special Studies," U.S. Census Bureau Current Population Report2005.

[2] J. M. Wing, "Computational thinking," *Communications of the ACM,* vol. 49, p. 33, 2006.

[3] A. Ruthmann, J. M. Heines, G. R. Greher, P. Laidler, and I. Charles Saulters, "Teaching computational thinking through musical live coding in scratch," presented at the Proceedings of the 41st ACM technical symposium on Computer science education, Milwaukee, Wisconsin, USA, 2010.

[4] E. A. Freudenthal, M. K. Roy, A. N. Ogrey, T. Magoc, and A. Siegel, "MPCT: media propelled computational thinking," presented at the Proceedings of the 41st ACM technical symposium on Computer science education, Milwaukee, Wisconsin, USA, 2010.

[5] CollegeBoard. (2010, Course Annotations. *AP Computer Science: Principles*. Available: http://csprinciples.org/

[6] S. Bennett, K. Maton, and L. Kervin, "The 'digital natives' debate: A critical review of the evidence," *British Journal of Educational Technology,* vol. 39, pp. 775-786, 2008.

[7] M. B. Rosson and J. M. Carroll, "Scaffolded examples for learning object-oriented design," *Commun. ACM,* vol. 39, pp. 46-47, 1996.

[8] A. Repenning, D. Webb, and A. Ioannidou, "Scalable game design and the development of a checklist for getting computational thinking into public schools," presented at the Proceedings of the 41st ACM technical symposium on Computer science education, Milwaukee, Wisconsin, USA, 2010.

[9] V. Barr and C. Stephenson, "Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community?," *ACM Inroads,* vol. 2, pp. 48-54, 2011.

[10] C. Kelleher, R. Pausch, and S. Kiesler, "Storytelling alice motivates middle school girls to learn computer programming," presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, San Jose, California, USA, 2007.

[11] J. Denner, L. Werner, S. Bean, and S. Campe, "The Girls Creating Games Program," *Frontiers: A Journal of Women Studies,* vol. 26, pp. 90-98, 03 2005.

[12] J. M. Carroll and H. van der Meij, "Ten misconceptions about minimalism," in *Minimalism beyond "The Nurnberg Funnel"*, J. M. Carroll, Ed., ed Cambridge, MA: M.I.T. Press, 1998, pp. 55-90.

[13] M. B. Rosson, A. Ioujanina, T. Paone, G. Sheasley, H. Sinha, C. Ganoe, J. M. Carroll, and J. Mahar, "A scaffolded introduction to dynamic website development for female high school students," presented at the Proceedings of the 40th ACM technical symposium on Computer science education, Chattanooga, TN, USA, 2009.

[14] J. M. Carroll, *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill*. Cambridge, MA, USA: MIT Press, 1990.

[15] J. M. Carroll, R. L. Mack, C. H. Lewis, N. L. Grischkowsky, and S. R. Robertson, "Exploring exploring a word processor," *Hum.-Comput. Interact.,* vol. 1, pp. 283-307, 1985.

[16] C. Kelleher and R. Pausch, "Using storytelling to motivate programming," *Commun. ACM,* vol. 50, pp. 58-64, 2007.

A Cross-Platform Framework for Educational Application Development on Phones, Tablets, and Tablet PCs

Wade Fagen Sam Kamin
Department of Computer Science
University of Illinois
Urbana, IL 61801 USA
{wfagen2, kamin}@illinois.edu

ABSTRACT

We describe a major extension of the SLICE framework, called SLICE 2.0. This new framework allows developers to program an application once and deploy the app natively on all major PC platforms (Windows, Mac, and Linux), tablet PCs, and Android based cell phones and tablets. The framework specifically provides functionality to create pen- and touch-enabled applications focused on active and collaborative learning. We introduce two classes of applications that are already in use in classrooms using the SLICE 2.0 framework and discuss several hardware devices running the SLICE 2.0 framework.

KEYWORDS

Software Framework, Active Learning Tool, Collaborative Learning Tool

1. INTRODUCTION

In recent years, a number of researchers have studied the effectiveness of active and collaborative learning technology in classrooms [1, 2]. These technologies often make use of laptop computers, tablet PCs, iOS or Android-based cell phones and tablets, and electronic voting systems (EVS) [3]. Many higher-education institutions have adopted various forms of EVSs, including i>Clickers, eInstruction clickers, and others [4]. Case studies and research alike have shown EVSs to be effective, though often note limited functionality as a drawback to complex active-learning exercises [3].

As a contrast to EVSs, researchers have developed and studied active learning applications on pen-enabled (tablet PCs) and touch-enabled (cell phones and tablets) devices [3]. Nearly all research we have found uses a single, custom-developed application for the task at hand running on a homogeneous platform.

Realizing that the modern classroom is filled with cell phones, tablets, laptop computers, and other intelligent devices, we have developed and successfully deployed a cross-platform, cross-device framework for developing applications primarily focused on educational use. The original framework, SLICE, or Students Learn In Collaborative Environments, was a software framework originally developed in 2006 and has been continually used in classrooms since then [5]. The applications originally developed using the SLICE 1.0 framework allowed the quick development and deployment of educational applications on Windows-based Tablet PC devices.

In this paper, we introduce SLICE 2.0, a significant milestone in the development of the SLICE framework. SLICE 2.0 builds on the SLICE framework by preserving nearly the entire existing API that developers have used to develop SLICE applications while allowing applications developed on SLICE to be deployed on heterogeneous hardware. In this paper, we will discuss some of the hardware devices on which SLICE 2.0 has been deployed and briefly introduce several of the successful applications that have leveraged the framework.

Supported Platforms
Windows XP, Vista, or 7
Windows Server 2000, 2003, or 2008
Mac OS X 10.6+
Linux supporting Java 1.5+
Android 2.2+
User Scripts
JavaScript
User Interface Design
XML

Table 1: Overview of Slice requirement

2. FRAMEWORK

The SLICE framework is based on the Model-View-Controller [7] design pattern, pushed to an extreme: The model is an XML tree that includes interface element (buttons, panels, ink strokes). The controller – that is, the code that is invoked when events like "ink stroke entered" or "button clicked" or "message received from another computer" occurs – does nothing but make changes in the model. Unlike most versions of MVC, this code does not have to indicate whether those changes in the model must trigger changes in the "view;" that is handled automatically. Thus, each event triggers the execution of a small piece of code that changes a model whose structure is quite simple. These small pieces of code are written in JavaScript; using a scripting language provides well-known advantages of convenience and short development times (for small scripts).

This structure allows for a high level of portability by separating application-specific functionality (the scripts that modify the model) from platform-specific functionality (capturing events and rendering the various components of the model). The latter is contained in the "core"

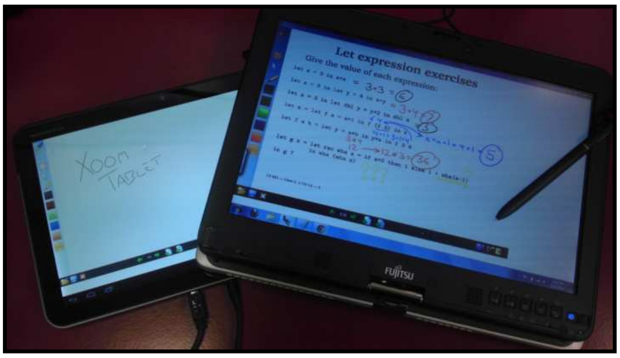


Figure 1: Two student instances of the SLICE Lecturer Application on heterogeneous platforms. Left: Android tablet; Right: Windows 7 Tablet PC.

code written once for each platform, while the former is the per-application code that runs on all platforms.

To an application programmer, SLICE provides a rich set of callbacks to run the JavaScript code the programmer designed. Additionally, SLICE provides over eighty functions available to application developers to interact with the "model" and "view". For example, the XML necessary to show a **`Button`**, and register a callback when the **`Button`** is clicked by the user, is simply the following:

```
<Button OnClick="ButtonClicked" />
```

The JavaScript function that is invoked when a button is clicked by the user could then be:

```
/* ButtonClicked(): Invoked by SLICE
   when the user clicks the button
   on the screen. */
Function ButtonClicked() {
    Sender["Text"] = "Clicked";
}
```

In the example above, the text displayed on the button that was clicked will be modified to the string "Clicked". Detailed documentation on every callback, API call, and variable is provided on the SLICE website [6].

3. APPLICATIONS

Since 2006, over a dozen known applications were developed using the SLICE framework. In the past year, we have deployed two notable classes of applications to meet specific educational objectives based on what we have learned. This paper will introduce these applications and how they are in use today.

These applications are samples of what can be developed and focus on specific learning objectives and needs of a class. We encourage the reader to use these pre-developed applications themselves or develop their own application using the SLICE framework for cross-platform deployment.

3.1. Lecturer Application

The "Lecturer Application" is a generic phrase describing the most broadly used SLICE application since 2006. As part of this application, small changes are made nearly every semester to meet instructor feedback and new educational goals. The original motivation came from instructors noting the advantages of having pen input on projected slides, as compared with using a whiteboard, transparencies, or PowerPoint presentations. With that in mind, the original Lecturer Application provided a way for PowerPoint or PDF slides to be loaded into a SLICE application on a pen-enabled Windows tablet. The instructor would connect his or her tablet up to the projection system and could write, draw, or otherwise annotate their slides with their tablet PC [7, 8].

In classrooms with a house computer, the Lecturer Application can be run in "networked mode." The display on the lecturer's tablet is wirelessly cloned on the room display, allowing the lecturer to move about the classroom, tablet PC in-hand, and even allow students to use the tablet to write answers to questions. This application proved to be a useful tool in creating an active learning environment and a more collaborative classroom.

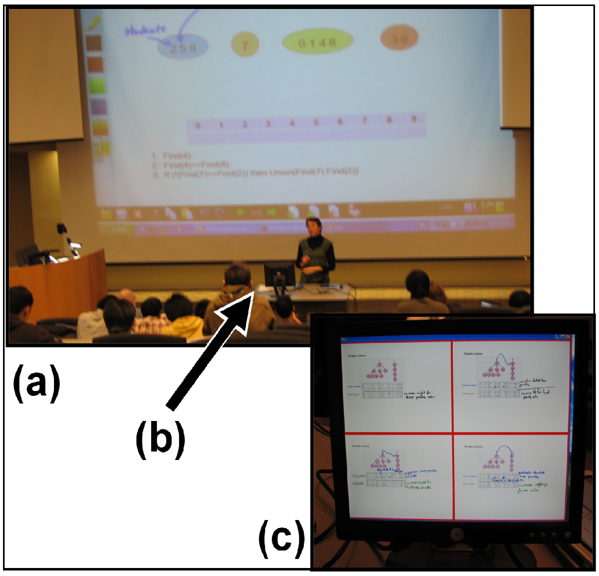


Figure 2: An iteration of the Lecturer Application in use in a classroom using a separate dashboard display. (a): An on-going lecture in a CS2-style course. (b): The dashboard display monitor. (c): The dashboard display visible to the lecturer.

With SLICE 2.0 in mind, this application was extended several more times focusing on increasing active learning. We are currently studying three specific variations on this application. All three variations focus on large classrooms where some students have a SLICE-enabled device and actively participate in the lecture by taking notes or answering problems proposed by the lecturer.

The first application allows the lecturer to *constantly* monitor a subset of students. It employs an additional display, placed in a location where the lecturer, but not the students, can see it. That display shows a dashboard of all of the current students' notes. Figure 2 shows a dashboard with four students' work displayed. The instructor can gauge student progress on a question, answer questions posed by the student through the tablet, or gauge understanding of a topic by the students' notes. This application can be helpful in an active learning environment

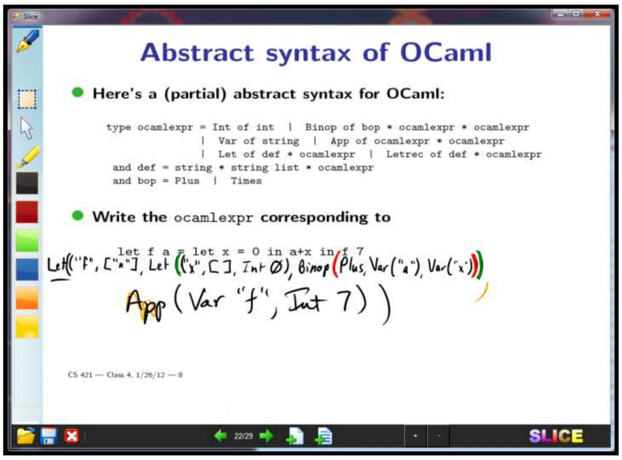


Figure 3: A student view of a Lecture Application where a student's work has been appended by the instructor.

(see Figure 2), but it is mainly intended for a more traditional classroom; the primary feedback to the lecturer is the notes that students are taking during the lecture.

Another application focuses on providing the instructor feedback *immediately after* the class, which can then be used to guide subsequent classes. Instead of a student's work being shown to the instructor, the student uses a special interface to leave questions on the slides. At the end of lecture, the instructor can review those questions.

The third application addresses classes that employ active learning, where a considerable amount of class time is devoted to student problem-solving. This is similar to the first application (the "dashboard"), but the dashboard resides on the lecturer's own tablet, and student work can be seen only by changing modes. In "monitoring mode," the lecturer can scroll through the students who are using tablets and see their progress on the current in-class exercise. The instructor may then write directly on a student's slide privately (in effect, sending a message to that student), but the more important feature is that he can show a student's slide on the classroom display. This use of a peer's answer publicly allows for the entire class to collaborate on an answer with the instructor being able to fill in errors or incomplete portions of the student's answer.

Figure 3 shows a student machine's display where the instructor added to the answer the student originally wrote. More details on this application are given in [9].

With these applications, we are exploring the possibilities for enhancing instruction, especially in large college classrooms, by using pen-enabled computers. A major barrier to deployment is the cost of these devices. SLICE 2.0 addresses this issue by allowing these apps to run on less expensive platforms (see section 4). As they are used in the classroom, we will report significant successes or failures of these applications in future work.

3.2. Code Review Application

During the past several years, an increasing number of undergraduate programs have begun to include a course designed to build a student's individual programming skills. Students in the Computer Science department at the University of Illinois are required to take such a course, called CS 242: Programming Studio, detailed in a 2007 publication [10]. The key feature of this course is the two-hour discussion section each week, where students present their programming assignment for the week. Each student presentation lasts 20-25 minutes and focuses on the program's design and source code.

To facilitate conversations on source code, each discussion section meets in a lab equipped with a large TV display. Students traditionally would connect their own laptop to the display and discuss their design and source code by pointing to the display or using their mouse to highlight specific code regions.

As part of an effort to increase peer collaboration among the six students participating in the discussion section and the discussion moderator, a SLICE application was developed to aid in code review. The setup of this application consists of each student having

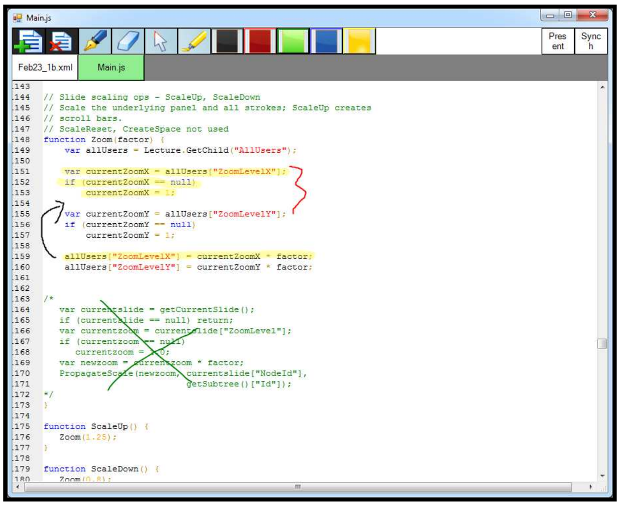


Figure 4: The Code Review application with several users highlighting and annotating the source code.

a SLICE-enabled device running the Code Review application and a server Code Review application running on the machine directly connected to the TV display. When a student is the presenter, his or her source code is displayed both on the TV display and every other student's device.

During the discussion of the code, the presenter, any student, or the moderator may write or highlight on any of the source code being presented by the presenter. These annotations are seen by every user and displayed on the monitor, making this application a sort of "shared whiteboard" with the ability to load source code (with syntax highlighting) as the ``background image'' on the whiteboard. We have preformed several detailed studies on the introduction of this application into CS 242. Early results suggest that it has made the code review discussions more efficient and more engaging. As we continue to analyze the results, we will report on the full results of these studies.

4. HARDWARE PLATFORMS

One of the primary advantages that SLICE provides that is unique in the space of educational software frameworks is the ability to code an application a single time and have it run on multiple platforms. While not all applications may be optimal for every platform (for example, it is difficult to write full sentences on cell-phone screens), the ability to give students access to the technology at no added cost to them or the university may prove key for wider adoption of sophisticated active and collaborative learning tools.

As part of the development of SLICE, we have tested the framework on a variety of devices, including:

- **Fujitsu Lifebook T730 Tablet PC**: With a Wacom active digitizer on a 12.1" display, this tablet PC is the easiest to write or annotate on and is the standard we use for comparison to other devices.
- **Motorola Xoom Tablet**: At 10.1", this Android tablet is nearly the size of many tablet PCs. At a fraction of the cost of a tablet PC, SLICE applications run as smoothly on the Xoom as the Fujitsu tablets. However, with only capacitive-touch input, some users feel drawing and annotating slides is natural but any form of writing proves difficult.
- **HTC Flyer:** Featuring a 7" display and an active digitizer, this tablet has provided the benefits of the pen-input at a substantially cheaper price than tablet PCs. While the Fujitsu tablet features a slightly smoother and more accurate active digitizer, the ability to write on the screen feels much more natural than writing on the Xoom tablet. Unlike the Fujitsu, this tablet also allows for touch-gestures, making tasks like zooming much more natural.
- **HTC EVO 3D:** With a duel-core 1.2 GHz processor, this cell phone is nearly as powerful as many of the tablets. However, the small (4.3") capacitive screen makes simple touch gestures intuitive but writing nearly impossible.

Additionally, we are particularly interested in continuing to follow the development of

Android tablets with active digitizers (devices similar to the HTC Flyer discussed above). At as little as a third the cost of a traditional tablet PC and with much of the same functionality, these devices may be the future hardware used in technology-enabled classrooms.

5. CONCLUSION

We have described the SLICE 2.0 software framework for developing cross-platform applications. The framework allows for a programmer to program an application once and deploy the application natively on a Windows, Linux, or Mac computer, a tablet PC, and an Android-powered cell phone or tablet. We have successfully deployed four applications in classrooms at The University of Illinois and SLICE is used on a daily basis in several undergraduate courses in the Computer Science department.

Building on the experience of over a dozen existing SLICE 1.0 applications, we preserved the most useful features while extending SLICE to run on a heterogeneous set of devices. Specifically, we described the classic Model-View-Controller (MVC) design pattern that is central to SLICE application development. Application developers need only to layout the desired graphical user interface components using XML and program application logic in JavaScript.

We encourage the reader to try out any of the applications described in this paper or to develop their own SLICE application. The applications described in this paper, the framework itself, and complete documentation can be found on the SLICE website at `slice.cs.illinois.edu`.

REFERENCES

[1] M. Nakakuni, M. Okumura and S. Fujimura, "Construction of a Collaborative Learning Environment through Sharing of a Single Desktop Screen," in *International Conference on Frontiers in Education: Computer Science and Computer Engineering*, 2011.

[2] C. H. Crouch and E. Mazur, "Peer instruction: Ten years of experience and results," *American Journal of Physics*, vol. 69, no. 9, pp. 970-977, 2001.

[3] K. Nahrstedt, L. Angrave, M. Caccamo and R. Campbell, "Mobile Learning Communities – Are We There Yet?," Information Trust Institute, University of Illinois at Urbana-Champaign, 2010.

[4] G. E. Kennedy and Q. I. Cutts, "Construction of a Collaborative Learning Environment through Sharing of a Single Desktop Screen," *Journal of Computer Assisted Learning*, vol. 21, no. 4, pp. 260-268, 2005.

[5] S. Kamin, M. Hines, C. Peiper and B. Capitanu, "A System for Developing Tablet PC Applications for Education," in *Proceedings of the 39th SIGCSE technical symposium on Computer science education*, 2008.

[6] SLICE Research Group, "SLICE: Students Learn In Collaborative Environments," [Online]. Available: http://slice.cs.illinois.edu/.

[7] R. Anderson, "Beyond PowerPoint: Building a New Classroom Presenter," *Syllabus Magazine*, June 2004.

[8] M. Wilkerson, W. G. Griswold and B. Simon, "Ubiquitous Presenter: Increasing Student Access and Control in a Digital Lecturing Environment," in *Proceedings of the 36th SIGCSE technical symposium on Computer science education (SIGCSE 2005)*, 2005.

[9] S. Kamin and W. Fagen, "Supporting active learning in large classrooms using pen-enabled computers," in *Proceedings of the 2012 International Conference on Frontiers in Education: Computer Science & Computer Engineering (FECS 2012)*, Las Vegas, Nevada, 2012.

[10] M. Woodley and S. N. Kamin, "Programming Studio: A Course for Improving Programming Skills in Undergraduates," in *Proceedings of the 38th SIGCSE technical symposium on Computer science education*, 2007.

[11] E. Spertus, M. L. Chang, P. Gestwicki and D. Wolber, "Novel Approaches to CS 0 with App Inventor for Android," in *Proceedings of the 41st ACM technical symposium on Computer science education* , 2010.

[12] G. E. Krasner and S. T. Pope, "A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System," in *System*, 1988.

A Gender-Inclusive Holistic Approach for IAS Education

[+]Myung Ah Park, Gang Qian
Department of Computer Science
University of Central Oklahoma
Edmond, OK
mpark5, gqian@uco.edu

Michelle Hepner
Department of Information Systems
University of Central Oklahoma
Edmond, OK
mhepner1@uco.edu

Kewei Sha
Department of Computer Science
Oklahoma City University
Oklahoma City, OK
ksha@okcu.edu

***Abstract*— The gender inequity in the IT field has been addressed with different approaches such as outreach programs for middle-school or high-school female students and gender-inclusive instructional methods to retain female students in the IT-related disciplines. However, efforts to promote future female IAS workforce have rarely been conducted although the gender inequity in the IAS field is even starker than it is in the general IT area. In this paper, we will discuss problems with existing approaches for IAS education and propose a holistic IAS educational strategy which not only addresses those problems but also may be effective in bringing up a female workforce for the IAS field.**

Keywords- information assurance and security education; curriculum; gender inequity; interdisciplinary; useful throughput

I. INTRODUCTION

Enrollment of women in computing-related disciplines is shrinking as the total enrollment has decreased. Furthermore, the recent data from the IES National Center for Education Statistics shows that the ratio of men and women receiving bachelor's degrees in those disciplines is roughly 82:19 in average, whereas this ratio was 63:37 twenty five years ago [4]. Such gender inequity is even starker when it comes to the information security workforce. According to a survey by the labor department, women currently compose only 12% of the total information security workforce [1][2]. Information Assurance and Security (IAS) is a relatively new branch of the Information Technology (IT) field, but in dire demand. In fact, recent reports suggested a shortage of approximately 20,000-30,000 qualified cyber-security specialists in the US Public Sector alone [3], even though these positions are highly compensated [5].

Several efforts have been made to recruit female students to the general IT field. These efforts focus on either the input problem or throughput problem. In order to address the input problem, outreach programs for middle/high-school female students through summer programs have been conducted. On the other hand, the throughput problem has been addressed by supporting mentors and peer group or using gender-inclusive instructional approaches. However, there have been very few reports on activities in the literature that specifically target IAS female workforce mainly because IAS is a relatively new field. In this paper, we discuss what distinguishes IAS education from the general IT education with regard to skillsets that the industry requires and what specific actions this suggests to educators to promote future female workforce in the IAS area. Finally, we discuss a gender-inclusive holistic solution that is under development in our institution. To be specific, we will discuss further detail on the input and "useful" throughput problems in the IAS field in Section 2. In Section 3, we will discuss the competency areas in the IAS field and what specific skills are required of personnel in the field. In Section 4, we will discuss what lacks in the existing approaches for IAS education. Finally, we will discuss our gender-inclusive holistic approach for the IAS education in the Section 5 along with our conclusions in Section 6.

II. GENDER INEQUITY IN IAS: INPUT AND USEFUL THROUGHPUT PROBLEM

The gender inequity in the general IT field is primarily due to fewer female students choosing computing-related majors or careers. So the problem is with input rather than throughput as discussed in [6]. Many studies have been carried out to reveal the causes of the "input" problem. Katja and Mikko laid out the categories of the reasons, which include gender discrimination in the work place, less exposure to computer usage, male-oriented leisure software, stereotyping of IT jobs, lack of role models, and low self-efficacy [7].

In order to break down some of the causes, academia has made significant efforts; summer camps have reached out to middle-school and high-school female students to help them be exposed to various areas of computing, to see the utilities of computing in real life and to create opportunities to meet IT role models [8-17]. Similar endeavors have been made to solve the "throughput" problem on the other hand. Specifically, a gender-inclusive instructional approach, mentor and peer-group support, role models, institutional support, collaborative work, and introduction to real applications have been adopted or suggested [18-23]. In essence, the characteristics of retention strategies are similar to those of recruiting.

The vast majority of these aforementioned efforts view general female students as their beneficiaries, thus having the general IT field as a target domain. However, IAS requires practitioners to have a higher skill set than other areas in IT [24]. This means that academia needs not only to pipeline throughput, but also to generate "useful throughput" in a certain branch of IT, at least IAS, that can be readily fed into industry demand. In reality, however, many primarily teaching schools like the institution where the authors work have difficulties to meet such need due to limited resources such as a lack of lab facilities, time scarcity, and financial scarcity to re-educate faculty or hire new faculty with expertise in IAS. Furthermore, not all students are interested in pursuing IAS as their careers or capable of acquiring a higher skill set. Thus it will be a reasonable approach to target a differentiated group of female undergraduate students at an upper division, further equip them with higher skills outside the classroom, and generate "*useful throughput*" for the IAS industry. On the other hand, it will be necessary to help lower-division female students be exposed to the IAS field in an early stage of their studies and to support them with mentors from the differentiated group in order to generate an input pipeline for IAS. In addition, outreach programs for middle-school or high-school female students should be enforced to promote their security awareness, thereby generating interest in a potential IAS workforce, as many existing efforts to promote IT workforce.

III. COMPETENCY AREAS AND REQUIRED SKILLS IN IAS

In order to foster a future workforce that meets the demand of the IAS industry, educators need to know what skills or competencies are required in that specific field, if it is different from the general IT area. The Department of Homeland Security – National Cyber Security Division (DHS-NCSD), in close collaboration with academia and industry, proposed an IT Security Competency and Functional Framework that links competencies and functions to IT security roles fulfilled by personnel in the public and private sectors [25]. Fourteen competency areas listed in Table 1 were identified in the proposal. Within each competency, the work functions are divided into four categories: Manage, Design, Implement, and Evaluate. Each security role has its own set of competencies and work functions. Note that different academic disciplines serve different subject competencies. For example, many Computer Science programs in the nation offer competencies in data security, digital forensics, network security/telecommunications, and system/application security, whereas Information Technology or Information Systems programs support competencies in IT Systems Operations/Maintenance, Risk Management, and Strategic Management.

Table 1 IT Security Competency Areas

1. Data Security
2. Digital Forensics
3. Enterprise Continuity
4. Incident Management
5. IT Security Training and Awareness
6. IT Systems Operations and Maintenance
7. Network Security and Telecommunications
8. Personnel Security
9. Physical/ Environmental Security
10. Procurement
11. Regulatory and Standards Compliance
12. Risk Management
13. Strategic Management
14. System and Application Security

On the other hand, it is worthwhile to notice recent research based on a survey of security IT professionals. The authors' found that, in addition to the essential body of knowledge such as programming languages, software development tools, and business application software, security personnel are required to have significantly higher skills in the following areas than the general IT employees [24], as shown below.

Table 2 List of Skills for IAS Jobs

1. IT skills and knowledge
2. Hardware equipment
3. Software tools
4. Operating systems and server software
5. Business skills and knowledge
6. Business processes
7. Strategies and planning
8. Law and legal systems and processes
9. Interpersonal relationship building/ management
10. Communication/presentation of facts, opinions, and ideas

This finding calls IAS educators not only to teach a broad range of subject competencies but also to carefully design an environment in which subjects are delivered and learned with a notion of the business of an organization. As often mentioned by the Accreditation Board for Engineering and Technology, Inc. (ABET) industry advisory board for STEM educators, it becomes more urgent and absolutely essential for college students in STEM disciplines to develop a mindset to connect scientific and technological ideas to business and marketing, turning them into commercial products in order to cope with the era of a global economy. This implies academia needs to be more vigilant in adopting new technologies into the curriculum but also to take interdisciplinary approaches when teaching discipline-specific subject competencies. The question to ask ourselves as instructors is "where are we now?"

IV. WHAT LACKS IN INFORMATION SECURITY EDUCATION?

Academia has been active and supportive to the call of the federal government to produce an information security workforce by introducing security into IT curriculum, creating security tracks, and developing security certificate programs in various disciplines [26][27][28]. As of 2011, 145 colleges or universities are designated as the National Centers of Academic Excellence in Information Assurance Education (CAE/IAE) by National Security Agency (NSA) and Department of Homeland Security [56]. Out of all, 118 CAE/IAEs; 50 CAE-research (CAE-R); and 13 and 2-year Education Program (CAE2Y). Thirty-six institutions are designated as both CAE/IAE and CAE-R. CAE/IAEs have been successful in producing a future security workforce who is exposed to the IAS field although it is debatable whether these students have the in-depth understanding of the field which enables them to practice their learning with expertise [57].

Most of CAE/IAEs offer a sequence of IAS related courses, each of which targets one or two competencies listed in Table 1. In order to intensify the concepts taught in a course, labs and case studies usually accompany lectures [27-46]. However, existing IAS education primarily concentrates on one competency area, namely network security. Other competency areas are relatively neglected. Furthermore, labs are taught in isolation, and do not provide students with a real-world environment where they can learn systems in entirety and full complexity, as pointed in [39]. The reality is few institutions can afford such a lab environment to teach system or network security to this extent. A few institutions use capstone projects that include a level of system complexity by putting isolated topics together. The common theme of these projects, however, is limited to securing systems in a configured network and sometimes includes simulating an IT department of an organization.

Another issue is that students are rarely provided with a learning stream of security topics throughout the curriculum. A learning stream, if security topics are strategically placed, can create an opportunity to integrate knowledge and skills learned separately from different courses and apply them into a system that students develop. Solving this problem is not an easy task and even requires coordination with an entire department on curriculum and the use of resources. A considerable commitment from numerous faculty members, department support, and even the coordination of multiple departments are needed to create a complete environment in which to teach information security to the full extent. Based on our survey, only one institution has attempted to embed security throughout the curriculum [46] and there have been no reports involving a capstone project or capstone course as a culmination of security education which integrates all the topics and skills into a system development project. However, the security-embedded curriculum used in [46][47] only embraces a set of software security issues.

One final issue is that the vast majority of existing lab-based approaches teach security by simulating some elements of cyberwar, which mainly appeals to a male audience. IT educators have been called to use more gender-inclusive classroom strategies and faculty or peer mentors to encourage women to enter the IT field [48-52]. In order to motivate female students to enter IT, specifically IAS field, the educators should use more than the previously suggested methods but also interdisciplinary links and real-world examples that emphasize the practical value of IAS education [53][54].

V. A PROPOSED SOLUTION

In order to address the aforementioned problems in IAS education and also generate input and useful throughput of female IAS workforce, we propose a four-fold approach as shown below in Fig. 1. Our approach consists of a holistic security-embedded curriculum, bootcamp, mentor/peer-group support, and outreach.

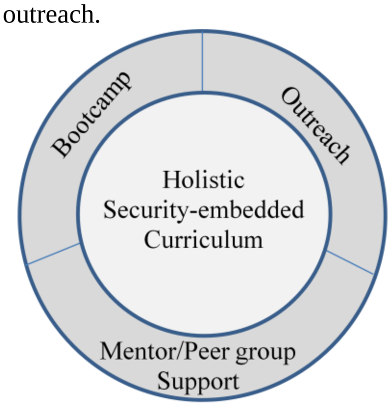


Fig. 1 A Four-fold Approach to Pipeline a Female ISA Workforce

A. Holistic Security-Embedded Curriculum

Most existing approaches for IAS education introduce security in junior or senior-level courses. Network or computer security has been most popular course to teach security. Embedding security into database or web programming courses has also been tried in several universities. More recently, a secure software engineering course is offered at the undergraduate level [29]. As mentioned in the previous section, however, these topics are taught in isolation and lack or only partially address the system-level perspective. So, existing approaches cannot effectively tackle the following issues:

- Students have already established their programming habits without security in mind when they are exposed to security concepts
- Students are not likely to be ingrained with the fact that security is an essential part of any computer/network system, either software, hardware, or both, that they will deal with in the future
- Students are not likely to have an opportunity to integrate their security knowledge learned from different courses

In order to solve these problems, we propose a holistic security-embedded curriculum where security topics are strategically placed according to carefully designed learning stream. Using this approach, students can progressively integrate their knowledge obtained from different courses and culminate their works through a capstone project. The justification of our approach is as follows.

First, developing a security track in the discipline requires resources. The reality is that many primarily teaching universities are not able to afford resources incurred in creating new courses. However, embedding security into existing curriculum does not cost as much or brings minimal cost. Several NSF grant-awarded institutes readily share their developments through their web sites. One can adopt those into curriculum [33][45].

Second, students in Computer Science develop small programs as the first step in the curriculum. Without software security, all the costs invested in building a secure corporate network with firewalls, intrusion prevention/detection system, and access control can provide no value. Starting software security education with low-level programming courses will help students not only be well-aware of software security issues but also master essential techniques at program exit. In fact, this approach will become necessary as industry demand such skill from graduates as mentioned in ACM curriculum guideline [55]. In addition, embedding security into low-level programming courses will particularly help female students be exposed to the IAS field at an early stage in the program while not giving them an impression that security is all about cyberwar. It has been suggested in the literature that when to start and how to deliver content matters a great deal when attracting female students to the IT field. One can use the excellent work by [46] to embed security into CS0-2. Readily available materials on the topics such as integer overflow, buffer overflow, input validation, and error handling are provided through their web site.

Third, in order to provide a continuity of student learning, security topics and labs should be strategically placed among relevant courses in the curriculum with well-defined program outcome specific to IAS education. Our institution is currently restructuring the following courses: lower-division programming, database, web programming, mobile application programming, computer networks, operating systems, file structures, computer security, software engineering, and capstone courses. In addition, we plan to add new courses such as fundamentals of information security and system administration and maintenance. With regard to topics to be included in the curriculum, the existing and upcoming ACM curriculum guidelines are being used. Specifically, the upcoming CS2013 draft recommends the security topics as shown in Table 3.

Since embedding security into multiple courses eases the burden of one and only one security course in the curriculum, an in-depth discussion of topics may be possible in the computer security course. Currently, we are planning to use the computer security course as an additional capstone course targeting only the IAS education outcome, in addition to the software design and development course as the capstone course for the program outcome. In order to conduct a capstone project for the IAS education, a network with reasonable complexity that includes necessary systems such as firewalls and intrusion prevention/detection system, a proxy server, and web servers is being considered. Students will further strengthen their skills with potentially interdisciplinary applications such as e-commerce application and health application running over the network. E-commerce and health web applications will run in a DMZ with backend database located either in local data center or in the cloud. The health data will be collected via wireless body area network. A set of tasks will be created with which each student group needs to demonstrate competencies relevant to the given tasks.

Table 3 Security Topics in ACM CS 2013 Draft

KA	Topics
*FC	Fundamental concepts of Information Security
*NS	Cryptographic Algorithms and their applications Network attack types Authentication protocols Digital signatures, Message Digest, Network Defense mechanism, network auditing
CG	Cipher types, Cryptographic algorithm design Encryption/Digital signatures, MAC, Hash functions Cryptanalysis, PKI
RM	Risk assessment, Business continuity planning Disaster recovery, Security auditing
SPG	Policy creation and update, Policy models Policy enforcement and compliance
SA2	Access control with least privilege principle Defense-in-depth, Human issues System threats and vulnerabilities Web application security, Penetration testing Proper usage of cryptography
SSDE	Building security into SDLC Secure Design principles and patterns Secure requirement engineering with use/misuse cases Secure coding with data validation and memory handling Secure testing with static/dynamic analysis Program verification and simulation
DF	Chain of custody, Search and Seizure Digital evidence methods and standards Data preservation/analysis/validation Legal/standard issues File system/application/network/mobile device/ computer forensics

FC: Fundamental Concepts
CG: Cryptography

NG: Network Security
RM: Risk Management
SPG: Security Policy and Governance
SSDE: Secure Software Design and Engineering
SA2: Security Architecture and System Administration
DF: Digital Forensics
*: indicates the core

B. BootCamp

A summer program or independent studies can be used to help students, particularly female students, further strengthen their IAS skills. Many different approaches are possible, depending on the choice of competency areas and instructional methods. Considering our resources and current status in developing the holistic security-embedded curriculum, we are currently planning to conduct our first interdisciplinary summer program called Female Warriors for Secure Enterprises (FWSE) this year.

The FWSE consists of two teams: core team and support team. The core team will be recruited from upper-division female students in Computer Science (CS) and Management Information System (MIS) who are highly motivated and willing to mentor lower-division students. The core team will learn the secure software development life cycle with two different points of views, one from project manager and the other from software engineer. The core members from Computer Science are required to have background in web programming and information security through independent study or regular courses offered by the department. The counterpart from Management Information System is required to have a background in project management and risk management. The support team consists of any female students in lower division courses in CS and MIS. The main responsibility of the support team is to assist the core team by playing a role of employee who is on the job training. Specifically, the support team will study recent security issues from the news or literature and give reports to the entire team. In addition, the support team will study security software tools to help the core team. Lastly, in order to reach out to female high-school students and promote their security awareness, the FWSE will give presentations to the participants of the summer bridge program in our institution.

C. Mentor/Peer Support

It is also necessary to provide mentor/peer-group support throughout the program of study to promote IAS workforce. The club similar to ACM student chapter may be an appropriate means to support the necessity. When the scarcity of female students who are interested in the IAS field is considered, an interdisciplinary club from different disciplines such as CS, MIS, nursing, and criminal justice may be a good start because such an interdisciplinary group can help students see the various utilities of the field.

D. Outreach

Outreach can be done in different ways. However, we believe that it is important to provide an environment where middle-school or high-school students can learn the field with continuity, and are not limited to one-time summer program participation. This means that we also need to put an effort on educating secondary teachers along with students.

VI. Conclusions

Although existing approaches for IAS education may be successful in generating workforce, they have the following problems. First, only one competency area namely network security has been a primary focus. So other competency areas are relatively neglected. Second, students are rarely provided with a continuity of learning of security topics throughout the curriculum because security is usually taught in an upper-level computer or network security course. Even when security is embedded into multiple courses throughout the curriculum, security topics are not strategically placed in a way that promotes competencies with a system-level perspective. Third, most of labs, especially early proposed ones, are the simulations of a form of cyberwar which mainly appeals to a male audience. In order to address these problems, we proposed a four-fold solution whose core is a holistic security-embedded curriculum. To further reach out female students and strengthen their various skills, our proposed solution includes a bootcamp, mentor/peer-group support, and outreach programs. We also discussed our current status in the implementation of the proposed solution.

References

[1] Eric Chabrow, "Women, Minorities Scarce in IT Security Field: Profession Does Not Mirror Rest of American Workforce," http://www.bankinfosecurity.com/articles.php?art_id=4143&opg=1, Oct, 2011, Extracted on Dec. 24, 2011 ,

[2] Information Security Analyst Employment Information, http://www.informationassurancefacts.com/information-security-analyst-employment-information, Extracted on Dec. 24, 2011

[3] Dale C. Row, Barry M. Lunt, and Joseph J. Ekstrom, "The Role of Cyber-Security in Information Technology Education," Annual Conference on Information Technology Education (SIGITE'11), October 20–22, 2011, West Point, New York, USA

[4] R. Guthrie, E. Yakura, and L. Soe, "How Did Mathematics and Accounting Get So Many Women Majors? What Can IT Disciplines Learn?," Proceedings of the 2011 conference on Information technology education (SIGITE '11), Oct. 2011

[5] 2008 Salary Guide, Robert Half Technology, 2008, http://www.roberthalftechnology.com

[6] Vicki R. McKinney, et al, "Women and men in the IT Profession," Communications of the ACM. Vol. 51 (2), Feb. 2008

[7] Katja Leiviskä and Mikko Siponen, "Attitudes of Sixth Form Female Students Toward the IT Field, SIGCAS Computers and Society," Vol. 40, No. 1, March 2010

[8] P. Doerschuk, J. Liu, and J. Mann, "Pilot Summer Camps in Computing for Middle School Girls: From Organization Through Assessment," ITiCSE'07

[9] Tec Camp, Georgia Tech College of Engineering, http://www.coe.gatech.edu/content/tec-camp

[10] I. Pivkina, E. Pontelli, R. Jensen, and J. Haebe, "Young women in computing: Lessons learned from an educational and outreach program," In Proceedings of the 40th ACM Technical Symposium on Computer Science Education (SIGCSE'09). ACM, New York, 509–513.

[11] M. Rosson, et al, "A Scaffolded Introduction to Dynamic Website Development for Female High School Students," In Proceedings of the 40th SIGCSE Technical Symposium on Computer Science Education (SIGCSE'09), ACM, 226–230.

[12] Gurer, D. and Camp, T. "An ACM-W Literature Review on Women in Computing," SIGCSE Bulletin, 34, 2 (Jun. 2002), 121-127.

[13] L. Pollock, K. McCoy, S. Carberry, N. Hundigopal, and X. You, "Increasing high school girls' self confidence and awareness of CS through a positive summer experience," In Proceedings of the 35th Technical Symposium on Computer Science Education (SIGCSE'04), New York, 185–189.

[14] P. Doerschuk, J. Liu, and J. Mann, "INSPIRED High School Computing Academies," ACM Transactions on Computing Education, Vol. 11, No. 2, Article 7, July 2011.

[15] First Bytes: Summer Computer Science Camp for Girls, the University of Texas at Austin, available at http://www.cs.utexas.edu/outreach/first-bytes

[16] W. Spencer, "Teaching advanced network security concepts to high school students in a laboratory-based summer camp environment," In Proceedings of the 4th Annual Conference on Information Security Curriculum Development (InfoSecCD'07), Kennesaw, GA, USA

[17] S. L. Orpheus et al, "Berkeley Foundation for Opportunities in Information Technology: A Decade of Broadening Participation," ACM Transactions on Computing Education, Vol. 11, No. 3, Article 15, October 2011

[18] Mei-Ling Liu and Lori Blanc, "On the Retention of Female Computer Science Students," Proceedings of the twenty-seventh SIGCSE technical symposium on Computer science education, Mar. 1996

[19] Amruth N. Kuma, "The Effect of Using Problem-Solving Software Tutors," Proceedings of the 39th technical symposium on Computer Science Education (SIGCSE'08), Feb. 2008

[20] Maureen Biggers, Tuba Yilmaz, and Monica Swea, "Using Collaborative, Modified Peer Led Team Learning To Improve Student Success and Retention in Intro CS," Proceedings of the 40th ACM technical symposium on Computer science education (SIGCSE '09) , Mar. 2009

[21] Brenda Wilson, "Improving Comfort Level of Females in the First Computer Programming Course: Suggestions for CS Faculty," Journal of Computing Sciences in Colleges, Vol. 23, Issue 4, Apr. 2008

[22] J. McGrath Cohoon, "Toward Improving Female Retention in the Computer Science Major," Communications of the ACM, Vol. 44, No. 5, May 2001

[23] Sarah B. Berenson, Kelli M. Slaten, Laurie Williams, and Chih-Wei Ho, "Voices of Women in a Software Engineering Course: Reflections on Collaboration," Journal on Educational Resources in Computing (JERIC) , Vol. 4, No. 1 , Mar. 2004

[24] J. K. Lee, S. Bagchi-Sen, H. R. Rao, and S. J. Upadhyaya, "Anatomy of Information Security Workforce," IEEE IT Professional, Vol. 12, No. 1, pp. 14-23, 2010

[25] Ellen Roth-Perreault, Booz Allen Hamilton, and Brenda Oldfield, "Strengthening the Security Workforce: A Competency and Functional Framework for Information Technology Security Professionals," Proceedings of the 11th Colloquium for Information Systems Security Education, Boston University, Boston, Jun. 2007

[26] Gabriele Meiselwitz, "Information Security across Disciplines," Annual Conference on Information Technology Education (ACM SIGITE'08), Oct. 2008, Cincinnati, OH, USA

[27] Stephen Cooper, et al, "An Exploration of the Current State of Information Assurance Education," SIGCSE Bulletin, Vol. 41, No.4, Dec. 2009

[28] Jeffrey Livermore, Kathryn Bake, Virginia Krolczy, and Ann Saurbier, "Capstone, thesis, or practicum? The state of the practice in IA education," In Proceedings of Information Security Curriculum Development Conference, October 7-9, 2011, Kennesaw, GA, USA

[29] James Walden, "Integrating Web Application Security into the IT Curriculum," Annual Conference on Information Technology Education (SIGITE'08), October 16–18, 2008, Cincinnati, Ohio, USA.

[30] Mike O'Leary, "A laboratory based capstone course in computer security for undergraduates," In Proceedings of the 37th ACM Technical Symposium on Computer Science Education(SIGCSE'06), Mar. 2006

[31] J. R. Aman, "A Capstone Exercise For a Cybersecurity course," Journal of Computing Sciences in Colleges, Vol. 25 , No.5, May, 2010

[32] A. Herath, et al, "Designing computer forensics courses using case studies to enhance computer security curricula," Journal of Computing Sciences in Colleges, Vol. 23, No. 1, Oct. 2007

[33] Wenliang Du, Ronghua Wang, "SEED: A Suite of Instructional Laboratories for Computer Security Education," Journal on Educational Resources in Computing (JERIC) **,** Vol. 8, No. 1, Mar. 2008

[34] Amanda M. Holland-Minkley, "Cyberattacks: a lab-based introduction to computer security," Annual Conference on Information Technology Education (SIGITE'06), Oct. 2006

[35] Luigi Catuogno and Alfredo De Santis, "An internet role-game for the laboratory of network security course,"

In Proceedings of Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE'08), Jun. 2008

[36] Susan J. Lincke, "Network Security Auditing as a Community-Based Learning Project," Proceedings of the 38th SIGCSE technical symposium on Computer science education (SIGCSE'07), Mar. 2007

[37] Rose Shumba, "Towards a more effective way of teaching a cybersecurity basics course," Working group reports from Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE'04), Dec. 2004

[38] J. R. Aman, James E. Conway, and Christopher Harr, "Capstone Exercise For a CyberSecurity Course," Journal of Computing Sciences in Colleges **,** Vol. 25, No. 5, May, 2010

[39] Steven Rigby and Amjad Ali, "Teaching Cybersecurity at the seams," 15th Colloquium for Information Systems Security Education, Fairborn, Ohio, June 13-15, 2011

[40] José Carlos Brustoloni, "Laboratory experiments for network security instruction," Journal on Educational Resources in Computing (JERIC) , Vol. 6 Issue 4, Dec. 2006

[41] Paul J. Wagner and Jason M. Wudi, "Designing and implementing a cyberwar laboratory exercise for a computer security course," In Proceedings of the 35th ACM Technical Symposium on Computer Science Education **(**SIGCSE '04), Mar. 2004

[42] Timothy Rosenberg and Lance J. Hoffman, "Taking the network on the road: Portable network solutions for computer security educators," Journal on Educational Resources in Computing (JERIC), Vol. 6, Issue 4, Dec. 2006

[43] Mark W. Bailey, Clark L. Coleman, and Jack W. Davidson, "Defense against the Dark Arts," In Proceedings of the 39th ACM Technical Symposium on Computer Science Education *(*SIGCSE'08), March 12–15, 2008, Portland, Oregon, USA.

[44] Richard Weis and Jens Mach, "Teaching Security Labs with Web Applications, Buffer Overflows and Firewall Configurations," Journal of Computing Sciences in Colleges, Volume 27 Issue 1, October 2011, pp. 163-170

[45] Lixin Tao, Li-Chiou Chen, and Chienting Lin, "Improving Web Security Education with Virtual Labs and Shared Course Modules," Proceedings of the 40th Annual Frontiers in Education (FIE) Conference, Arlington, Virginia, October 27-30, 2010

[46] Blair Taylor and Siddharth Kaza, "Security Injection: Modules to Help Students Remember, Understand, and Apply Secure Coding Techniques," Proceedings of the 16th annual joint conference on Innovation and technology in computer science education (ITiCSE'11), Jun. 2011

[47] B. Taylor, H. Hochheiser, M. O'Leary, and S. Azadegan, "Cross-site Security Integration: Preliminary Experiences across Curricula and Institutions," Proceedings of the 13th Colloquium for Information Systems Security Education," Seattle, WA 2009.

[48] Hannah M. Dee, Karen E. Petrie, Roger D. Boyle, and Reena Pau, "Why are we still here? Experiences of successful women in computing," In Proceedings of Annual Conference on Innovation and Technology in Computer Science Education **(**ITiCSE'09), July 6–9, 2009, Paris, France.

[49] J. McGrath Cohoon, "Recruiting and Retaining Women in Undergraduate Computing Majors," ACM SIGCSE Bulletin, Vol. 34, No. 2, Jun. 2002

[50] Maria Klawe, Telle Whitney, and Caroline Simar, "Women in Computing – take 2," Communications of the ACM , Vol. 52, No. 2, Feb. 2009

[51] Kathy Garvin-Doxas and Lecia J. Barker, "Communication in Computer Science Classrooms: Understanding Defensive Climates as a Means of Creating Supportive Behaviors," ACM Journal of Educational Resources in Computing, Vol. 4, No. 1, March 2004

[52] Elizabeth V. Howard, "Promoting Communication and Inclusiveness in the IT classroom," Annual Conference on Information Technology Education (SIGITE '05), Oct. 2005

[53] Martha Myers, Amy Woszxzynski, and Sherri Shade, "Opportunities for women in IT security," Proceedings of the ACM SIGMIS CPR conference on Computer personnel research, Apr. 2005

[54] Amy B. Woszczynski and Sherri Shade, "Welcome to the ISA Club! : Strategies for Recruiting and Retaining Women in ISA," In Proceedings of Information Security Curriculum Development Conference, September 25-26, 2009, Kennesaw, GA, USA

[55] CS2008, CS2013 Draft, ACM Curriculum Guidelines for Undergraduate Degree Programs in Computer Science

[56] National Centers of Academic Excellence, http://www.nsa.gov/ia/academic_outreach/nat_cae/index.shtml

[57] Matt Bishop and Carol Taylor, "A Critical Analysis of the Centers of Academia Excellence Program," Proceedings of the 13th Colloquium for Information Systems Security Education, Seattle, WA June 1 - 3, 2009

Impact of Virtualization Technology in the IT Classroom

S. Movafaghi[1], S. Hojjati[1], H. Pournaghshband[2], T. Chan[1], J. S. Collins[1]

[1]Computer Information Technology, Southern New Hampshire University, Manchester, NH, USA
[2]School of Computing and Software Information Technology, Southern Polytechnic State University, Marietta, GA, USA

Abstract - *In this paper virtualization technology, especially the virtual operating system, and its usage within an appropriate environment for the information technology (IT) classroom is discussed. We will categorize two types of environments using virtual technology, namely stable and unstable environments.*

Some universities are dismantling IT labs because of high cost to support and based on the assumption that all IT students have laptop computers. We agree with this assessment as long as the appropriate virtual lab is established for courses that require a stable environment. However, we continue to recommend that appropriate physical lab(s) be available also for courses that require an unstable environment. In this paper we will discuss the recommended number of guest operating systems for a specific number of students and resources.

Keywords: Virtualization Technology, Virtual Lab, Virtual Machine, Hypervisor, Fail Over Cluster, Load Balancing

1 Introduction

Some universities are dismantling IT labs because of high cost to support and based on the assumption that all IT students have laptop computers. We agree with this assessment as long as the appropriate virtual lab is established for courses that require a stable environment. However, we continue to recommend that appropriate physical lab(s) be available also for courses that require an unstable environment. In this paper we will discuss the recommended number of guest operating systems for a specific number of students and resources.

We argue that not only undergraduate and graduate online courses, or face-to-face information technology courses, should incorporate virtual technology concepts in each course, but also that both a stable and unstable virtual environment is essential to IT coursework. We provide examples that can be used in IT courses such as network management, database, data warehouse, e-commerce, computer languages, and/or software development (traditional or object-oriented) courses.

Three years ago we investigated various methods to obtain the hardware/software virtual environment for our students using the following requirements:

1. Hardware/software system reliability (highest priority)
2. Availability – MTBF (Mean Time Between Failure) > 6 months; MTTR (Mean Time To Repair) < 1 hour
3. Performance
4. Maintainability
5. Cluster with failover and load balancing capability. This requirement was essential for teaching purposes and to help the other requirements such as availability, performance and maintainability.
6. Cost
7. Security
8. Remote access capacity
9. Automated tools to ensure that students do not install unauthorized applications for which there is no license. Also, ensure that the student do not visit unauthorized sites.
10. Automatic login using student username and password
11. Deployment services
12. Ease of integration of other services such as documentation and media services

A virtual machine (VM) is a software implementation in a computing environment upon which an operating system (OS) or program can be installed and run. The virtual machine typically emulates a physical computing environment, but requests for CPU, memory, hard disk, network and other hardware resources are managed by a virtualization layer which translates these requests to the underlying physical hardware [1]. Different operating systems, such as Microsoft®, UNIX®, LINUX®, or Apple® operating systems can run in a virtual environment hosted on a single physical platform. These guest operating systems have most or all of the elements associated with an actual physical computer running these systems in native mode [2].

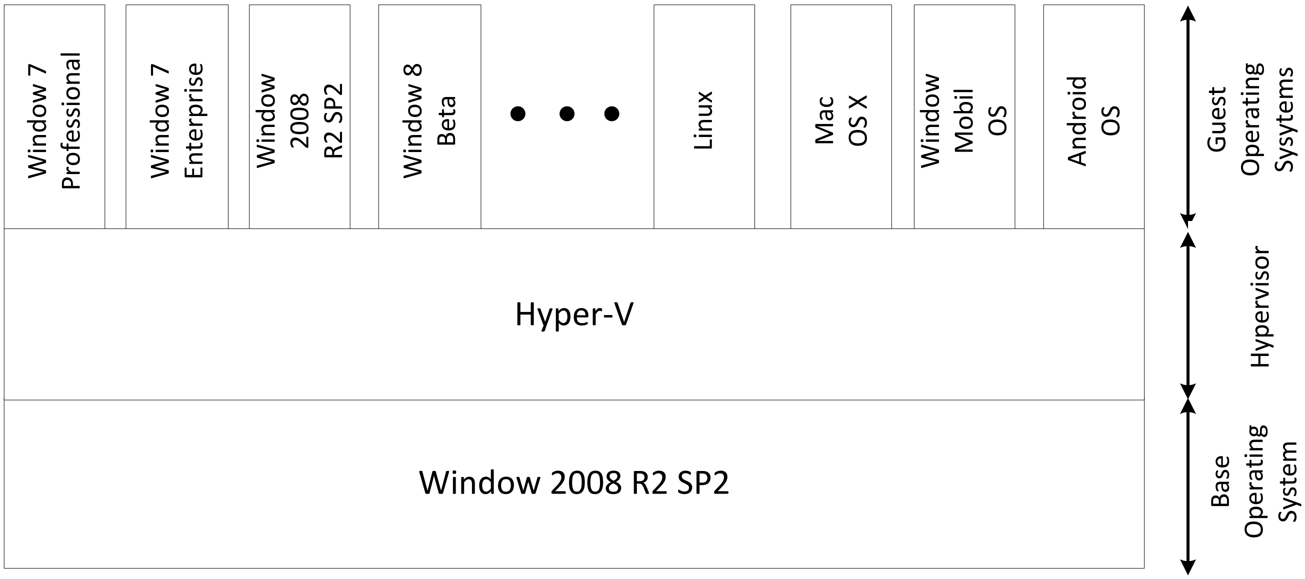


Figure 1- Base Operating System, Hypervisor and Guest Operating Systems.

We chose Hyper-V which is a hypervisor-based Windows Server virtualization platform that is included as a part of Windows Server 2008 R2 [3]. One has to first install a base operating system such as Window 2008 R2 SP2. Then, the appropriate hypervisor can be installed. In computing, a hypervisor, also called a virtual machine manager (VMM), is one of many hardware virtualization techniques allowing for multiple operating systems, termed 'guests', to run concurrently on a host computer. It is so named because it is conceptually one level higher than a supervisory program. The hypervisor is an operating system for other operating systems [4]. Finally the guest operating system can be installed. Figure 1 demonstrates a typical base operating system, hypervisor, and several guest operating systems.

The following sections describe our experience with the virtual server, fulfilling the above requirements, and our recommendations.

2 Stable/Unstable Environment Configuration

The stable environment is usually operational twenty-four by seven. The installation of new products or upgrades of existing products are fully tested in the maintenance environment before moving to the production environment. Only administrators who are fully qualified have specific administrators' privileges. The environment that we are describing is unstable, that is, we may bring down the entire system from time to time for various educational purposes, such as demonstrations of web server failover, database failover, clustering, load balancing, reconfiguring partial environment or entire environment, etc. All students have administrative privileges for their own guest operating systems in order to fulfill their assignments. Some of the courses require that the student have administrative privileges even for the base operating system. This brings new challenges that are described below.

Since all the students have administrative privileges for their own guest operating system, they could potentially abuse their own guest operating system. For example, we have seen that some students have written automated program to download financial instrument data periodically (like every few seconds) from various web sites, or have joined BOINC (Berkeley Open Infrastructure for Network Computing) [5] which degrades the overall performance of other guest operating systems. To avoid these abuses, we recommend periodic monitoring of the guest operating systems and creation of another user account with administrative privilege that the faculty member can use to login to students' guest operating systems. In this way, the faculty member can detect if students are abusing privileges in their guest operating systems and thereby degrading the overall efficiency of the entire system. The faculty member is then able to stop the infraction, and restore system efficiency.

3 Domain Controller Issues

Figure 2 shows our current two-node failover cluster. Since some of the *students* have administrative privileges with access to the base operating system for their assignments, they may accidently disrupt the main university network or other components. For this reason, the virtual server environment must be on a separate subnet from the university domain. For example, we recall one student that set up the cluster nodes instead of creating the domain CIT.SNHU.EDU in the ITSERVER, the student created an incorrect domain name, IT.SNHU.EDU, in the wrong server, namely ITSERVER01. Note that the domain must be outside the cluster nodes and then the cluster nodes can be joined to the domain. When that same student tried to delete the domain name IT.SNHU.EDU, he received a message that the system tried to delete the domain SNHU.EDU, but could not find it. Deleting the university domain name would have a devastating effect on the entire university. Therefore, for security reasons we recommend that the virtual server be in separate subnet and that the educational virtual server domain should not be a trusted domain in the primary university domain forest.

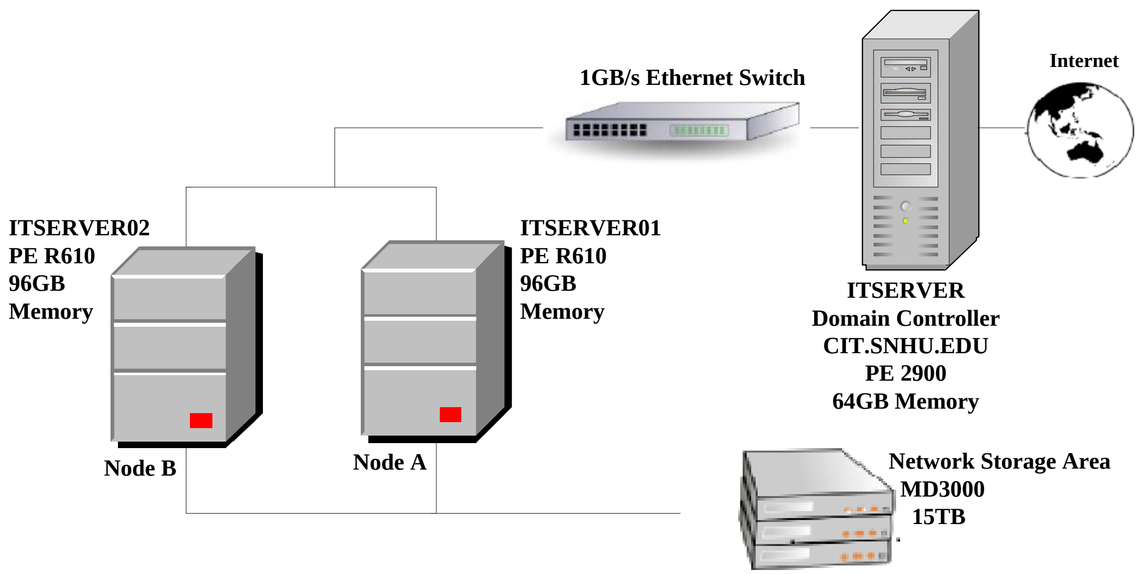


Figure 2 – Simple 2 Node Failover Cluster

4 Logs and Backup

The log file for the environment must be rigorously kept in order to be able to resolve any possible problems. Since the environment is unstable and the environment can go down for extended periods of time, for pedagogical purposes, we recommend having a robust backup plan, such as:

1. Ensure students have installed on their laptops all applications that are required by the course;
2. Provide appropriate lab (s) with adequate PCs having properly installed applications. Make sure these labs are in the same subnet as the virtual servers; and
3. Instructors should be prepared to change the assignments if the virtual server is down for an extended period of time.

5 Installation and Update Cycles and Resources

Note that at the end of each quarter for graduate students and each semester for undergraduate students, we have to delete the guest operating systems and create a new set of guest operating systems with appropriate applications. This creates an issue, because we also need many licenses for the operating systems and all applications. This issue must be resolved with each vendor prior to setting up the guest operating system.

Installation and maintenance of the unstable virtual servers is an extremely time and labor intensive effort. The responsibility for the unstable virtual servers should be assigned to only one faculty member who is teaching virtual server related courses. This opinion is based on the following reasons:

1. There is no guarantee that the servers will be up and running for other faculty members since the servers may break down when they are being reconfigured for training purposes.
2. When the server breaks, it is an excellent teaching opportunity for students to research ways to fix the server. However, this can delay repairs.
3. Generally, a university computer center is responsible for maintaining stable servers and cannot be expected to keep fixing a virtual server that is broken by

reconfiguration or by design of a faculty member to teach an IT lesson.

4. It is not cost effective to employ university computer center personnel to administrate the virtual server. For pedagogical purposes, we would like students to fix the "broken" servers for teaching purposes and would not use the services of a professional unless absolutely necessary.

We further recommend a student as a lab assistant working part time to help set up the labs. To accomplish all the requirements for a virtual server set ups may take a few days, a few weeks, a few months, or even a few years. At this time, not all of the requirements that are mentioned in the previous section have been accomplished to our satisfaction. We do recommend that you automate your processes as much as possible and encourage volunteer faculty members and students to help you. At this time we are contemplating the establishment of a Eucalyptus Systems' private cloud solution for our institution [6].

In next section we will discuss some sample assignments for the virtual servers and estimate the number of students that can be taught using the configuration shown in figure 2 and information in table 1.

6 Sample Configurations Using Virtual Technology

The least expensive cluster configuration is two nodes as shown in figure 2. In this section, we explore the number of students that can use the guest operating systems. Of course, the number of students who can use the guest operating system is dependent on the size and number of assignments given to students and memory/disk space allocation. Table 1 shows the minimum, average, and maximum memory and disk space requirements under various conditions based on the number of students and their guest operating systems to support. The configuration on figure 2 assumes 256GB memory. The first 56 GB of memory is used for the base operating systems and various servers such as the WDS (Window Deployment Services), SharePoint Server, and CMS (Customer Management System). Therefore 100 students can use the virtual server if we allocate only one guest operating system per student.

Table 1 – Minimum, Average, Maximum Memory and Disk Space Requirements

No. Of Guest OS (Virtual Servers)	Min. Memory 2 GB / Guest OS	Avg. Memory 4 GB / Guest OS	Avg. Memory 6 GB / Guest OS	Min. Disk Space 50GB/Guest OS	Avg. Disk Space 75GB/Guest OS	Max. Disk Space 100GB/Guest OS
1	2	4	6	50	75	100
10	20	40	60	500	750	1000
20	40	80	120	1000	1500	2000
30	60	120	180	1500	2250	3000
40	80	160	240	2000	3000	4000
50	100	200	300	2500	3750	5000
60	120	240	360	3000	4500	6000
70	140	280	420	3500	5250	7000
80	160	320	480	4000	6000	8000
90	180	360	540	4500	6750	9000
100	200	400	600	5000	7500	10000

On average, we allocate three guest operating systems per student (see figure 3). In this example, only 33 students have guest operating systems concurrently. For some assignments, each student may need 10 guest operating systems as shown in figure 4. This is a more sophisticated class of assignment and only 10 students can have appropriate operating systems. Sometimes we are forced to give a set of guest operating systems to a group of students, rather than individual students, due to lack of sufficient resources.

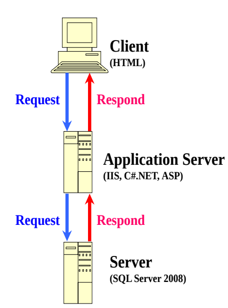


Figure 3: The n-Tier Architecture

We have noticed that students appear more satisfied and perform better as they obtain a greater number of virtual servers to use individually rather than in groups. Also, the research evidenced that the provision of low-cost and easy-to-create online resources quantitatively enhanced students' academic performance: students who accessed the online resources achieved greater academic success [7].

We have been able to teach networking using virtual server capabilities. A student may, by using virtualization, create a virtual network, completely housed on one physical server platform. The configuration is shown in figure 4. Note that all nodes on this network are on a single server device, which is supporting multiple virtual servers and multiple virtual Windows7 clients. A local area network with subnetworks was created, which contained a Domain Controller, Active Directory services, an Application Server running Microsoft SQL Server, and CMS. A student may connect through a WAN to one of the Windows7 OS instances from a thin client located anywhere.

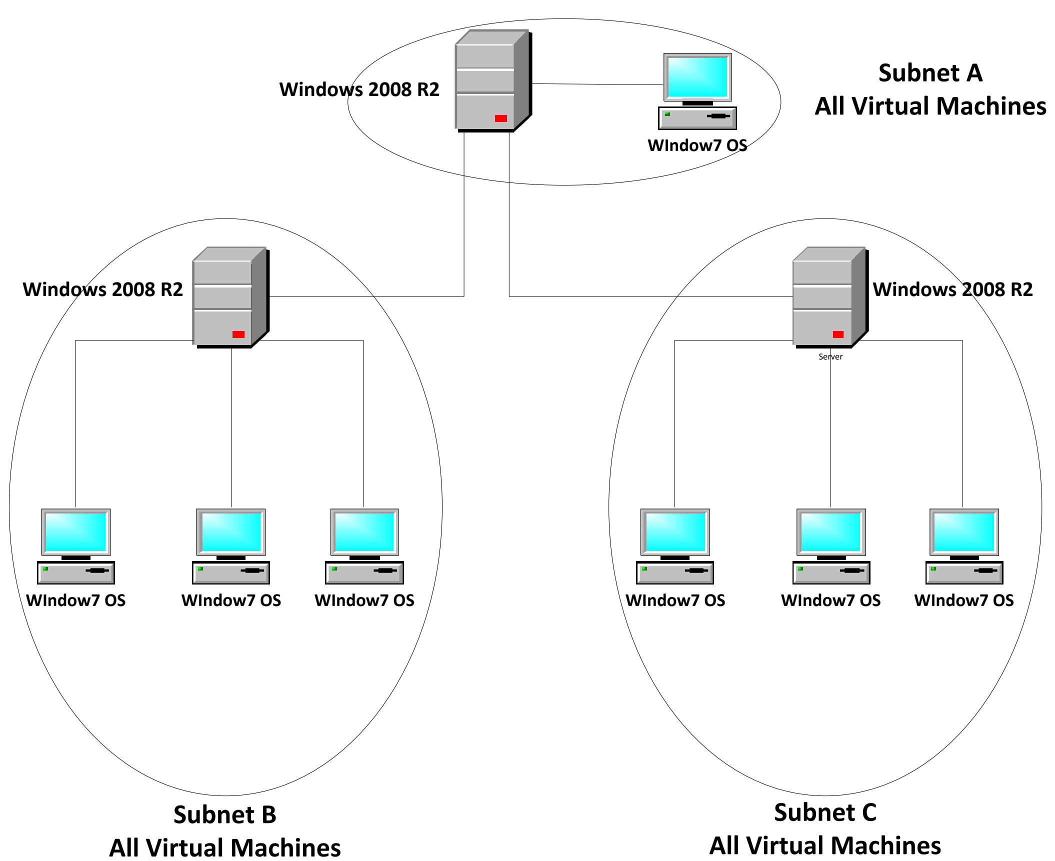


Figure 4 - Various Subnet and Several Application Using 10 Guest OS

The connection to our virtual servers is from a LAN or WAN. Note that the LAN connection has to be from CIT subnet because we choose not to take the chance of interfering with the university's production network or its servers. This is illustrated in figure 5. Students, whether using university computer lab computers or their own computers, access the virtual server through the Internet. Partly due to this, access tends to be slower than we would like. The software that students use currently is LogMeIn.Com®, and Microsoft Remote Desktop ®. In addition, Https://Join.me® will be used in future to allow instructors to remotely assist students during a session. For students, all these utilities are free.

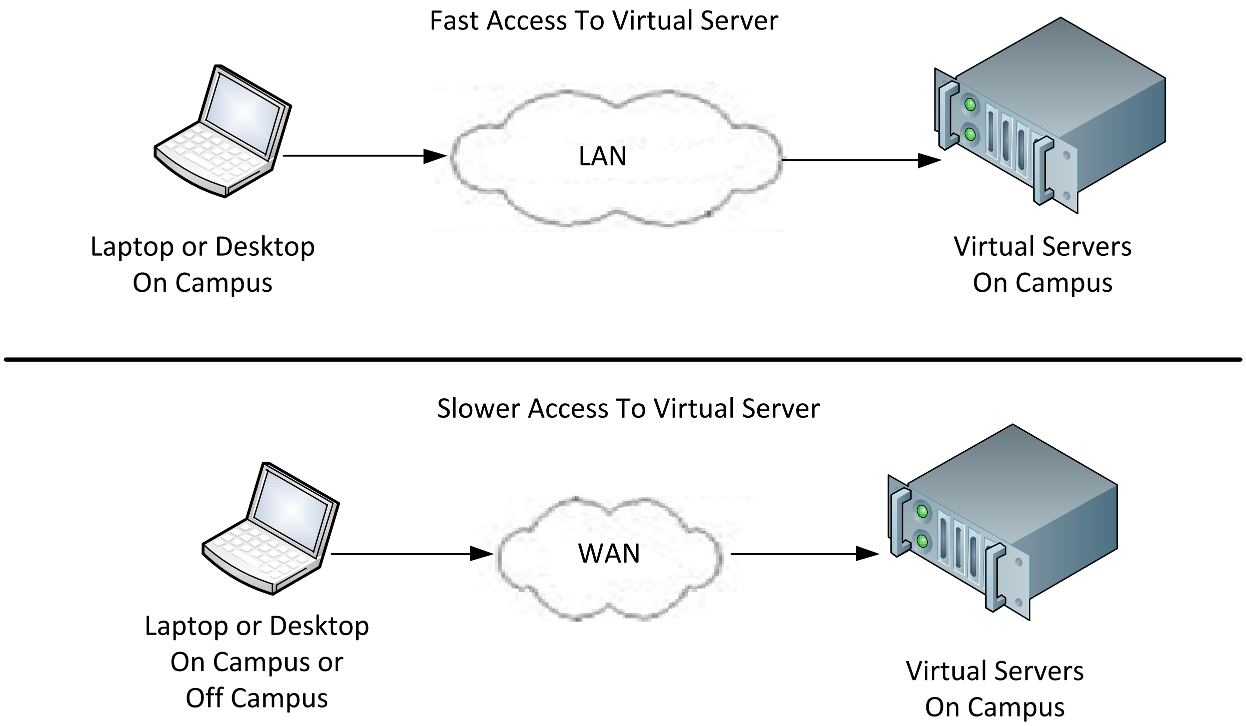


Figure 5 – LAN and WAN Access to Virtual Servers

Summary of our recommendation for unstable educational virtual servers is listed below:

1. Ensure that the virtual server is in a separate subnet from the university's production system.
2. The educational virtual server domain should not be a trusted domain in main university domain forest.
3. Ensure that the issue of the appropriate applications and extensive licenses required for the guest operating systems and applications is resolved with vendors prior to creating the guest operating systems.
4. Monitor the guest operating system for abuse by students and provide a faculty member with an administrative user account to monitor students' guest operating systems. Therefore, each guest operating system should have at least two user accounts, one for the student and another for the faculty member.
5. The log file for the environment must be rigorously kept and monitored.
6. Since the environment is unstable and the environment can go down for extended periods of time, a reliable back up plan must be in place.
7. Responsibility for the unstable virtual server should be assigned to only one faculty member who is teaching virtual sever related courses. We also recommend that a student is assigned as lab assistant on a part time basis to help set up and maintain the labs.
8. Automate processes as much as possible to alleviate some of the work of maintenance.
9. The number of guest operating system is dependent on system limitations and the assignments given. For pedagogical purposes, it is better to allocate guest operating systems to individual students and not to groups of students.

7 Conclusion

Server virtualization technology continues to be among the top technology trends. This technology should be included in all IT curriculums. In this paper, the simplest configuration for teaching virtual server technology was presented. The requirement of two node failover cluster is discussed. The virtual server environment is unstable due to the fact that this environment is for teaching rather than production.

Various issues such as reliability, performance, availability, security, cost, maintainability and supportability are discussed. We listed the summary of our recommendation for the unstable virtual server.

The number of guest operating systems that each individual student can receive is dependent on limitations of memory, disk space and assignments given to students. A sample configuration is described. The minimum, average, and maximum virtual servers to allocate to each student are one, three and ten respectively. This gives a rough idea for ROI for the teaching institution.

The applicability of using virtual server technology for teaching networking was also discussed, and an example presented.

8 References

[1] Definition of Virtual machine (VM) - http://searchservervirtualization.techtarget.com/definition/virtual-machine

[2] Chandhok, N.V.; Michael, M.L; Fries, R.M, “Using a data protection server to backup and restore data on virtual servers,” US Patent 7,899,788, 2011, Assignee: Microsoft Corporation , Redmond, WA (US).

[3] Morimoto, R.; Noel, M.; Droubi, O.; Mistry, R.; Amaris, C., “Windows Server 2008 R2 Unleashed,” 2010, Pearson Education, Inc., ISBN-13: 978-0-672-33092-6.

[4] http://www.ibm.com/developerworks/linux/library/l-hypervisor/

[5] Anderson, D.P., “BOINC: a system for public-resource computing and storage,” Space Sci. Lab., University of California , Berkeley, CA, USA, Grid Computing, 2004. Proceedings. Fifth IEEE/ACM International Workshop, 8 Nov. 2004, On page(s): 4 – 10, ISSN: 1550-5510, ISBN: 0-7695-2256-4.

[6] Shah, D; Bhingarkar, S.; Ainapure, B., "A Novel Architecture Style: Diffused Cloud for Virtual Computing Lab," International Journal of Computer Applications, Volume 39– No.6, February 2012.

[7] Cramptona, A.; Ragusab, A. T. and Cavanagh, H., “Cross-discipline investigation of the relationship between academic performance and online resource access by distance education students,” Research in Learning Technology, Vol. 20, 2012.

The authors wish to acknowledge the help and support of the following:

Daryl Dreffs, Director, SNHU Computing Services
Tom Mersereau, SNHU Computing Services
Shannon Boelter, Dell Corporation
Microsoft DreamSpark Group

The Students who helped with installation and configuration:
Lancer Zafra Uy
Benjamin Nitzburg
Justin Car

Teaching Writing in Computer Science

A. Tircuit[1]
[1]Computer Science, Seton Hill University, Greensburg, Pennsylvania, USA

Abstract – *Writing is an important skill for all graduates to master, yet it is often difficult for CS professors to embrace writing due to the inherent difficulties in teaching writing and CS faculty's lack of experience with writing based pedagogies. Teaching writing can be done well as a complement to a curriculum, providing both added value to learning and valuable skills for the future. In this case, writing was added to a Programming Languages course. As the students designed their own programming languages, they were required to provide a language specification and analysis of features. Using creative pedagogies, writing instruction was added to the course without neglecting content. Many writing techniques were also adapted to be suitable in a Computer Science course.*

Keywords: writing intensive, undergraduate CS education, programming languages

1 Introduction

I would rather teach students to write in C++, Java, or even assembly language than English. Really. Yet writing is recognized as a vital skill by both many computer science factions, and at my university. The ACM includes writing as a part of its CS curriculum guidelines [1]. More recently writing has been a subject of CS education research [2, 3, 4, 5, 6]. And as Dugan relates in his experiences involving writing in the technology workplace [4] a skillful writing ability is an asset to any technology professional. The writing intensive (WI) wave that swept through Seton Hill University (SHU) in spring of 2006 was the final motivation I needed to get past my squeamishness of teaching writing. Our Computer Science program needed a writing intensive course. I had the motivation and resources to turn my existing writing assignments into something meaningful both to fulfill this requirement and enhance the students' education. For better or for worse, I was caught up in the writing current. As you can probably imagine, the ride was both more and less difficult than I feared. The journey has been worthwhile however, as it has both improved my course and added value to the students' education.

1.1 Contents

The remainder of this paper is organized as follows :

2 Attempting to please two masters: CS meets writing intensive

Given that my Programming Languages course already featured a large writing project, it seemed like the logical place to implement the WI initiative. It also fit several of the requirements of the WI committee: it is a required class for all CS majors, is taught to sophomores or higher leveled students, at which point the students are expected to be ready for advanced writing instruction. With high hopes, I submitted my WI proposal.

Unfortunately, the WI committee did not speak "computer geek," and I fail at "WI language" more often than not. My first proposal was rejected, on the grounds that it did not meet many of the requirements for a WI course. Describing my writing project as the process in which students choose a software domain and design a programming language to best-fit that domain was less than helpful to the WI committee. I did not do such a great job of interpreting their requirements in the spirit they had written them, either. Fortunately we geeks stick together, and I enlisted the help of a friend whose specialization is in technology combined with journalism. He certainly knew the writing part, and he knew enough of my language to help me translate my proposal into something the WI committee would understand. I continue to be grateful for his help. Dugan utilized a similar tactic when designing his writing course. However he used a slightly different pedagogical model than ours at SHU by team teaching the course with a writing program specialist [4]. Our model insists that the experts in each discipline teach writing specific to that discipline.

Once I sufficiently understood the WI requirements, the second challenge was to incorporate them in to my course, without sacrificing anything I thought was critical to the curriculum from a CS prospective. This continues to be the challenge today. Programming Languages has tons of content, enough to easily fill an entire semester. At the same time, I now had to incorporate pre-writing exercises, peer review, and 20 pages of submitted and revised writing by the students. If there were one thing I did really poorly the first semester, this would be it. Three attempts, and several creative ideas later, the goals of teaching content and writing are much closer together.

2.1 The writing assignment

At the heart of my WI class is the “design your own programming language” writing assignment. Students are instructed to take the principles they learn in Programming Languages, and design a language of their own. Students pick the domain, make a plan for development, and choose features for their language. Students are required to justify their choices, either by domain suitably, programming language characteristics, technological necessity, or some combination. “Because it looks pretty” is not an acceptable reason for choosing a language feature. “Because it is the most suitable option for my domain” is.

Pre-WI, one of the mistakes I made was in not sufficiently specifying the learning objectives for the writing assignment. It was a way to evaluate the students’ analytical ability, but not much else. Without these specific guidelines, variance existed in abundance in the quality of students’ assignments, and it left me without an appropriate mechanism for evaluation. Teaching is a learning process for me as well, and this is one thing I learned to do better. Now the assignment has a defined audience: a person or organization who will fund the development of your programming language, if it should be sufficiently well designed and you have expressed your ideas clearly and with strong rationale for your language choices. Furthermore, I have a goal for the students: learn to write well enough to write a proposal (email or formal) at your future place of employment. Given these clear objectives, the students’ work has become better defined, and I have an improved assessment rubric.

One question that often occurs is concerning my decision to have the students practice writing a proposal instead of a research paper. Seton Hill University’s CS program is an undergraduate-only program. A few of our graduates aspire to go to graduate school, however most see themselves working in the CS industry. Teaching writing as they will most likely use it in their careers is both a logical choice, and an “easier sell” for the students. I am able to critique them with the question, “Would you write the argument this way if you were writing this for your boss?” providing both a framework for the writing assignment and meaning to the effort of completing the assignment. The tradeoff is that research methods are not a focus. Sources are required “in sufficient quantity to make your argument.” Were we a graduate level program, or if we attracted a different population of students, this lack would have to be addressed. The "design your own programming langauge" assignment also has the benefit of being closely tied to the content of the course. A portion of the course content is explored through writing, as suggested in [2]. This frees up time during class for writing instruction and writing exercises such as peer evaluation. This strategy has made increasing writing in a content-heavy course quite managable.

3 Writing as a process

I firmly believe that writing is a process. I teach programming using that methodology [7, 8], and I also use a process model in my WI course. Of course, I am not alone in using a process-based pedagogy [9]. And as [6] mentions, getting students to engage in the process is often difficult. My strategy is to require the students to submit a variety of prewriting exercises and drafts, enforcing at least some forethought and revision.

3.1 Prewriting

Prewriting is an essential step in the writing process. When addressing prewriting for my WI course, I returned to my roots in developing pedagogy for programming classes. [7, 8] Prewriting activities are brief, worth only a small percentage of the grade, and feedback is provided immediately via class discussion. Most often, I grade these on a participation basis. Class discussion has an additional benefit: it gives the students input from their peers as well as from me. The one exception is the final prewriting assignment in which students define the domain and main goal(s) for their language. This acts at the thesis statement. I carefully evaluate these to ensure the student has a focus before continuing his/her paper.

The domain of the language becomes the theme around which all other decisions are built. A student who wants to write a language for mobile computing must address issues such as working in a networked environment and portability of code. A student choosing the domain of video game development will likely spend time exploring helpful language elements when creating game engines and processing graphics. Another popular domain is beginning programmers. These students will explore issues with readability and writability on a higher level. Students who choose “general purpose” as their domain have the greatest difficulty in making language decisions later on. Many general-purpose languages exist, and are used quite commonly [10, 11]. It is difficult to come up with new features for something that is already well studied.

3.2 Drafts

For each draft the student is required to make a variety of decisions about his or her programming language. The first part of the assignment focuses on overall decisions: domain, criteria, important characteristics, and plans for development. As we cover programming language features in class, the students must decide which ones they want to add to their languages. Students must decide upon variable types, control structures, data structures, and advanced concepts such as object orientation. For each decision, the student must define the rationale behind it, often via analyzing the features of existing programming languages and their resulting impact. It is not sufficient to say, "I chose to include object orientation because C++ does it and I like C++." An appropriate argument would discuss the need for organizing large projects, writing reusable code, and benefits of using an object model as a design tool (modeling programming objects after real life).

I expect the first draft to have complete thoughts and paragraphs focused around each subject. Students should have already searched for any sources they need and provide proper citation of sources. In some cases this may require a student to do some revision before they submit each draft (the drafts are submitted in parts, and the four parts combine to make the final paper). It is up to each student ascertain this for his or herself.

I evaluate drafts first on the strength of the argument each student makes for his/her language choices. I will point out places where the student does this particularly well or particularly poorly. I will note anything that I find unclear or contradictory. In keeping with the practices of English composition experts, I look at grammar and sentence structure as secondary issues, except in the case where they inhibit understanding of the paper.

It is often said in hockey that a penalty is only a penalty if the referee notices. My rule of thumb when evaluating grammatical issues is the same: "it's only an error if I notice it's wrong." My grammar and proofreading skills are sufficient to catch gross mistakes. They will not catch every detail. I find this to be an acceptable compromise. I do not want to be a "grammar police" as it would distract from the primary focus of building a strong, consistent argument. At the same time, students will be expected to be able to write reasonably well in their future employment. It is in this vein that I note significant grammatical issues and require the students to fix them.

I find that it is not helpful to inform the students of my rule of thumb. Unfortunately, many seem to believe that if they do not notice a grammatical error, I will not either. This is not the case, particularly when everyday speech (at least in this area) has a noticeable lack of basic grammatical principles, such as subject-verb agreement. Instead I instruct the students to use grammar and language that they would find appropriate in a work environment. When a draft has particularly poor grammar, I refer the student to our writing center for expert assistance.

I hate page lengths. Were I given complete free rein, I would have a list of requirements, without any mention of how many words it should take a student to fulfill them. The WI committee however feels it is necessary to require 20 pages of finished work. (The 20 pages may be divided into multiple assignments.) I give the students some leeway in their final page count: 18 pages is sufficient for full credit. That is about as much as I can stretch that rule though and still stay within the WI guidelines.

In order to meet the page count, many students repeat themselves, add unnecessary quotes, and use creative formatting techniques. My blanket policy is: if your paper is not long enough, find something else meaningful to add to it. I take off points for any extraneous text or creative formatting. While I do not agree with a mandatory page-length, if we are going to have one, we are going to address it properly.

3.3 Final Submission

The final submission includes an executive summary, the paper itself, and a list of sources. As noted in [4], CS does not have a single style format for all publications. As spending time on formatting and style is not a priority for me, I give basic guidelines: 1-inch margins, 12 pt font, any recognized citations style. Most students choose MLA, with which they are familiar [12]. Should a student express interest in using a CS-specific style, I would suggest the ACM publication guidelines. When grading the final version I specifically look for corrections and improvements for any issues I noted in the drafts. This process is streamlined by our online course management system. Students upload digital versions of all assignments. When grading each draft, I make comments in the system. These comments are both immediately available to students and preserved for future referencing by me when grading their final papers.

4 Results

Implementing the writing assignment was relatively straightforward. The challenge comes in evaluating student product to make the WI elements stronger in the future.

4.1 Students write both better, and worse, than I give them credit for

One of the challenges of a small university is that many of our upper division courses, including Programming Languages, is only taught once every 2 years. Traditional 4-year students will take Programming Languages in their junior or senior year. They will have completed 2 semesters of programming and either have had data structures previously, or be taking it concurrently. Quite a few of our students instead follow what I call "the three year plan." That is, they are attempting to meet 4 years of CS requirements in 3 years. It can be done, but it often means a student takes upper-division courses without the ideal prerequisites. If I were a dragon insisted that all prerequisites be met before entering my courses, we would lose majors. Since retention is also critical, I do the best I can to work with variety of preparedness of my students. The only guarantee is that every student has successfully completed CS1, meaning they have basic programming skills.

After several years of running the WI class it has become apparent to me that the writing assignment is not challenging to the better-prepared students. It is relatively simple for someone with sufficient background knowledge to analyze the language information we discuss in class, to seek out additional information where necessary, and to form their language goals into coherent sentences. Some of these students surprise me by going above and beyond the assignment instructions. They will address higher-leveled concepts without my specific direction to do so. I am always impressed by a student who goes out of his or her way to gain worthwhile knowledge from an assignment, even if I have not done a particularly good job of requiring them to do so. Some of the "better than average" students write papers that adequately meet the goals of the assignment, and express their boredom with the topic.

The obvious answer of making the assignment overall more challenging is not necessarily the best one. While some students surprise me with the strength of their papers, I am also surprised by the inadequacy of some submissions. For example, many students stated on their initial drafts that they wanted their language to be simple and offer many advanced features. Wouldn't we all! A second irresolvable conflict is the tradeoff between efficiency and reliability. A programming language with built-in structures to ensure reliability is unlikely to be as efficient as one without them (efficiency of executable code). Yet many students make these two goals their primary concern. We discuss these issues in class, they are well defined in our textbook [14], yet they are persistent and make it into many papers.

4.2 Peer review… sort of

I have bad experiences with peer review. When the WI committee listed this as one of their criteria for a WI course, I cringed and vowed to do it as little as possible.

Writing pedagogical experts consider peer review to be essential as it teaches students to work collaboratively, learn how to give advice, and learn how to take advice. In many cases, the purpose of peer review is not the outcome, but the process. Students who are struggling might review papers that are better than their own, giving them examples of what they could accomplish. Students who are already effective writers get the chance to practice evaluating ideas of less adept writers [13].

I have been slow to adopt true peer review. Instead I have focused on peer collaboration. After each draft submission I select a number of quotes from student papers that illustrate the most frequent and most serious issues. I copy and paste the quotes into a blank document, and remove anything that could identify a particular student. In class, I will put these quotes on the projector, and we discuss them as a group. This is not a graded activity. I frame it as an opportunity to assist one another in improving their writing skills.

One example from a student paper, when discussing reserved words and keywords:

"To start out, I feel like a well-structured language needs to have a collection of words that are set aside to perform a certain action. It increases reliability also makes it more readable. In my opinion, there's really no reason to have keywords where a variable could potentially have the same name. That leaves so much room for ambiguity and ambiguity does not sit well with me. Although it might cut down on the flexibility of the language, I think it is a minor and acceptable compromise to ensure that reserved words do not get confused with variable names and such."

I often ask questions to stimulate the discussion. Questions I have asked about a paragraph such as this one:

1. What language issue is the author talking about?
2. What argument is the author attempting to make?
3. Is that argument clear from the text? How could we define it better?
4. Is the language formal enough for our purposes?

In this case, the students reported that they were unsure what the author was attempting to discuss. Upon hearing that it was concerning keywords verses reserved words in a language, they pointed out that the author did not state his argument well – the term reserved words was not used. They also expressed concern with the over-use of the word "I", and of personal opinion. Phrases such as "does not sit well with me" were noted for their lack of formality.

Another example of a quote from a student's paper:

"Primitive data types are defined as "not being defined in terms of other data types" (Sebesta, 250). XXXXX will have only three primitive data types that can be used while programming. They are going to be the types of Integer, Boolean, and Char. These types are going the help XXXXX achieve the goal of being simple to use. For the types of programs that are going to be written with XXXXX there is no need to have many primitive data types in the language. Integer, Boolean, and Char will allow the new programmer to write effective code without causing confusion on which type to use. With XXXXX there is only three data types Integer for storing values in whole number and has several sizes; and Boolean for returning true or false, and Char for stings; this will keep simplicity (Sebesta). A disadvantage of only having only these three data types is that values will not be able to have decimal places with using a float data type. The Advantage of only using these three types and not decimal is that decimal does not use memory effective and it has a restricted range (Sebesta)."

In this case students pointed out that the author's language had a higher level of formality. The author cited the text, and made a reasonable attempt at making language decisions based on his goal of making the language simple for new programmers. The students however also felt that the author's decision to eliminate floating-point variables was going to limit the language and was not well justified. In particular, I raised the question, "Do you think kids will be learning programming before they are learning math with decimal places? If there are no doubles, can a kid do something like compute sales tax on a pizza? Do you think that's a reasonable expectation for someone learning programming?" It was decided that omitting floating-point variables (and subsequently floating point math) is not necessarily a terrible decision, however it was not sufficiently justified by the author. Two issues remained that the students did not uncover. One was that the author mixed up discussion of floating point type with that of decimal type. The comment concerning inefficient storage and limited range of decimals is applicable given a decimal type stored as a string. It is not relevant to the most common floating-point type. The second involved the academic level of the paragraph. The target audience for the paper has experience with programming and programming language theory. It is not necessary that the student explain basic concepts such as the purpose of an integer. I suggested that the student would be more likely to lose the interest of his audience if he kept such basic information rather than focusing on the analysis of language decisions.

One of my goals of the peer collaboration exercise is to teach students to critically evaluate the feedback they are given. Initially, I do this. If a student says that a passage "doesn't make sense," I may respond with "How does it not make sense?" If a student says they like another students idea, I will ask, "What do they like?" By the end of the exercise, both the initial feedback and responses to it should come from the students.

4.3 Enhancing student learning

Writing is a valuable skill for all graduates. However, simply teaching writing would be a poor use of our time. A writing assignment may often be used to require the students to develop a deeper understanding of the course content by requireng higher-order academic skills such as synthesis and analysis. In doing this, we provide a higher value to student education, and implement courses on a level that is consistent with the expected abilities and the level of product required of upper division students.

Programming Languages is a content-heavy course; perhaps the most content-heavy of any I teach. It would be simple to quiz students on their ability to memorize the material. What little problem solving is inherent to the course is limited to the units on lexical analysis and parsing. This is far less critical thinking than I require in my upper divsion courses. Adding the writing component to this course in particular provides educational value in that it greatly increases the higher order thinking skills required of the students. Throughout the process of designing their own programming language, students must synthesize various concepts taught in the class. At each stage of the paper students must refine their language and increase its features. As students add additional features, they must analyze their choices on multiple levels. The first level is that of their language: the language features should unify to produce a consistent, cohesive programming language. The second level of analysis requires students to look at their language choices in relation to programming language principles at large. They must ensure they have designed a language that is capable of being implemented. Examples of existing programming languages are used to analyze the impact of feature choices. For example, a language that is intended to be general purpose may be compared to C++. A language intended for teaching purposes may be compared to Pascal, BASIC, or even the Scratch programming environment [15]. (Scratch has many similar features to a good beginner's language, despite its graphical format.) This process ensures the students' language decisions are well made, and requires using analytical skills that will be useful in the future.

Perhaps the most obvious benefit of teaching writing is that it provides an additional way to assess student learning. I discovered the benefit of this when repeated papers that incorrectly interpreted a passage from the text. The following passage refers to the AND and OR operators available in C++, and their short-circuit characteristics [14]:

In the C-based languages, the usual AND and OR operators, && and ||, respectively, are short-circuit. However, these languages also have a bitwise AND and OR operators, & and |, respectively, that can be used on Boolean-valued operands and are not short-circuit. Of course the bitwise operators are only equivalent to the usual Boolean operators if all operands are restricted to being either 0 (for false) or 1 (for true).

The students used it to claim that the && and || operators are overloaded to have multiple functions. A traditional programming languages assignment or test would never catch that error, which became obvious when evaluating writing assignments. During the following class period I followed up my comments on student papers with a discussion of that passage and the various ways it had been interpreted by the students. This not only assisted the students in learning the specific differences between the & and && operators, it also provided an immediate motivation for the students to learn how to read and understand academic materials.

5 Future Plans

In particular, addressing the boredom level of the more advanced students is of particular concern. One approach I am considering is to come up with a second level to the assignment. The first level would allow anyone to complete it and to obtain a passing grade. The second level would require more depth both in obtaining sources and building arguments. Students may be required to study the available programming languages beyond the ones we see most commonly. Students might be required read a reference written on a higher level and display their understanding by using it in their paper. Or students could be asked to provide additional information concerning their language such as BNF notation for some of the structures [16].

Another persistent issue is that many students basically recreate an existing programming language. Some students are attempting to stay in their comfort zones. Some students enjoy using a programming language they are familiar with, and do not see any need for a different one. One of my goals with the Programming Languages course is to get students out of their programming comfort zones! Designing your own language is a great way to do that. Given the product of the students so far, it appears I am going to have to find a way to make that a requirement, not just a suggestion. I am still looking for creative ways to accomplish that beyond just saying, "Choosing a language feature because it is like your favorite language is not an acceptable argument."

My understanding of peer review has grown as I have worked on the course. I went into teaching writing with the belief that peer review was a way faculty passed some of their work onto their students. I have a better understanding now that it is about the students learning and engaging in the writing process. I am currently working on the details of a true peer review assignment that I am going to add to the course.

My intention is to continue to improve both my own skills in teaching writing and the writing instruction provided to my students. While the "design your own language" writing assignment could in theory be used indefinitely, I also plan on exploring additional options for the course. One possibility is to have students research a cutting-edge programming language, and perform an analysis on the new language.

6 Current research in teaching writing in computer science

Anewalt speaks of teaching writing in CS for the first time in her paper [2]. Her experiences mirror my own in that she found herself teaching at a university that requires writing across the curriculum. She addresses a number of fears faculty face when embarking on the journey of teaching writing. She addresses the ever-present concern that adding to a course takes time away from content, the suitability of writing in various courses, the time necessary to grade writing-based assignments, and the situation that faculty of various disciplines face when they attempt to teach a subject out of their comfort zone. I found her suggestions on the design of assignments to be particularly helpful. I find her insights to be on-target with my own experiences, and her suggestions applicable to a wide variety of writing situations. The feedback she gathered from students' notes that writing assisted them in developing an understanding of the course material. The main criticism I have of her article is that it speaks of the experience of teaching writing over just one semester. She has good ideas, but they are not well tested. She also focuses primarily on the writing component of the course, as opposed to my synthesis of writing and content.

Meanwhile Dugan and Polanski explore teaching writing in CS from two perspectives, that of a CS professional and that of a writing program faculty member [4]. They chose to team-teach a software engineering course, with a strong writing component. Dugan brought his software engineering expertise and experience to the table. Polanski provided the writing components. Their research focuses on developing a writing taxonomy with three concentrations: writing for learning, writing for academic communication, and writing for industrial communication. In addition, they provide a variety of general-purpose advice when teaching writing. Perhaps the most valuable piece of their research is the large number of references provided for the various taxonomies. Perhaps a writing instructor would be familiar with sources such as these, but they are not the resources commonly circulated among Computer Science professionals.

At a recent CCSC:Northeastern conference a panel of Computer Science and English faculty met to discuss integrating writing across the Computer Science curriculum [5]. Faculty represent two universities and their writing across the curriculum initiatives. At Widener University engaged in a WI initiative much like our own at Seton Hill, requiring students to take a certain number of writing-enriched courses. Quinnipiac University initiated training for their faculty in teaching writing and collaborations with the English faculty members. The result of their program is the use of writing as a pedagogical component in all CS classes. In particular, writing is used to enhance student engagement and understanding. As writing is a valuable skill for all graduates, their model assists in achieving multiple important goals.

Kaczmarczyk reports a different approach taken at the University of Texas at Austin [3]. Their CS students take a technical writing class specifically focused on academic technical writing and industrial technical writing. A study was conducted which attempted to measure student motivation towards the course and material, mastery of academic writing skills, and self-efficacy towards academic writing. The study partially proved the goal of self-efficacy of students towards academic writing. The skill-based goals were more strongly supported. Students reported an increase in confidence in expressing themselves through writing and in mastery of the material. Students did not however report an increased motivation towards the class and the material, an additional goal of the course. The discussion of their survey results makes a questionable attempt at understanding the meanings behind the results of the study. Better evidence of the usefulness of the study may be found in their discussion on the changes that were made to the course. Results from the course after these changes would be interesting.

Garvey also discusses teaching Programming Languages as a writing intensive course [6]. Instead of focusing on one large project, Garvey requires a variety of assignments in his course. The primary focus of his paper is his argument that writing can be beneficial in many forms, such as summarizing material, documenting and peer reviews of code, reviews of the writing of others, speculative writing, arguing for a particular perspective, scientific research papers, interdisciplinary writing, and team-based writing. He also addresses the issues of requiring students to engage in writing as a process, and faculty evaluation of student writing.

7 Conclusion

Having taught Programming Languages as a WI course for three years, I can say I have definitely faced a learning curve. Initially my writing assignments did not effectively express my vision of the assignment to the students. I clumsily led writing discussions. And evaluating student product required a significant amount of time. As my skills have grown, I have become both more effective and more efficient. I have reached my goals of adding a significant writing component to my Programming Languages course, and enhancing the educational value of the course. In addition my comfort level with teaching writing has increased; I no longer dread every other fall semester.

Like many tasks, the first 80% took me 20% of the time involved, and the last 20% will take 80% of the time. I continue to work to fine-tune my WI pedagogy to increase the critical thinking required of the students, to bridge the gap between the background level of the students, and to keep the writing assignment creative and interesting.

7.1 A final thought

Every year I learn something new. The past semester, I attempted to instruct the students to use the voice of authority in their writing by saying, “You are the master of your language.” The students apparently translated this statement into something like, “That means I can use ‘I think’ in every other sentence.” Not my intended result! Eventually I might figure out all of the little pitfalls. That is a big might. In the mean time, I will have an interesting time learning.

8 Acknowledgements

Special thanks to Dr. Dennis Jerz, Associate Professor of English and New Media Journalism at Seton Hill University and the students in CS250, Programming Languages, in Fall of 2011.

9 References

[1] Joint Task Force on Computing Curricula, IEEE Computer Society, Association for Computing Machinery. “Computing Curricula 2001 Computer Science Final Report”. http://www.acm.org/education/education/education/curric_vols/cc2001.pdf. December 15, 2001.

[2] K. Anewalt. “Experiences Teaching Writing in a Computer Science Course for the First Time”; Journal of Computing Sciences in Colleges, Consortium for Computing Sciences in Colleges. Volume 18. Issue 2. 346-354. December 2002.

[3] L. Kaczmarczyk. “A Technical Writing Class for Computer Science Majors: Measuring Student Perceptions of Learning”. SIGCSE’03: Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education. 341-345. February 2003.

[4] R. Dugan and V. Polanski. “Writing for Computer Science: A Taxonomy of Writing Tasks and General Advice”. Journal of Computing Sciences in Colleges, Consortium for Computing Sciences in Colleges. Volume 21. Issue 6. 191-203. June 2006.

[5] Y. Kortsarts, T. Dansdill, M. Hoffman, A. Fischbach and J. Utell. “Writing Intensive and Writing Extensive: A Continuum for Advancing Writing in Computer Science Education”. Panel discussion summary. Journal of Computing Sciences in Colleges, Consortium for Computing Sciences in Colleges. Volume 25. Issue 6. 205-209. June 2010.

[6] A. Garvey. “Writing in an Upper-Level CS Course”. SIGCSE’10: Proceedings of the 41st ACM Technical Symposium on Computer Science Education”. 209-213. March 2010.

[7] A. Tircuit. “Teaching Programming as a Process”. FECS’08: Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering. 180-186. July 2008.

[8] A. Tircuit. “Programming as a Process: Results and Phase 2”. Poster presentation. SIGCSE’10: Proceedings of the 41st ACM Technical Symposium on Computer Science Education”. March 2010.

[9] Institute for Writing and Rhetoric. “Teaching the Writing Process”. http://www.dartmouth.edu/~writing/materials/faculty/pedagogies/process.html. [May 2012]

[10] B. Stroustrup. “The C++ Programming Languages: Special Edition”. 3rd Edition. Addison-Wesley Professional. February 2000.

[11] K. Arnold, J. Gosling and D. Holmes. “The Java™ Programming Langauge”. 4th Edition. Prentice Hall. August 2005.

[12] Modern Language Association. “MLA Handbook for Writers of Research Papers”. 7th Edition. Modern Language Association of America. March 2009.

[13] J. Ruszkeiwicz, D. Seward, C. Friend and M. Hairston. “The Scott, Foresman Writer” 5th Edition. Pearson. 2011.

[14] R. Sebesta. “Concepts of Programming Languages”. 9th Edition. Addison-Wesley. April 2009.

[15] Lifelong Kindergarten Group. “Scratch”. MIT Media Lab. http://scratch.mit.edu/. [March 2012]

[16] N. Chomsky. “Three Models for the Description of Language”. IRE Transactions on Information Theory. Volume 2. Issue 2. 113-124. 1956.

Teaching Finite Elements to Structural Engineers

M. Meštrović[1]
[1]Faculty of Civil Engineering, University of Zagreb, Zagreb, Croatia

Abstract— *Teaching finite elements to structural engineers is divided in few connected courses. On the undergraduated level, they get basic information about finite elements in Matrix Structural Analysis. The sence of the next course, named Numerical Modelling of Structure, is: How to use finite elements. Software packages are used to show how to calculate different types of structures. On the graduated level, they have first a course of type: How to make finite elements. This course gives an introduction to variational formulation of standard structural analysis problems. The course is generally focused on derivation of stiffness matrix for different structural analysis problems. Further, they have courses like Theory of Elasticity, Theory of Stability, Plate Structures, Dynamics of Structures and Earthquake Engineering, where can use knowledge from former courses for modelling a real behaviour of the structures and interpretation of numerical results calculated by standard structural analysis software packages.*

Keywords: finite elements, education, variational formulation, interpretation of calculated results

1. Introduction

After graduation, the most of students get opportunity to work on the construction site or as an assistant of senior engineers in design engineering office. Those, who are going to work in design engineering office, must be capable to prepair numerical models of different structures for further numerical calculation with commercial software packages under the leadership of senior engineers. The commercial software packages, based on finite elements, are standard and helpful tool in modern design engineering offices.

The finite element method is very important, powerful, most involved and widely presented numerical method in modern structural engineering. Almost all commercial software packages for structural analysis are based on the finite element method. FEM based software packages are, for commercial reasons, highly developed and very easy to use (user-friendly). That easy usage of all commercial packages gives to structural engineers (designers) impression that they can calculate more complicated structures without a proper knowledge of theoretical background about basic structural analysis and the concept of the finite element method. That extensive and uncritical application of the finite element method on real structural analysis problems, using commercial software packages, leads to the significantly risk of provided structural analysis. Most frequently, it leads on more and more numerical models, that are not very close to real behaviour of structures. Another problem is also a wrong interpretation of calculated numerical results for practical structural analysis problems. The understanding of the finite element method is then necessary for users (engineers) of structural analysis software packages. It means that proper finite element lectures must be included in structural engineering education program.

The given courses are conceived to introduce the finite element method, to show the concept and basis of the finite element based software in structural engineering, how to make numerical models for computational analysis, how to handle with numerical results calculated by software packages and how to understand and use evaluated results.

So conceived finite element method lectures are then divided in few steps, few connected courses. In the undergraduated courses students get first basic theoretical background in mechanics and matrix structural analysis. In the next step they get basic knowledge for using software packages for linear problems (beams and frames). The first graduated course gives developing of finite elements for different structural analysis problems. In the lectures is also presented an introduction to variational formulation of standard structural analysis problems. The course is generally focused on derivation of stiffness matrix for basic structural analysis (beam bending and plate bending) problems. Further courses are based to teach students to use finite elements for different serious problems (stability, dynamics, nonlinear problems, plates, shells). The main topic of this courses is the modelling of real behaviour of structure and the interpretation of numerical results calculated by standard structural analysis software packages.

2. Undergraduated courses

Basic theoretical background is given through Mechanics 1 (Statics) in second semester, Mechanics 2 (Kinematics and Dynamics) in third semester, Strength of Materials 1 & 2 in second year and Theory of Structures, where Matrix Structural Analysis is included, in fourth semester. In matrix Structural Analysis is finite element method mentioned for the first time. Basic functions are given for beam element and demand for displacement and deflection field continuity is discussed through experience with , in formerly courses introduced, displacement method. In fifth semester, the new course named Numerical Modelling of Structures, [1], is presented. The main topic of this course is the modelling of real behaviour of beam and frame structures and the interpretation of numerical results calculated by commercial structural analysis software packages. Theoretical part of this course is focussed on foundation of continuum mechanics, definition of strong and weak formulation with its solutions and historic development of numerical methods in structural analysis problems (from Ritz method through finite difference method to finite element method). Exercise part of this course is organized in numerical lab where each student sits alone by PC and has to do all given examples by himself. Some simple structural analysis examples are used as introduction into modelling of more complicated structures. The foundation for using software packages is also outlined. As final exercises they have to make complete numerical model for some simple building (for example one-family house). The most students bring arhitectural design project from students at Faculty of Arhitecture. The goal of this project is their first contact with arhitectural idea and how to make from this idea a reasonable building structure. Each student have to put the given building in area where he is coming from and find in the codes all loads for his area (wind, snow, earthquake). The students are also encouraged to take this subject as final undergradution work on the end of sixth semester. In that final work, they have to make full designer project for given building of their own choice, from definition of construction and loadings, through numerical model of construction, calculation of internal forces and reactions and designig the structure to reinforcement drawings for typical construction elements. Special accent is given to motivate the students for using this way for structural analysis in applied courses of fifth and sixth semester (Concrete Structures, Steel Structures, Timber Structures).

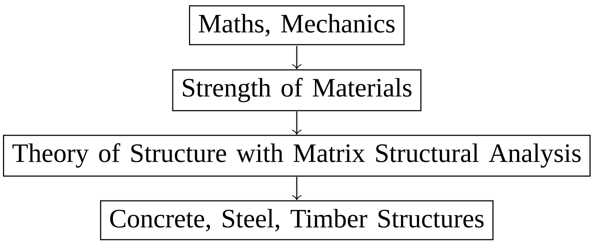


Fig. 1: Undergraduated courses.

3. Basic graduated course

The basic graduate course named Finite Element Method, [2], is given already in the first semester of the graduate study. The main topic of this course is the introduction to finite element analysis in the context of application in structural analysis. This course gives basic equations from theory of elasticity, an introduction to variational formulation of different structural analysis problems. Basic equations for beam bending, in-plane and plate bending problems are developed behalf on theory of elasticity. Variational formulation is developed and expressed for axial loaded bar, beam bending and torsion of prismatic bar in one-dimensional case and for plane problems and plate bending problems as two-dimensional cases. According the variational formulation, the finite element formulation is described and given for formerly described basic structural analysis problems.

Element stiffness matrix is introduced and calculated in detail for truss and beam element. The finite element formulation is completely given for simple and cantilever beams. The obtaining the shape functions and the strain matrix is described. After the definition of matrix of transformation, the finite element method is applied to frame structures. The construction of global stiffness matrix, by assembling element stiffness matrices, is explained on simple frame structure. The accent is given on similarity with displacement method.

In two-dimensional case complete finite element formulation is given for plate bending problems. The needed matrices for rectangular plate elements with 16 degrees of freedom are derived in detail. The element matrices are also derived for plane problem. The finite element formulation for stationary heat problem is given in one-dimensional and two-dimensional case. The application of FEM to dynamics, stability is explained on the problems of eigenvalues in dynamics and critical force in stability. The application of FEM to fluid mechanics and 3D problems is described just briefly.

The focus with excercises is given on matrix linear algebra (solution of system of linear equations) and numerical integration in the first part of the course and on complete numerical calculation using finite element formulation of some examples in the further part of this first course. The Gaussian numerical integration in one-dimensional and two-dimensional case is derived. Calculated examples must always be compared with analytical solutions. Gauss-elimination, iterative and banded solvers for solving the system of linear equations are described. The students have to develope their own algorithm for some solver. The topic in this part is how to sign the knots for the smallest brightness of band. Some algorithms for knots numeration are also presented. Numerical examples, completely calculated, are beam bending problems (cantilever or simple beam under uniform or concentrated load), frame structures (simple frame under gravitational and seismic load), plane problems (wall structures) and plate bending problems (rectangular plate with simply supported, clamped and free edges). Finite element procedures for further implementation are given for each solved problem. The general idea is to explain finite element procedures as early as possible because computational procedures are useful for understanding many of theoretical based concepts.

The goal of this course is to explain the concept of finite element method applied on structural engineering, to accent the importance of proper knowledge about basic structural analysis and theoretical background of finite element formulation for future usage of finite element based software packages.

Elasticity, Variational Formulation

↓

Beams, Trusses, Frames

↓

Wall Structures, Plates

↓

Dynamics, Stability

Fig. 2: The concept of FE course.

4. Further graduated courses

The concept of teaching finite elements is, further in graduate study, given through different courses and applied on real problems. The courses are Non-linear Structural Analysis, Dynamics of Structures, Theory of Elasticity in first year, and Plate Structures, Fracture Mechanics, Theory of Stability, Stochastic Finite Element Analysis in the second year. The goal of this part of finite element courses is to educate the students to use properly commercial software packages in their further education and engineering practice. They are going to comprehend that software packages could calculate whatever has been given as input, but problem is that software is forced to calculate just what is contained in input model and nothing more. Software packages are not going to calculate what they need to calculate to get proper numerical results if it would not be described in input datas. Very important is how close is the input model to real behaviour of the subjected structure. For that purpose, numerical model has to pursue, as good as possible, approximation of the structure. Some ideas to overcome usual modelling difficulties in engineering practice are also presented. As final work, students are asked to formulate problems or model of the structures of their own field of interest. That final work has to be done as standard calculated structure made from public structural engineers.

The central course is Plate Structures. The main topic of this course is the modelling of real behaviour of structures and the interpretation of numerical results calculated by commercial structural analysis software packages. As in undergraduated course, some simple structural analysis examples are again used as introduction into modelling of more complicated structure. First, it has been shown the influence of the choice of different finite elements for same structural analysis problem. Same problems are calculated with different finite elements (different type, not just different degrees of freedom or different mesh sizes). The foundation for using software packages is also outlined. Different numerical models are presented and explained to express the importance of a good modelling of structure. Wrong chosen numerical model leads to numerical solutions which do not represent the real behaviour of structure. Often problems from modelling in engineering practice are discussed using commercial software packages. Wrong models are used to show the evaluation of unusable results in comparison with real behaviour and according known analytical solutions. Special accent is given to motivate the students for increasing of their interest and active participation in discussion about wrong models of examples with obvious analytical solutions.

In other courses, students have to do all exercises with FE-solver and compare calculated solution with analytical solution or with numerical solution calculated through given procedure. Students are encouraged to find their own examples and discuss with lecturer and other students during semester.

Finite Element Method

↓

Dynamics, Elasticity, Non-Linear Anlaysis

↓

Plates, Fracture, Stability, Stochastics

Fig. 3: Graduated courses.

The students are also encouraged to take final master degree work from courses in this area. For example, they can design the structure over large span across sports field or sports hall, design the structure of skyscraper in in earthquake area (a_g for most location in our country is $0.2g$ or $03.3g$). They have to define their own structure for given geometry, define loadings, construct the numerical model, design complete structure and draw reinforcement drawings and construction details for project realization.

definition of structure and loadings

↓

numerical model

↓

designing of structure

↓

reinforcment drawings, details

Fig. 4: Concept of master degree work.

5. Timing for finite element courses

The recommended timing for first introduction of finite element method is second year of undergraduate education (sophomore level), just after basic courses in mathematics, mechanics and during theory of structures. The students can apply this new knowledge in further years of engineering education in different practice-oriented courses (concrete structures, steel structures, timnber structures bridges,...) where the given finite element courses could be reinforced through the applications in different structural engineering areas.

When this finite element courses are required in later years of education, the students do not have enough time to find the relevance of this courses to their further engineering practice. The interest of students increased with more presented simulations and examples from engineering practice.

More application-oriented second course is much better received by graduated students and resulted with better motivation and attention on the lectures. The possibility to choose the structure of their own interest from different structural engineering areas for their final work for master degree was found very attractive and motivating for under-graduate students. This timing for finite element courses has caused by students better interest also for the other courses in their further education where they could apply this new knowledge about the modelling of structures with finite elements. Students knowlegde about finite elements resulted with better quality of their work (better theory understanding and better quality of their exercises) in further courses (dynamics, stability, fracture mechanics, elasticity).

6. Conclusion

The concept of teaching finite elements to civil engineering students is exposed. The lectures are divided in few connected courses. The undergraduated courses are focused on the basic knowledge of finite elements, using and understanding the finite element based software packages in the context of their application in structural analysis and educating the undergraduate students to use properly commercial software packages in their future work during education and following engineering practice. More application-oriented lectures are better received by undergraduate students. More discussion about real problems from engineering practice motivate students to active participation during the courses. Possibility to choose the structure of their own interest from different structural engineering areas for final course work caused very good response by students.

Teaching students on graduated level is focused first on concept of finite element packages. Geeneral concept of finite elemenet method is described in detail. It has been lectured how to make their own procedures with different finite elements for different kind of structural problems. In further courses they have to apply former knowledge on some difficult real problems (stability, dynamics, elasticity, fracture). After this courses we get very good response from students and their understanding of serious structural problems. Students interest for master thesis in this area is very large. It results with master thesis of high quality.

References

[1] M. Meštrović. (2007) “Numerical Modelling of Structures," *Lecture Notes in Croatian*, [Online]. Available: http://www.grad.hr/nastava/nmk

[2] M. Meštrović. (2009) “Finite Element Method," *Lecture Notes in Croatian*, [Online]. Available: http://www.grad.hr/nastava/mke

Computer-Added Extraction of Chinese New Vocabulary for Language Teaching

Yuen-Hsien Tseng and Zih-Ping Ho
Information Technology Center, National Taiwan Normal University
No.162 , Sec. 1, Heping East Road, Taipei City, Republic of China

Abstract - *In teaching Chinese, there is a need to collect emerging vocabularies popular in the local Chinese society for updating the teaching materials; this need is particularly a true issue when the teachers are non domestic persons of the society where the Chinese language is in daily use. Non-domestic teachers have less contact with the everyday language used in the domestic society and they would need such information about new terms. In this paper, we propose an approach to extract the candidate new terms from a large and long-lasting stream of news articles. Based on some text mining techniques such as key term extraction, term association analysis, and hot topic detection, we are able to extract candidate vocabularies in support of Chinese language teaching.*

Keywords: Computer-Added Education, Chinese New Vocabulary Detection, Information Extraction, Language Teaching, Terms Mining.

1 Introduction

In teaching Chinese, there is a need to collect emerging vocabularies popular in the local Chinese society for updating the teaching materials or for timely use in classroom. This need seems not to be an issue at first sight since Chinese teachers are assumed to know which terms are fashionable in Chinese society's daily life. Nevertheless, even the knowledge of the Chinese teachers needs to be updated from time to time, just as elder citizens may occasionally find new terms or new usage of terms from their youngsters, such as “火星文” (“Mars language” translated literally). This new term does not really mean a signed language or a set of unusual sequences of characters such that a normal terrestrial would not understand. It is used as metaphor to denote a set of odd vocabulary commonly seen in the Internet chat rooms, discussion lists, messengers, or blogs used by youngsters. Terms like this have become more and more common in the local society and those who do not know its metaphor may not well integrate into the discourse of the local speakers. The need for obtaining new terms for Chinese teaching or learning is particularly a true issue when the teachers are outsiders of the society where the Chinese language is in daily use. Non-domestic teachers have less contact with the everyday language used in the domestic society and they would need such information about new terms. Even if a Chinese teacher has knowledge of them, an evidence-based statistics would help to include those new terms in the updated teaching materials.

To collect those terms to narrow the gap between the language been currently taught and the language been daily used, a technology-enabled mechanism to digest everyday language usage and to report on possible new terms would be of great help. One possible source for the everyday language usage is from news. Daily news reports on various topics concerning the life of a society. Therefore news articles represent a feasible source for finding the candidate terms. However, simple key term extraction is not satisfiable as technology buzzwords or name entities of particular events may often occur. Therefore, it is not trivial to extract those new terms that are suitable for Chinese teaching as a second language. The computer-added preparation of Chinese new vocabulary has thus become a topic worth of study.

In this paper, we propose an approach to sieve out the candidate new terms from a large and long-lasting stream of news. This approach is based on the previously developed techniques for keyword extraction, term association analysis, and hot topic detection [1, 2]. We apply them in a novel way to hopefully meet the need as mentioned above.

2 Literature review

There were few literatures directly related to computer-added selection of Chinese new vocabulary for textbook compilation. We briefly review some of related studies. The previous literature would be discussed in (1) IT for Chinese vocabulary learning; (2) techniques for detecting new Chinese vocabulary.

2.1 IT for Chinese vocabulary learning

Although computer added education were discussed for several decades, there were still few papers about Chinese vocabulary compilation. Blake [3] described a way of distance learning for second language (SL) education and showed that it was effective because SL students could involve in multi-culture environments. Levy [4] stated SL learning should emphasize on grammar, vocabulary, reading, writing, pronunciation, listening, speaking, and culture and IT could play more and more important roles on these topics. In Levy's study, vocabulary learning software made hyperlinking keywords simple and that would help students recall what the words really meanings.

2.2 Techniques for detecting new Chinese vocabulary

New vocabulary detecting is a pretty new topic in natural language processing (NLP). Until now, new word detecting in computational linguistics pay more attentions to new word collection in an updated dictionary, e.g. Ha et al. [5], Li et al. [6] and Jiang et al. [7].

Kuba et al. [8] stated that they selected and re-evaluated on a manually annotated corpus containing 1.2 million words, and six topics: fiction, school compositions, newspaper articles, computer-related texts, law and short business news. The methods were most common POS tagging approaches, building a tagger based on Hidden Markov Models (HMM). After they testified several different POS tagging methods, they found large corpora would improve the tagging quality; and also found combined methods outperformed using a single POS tagger. Ha et al. [5] presented a SVM (support vector machine) based method that predicted the unknown words, established a two-phase unknown word prediction procedures. The experiment results showed that the experiments were very promising by showing high precision and high recall while high speed. Li et al. [6] defined NWI as a binary classification problem. A SVM trained data were to identify the new words. Pakhomov et al. [9] trained corpus of clinical notes to perform manual annotation. The corpora were used for training POS taggers. If doctors had these POS taggers, the new terms and correct POS taggers of clinic notes would improve the clinic writing.

Brown et al. [10] gave a new idea for automatic measurement the density of POS tagging. In their papers, The Computerized Propositional Idea Density Rater (CPIDER, pronounced “spider”) was proposed. It was a computer program that determined the propositional idea density (P-density) of an English text automatically on the basis of POS tags. The main idea was propositions corresponding roughly to verbs, adjectives, adverbs, prepositions, and conjunctions. The CPIDER counted correlated very closely with the human results. The potential application maybe applied this method into readability and reading comprehension. Hong et al. [11] proposed statistics-based scheme for extraction of new words based on the categorized corpora of Google News retrieved automatically from the Google News site.

Sun et al. [12] stated that the procedure of new words identification and POS tagging were usually separated and the features of lexical information cannot be fully used. They used Latent Dynamic Conditional Random Field and Semi-CRF model to detect new words. The results showed that the proposed method was capable of detecting even low frequency new words together with POS tags. Jiang et al. [7] proposed The Potential Unknown word Detection (PUD) Algorithm was used to detect longest potential unknown words for the single-character model and potential unknown words for the affix model. Their algorithm was statistical based words frequency prediction approaches.

Although numerous literatures were related to automatic detecting new vocabulary, there was only a limited amount of these studies available for Chinese New vocabulary language teaching.

3 Fundamental of Term Networks

Our proposed approach relies on the concept of term co-occurrence or term network. Traditional term co-occurrence (or co-term) network is built by representing terms in vector form which denotes the occurrence of the terms in all documents. To allow more in-depth exploration, we believe that a sentence-level co-term network would be more suitable because terms co-occur in the same sentences tends to exhibit clearer relationship than those in the same documents only.

The seed terms for the network can be obtained from a given list, output of an NLP parser, or any key term extraction algorithms [13]. Term similarities are then calculated based on these vectors to form the similarity matrix of terms [14], which is the input to the co-term network. If there are n terms from m documents, the time complexity can be on the order of $O(n^2m)$, where m steps are required to calculate similarity between any of n^2 term pairs.

The major difference of ours from the above is to limit the terms to be associated to those that co-occur in the same logical segments of a smaller text size, such as a sentence. Association weights are computed in this way for each document and then accumulated over all documents. This changes it into a roughly $O(mk^2s)$ algorithm, where k is the average number of selected key terms per document and s is the average number of sentences in a document. As can be seen, the larger the n and m, the bigger the difference between $O(mk^2s)$ and $O(n^2m)$, because k can be kept less than a constant and so can s by breaking large documents into smaller ones.

The technique for term co-occurrence analysis to be used in our work is described as follows: key terms identified from each document based on Tseng’s algorithm [2] are first sorted in decreasing order of their term frequencies (*TF*), or *TF* x *Term_Length*, or other criterion such as *TF* x *IDF* (Inverse Document Frequency) if the entire collection statistics are known in advance. Then the first k terms are selected for association analysis. A modified Dice coefficient was chosen to measure the association weights as in (1).

$$wgt(T_{ij}, T_{ik}) = \frac{2 \times S(T_{ij} \cap T_{ik})}{S(T_{ij}) + S(T_{ik})} \times \ln(1.72 + S_i) \quad (1)$$

where S_i denotes the number of sentences in document i and $S(T_{ij})$ denotes in document i the number of sentences in which term T_j occurs. Thus the first term in Equation (1) is simply the Dice coefficient similarity. The second term $ln(1.72+S_i)$, where ln is the natural logarithm, is used to compensate for the weights of those terms in longer documents so that weights in documents of different length have similar range of values. This is because longer documents tend to yield weaker Dice coefficients than those generated from the shorter ones. Association weights larger than a threshold (1.0 in our implementation) are then accumulated over all the documents in the following manner as in (2).

$$sim(T_j, T_k) = \frac{\log(w_k \times n/df_k)}{\log(n)} \times \sum_{i=1}^{n} wgt(T_{ij}, T_{ik}) \tag{2}$$

where df_k is the document frequency of term k and w_k is the width of k (i.e., number of constituent words). Variations of the method for dealing with other document types can be found in [15, 16].

4 Research Method

The above term co-occurrence analysis has been applied to Tseng et al [2] to extract junior high school textbook terms from news reports for measuring civic scientific literacy. This work extended their approaches by replacing the textbook terms with those basic contextual terms from a teaching material developed by Hsin et al. [17]. The flow chart of finding the candidate terms is shown in Figure 1. From about 2,900,000 articles of 10 years daily news, we applied the key terms extraction and association analysis techniques to yield 670,000 key terms, each with some related terms. A key term is a frequent topical term in a news article and its related terms are other key terms that frequently co-occurred with the key term in the same sentences. For example, “霸凌” (bullying) is a keyword in a certain article and “學校” (school), and “老師” (teacher) may be its related terms, because bullying often occurs in elementary schools and junior high schools and teachers are often involved in dealing with bullying events. To detect those key terms that are suitable for Chinese teaching, we then match the related terms of the key terms with the basic contextual terms from the teaching material developed by Hsin et al. [17].

The basic contextual terms were compiled to help foreigners to learn basic Chinese terms, such as “school”, “teacher” categorized in its education category. There are twelve categories in this teaching material including personal information, living, occupation, leisure, travels, social relationship, medical cures, educations, shopping, food and bakery, banking, and safety categories. In total, there are about 1100 basic contextual terms and the number of key terms that have matched related terms are about 94,000. As an example, “bullying” is one the 94,000 key terms because two of its related terms “school” and “teacher” are in the set of about 1100 basic contextual terms. To know the trend of each key term over the 10 years of daily news articles, we computed the number of articles the key term occurs (i.e., document frequency) for each year, which constitutes a time series of 10 document frequencies for the key term. We then applied the following trend slope formula, verified by Tseng et al (2009), to know their trend. The key terms were finally sorted based on their trend slope for ease in new term browsing and selection. Tseng et al (2009) proposed to determine whether a topic is a hot topic or not based on the formulas as follows:

i: time index (i.e., the year index)
d_i: number of documents of the key term in year i
slp: slope of the linear regression
x_i: *normalized value of index i*
y_i: *normalized value of document frequency*

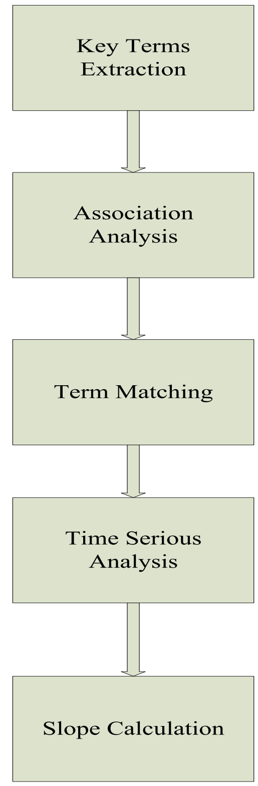


Figure 1. Flow chart of finding new vocabularies

$$x_i = i - \frac{1}{n}\sum_{i=1}^{n} i, \quad y_i = d_i - \frac{1}{n}\sum_{i=1}^{n} d_i, \quad slp = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} x_i^2} \tag{3}$$

In the above formula (3), the more the documents in which a key term occurs in recent years, the higher the score of slp. It is noted in Tseng et al (2009) that it is the rank of the slope that matters in determing the novelty of the terms, rather than the absolute slope values of the terms.

5 Research Results

We not only implemented the above approach, but also built a search system for ease of verification. Figure 2 shows an example of a candidate term. A user inputs a query item, such as “霸凌” (bullying), then the system responds with (1) short snippets of 3 of 195 search results in news database (2) the related terms of the query term “bullying” to know its category context (3) a time series (trend) of the query item. It can be seen that bullying started in 2004 and grew into a salient term year after year. Ten more new terms similar to the above for possible Chinese language teaching were listed in Table 1; the Chinese translations were located at the bottom of table. In table 1, slope represented *slp* in formula (3) of ten years data from 2000 to 2009. The new term, "金融海嘯" (pronounced "Jin-Rong-Hai-Xiao") (means global financial crisis) began to exist since 2008; before 2007, the news database did not have this vocabulary, the frequency equaled to zero before 2007. The term, "部落格" (pronounced "Bu-Luo-Ge") (means blogs) began from 2002; "火星文" (pronounced "Huo-Xin-Wen") began since 2005. These Chinese new terms were novel prevailing phenomenon in Taiwan society and could be used as new Chinese vocabularies for teaching.

6 Conclusions

New vocabularies extraction for compilation of vocabulary textbooks are useful to close the gap between the language been taught and the language been used. Thus a software tool which helps collect new daily use Chinese vocabularies automatically is desirable. However, such a technical tool is not trivial to develop. Based on some text mining techniques such as key term extraction, term association analysis, and hot topic detection, we are able to extract candidate vocabularies in support for Chinese language teaching; life style vocabularies could be included into the teaching materials, and students could more easily join to Chinese culture society. The results were welcomed by Chinese teaching professionals, showing the potential usefulness of this work. Future work will collect more corpora and examples for further applications.

7 Acknowledgment

The work is supported in part by the "Aim for the Top University Plan" of National Taiwan Normal University, sponsored by Ministry of Education, Taiwan, R.O.C.

8 References

[1] Y. H. Tseng, Y. Lin and Y. Lee. "A comparison of methods for detecting hot topics". Scientometrics, Vol.81 No.1, 73-90, 2009.

[2] Y. H. Tseng, C. Y. Chang and S. N. C. Rundgren. "Mining concept maps from news stories for measuring civic scientific literacy in media". Computers & Education, Vol.55 No.1, 165-177, 2010.

[3] R. J. Blake. "The use of technology for second language distance learning". The Modern Language Journal, Vol.93 No., 822-835, 2009.

[4] M. Levy. "Technologies in use for second language learning". The Modern Language Journal, Vol.93 No., 769-782, 2009.

[5] J. Ha, Y. Zheng and B. Kim. "High speed unknown word prediction using support vector machine for Chinese text-to-speech systems". Lecture Notes in Artificial Intelligence , Vol.3248 No., 509-517, 2005.

[6] H. Q. Li, C. N. Huang, J. F. Gao and X. Z. Fan. "The use of SVM for Chinese new word identification". Lecture Notes in Computer Science, Vol.3248 No., 723-732, 2005.

[7] X. Jiang, L. Wang, Y. Cao and Z. Lu. "Automatic recognition of Chinese unknown word for single-character and affix models". Knowledge Engineering and Management, Vol.123 No., 435-444, 2011.

[8] A. Kuba, A. Hocza and J. Csirik. "POS tagging of Hungarian with combined statistical and rule-based methods". Lecture Notes in Artificial Intelligence , Vol.3206 No., 113-120, 2004.

[9] S. V. Pakhomov, A. Coden and C. G. Chute. "Developing a corpus of clinical notes manually annotated for part-of-speech". International Journal of Medical Informatics, Vol.75 No.6, 418-429, 2006.

[10] C. Brown, T. Snodgrass, S. J. Kemper, R. Herman and M. A. Covington. "Negation recognition in medical narrative reports". Behavior Research Methods, Vol.40 No.2, 540-545, 2008.

[11] C. M. Hong, C. M. Chen and C. Y. Chiu. "Automatic extraction of new words based on Google News corpora for supporting lexicon-based Chinese word segmentation systems". Expert System With Applications, Vol.36 No.2, 3641-3651, 2009.

[12] X. A. Sun, D. G. Huang, H. Y. Song and F. J. Ren. "Chinese New Word Identification: A Latent Discriminative Model with Global Features". Journal of Computer Science and Technology, Vol.26 No.1, 14-24, 2011.

[13] Y. H. Tseng. "Multilingual Keyword Extraction for Term Suggestion". 21st International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR 1998, Australia, 1998.

[14] G. Salton. "Automatic text processing: The transformation, analysis and retrieval of information by computer". Addison-Wesley, 1989.

[15] Y. H. Tseng. "Automatic thesaurus generation for Chinese documents". Journal of the American Society for Information Science and Technology, Vol.53 No.13, 1130-1138, 2002.

[16] Z. P. Ho, Y. H. Tseng, K. S. Yang and C. C. Chen. "Term mining for relation visualization and exploration-Some practical applications in crime investigation". The 3rd International Conference on Data Mining and Intelligent Information Technology Applications (ICMIA2011), Venetian, Macau, China, Vol.1 No.1, 192-195, Oct 24-26, 2011.

[17] S. C. Hsin, S. H. Deng and M. Y. Lee. "Basic wordlist for teaching Chinese as a second language". National Taiwan Normal University Published, 2010.

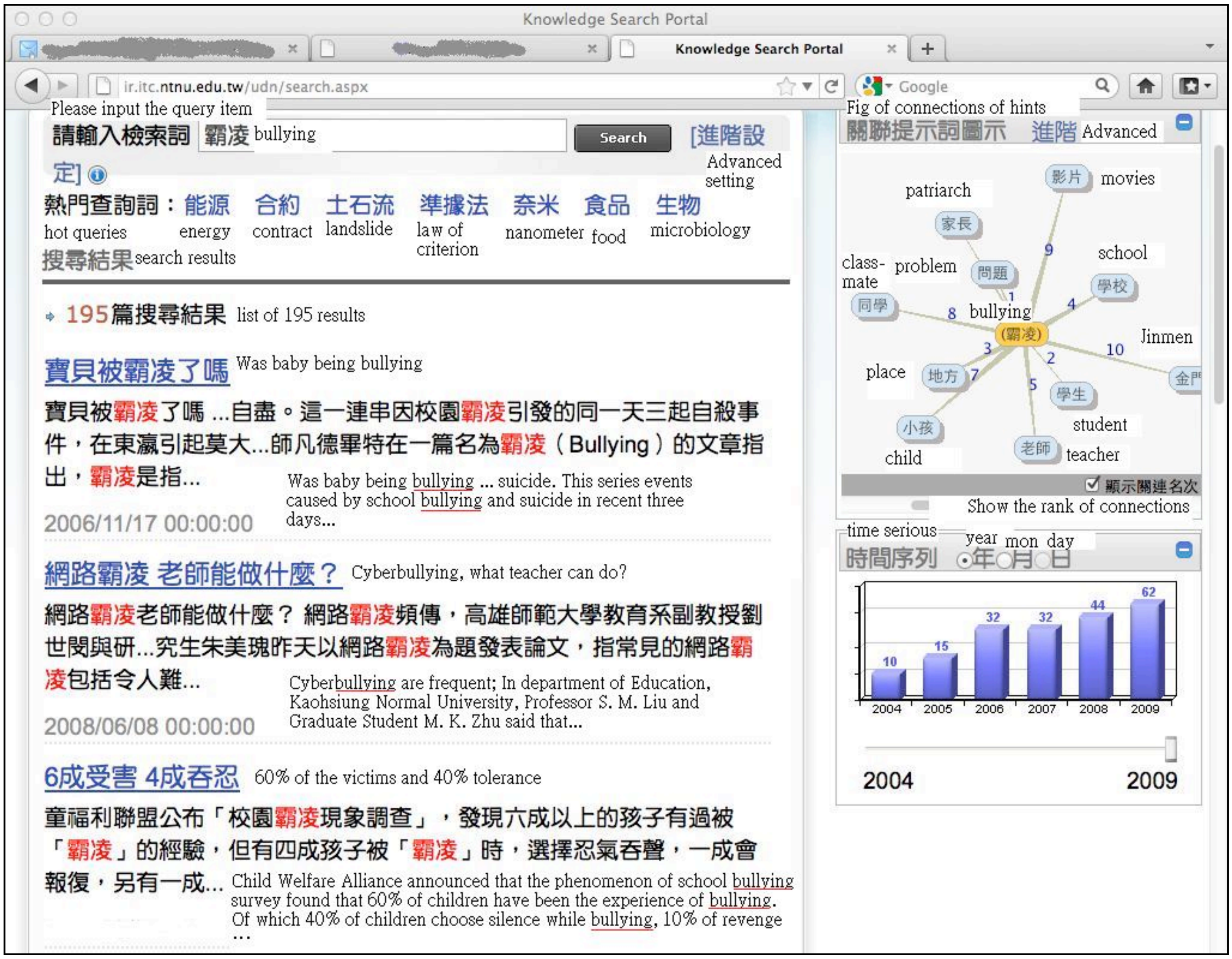


Figure 2. Example of a search term for verification.

Table 1. The demonstration of trends of selected vocabularies from 2000 to 2009.

Term	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	slope
A:金融海嘯	0	0	0	0	0	0	0	0	1810	6321	421
B :黑心商品	0	0	0	0	21	44	24	28	102	33	8
C :減碳	1	1	0	0	0	4	3	106	2131	1775	190
D :山寨	9	8	14	11	6	9	8	4	43	704	40
E :產銷履歷	0	0	0	0	2	9	20	26	10	9	2
F :部落格	0	0	1	2	15	312	1188	1370	1476	1541	212
G :火星文	0	0	0	0	0	22	193	44	28	34	8
H :粉絲	55	69	62	96	721	1811	1866	1262	1418	1172	193
I :台客	47	70	67	73	67	204	162	174	143	254	20
J :詐騙集團	356	389	424	706	954	1020	946	807	702	490	37

Notes: Term A means global financial crisis (belongs to the banking category); B means black heart of goods (shopping); C means carbon reduction (living); D means cottage (shopping); E means traceability resume (food and bakery); F means blogs (social relationship); G means odd wording (social relationship); H means fans (social relationship); I means Taiwanese in non-native countries (social relationship); J means fraud groups (safety).

University Social Space for Consulting and Development of Reliable Educational Multimedia Content

Agustín León-Barranco and J. Martín Mejía-Estrada
agustin_leon@my.uvm.edu.mx, jmejia@uvmnet.edu

Engineering Department
Universidad del Valle de México (UVM)
Camino Real a San Andrés Cholula No. 4002
Col. Emiliano Zapata, C.P. 72810
San Andrés, Cholula, Puebla, México.

Abstract - *This paper states the need for the creation of a university social space for consulting and development of educational reliable contents. A public virtual space will allow internauts not to depend of physical space or schedules to work; to guarantee reliable information will avoid the internauts to receive incorrect information or to set up new knowledge upon fake foundations; multimedia contents or which information could be shown in several ways will facilitates their learning; to avoid effort in information searching so to increase the time they can spend to process in a better way their knowledge; to work within a group will allow them to confirm or to question what they and others learn.*

Keywords: Collaborative learning, educational multimedia content, education 2.0, social space, reliable information, ICT, social networks, Internaut.

1 Introduction

The easiness and comfort we have to communicate with other people have allowed Internet to become the highest interest communication mean in the last years; the huge information amounts have allowed the Web to become the preferred consulting space among students around the world. Unfortunately, only a very reduced amount people or institutions is responsible of the veracity of information they publish. In countries like Mexico with a lot of educational backwardness, it is too dangerous that people has a limited or null sense of research, because by swimming alone without a lifeguard into an immensity of information, there is a risk to get drowned into the ignorance. Owing to the fact that it is too difficult changing the bad researching habits in students, this work states and encourages the creation of a university social space for consulting and development of educational reliable contents. A public virtual space will allow internauts not to depend of physical space or schedules to work; to guarantee reliable information, to avoid the internauts to receive incorrect information or to set up new knowledge upon fake foundations; multimedia contents or which information could be shown in several ways, will facilitates their learning; to avoid effort in information searching so to increase the time they can spend to process in a better way their knowledge; to work within a group will allow them to confirm or to question what they and others learn.

2 ICT and competitiveness

Development of Information and Communication Technologies (ICT) has been really dizzy and shocking in the last years of 21th century, transforming the way people communicate each other and with their surroundings. The great amount of information we create every day and the easiness to communicate with other individuals have strengthened the new age, called the "Knowledge Society" by Peter Drucker (1993). Today, the knowledge constitutes a fundamental resource in production processes of goods and services in our society, and by acquiring proper competences and values, it helps the self-sustaining development and competitiveness of any country.

2.1 ICT in México

The access to ICT in Mexican houses has grown in the last years but this growth has been too slow regarding to the access to a computer or Internet. Next, some charts are shown to see penetration of ICT in México from 2001 to 2002 and from 2004 to 2010. Up to 2010, from each 100 Mexican houses: 94.7% had a TV, see Figure 1; 26.7% had cable TV, see Figure 2; 80.6% had telephone service, see Figure 3 (46.3% had both kinds: fixed line telephone and mobile phone service; 42.2% had only mobile phone service and 11.5% had only fixed line, see Figure 4); only 29.8%, barely a quarter, had a personal computer at home and 22.2% had access to Internet, see Figure 5 (INEGI, 2001-2002 & 2004-2010).

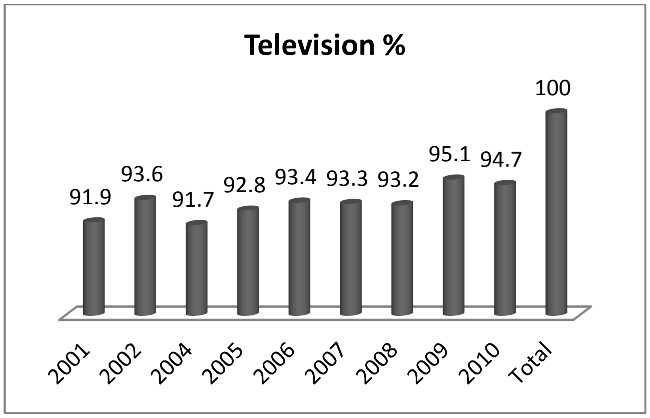


Figure 1. Percentage of Mexican houses with access to TV

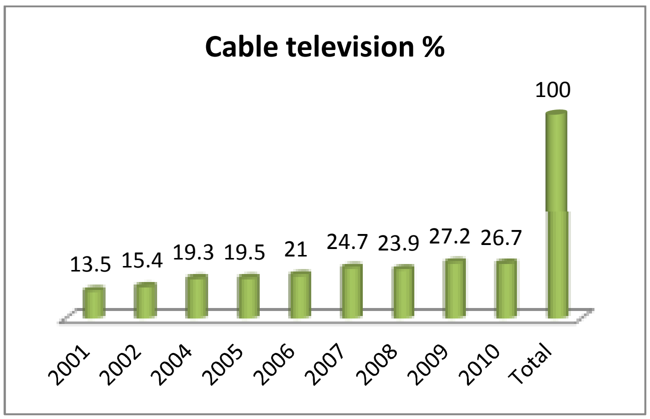


Figure 2. Percentage of Mexican houses with access to cable TV

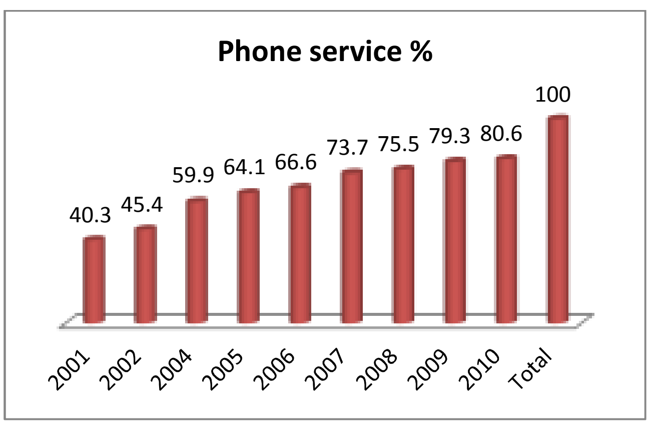


Figure 3. Percentage of Mexican houses with access to telephone service

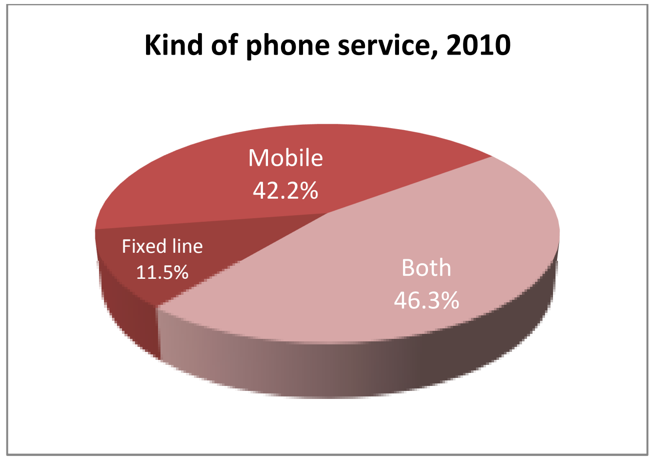


Figure 4. Distribution of Mexican houses with access to telephone service

Software Development and Services (SDS) industry has gained a lot of strength around the world within the ICT industry. Nations competitiveness has become dependent on this sector where, usually, nations like U.S.A, Japan, or countries from European Union used to dominate, but now, new countries such as China, India, Canada, Ireland among others have emerged. Software Development has become a great industry where, unfortunately, nations like México have fallen behind because their competitiveness has slowed the investment down in this sector, by causing and prolonging a dependence of expensive foreign products and services.

To assess the importance of SDS industry into the competitiveness of countries, for instance, let us compare two American countries like México and Canada. According with the "Global Competitiveness Report 2010-2011" of the World Economic Forum (WEF, 2010) Canada was 10th in the list and México was the number 66, see Table 1, besides, "The Global Information Technology Report 2009-2010" presented by the INSEAD[1] business school and the WEF (Dutta & Mia, 2010), Canada was 7th and México 78th in the list with regard to conditions to facilitate development and spreading of ICT to increase the development and competitiveness. For 2006, in the "Digital Planet Report" published by the World Information Technology and Services Alliance (WITSA, 2006) Canada was developing its software industry mostly focused on entertainment, such as 3D animation, special effects and multimedia. Canada had more than 1,100 technology companies and its industry generated more than 6 thousand million in exportations and more than 7 thousand million dollars in buying. On the other hand, for instance, there is not digital animation industry in México; from the legal

[1]At the time of its creation, it was the acronym for "*Institut Européen d'Administration des Affaires*"

viewpoint, there is no governmental organization to support digital animation as an industry, being the National Chamber of Electronics, Telecommunications and Information Technologies Industry (”Cámara Nacional de la Industria Electrónica, de Telecomunicaciones y Tecnologías de la Información”, better known as CANIETI) the organization that has associated the very few animation studios and videogames developers. It was until the year 2006 that CANIETI organized the “First National Festival of Animation and Videogames Creanimex 2006” carried out in Guadalajara, Jalisco, México to promote the creation and growing of more professional companies that allow México to establish and strengthen an industry capable to satisfy the needs of national and international markets.

3 Google generation

The great amount of information we generate today just as the easiness we have to communicate with other individuals have made the Web to become the favorite consulting space among Internet users, from now on referred as “Internauts”. Unfortunately, it is just a very reduced amount of people or institutions responsible of the information veracity that is published. In the research “Information Behavior of the Researcher of the Future” (2008) hired for the Joint Information Systems Committee (JISC) and the British Library, it was intended to find out if the ICT were creating new ways to look for information in young people born after 1993 known as “Google Generation”, where it was discovered, among other things,

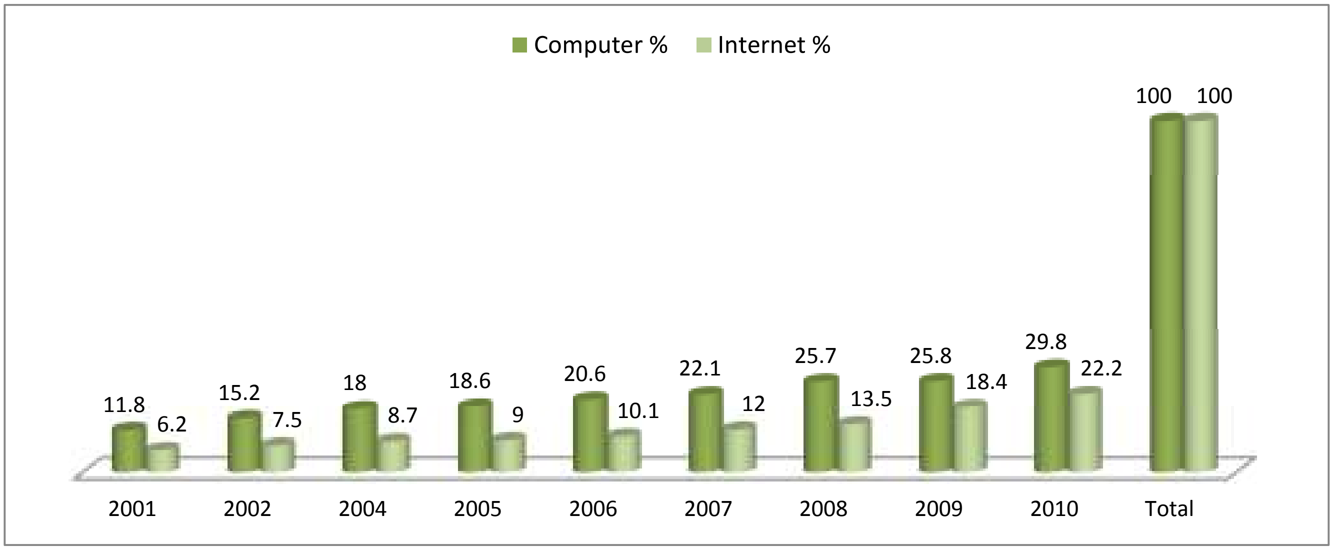


Figure 5. Percentage of Mexican houses with a computer and access to Internet

Facts like those stated ICT, and especially SDS, started to increase their importance lately in México because of their impact upon its competitiveness. It was until 2002 when, as part of the National Development Plan (NDP) 2001-2006 (“External Evaluation of National Program for Development of Software Industry”, 2006), the Secretary of the Economy launched the National Program for Development of Software Industry (“Programa Nacional para el Desarrollo de la Industria del Software”, better known as PROSOFT) with the objective of contributing to create the necessary conditions that could help México to have a competitive software industry worldwide-level that could rise and spread the nation competitiveness, guarantee the development, and promote a well-balanced regional economic development.

Country/Economy	Rank
United States	4
Canada	10
Chile	30
Puerto Rico	41
Barbados	43
Panama	53
Costa Rica	56
Brazil	58
Uruguay	64
Mexico	66

Table 1. Global Competitiveness Index 2010–2011 rankings in America (top ten), WEF

their lack of analytical and critical skills to judge the relevance and reliability of information they find in

Internet. It was revealed that 89% of higher education students use searching engines to start to look for information; it was also revealed that 93.3% were satisfied or very satisfied with the general experience of using search engines; this is because they accommodate or fix better to the lifestyle of higher education students than physical or on-line libraries.

4 Social networks in education

When Tim Berners Lee created the World Wide Web (WWW) or simply Web (aka Web 1.0) in the early 90s, the interaction of Internet users was mainly through email, web browsers or static HTML pages or with very limited dynamic. The concept of Web 2.0 was born in a brainstorming session in 2004 between Dale Dougherty of O'Reilly Media and Craig Cline of MediaLive International (O'Reilly, 2007). While it is true that Web 2.0 concept is in large part to the marketing promoted by Tim O'Reilly, founder and leader of the publishing of books and computer technology O'Reilly Media, also marketing of the same environment (authors, organizations, businesses, institutions and others) helped to strengthen this new phenomenon of social impact selling products, generating content and tools 2.0: governments 2.0, businesses 2.0, education 2.0, etc.. Under this ideology, in 2006 John Markoff of the New York Times describes Web 3.0 as the sum of Web 2.0 and Artificial Intelligence, i.e., a "web guided by common sense" (Markoff, 2006). This "smarter" network is a semantic network that seeks, for example, that machines can understand the information that is hosted on the Web and filter information automatically and precisely.

Web 2.0 is now a "social ownership" for allowing the user to become not only a reader of static pages as usually happened in Web 1.0, but also provided the user the opportunity and freedom to become a creator and editor of interactive dynamic pages, enabling collaboration and social appropriation of Internet.

4.1 Education 2.0

This document refers to social networks on the Internet as a virtual form of interaction between individuals, groups or institutions whose purpose is linked to share information, knowledge, ideas, interests, opinions, differences, etc. Therefore, the use of Web 2.0 in education (Education 2.0) promotes a collaborative learning which looks for the development and exchange of knowledge within small peer groups, aimed at achieving academic goals. An appropriate use of social networking in education could bring many advantages: some teacher autonomy, freedom of action by students, stimulation of autonomous learning, interaction and collaboration with others, rich multimedia content (text, graphics , videos, sound, hypertext, etc..), organization and systematization of information, development of critical thinking during the process of discussion forum, practice writing ability, teaching-learning process beyond the physical and temporary space, etc.

The number of Internet users has grown around the world, where social networks have been the most striking phenomenon in recent years. The worldwide growth of the online population was of 10.4% from 1244.1 to 1374 million. Middle East-Africa was the region with the highest growth (31.5%) followed by Asia Pacific (14.1%), Latin America (13.7%), Europe (4.9%) and North America (0.2%), see Figure 6 (AMIPCI[2], 2011).

Figure 6. Online population around the world and by regions, May 2010 vs May 2011

Another point to note is that 60% of the audience is concentrated in Latin America between Brazil and Mexico, see Figure 7. Of all the online population in Latin America, more than half, 62% are young people less than 35 years of age, see Figure 8.

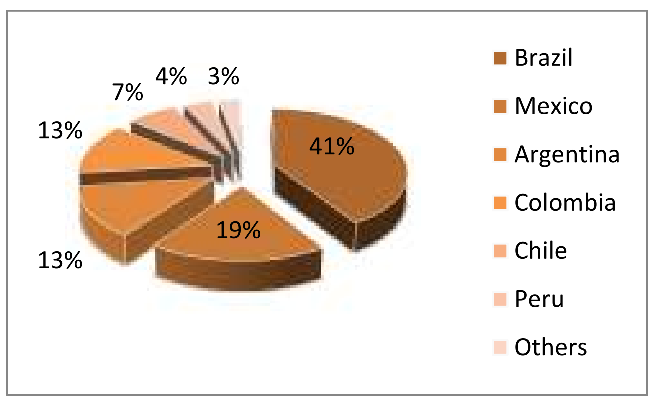


Figure 7. Distribution of the audience in Latin America, May 2011

Specifically in Mexico, the number of users has grown considerably in recent years, rising from 17.2 million in 2005 to 34.9 million in 2010.

In Latin America, from the total online population who is larger than15 years old and access the Internet from

[2] The Mexican Internet Association (Asociación Mexicana de Internet, better known as AMIPCI)

home or work, a high percentage (62%) is between 15 and 35, of whom 33% are between 15 and 24. In Mexico, from the total online population over 6 years, 77% are between 6 and 34 years old with the following distribution: 10% (6-11), 27% (12-17), 23% (18-24) and 17% (25-34).

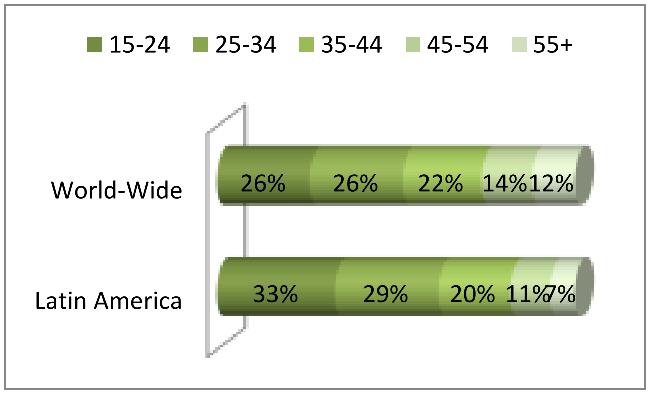


Figure 8. Age distribution Latin America vs. the World, May 2011

A little more than 1.100 million Internet users visit a social network, i.e., almost all of the online population. The distribution of the social networking world population by region is as follows: 30.1% Europe, 18.1% North America, 32.5% Asia Pacific, 10.2% Latin America and 9.1% Africa-Middle East. It is estimated that in Latin America are almost 115 million users on social networks[3]. On the issue of social networks, it is noteworthy that in Mexico, 61% of Internet users access a social network. This means that if there are 34.9 million Internet users and 61% use a social network, we're talking about 20.74 million social network users in the country.

The frequency with which the Mexican Internet users access to social networks is very high because 60% do so daily and 28% two or three times a week, the three most used social networks in Mexico are Facebook (39%), YouTube (28%) and Twitter (20%). It highlights that younger users of Mexico spend more time online than other young users in the world. The Mexican Internet users between 15 and 24, spends 32.7 hours online per month average (8 hours more than the global user 15 to 24 years). Almost 89% of Mexican Internauts use search or browse sites like Google, and because of its young population, also has a high reach in the category of entertainment sites (76.3%) where YouTube dominates. Almost 9 out of 10 Mexican Internauts look for a site within the category of search engines to conduct information searches. In fact, Mexico with Colombia and Venezuela have the most intense users search the world, with rates far above average regionally and globally.

If we were able to exploit the potential of social networks in education, there is no doubt that we would witness a real "new era" of education suitable to boost the competitiveness of all countries.

5 Virtual social space

According with the report "Human Development Report 2011. Sustainability and Equity: A Better Future for All" of the United Nations Development Programme (UNDP, 2011), Latin American countries like Chile (44[4]) and Argentina (45) have a very high human development; countries like Uruguay (48), México (57) and Brazil (84) have a high human development, see Table 2. Despite that several Latin American countries have a very high or high human development, the real truth is that most them are still worried for bringing the ICT to people, for instance, as was stated earlier, México still faces the challenge of bringing the ICT to all Mexican houses while many other serious problems are appearing or even getting stronger such as the one of the Google Generation.

Country	Rank
very high human development	
United States	4
Canada	6
Chile	44
Argentina	45
Barbados	47
high human development	
Uruguay	48
Cuba	51
Bahamas	53
Mexico	57
Panama	58

Table 2. Human Development Index 2011 in America (top ten), UNDP

There is no doubt that, for example, in the Mexican education, there are still two important issues: on one hand, to provide the necessary ICT to strengthen the sector, on the other hand, to prepare or help people to use them. But, ¿what happens with those internauts that already have access to ICT and that lack of analytical and critical skills to judge the relevance or reliability of information they find in Internet?, ¿how to develop research habits in internauts or future internauts?, ¿how to avoid internauts fall in sites where the information is not reliable? These and many other questions need to be answered before the problems get stronger and become more difficult to solve. It is time to pay more attention to problems that have been generated in those places where the ICT have arrived. This work is

[3] Source: comScore Media Metrix, May 2011

[4] Raking in the Human Development Index

addressed in a way which education is the starting point, the information is the path, and the knowledge is the destination.

Varied ways of intercommunication that today has the Web, particularly social networks, have allowed Internet to become the favorite communication mean of millions of people around the world. By giving the user the opportunity to become not just a reader of the Web, but also into an editor of the same one, allowed to cumulate huge amounts of information to the existing ones. By giving the user the opportunity to create and edit virtual communities just as collaborating in other communities allowed an uncontrollable information growth. The two problems in the "information universe" that are addressed in this document are: on one hand, there is just a very reduced amount of people or institutions responsible of information veracity that is published; on the other hand, in emergent nations like México with a lot of educational backwardness, it is too dangerous that students have a limited or null sense of research, because by surfing alone without lifeguard into an immensity of information, there is a risk to get lost or drowned into the ignorance.

information sources, and thus, the obtained knowledge will be wrong. If the knowledge is the fundamental core in goods and services production processes in our society, the wrong constructed knowledge will seriously be compromising the development and competitiveness of a nation.

To avoid undesired endings, it is necessary the internaut may acquire the basic competences and appropriated values to reach a correctly constructed reliable knowledge. It is vital the internauts acquire research skills in such a way their methods to look for useful information can be improved. Unfortunately, it is a hard task of a long time to change the bad habits of a Google Generation that has got used to trust and depends on a searching engine because it fixes better to her lifestyle. It is a great challenge for education to achieve the internauts can dedicate more time to investigate the information they read, because for them, there are other more important activities such as the socialization itself.

So far, it has not been effective to tell students to be careful with the information they read in the Web because most of them still make the same mistakes. If at the same

Information	Request
Unreliable source A: In September 16th, the beginning of army movement for Independence is celebrated…	¿What happened in September 15th, 1810?
Unreliable source B: In September 16th, the Independence of México is celebrated…	
Unreliable source C: In September 15th, the "declaration of independence" is celebrated …	
Unreliable source D: In September 16th, the "grito de Dolores" is celebrated …	
Reliable source: The "declaration of Independence" or "Grito de Dolores" was given in September 16th, 1810; and it marks the beginning of the army movement for México's Independence. It used to be celebrated in September 16th but for command of President Porfirio Diaz it was moved a few hours to September 15th to coincide with his birthday.	

Table 3. Reliable and unreliable information about Mexico´s independence history

To go from information to knowledge is not so simple, and less reliable. To explain this, information about Mexico's independence is shown in Table 3; there is a question, four not reliable information sources and one reliable source. Let us suppose that an internaut is requested to respond the question of the second column. If the internaut access to one of the four not reliable information sources shown in the first column without to be worried or concerned about who is the source, ¿can his answer be reliable? let us suppose the internaut has the skill to read up the four sources and he bump into A, B, C and D, ¿will his answer even be reliable? It is evident that the internaut is not going to find his answer in those four

level of instilling students the necessary competences to carry out a research, we were capable to provide them a reliable space where they could find and work all the information they are looking for, it could be assured that at least the information upon they are constructing their knowledge is reliable. If we come back again to Table 3, there is no doubt that by providing a reliable source to students, they will gain more time to process calmly the information and construct their own knowledge either alone or within a group by collaborating with someone else. The most probably is that internaut responds that in September 15th of 1810, nothing happened to celebrate officially and that the 15th day is just used to celebrate the "Declaration of Independence”. This mean that it is better to provide students a reliable virtual space where student can access information anytime and anywhere to take in a direct way all the information they need.

The main idea is to create a university social space for consulting and development of reliable multimedia educational content. This space is initially thought for university people such as: students, teachers, administrators, researchers, professionals, experts and all interested. The participation level goes from the simple interest for consulting reliable multimedia content up to participating in the development of reliable information. To add new multimedia content to the reliable source, it must be reviewed and approved for at least two professional experts in the respective involved area, they will be on charge of verify the validity and reliability sources of the information itself. Most of the information must be free access but there is also the option for some authors to sell their work, obvious that, the content to be published must also be reliable. The plan is to promote the development of multimedia content of high quality by selling videos, animations, sound files, documents, and so on at a very low cost accessible for any internaut; this way students, teacher, researchers and others will be motivated by receiving earnings from their work because if their content is good, more users will download their work, and thus, the economic benefits will be higher, but above all, an author can gain more recognition from the academic and professional community.

6 Conclusions and future work

The university social space introduced in this work has in natural way the academic impact upon all educative levels. This proposal answers the previously stated questions, and even, it could cover all educational levels: kindergarten, elementary, high school, university and post degree. This answer just comes out in a proper way the ICT resources at our disposal by giving to people a virtual public space with reliable information. This space is open 24 hours everyday of the year and accessible from anywhere by means of a computer or mobile device; in this place the internauts can find and work all the information they are looking for. We know bad habits in students are difficult to change because of their lifestyle. They prefer to dedicate the least time to educative labors. If internauts do not spend much time looking for reliable information, they will have the opportunity to work and process information to reach a better constructed knowledge. The advantage of this proposal is that it allows the opportunity to all interested people (students, teachers, experts, professionals, and so on) to participate in this project.

7 References

[1] Drucker, P. F. (1993). *Post-Capitalist Society*. USA: HarperCollins Publishers Inc.

[2] Dutta, S. & Mia, I. (Eds.). (2010). *The Global Information Technology Report 2009-2010*. Ginebra, Suiza: INSEAD & WEF.

[3] Secretary of Economy, *External Evaluation of National Program for Development of Software Industry* (2006). Survey carried out by the Faculty of Economics, UNAM. México.

[4] Secretary of Economy, *PROSOFT 2.0: Program of Development of the Service Sector of Information Technologies* (2008). México.

[5] OECD, *ICT and Economic Growth* (2003).

[6] *IMD World Competitiveness Yearbook*, WCY (2010). Suiza.

[7] British Library and JISC, *Information behavior of the researcher of the future* (2008).

[8] WEF, *The Global Competitiveness Report 2010-2011* (2010). Geneva, Switzerland.

[9] National Institute of Statistics, Geography and Informatics (Instituto Nacional de Estadística, Geografía e Informática, INEGI). http://www.inegi.org.mx, *Statistics on Availability and Use of Information and Communication Technology in Households* (Estadísticas sobre Disponibilidad y Uso de Tecnología de Información y Comunicación en los Hogares, EDUTICH). 2001-2002, 2004-2010. México 2012.

[10] WITSA, *Digital Planet 2006* (2006).

[11] UNDP, *Human Development Report 2011. Sustainability and Equity: A Better Future for All* (2011).

[12] O'Reilly, Tim (2007). *What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software*. Published in: International Journal of Digital Economics No. 65 (March 2007): pp. 17-37.

[13] Markoff, John (2006). *Entrepreneurs See a Web Guided by Common Sense*. The New York Times.

[14] AMIPCI, *Social Networks in Mexico and Latin America* (2012).

Development of Use of a Virtual Construction Company Simulation System for Education

T. Korman[1] **and H. Johnston**[1]
[1]Construction Management Department / California Polytechnic State University
San Luis Obispo, CA, U.S.A.

Abstract - *Simulations and learning games use technology to create real-world experiences to provide the opportunity to engage, have fun, and truly learn. Many have been designed to meet specific learning goals, i.e. sharing case studies to demonstrating very complex situations. Gaming is not new to higher education but in the past was done in a very narrow vein and because of the complexity and development time required to produce them. Most have not been robust enough to engage students. Managing Construction involves being able to make decision to balance time, cost, quality, resources, and identifying and solving a variety of issues. As the millennium generation enters the higher education system many have spent many hours playing computer games as they have in the classroom during their lifetime; therefore, it is a natural transition that our learning environments begin to use techniques from the gaming world. The skills required of today's construction management personnel are a combination of management skills and technical knowledge. This paper describes the development of gaming system designed and developed at California Polytechnic State University, San Luis Obispo to educate construction management students.*

Keywords: Simulation, Gaming, Construction, Education

1 Introduction

The faculty at California Polytechnic State University, San Luis Obispo (CPSLO) struggles with developing appropriate teaching methods and content to prepare students in the engineering and construction industries. This struggle results from the desire to teach students to how to develop constructible designs while retaining the delivery of core engineering fundamentals. In the current global environment, it is imperative that graduates are prepared to enter the workforce with the skills necessary to make immediate contributions. The complex nature of engineering and the construction industries often finds recent graduates performing entry-level tasks many years after they graduate because they lack the skills and experience required to be involved in day-to-day operations. Industry reports that students' limited preparation often delays their making contributions to integration, collaboration, productivity, and accuracy, all of which are necessary in the engineering and construction industries. Project-based learning combined with multidisciplinary training significantly enhances the ability of students to successfully enter the modern work environment and to respond effectively to the rapid evolution of knowledge and the ongoing iteration of problems in complex systems.

Dialogue with our IAB has revealed the following important issues and obstacles our students experience upon entering industry. First, students often have not encountered large-scale team design projects and, therefore, have to learn how to work in such an environment. They must gain experience in the process, develop a technical specialization to support their project role, and build their ability to collaborate on and contribute to multidisciplinary projects. Secondly, we discovered that our students were not prepared to apply design and construction engineering fundamentals to real world complex projects. For example, in order to begin work on complex building and civil infrastructure type projects, students must first be exposed to smaller components of the project and guided through specific aspects of a project in order to succeed in accomplishing the task at hand; and even further removed is the opportunity for students to gain knowledge utilizing project controls to monitor and evaluate an active project.

In addition to the educational deficiencies noted in our curriculum, CCE curriculums generally do not present an integrated approach to engineering education that includes practical applications of theoretical knowledge incorporating constructability issues. Students often master the course and laboratory work associated with courses in the curriculum, but they do not gain a comprehensive engineering experience that requires them to synthesize what they have learned in their curriculum and extend their knowledge through independent learning that reaches outside their field of study, specifically in the topics of constructability. This is further observed at the community colleges where students do not have the opportunity of being immersed in a large-scale engineering academic environment of a four-year institution and frequently lose interest in pursuing an engineering career.

Traditionally, students did not acquire these skills at CPSLO. In fact, our experience and research indicates that while many universities and community colleges offer lower division courses that teach students about project control theory; they are not able to provide an educational experience where students can practice these skills. Therefore, in an effort to produce a multidisciplinary project-based learning experience, the faculty at CPSLO developed COINS - COnstruction INdustry Simulation - to reinforce several key learning objectives and to provide a valuable experience for students to work on projects that require the application and synthesis of material from a variety of courses and disciplines. Through the use of COINS, students are placed in a virtual environment to replicate, as nearly as possible, the working environment they will encounter after graduation. Students are exposed to exercises that are significantly different from typical homework assignments in conventional courses. Students are required to do independent research, to work collaboratively with students from different disciplines, and to use effective communication skills. Our experience is that the use of COINS engages students unlike any other teaching intervention and we expect that it will become a model for other civil and construction engineering programs

2 Development of COINS

COINS originated from a vision for enhancing the CCE curriculum articulated by Professors Hal Johnston and Jim Boland, Emeritus Faculty at CPSLO. The first version of COINS was known as Building Industry Game (BIG). The BIG simulation game focused on the commercial building sector of the construction industry. BIG had a built-in estimating and scheduling simulation and some accounting, which students used to emulate managing a commercial building contractor's role. The origins of BIG began with Glenn Sears, Professor Emeritus, of the University of New Mexico. Professors Johnston and Borland were granted permission from Professor Sears to write, modify, convert BIG to C++. Their idea was to transform BIG into a larger integrated construction industry simulation that incorporated more sectors of the construction industry in order to create a more robust simulation tool for students to apply project control fundamentals in a virtual environment. The name COnstruction INdustry Simulation (COINS) incorporates this vision. Using BIG as a template, COINS was developed into a web-based simulation written with a JAVA front-end and using a PostgreSQL database. The existing version of COINS was designed to enhance learning in the CCE curriculum and addresses learning objectives from the following bodies of knowledge: Project Planning, Project Procurement, Schedule Control, and Cost Control.

Using COINS students are placed in teams to manage a construction company. Their virtual company comes complete with dynamic financial statements, injury illness and prevention (IIP) programs, continuous improvement programs (CIP), an employee handbook business plan, personnel descriptions for all employees, company backlog of work if wanted history, etc. Using COINS, student teams make business decisions over time and the simulation provides a framework for company accounting, project management, procurement of work, etc.

COINS BIG/COINS was developed into a web based simulation written with a in JAVA front-end and using POSTGRESql database by the Cal Poly Faculty Multimedia Group under the direction of Professors Hal Johnston and Jim Borland. BIG/COINS was developed using open source software and continues to be open source in nature. The simulation uses a dynamic accounting engine, as a core to its many other parts and modules.

More than an estimating simulation, the developers goal was to produce a simulation that included bid strategy, project management, construction accounting, strategy, etc. Parameters that can be set by the instructor include the following:

- Method parameters for each activity type
- Size statistics for each job type
- Network dependencies for each job type
- Number of jobs available
- Material cost index
- Consulting costs
- Liquidated damages parameters
- Computer controlled contractor parameters
- Negotiation game parameters
- Personnel parameters
- Other miscellaneous game parameters

Future changes are planned for COINS simulation in effort to create a more robust simulation. These include:

- Equipment management
- Unit price bidding
- Unit price billing
- Equipment parameters
- Equipment as a cost item
- Dynamic depreciation
- Equipment feedback loop
- Personnel resumes and interview
- Case studies

Considering the market place has both private projects in which contracts are most frequently negotiated and non-bonded and public projects where contracts are typically bid and must be bonded. COINS contains both. After some success, a company may be put on select bidders lists and even move onto being considered for negotiated projects.

For the commercial building construction sector, each project is made up of nine activities, which are included in a projects schedule. These are:

- Excavation
- Foundation
- Basement
- Framing
- Closure
- Roofing
- Siding
- Finishing
- Mechanical, electrical, and plumbing

For the heavy civil construction sector, each project is made up of nine activities, which are included in a projects schedule. These are:

- Clear and grubbing
- Rough grading
- Excavation
- Underground utilities (water, sewer, storm drain)
- Concrete placing and finishing
- Backfilling and compaction
- Aggregate base placement and compaction
- Paving
- Finish grade

For each activity, there are five (5) different construction methods to select from; therefore, the cost and overall schedule is determined by the methods picked for each activity.

The following types of commercial building construction projects (public and private) are included in the simulation:

- Multi-family housing
- Educational facilities
- Hospitals and medical office buildings
- Commercial office buildings
- Industrial manufacturing facilities

The following types of heavy civil construction projects (public and private) are included in the simulation:

- Highways projects
- Bridges
- Residential site development
- Mass excavation
- Underground utilities

3 Overview of Game Play

COINS allows the game administrator (instructor) to place the player or team into a situation or incident that could require a quick short term solution or possibly a long term change in the company. Situations also take the form of cases that require ongoing management by the player or team over an extended period of time. The game can simulate the month-to-month problems, issues and decisions required to manage a construction company successfully.

Student teams are able to hire virtual staff as needed, deal with monthly problems, make choices, and experience the results of their decisions. During game play, participants gain experience and exposure to a cadre of real world scenarios, are provided with the opportunity to gain experience, learn from their mistakes, and to experience the totality of management required of the construction professional. Each team is given an equal amount of capital at the start of the game.

Time is represented as "periods," each period being two months of real-time. The period is advanced once or twice per week. Each period, new projects are available for bidding. With the increasing number of awarded contracts, companies must recruit other overhead personnel or the companies must pay large additional sums for the employment of external consultants.

Ultimately, the goal of the students during gameplay is to achieve the best possible project and company outcome by developing the ability to analyze situations, gather data and make strategic decisions between time, cost and quality - the essence of project management. Specific aspects of the game play is described below.

3.1 Personnel Management

The first order of business for game play is for students to form a multiple student team and create a company, develop a mission statement and core value statement. Student teams are given a username and password by instructor. Teams then register their team members. Each student team member plays the role of a construction firm. Teams are required to hire personnel, creating main office overhead, i.e. President, Marketing Director, Estimator, Student Intern, Scheduler, Accountant, etc. They are permitted to change personnel, as they need either for growth or other reasons.

3.2 Business Development and Work Procurement

Teams are given quantities, expected production, and costs for each activity on every job available for bid. The student teams must decide which jobs to bid on, select a method for each activity, determine their direct cost, then settle on a bid price by adding jobsite indirect, contingency, mark-ups and desired profit. The simulation is also able to internally generate bids on every job to keep the players cost estimates and bids within reason. As the period advances, the computer evaluates bids on each project and awards contracts to successful bidders. Players evaluate the results and attempt to interpret their competitor’s strategies as the game progresses. Construction bids are rejected if they fall below a minimum amount (calculated by the computer). In order to bid on a project, student teams must have cash-on-hand, along with other strong financial indicators. These items help the computer set individual project size limits for bonding purposes, which is at least 5% of their bid amount. Companies may not be overloaded with too many other jobs and therefore, all companies have a work in progress bonding limit that may not be exceeded also.

3.3 Project Management

Players must monitor their financial position as work progresses, and bill for their progress payments. Also, teams must create strategies to improve their bonding limits. A record of successful projects creates an opportunity to obtain negotiated work. At the end of every period, each team receives a:

- Progress Report
- Complete Dynamic Financial Report
- Analysis Report of the work accomplished, and
- financial result to date.

The amount of work completed during a period depends on: the production rate for the work packages selected on each activity and the uncertainty factors, including - weather conditions, labor availability, and fluctuating cost of materials.

The end-of-period financial reports show expenses incurred for:

- Direct construction costs
- Bidding costs
- Consulting services
- Liquidated damages, and
- Interest on borrowed money

Student teams must monitor their financial position as work progresses, and bill for their progress payments. Also, teams must create strategies to improve their bonding limits. A record of successful projects creates an opportunity to obtain negotiated work. Changes in company's financial position will change ratios and are also logged along with changes to the company's appraisal metrics:

- Financial liquidity
- Financial success
- Responsibility
- Pace
- Ethics
- Name recognition

As gameplay progresses, teams have the following options:

- Pay a consulting fee to receive information on weather forecasts, material prices, labor and material availability, and market projections for future periods.
- Apply for loans.
- Make a change and specify a different method for the following periods.
- Use overtime to speed up certain activities (greatly increasing the labor costs).

Each company must evaluate the projects they have, changing methods if needed, and at the very least, bill for the work they completed during that period. Billing affects cash flow and cash position on the balance sheet and potentially the company's bonding capacity.

A financial report shows the final total worth of the firm in either case. Maximization of profit & strong financial condition are main objectives, but additional emphasis can be placed on the company appraisal metrics. At the conclusion of the gameplay, the instructor can either:

- have the simulation forecast the expected results of any on-going projects or
- use the actual results at that time.

4 Using of COINS in Education

At Cal Poly, COINS has been used in several courses including:

- Professional Practice
- Construction Estimating
- Construction Accounting
- Management of the Construction Firm
- Business Practices

During the 2005/2006 academic year, the simulation was used for regional competition between multiple universities in the Associated Schools of Construction Regional 6 and 7 Student Competition.

Most recently, in November 2009, universities from the Czech Technical University (CTU) - Prague, Czech Republic, Auburn University – Alabama, California State University, Fresno - California, Illinois State University - Illinois, Boise State University - Idaho, Western Carolina University - North Carolina, and Washington State University – Washington, participated in an international competition. Competition Results were evaluated in three categories: Highest Retained Earning - received the highest profit, Highest Appraisal Metrics - the best valuation metrics and third, Most Awarded Projects - the company with the most awarded projects.

5 Conclusions

The simulation has a built-in grading module that can be used to obtain statistic on the various companies for comparison or to use in the classroom for grading the simulation. Each faculty can have their own method of grading. The following on faculty used a criteria for assessing participation and student learning:

- Number of jobs bid
- Minus the jobs rejected (i.e., not enough bonding capacity, substantially low cost estimate, etc.)
- Number of times the number jobs you are the lowest cost
- Number of times the company retained earnings
- Company's appraisal metrics

Using the seven principles of good practice as an evaluation metric, the COINS system performs well. It encourages contact between students and faculty by encouraging frequent student-faculty contact in and out of classes, which is an important factor in student motivation and involvement. Faculty concern helps students get through rough times and keep on working. Knowing a few faculty members well enhances students' intellectual commitment and encourages them to think about their own values and future plans. It develops reciprocity and cooperation among students. When using the COINS systems, learning is enhanced when it is more like a team effort than a solo race. Good learning, like good work, is collaborative and social, not competitive and isolated. Working with others often increases involvement in learning. Sharing one's own ideas and responding to others' reactions sharpens thinking and deepens understanding. COINS encourages active learning. Learning is not a spectator sport. Students do not learn much just by sitting in classes listening to teachers, memorizing pre-packaged assignments, and spitting out answers. They must talk about what they are learning, write about it, relate it to past experiences and apply it to their daily lives. They must make what they learn part of themselves.

COINS gives prompt feedback. Knowing what you know and don't know focuses learning. Students need appropriate feedback on performance to benefit from courses. When getting started, students need help in assessing existing knowledge and competence. In classes, students need frequent opportunities to perform and receive suggestions for improvement. At various points during college, and at the end, students need chances to reflect on what they have learned, what they still need to know, and how to assess themselves. The use of COINS emphasizes time on task. The time plus energy equals learning. There is no substitute for time on task. Learning to use one's time well is critical for students and professionals alike. Students need help in learning effective time management. Allocating realistic amounts of time means effective learning for students and effective teaching for faculty. How an institution defines time expectations for students, faculty, administrators, and other professional staff can establish the basis of high performance for all. Use of COINS communicates high expectations. Expect more and you will get more. High expectations are important for everyone -- for the poorly prepared, for those unwilling to exert themselves, and for the bright and well motivated. Expecting students to perform well becomes a self-fulfilling prophecy when teachers and institutions hold high expectations for themselves and make extra efforts. COINS respects diverse talents and ways of learning. There are many roads to learning. People bring different talents and styles of learning to college. Brilliant students in the seminar room may be all thumbs in the lab or art studio. Students rich in hands-on experience may not do so well with theory. Students need the opportunity to show their talents and learn in ways that work for them. Then they can be pushed to learn in new ways that do not come so easily.

6 Conclusions

Some early recommendations during the first stage the simulations development included: creating learning objectives, creating an outline or direction, and to create modules. Even the simple simulations generally cannot be completed during the first development stage. Having a framework of different modules and what each might accomplish is critical to success and the development process. Most times having a group to develop this direction and the different modules that might be needed is a key to creating complex and broad simulations.

To assist in the development of COINS, the developers have developed an Industry Advisory Board (IAB) from the construction industry as well as a working group of educators to continue the development and ideas for changes. Because of the idea of module development COINS can turn on and off some of its modules, making it a better fit in different classes. For example, estimating can be turned to an automatic mode which in a construction accounting class helps the student focus on accounting and not on the estimating itself which can be very time consuming and complex. Periods can move much quicker giving the students more accounting to analyze and in a shorter time in which they can see the changes that occur within a company without being bogged down in the estimating/procurement of work. Billing can be turned on to auto mode and additional projects can be added to each team to create additional project or backlog. The game play between commercial and heavy/civil construction is also modulized so a faculty can play only commercial, heavy/civil or both can be played in one game. Future additions are also planned as modules, i.e. personnel additions, case studies, and wide use of equipment management.

7 References

[1] Aldrich, 2005 - Learning by Doing: A Comprehensive Guide to Simulations, Computer Games, and Pedagogy in e-Learning and Other Educational Experiences by Clark Aldrich. (John Wiley & Sons, 2005)

[2] Beck - Got Game by John C. Beck, Mitchell Wade

[3] Kaye, 2002 - Flash MX for Interactive Simulation: How to Construct & Use Device Simulations by Jonathan Kaye, PhD and David Castillo (Delmar Learning, 2002) Companion CD-ROM with full source code.

[4] Kraft, 1994 – "The theory of experiential education, 2nd ed.". Boulder, CO: Association of Experiential Education. Richard Kraft and Mitchell Sakofs (Eds.) (1994

[5] Kolb 1984 - David Kolb (1984). Experiential learning: Experience as the source of learning and development. Englewood Cliffs, NJ: Prentice-Hall.

[6] Koster - A Theory of Fun for Game Design by Ralph Koster

[7] Lwein, 1951 - Field theory in social science. New York: Harper Collins Kurt Lewin (1951).

[8] Pfeiffer, 2005 - Engaging Learning: Designing e-Learning Simulation Games by Clark N. Quinn, forward by Marcia Conner. (Pfeiffer, 2005)

[9] Prensky, 2004 - Digital Game-Based Learning by Marc Prensky (McGraw-Hill, 2004)

[10] Senge, 1990 - Peter Senge (1990). John Dewey in Experience and Education in a chapter entitled, "The Means and Goals of Education."

[11] Schrage – “Serious Play” Michael Schrage's

[12] Shank - Lessons in Learning, e-Learning, and Training: Perspectives and Guidance for the Enlightened Trainer by Roger C. Schank

[13] Schank, 1994 - What we learn when we learn by doing. Technical Report 60. Evanston, IL: Northwestern University's Institute for the Learning Sciences. Roger C. Schank (1994, October).

[14] Watkins - e-Learning Activities: Making Online Learning Interactive by Ryan Watkins

[15] Whitney, 2004 - "Performance-Based Simulations: Customizable Tool" by Kellye Whitney. Chief Learning Officer Magazine, October 2004

Toward Exhaustive Teaching Of Relational Database Indexes

Massimo MACERA[1], **Paolo ROCCHI** [2]
[1]IBM, via Sciangai 53 Rome 00144, Italy
[2] LUISS University, via Alberoni 7 Rome , Italy

Abstract *- There are several indexing techniques that can optimize system performances. The choice of the right index in a relational database can improve the performances by up to 80%. Unfortunately the illustrations of the various techniques are scattered over a number of texts and manuals, and the courses for database designers are often somewhat incomplete. The present paper shows a didactical experience which tackles this kind of limitations. In particular we have prepared a textbook that includes twenty-one different index formats, it discusses advantages and disadvantages of each indexing technique, and has been positively validated during advanced courses on relational database design.*

Keywords: Database index design, DB Indexing

1 Introduction

Ricardo and others examine challenging aspects of modern education on database (DB) systems [1] in the light of CC2001 recommendations [2]. These authors are inclined to focus on introductory lessons [3], but significant defects emerge even in advanced database courses which appear insufficient to satisfy the requirements of professional practice. Some educational weaknesses appear evident respect to working environment. As first, a number of courses on database systems concentrate on abstract principles and theoretical topics. Relatively little mathematics turns out to be important for software engineers in practice and it tends to be forgotten. As second, teachers tend to use rather simple, well-analyzed and well-understood datasets as both examples and project data in their courses [4]. Topics covered in undergraduate and graduate courses are often simplified because it is too difficult to teach a topic with all the complexity of the real world [5]. At the other hand young people, trained in simpler domains, are not adequately prepared for the actual assignments they will be facing in their future works [6]. Students will find much more complex database systems when they will take a job position and will strive to implement and to handle software applications.

Junior professionals normally fill the gap by training on the job (1) or by the assistance of a mentor (2) or even by the attendance of a professional course (3). However these three educational approaches normally cover narrow areas of specialization and learners strive for exhaustive preparation.

We are fully aware of the negative consequences coming from simplified lessons on computer science (CS) and of limited training that technicians find in the working environment. We have designed various courses that tend to improve the knowledge of students for successful integration into employment. A course on CS has been amply described in [7]. Special lessons on the conceptual and logical DB design have been detailed in [8]. As third, the present paper deals with relational databases index design [9].

2 Teaching indexes

Briefly we remind that an index is an ordered set of values which serves a single database table, in particular an index points to the rows of the intended database table. An index indicates the position of data inserted in one or more columns of the database table (Figure 1).

Paris	7-2, 8-1, 9-5
Dublin	2-1, 2-3
London	4-5, 5-4
Berlin	6-7
...	...
Moscow	3-1, 6-2, 8-4

Figure 1

Indexes can be created by using one or more columns of a database table, providing the basis for rapid and efficient access to records. In fact an index has an order and improves the speed of data retrieval operations on a database. An index is an object stored apart from the database tables, and the disk space required to store the index must be less than that required by the addressed table.

Dozens of database indexing methods have been devised but the majority of modern writers tend to filter this broad matter. Authors show inclination toward a restricted set of solutions. For instance [10] reports three basic methods: single column indexes, composite indexes and a unique indexes; [11] describes b-tree indexes, bitmap indexes, hash table indexes, clustered and non-clustered indexes; [12] examines the methods just listed and adds up composite indexes, covering sparse indexes and dense indexes.

Unit 1.	Generalities.
Unit 2.	Different Index Formats and Reasons for Using Indexes.
Unit 3.	Index Enhancements in DB2 v9.
Unit 4.	Indexes for Joins.
Unit 5.	Index Versioning.
Unit 6.	Less-Known Index Features.
Unit 7.	Operational Utilities for Indexes.
Unit 8.	Index Cost.
Unit 9.	Indexes for Star Schema.

Table 1 – Table of Contents

A second group of writers – usually highly specialized writers – tend to dedicate a manual to a sole index technique so that the indexing contents spread out in several handbooks. Both simplified explanations and illustrations scattered over a considerable number of hand books cannot help specialists who are called to design and optimize large databases.

We have conducted a two stage research in order to assist technicians who are presently working in the DB domain or will work in the next future. Firstly we gathered the various index formats treated in the large set of technical manuals. In particular we examined the manuals published by the major producers of DB systems (IBM, Oracle etc). Secondly we systematized the collected material and prepared a text-book for educational purposes [13]. We arranged a course whose lessons follow the same order of the book units (Table 1). Hereafter the terms 'unit' and 'lesson' are synonymous.

3 Comments on Lessons

After the introduction, the second lesson presents different index formats. Lessons from 3 to 9 expand on various technical topics dealing with indexing techniques. The course explains twenty-one kinds (or formats) of indexes which exhaust current technologies at the best of our knowledge. The Appendix exhibits all the index formats.

Actually an index is defined by using the SQL statement Create Index, and a parameter of this statement frequently equals to the index format label.

We have cataloged the index formats into ten groups (see Appendix) so to help a student understand similarities and differences and to reduce the memory load. Each group of indexes has specific technical features briefly commented aside. A group contains one or more formats; the index formats belonging to a group can be mutually exclusive (e.g. Padded and Not-Padded) or may overlap in a way (e.g. Clustering and Clustered). Students learn to combine the various kinds of indexes so that one can create - for example - an index which is simultaneously Unique, Clustered, Multiple-Columns, Not-Padded, Secondary, Not-Partitioned. Lessons from 3 to 7 dissect pros and cons deriving from each method; this discussion is crucial for an attendee who aims at grasping the possibilities and the limits of a practical solution.

We teach the students to determine the right indexes soon after they will create a database table and to check the designed indexes against various SQL queries for that table. This is because an index can be used for many different reasons and may be influenced by an assortment of functions. In broad strokes an index can be used for:

- Enforcing uniqueness of primary keys and unique constraints.
- Improving performance in data access, through different index probing, such as Index-Only, index Matching, index Screening.
- Avoiding Sorts.

In principle the access to data is faster with an index than with browsing all the table data. For example, to speed up data retrieval against a predicate such as `Where Capital = 'Paris'` one can create an index on the ‘Capital’ column of the ‘Nations’ table to easily locate a specific capital and avoid reading through each row of the table 'Nations' (or scanning the whole table). An index can improve performances in several ways which we comment on during the course and briefly sum up as follows:

1) Indexes can be used to reduce the number of get pages and input/output operations.

2) Applying an SQL predicate to the index may be more efficient than applying the same predicate to the data. The use of an index can reduce the number of data pages and rows scanned. If a row does not qualify the predicates in the index, there is no need to retrieve the data page. This applies to SQL statements that access only a few rows of data in a table, but also to processes that sequentially access a large quantity of data in the table, and to single table access, as well as multiple table access.

3) Reducing the number of data pages accessed is particularly interesting if the access is via an index that has a low cluster ratio (i.e. a low order of data row). A low cluster ratio could increase the probability of synchronous input/output operations and therefore the cost of the statement.

4) The best option is to make the predicate indexable. However, if this is not possible, one can still benefit from index screening or index-on expression.

5) Using an index can completely eliminate access to data pages when index-only access is possible.

6) Because data access is completely eliminated when using index-only access, such an index creates an 'alternative' clustering sequence for those queries.

7) An index can also eliminate a Sort (due to Order By, Group By, Distinct, Join) if the requested order matches with the index columns. A sort can be eliminated by scanning an index, no matter the direction (forward index scan or backward index scan). Please note that any index can avoid Sorts, not only clustered indexes.

Indexes have a cost of maintenance and the lessons address the students to evaluate the access speed improvement that an index provides against the cost of maintaining that index. Listeners are carefully invited to consider the price of each technical improvement. We summarize a number of reasons explained in the last lesson of the course.

a - Indexes require storage space. Padded indexes require more space than not padded indexes for long index keys. For short index keys, not padded indexes can take more space.

b - Each index requires an index space and a data set, or as many data sets as the number of data partitions if the index is partitioned, and operating system restrictions exist on the number of open data sets.

c - Indexes must be changed to reflect every insert, update or delete operation on the base table. If an update operation modifies a column placed in the index, then also the index must be changed. The time required by these operations increases accordingly to the number of indexes.

d - Indexes can be built automatically when loading data, but this takes time. They must be recovered or rebuilt if the underlying table space is recovered. These operations might be time-consuming.

Appropriate indexes speed up the data retrieval considerably, but in counterpart the operations which need to maintain the index, such as Update/Insert/Delete statement, require extra work when an index is involved. In fact not only the intended row has to be updated/inserted/deleted into the table, but it also has to be updated/inserted/removed in all the indexes that are defined on that table. The lesson concludes that the cost of maintaining multiple column indexes can sometimes be higher than the costs of maintaining the data.

4 Didactical Conclusions

The present educational project produced an advanced course on DB design, and eleven editions of this course have been held in the United States, Germany and Italy on request of organizations and individuals. Globally one hundred students attended the lessons; all were male. The attendees can be subdivided into junior technicians (18-22 year old) and practitioners at various levels of specialization in the database domain; the latter made the majority group (over 95%).

Lessons have been illustrated through oral teaching and labs which globally last eighteen hours and cover three working days. The didactical material – the text-book and the slides – have been arranged in the following manner. Every chapter of the book encompasses a fixed number of topics and a slide summarizes each topic. A topic begins with a new page in the book that exhibits the relative slide on top. The subsequent pages illustrate the intended topic in any detail and are verbally commented by the teacher. This accurate correspondence amongst a projected image, the pages of the book and the oral lessons has been appreciated by the attendees.

We do not have formally evaluated this course except for the satisfaction factor (SF) whose mean lies over 98 points in eleven editions. A student was allowed to express his satisfaction using five values: 100 (very-satisfied), 75, 50, 25 and 0 (absolutely-unsatisfied). In addition students manifested their oral feedback on the good quality of the material and the educational method which was easy to follow. Experienced participants provided the most significant return of information about this course as they used the lesson contents to solve practical issues they faced in the living environment in advance of the lessons. The answers to unresolved problems emerged as the most immediate and satisfactory fall out of this course.

These researches aimed at out-pacing current restrict visions on the indexing techniques by providing the systematic account of the index formats. This was the main didactical purpose of the present work. We summed up numerous manuals which turn out to be impracticable or too expensive for a practitioner.

The personal working experience of the authors in the DB domain suggested how to improve the didactical material which can be exploited in a variety of educational environments. For example a teacher can reshape and ameliorate a graduate course on DB using the Appendix. This easy summary makes an expert-to-be aware of all the methods in use to optimize a database design.

5 Appendix

Group	Index Format	Comments
1	*Single-Column, Multiple-Column.*	An index can be created on one column only or on many columns simultaneously (max 64).
2	*Unique,* *Unique Where Not Null,* *Duplicate.*	Unique indexes ensure that a value never repeats in the table columns. For example, the 'Nations' table does not allow duplicate 'Capital' attributes. Creating a unique index prevents duplicate values. This restriction can be applied to generic values (including Nulls) or otherwise to values that are Not Null. Duplicate indexes allow repetition of the same index-key value into the table.
3	*Clustering,* *Clustered,* *Not-Cluster.*	A cluster index suggests the physical order in which you would like the rows to be stored in the data table. Since rows can be stored only in one physical sequence, a practitioner can define at most one clustering index per table. Both unique and duplicate indexes can be defined as cluster. When you just create an index with the 'Cluster' option, that index is called Clustering index. When the table rows are perfectly in order according to this index, the index is also a Clustered index.
4	*Padded,* *Not-Padded.*	This feature does not pertain to indexes with fixed length columns. These clauses suggest how varying-length string columns are to be stored in the index. A padded index has varying-length string columns filled with blanks characters to their maximum length. In a not-padded index the varying length string columns are not padded to their maximum length in the index. Instead the length information for a varying-length column is stored with the key.
5	*Ascending,* *Descending,* *Random.*	This is the direction of the keys sequence inside the index, as any index is always an ordered set of values even when the option Random is specified.
6	*Partitioning, Partitioned, Not-Partitioned.*	An index is partitioned when the index itself is subdivided according to the partitioning scheme of the underlying data. An index is considered a partitioning index if its leftmost key columns match the columns specified in the partitioning key of the table. The leftmost index key columns are in the same sequence and have the same ascending or descending attributes of the partitioning key of the table.
7	*Primary,* *Secondary.*	A primary index is created for the primary key. Any index on a partitioned table space whose leftmost key columns do not coincide with the partitioning columns of the table is a secondary index. A secondary index can be partitioned or not. A partitioned secondary index is spread on different physical data sets, one for each index partition. Its leftmost columns are different from the columns which partition the table. A non-partitioned secondary index is not spread on different partitions and its columns are different from the partitioning columns of the table.
8	*Indexes on Expressions*	An index on expression is created on column expressions and not only just on simple columns. Indexes on expression can be used to retrieve data whose predicates are expressed by using a general expression. To exemplify, these indexes are good for solving predicates such as "where col1+col2 > 100". In contrast to simple indexes whose key values are the ones contained on simple table columns, the key values of an index on expressions are not exactly the same as values in the table columns. Instead, these index key values are the result of the specified expressions. Also an index on expression can have at most 64 columns.
9	*B-tree*	The structure of an index is a balanced tree (B-tree), automatically maintained by the database management system. At the bottom of the tree are the leaf pages of the index. Each leaf page contains a number of index entries consisting of the index key itself and the pointers, known as 'record identifiers', which one uses to locate the indexed data rows. Each entry in the intermediate nonleaf index page identifies the highest key of a dependent leaf page along with a pointer to the leaf page's location. At the top of the tree, a single page, called 'root page', provides the initial entry point into the index tree structure.
10	*Bitmap*	These indexes are appropriate for columns which have a small number of distinct values, e.g., male and female. A bitmap index is smaller than a B-tree index and stores only the row-id and a series of bits. In a bitmap index, if a bit is set, it means that a row in the corresponding row-id contains a key value. Also Bitmap indexes can be very helpful in data warehousing because in the depicted situation they can be quite fast when you are only selecting data.

References

[1] E.S. Adams, M. Granger, Don Goelman, C. Ricardo - Managing the Introductory Database Course: What Goes In and What Comes Out? - *SIGCSE Bulletin* Volume 36 Issue 1, March (2004)

[2] Computing Curricula 2011 Retrieved June, 2011 from http://www.computer.org/education/cc2001/.

[3] M. Robbert, C. Ricardo - Trends in the Evolution of the Database Curriculum - *ITiCSE* (2003).

[4] J.P. Wagner, E. Shoop, J.V. Carlis - Using Scientific Data to Teach a Database Systems Course - *Proc. ACM SIGSE*, pp. 224-228, (2003).

[5] J.D. Wilson - Problems Teaching Database Design with Information Complexity to Information Systems Undergraduates - *Proc. ACM SIGCSE*, 18 (1), pp. 2-7, (1986).

[6] S.D. Urban, S.W. Dietrich - Advanced Database Concepts for Undergraduates: Experience with Teaching a Second Course - *Proc. ACM SIGCSE,* pp. 357-361, (2001).

[7] P. Rocchi - Lectures on CS Taught to Introduce Students with Different Background - *Proc. Informatics Education Europe III* (IEEIII), pp. 115-124, (2008).

[8] P.Rocchi – A Method to Validate the Relational Database Design – in *Relational Databases and Open Source Software Developments*, J.R. Taylor (ed), Nova Science, pp. 105-118, (2010).

[9] S. Chaudhuri, M. Datar, V.R. Narasayya - Index Selection for Databases: A Hardness Study and a Principled Heuristic Solution - *IEEE Trans. Knowledge and Data Engineering*, 16(11), pp. 1313-1323, (2004).

[10] R. Plew, R. Stephens - *Teach Yourself SQL in 24 Hours* - Sams Publishing (2002).

[11] P. O'Neil, E. O'Neil - *Database: Principles, Programming, and Performance* - Morgan Kaufmann (2000).

[12] S.S. Lightstone, T.J. Teorey, T. Nadeau - *Physical Database Design: The Database Professional's Guide to Exploiting Indexes, Views, Storage, and More* - Morgan Kaufmann (2007).

[13] M. Macera - *DB2 for z/OS Index Modeling and Design* - IBM Corp. (2009).

Improving Efficiency in Interactive Educational Software

A. Jesse Keogh[1], B. Medha Sarkar[1], C. James William[2], and D. Michael Fletcher[2]
[1]Dept. of Computer Science, Middle Tennessee State University, Murfreesboro, TN, USA
[2]Dept. of History, Middle Tennessee State University, Murfreesboro, TN, USA

Abstract - *There are many ways to measure efficiency when it comes to educational software. Primarily, as a computer scientist, you must ensure the software itself is efficient in software terms. This means cleaning up code, enhancing usability, and adding features that improve the overall quality of the software. A second measure of efficiency, equally as important as the first, is how efficiently the software can actually educate a student. That is, measuring how much educational value can be attained by a student who uses the software. The primary goal is to create an effective product that uses efficient methods. We were able to spend a good amount of time working with such a system, Digital Democracy, and have improved system inadequacies tremendously by evaluating efficiency and effectiveness at each step of the development cycle.*

Keywords: efficiency, education, history, value, interaction

1 Introduction

Interactive systems are quickly becoming a popular way of teaching a wide range of materials. It is becoming more common to see video games that are geared towards education instead of solely recreation. More importantly, it is becoming more common to accept the idea that these simulations are actually able to educate students effectively. James Paul Gee of the University of Wisconsin-Madison states that "… schools, workplaces, families, and academic researchers have a lot to learn about learning from good computer and video games. Such games incorporate a whole set of fundamentally sound learning principles, principles that can be used in other settings, for example in teaching science in schools. In fact, the learning principles that good games incorporate are all strongly supported by contemporary research in cognitive science, the science that studies human thinking and learning through laboratory research, studies of the brain, and research at actual learning sites like classrooms and workplaces [4]." We wanted to take this idea and create an interactive learning system that could teach any subject matter effectively. The first subject we chose to pursue was history. Our starting point for this attempt would be to create a fully functional interactive educational system that is able to teach content related to the American Civil Rights Movement from a variety of perspectives.

Creating a successful interactive educational system is a daunting task. As with any other project, a primary goal is to design an efficient system. With systems that assist in the education of others, efficiency is measured in a variety of ways. As a computer scientist, it is easy to focus on the code. Making sure you have a speedy project that is written properly is always a major concern. Perhaps the next most important thing would be a sleek user-interface. What may be less apparent is the value that can be attained by the user. That is, how efficiently does this software assist in the education of a student? When working with groups that don't understand, and probably care much less about code efficiency, this last criterion quickly becomes the most important measure of efficiency to consider.

Working with such a project can be extremely rewarding. Not only are you responsible for being a good programmer, but you are responsible for creating your system in such a way that it can seamlessly teach any subject to any student. Small things that are often overlooked quickly become the most important aspects of the system. Some ideas were as simple as displaying supplemental reading material in a new window, instead of cluttering user space. While many of these small changes require little coding, they exponentially increase the user's experience of the software.

We have had close experience working with such a system over the past two years. The project, named Digital Democracy, is a direct implementation of this ideology. The idea is to create a fast, user-friendly, value-conscious software framework that will be able to assist in the education of students over a variety of subjects. Initial ideas varied from low-level console applications to virtual worlds. A smart choice was right in the middle, a web framework that can be accessed anywhere, anytime. The first subject that would be attempted would be the American Civil Rights Movement. We collaborated with members of the MTSU (Middle Tennessee State University) History Department to make this implementation possible.

Prior to the current implementation of the system, two partial implementations had been attempted. Both were filled with holes and left the user more concerned with glitches than historical content. We were able to take control with a variety of code improvements, a more functional and user-friendly UI (user interface), and a working framework that is ready to be "programmer-free." That is, a fully functional framework that only waits for new historical content to be added, not additional software updates.

Throughout the following sections, all of the challenges and goals of creating such a system will be addressed. Understanding and explaining the initial specification of the project, evaluating previous work on the project, discussing improvements and changes made, and analyzing the improved final implementation.

2 Project background

We took the idea of a simulated learning environment and decided to design a framework that will allow any subject matter to be taught in a fun, functional, interactive learning environment. The initial project was designed to be a console application that would be distributed by compact disk to interested groups. The expected audience was middle school classrooms of approximately 30 students [1]. While the target audience has remained the same, we decided on a web-based system as opposed to a distributable console application.

The initial design idea was to write a virtual software simulation that allows a student to act as an elected official that makes decisions in the American legislative system. Our test project within this field would be using content surrounding the Civil Rights Movement in America with a focus on the events that occurred in that time period in Tennessee. With this in mind, a variety of character roles from Senator Albert Gore Sr. to just a standard Tennessee citizen were created to provide the user an interesting perspective on the material. As the simulation progresses, the student receives information from archived documents provided by the MTSU History Department. Throughout the simulation students have the opportunity to provide responses, reports, and feedback over the content. This feedback can also be in the form of a decision which will allow the user to take one of many paths through the content, perhaps taking on a persona to go with their character choice. The simulation ends with the passage or defeat of a bill based on the choices the user has made. We hope that this use of our framework can serve as a model for democracy education in Tennessee and the United States as a whole [2].

Over the course of two years, the first two attempts of this system were created and analyzed. The first being a very basic web site designed for simply displaying content. The design worked fairly well, but lacked many of the necessary elements to create a fully functional framework. The second implementation had a much better UI and included a content management system, but was quite faulty in certain areas. As these two implementations struggled to impress our target audience, we decided it was time to take a new approach. A collaboration of the ideas, simplicity and functionality, led us to our current implementation, which will be explained in detail in Section 4.

3 Early implementation

3.1 Version one

The first implementation of our system was created for a simple presentation of the historical content. The system was designed using HTML (Hypertext Markup Language) and PHP (PHP: Hypertext Protocol) to design a web site that flowed from page to page telling one of many stories. The initial design had three basic modes: beginner, intermediate, and advanced. Each of these would potentially have harder material and questions throughout the user experience. Although the interface allowed for these three modes, only the Basic mode was completed. Once your mode was selected, you could choose between various roles to act as during the story. These roles included various Tennessean characters that would have had influence during the time of the Civil Rights Movement. A few of the proposed roles were a standard TN citizen, a lobbyist paid to influence Senator Gore Sr., and Senator Gore Sr. himself. The UI allowed for all of these character choices, but only the TN Citizen was completed. Once your role was chosen, you were led through the storyline of a Tennessee citizen as the Civil Rights Movement took place within Tennessee and America as a whole.

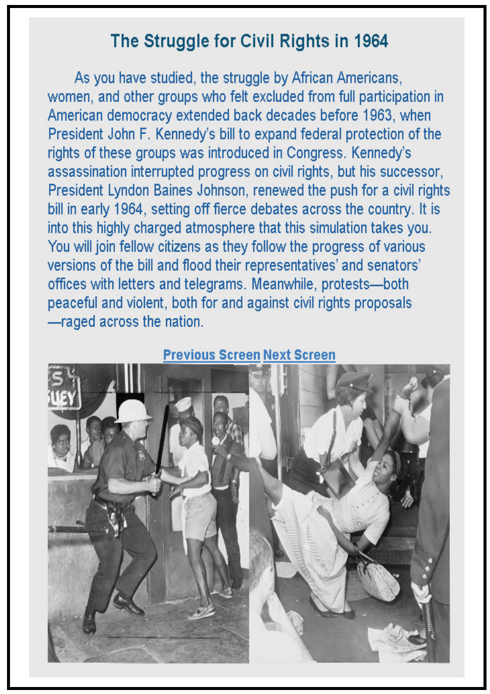


Figure 1 Version One Interface

On each page the student sees a newspaper article, legal material from the time, further description of such materials, or an input text box for user feedback. These text boxes could be used for students to write letters to legal representatives, acting as their character, or to write reports based on what they had learned so far. Database support was limited and many of the supplementary materials were missing. Although this software was lacking many key features, it was a great step in the right direction as the project was getting started.

3.2 Version two

The second implementation of the system was a much bigger stride towards our goal. While we still used a basic web site structure to display content, we were able to utilize a variety of great functions offered by the jQuery library.

jQuery is a free, cross-platform JavaScript library that greatly enhances the client-side functionality available when writing web content. This basically allowed us to use dynamic functionality to load content as the story was being told. This helped to eliminate the slide-show feel of the previous system.

Functional selection of roles and animated transitions between pages were the highlight of the user-experience enhancements. The user now had a sleek looking layout with working buttons and choices that affected the traversal through the content. It was possible to choose from a variety of paths as the storyline progressed. The user was presented with a list of seven character roles to act as, but once again only the TN Citizen was properly implemented. Although the interface was much nicer, we were still lacking some of the key functionality that we had initially designed.

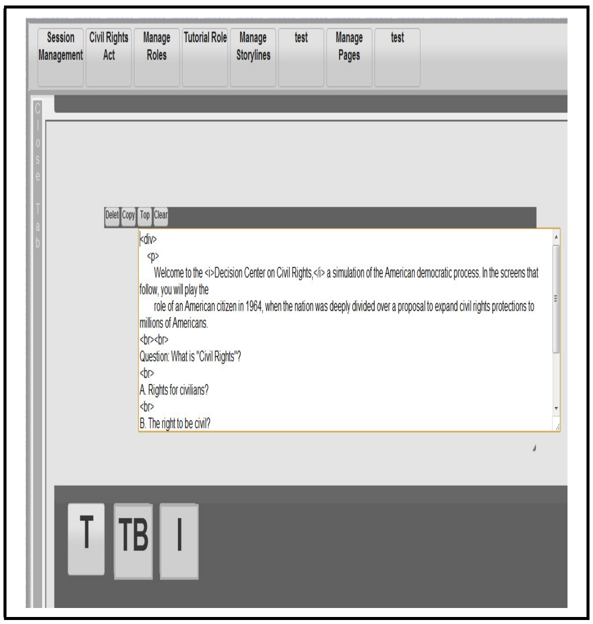


Figure 2 CMS: Edit Page

The biggest improvement by far was the CMS (Content Management System). In Version One a new document was required for every page. We now had a working framework to dynamically add new content and storyline material. This framework broke down into four hierarchical modules: Sessions, Roles, Storylines, and Pages. The current Session would be the broadest representation of the type of user interacting with the system. The main Sessions implemented were Administrator and User. Where the Admin would see content management modules, the User would simply see the content being presented. As an Administrator, you had access to the underlying modules. Each Role would be a representation of a character and would contain a Storyline for that character. Each Storyline would be a series of Pages related to the content. All of the Role modules contained the Civil Rights Act Storyline, each being modified with different individual Pages to tell that version of the story. Within each Page was a Toolbox that allowed for quick additions of images, text boxes, or paragraph-like text areas to the page. Although the feature was dynamic and impressive, the back-end code needed improvement. Editing and adding new objects in any of the modules was extremely simple, only requiring a few clicks and the typed description of the content. Another great feature was the ability to copy all of the content and formatting from any other Page to the current Page. This was extremely useful in making small changes within Storylines that were extremely similar to one another.

Although the underlying framework was wonderfully designed, the implementation was quite flawed. The system was slow and lacked standardization. These issues left a promising system unsatisfactory in terms of truly being an efficient educational system.

3.3 Implementation issues

The various problems with each of the systems left us struggling to find a proper starting place for Version Three. We had one system that ran quickly but the UI was bleak and it lacked the necessary features and content to keep the user interested. The lack of database support couldn't allow the instructor to evaluate each student's progress. Our second attempt gave us a much nicer UI and a verbose content creation system, but was slow and full of glitches that hindered the user's ability to focus on the material instead of the system itself. Wanting the software to be virtually transparent from the content, we were not satisfied with the results of Version Two.

4 Current work

4.1 Version three

The current version of the software has been in development since Spring 2011. After evaluating Versions One and Two, it was decided that a combination of the objectives of each should be combined into one final system. The aspects of Version One that needed to stay intact were simplicity of presentation, the speed of the software, and the standardized database support. The aspects of Version Two that remained were the framework for the content management system (adding, updating, and deleting module content), and the framework for transitioning between content pages. We were able to reconstruct many of these features to combine them into a completely efficient and effective system.

An interesting feature of the final product is that the system is actually split in to two sub-systems that work together. The content creation system from Version Two retained the hierarchical format of before with Session, Role, Storyline, and Page modules. The user still sees a familiar content layout when switching from Version Two to the current version. The database support framework was transferred from Version One and then redesigned using simple HTML and PHP for the formatting and MySQL for the querying language. The combination of these two designs resulted in a clean interface with fast underlying functionality. The usual situation when using the system is that you have a sleek design to look at filled with relevant content, and you have hyperlinks that can pop-up text boxes for database interaction or supplemental material to enhance the learning experience. This simple enhancement allows the user to maintain the idea that they are in the original interactive environment while offering new features that can behave independently, which allows for more efficient code to be written within both sub-systems.

4.2 Improvements

Throughout the course of the current development cycle we have strived to produce a well-written and complete framework for content management and effective presentation of educational materials. We were able to take the previous systems, which each only had one functional Role, and expand to have eight fully implemented Roles related to the Civil Rights Act. These Roles are the Influential Lobbyist, Newspaper Reporter, Political Aide to Senator Gore Sr., Senator Albert Gore Sr. himself, a TN Citizen in favor of the bill, TN Citizen opposed to the bill, TN Citizen undecided about the bill, and a general U.S. Senator character. Hundreds of new supplemental documents were added throughout these Storylines as well. The new and robust system was now full of valuable content.

As the system was plagued with bugs in the previous versions, we conquered many critical bugs in the process of recreating the system into what is now the current version, Version Three. Some of the most critical bugs are listed in this section. The initial system had a fixed size for all content displayed within the user and administrator interfaces. This often resulted in oversized windows within the monitor space that led to clipping of the content on the web page. A new dynamic size adjustment function allowed for us to display the same material regardless of the limitations of the window size. Each UI now takes up roughly 80% of the total window size, regardless of what that size may be. This had no negative effects on data presentation. Another UI related bug was that there was no functionality that allowed the user to view supplemental material at the same time that he or she was writing a report on the material. A simple addition to our hyperlink template allowed the user to open all supplemental material in a separate window. We implemented the same functionality for text boxes that interact with the database. This allowed the user to have control over what he or she needed to view at any given time without redirection. This new organization greatly increased the value that could be obtained from a single page. The database management system itself had to be reconstructed to work for our new collaborative system. Instead of editing the UI framework code to allow for such interaction, we decided to create a separate routine to handle all interaction between the user and the database. A few simple PHP and JavaScript scripts allowed us to pop-up text boxes whenever needed so that the user would have a simple way to type in and submit content. All text box content is saved with a session ID, user ID, form ID, and timestamp for viewing by an administrator or instructor at a later time [3].

Allowing for the complete software to be a collaboration of two sub-systems allowed us to maintain the sleek UI functionality that can be achieved with jQuery while maintaining fast and efficient back-end scripting. The user experience is exponentially more valuable due to these critical changes.

4.3 Existing problems

Although the enhancements to the system have produced a satisfactory interactive learning environment, there are still small issues with the software. The software has trouble with all versions of Internet Explorer. Fortunately, free browsers like Firefox and Google Chrome ensure that any organization interested in using the software will have no additional financial requirement to make the software accessible on their system. Oddly enough we also ran in to the issue where some dynamic content can actually load too fast. The most obvious occurrence is if an administrator is scrolling through module objects, it is possible for the system to scroll too fast causing the administrator to overshoot the target object. This is being rectified to allow more control over the dynamic loading of the content. Other than these issues, only minor flaws that are unable to hinder the performance of the system have been found. Smaller issues have been corrected quickly, while the existing issues are still under evaluation.

5 Analysis

A group of four test candidates were allowed to evaluate our system and give feedback in summer 2011. The candidates were middle school and high school educators from neighboring counties. Suggestions were broken down into content and design categories. The majority of the design suggestions were easily achievable. They suggested adding the supplementary material links and checkpoint database support that ended up in Version Three. A design suggestion that has not yet been achieved is the addition of a notepad-like toolbar that remains on the side of the content as the user progresses through the story. This toolbar could be used for note taking as the student learns more about the current storyline. Content suggestions were mostly handled by our colleagues in the History Department. They added new audio and video clips and additional background information to dozens of pages, giving a more thorough virtual experience to the user. A content-related suggestion that is being worked on currently is the addition to having different personas for some of the more general roles. This would allow the student to view the content from the perspective of one of many very different characters as they perceived the Civil Rights Movement at the time.

We were extremely happy with the evaluations given by the educators. They were thrilled with the idea of making education more fun to young students, even offering the idea of a version of the system aimed at grade-school level classrooms. These results suggest that we are on the right track with Version Three, and hope to improve upon it in the future.

6 Future work

An interesting idea that has been suggested by both the History Department and the Computer Science Department is the possibility of expanding this interactive simulation to a virtual world scenario. Virtual worlds are being used by many educational institutions to teach a variety of subjects in a more exciting environment. Our target audience is perfect for such a system, as it can easily be thought of as a video game instead of an educational entity. This idea isn't possible given current resources, but an extension of this program at a later date could result in a virtual world replication of the current system.

7 Conclusion

This project has been extremely rewarding to work on over the past two years. We found a new understanding of efficiency and the many variables it may apply to. It was this new understanding of efficiency that made the project successful. To continue disregarding value of content as a measure of efficiency would have caused the system to be a failure. We started with a buggy system that didn't offer a valuable learning experience to the user. Reconstructing the software to provide a polished user interface with fast and efficient back-end code resulted in software that can be described as efficient in all categories. We used methods of sleek UI design along with efficient, standardized back-end support. The addition of database support and supplemental material access were two key features that enhanced the system. The evaluations and suggestions provided by local educators ensured that we had a product that was up to the standards of our target audience. The next step is to prove that the system can educate a student better than traditional methods. Perhaps these results will be available in the near future. Overall, we created a fast, fully functional, valuable, and efficient interactive system to assist in the education of any type of student.

8 References

[1] SARKAR, M., AND WILLIAM, J. 2010. Digital Democracy: Creating an Online Democracy Education Simulation in a Software Engineering Class. Proc. ACM Conference '10, Oxford, Mississippi, April 2010.

[2] SARKAR, M., AND KEOGH, J. 2011. Digital Democracy: Creating an Online Democracy Education Simulation in a Software Engineering Class. Proc. NUicone - 2nd International Conference on Trends in Technology, Ahmedabad, India, Dec. 8-9, 2011.

[3] KEOGH, J. AND FLETCHER, M. 2011. "Gore Center v2.1 Content Creation Tutorial," unpublished. http://www.cs.mtsu.edu/~jkk2p/GCTutorial2-1.pdf

[4] GEE, JAMES. 2003. "What Video Games Have to Teach Us About Learning and Literacy." Proc. ACM Computer in Entertainment Vol. 1, October 2003.

A Comprehensive Experiment Scheme for Computer Science and Technology

Lisheng Wang[1], **Renzhi Huang**[2]
[1, 2]Department of Computer Science and Technology, Tongji University, Shanghai, China

Abstract - *Nowadays, there are some problems in experiment courses for the students major in CS (Computer Science and Technology), such as experiment teaching is focused on validating experiments and the experimental contents are scattered. To solve these problems, this article puts forward a comprehensive experiment scheme. It guides students to design hardware components first, then achieve their own processor, build a compiler for an advanced language and design a simple operating system. Contents of the comprehensive experiments which are closely related with each other cover most of foundational and core courses in CS. The feasibility of the experiment scheme is still being studied currently, and part of the task has been completed. But some experiments are still lack of detail plan, on which further study is needed.*

Keywords: Computer Science and Technology, Comprehensive experiment scheme, Experiment teaching

1 Introduction

CS (Computer Science and Technology) is a highly technical and practical specialty. While attention is paid to the theory teaching, the experimental teaching also can't be ignored. The scientific and systematic experiments for teaching not only strengthen the students' theoretical foundation, but also help them transfer professional knowledge into the ability for practical application, improve the ability of solving engineering problems and then raise innovative spirit. Colleges and universities are paying more and more attention to the experiments in teaching arrangement. Though the experiment teaching has been strengthen, actually sometimes the effect is not very satisfactory. There are still some problems, such as, with lesser difficulties, the validating experiments are in the majority. When conducting an experiment, the students don't need to understand the principle behind, just finish the experiments according to the experiment guidance. Besides, experimental contents are scattered. Experiments between different courses are unrelated. They are based on a single knowledge, and can't bridge different knowledge to help students find out the importance of certain knowledge points and set up a complete concept of computer system.

In view of the experiment teaching situation and problems described above, this paper proposes a comprehensive experiment scheme for the students major in CS, to help them consolidate the basis theories, also get familiar with the concrete application of certain theory and make innovation in the experiments [1].

2 Basic idea of the comprehensive experiment scheme

This comprehensive experiment scheme aims to design hardware components first, implement a processor, then write an advanced language compiler and a simple operating system. In undergraduate education, experiments in this comprehensive experiment scheme are divided into each course, including Digital Logic, Principles of Computer Organization, Microcomputer Principle and Interface Technology, Assembler Language Programming, Compiler Construction Principle and Operating System. This comprehensive experiment scheme follows two principles. The first is that experiments in different courses correlate with each other. Results of the experiments in pre-requisite course will be used for the experiments in successor curriculum. For example, basic logic components such as register which is designed in experiments of Digital Logic will be used in experiments of Principles of Computer Organization to construct a processor, besides the compiler designed in experiments of Compiler Construction Principle will compile the operating system designed in experiments of Operating System. The second principle is that the comprehensive experiment scheme is focused on designing experiments, and advocates innovation. As compared to validating experiments, designing experiments are more difficult. But if given enough guidance, the students would spend more time to complete the experiments, so the experiments will get better effect. Furthermore, innovation is advocated in experiment conducting. The students are allowed to add new ideas to the experiments [2].

3 Comprehensive experiment design

3.1 Digital logic component design

Been widely used in every field of science, Digital Logic is a practical and important course. As a series of basic experiments in the comprehensive experiment scheme, digital logic component design includes combinational logic and temporal logic component design. Encoder, decoder, data selector, adder and numerical comparator are the typical combinational logic components to be designed, while flip-flop, shift register, synchronous and asynchronous counter are the temporal logic components. There are two ways to implement the digital logic components described above. First, after getting familiar with one hardware design tool, such as ISE design suite supplied by Xilinx, the students draw the circuit of the component as input to finish the design. The second way is that, the students learn a hardware description language, and then use the language to describe the component to finish the design.

In these hardware foundational experiments, there are three tasks to be finished. First, the students should be skilled in the use of one hardware design tool. Second, a hardware description language should be mastered. Then, the logic components stated above would be designed and validated.

3.2 Assembler language programming

The assembly language is a kind of programming language related to the specific system architecture. In view of MIPS32 instructions are open and neat, and the MIPS32 assembler can be got from the Internet easily, the MIPS32 architecture will be introduced in the specific teaching arrangement. Students learn the instructions' classification and format, addressing ways about MIPS32 architecture, and the design method about assembler language programming in class. Experiments for this course are based on MIPS32 simulator. The experimental contents are writing different assembler language programs to satisfy different requirements. For example, designing programs to calculate complex mathematical expressions can help students be familiar with the arithmetic and logical operation instructions, while designing programs to compare two strings can help students be familiar with the conditional jump instructions and recycling program design method. Besides, designing a sort or search program using subroutines can help the students master the thought of designing program in modular.

In these software foundational experiments, there are two tasks need to be completed. First, after getting familiar with the basic instructions and register organization of MIPS32 architecture, the students should know how to use the MIPS32 simulator. Then, students should be skilled in writing assembler language programs and validate them on the MIPS simulator. Experiments for this course lay a foundation for the later processor design. The students who are interested can write BIOS (Basic Input and Output System) for the later designed computer.

3.3 Computer components design and processor integration

There are also series of experiments for Principles of Computer Organization in this comprehensive experiment scheme. Along with theory teaching, guiding the students to implement kinds of computer components could help the students deeply understand the composition structure and work principle of computer components. And it will be helpful in processor designing. The experiments of computer component design are based on MIPS32™ architecture that the students master in the experiments of Assembler Language Programming. Using the logic components designed before, the students can describe the computer components in a hardware description language. The components to be designed include ALU, barrel shifter, multiplier, divider and controller. At last, all kinds of components will be integrated into a full processor, and which would be validated [3].

Tasks to be finished in these experiments are listed below. First, finish all kinds of computer components design. Then learn the method of processor design and integration with computer components and basic digital logical devices [4]. Flexible experiment arrangement can be taken to adjust the difficulty. For example, the students can be divided into groups, so that each group finishes the experiments cooperatively.

3.4 Periphery connection parts design and system integration

The standard computer system includes ALU (Arithmetic Logic Unit), controller, memory, and input/output device. Processor only consists of ALU and controller. In order to facilitate the interaction between users and the processor, also to set up a complete computer system of hardware level, the bus module, interrupt control module and some commonly used I/O modules need to be added. To adjust the difficulty, the bus arbitration function will not be considered, leaving the processor as the only master equipment. Besides, less number of interrupt sources and simple interrupt priority is considered in the design of interrupt control system. Input and output module design can take GPIO (General Purpose Input/Output) and UART (Universal Asynchronous Receiver/Transmitter) as examples. After the completion of the above design, they will be integrated with the processor into a complete computer system. Using some EDA (Electronic Design Automation) tools, the system can be implemented on a specific FPGA development board [5].

As to perfect the computer hardware system, experiments in this section need to finish the following tasks. First, complete the bus module design. Second, design the interrupt control system. Third, write the GPIO and UART module. Last, integrate the computer hardware system, and implement it on a specific FPGA development board.

3.5 Compiler building

Compiler Construction Principle is a quite theoretical course. If there is no experiment, the students cannot find out the practical utility of this course. It must become boring. To help students build a more profound understanding about compiling principle, experiments in this section aim at designing a programming language compiler to translate advanced language programs into binary programs based on the completed hardware architecture. The big experiment task can be divided into many small stages, including lexical analysis, syntax analysis, symbol table management, intermediate code generation, object code generation and code optimization [6]. Lexical analysis stage finishes an advanced language lexical scanner, while syntax analysis stage finishes a syntax parser according to the output of the lexical scanner. As a data structure to save the information of all symbols defined in the source program, the symbol table needs to be well designed. Intermediate code generator finished in the intermediate code generation stage, generates particular intermediate language code. Then the particular intermediate code will be translated into binary code in object code generation stage according to the instruction format and register organization of target machine. The code optimization stage is responsible to the goal of code optimization [7]. Considering the difficulty of building a compiler, it is suggested that the students can be divided into groups to complete the task cooperatively.

3.6 The operating system experiment design

As a core course in CS, Operating System covers many great human ideas. If there is some way to make these ideas concrete, it would be of great help to students in understanding the essence of the operating system. Experiments in Operating System use the computer hardware system designed in pre-requisite courses' experiments as a platform. The students will use MIPS32 assembly language and an advanced language that can be compiled by students' compiler, together with principles of operating system learned in class to design a simple operating system or some function modules. Specific experimental contents are still in exploration. But the general design principle is that while controlling the difficulty in experiment; try best to help students design a simple but complete operating system.

4 Summary

A comprehensive experiment scheme is proposed in this paper. Following it, the students design hardware components first, and implement a processor. After the hardware is built, an advanced language compiler and a simple operating system will be designed. This comprehensive experiment scheme is mainly focused on designing experiments. It guides the students to finish series of experiments related with different courses, so as to strengthen the students' theoretical foundation, and to help them improve the ability of solving engineering problems, furthermore to cultivate their design concept and spirit of innovation. Now, exploration about the comprehensive experiment scheme is ongoing. A processor based on MIPS I instruction set has been built. Some periphery components and the compiler are being designed. But there are still some drawbacks. A detail plan cannot be proposed for operating system experiment design. More efforts are needed.

5 Reference

[1] Z. Yifang and Y. Binwei, "A mode of ability-targeted computer curriculum reform based on projects in engineering education," in Computer Science & Education (ICCSE), 2011 6th International Conference on, 2011, pp. 154-157.

[2] Y. Quansheng and X. Hancheng, comprehensive course design of the computer system. Beijing, China: Tsinghua University Press, 2008.

[3] A. Hoffmann, F. Fiedler, A. Nohl, and S. Parupalli, "A methodology and tooling enabling application specific processor design," in VLSI Design, 2005. 18th International Conference on, 2005, pp. 399- 404.

[4] D. Harris and S. Harris, Digital Design and Computer Architecture. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007.

[5] K. Nakano and Y. Ito, "Processor, Assembler, and Compiler Design Education Using an FPGA," in Parallel and Distributed Systems, 2008. ICPADS '08. 14th IEEE International Conference on, 2008, pp. 723-728.

[6] K. Abe, T. Tateoka, M. Suzuki, Y. Maeda, K. Kono, and T. Watanabe, "An integrated laboratory for processor organization, compiler design, and computer networking," Education, IEEE Transactions on, vol. 47, pp. 311- 320, 2004-01-01 2004.

[7] S. Chattopadhyay, X. Yaodong and W. Haitao, Compiler Design. Beijing, China: Tsinghua University Press, 2009.

Effectiveness of Peer Tutoring: Computer Science Students' Perspective

L.-C. Chang[1], **C.-C. Yang**[2], **M.-T. Chiang**[2], **and G. C. Lee**[2]
[1]Dept. of Information and Computer Education, National Taiwan Normal Univ., Taipei, Taiwan
[2]Dept. of Computer Science and Information Engineering, National Taiwan Normal Univ., Taipei, Taiwan

Abstract - *This study is an exploratory study developed through peer tutors' and students' learning experience to examine college peer tutoring. The goal of this tutoring service is to enable undergraduate students to deepen their understanding of the course materials through discussion with peer tutors. This study used both quantitative and qualitative research methods to investigate if this tutoring service is being used effectively to facilitate undergraduate students' learning. The research results indicate that the peer tutoring service was positively embraced by the students, as the tutoring service provides an alternative learning resource in a learner-centered environment. In addition, both students and tutors agreed that their interpersonal skill also improved during the tutoring process.*

Keywords: Tutoring Service, CS Education, Peer Tutoring

1 Introduction

Peer tutoring has gained more attention over the years to support instructors' teaching and help students to promote their learning [2]. Covin noted that "peers are considered the most powerful influence in undergraduate education, even more so than advisors and instructors" [4]. Thus, the use of peer tutoring has become widespread in the teaching workforce to facilitate student learning worldwide [3].

There are positive effects especially for peer tutors. Several researches have highlighted the constructive effect of having senior students acting as tutors for junior students [5][8]. Alsup, Conard-Salvo, and Peter [1] also found that being a peer tutor could help future teachers gain a richer teacher identity, a better-developed philosophy when teaching writing, added confidence, and increased ability using language to talk about writing.

The benefits of tutoring at the elementary and middle school levels are strong and clear [6][7]; the evidence seems weaker for college-level tutoring services [9]. Therefore, this paper describes our initiative to conduct college peer tutoring and to report on the research findings. This research used a mix-method design through the peer tutors' and students' learning experiences to describe the implementation of a college peer tutoring service. We hope the results of this study can provide a valuable case study for educators and educational reformers to facilitate a reformed program within the curriculum in higher education.

1.1 Research purpose and questions

The objective of this tutoring service initiative is twofold. First, it is to enable students to deepen their understanding of the course materials. Second, it is to enrich the learning experience of undergraduate students by providing increased opportunities to interact with the peer tutors. Thus, the research goals is to find out:

1. What do students think about the tutoring service?

2. What do students like/dislike about the tutoring service?

2 Method

Most colleges use TAs (teaching assistants) to support instructors' teaching and to help promote undergraduate students' learning [10]. Aside from the use of TAs, we experimented with tutoring service to provide additional learning support. In our setup, four qualified graduate students were selected as tutors. The tutoring service is provided at a fix time at the designated tutor room. The tutors together provide learning help for eight computer science courses, including Introduction to Computer Science, Programming in C, Data Structures, Assembly Languages, Linear Algebra, Software Engineering, System Programming, and Automaton.

2.1 Participants: peer tutors

Four graduate students who demonstrated course competency through high marks from previous course works were selected as tutors for this service. Each tutor is responsible for four to six subjects. The tutors must schedule regular office-hours in the tutor room. For this experiment, the office-hours are scheduled on 1:00pm to 2:00pm, 4:00pm to 7:00pm, Monday through Friday. Those time slots are chosen to avoid most popular class hours, which are 9:00am to 12:00 noon and 2:00pm to 4:00pm.

2.2 Participants: students

The tutoring service was announced to all students and all faculties so they can encourage students to make use of the new service. There were 160 undergraduate students in the department, an average of 40 students per class. At the end of the semester, a total of 85 different undergraduate students had used the tutoring service: 53 are from the freshmen and sophomore classes (78% of freshmen and sophomore class students), and 30 are from the junior and senior classes (50% of junior and senior class students). In addition, five graduate students also made use of the tutoring service. Among those 83 students, four students, designated as students A, B, C, D, were randomly selected for post-experiment interview. Those students all have used the tutoring service more than 10 times throughout the semester.

2.3 Data collection and analysis

This study used both quantitative and qualitative research methods to investigate effectiveness of the tutoring service. Questionnaire and interview were conducted to examine students' learning experience. The questionnaire was a feedback form that was completed by students right after each tutoring session. The questionnaire contained only five 5-point Likert scale questions and asked about the particular tutoring experience. At the end of the semester, four students as well as the four tutors were interviewed in person. The interviewed was to confirm the findings aggregated by the questionnaire collected and to have in-depth interpretation of the findings.

3 Results and Discussion

3.1 Quantitative feedback on the tutoring service received

At the end of the semester, 88 out of 160 undergraduate students had used the tutoring service. Every time the tutoring service was used, a feedback form was completed. In all, 226 tutoring feedback forms were received. At the end of the semester, 226 documents of feedback forms were gathered. The feedback form asks students which tutoring service did they receive and their satisfaction level. The results are shown in Table 1. Table 1 shows that more than 53% (120 out of 226) of students used the service to workout problems on the exams of the last few years. About 31% (76 out of 226) came in for help on homework or project. Only 16% (30 out of 226) used the tutoring service to clarify misconceptions or questions from the lecture or textbook. This result shows that students used the service when it matters most with course grades (i.e., exams). In addition, regardless of which tutoring service was used, the satisfaction level was all very high. The percentage of students who were either satisfied or very satisfied with tutoring on homework/project, on lecture/textbook, and on past exams are 94.8%, 100% and 87.5%, respectively. Overall, it seems that the tutors did a good job in helping out the students.

3.2 Qualitative feedback from the students

At the end of the semester, six students among the 83 whom used the tutoring service were randomly selected for interviews. They are designated as student A, B, C, D, E, and F in the following discussion. Students A, B, C, D had each used the tutoring service more than 10 times throughout the semester, while students E and F had used the tutoring service only once. Two major conclusions are drawn from these interviews.

3.2.1 About helpfulness with each of the tutoring service

It was found that questioning and answering (Q/A) for questions arising from in-class lectures or from textbook reading is the most useful tutoring service. The main reason being that students were able to quickly clarify any misconceptions or questions they have.

> *"The tutors helped me understood the key points in a problem and refer back to the textbook for further reading. And when I still don't understand, they would explained to me right away."*
>
> *-- Student A*

> *"I think the lecture/textbook Q/A is most helpful because I thought I understood everything from the lecture. However, when I work on the assignment, I quickly realize that I didn't really understand everything that well. The tutoring service helped me clarified the missing points, which allowed me to complete my assignment."*
>
> *-- Student C*

Interestingly, although more than 50% of students coming to the tutoring service were for help on past

Table 1: Tutoring services and satisfaction level.

Tutoring service	Satisfaction Level					
	Very satisfied	Satisfy	Neutral	Not satisfied	Very dissatisfied	Total
Homework/ Project	55 (72.4%)	17 (22.4%)	4 (5.2%)	0	0	76 (100%)
Lecture/Textbook	25 (83.3%)	5 (16.7%)	0	0	0	30 (100%)
Past Exams	58 (48.3%)	47 (39.2%)	14 (11.7%)	1 (0.8%)	0	120 (100%)
Total	138	69	18	1	0	226

exams, this is the service in which students found to be least helpful. The main reason being that students usually came for help in a group, but since students are not on equal footing, tutor's explanation may be too detail for some, while too abstract for others. Thus students did not get the same timely and right level of help they need as in the lecture/textbook Q/A service.

> *"I attended two past examination explanation session this semester. The tutors worked very hard working thought the exam, but it took too long. Some of the questions raised by the other students made me wonder if they have studied at all. A lot of time was wasted going over the very basic stuff for those selected few."*
>
> *-- Student B*

> *"The tutors did too much to help all, even those who did not study. At one time someone finally shouted to the tutor that 'that was for someone who hasn't studied at all.' I think many of my classmates share the same feeling that the tutors don't need to go in that much detail when working through the problems."*
>
> *-- Student D*

3.2.2 About Learning in the Learner-centered Environment

Many students stated that tutors are more approachable than the faculty members. The support and after class hours (4:00pm to 7:00pm) schedule made students feel relaxed and welcomed. Furthermore, the tutors, being their peers, have similar age and more importantly do not behave like an authority figure. Students perceive tutors as friends. And so the interactions between tutors and students are more like learning partnership during the tutoring sessions. In addition, students indicated that learning outcome is much better when students know what they need to learn (because they asked the questions).

> *"Asking tutors is very different from asking the instructors. Tutors don't teach to us. Tutors are peers and know me personally, so it is like discussing the problems together. The learning environment is not as frightening as asking the instructor the same question during his office hour.*
>
> *-- Student B*

> *"There is no pressure to ask tutors about anything. Don't have to worry the impression I leave with the instructor. Tutors all are very willing to help, so I like coming in for help. I learned more because I get my questions answered."*
>
> *-- Student D*

4 Conclusions

While individual course teaching assistants can provide learning help, not all course TAs can schedule coincidental office hours. On the other hand, tutoring service has the advantage of providing learning help at a fix time at a fix place for many subjects at once. So in this study, we conducted peer tutoring service across many different subjects of a Computer Science curriculum. Data were gathered from the students to look into the effectiveness of the additional service.

Both quantitative and qualitative data showed that students generally have high satisfaction about this tutoring service. Students are most satisfied with the timely answer that the tutors can provide to clear up any misconceptions or questions after attending in-class lecture or after reading the textbook. In addition, having knowledgeable and capable tutors is a key component for students' repeated use of the tutoring service. Students generally preferred bring their questions to the tutors over course instructor or course TA because students see tutors as their friends. After a few sessions, students and tutors developed a bond to learn together, which students thought were not possible with the more authoritative course instructor. Thus, the tutoring service was seen as a friendly learner-centered environment for which students are willing to come for help, which we can say is a big improvement over the few office-hour visits to the course instructors by the students in the past.

5 Acknowledgement

This research was partially supported by the National Science Council of Taiwan, R.O.C., under grant numbers NSC-100-2511-S-003-020-MY2, NSC-100-2631-S-003-006 and NSC-97-2511-S-003-007-MY3.

6 References

[1] Alsup, F, Conard-Salvo, T., & Peters. S. F. (2008). Tutoring is real: the benefits of the peer tutor experience for future English educators. *Pedagogy, 8*(2), 327-347.

[2] Chalmers, D., Herbert, D., Smeal, G., Whelan, K., & Hannam, R. (2003). *Training support and management of sessional teaching staff: a review of the literature*. Retrieved July, 15, 2011, from University of Queensland, sessional teaching project website, http://www.tedi.uq.edu.au/sessionalteaching.

[3] Choat, D. (2006). Strategic review of the tertiary education workforce: casual, part-time and contracted staff. Wellington, NZ: Tertiary Education Commission.

[4] Colvin, J.W. (2007). Peer tutoring and social dynamics in higher education. Mentoring and Tutoring. 15(2), 165-181.

[5] Nikendei, C., Andreesen, S., Hoffmann, K., & Junger, J. (2009). Cross-year peer tutoring on internal medicine wards: effects on self-assessed clinical competencies-a group control design study. Medical Teacher, 31, e32-e35.

[6] Phillips, N. B., Hamlett, C. L., Fuchs, L. S., & Fuchs, D. (1993). Combining class wide curriculum-based measurement and peer tutoring to help general educators provide adaptive education. Learning Disabilities Research & Practice, 8, 148-156.

[7] Robinson, D. R., Schofield, J. W., & Steers-Wentzell, K. L. (2005). Peer and cross-age tutoring in math: outcomes and their design implications. Educational Psychology Review, 17 (4). 327-362.

[8] Sobral, D. T (2002). Cross-year peer tutoring experience in a medical school: conditions and outcomes for student tutors. Medical Education, 36, 1064-1070.

[9] Topping, K. (2005). Trends in peer learning. Educational Psychology, 25 (6). 631-645.

[10] Yang, Y. C. (2008). A catalyst for teaching critical thinking in a large university class in Taiwan: Asynchronous online discussions with the facilitation of teaching assistants. Educational Technology Research and Development, 56(3), 241-264.

Remote Hardware Experiment Teaching

Corresponding author CHAN Yong-qiang Email:chenyongqiang@tsinghua.edu.cn
QUAN Cheng-bin Email:quancb@tsinghua.edu.cn
（Department of Computer Science and Technology Tsinghua University, Beijing 100084）

Abstract： University computer hardware experimental teaching mode can not meet the current needs, it is with chip experimental system, students need to go to the laboratory, and the students spend a lot of time in the hardware equipment maintenance and use, and can't concentrate to pay attention to the programmable logic device and the use of hardware design language. Beijing Tsinghua University has created a set of remote computer hardware experimental system, this system can be local or remote hardware experiment, let the students do experiments to reduce the hardware itself stability dependency. At the same time, the teacher can also have more time to develop the hardware experiment, rather than to maintain hardware., students may at any time, any place (local network), for hardware design.

Key words:Computer hardware experiment;Remote;Network

1、Hardware Experimental Situation

The computer hardware experiment adopted a chip in the experimental system, experimental technology has lagged behind; The experiment is verified experimentally, class of students creative difference; using experimental mode is manual Logic design and manual connection mode, resulting in experimental complicated steps, cannot undertake high level design, influence Experiment on the effect of. Therefore need to update the experimental equipment, the better experimental technology, large-scale programmable logic Logic device and the use of hardware design language is the development direction of computer hardware experiment, using this scheme To solve the above problems in experiment. Computer hardware series courses mainly include digital logic, computer composition principle, structure of computer system Courses, schools and departments are often required for each course to provide a set of separate experimental equipment, but also respectively Provide test site, at the same time for these equipment update and maintenance also need manpower and financial support, so that It greatly increases the investment of equipment and field experiment, the relatively high cost of.

For teachers and students, In each course experiments are required to learn a different experimental equipment use, accentuated the burden of teachers and students, And maintenance and management of the equipment also increases in the laboratory staff. Due to the difference in the course of mining Using different equipment, to establish a complete set of hardware experiment environment, students do not have the overall hardware experiment System. This requires the establishment of a unified hardware experimental system, the computer hardware course experiments, the whole Together, achieve resource recycling, for students and teachers to provide better experimental environment. At the same time, domestic computer hardware experiment are locally complete the experiment content, need the experimenters in practice Inspection equipment operation in front of experimental equipment, obtained experimental results, to achieve the purpose of the experiment, but due to the experimental site Open time and experimental device for limiting the number of experimental equipment in the field, causing only open time can be used To finish the experiment, and many are free.

At the same time as the local experimental process students frequent The actual hands-on laboratory equipment, result in equipment damage rate is higher, the teacher often spend a lot of energy In the maintenance and maintenance of hardware devices, increasing the cost of equipment maintenance. If you use a network to the remote experiment To solve these problems, let the experimenter in remote in any place any time experiment, greatly increased The experimental sites and the experimental time flexibility, improve the using rate of the equipment, reduces to the lab space Requirements, but also reduce the equipment damage rate, thus saving the cost of experiment. Existing remote. Inspection system, are not based on actual hardware equipment, but on the server to do simulation, get the client The experimental results and simulation results, and not the actual hardware operating results, so there could be simulated. Fruit and the practical operating results inconsistency, greatly reduce the experimental results of the reliability and validity of, So the computer hardware experiment must be based on the actual hardware basis, so as to obtain the very good Experimental effect.

2 The Remote Hardware Experimental System

Computer hardware experiment platform effectively solves the problems in the experimental process, using the The unification of the experimental latform, advanced experimental techniques and models, which greatly improves the experimental platform of comprehensive Utilization rate. The experiment platform has the following characteristics:

(1) unified experimental platform

The system completed on the computer hardware experiment integration, support the computer hardware series courses in Trunk course experiment (digital logic, computer principle and structure), provides a rich Experimental support for content, reducing the waste of teaching resources of hardware, shortens the students familiar with the experimental equipment of the time, Improving students' experimental system.

(2) to support local \ remote test and management

This system not only can be done locally experiment, also can be in any location at any time through the network. Check, two are based on the same hardware, the experimental effect is the same, there is no essential difference. All two Through the client on the hardware experiment equipment operation, remote to finish the experiment, but also can be remote management Inspection and test equipment, so as to better service to teachers and students, convenient experiment.

(3) advanced experimental technology

The system uses a programmable logic device FPGA as experimental core chip, supplemented with FPGA and CPLD as the control means to avoid experiments in the wiring steps, with universal USB interface and JTAG interface combined communication method on the remote experiment. Using the hardware design language and EDA tool to experiment, so that students can master the latest hardware design, closely follow the technical direction. The client interface is friendly, simple experimental method, each experiment has independent experimental interface, while providing the software logic analyzer experiment method is convenient for students to experiment. There is provided by the client 's experiments and equipment management tool, is convenient for teacher to experiment and equipment for remote management.

(4) support the experimental innovation

The system not only provides a wealth of experimental content, while also supporting the open innovation experiment, science Students can finish above many innovative hardware experiment content, and not just in we provide Content. At the same time can support teachers to carry out experiments in the course of development, help teachers complete has its own characteristics of the experiment Content.

3 Platform

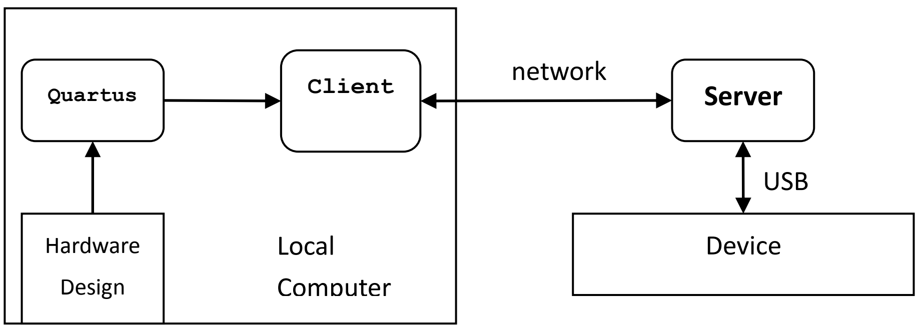
Unified platform of computer hardware experiments using two frame, support local and remote experiment mode, use the Users can choose according to need to use which kind of frame.

3.1 Local Mode

Local mode using an independent chassis, each experiment board is placed in an enclosure, through the USB interface and PC communications. The experimental equipment is responsible for writing code, control board circuit, is provided for experimental microarray experiments Support and monitor, emit monitoring content control chip; responsible for downloading the program to the experimental chip and the control core Films download chip, chip at the same time with the control chip communication relay; storage experimental test range Sequencing and data memory; external communication USB communication interface; available for download chip storage of downloaded content FLASH memory; support a variety of hardware experiments required different kinds of experimental interface. Students in the experimental process, can be directly in the experimental box for operation, directly to the code download In FPGA, can also be through PC client software download FPGA experiment box.

3.2 Remote Mode

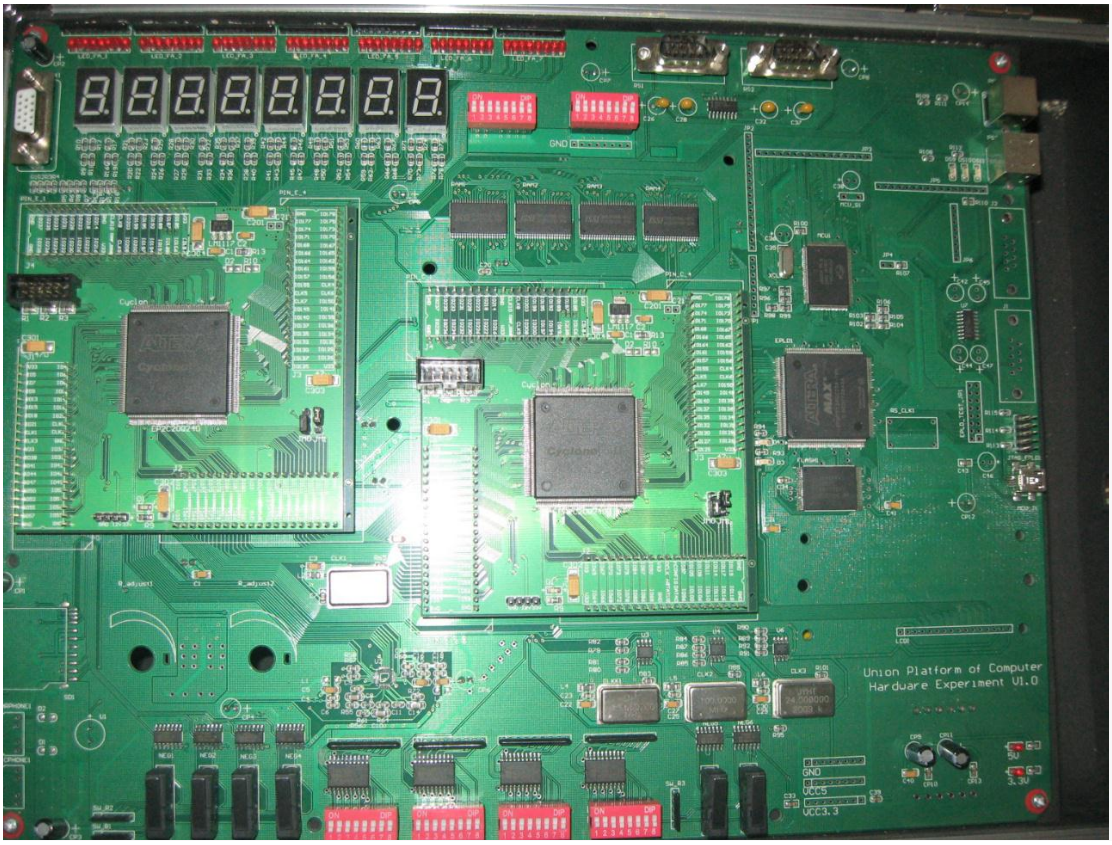
The remote experiment system to a client through a network hardware equipment to complete the experiment as the main experiment mode. The experimental system includes the following three parts:



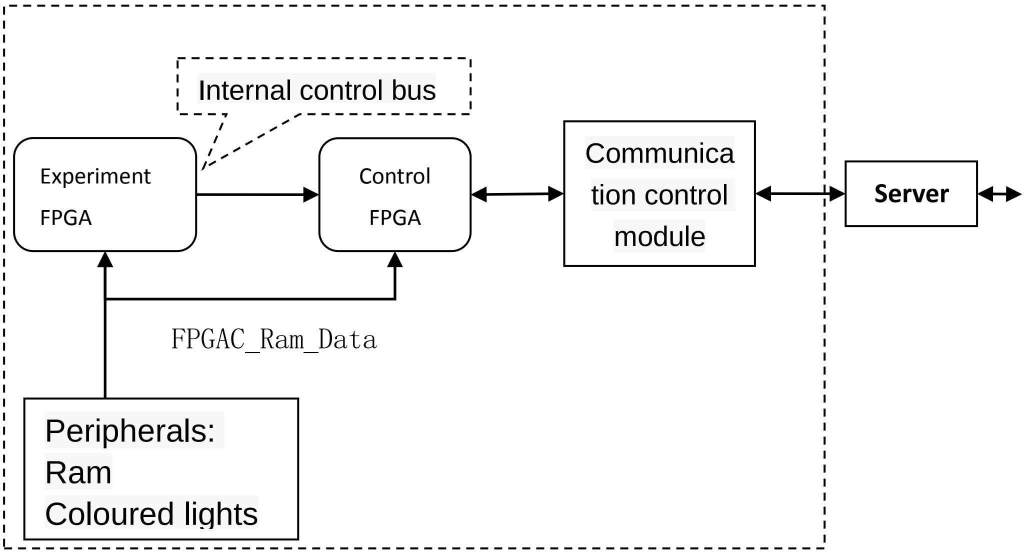
Remote experimental system

(1) The Hardware Experiment Equipment

Includes several blocks experiment board and a server, the experimental board and the local patterns in the same experimental board, server Responsible for the management of these experimental board.



A single experiment hardware equipment



Schematic diagram of experimental equipment

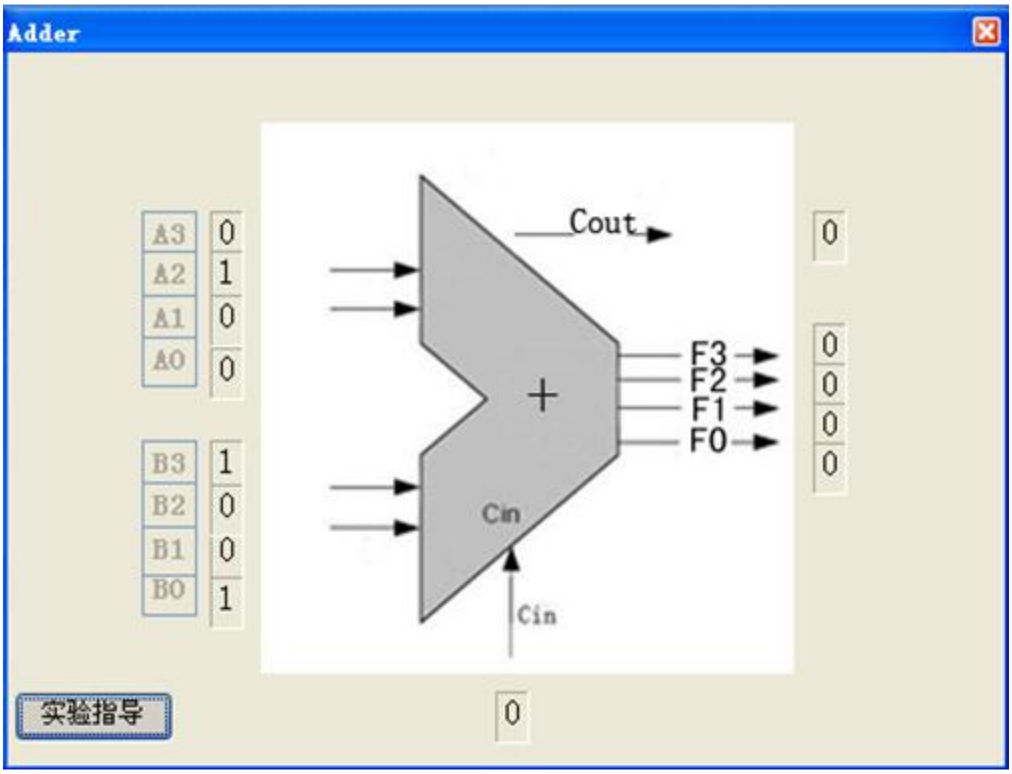
(2) The Management of Experimental Equipment and The Server Software

Server using the windows server, with USB communication interface and Ethernet interface, run USB Device management software, database software and experimental service software, complete the experiment and equipment management and customer Terminal communication.

(3) The Client Software for Experiment.

The client computer using the windows system, with Ethernet interface, running the client software, customers End by specific experimental interface, provide experimental operation and show experimental results. Experimental interface and operation and Local mode.

4. The Experiment Content



Adder Experimental interface

The system supports multiple experimental course, the main contents are as follows:

Digital logic

(1) experiment 1 Multiplexer design

(2) experiment 2 four bit adder (cascaded carry, carry look-ahead)

(3) in Experiment 3 decimal addition and subtraction device design

(4) Experiment 4 multiplier design

(5) in Experiment 5 and the multiplexed signal minimum value

(6) experiment 6 counter design
(7) experiment 7 clock divider circuit design
(8) the 8 state machine design experiment
(9) experiment 9 VGA display controller design
(10) the experiment of 10 PS /2 keyboard interface controller design
(11) experiment 11 comprehensive experiment
Principles of computer composition
(1) experiment 1 coding experiments
(2) experiment 2 computing experiment
(3) Experiment 3 memory experiment
(4) Experiment 4 CPU comprehensive experiment
Computer system structure
(1) experiment 1 single cycle CPU experiment
(2) experiment 2 lines CPU experiment
(3) in Experiment 3 with CACHE CPU comprehensive experiment
(4) open experiment
(5) the multistage memory
(6) the instruction scheduling
(7) open processor design
In addition to the above experiment contents, the system also supports custom experiment content, only need to provide experimental scheme, Can the client to add the corresponding experimental content.

5. Effect

In short, the remote hardware experimental system on traditional computer hardware experimental system undertook reform, make students more convenient to do hardware experiment, teachers more relaxed management course. And now at the Tsinghua University have related courses were used, the results were very good.

Refference

LynuxWorks, Inc. Using the Microprocessor MMU for Software Protection in Real-Time Systems. American,2004.

Advanced Design Tools for Safety Critical Systems SafeAir II Final Report. 2004.

Luolei, Minyuan Zhu. A Formal Semantic Definition of DEVIL. [M],American, ACM Sigplan Notices. April 2003:21.

SESSION

CASE STUDIES, PROJECTS, RETENTION METHODS, AND NOVEL EDUCATIONAL ENVIRONMENTS

Chair(s)

TBA

Building a Secure Virtual Lab Infrastructure for IT Education

R. Sands, G. Hsieh, W. Hendricks and A. Williams
Department of Computer Science, Norfolk State University, Norfolk, Virginia, USA
[rasands, ghsieh, wthendricks, atwilliams]@nsu.edu

Abstract – *This paper presents the design and implementation of a secure virtual lab infrastructure to facilitate the teaching and student learning, through hands-on lab assignments and projects, in information technology and related computing disciplines. This infrastructure is designed initially for the network defense courses taught at Norfolk State University. It utilizes VMware's virtualization infrastructure products and Microsoft's Active Directory Domain Services and Domain Name Service to manage the users, computers, groups, applications and networking for the virtual lab infrastructure which can be totally separated from other lab or production environments. Each authorized user can access a number of dedicated virtual machines via virtual desktops persistently for performing hands-on lab assignments and projects throughout the course.*

Keywords: network defense; virtual labs; secure access; virtualization infrastructure; IT education

1 Introduction

To enhance the teaching and student learning in information technology and related computing disciplines, it is important to utilize hands-on lab assignments and projects in a problem-based and cooperative learning environment [1]-[3]. For example, the network defense courses [4], at both the undergraduate and graduate levels and taught by the Department of Computer Science at Norfolk State University, uses hands-on lab assignments and projects heavily, and students have found this approach very conducive to their gaining in-depth knowledge and skills beyond just conceptual understanding.

Computing resources in the traditional labs have typically been configured with fixed and dedicated environments in terms of operating systems or applications. Often there are simply not enough computing resources to support the number of users or the range of configurations needed. In addition, the users often need to change the system configurations which can cause negative impact or interferences on other users of the labs or even on the production environments. Security is another major challenge as the users often need to perform tasks that require administrative privileges which need to be carefully controlled. Furthermore, the users usually need to be physically in the labs to access the computing resources, set up the required configurations, and then perform the curriculum design or learning tasks at hand.

The issues, challenges, and limitations mentioned above have all been very real for the faculty, students, and IT support staff involved with the network defense courses.

To improve the capacity of the labs, the variety of configuration options supported, and the convenience for users, it is very beneficial and often necessary to utilize virtualization technologies. To ensure that only authorized users can access the shared resources and to prevent major negative impacts or interferences on other users in both the lab and the production environments, a comprehensive set of security and segmentation measures must be established and enforced.

This paper presents the design and implementation of a secure virtual lab infrastructure to overcome the problems and limitations mentioned above with the traditional lab environments. The first-phase development of the infrastructure uses the network defense courses as a case study, and is completed through a M.S. Project research [5].

The secure virtual lab infrastructure utilizes VMware's vSphere® virtualization platform that is widely used for building cloud infrastructures [6]. The major VMware components used for this infrastructure include: VMware ESXi (hypervisor), VMware vCenter Server (virtualization management), and VMware View™ (desktop virtualization) products and solutions.

This infrastructure also uses Microsoft's Active Directory Domain Services and Domain Name Service to manage the users, computers, groups, applications and networking for the virtual lab infrastructure which can be totally separated from other lab or production environments.

To use this infrastructure for IT education, e.g., the network defense courses, the faculty and IT support staff first create a set of virtual machines (VMs) configured with the desired operating systems, device settings, application software, tools, etc. Next, they create multiple pools of virtual machines, using the previously configured virtual machines as templates. The virtual machine templates and pools are used to facilitate the management and automated deployment of virtual resources. Then, the faculty and IT support staff grant entitlement to designated groups of users for authorized access to the pools of virtual machines.

Each student in the class will be given an individual user account under the control of the Active Directory Domain Controller attached to the secure virtual lab infrastructure. To gain access to any resources provided by this infrastructure, the student must first be successfully authenticated by the Active Directory Service. Students are also put into the designated groups that are entitled to access the pools of virtual machines.

Each authorized student is allowed access to multiple VMs, delivered as virtual desktops, simultaneously on a remote client machine, e.g., a Windows PC or an Apple iPad. When the student accesses a VM pool for the first time, one available instance of the VM will be automatically allocated from the pool and assigned to the student. This VM instance is dedicated to this student throughout the course or until released/reassigned. Changes made to the VM (e.g., software installed, configuration settings, or files) by the student are saved and persisted across login sessions, so students can work on complex problems in an incremental manner without having to restart every time. Furthermore, this VM instance is only accessible by this student who is granted the dedicated assignment of the VM, to prevent intrusion, tempering, or unauthorized access of information by others.

1.1 Motivation and Objectives

Developing this secure virtual lab infrastructure was motivated by the needs to address the following challenges in providing a robust hands-on lab environment that is conducive to student learning in information technology and related computing fields.

1) **Convenience:** physical labs are difficult or inconvenient for users to gain access and use. It is especially difficult to provide 24 hours by 7 days access to the students in an academic environment.
2) **Capacity:** there are often not enough computers for larger classes. This problem is further exacerbated by the limitation of supported configurations and the students' schedule constraints.
3) **Configurations:** there is not enough variety (e.g., operating systems or applications) of configurations supported, especially when the limited computing resources are configured with fixed options.
4) **Security:** the users often need privileged access to install additional software or change system configurations (e.g., changing IP configuration). This privileged access needs to be tightly controlled such that it does not compromise the security of other systems for which the users are not authorized to access at an elevated level of privilege.
5) **Protection:** other labs or production environments need to be protected from major negative impact or interferences caused by this lab environment.
6) **Administration:** the users often need to change configurations frequently without affecting other users or being affected by others. The users need to be able to perform the authorized functions autonomously, and maintain stable configurations for a sufficient period of time in order to get the work done.

1.2 Objectives

To help address the issues, challenges and limitations mentioned above, the secure virtual lab infrastructure was developed with the following specific objectives.

1) Use VMware's vSphere virtualization platform and solutions which are commonly used for building cloud computing infrastructures.
2) Use Microsoft Active Directory Service and VMware administrative services to implement and enforce a comprehensive set of authentication, authorization, and access control mechanisms.
3) Allow each user to simultaneously access multiple virtual machines, delivered as virtual desktops, assigned to the user on a persistent basis.
4) Enable secure remote access to the virtual lab infrastructure.
5) Build a secure virtual lab infrastructure for the network defense courses initially.

1.3 Outline

This paper is organized as follows. Section 2 discusses the requirements for the secure virtual lab infrastructure, from the curriculum and systems perspectives. Section 3 describes the major software components, along with their features, functionalities, and configuration considerations, used for the secure virtual lab infrastructure. Section 4 presents the design of this infrastructure in its first phase. The implementation steps for the first-phase infrastructure are described in Section 5. A summary and discussion on future work are presented in Section 6.

2 Requirements

2.1 Curriculum Requirements

The network defense courses taught at NSU cover both conceptual and practical aspects of network security. They first review the threats to network security, and the defense-in-depth strategy and technologies. The courses next discuss the techniques and tools used for network surveillance, traffic analysis, and penetration testing. They then cover three key network defense technologies in depth: firewalls, intrusion detection and prevention systems, and virtual private networks. The courses have a heavy emphasis on collaborative and inquiry-based learning, requiring students to complete eight to ten lab assignments and three projects, ranging from designing and implementing firewall rules on Cisco routers to designing and implementing custom intrusion detection rules using the network intrusion detection system Snort [7].

The network defense courses have dedicated use of a physical lab that is equipped with sixteen Dell workstations (currently running Windows 7), six Cisco 2811 Integrated Service Routers, six Cisco 1811 Integrated Service Routers, six Cisco 2960 Ethernet Switches, and additional Ethernet hubs and switches. There are additional Dell workstations and servers that can be used as shared resources.

The lab assignments for the courses require access to a variety of hosts with different operating systems, including Windows 7 and Ubuntu Linux [8]. They also require the use of a large number of tools and software packages, including the network protocol analyzer Wireshark [9], the network security scanner Nmap [10], and the Eagle Server from Cisco Networking Academy [11]. These lab assignments often require the simultaneous use of three hosts to emulate the roles of an attacker, a target, and an observer.

2.2 System Requirements

The specific system requirements for the development of the first-phase secure virtual lab infrastructure are as follows.

1) Allow students to gain access to the VMs which are hosted on a small-scale datacenter managed by the Department of Computer Science.
2) Each student should be provided with up to three VMs of potentially different configurations.
3) The infrastructure should be able to support up to thirty students enrolled in the courses.
4) The VMs assigned to a student should remain assigned to the same student, until explicitly released, such that the student can maintain the control for the configuration and capabilities of the VMs.
5) Only the authorized student can access the assigned VMs to prevent interferences from other users.
6) Each student should be able to change key configuration parameters for their assigned VMs (e.g., IP addresses) to match the individual configuration requirements specified by the lab assignments or projects.
7) A template should be created for each type of VM such that it can be used to facilitate the initial deployment or redeployment of VMs with a clean known state.
8) A VM archive should be provided for students so they can back up their VMs and save snapshots for troubleshooting, reporting, and grading purposes.
9) Each student should be able to change the IP configuration for the virtual ports on their assigned VMs, such that the student can connect multiple VMs to the same subnet.
10) Only the instructors or IT support staff can create VM templates and VM pools, entitle authorized students to VMs, and remove, restore or redeploy VMs.
11) Only the instructors or IT support staff can provision the authentication, authorization, and access control mechanisms.

3 Software Components

This section presents an overview of the software and hardware components used for the first-phase design and implementation of the secure virtual lab infrastructure.

3.1 Infrastructure Software

To secure access to the virtual lab infrastructure, Microsoft's *Active Directory Domain Controller* and *DNS Server* software are used. These services run on Microsoft Windows Server 2008 R2 Enterprise operating system.

3.2 Virtualization Software

The VMware platform is chosen because it has been a commercial market leader worldwide in virtualization technologies. Furthermore, certain VMware desktop virtualization products (VMware Player [12] and VMware Workstations) have been used for several years for the network defense courses at NSU, so a level of familiarity with these products already exists.

3.2.1 Hypervisor

VMware vSphere *ESXi 4.1* is used as the host operating system upon which virtual machines are configured and installed with guest operating systems. ESXi 4.1 is VMware's hypervisor engine derived from a core Linux kernel. It is installed and operated as a native operating system to avoid performance degradation from running on top of another base operating system.

3.2.2 Virtualization management

VMware *vCenter Server 4.1* is used for centrally and conveniently managing all the ESXi hosts and the VMs running on these hosts in the virtual lab infrastructure. This software runs as a service on a Windows 2008 Server R2 Enterprise host. Microsoft *SQL Server 2005* is used for administrative databases required for virtualization management, and is installed and configured to run on the same Windows 2008 Server R2 Enterprise server that also hosts the vCenter Server service.

3.2.3 Virtual desktop

VMware View products are used to simplify desktop management while increasing security and control for the virtual lab infrastructure.

3.2.3.1 Virtual desktop creation

VMware *View Composer 2.6* is used to create pools of linked clones from specified parent virtual machines. Each linked clone acts like an independent desktop, with a unique host name and IP address, yet the linked clone requires significantly less storage because it shares a base image with the parent.

Because linked-clone desktop pools share a base image, updates and patches can be quickly deployed by updating only the parent virtual machine. End users' settings, data, and applications are not affected.

The View Composer software is installed and configured to run as a service on the vCenter Server instance that manages VMware's virtualization infrastructure.

3.2.3.2 Virtual desktop connection

VMware *View Connection Server 4.6* is used as a broker for client connections. It is used to authenticate users through Windows Active Directory and direct the request to the appropriate virtual or physical machine.

View Connection Server also provides management capabilities for managing desktop sessions, entitling users to specific desktops and pools, establishing secure connections between users and desktops, and setting and applying policies. The Connection Server is installed on a Windows Server 2008 R2 Enterprise server.

Another instance of the Connection Server is installed and configured as a *View Security Server* on a Windows Server 2008 R2 Enterprise server running on another VM.

3.2.3.3 Virtual desktop administration

VMware *View Administrator* is used to configure View Connection Server, deploy and manage virtual desktops, and control user authentication. It is a Web-based application that is installed and hosted on the View Connection Server.

3.2.3.4 Virtual desktop clients

VMware *View Client* enables access to centrally hosted virtual desktops from Windows PCs, Macs, thin clients, zero clients, iPads, and Android-based clients.

The *View Client with Local Mode* allows access to virtual desktops running on a local Windows based endpoint regardless of network availability. The local desktops can be synchronized with the centrally hosted virtual desktops when the user is connected to the network and chooses to do so. This local mode support is very useful as it allows the students to continue working on the lab assignments and projects even when they are offline and away from the campus.

To support View Client with Local Mode on client computers, VMware *View Transfer Server* is used to transfer data between local desktops and the virtual lab during check in, check out, and replication. The View Transfer Server is installed as a service on a Windows Server 2008 R2 Enterprise VM.

4 System Design

Developing the secure virtual lab infrastructure requires the installation, configuration and interworking of a large number of features, services, resources, platforms, and devices. Careful and thorough planning, design and coordination are necessary in order to manage the complexity of this challenging task.

4.1 Computing Infrastructure

A total of eight Dell servers and workstations are used for the infrastructure in order to provide the targeted level of capacity, performance, availability and ease of management. Table 1 shows the designated name, hardware platform, base operating system, and primary functions for each computer.

Table 1. Physical Computing Hosts

Host Name	Hardware Platform	Base OS	Primary Functions
esxH1	PowerEdge T710	ESXi 4.1	VMware VM resource pools
esxH2	PowerEdge R200	ESXi 4.1	File Manager cluster
esxH3	PowerEdge R200	ESXi 4.1	File Manager cluster
esxH4	PowerEdge R200	ESXi 4.1	View Transfer & Security Servers
esxH5	PowerEdge R200	ESXi 4.1	View Transfer & Security Servers
CCSADM1	Precision T3500	Windows Server 2008	Microsoft Active Directory & DNS
CCSVCS	OPTIPLEX 760	Windows Server 2008	vCenter Server & View Composer
CCSVMS	Precision T7400	Windows Server 2008	View Connection Server & Administrator

The main ESXi host, *esxH1*, is used to host all linked-clone VMs to be deployed. It is the most powerful host in the system, with the largest amount of disk storage, memory, and CPU resources. The total resources of esxH1 are divided into four resource pools. The first three are used to provide three VM clones for each student. The last resource pool is used to create and manage the templates upon which the linked-clone VMs are created.

The four Dell blade servers, *esxH2* to *esxH5*, are used to host a network file manager and two VMware View infrastructure components. The blade servers have limited storage resources, and are organized into two clusters for the purpose of sharing storage resources. Each cluster forms one storage resource pool by combining the storage resources from the two blade servers belonging to the same cluster.

In addition, three virtual servers are created and configured to run on these two clusters. The *VS Cluster*, consisted of esxH4 and esxH5, is created to run one instance of VMware View Transfer Server (named *CCSVTS*) and one instance of VMware View Security Server (named *CCSVSS*). The *FM Cluster*, consisted of esxH2 and esxH3, houses the NFS Server (named *CCSFMS*) which is a file server put in place such that users can upload or download files and documents to and from the virtual lab infrastructure. Table 2 summarizes the services and resources configured for the secure virtual lab infrastructure.

Table 2. Services and Resources

Type	No.	Physical Host	OS
VM Resource Pools 1-4	4 (x 30 VMs)	esxH1	ESXi 4.1 / VM
NFS Service (CCSFMS)	1	esxH2 & esxH3	ESXi 4.1 / Win2008 VM
View Security Server (CCSVSS)	1	esxH4 & esxH5	ESXi 4.1 / Win2008 VM
View Transfer Server (CCSVTS)	1	esxH4 & esxH5	ESXi 4.1 / Win2008 VM
Microsoft AD Domain Controller	1	CCSADM1	Win2008 Server
Microsoft DNS	1	CCSADM1	Win2008 Server
vCenter Server	1	CCSVCS	Win2008 Server
View Composer	1	CCSVCS	Win2008 Server
View Connection Server	1	CCSVMS	Win2008 Server
View Administrator	1	CCSVMS	Win2008 Server

Figure 1 shows the ESXi hosts, clusters, virtual servers and resource pools that are configured for the datacenter managed by the vCenter Server instance named CCSVCS. The first-phase implementation of the secure virtual lab infrastructure does not utilize the high availability features provided by VMware virtualization infrastructure for host, storage, or VM redundancy.

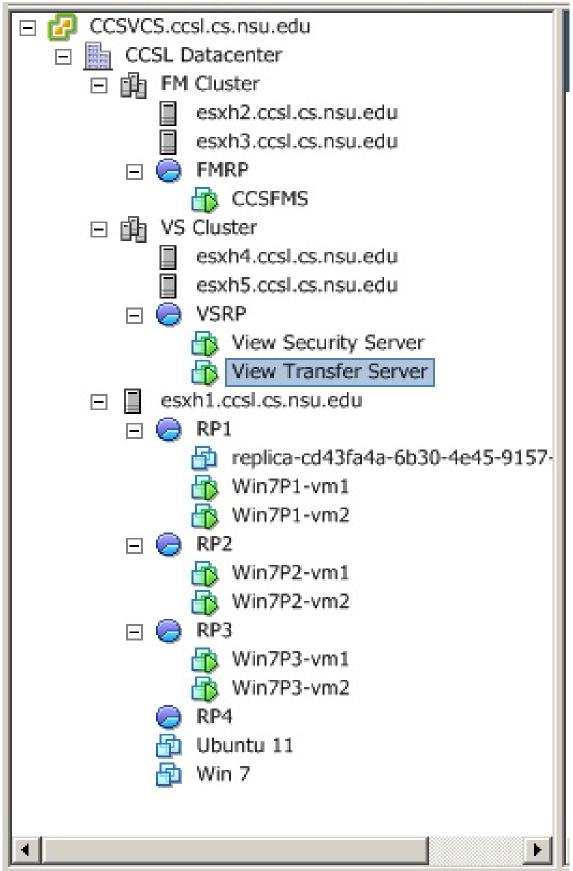


Figure 1. Datacenter Configuration

4.2 Networking Infrastructure

Networking is a complex aspect of cloud or datacenter design and implementation. Due to the small scale of the "datacenter" needed for the secure virtual lab infrastructure, only the basic physical and virtual networking capabilities are used at this time.

The secure virtual lab infrastructure is designed to support up to thirty students in a class. Each student can be provided up to three virtual machines. Each virtual machine can be configured with two virtual network interfaces. Thus, a total number of 180 virtual network interfaces is required for the VMs to be deployed for student use. Each virtual network interface requires a statically allocated IP address.

The secure virtual lab infrastructure currently uses a single private Class C subnet, 192.168.12.0, for assigning IP addresses to all of its physical and virtual network interfaces. A Cisco 2960 Ethernet switch is used for interconnecting all physical network interfaces in the secure virtual lab infrastructure. A Cisco 1841 Integrated Service Router is also installed as a default gateway for connecting to the outside world, although it has not been actively used yet for network security and isolation considerations.

Table 3 shows the IP address configuration for the physical and virtual servers and network interfaces.

Table 3. IP Configuration

Hosts/Interfaces	**IP Address(es)**
Default Router	192.168.12.254
CCSADM1	192.168.12.253
CCSVMS	192.168.12.252
CCSVCS	192.168.12.251
esxH1	192.168.12.245
esxH2	192.168.12.249
esxH3	192.168.12.248
esxH4	192.168.12.247
esxH5	192.168.12.246
CCSFMS	192.168.12.244
CCSVSS	192.168.12.239
VMs	192.168.12.1 – 192.168.12.200

4.3 Security

To enforce the principles of separation of duty and the least privilege provisioning, multiple groups and users are defined with different privileges for managing or accessing different resources. Table 4 shows the groups and users, along with their privileges, that are configured for the secure virtual lab infrastructure.

Table 4. Users, Groups and Privileges

Groups	**Users**	**Privileges**
Domain Admins	ccsadmin	Designated admin account for AD Domain Controller.
View Admins	vcadmin	Designated admin account for View Composer having a GPO attached that limits its overall network access.
	vcsadmin	Designated admin account for vCenter Server.
	vmsadmin	Designated admin account for View Connection Server and View Administrator console.
	vssadmin	Designated admin account for View Security Server.
	vtsadmin	Designated admin account for View Transfer Server.
vDesktop Users	Students	Access to assigned desktops only with elevated privilege to configure assigned desktops.

In addition, authentication, authorization and access control mechanisms are used to safeguard the security and privacy of the infrastructure components as well as the VMs deployed for student use. Security measures are established for three types of access: administrative, student, and system service.

Administrative access to the VMware servers is controlled through the servers' permissions settings. The View Admins group is granted administrative privileges for the View Manager functions, and it is added to the Active Directory. For the vCenter Server and View Composer administrative functions, separate users of the View Admins group are created and added to the Active Directory. The vCenter Server user is granted further privileges as a domain administrator. The View Composer user has a Group Policy Object (GPO) specified which adds restrictions to the networking privileges for this user.

Student access is permitted only for the VMs assigned to the student through a process known as entitling which attaches a user group, *vDesktop Users*, to the pool of virtual desktops. This user group is defined in the Active Directory. For each student in the class, a separate user account is created in the Active Directory as a member of the vDesktop Users group. Before a student can access the virtual lab infrastructure, the student must be authenticated by the Active Directory Service first. The View Security Server is used to establish a secure connection between the student's client machine and the virtual machine or desktop, which is authorized for the student to use and is housed on one of the ESXi hosts in the datacenter.

Table 5 shows the servers that are used to provide the authentication, authorization, and access control security services.

Table 5. Security Services

Services	**Service Providers**
Authentication	Active Directory Domain Controller
Authorization	View Security & View Connection Servers
Access Control	View Transfer & View Connection Servers

5 System Implementation

Implementing the secure virtual lab infrastructure also requires very careful planning and coordination to ensure that the prerequisite features, capabilities and data are in place and operational before attempting the task at hand.

Due to the high complexity of the implementation, it is very important to take an incremental and iterative approach to make sure the foundation is solid and robust, before the next layer of features, capabilities and data are added. It is also important to conduct integration tests frequently as new functionalities and data are gradually incorporated.

5.1 Infrastructure Implementation

The following implementation steps are used to install, configure, and administer the secure virtual lab infrastructure in its first-phase development: 1) Microsoft Active Directory Domain Controller and DNS Server; 2) VMware vCenter Server; 3) VMware ESXi 4.1 Hosts; 4) VMware View Composer; 5) VMware View Connection Server; and 6) VMware View Transfer and Security Servers.

Microsoft AD DC and DNS. Install Dell Precision T3500 computer. Install Windows Server 2008 R2 Enterprise operating system. Activate and configure the Active Directory Domain Controller role. Activate and configure the DNS role. Set up designated domain admin group and users.

VMware vCenter Server. Install Dell OPTIPLEX 760 computer. Install Windows Server 2008 R2 Enterprise operating system. Set up designated group (View Admins) and user (vcsadmin) in AD DC. Add CCSVCS and its IP address to DNS database. Install VMware vCenter Server software. Install and activate Microsoft SQL Server 2005. Initialize vCenter Server Databases. Create a Datacenter and store the configuration information in vCenter Server Databases.

VMware ESXi 4.1 Hosts. Install ESXi 4.1 operating system on Dell PowerEdge servers. Add the hosts and their IP addresses to DNS database. Add each ESXi host to the Datacenter which is defined on the vCenter Server. Create the VS Cluster on vCenter Server to include esxH4 and esxH5. Create the FM Cluster on vCenter Server to include esxH2 and esxH3.

VMware View. To prepare Active Directory for installation and configuration of View components, first create an Organization Unit (OU) for View Desktops. Next create OUs and groups for Kiosk mode client accounts. Create groups for View Users. Create a user account (vcadmin) for View Composer. Configure the Restricted Groups Policy for View Composer users.

VMware View Composer. Create a View Composer Database on SQL Server 2005 Server running on CCSVCS. Install VMware View Composer software and activate the service. Add View Composer and vCenter Server to View Manager, and DNS database.

VMware View Connection Server. Install and configure View Connection Server software on CCSVCS. Configure View Client Connections on the View Connection Server. Add View Connection Server to DNS database.

VMware View Transfer Server. Create a new VM on VS Cluster. Install Microsoft Server 2008 R2 Enterprise operating system on the VM. Install VMware View Transfer Server software on CCSVTS. Create View Transfer Repository on local storage. Configure firewall rules for CCSVTS. Add CCSVTS to DNS database, View Manager, and Datacenter on CCSVCS.

VMware View Security Server. Create a new VM on VS Cluster. Install Microsoft Server 2008 R2 Enterprise operating system on the VM. Install VMware View Transfer Server software on CCSVSS. Add CCSVSS to DNS database, View Manager, and Datacenter on CCSVCS.

5.2 VM Provisioning

Once the infrastructure layer for the secure virtual lab infrastructure is constructed and operational, VM pools are created and provisioned for automated deployment to authorized users. The first step in this process is to create a VM template. Next, a pool of VMs is created as linked clones of the template VM. Then, authorized users are given entitlement for accessing the VM pool.

VM Template. To create a VM template, first create a VM and configure it with the desired guest operating system such as Windows 7 or Ubuntu Linux. Next, install VMware View Agent on the VM. This agent communicates with View Client to provide features such as connection monitoring, virtual printing, and access to locally connected USB devices. Specify a name and template inventory location for the VM template. Specify the host/cluster (esxH1) and the datastore for the VM template. Then make a snapshot of the VM template.

VM Pool. Use View Administrator to create a VM pool. Next, configure the pool as an automated pool such that VMs in the pool can be automatically deployed. Configure the pool for dedicated user assignment. Next assign an ID and display name for the VM pool. Change the pool settings to "enabled". For user convenience, set the 'remote desktop power policy' to "suspend", and 'allows users to reset their desktops' to "yes". Configure the persistent disk size, VM naming convention, and the pool size. Finally, configure the vCenter Server settings to specify the VM settings (folders and name of the default template), resource settings (esxH1, resource pool, and datastore), and guest information (domain, and users).

Entitling Users. After a VM pool is configured and enabled, the "vDesktop Users" group in the domain is selected as entitled users. This in effect grants the authorized students permission to access this VM pool.

5.3 VM Deployment

Users access the VM pools using remote systems installed with View Client software. A Windows-based View Client also supports the Local Mode such that a user can download the vDesktop to the client system and work with it without having to maintain a network connection to the secure virtual lab datacenter.

After a user connects to the secure virtual lab datacenter and is successfully authenticated by the Active Directory Service, the user's View Client automatically displays the three VM pools that the user is entitled to access. The user then obtains a new VM from a pool for the first-time access, or reconnects to the VM that has been assigned and dedicated to the user.

Using the View Client, the user can display the three virtual desktops simultaneously on the screen, as shown in Figure 2. This is a very useful feature as the user can see what is happening on all three VMs at the same time.

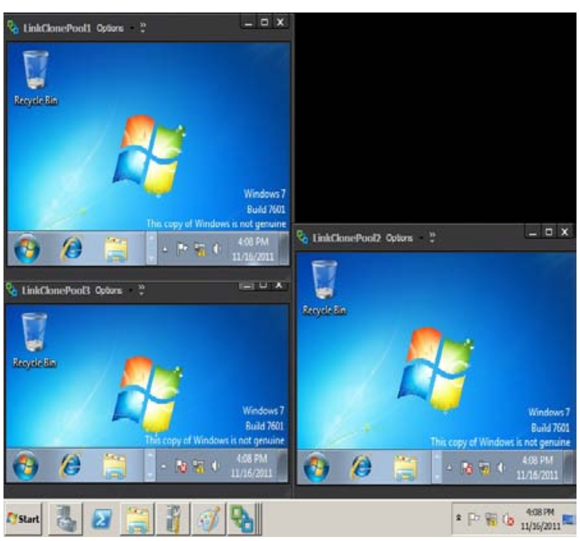


Figure 2. View Client Access to VMs

6 Summary and Future Work

This paper presents the design and implementation of a secure virtual lab infrastructure to facilitate the teaching and student learning, through hands-on lab assignments and projects, in information technology and related computing disciplines. The first-phase implementation, targeted for the network defense courses taught at NSU, is operational, although no student users have yet used this newly developed lab environment which is planned to be deployed for the Fall 2012 semester.

This secure virtual lab infrastructure is constructed with a large number and variety of systems, applications, and features in order to provide the targeted level of security, capacity, performance, functionality, and flexibility. In addition, it utilizes an extensive set of administrative and management services, such as Active Directory, vCenter Server, View Composer, and View Manager, to simplify the administrative tasks and thus enhance the sustainability of this infrastructure.

The design, implementation, and operation of this infrastructure require a variety of knowledge and skills and are quite complex in nature. Thus, they provide good learning and research opportunities for students as well.

For security consideration, the lab infrastructure is currently accessible only on campus. Going forward, the team is very interested in expanding this infrastructure into private-cloud based services such that users can access the lab services from anywhere. The team is also very interested in implementing high availability features to further improve the robustness of the secure virtual lab infrastructure.

7 References

[1] Capon, N. and Kuhn, D. (2004). "What's so good about problem-based learning." *Cognition and Instruction* 22(1): 61-79.

[2] Brickman, P., et al. (2009). "Effects of Inquiry-based Learning on Students' Science Literacy Skills and Confidence." *International Journal for Scholarship of Teaching and Learning*.

[3] Buck, J. R. and Wage, K. E. (2005). "Active and cooperative learning in signal processing courses." *IEEE Signal Processing Magazine* 22(2): 76-81.

[4] Department of Computer Science. http://www.cs.nsu.edu.

[5] Sands, Reginald A. (2011). Building a Secure Virtual Lab Environment using VMware Virtualization Infrastructure. M.S. Project. Department of Computer Science, Norfolk State University.

[6] VMware. http://www.vmware.com.

[7] Snort. http://www.snort.org.

[8] Ubuntu. http://www.ubuntu.com/.

[9] Wireshark. http://www.wireshark.org/.

[10] Nmap. http://nmap.org/.

[11] Cisco Networking Academy. http://cisco.netacad.net.

[12] VMware Player. http://www.vmware.com/products/player/.

The Software Productization Center at Millersville University: A Study in Successful Cross-Disciplinary Collaboration at the Undergraduate Level

Stephanie Elzer[1]*, Nancy Mata[2], Patrick McCaskey[3], and Stacey Irwin[4]
[1]Computer Science, [2]Art & Design, [3]Management and Marketing, [4]Communication and Theatre
1 S George Street, Millersville University, Millersville, PA, US
{stephanie.elzer,nancy.mata,pat.mccaskey,stacey.irwin}@millersville.edu
*Corresponding author.

Abstract— *The Software Productization Center (SPC) is a cross-disciplinary center whose mission is to provide emerging technology-focused entrepreneurs within the Central Pennsylvania region with assistance in advancing software products from concept to marketable product. Faculty and undergraduates from the Computer Science, Art & Design, Management and Marketing, and Communication and Theatre departments actively collaborate with entrepreneurs to ready their products for the next stage of development, whether it is beta testing or obtaining additional financing. The cross-disciplinary collaboration provides rich opportunities not available in the classroom, and the SPC model has yielded concrete positive results for both the entrepreneurs and the students involved.*

Keywords: collaboration, cross-disciplinary, mentoring, entrepreneurship, product development, software

1. Introduction

The Software Productization Center (SPC) at Millersville University is a novel model for cross-disciplinary collaboration that capitalizes on the strengths of a primarily undergraduate institution to achieve goals that are included in many universities' vision statements or strategic goals, such as strengthening regional economic growth and enhancing the educational opportunities available to students. The SPC delivers tangible benefits for entrepreneurs while expanding on traditional models of university-industry collaborations and university business incubators to actively engage and mentor undergraduate students in the process of software productization. Unlike many single discipline university-industry collaborations, the SPC includes faculty and students from the Computer Science, Art & Design, Management and Marketing, and Communication and Theatre departments.

The term "software productization" refers to the process of turning a rudimentary software concept or early prototype into a marketable product. Thus, the efforts of the SPC are primarily focused on assisting local entrepreneurs with market research, the technical implementation of a working prototype of the software concept, the development of business and marketing plans for the launching of the product, and the branding and physical packaging of the product. Once the collaboration between the local software entrepreneur and the SPC is complete, the entrepreneur should be prepared to take the next step with the product, either by attempting to raise additional funds (through business loans, angel investors, venture capitalists, etc.), by performing beta testing, or by directly selling to a customer base.

The significant contribution of the SPC as a model for an academic center comes from the combination of disciplines that are included. Developing a software *product* requires more than technical expertise, and the SPC encompasses many of the additional skills that are required to transform a software prototype into a full-fledged product, while simultaneously achieving significant results in terms of providing a quality undergraduate educational opportunity.

2. Measurable Results

Since its inception in January 2008, the SPC has completed five collaborations involving entrepreneurs and software products representing a wide variety of markets and technologies. The faculty of the SPC have mentored over 25 undergraduate students as they worked to bring these projects to fruition. The entrepreneurs who have worked with the SPC have thus far attributed over $100,000 in sales of their products, $100,000 in cost savings in developing their products, and $80,000 in financing opportunities for their companies/products to their participation with the SPC. These numbers are particularly satisfying since the entrepreneurs with whom the SPC works are typically in the very early stages of building their product and/or company. In the course of completing these projects, some of the SPC's deliverables have included software prototypes, websites (to support the sale or administration of the product), marketing plans, SWOT analyses, physical packaging design, product identity packages, and animations demonstrating a product's use (see Section 5 for details).

3. Selection Process

After some trial and error, the SPC has settled into a workable and repeatable process. The model upon which the SPC is based leverages faculty expertise to guide and mentor students in working with the participating entrepreneurs. In fact, approximately 45% of the SPC's annual budget is comprised of student wages, a fact that underscores how vital student participation is to the success of the center. Thus it is not surprising that the students' availability and schedules have strongly influenced the project schedule of the SPC. Students have larger blocks of time to dedicate to the project during the summer, and this makes summer the ideal time to begin the collaboration and complete the bulk of the project. Attempting to begin a project during a semester when students are juggling classes and other commitments along with the SPC has not been nearly as successful. However, projects can be successfully completed during the fall semester. A large factor in this seems to be that the project is well underway, the tasks are understood, and perhaps even more importantly, the team has coalesced, and relationships and communication channels are well established. Therefore, as you can see in the timeline described in Section 5, we have recently settled into a project cycle that begins in May and wraps up by the end of the fall semester.

We have also found that attempting to have the team work on more than one project at a given time has adverse effects on the degree of collaboration that we usually achieve. The strength of the SPC lies in the cross-disciplinary collaboration and the unique ideas that come out of it, and when the team is faced with multiple projects, the tendency seems to be for one discipline to concentrate on one project, while another discipline concentrates on the other. Armed with this observation, we now restrict ourselves to one project per year.

3.1 Application process

Eligible applicants are limited to those entrepreneurs or businesses with revenue that comes, or will come, from selling software products. This eliminates software that simply supports the business, website development to support a business (rather than a new product), and so forth. Applicants may fall into three categories: 1) individuals, 2) startups (we define this as businesses with five or fewer employees which have existed for less than two years), and 3) small businesses seeking to develop a new software product. Applicants are also restricted to being located within a particular geographical region, defined by counties.

The SPC utilizes a three-stage application process. The first stage (Request for Qualifications) is a pre-proposal process where the solicits brief (3-page) applications that describe the general qualifications of the applicant and a brief description of the proposed project. One of the main reasons for employing this first stage is as a screening process. We are soliciting ideas from the general community, and some may be poorly conceived, or some may be terrific ideas that cannot possibly be implemented by undergraduates. This screening process is lightweight enough that the applicant does not have to invest an inordinate amount of time if there is not a good match between the project idea and the SPC.

The applications from the first stage are reviewed (there are typically between twelve and fifteen initial applications) and the top applicants, about half, are invited to submit a full-length proposal. The full-length proposal involves the preparation of financial data, far more detailed project information, initial marketing projections, and so forth. After reviewing the proposals, two or three applicants are invited to present their concepts to the Faculty Steering Committee and Community Advisory Board (see Section 3.2), who then select the entrepreneur who will work with the SPC for the coming months. The Faculty Steering Committee is composed of the faculty who participate in the productization process and mentor the students throughout the project.

3.2 Community Advisory Board

Advisory boards are becoming increasingly prevalent at academic institutions as a way to formalize the channels of communication between the university and industry [1], [2], [3]. They have long played a vital role for businesses, especially startup companies [4], but are becoming increasingly recognized as valuable even for organizations such as small family businesses [5]. Thus it makes sense that the SPC, whose mission encompasses both education and entrepreneurship, would make use of this device.

The SPC's Community Advisory Board plays a key role in guiding the SPC, providing a practical bridge between the academic environment of the university and the business community. The Advisory Board not only advises the SPC on direction, but it also augments the knowledge of the Faculty Steering Committee in several areas. Fortunately, the members of the Advisory Board provide a mix of knowledge and skills that we have found to be nearly ideal. The following are key areas of expertise that have been identified:

- *Banking:* Having at least one board member with banking expertise ensures that the financial statements of the entrepreneur are appropriately understood, and that areas of financial weakness and risk are well understood (it should be noted that areas of weakness should be expected, given that these are early stage entrepreneurs, but it is still important to clearly assess the situation).
- *Local Business:* Expertise in the local business community enables the board to advise the SPC on other companies who might be attempting similar things, the long-term viability of the enterprise, and even specific background information on applicants that might be common knowledge in the business community, but unknown in the academic environment.

- *Entrepreneurship:* Board members with history as entrepreneurs and/or with responsibilities in assisting entrepreneurs can help to root many discussions in actual experience and examples. Successful entrepreneurs will also hopefully have knowledge about alternative types of financing, such as angel investors.
- *Legal:* Legal expertise has repeatedly proven to be invaluable during board discussions, from navigating potential conflicts of interest to debating intellectual property issues. This knowledge can also be extremely helpful in the early stages of forming the center, when participation agreements between the entrepreneurs and the center need to be drafted.
- *Technological:* The applications received by the SPC cover a wide variety of technology and technical skills (this can even been seen in the selected projects in Section 2). Having multiple board members with practical experience in software development allows projects to be more thoroughly vetted with regard to overall feasibility, and, equally important, as a match to student skills.

Studies of advisory boards engaged with academic organizations have found that when the mission and goals of both the organization and the role of the advisory board are clearly defined, the partnership tends to be more effective [1], [2], [3]. This can be somewhat more difficult within purely academic organizations, and the SPC benefits from both a well-defined mission and very clear tasks, such as the active participation in project selection, for which it requires the Advisory Board's input. The SPC also relies on the Advisory Board for high-level guidance (see Section 7), and here the relationship becomes more representative of academic advisory boards. One area that should be clearly delineated early on is the advisory board's expected role in fundraising [3]. Because the SPC's Advisory Board includes several successful entrepreneurs whose participation is considered vital, the SPC (in alliance with the university development office) assured board members that they would not be solicited for monetary support in the context of their Advisory Board participation. We posit that this agreement has simplified the relationship between the Faculty Steering Committee and the Advisory Board considerably.

3.3 Application evaluation

A list of selection criteria has been defined for evaluating applicants. Some of the items address the viability of the company and product, while others deal with the nature of the collaboration with the SPC. Current selection criteria include, but are not limited to:

- Qualifications/experience/ability of proposed leadership team
- Potential for sustainability of business once launched
- Degree to which the proposed activity addresses a significant industry/market need or opportunity
- Potential for near-term commercial application in terms of job creation, capital investment or other identifiable economic activity
- Degree of innovation and originality of proposed software product
- Match of required assistance with SPC resources and skills

One other aspect that is always taken into consideration should be noted, and that is the perceived ability of the applicant to work well with our undergraduate team. Many applicants are excited about the prospect of working with young, energetic undergraduates, hearing their ideas, and even about connecting with some potential future employees. Others (a much smaller number) seem to approach the collaboration as far more of a consulting relationship where they will be specifying the outcome and results will be delivered along a set schedule. Applicants in the former category have a far greater likelihood of being satisfied with both the process and the outcome, and projects which have appeared to have strong potential have been rejected due to indications of a mismatch in outlook and expectations.

4. The Collaborative Process

The significant contribution of the SPC as a model for an academic center comes from the combination of disciplines that are included. Software might be developed by teams of software engineers (and even that is a limited view, as will be discussed in Section 4.4.1), but developing software *products* requires more than technical expertise. Marketing, business planning, product identity (design), public relations, testing, documentation, customer support, and software development all need to interact to yield a successful product.

4.1 Team composition

The SPC project teams include the entrepreneur(s), faculty, and undergraduate students. Each project has included a faculty member from the Computer Science, Art & Design, and Management and Marketing departments, along with two computer science students, at least two design students, and at least one marketing student. Faculty members and students from Communication and Theatre have also been involved in several projects. When the SPC was formed, a faculty member from the Communication and Theatre department was part of the Faculty Steering Committee. However, due to the rather demanding time required to participate with the SPC, she couldn't sustain her involvement. After distributing the communication/public relations needs of several projects among the remaining faculty for several projects, and recognizing that this was not a sustainable solution, we happily have formal representation from this discipline once more.

4.2 Teamwork vs. collaboration

While most computer science programs recognize the importance of teamwork, most coursework focuses on individual contributions [6]. This is despite the fact that the Accreditation Board for Engineering and Technology (ABET) is emphasizing teamwork skills in the accreditation criteria for computer science programs [6]. When teamwork is incorporated into computer science education, it nearly always involves homogenous teams of computer science students (for example [7]). There are practical reasons for this, of course, given the complications in offering mixed discipline courses that meet the needs of all involved, as well as dealing with mixing faculty teaching loads across departments and even different schools, and so forth. But this insular view of software development will almost certainly not reflect the reality that the students will be faced with when entering the software industry.

In fact, even when teamwork is included in coursework, it may not effectively reach the level of collaboration. Teams are often utilized in upper-level (junior or senior year) undergraduate Computer Science courses, and the teamwork is frequently centered around a division of labor necessitated by the scope or size of the task rather than its complexity [8]. When viewed in a learning context, collaboration is not more than one individual performing together, it is a more specialized form of interaction where "because of engagement in collaborative activities, individuals can master something they could not do before the collaboration" [9]. That is, something is accomplished as the result of collaboration that could not have been accomplished as individuals.

It is in this sense that the SPC provides a unique learning experience for everyone involved – each discipline, and each team member, brings skills, knowledge and perspective that the other team members' not only do not possess, but cannot accomplish the project without. The SPC provides an eye-opening experience in dealing with very different ways of communicating, framing a problem, completing tasks, and, ultimately, working together to succeed.

4.3 Project management

Team cognition refers to "the mental models collectively held by a group of individuals that enable them to accomplish tasks by acting as a coordinated unit" [10]. Pragmatically, at the beginning of an SPC project, we are dealing with a team of inexperienced undergraduates from different disciplines (which often manifests itself as different cognitive styles [11]), faculty who very definitely have different cognitive styles as well as preferred styles of interactions, and an entrepreneur who might or might not share background knowledge with some subset of the rest of the team. Getting this diverse group to come to a shared mental model of the task at hand so that team cognition can begin to transpire naturally takes some time, especially given that the overall task is often only loosely defined at the beginning of the project.

We have found that one of the most valuable activities at the beginning of the project is to actively brainstorm the product identity (this may include the company identity if the company is a startup). By doing this as a group, team members begin to form a collective vision for the project (the early stages of a shared mental model), and, just as important, they begin to understand one another, and how each person perceives and approaches problems differently. Our discussion will often involve high-level concepts that pertain to specific disciplines, such as the target market, potential logo features, and vision statement. However, there is only so much time that can be spent productively brainstorming as a group, and undergraduates at this stage are anxious to be actively producing something. If not given enough concrete tasks at the beginning of a project, we have found that this initial lack of productivity can establish the pace for the whole project, with team members expecting light loads and relaxed schedules. Therefore, in addition to the group meetings, each faculty member assigns the students from his or her discipline specific tasks that will eventually contribute to the overall deliverables.

Communication frequency in teams has been shown to be a significant factor in the team's performance, with high-performing teams communicating more than low-performing ones [10]. In addition, the number of face-to-face meetings and phone calls have been shown to have a positive impact on shared understanding in software project teams, while emails were not [10]. This is not to say that email is not a valuable tool, but it does not seem to carry the same value in terms of building team cognition. Our experience has certainly mirrored these research results. Team meetings have been shown to be essential for the SPC, and we strive to have team meetings of approximately two hours at least once per week (often more frequently in the beginning of the project). In the later months of the project, the face-to-face meetings may be replaced by phone calls if the entrepreneur is located a signficant distance from campus. Email has actually been seen to cause some interdisciplinary friction, with computer science team members expecting it to be more effective and immediate than it proves to be, since team members from the other disciplines are far less likely to check it with the anticipated frequency. Practical file-sharing and other team communications are facilitated by the use of software such as Basecamp.

4.4 Cross-disciplinary interactions

Although the overall interaction of the team is collaborative, time has demonstrated that some of these collaborations are "looser" than others. For example, the marketing and business planning goals within the project are dependent on the shared mental model of the team, and on design deliverables, but there is not typically the need for daily interaction since the tasks can be completed somewhat independently. The communication/public relations deliverables are of a similar nature – they certainly represent team knowledge and shared vision, and will be less than satisfactory if the mental model is not in place, but the actual work can be done independently with "checkpoints" of revisions and approvals by the rest of the team.

4.4.1 Computer science and design

The relationship that has proven to be the most tightly bound throughout the process is that of computer science and design. Perhaps naively, this came as somewhat of a surprise as the SPC developed. When the SPC was originally conceived, involvement with the Art & Design department was envisioned as somewhat of a consulting relationship rather than as having that discipline represented as a full collaborator. The idea was that the designers might produce logos and some packaging for the product, while interacting independently with the entrepreneur and marketing team members. In reality, as project after project has unfolded, this original concept has given way to a rewarding collaboration.

The observation that art and science can make for a rich partnership is not unique to the SPC, of course. Elfman et. al [12] observe that recent advances in graphic software make this pairing even more relevant than it has been in the past because of the visualizations that can be created in order to make scientific concepts accessible. Computer science and design are obviously and effectively paired in the development of video games, and there has been recent research in taking some of the team creative exercises from this industry, such as "game jams", and applying them to general software development [13]. We believe, though, that what we are seeing in terms of the naturalness, and even the *necessity*, of the collaboration between computer scientists and designers in creating a software product is a reflection of what happens when we take software development out of the classroom and begin to try to meet the real expectations of the market.

Zyda [14] posits that we are living in the conceptual age of technology, which he defines as "cognitive and creative assets, with the design side being just as good." The expectations of the users of software are changing, and the way that we develop software needs to change with it. The opportunity for computer scientists (and designers) to learn this at a formative stage in their education is, we believe, one of the most valuable contributions of the SPC.

For undergraduate computer scientists, who are still mastering the art of software engineering, and designers, who may not yet have a complete understanding of the design needs of technology, to learn to work together to produce a marketable software product is not an easy task. However, it is from this collaboration that we see the most concrete examples of team cognition and knowledge transfer. In early team meetings, computer science students join in the discussion of designs being presented and, gradually, as the project progresses, the ideas that they are expressing begin to use the vocabulary and concepts of the design students. In meetings late in the project, the computer science students might proudly point out aspects of a design that they have actually contributed and explain why they made those suggestions. In relaying their ideas to the team, design students have been known to abandon their wireframes and communicate via a flowchart that they have devised (much to the shock of the computer scientists).

These examples of a flourishing shared mental model do not happen, especially at this undergraduate experience level, without an abundance of communication. Early SPC projects had the computer science students and design students working in separate labs, and communicating at meetings and via Basecamp or email. This caused frustration on both sides. For example, computer science students would repeatedly get design deliverables in the wrong format, and created by proprietary software, so that they could not convert the files. They would send an urgent email, which the design students wouldn't see, sometimes for days. The design students, for their part, didn't understand why the file format was causing such an issue, and didn't share the expectations of the timeliness of email communications. The students themselves solved this problem in the middle of a project when the design students picked up their computers and moved them into the lab occupied by the computer science students. The co-location of the computer science and design students facilitated the communication in a way that nothing else could, and this example has been followed in subsequent projects with great success. While teams with greater levels of expertise or greater shared knowledge might be able to overcome a geographic separation, the proximity of their learning partners is crucial for the computer science and design undergraduates. As one student noted after a particularly grueling but productive team meeting, "It's good that we keep talking. I keep forgetting how different we are."

4.5 Student benefits

Students unanimously report their experience with the SPC as being extremely valuable, with many classifying it as the most important experience that they had as an undergraduate. They cite the satisfaction of working on a project that was "real" and their newly acquired skills in interacting within a collaborative team as some of the most rewarding aspects. Numerous students attribute their success in finding desirable jobs to their work with the SPC, and say that discussion surrounding that experience dominated their job interviews.

5. Completed Projects

The SPC has worked on projects representing a variety of target markets and different technologies. The SPC's completed collaborations include:

- *Haydenfilms: May 2011 to November 2012.* The SPC worked with Haydenfilms toward the productization of Cemboo, an innovative new software as service (SaaS) that empowers content owners to control, monetize and distribute their own digital media content. The SPC worked with Haydenfilms to create the software infrastructure, animation describing the product concept, product logo and identity, website for customer account administration, website copy and business plan for this new product.
- *Runoff Studios: May 2010 to January 2011.* Runoff Studios is a local game development company with an educational and environmentally friendly mission. The SPC worked with Runoff Studios to develop their first game, Face the Waste, now available in Apple's iTunes App Store. The SPC collaboration produced the game itself, the website for Runoff Studios, the logo and branding for Runoff Studios, and marketing strategies for Face the Waste.
- *MRG Power Laboratories: Sept 2009 to April 2010.* Maintenance Reliability Group (MRG) specializes in the development and implementation of predictive maintenance programs, as well as the application of diagnostic technologies for industrial equipment. The SPC worked with MRG Power Labs to develop software to interact with their new product, the Grease Thief Analyzer. The software automates necessary calculations and allows lab technicians to view the results on any networked computer. SPC members also collaborated with MRG Power Labs staff on existing branding, marketing, and business plans for the product.
- *Cruzstar: Sept 2008 through October 2009.* The SPC worked with Cruzstar to create the software for their new service, Cruzcourt. Cruzcourt is an online cafeteria system for small to mid-size businesses. The software the SPC developed provides the employees of CruzcourtŠs corporate clients with a secure, web-based ordering system through which they can order meals to be delivered to their offices.
- *WorkXPress: December 2008 through May 2009.* WorkXPress (formerly Express Dynamics) is an innovative business management software solutions provider located in Central Pennsylvania since 2002. Their WorkXPress software engine is designed to make developing custom business software easy, affordable and fast. The SPC collaborated with WorkXPress on their online marketing strategies.

6. Similar Models

The SPC combines some characteristics of traditional university-industry collaborations with the model of a business incubator, and in doing so, no longer fits neatly into either category. Santoro [15] characterizes industry-university technology relationships as including four components: 1) research support, 2) cooperative research, 3) knowledge transfer, and 4) technology transfer. Research support is not really occurring in the SPC, since the entrepreneurs are not currently supporting the university through funds or equipment. Cooperative research is perhaps a component in a broad sense, though most SPC projects do not include ground-breaking technology that one would typically consider research. Knowledge and technology transfer are certainly occurring, though not in the same sense as most university-industry partnerships where the relationship is centered on a highly specialized area of expertise (such as nanotechnology).

Using a classification of university-industry relationships that includes knowledge generation, ideas testing, technology development and problem-solving [16], [17], the SPC clearly falls into the category of technology development, which focuses on "improving or developing specific technologies relevant to commercial users" [16]. However, this type of relationship does not typically encompass the aspects of undergraduate education and cross-disciplinary engagement that the SPC emphasizes.

This emphasis on the educational component of university-industry collaboration seems to be emerging in Europe, where "in relation to the Lisbon strategy, interaction between education, research and innovation has been conceptualized as a "knowledge triangle", and the vital importance of education and training for innovation has been repeatedly emphasized" [18], [19]. This has led to university-industry partnerships with a focus on training, entrepreneurship and employability of the graduates [18]. In the US, there is a dramatic increase in interest in entrepreneurship education, which is manifesting itself in the form of new programs and "Entrepreneurship Centers" [1], but these lack the applied focus of the SPC.

The SPC also shares some characteristics with the concept of business incubators, an effective and rapidly growing model for spurring regional growth [20]. Hackett and Dilts [21] give this definition of a business incubator:

> A business incubator is a shared office space facility that seeks to provide its clients (i.e., "portfolio" or "client" or "tenant companies") with a strategic, value-adding intervention system (i.e., business incubation) of monitoring and business assistance. The incubator can control and link resources that assist in the development of its clients' new ventures, and simultaneously helps contain the cost of their potential failure.

By this definition, it is clear that the SPC is not a business incubator, since we do not provide office space. We do, however, share many of the same goals of incubators, in terms of attempting to assist young companies and foster regional economic development. The SPC also adds to the incubator model the aspects of concrete assistance with the development of software prototypes and product identity.

7. Sustainability

The SPC was initially founded by a three-year infrastructure grant from the Pennsylvania State System of Higher Education. Since this time, the center has been sustained by gifts from the Innovation Transfer Network and individual donors, the university (primarily in terms of student wages), as well as an additional grant focused on a specific collaboration. Because the SPC does not charge the entrepreneurs for services rendered, sustainability of the model is a serious concern.

Various revenue models have been considered. The possibility of changing the model to one where the entrepreneur would pay at least nominal fees during the project has been investigated, but rejected. The Faculty Steering Committee had serious reservations about the exchange of money altering the nature of the relationship between the entrepreneur and the students, changing it to one of client to consultant and lessening the mentoring and collaboration components. The Community Advisory Board was firmly against the idea for two reasons: 1) the potential liability of having the undergraduate students being unable to deliver on a piece of a project, and 2) the perception that the SPC could be taking potential business from other small entrepreneurs in the region (such as designers, software developers, and so forth) once revenue was involved.

Another possible avenue for sustainability is to begin entering contracts with collaborating entrepreneurs based on their future revenues. This is a variation on the model of for-profit incubators that take a percentage of equity in a company in exchange for a spot in the incubator [22]. Rather than taking equity in the company, the concept would be based on the company's success. That is, if the company crossed a predefined threshold in profits from the product, they would then be expected to pay either a dollar amount or a percentage back to the SPC. This way, the SPC would still be able to take on very early stage entrepreneurs, the entrepreneurs would not feel that they were giving away their company before getting off the ground, and the nature of the team collaboration would hopefully be unaltered.

8. Conclusion

Although sustainability of this model is a concern, and is still being investigated, there are signs that the benefits of this type of "grassroots" economic development are being recognized, and increased support is becoming available [22]. The SPC model achieves several benefits simultaneously: 1) strengthening the regional economy by assisting fledgling businesses (and hopefully thereby creating local jobs), and 2) providing valuable practical learning opportunities for undergraduate students in multiple disciplines.

References

[1] S. A. Zahra, L. R. Newey, and J. M. Shaver, "Academic advisory boards' contributions to education and learning: Lessons from entrepreneurship centers," *Academy of Management Learning & Education*, vol. 10, no. 1, pp. 113–129, March 2011.

[2] P. Dorazio, "Professional advisory boards: Fostering communication and collaboration between academe and industry," *Business Communication Quarterly*, vol. 59, no. 3, pp. 98–104, September 1996.

[3] S. R. Genheimer and R. L. Shehab, "A survey of industry advisory board operation and effectiveness in engineering education," *Journal of Engineering Education*, vol. 98, no. 2, pp. 169–180, April 2009.

[4] A. Hsiao and S. Brown, "The role of an advisory board in the incubation stage of a technology-based start-up," in *Proceedings of the International Conference on Intellectual Capital, Knowledge Management & Organizational Learning*, October 2009, pp. 148–155.

[5] P. J. Horan, "Establishing an advisory board for the family business," *Journal of Financial Service Professionals*, vol. 57, no. 6, pp. 61–65, November 2003.

[6] R. W. Lingard, "Teaching and assessing teamwork skills in engineering and computer science," *Journal of Systemics, Cybernetics & Informatics*, vol. 8, no. 1, pp. p34–37, 2010.

[7] J. Schilling and R. Klamma, "The difficult bridge between university and industry: a case study in computer science teaching," *Assessment & Evaluation in Higher Education*, vol. 35, no. 4, pp. 367–380, July 2010.

[8] S. Goel and V. Kathuria, "A novel approach for collaborative pair programming," *Journal of Information Technology Education*, vol. 9, pp. 183–196, 2010.

[9] L. Lipponen, "Exploring foundations for computer-supported collaborative learning," in *Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community*, ser. CSCL '02, 2002, pp. 72–81.

[10] H. Jun, B. Butler, and W. King, "Team cognition: Development and evolution in software project teams," *Journal of Management Information Systems*, vol. 24, no. 2, pp. 261–292, 2007.

[11] M. Schilpzand and L. Martins, "Cognitive diversity and team performance: The roles of team mental models and information processing," in *Academy of Management Annual Meeting Proceedings*, August 2010, pp. 1–6.

[12] M. E. Elfman, B. Hayes, B. Boyer, and K. Michel, "Scientists and artists: "Hey! you got art in my science! you got science on my art"," in *Proceedings of the Institute of Nuclear Materials Management Annual Meeting*, July 2010, pp. 1–7.

[13] J. Musil, A. Schweda, D. Winkler, and S. Biffl, "Synthesized essence: What game jams teach about prototyping of new software products," in *Proceedings of ICSE: International Conference on Software Engineering*, May 2010, pp. 183–186.

[14] M. Zyda, "Computer science in the conceptual age," *Communications of the ACM*, vol. 52, no. 12, pp. 66–72, 2009.

[15] M. D. Santoro, "Success breeds success. the linkage between relationship intensity and tangible outcomes in industryŰuniversity collaborative ventures," *Journal Of High Technology Management Research*, vol. 11, no. 2, pp. 255–273, 2000.

[16] M. Perkmann and K. Walsh, "The two faces of collaboration: Impacts of university-industry relations on public research," *Industrial and Corporate Change*, vol. 18, pp. 1033–1065, 2009.

[17] S. Kabins, "Evaluating outcomes of different types of university-industry collaboration in computer science," in *Academy of Management Annual Meeting Proceedings*, August 2011, pp. 1–6.

[18] T. Thune, "Success factors in higher education-industry collaboration: A case study of collaboration in the engineering field," *Tertiary Education & Management*, vol. 17, no. 1, pp. 31–50, 2011.

[19] H. Shapiro, J. H. Haahr, and I. Bayer, "Background paper on innovation and education," Prepared for the European Commission, DG Education and Culture, Tech. Rep., 2007.

[20] T. O'Neal, "Evolving a successful university-based incubator: Lessons learned from the ucf technology incubator," *Engineering Management Journal*, vol. 17, no. 3, pp. 11–25, 2005.

[21] S. Hackett and D. Dilts, "A systematic review of business incubation research," *Journal of Technology Transfer*, vol. 29, pp. 55–82, 2004.

[22] D. Romero, "Incubators," *Entrepreneur*, vol. 37, no. 5, pp. 68–77, May 2009.

INSIGHT: Infusing System Design and Sensor Technology in K-12 Education

M.L. Neilsen[1], N. Bean[1], G. Singh[1], J. Spears[2], V. Wallentine[1], and N. Zhang[3]
Dept. of Computing and Info. Sciences[1], Center for Science Education[2], Dept. of Bio. and Ag. Engineering[3]
Kansas State University
Manhattan, KS, USA

Abstract

This paper describes an innovative, new GK-12 STEM Fellowship Program that incorporates contemporary embedded real-time sensors and system design into the existing K-12 curriculum. Unlike other GK-12 programs, more focus is placed on Technology (T) and Engineering (E), and less focus is placed on Science (S) and Mathematics (M). The underlying goal of the program is to link real-time embedded systems research with science and technology curriculum and to create a community of learning, teaching, and mutual support between the higher and pre-college education participants from rural backgrounds.

Keywords: Education, embedded systems, inquiry, outreach, real-time sensor networks.

1 Introduction

There is a well-recognized national need to inspire K-12 students to prepare for and pursue careers in Science, Technology, Engineering, and Mathematics (STEM) disciplines. These disciplines are needed to support innovation and the information age. Virtually all engineering disciplines have seen an increased emphasis on the use of computing technology. It is important for us to increase the number of students from the K-12 level interested in pursuing STEM careers with an emphasis on computing technology. When students get excited about science and engineering as a result of experiences in school or informal education settings, they are more likely to pursue classes that properly prepare them for success in undergraduate and graduate programs in STEM fields. Thus, there has been an increased interest in developing programs that are designed to enhance teacher preparation and classroom support for the type of experiences and content themes that inspire students to view STEM fields as an achievable, exciting option for them.

There has been an attempt to better understand why there has been a decrease in the number of Computer Science courses taught at the K-12 level in the US [1]. The top three reasons cited by K-12 teachers include: the rapid changes in technology, a lack of staff support, and a lack of curriculum resources. As noted by Shreck and Latifi, it is important to develop an infrastructure in which teachers have adequate support and where they are able to change with the technology. This will help them to maintain the interest of K-12 students in Computer Science [2].

As these programs and curriculum resources are developed and assessed, it is important to disseminate the results of such initiatives so that those with similar goals can consider how to adopt or adapt the successful elements of other programs into their own programs. Although many of the details are omitted, the goal of this paper is to report on our GK-12 program and some of the innovative curriculum modules that have been developed and used successfully in a K-12 setting.

The U.S. National Science Foundation's program to support Graduate Teaching Fellows in K-12 Education (GK-12) strives to improve graduate students' communication, teaching, collaboration, and team building skills through professional training, interactions with faculty, and work in the classroom with K-12 teachers and students. By working with K-12 teachers to integrate their knowledge and research to enhance the classroom, graduate fellows and faculty members also have the opportunity to build partnerships with schools and teachers, and to enrich learning opportunities and increase motivation for K-12 students. The graduate fellows also serve as role models for the students that they work with, and they talk with the students about the diverse and exciting careers that can be pursued by those who are interested in STEM disciplines. Although the main focus of the GK-12 program is on the development of graduate students, this paper will focus more on the innovative aspects of the program and the modules developed to date for use in the K-12 classroom.

2 INSIGHT GK-12 STEM Fellowship Program

Infusing System Design and Sensor Technology in Education (INSIGHT) is the title given to an innovative GK-12 STEM Fellowship Program at Kansas State University. Our program focuses on integrating real-time embedded systems and sensor technology with computing and information science through a standards-based science, technology, and engineering curricula.

The underlying goals of the program are to: enhance the usefulness, practicality and relevance of sensor, computing, and information technology education by linking embedded systems research for fellows to science and technology curriculum, and practice for classroom teachers; support technology in rural Kansas through two of the most important aspects impacting rural life in Kansas: agriculture and health; improve the teaching and learning of technology and engineering design in elementary through high school classrooms; and create a community of learning, teaching and mutual support between the both the higher education and pre-college participants from rural backgrounds.

Project activities team GK-12 fellows with K-12 STEM teachers through summer and academic year training and orientation, and place the fellows in the classrooms of rural Kansas schools. In the summer, project staff provides fellows with training in hands-on sensor-driven systems, STEM concepts and development, Kansas Curriculum Standards, and classroom instruction methodology. Participating teachers also have two weeks of training and orientation focusing on real-time sensor technology, computing and information science topics, selected science and technology content areas and the use of appropriate pedagogical and assessment strategies.

During the academic year, fellows support two participating teachers in the classroom an average of two times a week in their area with content-specific sensor technology and computing and information sciences. Program staff provides semester-long professional development opportunities via guided research and investigations of practical applications of technology integration on agricultural farm fields and within the classroom. Weekly meetings between project staff and fellows provide supervision and feedback.

Sensor systems are poised to revolutionize the way that the physical world is monitored and field experiments are performed with remote, automated real-time data collection and feedback replacing traditional manual methods. The development of cyber-physical infrastructure represents the next step in enabling applications wherein physical entities (humans with body parameters such as heart and respiratory rates, crops with different fertility and growth rates, etc.) and cyber-subsystems collaborate and interact to achieve a common goal.

For example, in health-care systems, the cyber-infrastructure can augment the capabilities of the hospital staff in patient monitoring, issuing alerts, and coordinating usage of resources. Likewise, in rural Kansas, remote monitoring can enable elderly residents to stay in their own homes safely for an improved quality of life and an on-site pharmacist is replaced with a robot that can dispense prescriptions to elderly patients and allow patients to consult with a pharmacist remotely.

As another example, although farm equipment operators can operate with a local visual view of the field, cyber and remote sensing infrastructure in the field can assist them by providing a correlated geographic information system (GIS), climatic and vegetation data to support variable rate application of chemicals with precision using the global positioning system (GPS). This results in both economic and environmental benefits [3].

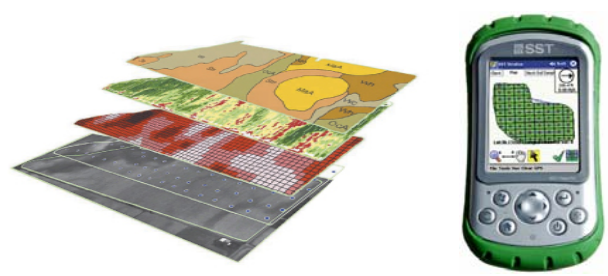


Figure 1. Variable-rate technology [SST Software]

Typically, these systems are difficult to develop because their development requires knowledge about many parts of a complex system involving a number of heterogeneous subsystems and components [4]. Their design is often a multidisciplinary exercise involving a variety of domain experts with different views of the system, and there are few formal techniques that can be used to address the integration of individual components. Designers often work on subsystems without fully understanding its impact on other components and the rest of the system.

Design of such real-time embedded sensor systems has been the focus of researchers from several departments at Kansas State University.

This program represents a unique synergistic opportunity for us to collaborate with our K-12 colleagues in a similar manner, and create and strengthen mutually beneficial partnerships with the many rural school systems in Kansas. These partnerships enhance the education of K-State's technologically-oriented graduate and undergraduate students while simultaneously advancing computing, science, and technology education in rural Kansas middle school and high school classrooms.

The program places an average of eight graduate students each year in up to eight different rural Kansas schools twice a week to assist an average of sixteen K-12 school teachers per year in integrating sensor, computing, and information technology into standards-based science and technology curricula and instruction. Participating teachers and fellows attend a two-week pedagogical summer institute in computing, problem solving, engineering and science. Participating teachers receive support in computing and technology content areas during the academic year. Science and technology teachers are selected from each of the participating schools and receive additional training and on-site support. During the five year program, approximately 2800 under-served rural and Hispanic students will benefit from the enhanced computing, science, and technology instruction. Sensor technology is the enabling element that pervades the entire science, engineering and technology curriculum, rather than as an entirely new and separate subject or curriculum area, whose introduction would be more problematic. Deductive reasoning, analysis and synthesis, algorithmic problem solving and design, and inquiry techniques are at the heart of each of these disciplines. Regardless of the scientific area, students must learn to formulate questions and hypotheses, plan experiments, conduct systematic observations, interpret and analyze data, draw conclusions and communicate results, using powerful classroom tools. Indeed, these skills are tested in a statewide assessment of students' achievement in science, engineering and technology. Aligning instruction with science, engineering and technology standards requires significant changes to classroom practice, from content, activities, and assessment to classroom management, interaction with students and learning tools. This program has helped to establish hands-on engineering and technology development as a foundational skill for vocational agriculture, and other areas. Instead of focusing on Mathematics and Science, the novelty of this project is on its primary focus on Technology and Engineering. In the Environmental Science and Natural Resources Section in the Kansas Standards for Agricultural Education, an important new standard is on Sustainable Agriculture.

Sustainable Agriculture

3 2 1 0	1.	Explain sustainable agriculture (LA)
3 2 1 0	2.	Describe sustainable ag. practices (LA)
3 2 1 0	3.	Describe the use of nutrient management (LA)
3 2 1 0	4.	Explain site specific agriculture and the use of GPS (S)
3 2 1 0	5.	Demonstrate GPS/GIS use (S, E)

Figure 2. State Agricultural Education Standard

Unfortunately, many of the rural school districts in Kansas don't have the resources or technical expertise to go beyond the basics of just reading about sustainable agriculture. The PI of this project was a founding member of an innovative company devoted to information technology in precision agriculture, called the Site Specific Technology Software (http://www.sstsoftware.com/). In addition, researchers at Kansas State University have been at the forefront in developing sensor technology for agricultural and military applications.

One of the Kansas State Standards for Agricultural Education is on pest management:

> **Performance Element: Develop and use a plan for integrated pest management.**
> Develop pest management plans based on pest life cycles. Implement pest control plan with appropriate treatments. Evaluate pest control plan.

Little guidance is given with respect to developing an appropriate plan or applying those treatments in a targeted fashion to reduce environmental impacts. An important goal for all citizens of Kansas is to make agriculture sustainable and to minimize the environmental impacts of chemical applications by using variable-rate technology and appropriate sensor technology. By using the hand-held devices and GPS/GIS software as shown above, students are able to develop the skills necessary to generate site-specific prescriptions for pest or weed control and minimize the overall environmental impacts while improving the economics of a farming operation.

This project's focus on sensor, computing, and information technology education makes it an ideal vehicle to provide graduate STEM fellows with the opportunity to enhance their academic experiences. Students interested in investigating the integration of sensor technology and computing in science, engineering and other disciplines are the focus of our recruitment efforts. In addition to receiving financial support, the fellows gain hands-on experience in applying technologies that relate to their learning, research and future careers, while having an opportunity to appreciate the use of such tools in education, especially in rural K-12 settings.

The project provides graduate students with hands-on experience in the integration of sensor, computing, and information technologies in science, engineering and technology education while they engage teachers and students in active, standards-based technology education. This project allows us to continue working with the teachers and administrators at middle schools and high schools where we have already established successful collaborations. Fellows gain valuable instruction, communication, and interpersonal skills as they work with teachers and students in their classrooms to integrate sensors and computing into the curriculum. Participating teachers, many of whom are inexperienced in working with sensors and computing technology, receive contemporary instruction in sensors, computing and information technology, as well as support from university faculty and fellows. The presence of fellows in their classrooms two times a week assists teachers in the tasks associated with ensuring the coherent delivery of instruction. To facilitate the rural experience, we incorporate video conferencing, using Adobe Connect and Skype, with the most rural classrooms. Some classrooms are several hundred miles from campus, so it is not economically feasible to place fellows in those classrooms twice a week. Although, we do plan monthly visits, other contact will be through video conferencing using Adobe Connect. The added advantage is that recorded interactions can be used in other settings, and the curriculum resources that are developed continue to grow. Rural students acquire new skills and experiences that will encourage them to study the subjects necessary to pursue careers in computer science or technology.

3 Sample Curricular Modules

In this section, we give a brief overview of a few modules that have been developed and/or delivered by fellows in the K-12 classroom. Details can be found on-line at, http://gk12.cis.ksu.edu, through our GK-12 program web-site.

Water Filter

In this activity students were asked to create their own sediment water filter using a water bottle and some basic materials. Some of the materials include: flour, sugar, sand, gravel, plastic beads, cotton balls, etc. The students were only allowed to use three materials and it was up to them to create the best filter in the class based on the types of materials selected, and the order in which the materials were placed into the filter. The dirty water, shown in the upper-left corner in Figure 3, is stirred occasionally to keep the sediment suspended, and the sediment shown in the upper-right is used to test filtered samples.

Figure 3. Sediment water sensor and filters

Wireless Sensor Network to Measure Sediment

At the high school level, students take samples using a hand-held sensor (top left) and take readings using sediment sensors deployed in the field and connected by a three-tiered wireless sensor network

[5,6,7]. A solar panel is used to power the middle tier of the wireless sensor network (lower right).

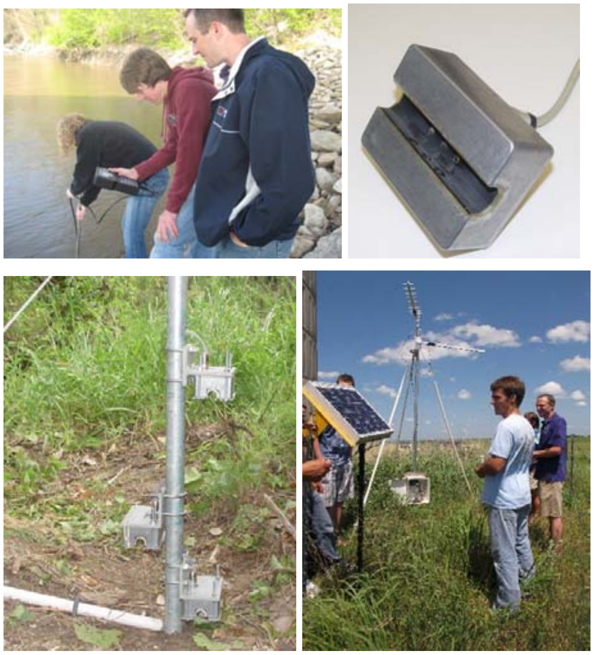


Figure 4. Wireless sensor network

To measure sediment discharge, turbidity sensors developed here at Kansas State University are organized into a wireless sensor network, and they continuously monitor sediment discharge. The system is organized to automatically adjust sensor reading rates based on the data to limit the power requirements of the wireless sensors. The data is then transmitted to a wireless base station, and then on to a centralized database from which the data can be analyzed [5,6,7]. Sediment concentration is defined as the weight of suspended soil particles per unit volume of water. Turbidity is usually referred to as the optical properties of suspended/dissolved materials in water on transmitting, reflecting, absorbing, and scattering light. Thus, traditional turbidity sensors are not sediment-concentration sensors. A sediment sensor developed in this study uses LEDs that emit lights at three visible and infrared *feature wavelengths*, which were selected through a spectroscopic analysis, with light detectors arranged at different angles from the light sources. Statistical models established based on test data allowed the sensor to be basically insensitive to non-soil, suspended and dissolved objects, such as algae, organic matter, and various microorganisms, and less sensitive to soil texture. A prototype sensor was tested at combinations of four water types and five soil textures in the laboratory. Statistical and neural-network models successfully predicted sediment concentration across samples of all the combinations with R^2 values of no lower than 0.95. An outdoor experiment proved that the influence of ambient light on sediment measurement can be largely eliminated by modulating the lights. More than ten prototype sensors of different designs have been fabricated and calibrated. Several sensors were placed at low-water crossings at Ft. Riley and Ft. Benning for long-term, sediment-runoff monitoring [6]. The sensor case has been modified to improve its waterproof capability. Difficulties encountered during the long-term tests included signal drifting and occultation of the optical lenses by algae and soil particles. Modifications in sensors and software have been made to solve these problems [5].

Water Sediment Concentration

At the elementary level, sediment concentrations in water are measured manually using filter paper and drying equipment after collecting samples at a local lake. Students also compare the results they obtain with measurements taken using an electronic sediment sensor.

Figure 5. Sediment concentration measurement

Newton's Laws of Motion

Seventh grade students, using Wii remotes, bungee cords, and toy trucks, test Newton's Laws of Motion and gain a better understanding of acceleration. There are four different variable stations: mass vs. velocity, friction vs. velocity, force vs. velocity, and force vs. acceleration. Acceleration is measured using the accelerometers embedded in WiiMotes via Bluetooth and appropriate software.

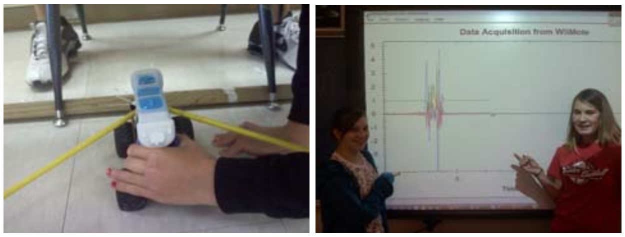


Figure 6. Newton's Laws of Motion

Olympic Bar Acceleration during Bench Press

High school students in weight lifting classes at Wamego High School use Velcro to strap on a Wii remote to an Olympic bar to measure each student's bar acceleration in all directions while doing the bench press lift. A graph of the accelerometer values is projected onto the ceiling of the weight room so that students can watch their acceleration and movements during the lift. The red line represents the number of repetitions, the blue line represents the left/right acceleration, and the green line represents the up/down acceleration. It was a great experience for the weightlifting classes.

Figure 7. Bench press acceleration

Variable Rate Technology

High school students analyze the effectiveness of applying pest-controlling chemicals at a variable rate to control pests, both in terms of economics and environmental impact. The power of the software lies within its innovative analysis functions, which allow producers to gain precise management information from field-level data stored in a Geographic Information System (GIS). GIS software links information to its location. Layers of information are organized, stored, analyzed, and queried for management decision-making. A farm field boundary is used to organize all pertinent information relating to the field, such as soil types, soil fertility, yield results, hybrid/variety selections, remotely-sensed imagery, aerial photos, chemical applications, field scouting, and more [3].

Access to this information enables people to make better management decisions, lower input costs, provide better stewardship of the land, and increase yields. Chemical applications can be selective and focused. Crop stresses caused by pests can be pinpointed long before they are visible to the naked eye. It doesn't matter if you are approaching precision agriculture from the perspective of producer, retailer, agronomist, or consultant. If the technology is properly applied it can benefit everyone.

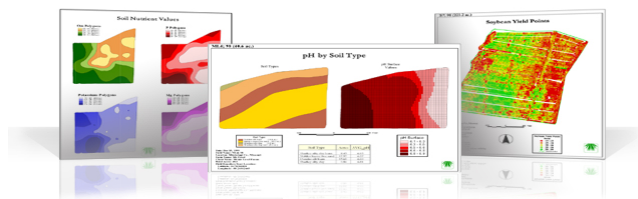


Figure 8. Variable rate technology

For the example hands-on classroom activity, students use a PDA (SST Field Scout) with built in GPS to create a field boundary, and delimit well locations and other hydrologic features (terraces – yes, there are terraces in Kansas, even though it is flatter than a pancake). From point data gathered in the field and analyzed in a lab, or obtained from yield monitors on-board modern combines, the software can be used to compute acreage and generate yield and fertility surface layers in graphs as shown above. In addition, point in polygon analysis can be used to compute average yield by soil type or hybrid seed type, along with a host of more detailed statistics and enhanced yield data processing tools. These GIS layers can be combined with satellite imagery, aerial photography and ortho-photos as base layers for field boundaries.

Based on the information gathered, the students can derive a variable rate prescription map to apply only the necessary chemicals at precise locations in the field to control the insect infestation. This prescription can be delivered to a farm input supplier to apply the chemicals at a precise variable rate across the field.

Scratch Programming

Middle school students learn the basics of object-oriented programming using Scratch, available for free from http://scratch.mit.edu [8].

Figure 9. Scratch programming

Styrofoam Cup Speaker

Figure 10. Styrofoam cup speaker

Eighth grade physics students learn about sound waves and electricity by building foam cup speakers. A similar foam cup speaker design can be found at: http://cse.ssl.berkeley.edu/lessons/indiv/regan/speakerlab.html.

4 Conclusions

Overall, the INSIGHT GK-12 program has been very effective and we have received very positive feedback. In this second year of the program, we plan to provide a smoother transition for the participating teachers and fellows just joining the program. We believe that by having a cohort of veterans as mentors will smooth the transition for incoming participants.

Other programs, such as MOBILIZE, led by Dr. Debra Estrin from UCLA, allow students to pursue scientific studies through the use of participatory sensing technologies; e.g., cell phones. We also have several activities that involve the use of Android cell phones. Linking science to commonly used devices provides students with a concrete look at the power of computational technology in daily life [2]. This can also motivate students to realize how Computer Science is required to develop novel systems.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 0948019. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] David Nagel, "Computer science courses on the decline", The Journal: Transforming Education through Technology, Aug. 4, 2009.

[2] Stephanie Shreck and Shahram Latifi, "K-12 computer education deficiencies in Nevada", in Proc. of the 2011 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'11), pp. 114-118, 2011.

[3] M.N. Rao, D.A. Waits, M.L. Neilsen, "A GIS-based modeling approach for implementation of sustainable farm management practices", Environmental Modeling and Software, Vol. 15, pp. 745-753, 2006.

[4] J. Hatcliff, G. Singh, D. Andresen, and S. Warren, "Observations and directions for high-confidence sensor networks in the real world", In Proc. of the Composable Systems Technology for High Confidence CyberPhysical Systems Workshop, 2007.

[5] N. Wang, N. Zhang, and M. Wang, "Wireless sensors in agriculture and food industry--Recent development and future perspective", Computers and Electronics in Agriculture, Vol. 50, No. 1, pp. 1-14, 2006.

[6] Y. Zhang, N. Zhang, G. Grimm, C. Johnson, D. Oarrd and J. Steichen, "Long-term field test of an optical sediment-concentration sensor at low-water stream crossings (LWSC)", *ASABE Paper No.* 072137, St. Joseph, Michigan, 2007.

[7] J. Wei, N. Zhang, D. Lenhert, M.L. Neilsen, M. Mizuno, and J. Schmidt, "Using smart transducer technology to facilitate precision agriculture systems", In Proceedings of the ASAE Annual International Meeting (ASAE Paper 03-3145), Las Vegas, NV, July 27-30, 2003.

[8] John Maloney, Mitchel Resnick, Natalie Rusk, Brian Silverman, and Evelyn Eastmond, "The Scratch Programming Language and Environment", *Transactions on Computer Education*, 10, 4, Article 16, pp. 16:1-16:15, Nov. 2010.

[9] M.L. Neilsen, D.H. Lenhert, M. Mizuno, G. Singh, J. Staver, N. Zhang, K. Kramer, W.J. Rust, Q. Stoll, M.S. Uddin, "Encouraging interest in engineering through embedded system design", In American Society of Engineering Educators (ASEE) Computers in Education Journal, Vol. XV, No. 3, pp. 68-77, July 2005.

40 Years of Processor Design in One Semester

G. Dueck
Department of Mathematics and Computer Science, Brandon University, Brandon, Manitoba, Canada

Abstract - *Logic-circuit design tools along with the programmable assembler introduced in this paper allow students to cover more ground in less time. The programmable assembler and its IDE enable and support programming at both the microcode and macrocode level. In addition, the assembler can be programmed to recognize the syntax of different assembler-language dialects as disparate as Intel, PDP and BAL.*

Features of the programmable assembler are described. Techniques to combine features in order to recognize and encode source-language statements are presented. Some of the ways in which the programmable assembler is used in a computer organization course are also discussed.

Keywords: Assembly language programming, computer organization, computer science education, microprogramming, processor design, logic-circuit simulation

1 Introduction

With great hubris, undergraduate courses in computer organization cover instruction-set architectures, memory architectures, function units, instruction pipelines, addressing modes, assembly language programming and microprogramming. Given this rich set of topics, it is possible for any instructor to start at almost any talking point and wander forth productively in almost any direction with students hanging onto his coat tails. Along the way, references will be made to this feature or to that of some micro processor or to some other mainframe, abundantly imaged in the instructor's mind but received – opaquely and somewhat distorted – by the student. What sort of pedagogical framework can we provide upon which to hang these rich forays into computer-organization design problems?

Forty years ago, processor architects worked on paper with Boolean logic and circuit diagrams. They created experimental realizations on breadboard circuits. They laid out circuit diagrams using line tape which they then photo reduced to fabrication size. In the present day, although we can and do teach the processes that lead from specification to design for combinational circuits and finite-state machines, it is still the case that experimental constructions reveal the flaws in our processes and the shortcomings of our designs, and allow us to experience the fulfillment of building a working model. In this way, the pedagogical imperative of learning by doing is attained.

The options for laboratory work in a computer-organization course range through paper-and-pencil design, simulation languages, purpose-built simulators, hardware simulators, logic simulators, hardware, FPGAs and VLSI fabrication. Indeed, there is an entire journal issue devoted to "specialized computer architecture simulators that see the present and may hold the future" [1] . At Brandon University, the computer organization course has evolved through three approaches to exercises in proving processor designs: a purpose-built simulation program, a hardware simulator, and Logisim [2], a logic simulator. It is the latter approach, in conjunction with a novel programmable assembler (described in this paper) that has made possible elaborate realizations of ground-up processor designs.

As a regular assembler, the programmable assembler produces binaries for execution on experimental processors. Pressed into service as a micro-code assembler, it aids in the development of simplified decoder designs and enables exploration of multi-cycle processor designs. Using its language-recognition facilities, the programmable assembler is used to explore existing instruction set architectures in the dialect of their native assemblers. It has even been conscripted into the related systems programming course to provide visualizations for static binding problems.

Assembly languages go hand in hand with processor design. The purpose-built simulation program mentioned above made use of spreadsheets to capture micro-control settings and encode micro-control words, a usage foreshadowed in [3]. Assemblers and related tools are also used for exploring programmable single-chip systems [4] and embedded systems [5]. To enable students to go farther and faster, hand-built test programs no longer suffice. Beyond the scope of the subject at hand lies the admission that even assemblers are inadequate to explore systems that require high-level languages and operating system frameworks.

This paper introduces some features of the programmable assembler and shows how the assembler fits with the requirements of a computer-organization course.

2 The Programmable Assembler

The novelty of the programmable assembler (PA) is not just that it produces both microcode and macrocode for a variety of architectures, but that it does so by starting from an empty language slate. By analogy, this kind of starting point is not so unusual for graphical design programs, which usually start with a blank canvas. But it is quite unusual to find a graphical design program that starts with an empty toolbox. Applying this analogy to the programmable assembler, a meta-language in the form of assembler directives is used to define an assembly language that meets some instruction-set specification. Reversing the analogy, it is as if the graphics program has a meta-toolbox with which to design an application-specific set of graphical tools, which are then used to create a graphical design.

Assemblers normally implement a language translator that recognizes source statements for a particular instruction-set architecture (ISA) and encodes them into machine-language binaries. PA provides meta-language tools that allow source code to control encoding mechanisms and define new encoding rules. It also allows control of recognition mechanisms for assembly language syntax. These facilities are used to create assembly languages for three purposes in a computer-organization course: generate microcode, create problem programs and define dialects for ISA analysis. The following sections show how the meta-language is used in each of these three domains, and hopefully demonstrate how PA can be used to help students embrace processor designs by approaching them more rapidly and seeing more clearly the points at which they can be extended.

2.1 Micropogrammable decoders

Decoders can be the most difficult part of a processor design. In the days of the 4004 and 6502, with designers coming from the background of relatively well-ordered memory-chip layouts, the decoder was referred to as “random logic” [6][1]. This is where dirty tricks and clever designs were used to save transistors and clock cycles.

The problem is that decoder designs rapidly become inaccessible to students. Mano & Kime [7], in their wonderful pedagogical evolutionary path from single-cycle to multiple-cycle processors, present a simple decoder for the former and an unrealizable decoder for the latter. That is, it is not realizable with the tools provided, and the textbook seems to all but walk away from it in pursuit of more fruitful discussion.

Classroom experience shows, however, that just the suggestion of replacing the combinational logic of the single-cycle decoder with a lookup table simplifies random-logic design to the point where students begin to ask their own what-if questions. Moreover, provoking them with a working lookup-based decoder for the multi-cycle processor soon has them writing their own micro-code and proposing instruction definitions. This provides a natural segue into addressing modes and ISA types, and CISC vs. RISC processors.

Microprogramming a decoder is not a natural starting point for introducing assembly language programming. Instead, regular assembly language will be discussed first, before showing how the same assembler can be used to produce microcode.

2.2 Regular-Instruction Assembly

In the context of the programmable assembler, regular-instruction assembly programming refers to the level of programming that would normally be considered the domain of an assembler. As usual, a regular instruction consists of label, mnemonic opcode and operands. However, as mentioned earlier, PA is a clean slate with respect to regular assembler instructions.

Underlying PA is an engine that combines regular-instruction operands with meta-language definitions. Here, meta-language refers to the language implemented by directives that interact with assembler mechanisms. That is, meta-language definitions describe the regular-instruction language.

Like regular assembler instructions, directives consist of a mnemonic instruction and a comma separated list of operands. Directives are distinguished from regular assembler instructions in the usual way using a dot prefix on the mnemonic. Operands of directives are called specifications, to distinguish them in discussion from operands of regular instructions.

The first directive encountered tutorially is the field directive. A field is named by its first operand and typed by its second. Notationally, fields are referred to by their type using adjective terms like lbl-type field or opc-type field.

There are several types of field definitions in the meta language of PA. Those of interest at this point are insertion-field definitions that simply provide specifications for bit fields of various widths in an encoded instruction. An example instruction set [7] encodes to a 16-bit word, comprising a 7-bit opcode and three 3-bit operands used for registers, immediate values and branch addresses. This leads to the following insertion-field definitions.

```
.field  opc,opc=opcodeTable,9,7
.field  dr,tbl=regTable,6,3,0,1
.field  sa,tbl=regTable,3,3,0,1
.field  sb,tbl=regTable,0,3,0,1
.field  imm3,lbl,0,3,0,1
.field brh,rel,6,3,0,0,3
.field brl,rel,0,3,0,1,0
.field  word,lbl,0,16,0,1
.field  autoemit,emit,0,16
```

[1] This reference is clearly dated, but in a course that must cover all of the territory from combinational logic to processor architecture, it is well to remember that students must recover much of the ground painstakingly won by their predecessors.

An insertion field has up to 7 specifications of which the last 5 are integers that define the offset and length of the destination field, the insertion value, the incrementer and the source offset. Trailing specifications may be omitted if they are zero.

Each insertion field type performs a different function, but in reality they differ only in how they acquire an insertion value. Once acquired, insertion fields use the value in the same way: the value is inserted into emitted data, using the offset and length specifications to denote where and how much, and using the source offset to denote where from.

Opcode directives introduce specifications that allow a regular-instruction mnemonic opcode to bind to a list of field definitions. The field definitions are then applied in turn to operands of the regular instruction.

Many insertion fields take their insertion value from the source-language operand to which they are applied. Importantly, the same operand can service more than one field. This is the purpose of the increment specification, which indicates how to advance (or regress) through the operand list.

To show specifier-to-operand binding in action, a slightly more complex instruction will be used. The regular-language BRZ instruction has two operands: a register and a label. However, the resolved label will be encoded into two 3-bit fields. There must be a field definition for the opcode, the register and the two label fields. Two more definitions are required.

```
.def     opcodeTable,brz,0x60
.opcode  brz,opc,sa,brh,brl,autoemit
```

The first specifier of .opcode associates the regular instruction with the opcode definition. The remaining specifiers bind to previously defined field specifications. A source statement like the one that follows, with instruction mnemonic brz, is associated with the opcode definition above.

```
brz      r3,endloop
```

Following association, each of the two regular operands (r3, label) are matched in turn with the field definitions (opc, sa, brh, brl, autoemit). opc (not incremented) inserts the value of brz found in the opcodeTable (0x60); sa (incremented) is matched with operand r3; and both brh (not incremented) and brl (incremented) are matched with operand endloop. Finally, autoemit (not incremented, unmatched, doesn't care) writes the 16-bit value of the encoded instruction to the output stream.

Field definitions cause regular-instruction operands like r3 and endloop to interact with other assembler mechanisms. From the example, note that these operands match field types tbl and rel. Register references like r3 are constrained by the tbl-type field specifier to a list of register names in the key set of regTable, which in turn maps register names to insertion values.

The other operand in the example, endloop, is constrained by the matching rel fields (brh and brl) to be a symbol in the regular symbol table of the assembler; hence endloop must be a defined label or a constant. Fields of lbl type constrain matching operands just like rel fields but there is a difference in how the value is used. A lbl field simply inserts the value it is given. A rel field inserts the difference between the value and the location counter, modified by adding rel's own value specification. This permits the encoding of relative-address references and accommodates designs that choose between pre- and post-execution program-counter incrementing.

It is now clear that an opc-type field actually matches the regular-instruction mnemonic rather than an operand, constraining the mnemonic to the key set of opcodeTable and mapping it to values found in that table.

To complete the example, the field directives are preceded by some set-up directives to define the opcode and register tables. (Other setup directives define the word size and endedness of each memory section, how to split memory sections for loading into parallel memory devices, and how to format the listing.)

```
.table    opcodeTable
.table    regTable
.defrange regTable,r,0,8
.def      regTable,sp,7
.def      regTable,bp,6
```

The field types introduced above comprise almost the entirety of insertion-field types provided. Indeed, there are only two more: con and disp. The con field type silently inserts a constant value into a bit field, but there is a keyword-style operand format that allows regular assembler statement to provide an override value. The disp field type acts like rel, but uses a base label rather than the location counter as the relative-to operand in the value calculation.

The emit field will be revisited, and other non-insertion fields will be introduced, in the section on CISC instruction set architectures.

2.3 Microprogramming, revisited

Microprogramming differs from regular assembly programming only in minor ways. The word sizes tend to be larger; there are more fields; sometimes there are field-overlay tricks, for example when don't-care subfields are pressed into other service. However microprogramming essentially boils down to this: enable control lines that are important during a particular cycle, ensure that all others are disabled. In this context, the assembler's job remains the same: produce a stream of encoded binary values suitable for installation in a memory device.

Micro word fields fall into a small number of categories: control lines for multiplexor selector inputs, typically 1 to 4 bits; write and other enables, typically 1 bit; register addresses, typically 2 to 5 bits; function-unit selector inputs, typically 2 to 5 bits. It should be evident that the insertion-field directives used to specify regular (macro) assembler instructions can also be used to specify micro instruction formats.

At the risk of sacrificing reader comprehension for parsimony, assume – in place of a full presentation of field directives and logic-circuit diagrams – micro-word con-type field definitions, with default insertion values all zero, for multiplexor selectors MA, MB, MF, MM; write enables RW and MW; function selector FS, and register addresses AX, BX and DX with suitable offsets and lengths along with other field definitions introduced as the need arises. (Register addresses AX, BX and DX serve only to override register addresses SA, SB and DR from the macro instruction when the former are non-zero.) Then simple micro-words to implement (independent) register transfers like

$$R[DR] \leftarrow R[SA] - \overline{R[SB]} + 1$$
$$M[SA] \leftarrow R[SB]$$

can be specified as

```
.opcode mcadd,RW=1,FS=0b0101
.opcode mcstore,MW=1
```

where con-overriding keyword=value notation is used.

Each of these definitions is used exactly once by regular-language statements to populate a micro-word at a decoder ROM location determined by the 7-bit opcode, mnemonically referenced in the regular instruction. Thus, we have

```
.def opcodeTable,mcadd,0x05
.org opcodeTable.mcadd
mcadd
.def opcodeTable,mcstore,0x20
.org mcstore
mcstore
```

Multiple-cycle instructions require a decoder with its own version of a program counter (MPC) and the ability to stall the main program counter (PC) through its two control lines (PS). A fetch cycle uses control line MM to multiplex the PC onto the memory-address bus and a single bit instruction-register load enable (IR). IR also controls a multiplexor in the decoder that loads MPC from the regular-instruction opcode field or from a next-address field (NA) in the micro word. Thereafter, microinstructions return to the fetch state simply by virtue of a zero-valued NA. Fetch is specified by 4 control lines.

```
.opcode fetch,mm=1,il=1,ps=0b01
```

NA is implemented by a lbl-type PA field. N-cycle instructions have their first micro-word at the address corresponding to their opcode. The second and subsequent instructions are located in a higher section of the decoder ROM, at locations not reachable by opcode *cum* address. Thus a two-cycle instruction like LDW, with register transfers

$$R[8] \leftarrow M[PC], PC \leftarrow PC + 1$$
$$R[DR] \leftarrow M[8]$$

where the address of a memory-resident operand is in the second word of a 2-word instruction, is implemented by two ROM words, one at the opcode address, the other one located at the highest ROM address generated thus far.

```
.field  na,lbl,24,8,0,1
.def    opcode,ldw,0x41
.org    opcode.ldw
.opcode ldw,dx=8,rw=1,md=1,mm=1,ps=0b01,na
.opcode ldw1,ax=8,rw=1,md=1
ldw     $
.org    $
ldw1
```

Here, $ is a label operand that yields the highest generated location-counter value in the current memory-output stream. As an operand to ldw (the micro, not macro version), the $ operand is matched by the na field to encode the operand's value in the microcode. Changing the location counter to $ (using .org) causes the second microword, defined by ldw1, to be encoded and stored there.

A final example shows the 5-cycle call instruction using the label expression *+1 to link each of the second through penultimate micro-word to the next micro-word.

```
.opcode  call0,rw=1,fs=0b0110,ax=0b1010,na
.opcode  call1,rw=1,dx=8,md=0b10,na
.opcode  call2,mw,ax=8,bx=8,na
.opcode  call3,rw,ax=8,mm=1,md=1,na
.opcode  call4,ax=8,ps=0b11
.org     opcode.call
call0    $   ; dr ← dr - 1
.org     $
call1    *+1 ; r8 ← PS
call2    *+1 ; m[dr] ← r8
call3    *+1 ; r8 ← M[ps]
call4      ; ps ← r8; na=fetch
```

The important thing about this last example is the way in which normal assembler bookkeeping related to location counter and symbol table is leveraged to link together micro-control words.

2.4 Programming CISC ISA assembly recognizer

There are other ways in which PA can help in a computer organization course: to reinforce concepts like addressing modes and n-address designs; to review the way in which designers have constructed existing instruction sets; and ultimately, to compare the way in which instruction set design affects the performance of a processor in terms of clock cycles necessary to accomplish a task and the related performance measure of memory accesses. To do this, a student has to become familiar with different instruction sets. One of the lab exercises has the student constructing an assembler, necessitating a review of all features of an instruction set. Although assemblers can be constructed from ground up, or using toolkits, the already familiar PA can be pressed into service.

Addressing modes are nicely illustrated by reviewing PDP-11 modes. An additional instructional ISA based loosely on IBM 360 (called HAL 306) is used for further examples and, in another course, to explore the concepts of static binding and relocation. The point of these example choices is to show how PA can be used to demonstrate underlying principles in disparate assembly languages. As an aside, PA has been programmed to explore forays into recognizing IBM (BAL 360) and Intel (8086) assembly source.

PDP-11 is interesting because of its orthogonal separation of address modes and execution. On the one hand, the function unit operates on just two inputs. On the other hand, those inputs can be acquired in 8 different ways from each of two different sources for a total of 64 combinations of input access. The addressing modes are inferred at assembly time by the implications of language-specific operand notation. Compare, for example, PDPs "movb (r1)+,@mem(r2)" with 360's "STM 14, 12, 12(13)". Accommodating different language-specific source-code syntax requires bending PA to each new use.

PA is extensible by design. New mechanisms can be added to the framework by extending the internal object hierarchy to control those mechanisms. Then, the mechanisms are brought into the programmable meta language by adding a field type to introduce and control the framework mechanism. Finally, a regular language is introduced whose recognition and encoding is defined by the revised meta language.

For the purpose of this discussion, a new mechanism incorporating regular-expressions is introduced via a new (.pattern) directive. Then a new field type (pat) is introduced to control the application of the pattern mechanism to regular language operands. Actually a pat field specifies not one but a list of patterns, arranged from stronger to weaker.

Recall that a .opcode directive specifies a list of fields and that each field is applied in turn to operands of a regular language statement. When a pat-type field is applied to an operand, each pattern is tested in turn until a pattern match succeeds.

A pattern directive specifies more than just a regular expression. A pattern also contains a list of fields (by name) and operands (both simple and via back reference into regular-expression groups) which are used to modify both the field list and the operand list. This has some of the flavour of self-modifying code, but it is preferable to think of it in terms of programming by continuations.

For the PDP-11 example, a pattern recognizes the unique syntax associated with each addressing mode, extracts the register, and rewrites the operand list to include the register and a mode by name. These are then bound to two fields in the existing field list that will encode the extracted register operand and then encode the extracted mode.

For the HAL 306 example, this mechanism is leveraged to insert not just additional operands, but additional fields. It is well to dwell on this for a moment. The additional operands are analogous to placing new stepping stones in one's path, and the new fields are like donning new shoes appropriate to the stone just placed.

To explore the HAL 306 example, contrived though it may be, consider the following regular-language statements written in HAL 306 assembly.

```
mov r1,r2          ; r1 ← r2
add r3,word1       ; r3 ← r3 + M[word1]
sub word2(r4),r5 ;M[r4+word2]←M[r4+word2]-r5
```

Each instruction has a primary register (from the example they are in order r1, r3, r5) and a secondary register (r2, r0 (by inference), r4). The second two statements have a memory reference – the first direct, the second indexed. The exact value of the opcodes (mov, add, sub) is not known until after both the operand formats have been recognized.

To match the first operand, a pat-type field specifies three patterns: one to recognize a register operand, one to recognize direct memory references and one to recognize indexed memory references. The pattern that matches a register operand inserts not only the extracted register but a new pat-type field that specifies three more patterns: one to recognize a secondary register, one to recognize a direct memory reference and one to recognize an indexed memory reference. Whichever of these is successful will insert operands (reg or label or both reg and label in the case of an indexed operand) and appropriate fields. But more importantly, the successful pattern must also insert an opc-type field that specifies the appropriate opcode table – appropriate, that is, for the operand pair recognized by the path.

Although apparently complex at a Byzantine level, with the use of appropriate tutorial materials students have little difficulty mastering the concepts introduced here.

2.5 The Assembler Desktop

Logisim provides a graphical design environment. Although the programmable assembler can be used from the command-line, it is provided with a companion desktop that also provides a graphical design environment to manage the many small files associated with a project.

Aside from the obvious benefits, the most important feature of the assembler desktop is to provide simple interactions for frequent tasks. The desktop GUI and Logisim interact through the system clipboard. With this mechanism, debugging a design in Logism, making a change to assembler source, reassembling, and then reloading the Logisim memory device becomes the work of a few keystrokes.

3 Summary

First, some observations about the programmable assembler will be made, followed by some observations about its role in course delivery.

i) On the surface, PA is programmable at two levels: the meta-language level of definition, and the source-code level of regular assembly language. Viewed, however, as a framework, PA is also programmable at a third level, in which the class hierarchy of its implementation is leveraged to expose new mechanisms at the meta-language level.

Secondly, fitting the meta-language into the constraints of assembly-language definitions creates a certain lack of expressivity that would be found in higher-level specification languages. In a sense, like all assemblers, PA still finds a role as a target for those languages.

Finally, although the style of programming with continuations might be somewhat out of place in a computer organization course, it is a mechanism behind a mechanism. From this perspective, it is analogous to the use of linear algebra in the SPICE circuit simulator, used in turn as the basis of some hardware simulators.

ii) Turning to the role of PA in course delivery, the practical lab work in a computer organization course requires the construction, or at least the review, of accessible designs for processors and related peripherals. Because processors are intended to execute programs, exercising processor designs requires construction of programs. Simple programs can be created by hand, but complex processors require more than simple programs.

The advantage of Logisim is that it can be used to create working processors at a level of abstraction appropriate to a third-year computer organization course. The advantage of the programmable assembler presented here is that it can be used to define microcode, then assembler language and then programs that can be executed on those working processors. In conjunction with the assembler desktop, and in a nod to the jargon of the subject matter, the frequency of the clock used to measure design cycle time can be increased. Aside from the old argument against debugging systems into existence, this would appear to be an advantage.

The challenge of a discipline whose fundamental nature is that of change is to find appropriate points of entry and suitable levels of abstraction that enable students to successfully embrace basic principles that drive the discipline. The realization that post secondary education is less about learning tools, and more about learning ideas and how they fit together, influences the choice of tools.

Ultimately, in the context of computer organization courses, simple tools that fit together well, like PA and Logisim, provide payback by allowing students to cover more ground in less time.

4 References

[1] W. Yurcik. “Special issue on specialized computer architecture simulators that see the present and may hold the future”, Journal of Educational Resources in Computing. Journal on Educational Resources in Computing (ACM), Mar 2002, pp 1-3.

[2] C. Burch. “Logisim: a graphical system for logic circuit design and simulation”, Journal on Educational Resources in Computing (ACM), 5—16., Mar 2002.

[3] D. Shinners-Kennedy. “Using spreadsheets to teach computer science”, SIGCSE '86 Proceedings of the seventeenth SIGCSE technical symposium on Computer science education, (ACM) , 264—270, Feb 1986.

[4] P. Mishra, A, Shrivastava, & N. Dutt. “Architecture description language (ADL)-driven software toolkit generation for architectural exploration of programmable SOCs”, Transactions on Design Automation of Electronic Systems (ACM), 11, 3, 626—658, Jul 2006.

[5] Chang, & I. Lee, “Embedded system hardware design course track for CS students”, Microelectronic Systems Education (IEEE), 49—50, 2011.

[6] I. Shirakawa, N. Okuda, T. Harada, S. Tani & H. OzakiA. “Layout System for the Random Logic Portion of an MOS LSI Chip”, Transactions on Computers (IEEE), 30, 8, 572—581, Aug 1981.

[7] Mano, M. M., & Kime C. Logic and Computer Design Fundamentals, *4/E* (Prentice Hall), chapters 9 and 11, 2008.

Virtualization for a Cyber-Security Laboratory

Guillermo A Francia III, Aaron Garrett, and Thomas Brookshire
Jacksonville State University
Jacksonville, Alabama, USA

Abstract - *Virtualization has become an integral part of the businesses of today. The virtue of virtualization rests on its ability to cut down cost and to provide an effective means of managing Information Technology (IT) resources. The purpose of this study on virtualization is two-fold: first is to find an effective way to deliver online pedagogical materials and exercises, and second is to compare the merits of various virtualization systems using a common platform. We present our findings on this paper and direct the reader to avenues of possible future research directions.*

Keywords: Cybersecurity, virtualization, virtual desktop infrastructure (VDI), security laboratory.

1 Introduction

Virtualization has become an integral part of most businesses and is becoming more pervasive in several sectors of society. It has cut down on costs and increased revenue dramatically. The virtue of virtualization rests on its ability to cut down cost and to provide an effective means of managing IT resources. The purpose of our study on virtualization is two-fold: first is to find an effective way to deliver online pedagogical materials and exercise, and second is to compare the merits of various virtualization systems using a common platform.

The concept of virtualization began with the mainframe computers of the 1960's [1]. At that time, the prohibitive cost of hardware forced companies to institute time-sharing systems, which allow multiple users to make use of a single computer system [2]. To accomplish this, a piece of software called the virtual machine monitor layer (e.g., VMware) sits directly on top of the hardware and provides an abstraction for the operating system [1]. This abstraction allows the operating system to treat the virtual machine as if it were the hardware itself. A fortunate by-product of this early approach was that user processes were protected from one another, since each was run in a different instance of the virtual machine (often called a sandbox) [3].

This sandbox approach to partitioning users was available to some degree on non-virtualized systems. Subsequent developments in operating systems allowed processes to be tied directly to the users who spawned them, and new modes on the underlying processors provided distinction between regular and supervisory users [2]. This allowed a system to provide some access control in dealing with processes. However, since this was handled at the operating system level, the approach no longer worked when users began to require multiple different operating systems on the same machine. This led to a resurgence of virtualization in the late 1990's [2], [1].

The rest of the paper is organized into five sections. First, we provide a literature review on the applications of virtualization. Next, we examine a small selection of virtualization systems that are currently available. Third, we describe virtualization security on a level-by-level perspective, fourth we present two virtualization configurations with which we based our study, and finally, we discuss the metrics that we developed and present our performance scorecard.

2 Virtualization Applications

Virtualization technology is an integral part of the now-commonplace cloud computing paradigm. According to a survey in 2007 from The Strategic Counsel, a market research firm, "at least 39% of worldwide organizations with more than 500 employees" have adopted server virtualization [4]. Similarly, a more recent survey found that 51% of companies had virtualized at least half of their workloads and 63% plan to do so by the end of 2012 [5].

2.1 Virtualization in Business

One great thing about virtualization is that it allows businesses to do a test on upgrades to their equipment. Most small to medium businesses are unable to afford a whole other test lab with identical servers, in order to test new upgrades to software. It is crucial to test upgrades; otherwise, an upgrade can cause massive amounts of havoc in the once-stable environment. Having a whole other area full of servers just for testing purposes, is just too costly. Virtual managers (VMs) have the ability to convert the physical hardware and software into a new virtual environment. This gives the system administrators the ability to fully test out the new upgrade(s) without any major repercussions such as those instances when data gets deleted or users are locked out. There are a lot of companies that do not even have their equipment upgraded because they do not have the money to test out the upgrades. A risk assessment comparing the cost of an upgrade versus the cost of recovering from a system compromise would reveal that a company would be better off not performing the necessary upgrade. In this situation, top level managers do not understand that they are leaving their systems wide open for attack. Having the ability to fully test out upgrade(s) and not have to worry about data is almost invaluable in the grand scheme of things [6].

With all the recent buzz about going green and helping save the Earth, virtualization is a great way to have a company go green and save millions each year. It’s the quintessential double whammy. Everybody wins. As quoted from VMware, “For every server virtualized, customers can save about 7,000 kilowatt hours [of electrical expenses], or four tons of CO2 emissions, every year.” That is an insane amount that cannot be ignored. Companies can go virtual and get some great publicity out of it [6].

Virtualization provides a multitude of immediately tangible non-security-related benefits to companies. Virtualization allows companies to use their processing resources more efficiently [3]. A significant amount of a server’s resources remain unused for lengthy periods. Virtualization can be used to consolidate the processing needs onto fewer servers, which also decreases maintenance costs, hardware costs, and power consumption [3] [7]. Also, as mentioned previously, users benefit from virtualization’s ability to perform dynamic load balancing, which ensures that no single user monopolizes the system resources [3]. Those hardware resources also have greater software compatibility due to the virtualization abstraction [1]. Even more impressive, virtualization can provide the ability to migrate running virtual machines between different hardware platforms or easily “rollback” a machine to a previous stable state [1]. Virtualization technologies have been developed and deployed in diverse fields, including education [8] [9], medicine [10], and emergency management [11].

A much more novel use of virtualization is being developed for mobile devices [12]. In such cases, a company would be able, for example, to allow an employee to have a single smart phone provides separation between home and work profiles [13]. An addition of a virtualization layer on mobile devices would also allow handset manufacturers to develop systems for multiple types of hardware easily, as long as the hardware supported the embedded hypervisor [13].

2.2 Virtualization in Education

Virtualization in academia has likewise taken great strides. Stackpole, et al. [14] described a virtual laboratory environment of heterogeneous computing infrastructures that are implemented for hands-on coursework in system administration. Similarly, the use of a User-Mode Linux (UML) as a virtualization administration tool is described by Begnum, et al. [15]. Stewart, Humphries, and Andel [16] described an application of virtualization in cybersecurity education. The application utilizes network virtualization that enables virtual machines to connect and realize a network of complex topologies for a cyber-warfare education and training platform.

3 Virtualization Systems

According to VMware, virtualization is defined as “the separation of a resource or request for a service from the underlying physical delivery of that service”[17]. As it turns out, there are two types of virtualization. One is referred to as a “thick client”. A thick client is a machine that already has an Operating Systems (OS) on it. It also has a program on it that manages the virtual machines (e.g. VMware Workstation, VirtualPC). The other type is referred to as a “lean hypervisor”. It is where the OS is pretty much taken out, leaving very little between the guest OS and physical hardware. Having the lean hypervisor drastically decreases the amount of overhead to run and increases the amount of CPU available by 95%. The lean hypervisor actually acts like a very basic form of an OS, allocating the necessary amounts to the VMs [18].

The hypervisor is the key to the server’s maximum utilization process. It is basically the considered to be the manager of physical resources. One example is when the hypervisor notices a VM idling and uselessly running up the CPU, it will restrict how much CPU is given to that particular VM so that others will have more. Another example is when a certain amount of RAM is allocated to a particular VM and the VM is not currently using it, the hypervisor will de-allocate the RAM in order to free it up for some other VM. Hypervisor systems are also known as “bare metal” systems. They have greatest scalability between the thick client and the bare metal systems. One of the great things about virtualization is the fact that you can just load up an image each time you need a new VM up and running. This increases response time dramatically and in return helps save the company a lot of money. Any time something happens to one of the applications on a VM or the VM itself, conflicts, errors, and vulnerabilities are bound to arise. A quick re-imaging of the VM is a great quick fix for when the server is in need of a clean VM [17].

3.1 VMWare® ESXi

One of the greatest things about ESXi, besides it being inexpensive, is its size. It is approximately 100MB. This means it can be ran from a simple jump drive so as to allow the hard drive space on the server to be fully utilized. The second factor is security. Not having the complete OS means not having most of the vulnerabilities. The third would be maintenance. Since the OS has been stripped out, there are hardly any patches to apply. In fact, ESX, the proprietary version, is recorded as having 10 times more patches than ESXi, not because ESXi is relatively inexpensive, but because it does not need them. ESXi is primarily used for hosting servers. However, it can host regular desktop VM's as well. One key factor about ESXi is that the VM's cannot be managed at the machine. You have to use the VMware vSphere Client to remote into the ESXi server. Then you can manage the VM's and other configurations. This is what you would want as the foundation for all other VM servers/desktops.

3.2 Virtual Bridges™ VERDE

VERDE is a great proprietary server/lab manager. It's primarily for building a virtual desktop infrastructure (VDI). VERDE has to sit on top of an OS. This means it needs to be diligently patched as soon as a security patch becomes available. It gives you the ability to manage the virtual desktops (VDs) and other configurations through a web browser. One great feature about Verde is that it "seamlessly integrates with Active Directory, Novell E-directory and other Directory Services for AAA." [19]. Another great feature about VERDE is that it supports both Windows and Linux systems. VERDE even has high scalability. It allows you to cluster up to 10,000 VERDE servers, which are all stateless. On the hosts' side, it is a fairly simple setup. All you have to do is ensure Java and the appropriate VERDE tools are installed. Its use of Gold Images is nice, too. When the administrator finalizes a VM and makes it available to the other users, it will force them to open a non-persistent version of the VM and anything they do will not be saved.

3.3 VirtualBox

VirtualBox [20] is an open-source virtualization system. Much like VERDE, it needs an OS in order to run. It's primarily used for desktop VM's but can also run servers. One nice feature is that it does not require newer processor features (e.g. Intel VT-x or AMD-V) in order to run. Another great feature is its portability in its images. It does not matter if you are running VirtualBox on Windows, Mac, or Linux. If you save an image, it can be ported over to any other OS and still work. VirtualBox, although an open-source system, is very-well maintained and documented.

3.4 Citrix® XenServer

XenServer [21] is a virtualization platform that comes with a free edition. One excellent feature is its ability to integrate Active Directory. The great video playback is also a good feature. It is recorded as even rivaling physical PCs. Another great feature is that it has an integrated hardened SSL VPN. It also gives the ability for live migration.

4 Virtualization Security

Several security benefits of virtualization have already been discussed, including isolation and encapsulation. However, virtualization is not a panacea, and it can introduce security issues of its own. Not only must an organization be familiar with security issues relevant to virtualization, but it must also modify their security strategy to closely match the change in technology [22]. In a survey by Information Week cited in [22], 70% of the respondents were using virtualization in some capacity, but only 12% had a security plan that took this into account.

In order to discuss the security issues related to virtualization, we will adopt the approach of van Cleeff et al. [23] to serve as a model for our discussion. First, they consider a system composed of underlying hardware on which one or more virtual machine monitors reside. This hardware may be in separate physical systems that are separated and connected via network communication. The VMMs may have multiple virtual machines running simultaneously. Monitoring and controlling the entire set of VMMs in the system is done by what van Cleeff et al. call the virtual machine monitors’ management (VMMM) [23]. Finally, the systems at various levels may interact in unintended ways that lead to emergent properties that may be vulnerable.

4.1 Hardware Level

At the hardware level, the first feature related to virtualization security is the ability to trap program execution [2] (which was mentioned above and is implemented in all modern x86 processors). This gives the VMM control over processes so that it can determine how each process should be handled, which is the mechanism by which isolation is achieved [23]. However, according to Bratus, et al., there is some question as to whether this trapping ability as commonly implemented enhances security [24].

The second feature is an optional hardware component called the Trusted Platform Module (TPM), which is now implemented by many manufacturers, that allows verification that a given VMM has permission to run on the hardware [23]. This is accomplished by storing the encryption keys for the system in the TPM. In order to gain access to these keys, the system must verify (usually by calculating a hash of its state) that it is in a trusted configuration [25].

4.2 VMM Level

At the VMM level, the first feature is the small footprint of VMMs relative to those of full operating systems [23]. This is a security benefit because the smaller the code base, the less likely it is that vulnerabilities are introduced and the more easily that they're found and fixed.

The second feature is the hierarchical control that should be present to enforce flow of information from more to less secure [23]. However, implementations of VMMs often provide guest-host interfaces that break this hierarchy and introduce vulnerabilities [26].

The third feature is process isolation, which prevents a process in one VM to affect a process in another VM [23]. This has been discussed several times above.

The fourth feature is the ability to perform inspection of the individual VMs, which could include logging, state inspection, verifying that a VM is authorized, disabling a running VM, making backups, and rolling back changes to a secure, stable state [23]. This inspection is done outside of the VM, which theoretically obviates the need for a VM to perform self-monitoring and thus be subject to malicious erasure or modification of the logs, etc.

The fifth feature is load balancing of VM resource needs [23]. This is not necessarily a security feature, but it can be thought of as such, especially in the event of denial of service attacks.
The sixth feature is the ability to define the communication privileges of a VM with other systems and the ability to manage multiple VMs from another machine [23]. However, these features introduce additional communication channels and points of attack [26].

4.3 VM Level

The first feature at the VM level is the ability to store a VM as an image. This has benefits in that it allows VM backup, rollback, and copying, but the stored images could be nefariously inspected or modified [23]. The second feature is the ability to modify the software running on the VM as needed to respond to security issues [23], such as system patching or upgrading.

4.4 VMMM Level

At the VMMM level, the relevant features are the ability to transfer running VMs from one physical location to another; the ability to replicate a VM on another physical server; the ability to do load balancing across different physical machines; the ability (and need) to manage system patches across multiple operating systems; and the need to manage multiple VMMs from a central location [23].

4.5 Emergent Level

Finally, at the emergent level, the relevant features are the loss of uniqueness of machines and data; the loss of location information of data; and the loss of monotonicity of program execution, due to the ability to backup, copy, and rollback VMs [23].

4.6 Empirical Results

A security analysis performed by Google [27] considered six different virtual machines, including VMware, Xen, and QEMU. The results were not favorable to the idea of virtualization as a panacea for security concerns. The systems considered were tested in several ways—source code audit in the case of open source products, stress testing with crashme , and fuzz testing [28] with a custom tool built by the author. The VMMs were judged in three failure categories—total compromise (arbitrary code can be executed on the host), partial compromise (sensitive information is leaked about the host), and abnormal termination (the host exits or enters an infinite loop). All but one, Xen, showed either full or partial compromise. The author states that Xen may be susceptible to full compromise in a particular configuration with hardware-assisted virtualization, but this could not be tested because of lack of appropriate hardware. However, the reason for concern was that Xen makes use of an emulator derived from QEMU in such configurations, and the author was able to show that QEMU (and possibly other popular variants, such as VirtualBox) are subject to several attacks that could lead to full compromise. The author's conclusions were that Xen should be considered a viable candidate for virtualization as long as appropriate precautions are taken.

5 Virtualization Configuration

Our main goal is to create a virtual playground for cyber-security students to hack computers without worrying about anyone getting into trouble. Our plan is to create a victim web-server so the students could attack it however they wished. We accomplished this goal with two different pieces of software, Virtual Bridges VERDE and VMware ESXi. We investigated every aspect of the two virtual systems and we settled on the following key features: footprint, cost, user-account control, authentication, VM power, and implementation.

VERDE has to sit on top of an operating system (OS). However, that means it only requires one physical machine in order to operate and maintain it. VERDE will install on Linux/Unix only, so for our host OS, we have chosen Ubuntu Server 10.04 LTS 64-bit. Other supported OS's include: SuSE, Red Hat and CentOS. This means we had to install the specified OS and VERDE Software. The VERDE software comes with its own web-server software. This means all you have to install is the OS, the VERDE software, and the required support software (e.g. Java JRE).

Unfortunately, VERDE is not free and would require a modest initial investment.

User-Account Control is managed through the underlying OS. As with any Unix system, users are assigned in groups and VMs can be assigned using these groups. The permissions that are set to the VM are what constitute what the users of that VM can do. The VMs are assigned to a "Desktop Policy". "Desktop Policy" is where the admin can setup whether or not the users can make persistent changes or not. It is also where you can assign certain users/groups to VMs.

VMware ESXi is considered a "bare metal" hypervisor. It is its own OS and web server. It doesn't take very long to install, which is great. The reason that the footprint ESXi is such a big deal is because of how small it is. It's approximately 100MB. It can even be booted from a jump/flash/thumb drive. VMware has stripped ESXi of most OS features to accomplish this. One great thing about it is that it increases the security of ESXi. Any vulnerabilities that came with the OS are now gone. Another great reason for this is optimization. Having ESXi operate on such a small size allows for full utilization of the server.

The cost of ESXi is considerably inexpensive compared to VERDE. For an academic institution, a nominal fee is annually charged for the use of ESXi. However, it does have limited features because of it. You can upgrade it to ESX for a price, but for our session, we had no need to.

Roles that are pre-existing are "Administrator", "No access", and "Read-only". ESXi provides the ability to allow the user to control a substantial amount of things. That allows the administrator to create any type of user.

Unlike VERDE, the target VM does not have to be physically powered on and visually available in order to attack it. Whenever a VM is powered off, it just means the graphical interface is offline. The target VM is still able to communicate with other VM's in this mode. This is perfect for setting-up a target web-server.

6 Virtual System Scorecard

The score card is relative to our experience with the two virtual systems in this scenario. It is subject to change depending on the situation and person in charge.

Installation - ESXi has such a simple installation. It took maybe five to ten minutes to install. VERDE on the other hand, takes longer because the host OS needs to be pre-installed.

Setup - ESXi has a setup that takes maybe two minutes. VERDE does a prerequisite check of all required software. It took much longer and was much more complicated than ESXi.

Cost – ESXi is relatively inexpensive, at least in an academic setting, compared to VERDE.

System Management – ESXi seems so much easier to manage than VERDE. This can be subjective since the terminologies that VERDE uses, which are Unix based, are not commonly known to Windows users.

Footprint – ESXi is a lean hypervisor and VERDE is a robust server sitting on a host OS.

Security – Since VERDE sits on top of Linux, that system needs to be diligently patched as needed. However, if the system administrator has the discipline to start correctly with an SE Linux version, the virtual system can be very secure. The ESXi,, contrary to popular belief that it is very secure because it has been stripped of its OS, is not completely hack-proof. The fact that it is also stripped of its ability to enforce Defense in Depth is a problem. However, this limitation can easily be remedied by putting the ESXi system behind a secured firewall.

Tech Support – The tech support for VERDE was phenomenal. They were more than willing to help with any problem and even setup live meetings to assist through the problems. ESXi, however, is community based. This means you can get help from other users or FAQ sites.

Hardware Compatibility – Both systems are compatible with a wide range of hardware types. However, just like any other VM server, the memory requirements of both systems are inordinate.

Documentation – The documentation on both are great. The manual for VERDE is very in-depth and helpful. ESXi has documentation that is also very helpful.

Scalability of Deployment – The ultimate goal for using virtualization is to provide a number of virtual computer systems for cybersecurity experimentation and analysis. As the number of participants in these cybersecurity exercises grows, so is the need for scalability. In this regard, VERDE, being a commercial product, should be able to sustain the scalability requirement. Most likely, we need to utilize the ESX server to satisfy any future scalability need.

	ESXi	VERDE
Installation	**A+**	**B**
Setup	**A**	**B+**
Cost	**A**	**B**
System Management	**A**	**B**
Footprint	**A+**	**B+**
Security	**A-**	**B+**
Tech Support	**C+**	**A+**
Hardware Compatibility	**A**	**A**
Documentation	**A**	**A**
Scalability of Deployment	**C+**	**A+**

Table 1. The Virtual System Scorecard

6 Conclusions and Future Plans

This paper presented an overview of virtualization and its applications in business and academia. We looked at two prominent virtualization systems and provided metrics to compare them in a cybersecurity laboratory setting. Although the scorecards provide guidelines for implementation, the reader should consider another factor that could determine the ultimate adoption of such systems: the cost of deployment. The deployment cost factor will include the cost of the servers, the power and cooling requirements, the imaging infrastructure, and the administrative support.

The challenge for the authors will be in the continual development of these virtual systems to support an even larger and complicated network of virtual servers for cybersecurity laboratory exercises.

7 Acknowledgements

This paper is based upon a project partly supported by the National Science Foundation under grant awards DUE-0726486 and OCI-0959687. Opinions expressed are those of the authors and not necessarily of the Foundation.

8 References

[1] Rosenblum, M. (2004). The Reincarnation of Virtual Machines. ACM QUEUE , 2 (5), 34-40.

[2] Crosby, S., & Brown, D. (2007, Dec/Jan). Virtualization Reality. ACM QUEUE , 34-41.

[3] Ray, E., & Shultz, E. (2009). Virtualization security. Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies (pp. 1-5). Oak Ridge, Tennessee: ACM.

[4] The Strategic Counsel. (2007). Global Server Virtualization Survey Summary Report. Toronto, Ontario, Canada: The Strategic Counsel.

[5] Babcock, C. (2011, October 31). Virtualization's Next Wave of Problems. Information Week .

[6] Hall, Jeffrey W. "Top 10 Things System Administrators Need to Know About Virtualization." (2011): 2-7. Information Security Research Library. Website: http:// searchsecurity.bitpipe.com. Last access: April 15, 2011.

[7] Esensten, J. (2011). Security Strengths and Weaknesses of Virtualization as a Green Computing Solution. Master's Thesis, University of Oregon .

[8] Bower, T. (2010). Experiences with virtualization technology in education. Journal of Computing Sciences in Colleges , 25 (5), 311-318.

[9] Li, P. (2010). Selecting and using virtualization solutions: our experiences with VMware and VirtualBox. Journal of Computing Sciences in Colleges , 25 (3), 11-17.

[10] Sawa, T. (2011). Leveraging Devices, Data and Discovery for Smarter Healthcare in Japan. Healthcare Information Research , 17 (3), 184-189.

[11] Kelly, S., Mazyck, C., Pfeiffer, K., & Shing, M.-T. (2011). A Cloud Computing Application for Synchronized Disaster Response Operations. IEEE World Congress on Services (pp. 612-616). Washington, DC: IEEE.

[12] Sena, J. A. (2009). Virtualization and Security: A Perspective. Issues in Information Systems , 10 (2), 513-519.

[13] Kroeker, K. L. (2009). The evolution of virtualization. Communications of the ACM , 52 (3), 18-20.

[14] Stackpole, B., Koppe, J., Haskell, T., Guay, L., Pan, Y. "Decentralized virtualization in systems administration education," Proceedings of the 9th ACM SIGITE conference on Information technology education, October 16-18, 2008, Cincinnati, OH, USA [doi 10.1145/1414558.1414619].

[15] Begnum, K., Koymans, K., Krap, A., & Sechrest, J. (2004). Using Virtual Machines in System Administration Education. NLUUG, the UNIX User Group – The Netherlands. In Proceedings of the 4th International SANE conference.

[16] Stewart, K. E., Humphries, J.W., Andel, T.R., "Developing a virtualization platform for courses in networking, systems administration and cyber security education," Proceedings of the 2009 Spring Simulation Multiconference, March 22-27, 2009, San Diego, California.

[17] VMware. "Virtualization Overview." 2-4. Information Security Research Library. Website: http://searchsecurity.bitpipe.com. Last access: April 15, 2011.

[18] Egan, David. "The Value of Virtualization." (2009): 6-9. Information Security Research Library. Web. 15 Apr. 2011. < searchsecurity.bitpipe.com >.

[19] Virtual Bridges, "VERDE VDI" Website: http://www.vbridges.com/products/verde/verde-vdi/ Last access; March 05, 2012.

[20] Watson, J. (2008). VirtualBox: bits and bytes masquerading as machines. Linux Journal , 2008 (166).

[21] Citrix, "XenServer 6" Website: http://www.citrix.com/lang/English/lp/lp_1688615.asp. Last access: March 6, 2012.

[22] Yunis, M. M., Hughes, J., & Roge, J. (2008). Real Security In Virtual Systems: A Proposed Model For A Comprehensive Approach To Securing Virtualized Environment. Issues in Information Systems , 9 (2), 385-395.

[23] van Cleeff, A., Pieters, W., & Wieringa, R. J. (2009). Security Implications of Virtualization: A Literature Study. IEEE International Conference on Computational Science and Engineering (pp. 353-358). Vancouver, BC, Canada: IEEE.

[24] Bratus, S., Locasto, M. E., Ramaswamy, A., & Smith, S. (2008). Traps, Events, Emulation, and Enforcement: Managing the Yin and Yang of Virtualization-based Security. Proceedings of the 1st ACM Workshop on Virtual Machine Security. ACM.

[25] Anderson, R. (2003, August). "Trusted Computing" Frequently Asked Questions. Retrieved December 13, 2011, from University of Cambridge Computer Laboratory: http://www.cl.cam.ac.uk/~rja14/tcpa-faq.html

[26] Bratus, S., Locasto, M., Ramaswamy, A., & Smith, S. W. (2010). VM-based Security Overkill: A Lament for Applied Systems Security Research. New Security Paradigms Workshop (pp. 51-60). Concord, MA: ACM.

[27] Ormandy, T. (2007). An Empirical Study into the Security Exposure to Hosts of Hostile Virtualized Environments. Proceedings of CanSecWest 2007. Vancouver, BC, Canada.

[28] Miller, B. P., Fredricksen, L., & So, B. (1990). An Empirical Study of the Reliability of UNIX Utilities. Communications of the ACM , 33 (12).

Star Schema Implementation for Automation of Examination Records

Er. Muheet Ahmed Butt[1], Dr. S. M. K. Quadri[2], Er. Majid Zaman[3]
Directorate of Information Technology & Support Systems, University of Kashmir, Srinagar, J&K, India
Head & Director, PG Department of Computer Science, University of Kashmir, Srinagar, Srinagar, J&K, India
Directorate of Information Technology & Support Systems, University of Kashmir, Srinagar, J&K, India

Abstract: *Data warehousing systems enable enterprise managers to acquire and integrate information from heterogeneous sources and to query very large databases efficiently. Data in a data warehouse does not conform specifically to the preferences of any single enterprise entity. Instead, it is intended to provide data to the entire enterprise in such a way that all members can use the data in the warehouse throughout its lifespan [7]. This work explores using the star schema for Automation of a Data Warehouse. An implementation of a data warehouse for an Examination Automation System is presented as an example.*

Keywords

Data Warehousing, Data Mining, Star Schema, Data Set.

1.0 Introduction

A 'data warehouse' is a repository of an organization's electronically stored data. Data warehouses are designed to facilitate reporting and analysis [5]. This classic definition of the data warehouse focuses on data storage. However, the means to retrieve and analyze data, to extract, transform and load data, and to manage dictionary data are also considered essential components of a data warehousing system [11]. These operations depend more on the way the data is stored..

There are two leading approaches to storing data in a data warehouse

i. Dimensional approach and
ii. Normalized approach

In the dimensional approach, transaction data are partitioned into "facts", which are generally numeric transaction data, and "dimensions", which are the reference information that gives context to the facts [9]. A key advantage of a dimensional approach is that the data warehouse is easier for the user to understand and to use. The retrieval of data from the data warehouse also tends to operate very quickly. The main disadvantages of the dimensional approach are:

i. in order to maintain the integrity of facts and dimensions, loading of data from different operational systems is complicated, and
ii. it is difficult to modify the data warehouse structure if the organization adopting the dimensional approach changes the way in which it does business.

In the normalized approach, the data in the data warehouse are stored following, to a degree, the Codd normalization rule. Tables are grouped together by subject areas that reflect general data categories. The main advantage of this approach is that it is very easy to add information into the database. A disadvantage of this approach is that because of the number of tables involved, it can be difficult for both users to join data from different sources into meaningful information and then access the information without a precise understanding of the sources of data and of the data structure of the data warehouse.

These approaches are not exact opposites of each other. Dimensional approaches can involve normalizing data to a degree [12]. In this paper we have implemented a Star Schema Model of a Data Warehouse of an Central Automation of Examination System catering many colleges, Departments, Courses, Subjects, Subject Groups, Marks and tried to prepare results notifications at various levels which will enable us to build a build a Decision Support Database for future analysis.

The rest of the paper is organized as follows: Section 2 provides the information pertaining to various Data Warehouse Schemas used with their advantages. Section 3 provides the design of an example Data Warehouse for Examination Automation System giving detailed attribute information pertaining to the fact table. Section 4 provides the overall association of various dimensional table with the fact table. Section 5 provides the association of the fact Dimension of the Star Schema implementation for this example with other Dimensions in the schema. It also provides the results of the simulations of said implementation. Section 6 provides the means for aggregation of data present in the Star Schema Data Warehouse Design for Decision Support Systems. Section 7 provides brief description about the On-line Analytical Processing (OLAP) capabilities provided by the data warehouse or data mart. Section 8 provides the brief comparison between the 3rd normal form and star schema implementation on the same test data. Conclusions drawn are depicted in Section 9. Section 10 lists the references and Appendix 1 provides the pictorial representation of the star schema and its relationship of fact tables with other dimensions.

2. Data Warehouse Schémas

A schema is a collection of database objects, including tables, views, indexes, and synonyms. There is a variety of ways of arranging schema objects in the schema models designed for data warehousing. The main database Schemas are:

2.1 Star Schemas

The star schema is perhaps the simplest data warehouse schema. It is called a star schema because the entity-relationship diagram of this schema resembles a star, with points radiating from a central table [6]. The center of the star consists of a large fact table and the points of the star are the dimension tables. A star query is a join between a fact table and a number of dimension tables. Each dimension table is joined to the fact table using a primary key to foreign key join, but the dimension tables are not joined to each other. The optimizer recognizes star queries and generates efficient execution plans for them. It is not mandatory to have any foreign keys on the fact table for star transformation to take effect. A star join is a primary key to foreign key join of the dimension tables to a fact table. The main advantages of star schemas are that they:

- Provide a direct and intuitive mapping between the business entities being analyzed by end users and the schema design.
- Provide highly optimized performance for typical star queries.
- Are widely supported by a large number of business intelligence tools, which may anticipate or even require that the data warehouse schema contain dimension tables.

Star schemas are used for both simple data marts and very large data warehouses.

2.2 Snowflake Schemas

The snowflake schema is a more complex data warehouse model than a star schema, and is a type of star schema [6]. It is called a snowflake schema because the diagram of the schema resembles a snowflake. Snowflake schemas normalize dimensions to eliminate redundancy i. e., the dimension data has been grouped into multiple tables instead of one large table. While this saves space, it increases the number of dimension tables and requires more foreign key joins. The result is more complex queries and reduced query performance. The main advantages of Snowflake schemas are that they:

- save memory space for data.
- increases the number of dimension tables and requires more foreign key joins.
- the result is more complex queries.

2.3 3NF Modeling

Third normal form modeling is a classical relational-database modeling technique that minimizes data redundancy through normalization [6]. When compared to a star schema, a 3NF schema typically has a larger number of tables due to this normalization process. 3NF schemas are typically chosen for large data warehouses, especially environments with significant data-loading requirements that are used to feed data marts and execute long-running queries. The main advantages of 3NF schemas are that they:

- Provide a neutral schema design, independent of any application or data-usage considerations
- May require less data-transformation than more normalized schemas such as star schemas

3.0 Designing Data Warehouse

An example of a record in a fact table for an Examination Automation System for a University, on a single event, such as a result of a Student at a particular session of an Academic year at Under/Post Graduate Level, has been considered.

In addition to the fact tables Table 1, there are also dimension tables in the database. These dimension tables describe the options to "cut" or view the data in the fact table. The star and snowflake schemas all use more than one dimension table in their database [2][3]. The records in a single dimension table represent the levels or choices of aggregation for the given dimension [7][17]. The classic data warehouse example used is the Result dimension [10][12]. The records in the Result dimension table will indicate that the fact table data can be aggregated by Subjects assigned, Enrollment of Students, Marks Obtained etc. Another dimension would be date. Using the date dimension we would be able to analyze data by a single date or dates aggregated by month, quarter, fiscal year, calendar year, holidays, etc.

For an Examination Automation System, a simple fact table would have the following column variables is show in table below.

Table 1: Fact Dimension Details

fact_DIM

ROLLNO key	Rollno Pertaining to a Session	Numeric
REGNO key	Registration No. of Student	Alphanumeric
RESULT	Calculated Result	Alphanumeric
TOTALM	Total Marks	Numeric
RESGAZ	Showing Statue Result	Alphanumeric
Sesson_ID Key	Session of Examination FKey	Alphanumeric
college_id Key	College Code FKey	Alphanumeric
dateID Key	Date FKey	Alphanumeric
course_code Key	Course Opted FKey	Alphanumeric
facultyID Key	Faculty Opted FKey	Alphanumeric
groupID Key	Subject Group Table FKey	Alphanumeric

4.0 Star Join Schema

The star join schema (also known as the star schema) is a database in which there is a single fact table and many dimension tables. These tables are not normalized. They are unlike traditional operational data bases where one attempts to normalize the tables [10][14]. In the fact table there is one segment for each dimension. The fact table uses a compound key made up of the group of the dimensions. In addition, the fact table usually contains additional variables which typically are additive numbers, i.e., numeric facts. In our Examination Automation System example the individual dimension table would capture views by:

- Enrollment containing registration no, name and parentage
- Subjects taken by the student
- Student enrolled in the course
- Marks obtained in every subject
- Date of declaration, session, year
- College information
- Course information
- Faculty details etc.

For the full star schema of Examination Automation System see Appendix 1 at the end of the paper.

5.0 Data Sets Building Using Star Schema

Users of the Examination Automation System will want to look at the data summarized to various levels. Joining selected dimension tables to the fact table will provide the user with a dataset on which to aggregate the needed information [1].

For example, to generate the result of the student would require a the join of five tables namely Fact Table, Enrollment Dimension Table, Course Dimension Table , subject_groups Dimension Table and marks Dimension Table. The resultant data file will then be aggregated by using the Proc Summary step to produce a dataset for analysis. Below is a demonstration of this approach.

An Examination Automation System of 2500000 records in the fact table with 12 column variables, totaling to 30 megabytes of space. The memory taken by the dimension tables are depicted in table below.

Table 2: Dimension Table records in Megabytes

DIMENTION NAME	SIZE
Enrollment	176.388 MB
Subject	0.056 MB
Marks	150.5 MB
Course	0.020 MB
College	0.008 MB
Date	0.015 MB
Faculty	0.012 MB
Group	0.095 MB
Session	0.010 MB
Total Space	**327.10 MB**

5.1 Generating the Final Result Notification

An algorithm was devleloped and code implemented in SQL using SQL Server Management Studio Express as Front end and Microsoft SQL 2005 at the back end for testing the described schema. The results of the simulation are presented in table below

Table 3: Client Statistics for the above query resulted in the following details.

Client Statistics Information	**Trial 3**	**Trial 2**	**Trail 1**	**Average**
Client Execution Time	20:25:51	20:25:46	20:25:36	
Query Profile Statistics				
Rows returned by SELECT statements	27453	27453	27453	27453
Network Statistics				
Number of server round trips	3	3	3	3
TDS packets sent from client	3	3	3	3
TDS packets received from server	1974	1974	1974	1974
Bytes sent from client	2220	2220	2220	2220
Bytes received from server	8074601	8074601	8074601	8074602
Time Statistics				
Client processing time	551	568	817	645.3333
Total execution time	859	861	1127	949
Wait time on server replies	308	293	310	303.6667

6.0 Building the Decision Support Database

Similarly, other datasets could be generated for analysis. Using the building blocks of the fact table and the various dimension tables, one has thousands of ways to aggregate the data. For expedient analysis purposes, frequently needed aggregated datasets should be created in advance for the users [15][16]. Having data readily and easily available is a major tenet of data warehousing. For Examination Automation System, some aggregated datasets were:

- Generating the Final Result Notification per Subject, College, Subject Groups, Year Wise, Gender etc.
- Remuneration for Paper Checkers, Checking Assistants and other Officials.
- Students Count by Age, Gender, Pass, Fail, Reappear in subject pertaining to per college, subject, year, group of subjects.
- Interests of Various of Students in Courses, Colleges, Subjects etc and Improvements to be made in the Education System etc.
- No of Students enrolled for a particular course, subject, college, courses within a college, subject within a college.
- Students who have passed with and without statues.
- Percentage of result, subject wise, college wise, course wise, group wise.

As one can see, the Star Schema lends itself well for Custom analysis.

7.0 OLAP and Data Mining

On-line Analytical Processing (OLAP) is the analytical capabilities provided by the data warehouse or data mart. One can view granular data or various aggregations of data for business analyses using graphical-user-friendly tools [4][18]. Data warehouse and data marts exist to answer questions and find business opportunities. There are many ways to analyze data using procedures such as Proc decodeMks, Proc getResult, Proc fmaster, Proc rollidx, Proc Tabulate.

Finally, data mining is the name given to newer statistical techniques used to explore voluminous data stores. These techniques include decision trees and neural networks. These methods, like neural networks, can sometimes handle co-linearity better than the older statistical techniques.

8.0 Comparison with 3rd Normal Form

A comparative study was also performed by taking same amount of test data and the observations were tabulated in the below mentioned table. It was observed that there was a big tradeoff between the memory and the speed in the implementation of 3rd Normal form and Star Schema.

Table 4: Execution Time for Result Preparation

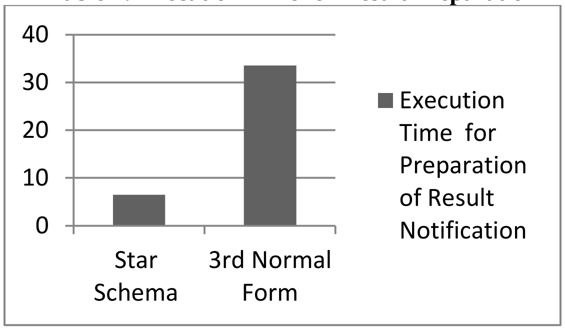
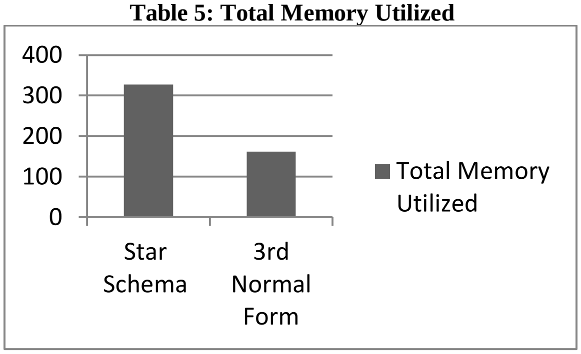


Table 5: Total Memory Utilized



9.0 Conclusion

The data warehousing technology i s gaining wide attention, and many organizations are building data warehouses (or, data marts) to help them in data analysis in decision for decision support. Data Warehousing is a newly emerged field of study in Computing Sciences. Due to its viz. multidisciplinary nature, it has overlapping area of studies in three different computing disciplines. This overlapping sometimes may cause contradictory definitions for a specific concept. To overcome this problem of data warehousing for Examination Automation System, it was considered for Star Schema Design. In this regard various functional dimensions of the Examination System were designed and connected to a Fact Transaction Dimension. Furthermore the general issues like the Client Statistics and Query Design were taken up and various Decision Support Databases were designed and implemented using the same star Schema.

10.0 References:

[1] A. Gupta, V. Harinarayan, and D. Quass. Aggregate query processing in data-warehousing environments. In Proc. 21th Int. Conf. on Very Large Data Bases, Zurich, Switzerland, 1995.

[2] ACM/ IEEE-CS Joint Task Force for Computing Curriculum 2005. “Computing Curriculum 2005”. The Over view report” 30 Sep, 2005

[3] C. Fahrner, and G. Vossen. A survey of database transformations based on the Entity-Relationship model. Data & Knowledge Engineering, vol. 15, n. 3, pp. 213-250. 1995.:

[4] CAI Yong, HE Guangsheng, “Designing Model of Data Warehouse with OO Method [J]”, Computer Engineering and Applications, 2003.6.

[5] Inmon, W. H., “Building the Data Warehouse”, Second Edition, John Wiley & Sons, Inc 1996

[6] Fon Silvers, “Building and Maintaining a Data Warehouse,” AN AUERBACH BOOK”, CRC Press is an imprint of the Taylor & Francis Group, an informa business

[7] Jeff Lawyer, Shamsul Chowdhury, “ Best Practices in Data Warehousing to Support Business Initiatives and Needs”, Proceedings of the 37th Hawaii International Conference on System Sciences – 2004

[8] Jorge Bernardino, Pedro Furtado, Henrique Madeira,” A Cost Effective Approach for Very Large Data Warehouses”, Proceedings of the International Database Engineering and Applications Symposium, 2002

[9] Kimball, Ralph, “The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses”, John Wiley & Sons, Inc, 1996.

[10] Krishna. “Principles of Curriculum Design and Revision: A Case Study in Implementing Computing Curricula CC2001”. ITiCSE’05, June 27–29, 2005

[11] Larry, Greenfield, LGI Systems Inc., "The Data Warehousing Information Center," 1997 pp http://pwp.starnetinc.com/larryg/index.html.

[12] LIN Yu,etc, “The Principles and Applications of Data Warehouse [M]”, Posts & Telecommunications Press, 2003.1

[13] R. Barquin, and S. Edelstein. “Planning and Designing, the Data Warehouse”,. Prentice Hall, 1996.

[14] REN Jinluan, GU Peiliang, ZENG Zhenxiang, “Research on the Methods of Designing Data Structure of Data Warehouse [J]”. Computer Engineering and Applications, 2001.22.

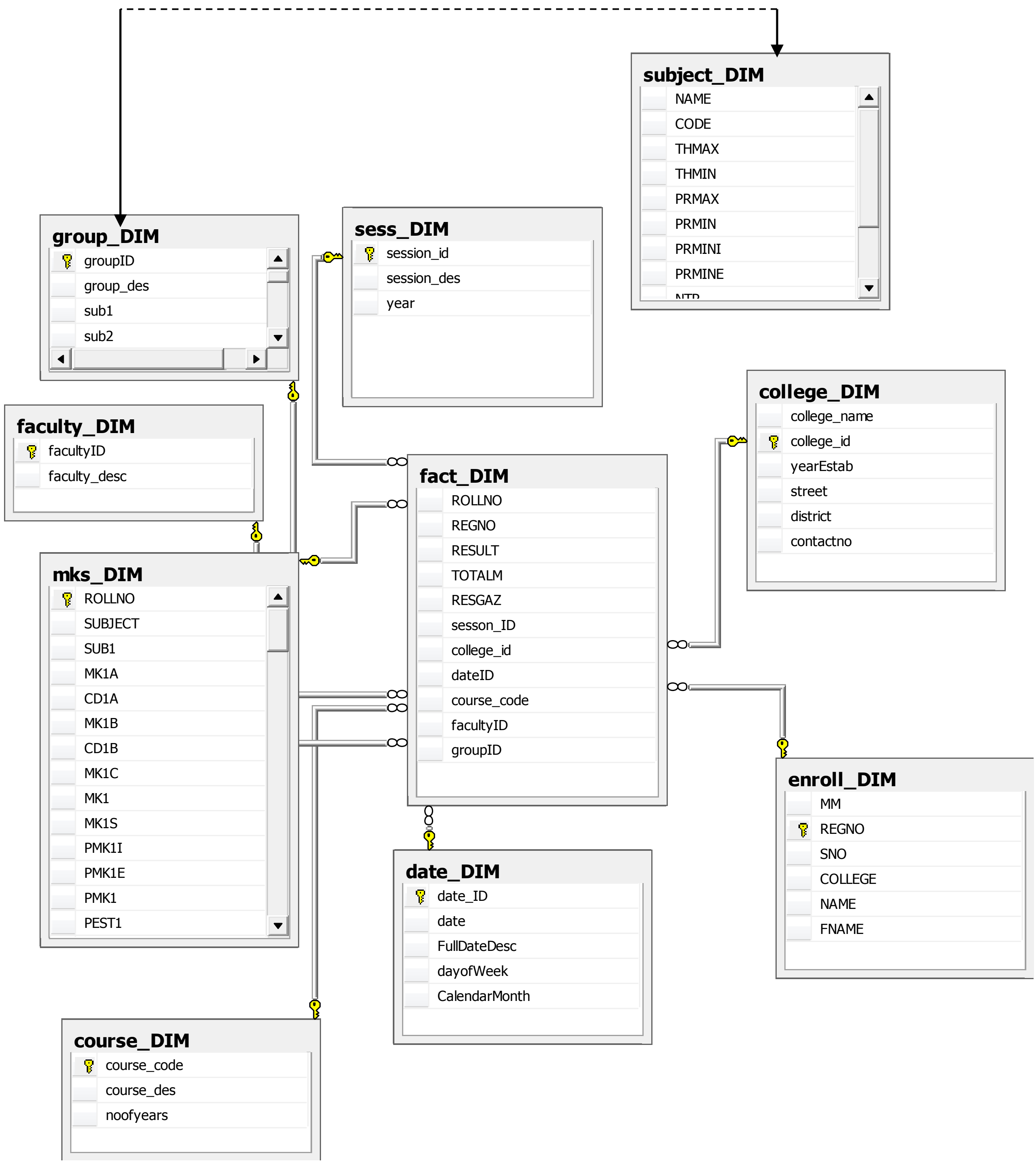
[15] Svetlozar Nestorov, Nenad Jukić, “Ad-Hoc Association-Rule Mining within the Data Warehouse”, Proceedings of the 36th Hawaii International Conference on System Sciences, 2002

[16] Syed Najam-ul-Hassan, Maqbool Uddin Shaikh, Uzair Iqbal Janjua,” Data Warehousing an Academic Discipline “Curriculum Development Approach, Methodologies and Issues”, 2006

[17] Wu Shuning, Cui Deguang, Cheng Peng ,”The Four-stage Standardized Modeling Method in Data Warehouse System Development” Proceedings of the IEEE International Conference on Mechatronics & Automation Niagara Falls, Canada • July 2005

[18] YUAN Hong, HE Houcun, “Online Analysis and Data Warehouse Modeling Technologies [J]”, Computer Application Research, 1999.12.

Appendix 1


subject_DIM
NAME
CODE
THMAX
THMIN
PRMAX
PRMIN
PRMINI
PRMINE
group_DIM
groupID
group_des
sub1
sub2
sess_DIM
session_id
session_des
year
college_DIM
college_name
college_id
yearEstab
street
district
contactno
faculty_DIM
facultyID
faculty_desc
fact_DIM
ROLLNO
REGNO
RESULT
TOTALM
RESGAZ
sesson_ID
college_id
dateID
course_code
facultyID
groupID
mks_DIM
ROLLNO
SUBJECT
SUB1
MK1A
CD1A
MK1B
CD1B
MK1C
MK1
MK1S
PMK1I
PMK1E
PMK1
PEST1
enroll_DIM
MM
REGNO
SNO
COLLEGE
NAME
FNAME
date_DIM
date_ID
date
FullDateDesc
dayofWeek
CalendarMonth
course_DIM
course_code
course_des
noofyears

Building An FPGA-Based Computing Platform

Mark White, Michael Gubody, Allison Nicol , Christopher G. Plachta,
Jeremy Strawn , Cori Thompson, Tai-Chi Lee
Department of Computer Science and Information Systems
Saginaw Valley State University
University Center, MI 48710

Abstract

This work involves a student research team engaged in activities to explore the use of FPGAs for computationally intensive applications. To learn the basics concepts and the fundamental logic design of a processor, a research team led by two senior students is formed. In order to fully understand how the FPGA works, a series of tutorials are targeted for analyzing and synthesizing the functionalities of various components in a processor. For example, the construction of clocks, registers, adders, multiplexors, and bus connections, etc. are implemented and tested. Both Altera and Xilinx tool kits with FPGA boards are used in the design. Once students have gained enough experience with construction and design of the basic components, they move on to build a computing platform, which consists of a special processor and custom instructions on FPGAs that will be called by the main process from the application source codes. Because of the low cost and lower energy consumption of FPGA, it makes an attractive, efficient processing element. In addition, the FPGA offers the flexibility of reconfiguring the hardware for various applications, thereby the advantages of using hardware/software co-design can be realized [7,11,12,14]. Further, this project provides students with research experiences needed for graduate study and design skills required in the workplace.

1. Introduction

General-purpose computers—such as those operating in homes, offices, and schools—have one or more Central Processing Units (CPUs). These CPUs contain a fixed number of hardware instructions that are determined at the time of the chip's creation. All computer programs, which run on these chips, are limited to the use of these fixed instructions. CPUs typically contain dozens of instructions, with the instruction set designed so that, in theory, any programming problem can be solved. While modern CPUs are capable of running any algorithm, there exist several drawbacks compared to custom hardware solutions. Using custom hardware to process the algorithm, one can expect significant increases in processing speed. This has significant impact for numerous time-critical applications in industries such as defense, weather forecasting, nuclear, power and chemical plants, etc. Their applications require the fastest and most efficient processing possible in order to make time-critical decisions.

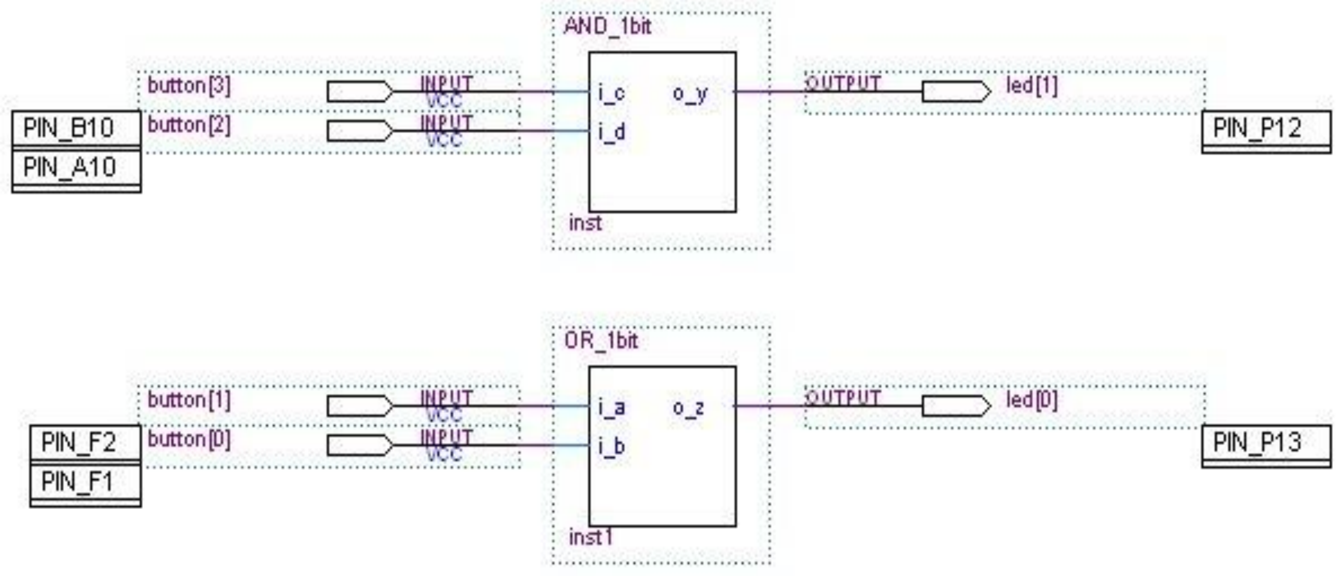
Due to the fixed instruction set on modern CPUs, some steps in certain algorithms require multiple clock cycles to complete. Often these algorithms cannot be redesigned, so the only option left to improve performance when using CPUs is to increase the speed of the CPU. This is where custom hardware instructions would be useful. A programmer could create the custom instructions in hardware on an FPGA via VHDL [5, 13] then use these custom instructions instead of the more generic instructions already provided. This provides the benefit of increased computational speed. Yet, one of the main problems prohibiting more widespread use of reconfigurable hardware is that the reconfiguration of the chips is a challenging task [3, 6, 15, 16]. There are some subtle issues which have not been very successfully resolved, at least not in a general platform. FPGAs, a special type of computing hardware that can be modified after they are manufactured, is one solution. This allows the chips to be repeatedly customized by a programmer to fit the specific needs of the target application. However, reconfiguring the chips (e.g., to add custom instructions) requires special knowledge from several disciplines, including computer science and computer engineering. There exists a wide range of application scenarios where high-performance computing plays a critical role. Among these: prediction of natural disasters, analysis of massive sensor and stream data, image compression [9,11], scientific computing and e-science, fraud detection, business intelligence, cloud intelligence, cryptosystem[4,8].

2. Constructions and Designs of Basics Components

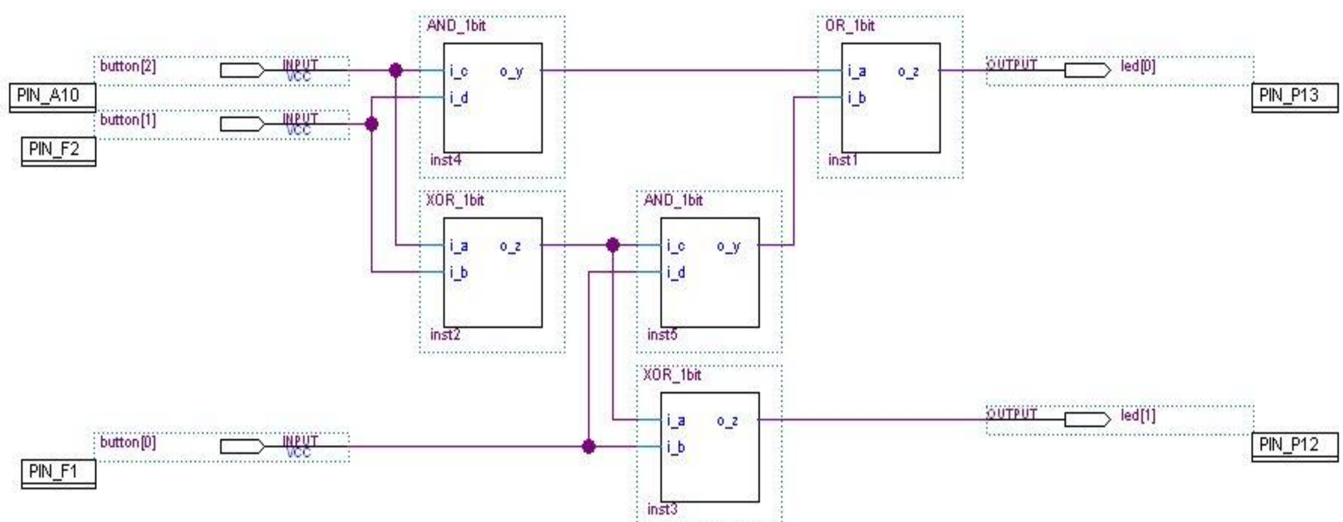
To orient the student researchers to the Altera and Xilinx tools [1, 17] and to circuit design itself, some tutorials had to be provided. Initially the default tutorials [2,18] for the development environments were used, with only slight modifications to account for updated software tools. Since the default tutorials were illustrating the capabilities of the tools rather than as an introduction to VHDL design, new tutorials had to be created, beginning with rudimentary circuits then increasing moderately in complexity. The following sequence of tutorials was developed:

1) Two simple but independent AND and OR gate circuits, connected directly to the button inputs and LED outputs, illustrating simple comparison logic similar to that used in

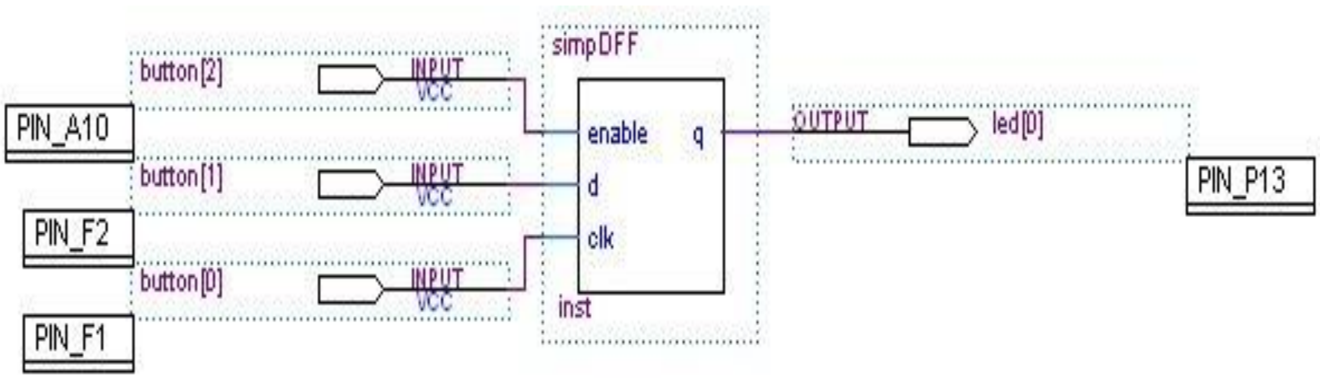
traditional computer language syntax. Additionally, this tutorial required the ability to make a component, which reinforced the default tutorials. For students who had little experience with hardware design, this tutorial allowed them to understand two of the most basic components. By pressing input buttons, associating those inputs to logic flags from their software curriculum, and seeing the results from the LEDs, students had to reinforce both the hardware design aspect and their own software background in a hardware environment.



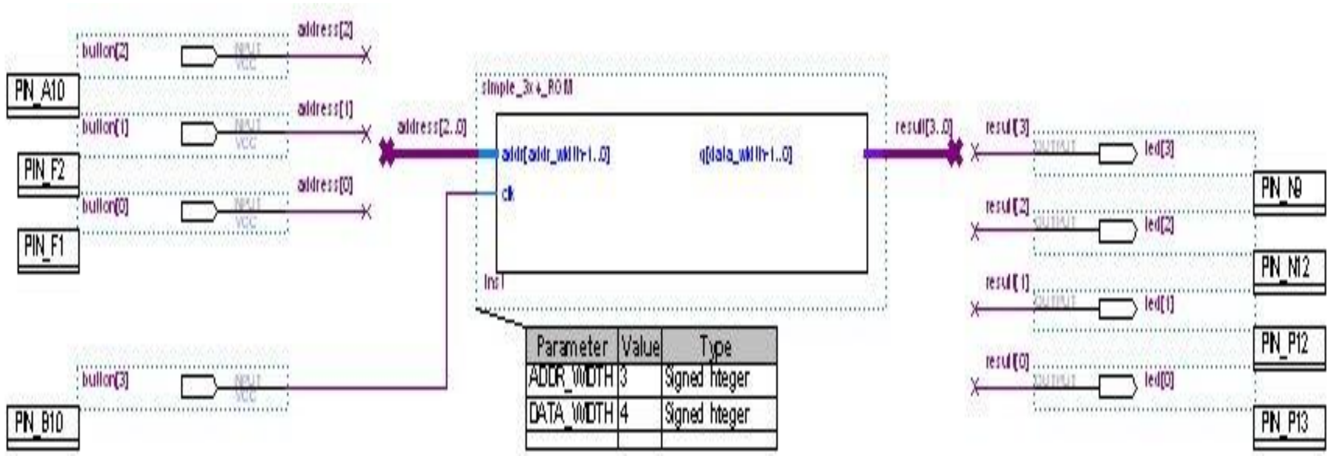
2) Two independent components from the first tutorial connected into a single combinational half-adder logic design of a 1-bit adder circuit with a carry-in signal. The student researchers were given the unnamed circuit diagram. The logic component icons and XOR gate were explained. The researchers were asked to construct this circuit and identify its purpose. The intent was to further develop logical circuit awareness, use of the tools, and reinforcement of the previous tutorial.



3) A simple one-bit D-flip-flop (DFF) circuit, implemented in VHDL, illustrated one type of simple data storage element. Not having been told beforehand, the researchers had to determine the behavior and purpose of the device. This tutorial was intended to impress the ability to express flag conditions in hardware.



4) A VHDL-implemented ROM device and associated memory initialization file reinforced and illustrated in hardware the idea of the look-up table from the student's software background. After initial construction and implementation, students had to modify the memory initialization file, re-implement, and then explain the changes to output that were occurring. Although the researchers already were quite familiar with volatile memory, the non-volatile memory aspect and the ability to make the component in VHDL were stressed.



5) Featured modifying the original default tutorial code to display onto a VGA screen a graphical representation of the content of a register. Students had to modify existing VHDL code, which had solely provided the raw signal to the VGA monitor, by both created the register and also giving a graphical representation of the data while positioned correctly onto the display. As new values were being given to the register, the result would be displayed upon the VGA monitor. The tutorial was designed with the intent of the researchers' ability to visually debug signals from within the FPGA.

More tutorials are being developed to further enhance the researchers' design abilities. After completion of these tutorials, the students will be able to construct a complete computing platform.

3. Proposed Computing Platform

Our approach is to make use of the NIOS II embedded processor, custom instructions [8,10], and adaptive compilation to prototype and build a system for algorithms with complex calculations. Our project will include the creation of a 64-bit Arithmetic Logic Unit (ALU) with its own custom instructions added to an Altera EP3C120 NIOS II embedded processor. These custom instructions are designed to be small and rearrangeable portions of a computationally complex algorithm. The addition of these custom instructions will allow sections of the algorithm to be expressed in C (a programming language) and other sections as custom instructions. These sections can then easily be reordered and refactored by recompiling the algorithm and uploading the overlay to the FPGA via TCP/IP (Internet Protocol Suite) allowing for the distribution of the algorithms over a network. Overall, we hope to build and investigate a platform where users can upload new operational code to FPGAs and easily interact with the hardware.(see Figure 1)

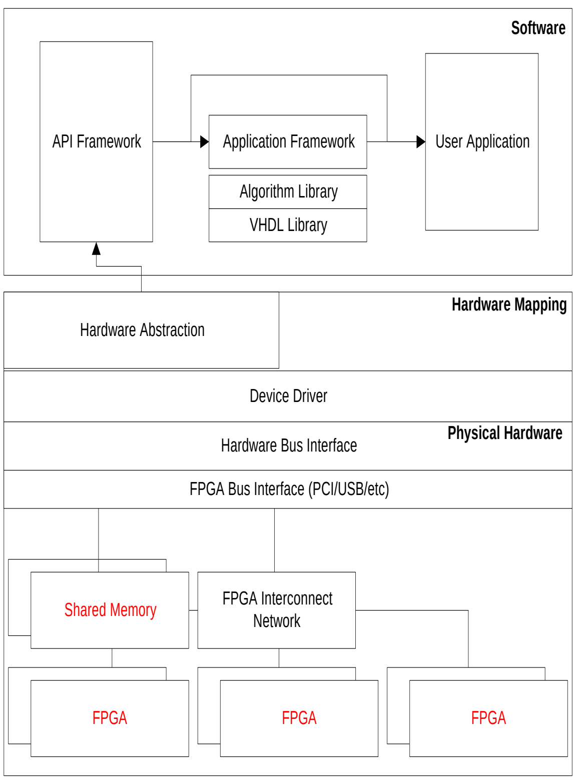


Figure 1

3.1 Description of Components:

PC Master Controller—Using a PC as a master controller will provide benefits over existing designs. It is capable of systematically assigning algorithms to logic units in the FPGAs based on the specific set of custom instructions included in the ALU. Our design will implement the most efficient way to delegate operations and take advantage of the parallelism obtained using FPGAs. The PC Master Controller will communicate through a TCP/IP interface with one or more FPGAs in a cluster. Each FPGA will execute the algorithm, using the custom instructions (see Figure 2).

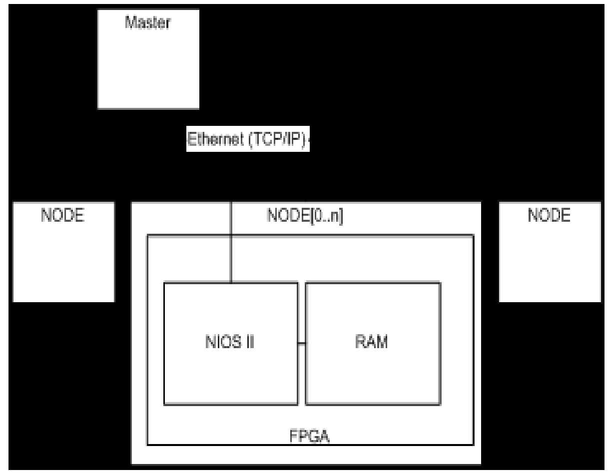


Figure 2

TCP/IP Interconnect—as in previous students' work from SVSU, the TCP/IP Interconnect will connect to a number of different FPGA cards forming a cluster of processing units. This inexpensive interconnect scales well when multiple FPGAs are used. For Logic Units, the FPGAs will contain our 64-bit custom ALU, along with several custom instructions. This ALU runs in parallel to the 32 bit general purpose ALU, shipped as part of the NIOS II CPU. The 32-bit ALU executes the operating system, our application, and the segments of the algorithm generated directly from C code (see Figures 3 and 4).

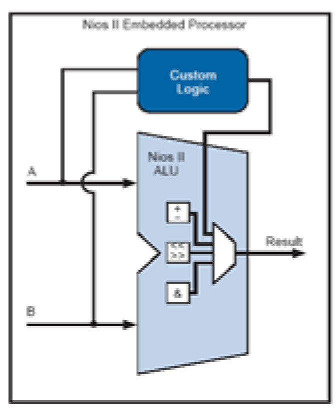


Figure 3

Custom Instructions and Overlays designed for the logic units isolate the custom instructions used within the algorithm to allow for parallel processing of these custom instructions with the rest of the algorithm, thereby eliminating the need to update the file system image stored on the NIOS II development board.

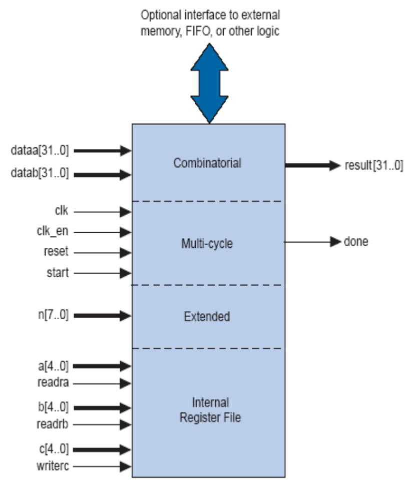


Figure 4

3.2 Assembly of Components:

Our custom ALU, implemented in the logic units of the FPGA, will have a list of all the custom instructions it supports. When the FPGA is started, it will read this list, then it will connect to the PC Master Controller and send it this list. Lastly, it will start requesting tasks from the Master Controller. The Master Controller determines which task to send to the FPGA based on the list of custom instructions the ALU contains; the task that would make the most use of the custom instructions is chosen. This task is then loaded to the FPGA through the TCP/IP interconnect and executed. The task continues to execute until it has completed or until the PC Master Controller interrupts the task and assigns the FPGA a new task.

4. Conclusion

FPGA co-design techniques remain a viable means to perform complex calculations quicker than general purpose processors. Our approach and design make use of the NIOS II embedded processor and custom instructions to help rapidly build and prototype a system for an application algorithm. We are expecting the customized system to out scale similar single purpose computing platform due to our use of custom-built instructions and embedded processors. The following is a list of anticipated benefits for our proposed FPGA-based design:

- FPGA Reconfigurable Computing Has The potential For Many Applications
- With Fast Hardware logic on FPGA's, it will reduce runtime.
- The Parallel Nature of Computer Chips Creates the Potential for Fast Running Algorithms
- An API Framework will Help Support Software Co-Design development
- A Platform to Support Reconfigurable Computing will Make it Easier to Develop and Deploy Software/Hardware Solutions

5. Acknowledgement

We are grateful for the support of the SVSU Resource Foundation for the work of this project. Also, we like to thank Professors Jurg Gutknecht, Alan Freed, and Mr. Paul Reed for sharing their ideas and assistance.

References

[1] The Altera Reconfigurable Computer with PCI interface (ARC-PCI), Altera Corporation, San Jose, CA, February 1998.

[2] Altera Corporation, July 2008, My First FPGA Design Tutorial. http://www.altera.com/literature/tt/tt_my_first_fpga.pdf

[3] Narges Bani Asadi et al., *Reconfigurable Computing for Learning Bayesian Networks*. Proc. 16th International ACM/SIGDA Symposium Field Programmable Gate Arrays, pp. 203-211, 2008.

[4] M.J Bastiaans. *FPGAs as Cryptanalytic Tools.* http://www.sps.ele.tue.nl/members/m.j.bastiaans/spc/rouvroy.pdf

[5] J. A Bhasker, *Verilog HDL Primer*, Star Galaxy Press, Allentown, PA, 1997.

[6] P.Glesner, and M. Zipf, Renovell, Eds., *Programmable Logic and Applications. Reconfigurable Computing Is Going Mainstream*, Proc. 12th International Conference, FPL 2002, Montpellier, France, September 2-4, 2002.

[7] Tai-Chi Lee, Mark White, *Using Software Emulation In FPGAs To Improve Co-Design Development Time*, Proc. Fourth International Conference Applied Mathematics and Computing, Plovdiv, Bulgaria, August 12-18, 2007, pp 359-360.

[8] Tai-Chi Lee, Mark White and Michael Gubody, *Blowfish Encryption/Decryption Using FPGA and Custom Instructions*, Proc. 2008 International Conference Parallel and Distributed Processing Techniques and Applications, Las Vegas, June 14-17, 2008, pp. 164-169.

[9] Tai-Chi Lee, Patrick Robinson *A FPGA-Based Designed for an Image Compressor***,** International Journal of Pure and Applied Math, Academic Publications, Volume 33 No.1 2006, pp. 63-67.

[10] Tai-Chi Lee, Richard Zeien, Adam Roach, and Patrick Robinson, *DES Decoding Using FPGA and Custom Instructions*, Proc. Third International Conference Information Technology: New Generation, Las Vegas, April 4-10, 2006, pp. 575-577.

[11] Tai-Chi Lee, Erik Henne, and Patrick Robinson, *Codesign for a Fractal Image Compression Decoder*, Proc. 2004 International Conference Computer Science, Honolulu, January 15-18, 2004, pp. 542-546.

[12] Tai-Chi Lee, and Erik Henne, *Hardware—Software Codesign for Matrix Multiplication*, Proc. 2003 International Multi-Conference Computer Science and Computer Engineering, Las Vegas, June 23-26, 2003, pp. 328-332.

[13] Tai-Chi Lee, Patrick Robinson, and Erik Henne, *Framework for Executing VHDL Code on FPGA*, Proc. 2004 International MultiConference Computer Science & Computer, Las Vegas, pp 1296-1299.

[14] Yufeng Lu, Erdal Oruklu, and Jafar Saniie, *FPGA-Based Hardware/Software Co-Design for Chirplet Signal Decomposition*, Proc. 18th ACM Great Lakes Symposium on VLSI, 2008, pp. 347-350.

[15] Roman Lysecky, Frank Vahid, *A Configurable Logic Architecture for Dynamic Hardware/Software Partitioning*, Proc. Conference on Design, Automation and Test in Europe—vol. 1, 2004, pp. 10480.

[16] Lesley Shannon, Paul Chow, *Using Reconfigurability to Achieve Real-Time Profiling for Hardware/Software Codesign*, Proc., 2004 ACM/SIGDA 12th International Symposium on Field Programmable Gate Arrays, 2004, pp. 190–199.

[17] Xilinx Inc., http://www.xilinx.com

[18] Xilinx, 2002, Inc. ISE 10.1 Quick Start Tutorial. http://www.xilinx.com/itp/xilinx10/books/docs/qst/qst.pdf

Promoting Interdisciplinary Initiatives using New Technologies: Solutions to a Human-Computer Interaction Problem

Rocío Abascal-Mena, Carlos R. Jaimez-González, Erick López-Ornelas, Wulfrano Luna and Sergio Zepeda
Universidad Autónoma Metropolitana – Unidad Cuajimalpa
Departamento de Tecnologías de la Información,
Av. Constituyentes 1054, 6° piso, Col. Lomas Altas, Del. Miguel Hidalgo,
C.P. 11950, México D.F., México

Abstract - *Human-Computer Interaction (HCI) involves a great variety of disciplines or fields. The term interdisciplinarity involves the combination of two or more fields or disciplines. Moreover, an interdisciplinary field crosses traditional boundaries because new needs are emerging. From this perspective, HCI is an interdisciplinary field, where its problems require from different disciplines in order to be solved. In this paper we present a design problem related to HCI, where we got three disciplines involved (Communication Sciences, Design, and Information Technology) to get a solution for it. To achieve this, we organized a contest, where interdisciplinary groups of students proposed solutions. We received 32 proposals of solutions, coming from teams, of the problem. To show the interest of proposals we have selected 9 of them in order to be discussed in this paper. The interdisciplinary elements that composed the solutions proposed are a clear indication that HCI is an interdisciplinary field.*

Keywords: Human-Computer Interaction, HCI, promoting interdisciplinarity iniciatives, case studies, team projects, web technologies.

1 Introduction

Human-Computer Interaction (HCI) involves a great variety of disciplines or fields. The term *interdisciplinarity* involves the combination of two or more fields or disciplines. Moreover, an interdisciplinary field crosses traditional boundaries because new needs are emerging. From this perspective, HCI is an interdisciplinary field, where its problems require from different disciplines in order to be solved.

In this paper we present a design problem related to HCI, where we got three disciplines involved (Communication Sciences, Design, and Information Technology) to get a solution for it. To achieve this, we organized a contest, where interdisciplinary groups of students proposed a solution. The problem has to do with the access to information on public transportation, where users need to locate a suitable route, tourist site, hospital, or restaurant. However, access to this information is sometimes difficult for certain sectors of the population in Mexico City. The contest was intended to make an proposal for solving the problem by designing an object, interface, system, service or communication strategy in order to help disabled people, tourists, vulnerable population or anyone else, to find and display information according to their specific needs, and taking into account their capabilities and context. For the contest, we received 32 proposals for solving the problem above, from which we selected 9 of them to be presented in this paper.

The rest of the paper is organized as follows. Section 2 describes the importance of interdisciplinary initiatives and HCI. In section 3 we focus on explaining the design problem, and the contest we organized to receive proposals. Section 4 discusses 9 out of the 32 proposals we received, where we highlight the main ideas and contributions of every proposal. Finally, we present the conclusions and future work in section 5.

2 Why is Interdisciplinarity Important to HCI?

The field of HCI involves knowledge of a great variety of disciplines, all of them focused on the human user for the design, evaluation and interactivity of computer systems development, and the several technical and sociological phenomena which comes from those activities [1]. As a consequence, there had been proposed a lot of approaches around HCI. These are: human centered, focused on the computational system, or well, the development driven by interaction, between human o computers, where the standards of usability are very important for making comfortable and relevant interactions [2].

As it was established before, HCI is a broad area of study which solutions need knowledge from different sources such as computer sciences, psychology, anthropology, design, ergonomy, economy, etc. There are a lot of techniques, methodologies and technologies used for developing interfaces, which are an important component in most computer systems, in order to generate adequate and productive interactions, for example: graphical design systems (drawing, photo and image edition and animations), text edition, spreadsheets, video games etc. The interface design must take into consideration the particular needs of users when they are doing certain task, the usability of the tools, etc. Does it need to render 3D computer graphics?, does the task imply the joint of static or dynamic graphics, sound and text?, does it need the use of voice recognition, or is it required to synthesize it?, would the application recognize the user's gestures (or some other biometric measure) for customizing its functionality?, would it be a cooperative work for accomplishing some task driven by a group of users? From those questions, a computer system was developed, where its construction process made evident the need for an interdisciplinary work around the HCI [1]: graphical interfaces based on icons, buttons, slice windows, several menus and controls, touch screens, mouses, joysticks, etc. It is clear that from HCI many other areas had appeared, such as virtual reality, enhanced reality, which development is a key field in computing applications. Those areas have generated a lot of knowledge and a diversity of techniques and methodologies; for example, the great WWW, which is growing due to the existence of appropriate interfaces for user interaction and communication (user-to-computer or user-to-user by means of a computer).

On the other hand, there are a lot of problems that must be studied from a HCI perspective, because of the attention that has been generated lately about people with some vulnerability, cognitive or physical disability. HCI helps those people to increase their way of interpersonal and remote communication. For example, the autonomous controlling of the environment where the user with disabilities live by means of wireless communication and command systems, such as automatic wheelchairs, automated doors, etc. The HCI area has become an invaluable instrument aimed to eliminate social exclusion too. The different efforts in making assistive systems take into consideration the adaptation of the existing systems or, in a more promising approach, the development and innovation of HCI techniques for the construction of interfaces, where the universal accessibility, understood as the elimination of barriers to access, due to the use of some particular technology, and the facility of being easily adapted are fundamental guidelines [3].

3 Focusing on the HCI Problem

As seen above, improving end-users systems requires interdisciplinary action. In our days, HCI teams need expertise in three main areas: Communication Sciences, Design and Information Technology. More than one person on the team may fit each dimension, and one individual may fill more than one role.

In our Web Technologies and Systems [4] research group, we specialize in working with interactive systems, which are adapted to the end-user and we use the Web as the main platform for interaction between users. Our research group is part of the *Information Technology Department*, which belongs to the *Faculty of Communication Sciences and Design* (http://hermes.cua.uam.mx). In this way, our disciplines related to information technologies, play an important role in the integration and definition of the synergies among different fields in order to work in interdisciplinary teams. In summary, one of the objectives of our research group is to try to operate as an integrator of the three disciplines of the faculty (Communication Sciences, Design and Information Technology). Given these disciplines are important but different in how they work, our strategy is to focus our efforts in the application of practices on the HCI field.

To achieve this goal, we defined a design problem related to HCI in order to get the three disciplines involved. We launched a student contest where interdisciplinary groups should propose a solution for that problem. The solution should be broad enough so that every field could feel free to interact and provide interesting elements to the solution from their own perspectives. The HCI problem we proposed was the following:

"Today, in Mexico City, citizens need to access certain information when they move through this big city. Usually, they use systems, devices and maps when they are looking for a suitable route, tourist site, hospital, or restaurant. However, access to this information is sometimes difficult or even impossible for certain vulnerable sectors of our population, such as indigenous people, migrants or people with disabilities. A clear example of this problem is observed in public transportation, where maps can be confusing, because they contain much information, which is not always relevant or useful."

In this way, the contest we designed was intended to make an interdisciplinary proposal for solving the problem described above by designing an object, interface, system, service or communication strategy in order to help disabled people, tourists, vulnerable population or anyone else, to find and display information according to their specific needs, and taking into account their capabilities and context. Once we defined the problem, we got our entire Faculty involved. In fact, the main objective was to motivate students to solve this problem by getting interdisciplinary and creative proposals.

With this in mind, we launched a student contest. The objective was to identify and support students who have interest in working with interactive systems and where the incidence of HCI, Web Technology, Design and Communication Sciences could represent an important element to the raised problem. An interesting feature of the contest was the fact that students should make proposals to solve the problem in an interdisciplinary way, so teams should be composed by a minimum of two students involving at least two different disciplines (departments).

Surprisingly, we received 32 proposed solutions. Every proposal met the initial goal of the competition with an interdisciplinary approach. The proposals had the characteristic of being feasible but dealt by multiple disciplines. Next section describes the most 9 outstanding proposals and exemplifies the interdisciplinary work to solve problems around HCI.

The maximum allowed number of pages is seven for Regular Research Papers (RRP) and Regular Research Reports (RRR); four for Short Research Papers (SRP); and two for Posters (PST).

4 HCI Proposals

This section presents 9 out of the 32 proposals we received for the HCI contest. Every subsection summarizes each proposal highlighting their main ideas and contributions. Web references are available for the original proposals, so they can be downloaded (unfortunately they are written only in Spanish).

4.1 Guide-ME

This proposal [5] was written by a team of 3 students from Information Technology and 1 student from Design. The "Guide-ME" proposal tries to help foreign people to transfer between different Metro stations in Mexico City. It uses 2 main elements: 1) a touch screen based on Microsoft's Surface Technology, which allows the user finding their destination through a friendly user interface, and 2) a bracelet, which will be the guide through the Metro. The bracelet, shown in Figure 1, will use a small display with lights and sounds to indicate whether the person is taking the right path.

In this proposal, we find the interdisciplinary work in the combination of technological elements such as the use of a touch screen that define the suitable paths in the Metro and the design of a bracelet. This bracelet will have the path information and it will guide the foreign person on his way. As part of the proposal, an ergonomic study was also conducted with the use of design elements to create a friendly user interface and an attractive bracelet.

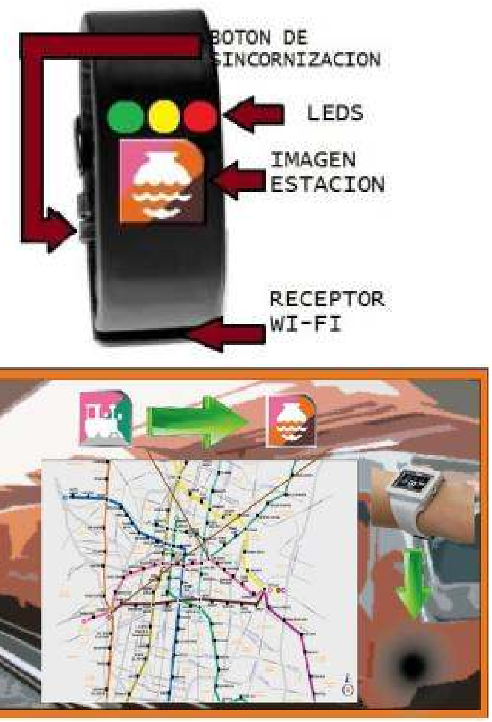


Figure 1. The bracelet prototype and the interface proposed.

4.2 Connect, intelligent interconnections

The proposal "Connect, intelligent interconnections" [6] tries to help users of Metro and Metrobus (similar to a tramway) in Mexico City. This is a system that uses an information kiosk, located in strategic locations, which has some alternative languages to allow different foreign people to interact with it. To help users, the system will not use regular maps, but some animations showing the path with some simplified instructions like "go upstairs", "turn right", "walk until you reach the intersection", etc. Also, an ideogram was implemented to identify the necessary connections or transfers between different stations. In Figure 2 we show the ideograms proposed to the connection and the interactive interface.

In this proposal, we find the interdisciplinary work focused on the design of ideograms, which are adapted to the population, and the design of relevant animations and interfaces for the application. This proposal was made by 2 students from Information Technology, and 2 students from Design.

Figure 2. Ideograms proposed and the interactive interface.

4.3 Distributed management of routes

"Distributed management of routes" [7] is a proposal made by 2 students from Information Technology, and 1 student from Communication Sciences. This proposal uses an expert system to help the general population with their transfers through Metro and buses.

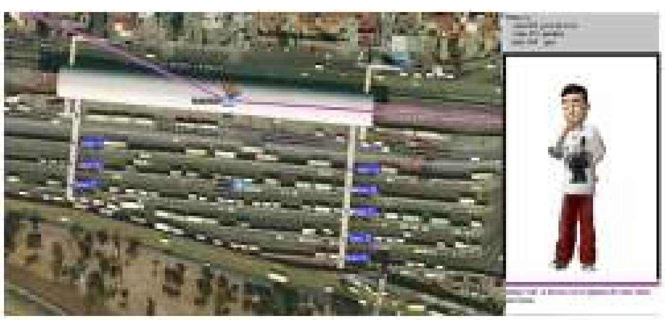


Figure 3. The proposed interface and the avatar.

The system has a knowledge database (maps, routes, directories) and uses the real-time events (traffic or accidents). It also uses a touch screen and a computer avatar, which interacts with the user in a simple way by using natural language and emotions. All this information and devices are used in order to guide and construct the best path to the user. Finally, the avatar will use the sign language for the interaction with deaf people, as illustrated in Figure 3.

In this is proposal we find some strong elements in terms of information technology, such as the use of expert systems and the use of a computer avatar. Expertise coming from the Communication Sciences discipline is applied to the way messages are given; clearly and precisely to the user by using emotions. This proposal is a good example that illustrates how two different disciplines converge to solve a problem.

4.4 The angel that helps you

The proposal "The angel that helps you" [8] is an interactive and audiovisual application that helps people with hearing problems to move through the Metro in Mexico City. The application uses videos showing a deaf person the path to follow by using the sign language, as shown in Figure 4. These videos guide users in the way they have to interact with the application and also, show the required path that the user needs to follow to reach the desired destination. The application uses text information and a specialized iconography that will be displayed to the user.

Figure 4. The proposed interface for deaf people.

In this application we find a proposal which has an important component in the Communication Sciences field, because it applies a study of how to transmit information to users with hearing problems. This application was proposed by 1 student from Information Technology, 1 student from Communication Sciences and 2 students from Design.

4.5 Sensorial exploration of the space to visualize and find information

A team of 1 student from Information Technology, 1 student from Communication Sciences and 1 student from Design, made the "Sensorial exploration of the space to display and find information" proposal [9], which is the design of an application that allows blind and visually impaired people to find a specific place in the downtown of Mexico City. This system will use an information module capable to produce sounds, smells and textures. This is the kind of information that users will find during their trip. This information will guide blind people to reach their destination. Also, the sensorial information must be part of the environment (the smell of a particular restaurant, or a coffee shop, the sounds of bells or an specific piece of music, the real texture form the streets and walls, etc.). In Figure 5 we show some elements in the environment and the position of the interesting places.

This proposal includes some information technology studies in order to implement the module to display the sensorial information and some design studies about the incorporation and exploration of sensorial information on the environment.

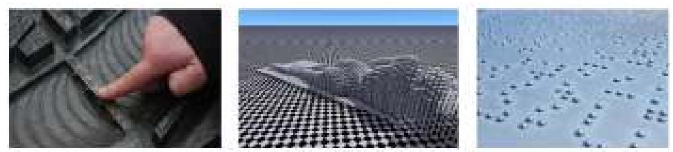
Figure 5. Sensorial elements and some interesting places to be modeled.

4.6 Touch tablet with dynamic relieves

This proposal [10] tries to provide a solution for disabled people, who are blind or visually impaired. These people normally use different ways in order to carry out their daily activities; such as walking sticks, which are used to identify close objects; trained dogs to help them move from one place to another; other people, who guide them; Braille tablets or signs, which are located on bus or metro stations; etc. Among these solutions, Braille tablets seem to be an ideal mechanism to get information in a very effective way, because they represent messages through the variation in the frequency of the dots in the tablet, which can be recognized by the target population. Unfortunately, the messages on a specific tablet are always fixed, and tablets cannot be found at any place.

A touch tablet with dynamic relieves is proposed, which dynamically provides important information to its user, such as the representation of streets, buildings, tourist places, etc. The tablet would detect, calculate and model the location of obstacles, so that they can be avoided by its user. It would also allow communicating with other people by translating voice into Braille code. The implementation of this tablet requires several components: low-frequency transceivers in order to detect obstacles; adapted microphones for voice recognition; special 3D and modeling software to translate voice into dynamic relieved text or Braille code; wireless antennas to capture information from different places of interest (museums, tourist agencies, government institutions, etc.); special material for the construction of the actual tablet, which will include resistant, durable and flexible spikes.

Figure 6 shows three mechanisms used by blind or visually impaired people: a) is a relieved map to recognize streets, bridges, buildings, and other elements of a city; b) is a set of spikes that builds models from information; and c) is a tablet with a message in Braille code. The proposed touch tablet with dynamic relieves will include some of these mechanisms in order to accomplish its purpose. This proposal was made by 2 students from Information Technology, and 1 student from Design. It is clear that the proposed tablet contains design elements to model it, as well as technological elements to implement the interface between the physical tablet and its user.



a) b) c)

Figure 6. Some devices (a, b and c) used by blind or visually impaired people.

4.7 Is here

An intelligent t-shirt [11] is proposed by 2 students from Information Technology, and 1 student from Design. The t-shirt is to be used by blind or visually impaired people, and will help them in several ways, such as: locating a specific place in Mexico City through a Global Positioning System (GPS); obtaining information about a place of interest; giving instructions about the routes to be followed, transportation and alternative routes to reach a destination; avoiding obstacles through a camera; providing bus timetables; etc. The intelligent t-shirt is composed of the following components, as can be seen in Figure 7: headphones, small keyboard on one of the sleeves, location vibrator device, microphone, small camera, ipod/iphone or mobile device that allows the interaction with the GPS, and a flexible solar panel.

The user of the intelligent t-shirt will make use of the keyboard to manipulate a built-in MP3 player, the incoming calls, and the destination places. Microphone and headphones are used to manipulate the mobile device through voice instructions; while the location vibrator device is used to indicate the direction where the user should go. The built-in camera will help the user to avoid obstacles; while the flexible solar panel allows charging the battery of the mobile device.

A simple example of the interaction between the intelligent t-shirt and its user is as follows: the user provides the voice command “is here”, so that the t-shirt will use the GPS to register the exact location of the user; the system will ask about the user destination, which can be provided in two different ways, through the keyboard by typing it, or through the microphone by talking to it. In case the destination is found, the t-shirt will communicate the user the most convenient way to reach the destination either on foot or using public transportation. The system will give its user the opportunity to choose between the two options.

This proposal only takes into consideration blind and visually impaired people, but with its infrastructure, it would also be possible to incorporate functionality for the identification of languages, which is useful for tourists and other people visiting a city. Interdisciplinary components are present in this proposal, where Design and Information Technology converge to implement this intelligent device.

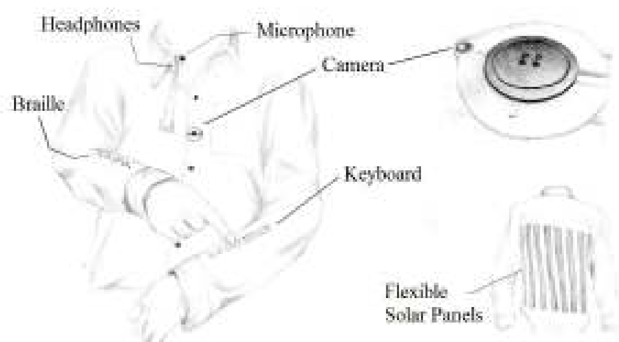


Figure 7. Components of the intelligent t-shirt.

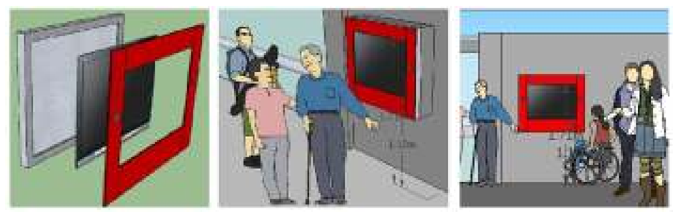
4.8 Metrobus interactive map

One of the main problems for the Metrobus users is the lack of visibility in the station signals and names, and the limited time to realize what station the users are arriving at. This represents a problem for all users of this public transport, but the problem increases for disabled people who are deaf, blind, or visually impaired. This proposal [12], made by 2 students from the Design department and 1 from the Information Technology department, consists in the implementation of interactive maps for every Metrobus station, which will be used not only by disabled people, but by all users.

The interactive map is presented on a 42-inches touch screen, which displays all the stations of the specific Metrobus line where the user is located; it also contains visual and sound indicators to show how to use the graphical user interface. Because the visual indicators are intended for visually impaired and elderly people, they need to be highly legible. Although the map will be configured to be used in Spanish, there will be some flag icons to change to other languages.

The basic functionality of the interactive map would be as follows: the user selects the destination Metrobus station; the map indicates with a line the path to be followed in order to reach the desired station, highlighting all the stations in the path. While the path is shown on the map, the system will also provide important information about places of interest near the destination station and its surroundings. The interactive map will also be located on the buses, so that users are aware of the station where they are traveling to. This proposal intends to improve the signalization of this important public transport, mainly for disabled and elderly people, but it is not restricted to other Metrobus users.

Figure 8 shows three images: a) illustrates the 42-inches touch screen with two protectors, one for the screen, and one for the frame; b) gives the suggested position for the screen on a station; while c) shows that the suggested position is also suitable for wheelchair users.



a) b) c)

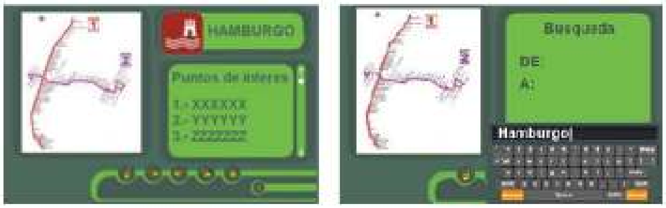
Figure 8. The Metrobus interactive map displayed on a 42-inches screen.

4.9 Information and services interactive map

This proposal [13] is presented by 2 students from Information Technology and 1 from Communication Sciences. This proposal aims to help Metrobus users finding a destination place through a touch screen, where they enter their current location and their destination; the touch screen system will advice them about the route they have to follow; additionally, users have the possibility to print out a map with the instructions to reach the desired place.

The system proposed not only covers the Metrobus stations and lines, but also streets and avenues near to the Metrobus stations, through the use of Google maps. Additionally, it is planned to incorporate Metro stations, so that the system can provide a better route to reach the user destination, improving time, cost and distance.

The system will have mainly four different interfaces for the user, as can be observed in Figures 9 and 10. In Figure 9, a) shows the main interface, where it is shown a map with the Metrobus lines, the station the where user is at, and some places of interest near the station; while b) shows a map, where the user will enter an initial and destination locations, so that the system can provide the necessary instructions.



a) b)

Figure 9. Touch screen with a) some places of interest, and b) the search system.

In Figure 10, a) shows the route interface, where the system provides different options to reach the chosen destination (the fastest, the shortest, and the cheapest option); while b) illustrates some places of interest and events near the destination.

a) b)

Figure 10. Touch screen with a) the route interface, and b) some places of interest.

5 Conclusions and future work

In this paper we have presented a design problem related to HCI, where we got three disciplines involved (Communication Sciences, Design, and Information Technology) to get a solution for it. To achieve this, we launched a contest, where interdisciplinary groups of students proposed a solution. We received 32 proposals for solving the problem, and we selected 9 of them, which were discussed throughout the paper, where we highlighted their main ideas and contributions. These proposals are freely available through our research group website.

The interdisciplinary elements that composed the solutions proposed are a clear indication that HCI is an interdisciplinary field. Some of these solutions are worth taking them to an implementation level, although some are quite ambitious in their goals. We are in the process of analyzing the best proposals, in order to determine whether it is feasible to implement them with the current technologies available.

In our curriculum, students are not always confronted with different areas. However, we believe that one of the main lessons that the university can give to the students is the ability to work with people from other fields. This leads them to adapt to different environments in order to explore joint solutions. In our experience, the teams made a great effort to integrate with other disciplines. This contest was an example of leadership that put in the table the need of generating more interdisciplinary courses or projects in order to support a more global education.

6 References

[1] HCIResources, 2011. Human-Computer Interaction resources. http://hcibib.org/.

[2] Brad, 1998. Brad A. Myers. "A Brief History of Human Computer Interaction Technology." ACM interactions. Vol. 5, no. 2, March, 1998. pp. 44-54. http://www.cs.cmu.edu/~amulet/papers/uihistory.tr.html.

[3] Abascal, 2005. Abascal J., Nicolle C. Moving towards inclusive design guidelines for socially and ethically aware HCI. Interacting with Computers 17 (2005) 484–505.

[4] TSWeb, 2011. Web Technologies and Systems research group (TSWeb). http://tsweb.cua.uam.mx.

[5] GuideMe, 2011. Guide Me. http://tsweb.cua.uam.mx/proposals/41_GuideMe.pdf.

[6] Connect, 2011. Connect, intelligent interconnections. http://tsweb.cua.uam.mx/proposals/42_Connect.pdf.

[7] Distributed, 2011. Distributed management of routes. http://tsweb.cua.uam.mx/proposals/43_Distributed.pdf.

[8] Angel, 2011. The angel that helps you. http://tsweb.cua.uam.mx/proposals/44_Angel.pdf.

[9] Sensorial, 2011. Sensorial exploration of the space to visualize and find information. http://tsweb.cua.uam.mx/proposals/45_Sensorial.pdf.

[10] Tablet, 2011. Touch tablet with dynamic relieves. http://tsweb.cua.uam.mx/proposals/46_TouchTablet.pdf.

[11] IsHere, 2011. Is here. http://tsweb.cua.uam.mx/proposals/47_IsHere.pdf.

[12] Map, 2011. Metrobus interactive map. http://tsweb.cua.uam.mx/proposals/ 48_Map.pdf.

[13] ISIS, 2011. Information and services interactive system. http://tsweb.cua.uam.mx/proposals/49_Services.pdf.

Utilizing Multi-Touch Technologies in Medical Rehabilitation: A Therapeutic Capstone Project

L. Donnell Payne
Computer Science Department, Texas Christian University, Fort Worth, Texas, USA

Abstract - *It has been reported that capstone course projects can be made more engaging by working with external clients, incorporating new technologies, and creating a socially relevant product. Researchers have reported the potential for valuable assistive therapy by incorporating multi-touch technology in neuromuscular and cognitive rehabilitation.*

In this paper the author discusses a capstone project that includes the development of a suite of therapeutic activities, with monitoring and reporting capabilities, which were designed and developed for medical rehabilitation. The activities were suggested by practicing therapists and developed on the Microsoft Surface multi-touch device. At project completion, the system will be delivered to a local rehabilitation center and we are hopeful that it can be used to provide valuable assistive therapy.

Keywords: capstone course, senior design project, multi-touch technologies, medical rehabilitation

1 Introduction

Capstone courses in computer science traditionally include such requirements as: 1) an extensive software design, implementation, and testing effort, often over an entire academic year; 2) development in a team environment, thus requiring students to use project planning and management skills; 3) numerous opportunities for the demonstration of oral and written communication – both in documentation and public project presentations; and 4) reviews and critical evaluation – both of the product and individuals.

Identifying the "ideal" capstone project for a senior class that meets these requirements and is both instructional and engaging is a challenge. Instructors of the capstone courses at other universities report several characteristics of senior projects that can be included to better engage the students. This includes such things as working with external clients [4,5,8], working on socially relevant projects [2,7], and working on projects that allow students to explore emerging technologies [7,8]. In our recent capstone course projects we have tried to include all of these characteristics.

This paper describes what we feel to be an exemplary capstone project using a multi-touch, multi-user device – the Microsoft Surface. Students were given the opportunity to work on a project that required learning a new technology, developing a very relevant medical rehabilitation application, and rolling out the final product to a rehabilitation facility upon completion.

The remainder of this paper is organized as follows. Section 2 provides background for the project, including how the project originated, an overview of the Microsoft Surface multi-touch device, explores the use of similar devices in medical rehabilitation, and overviews the initial system developed - Healing Touch. Section 3 provides a detailed description of the TheraTouch system. Section 4 summarizes project results and Section 5 offers some concluding remarks.

2 Background

2.1 In the beginning

Two years ago, one of the department's Industry Advisory Board members approached us expressing a desire to work on a project with our senior class. After some brainstorming, and a generous donation of two multi-touch Microsoft Surface units by RadioShack, it was decided to incorporate these devices into a medical rehabilitation application.

The capstone project for the 2010-2011 academic year started the initial work with the Surface units. Those students explored the use of both the iPad and Microsoft Surface by developing several games that could be used as rehabilitation exercises – basically offering a proof of concept. A brief discussion of that project is included later in this section. That project laid the groundwork for the follow-on capstone course of 2011-2012 which is described in Section 3.

2.2 Microsoft Surface

The Microsoft Surface is a commercial computing table-top platform that responds to touch and tagged objects [6]. It enables people to use touch and real world objects to share digital content. The Surface platform consists of software and hardware that combine vision based multi-touch PC hardware, Windows software, a 360-degree multiuser design to create a natural user interface.

The Surface 1.0, shown in Figure 1, consists of a rear projector display, integrated PC, and five IR cameras that can see fingers and objects placed on the tabletop. People interact with the tabletop by touch or placing objects on the display surface. The Surface is designed such that multiple people can approach and simultaneously use the unit from all sides. The device can recognize 52 simultaneous multi-touch points.

Figure 1. Microsoft Surface with TheraTouch Installed

The first Microsoft Surface 1.0 shipped in 2008. Since this project originally began, Microsoft and Samsung have released Surface 2.0. The applications described in this paper were developed on the original Surface 1.0.

2.3 Multi-touch devices in rehabilitation

The Surface was designed primarily for use by commercial customers in public settings. However, researchers have begun to realize the potential for using multi-touch devices in rehabilitation programs. Researchers from several universities are developing therapeutic applications for multi-touch devices [1,3]. The applications include developing activities for the rehabilitation of motor skills, upper extremity rehabilitation for children with cerebral palsy, and cognitive rehabilitation.

The traditional rehabilitation exercises can be monotonous, repetitive, and boring. The use of more interesting rehabilitation activities, even games, on multi-touch devices is more interactive and engaging. In addition, these multi-touch applications allow for multi-user interaction, can provide precise measurement of patient performance, and allow therapists to customize activities to specific patient abilities and needs. All of these benefits suggest that this type of technology has untapped potential in this field.

2.4 Healing Touch

As previously mentioned, the use of the Microsoft Surface units in our capstone course began with the 2010-2011 Senior Project – Healing Touch. In that project students developed a suite of games that could be used in medical rehabilitation. A neurologist and IT professional from Texas Health Resources, a local hospital system, helped identify appropriate therapeutic games. Screenshots of several of the games, including Air Hockey (game for social interaction and exercising motor skills), Meteor Defense (measuring motor skills), "Froggy Says" (memory recall), and Touchdown (deductive reasoning) are illustrated in Figure 2.

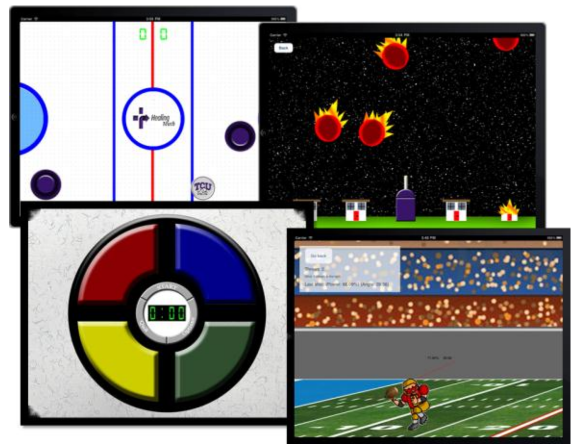


Figure 2. Screenshots of Games in Healing Touch

The games developed in the Healing Touch project were designed for both motor skill and cognitive rehabilitation. The initial Healing Touch project saved all game results and provided for Excel spreadsheets containing game results to be produced. The user could obtain desired reports or charts but it was dependent on his ability to use the plotting and charting capabilities of Excel. Figure 3 represents a simple plot of the length of a sequence that could be recalled in the "Froggy Says" game after varying delay times between showing the solution and asking the user to respond. The plot compares results obtained from users of two different mental capacities.

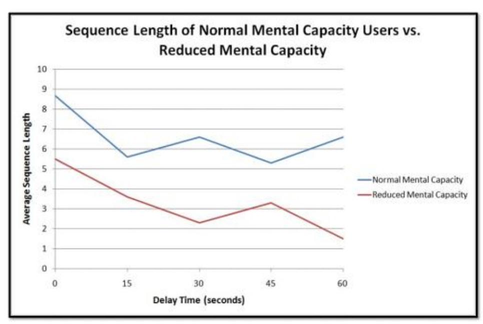


Figure 3. Sample Plot of Sequence Length Recalled

The project received very positive feedback from the client. At the end of the project, the client expressed a desire for enhanced reporting capabilities. The author also felt that a better understanding of actual clinical activities would be helpful in order to provide a system that could be utilized more effectively for assistive therapy. That outcome lead to the decision to develop a system that would provide activities similar to those used in therapy sessions, yet at the same time offer the benefits that multi-touch technology could provide. The following year's capstone project, TheraTouch, is that project.

3 TheraTouch System

3.1 System architecture

The TheraTouch system, illustrated in Figure 4, consists of three primary components: 1) the TheraLink backend; 2) the TheraLink frontend; and 3) the Microsoft Surface units. The TheraLink backend includes a web application and the system's database that stores patient information and activity results.

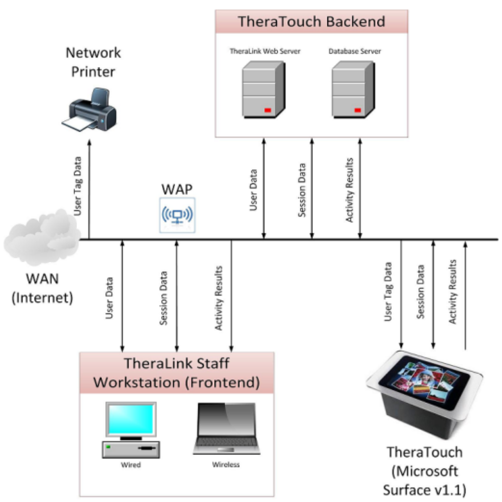


Figure 4. The TheraTouch System

The TheraLink frontend consists of staff laptops or workstations that interact with the web application in order to set up patient accounts, therapy sessions, and obtain patient reports. The staff members have the ability to customize therapy sessions, both with activities to perform and options to set for each activity. This important feature provides flexibility in the rehabilitation therapy by allowing therapists to customize activities according to a patient's abilities and increase difficulty in the program as the patient shows improvement.

The Microsoft Surface units have the entire suite of TheraTouch activities installed. They run in a "free run mode" (collecting no data) or in therapy session mode (collecting activity data). Login with a preprinted patient Surface tag is required for running the Surface in session mode in order to collect data.

3.2 The web application – TheraLink

The web application gives an administrator the ability to create staff accounts and perform other administrative tasks. Staff accounts are required to set up patient accounts, set up patient therapy sessions, and obtain individual and cumulative results and reports. Figure 5 illustrates a session with six activities. The SeekShape activity is expanded, allowing activity options to be configured.

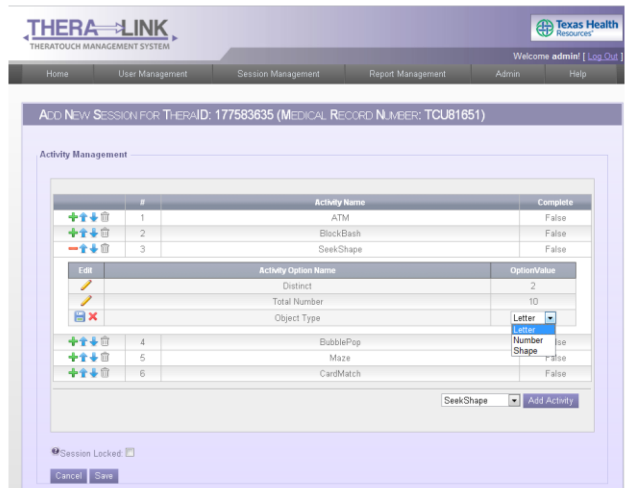


Figure 5. Configuring Activity Options

3.3 TheraTouch activities

The activities included in the TheraTouch system were designed with input from speech, occupational, and physical therapists. A list of the activities implemented, along with type of therapy primarily covered by each activity, is shown in Figure 6. All activities include some degree of physical therapy simply because of the need to interact with the Surface.

	Therapy Type		
Activity	**Speech**	**Occupational**	**Physical**
Alternate Trail Making	X		
Block Bash	X		X
Bubble Pop			X
Card Match	X		
Find The Way		X	
Maze	X		X
Metronome			X
Odd One Out	X		
Path Track	X		
Seek the Shape	X		X
Sequencing	X	X	
Shape Match	X		
Vending Machine		X	
What Time Is It?	X	X	
Wipe the Table	X	X	X

Figure 6. TheraTouch Activities

The activities shown in the table above were all implemented on the Microsoft Surface. In the prototyping of each activity, the appropriate activity options were identified as well as the activity results that would be collected and reported. Four sample activities are described below.

3.3.1 Maze

This activity was an implementation of the classic maze. From an initial starting point the user is required to draw a path to the exit without passing through walls. The activity provides for three levels of difficulty. This activity records wall hits and total time to negotiate through the maze. If the patient loses contact with the Surface display, he must pick up the path at the point where contact was lost. Figure 7 illustrates the *Maze* activity.

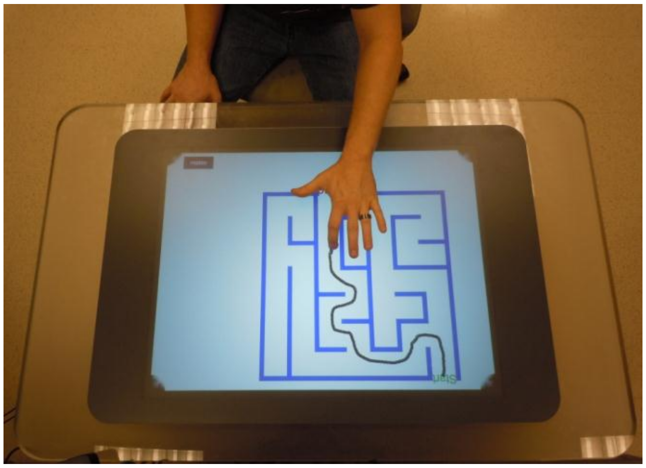


Figure 7. The Maze

3.3.2 Find the Way

The *Find the Way* activity was suggested by occupational therapists in order to have the patient find his way through a house to perform some routine task. For example, the patient might be asked to go get a snack from the refrigerator or go get a coat from the closet. This activity is shown in Figure 8. The patient moves a figure through the house with the controls at the bottom of the screen.

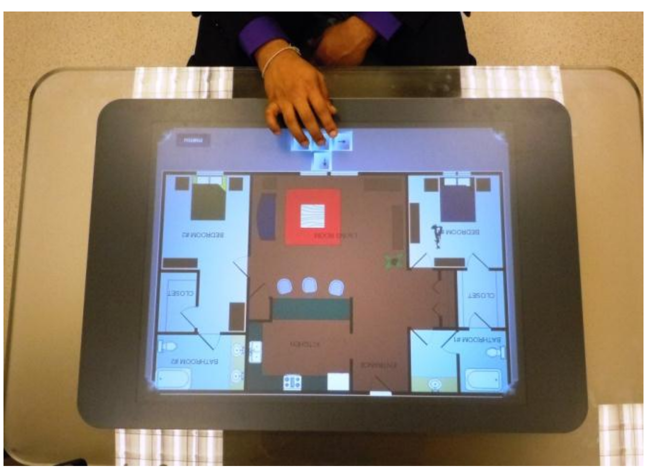


Figure 8. Find The Way

3.3.3 Odd One Out

Odd One Out simply provides a screen of shapes from which the patient was to identify the shape that did not belong. This activity's options include such things as number and type of shapes displayed and records time to complete and incorrect selections. A screenshot of this activity is shown in Figure 9.

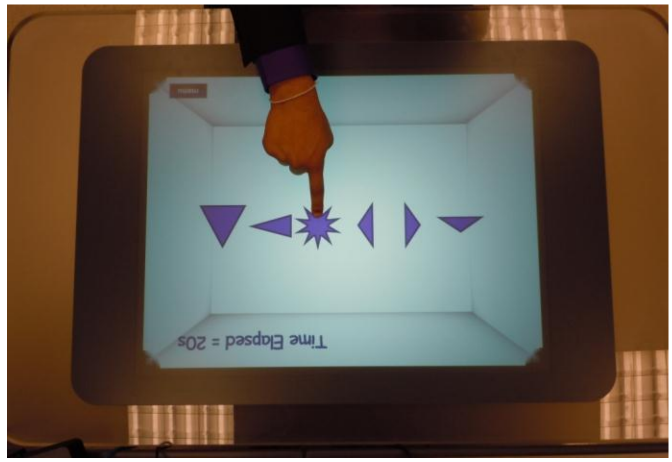


Figure 9. Odd One Out

3.3.4 What Time Is It?

Activities defined for occupational therapy deal with activities of everyday life. This activity was one of those. In *What Time Is It?* the patient is required to set the correct time. The therapist configuring activity options selects to have the analog time given and the patient sets the digital time to match, or vice versa. Accuracy and time to complete are reported for this activity.

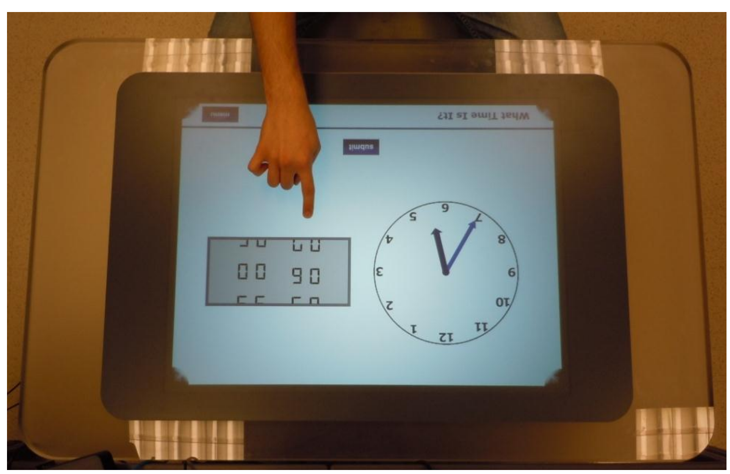


Figure 10. What Time Is It?

3.4 Reporting

As previously noted, activity results are stored to a database for subsequent reporting. Reporting can be accomplished in one of two ways. Several custom reports for each activity are provided in the web application, utilizing standard ASP.NET libraries. Such things as accuracy vs. time to complete and user progress reports are readily obtained. Comparative studies are also provided by observing results of a particular patient vs. the patient population. Additionally, any activity results can be selected and exported to an Excel spreadsheet for additional reporting and charting. Figure 11 shows a particular patient's improvement in accuracy (larger blue dots) and time elapsed (smaller red dots) as the activity was performed four times.

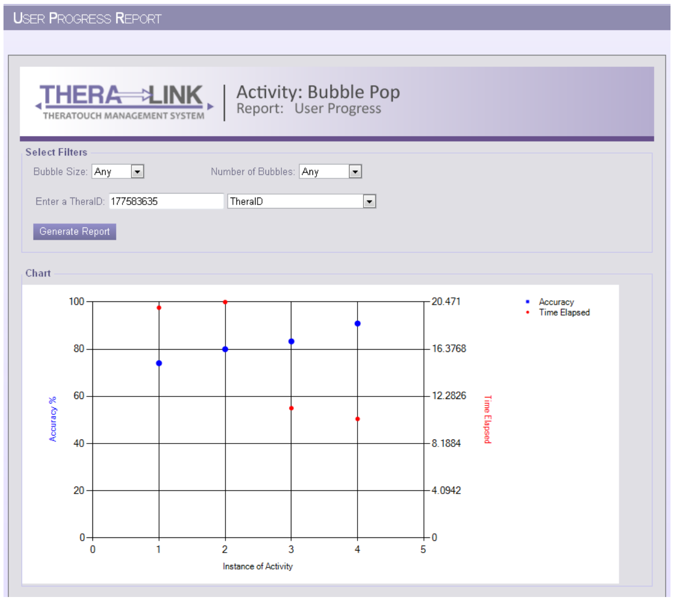


Figure 11. User Progress Report

4 Results and Observations

The original Healing Touch and the follow-on TheraTouch project have resulted in very interesting and relevant capstone projects. Though the pressure of actually producing a project to be delivered to external clients was often intense, the students responded well. An initial roll-out of the system to the rehabilitation clinic, complete with the donated Microsoft Surface unit, was accomplished three weeks prior to the end of the semester. We felt that this was particularly important in order to get initial feedback of the system for the students prior to their graduation. A subsequent roll-out of all completed activities and reporting will occur at the completion of the semester.

Since the activities were developed with input from practicing therapists at the facility, we are eager to see how the TheraTouch system will be accepted as an assistive therapy instrument. Preliminary feedback received at the final project presentation was extremely positive. That will have to be the subject of a follow-up report.

5 Conclusions

Identifying the "ideal" capstone project for a senior class is a challenge. It is often hard to find projects that both meet the requirements of the course and are engaging and interesting to the students. The initial enthusiasm for a senior project is often tempered by the grind of working on a project over an extended time (one academic year in this case) and of this magnitude. Up to this point in their academic careers, students have not had to work in groups of this size (eleven students in the case of TheraTouch), produce the required documentation, learn as much new material on their own, or be subject to as much review and evaluation.

In this author's opinion, the group size noted is probably larger than ideal (because of accountability and communication issues) unless the project being developed has some clearly defined components that allow for smaller working groups. That was the case here. A group of four students worked with the backend web application and database while others developed the individual Surface applications. All students were involved at various times with client interviews and developing documentation.

We believe that the interest in working with a technology that was new to the students, working to meet the requests of external clients, and realizing that the product delivered was actually intended for use in a rehabilitation facility provided additional motivation and incentive and helped keep the students interested and engaged.

6 References

[1] Annett, M., Anderson, F., Goertzen, D., Halton, J., Ranson, Q., Bischof, W.F., Boulanger, P. 2009. Using a Multi-touch Tabletop for Upper-Extremity Motor Rehabilitation. In *Proc. 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7* (Melbourne, Australia, Nov 23-27, 2009), 261-264.

[2] Buckley, M., Nordlinger, J., Subramanian, D. 2008. Socially Relevant Computing. In *Proceedings of the SIGCSE '08* (Portland, OR, Mar 12-15, 2008), 347-351.

[3] Dunne, A., Do-Lenh, S., Ó'Laighin, G., Shen, C., Bonato, P., 2010. "Upper Extremity Rehabilitation of Children with Cerebral Palsy using Accelerometer Feedback on a Multitouch Display". In *Proceeding of the IEEE Engineering in Medicine and Biology Society(EMBC)* (2010), 1751-1754.

[4] Gorka, S., Miller, J., and Howe, B. 2007. Developing Realistic Capstone Projects in Conjunction with Industry. In *Proceedings of the SIGITE '07* (Destin, FL, Oct 18-20, 2007), 27-32.

[5] Hadfield, S. and Jensen, N. 2007. Crafting a Software Engineering Capstone Project Course. *Journal of Computing Sciences in Colleges*, 23(1), 190-197.

[6] Microsoft Surface. http://www.microsoft.com/surface/en/us/default.asp.

[7] Pauca, V., Guy, R. 2012. Mobile Apps for the Greater Good: A Socially Relevant Approach to Software Engineering. In *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*. (Raleigh, NC, Feb 29-Mar 3, 2012), 535-540.

[8] Payne, L.D., Zlatkov, D., Jernigan, J., Scarbrough, J. 2009. A Location-aware Mobile System for On-site Mapping and Geographic Data Management. In *Proceedings of the 10th ACM Conference on SIG-Information Technology Education* (Fairfax, VA, Oct 22-24, 2009), 166-172.

National Center of Academic Excellence in Cyber Security and Information Assurance. An Implementation Case at Cameron University

Pedro A. Diaz-Gomez[1], **Mike Estep**[1], **Johnny Carroll**[1], **and Reza Kamali**[2]
[1]Computing & Technology Department, Cameron University, Lawton, OK, USA
[2]College of Natural Sciences, California State University at Stanislaus, Turlock, CA, USA

Abstract— *The goal of this paper is to present the experience of the implementation of a Center of Academic Excellence in Cyber Security and Information Assurance, in order to share the experience with the academic community, pretending to help future implementations of Centers of Academic Excellence following the guidelines of the National Security Agency.*

Keywords: accreditation, certification, cyber security, information assurance, information security.

1. Introduction

The Department of Computing & Technology at Cameron University (CU), proposed the creation of a Center of Academic Excellence (CAE) in Cyber Security and Information Assurance in September 2010 with a three years goal of implementation, using the synergy of the Cyber Security Education Consortium (CSEC), which is a National Science Foundation of advanced technological education, dedicated to building an information security workforce who will play a critical role in implementing the National strategy to secure cyberspace.

The requirements to become a Center of Academic Excellence in Information Assurance Education (CAE/IAE) and CAE-Research (CAE-R) programs are dictated by the National Security Agency (NSA)/Central Security Service and the Department of Home Land Security (DHS), and are granted for a period of five years, after which the Department has to reapply in order to retain the designation [4].

Centers of Academic Excellence are recognized by the Government and have opportunities of publicity for their role in the Nation‘s priority of securing Cyberspace. Students attending Centers of Academic Excellence have opportunities to apply to scholarships and grants in the Department of Defense (DoD), Information Assurance Scholarship program, as well as the Federal Cyber Service Scholarship for Service Program [4]

The order of this paper is as follows: Section 1.1 presents the Information Technology (IT) program at CU, Section 2 presents the requirements needed in order to be honored as a center of Academic Excellence, Section 3 relates to experience in the implementation of the center at Cameron University, Section 4 relates the experience with students, and Section 5 presents the conclusions and future work.

1.1 Information Technology Program at CU

The IT program at Cameron University is a 2+2 program; students can get the Associate in Applied Science (AAS) in IT as well as the Bachelor of Science (BS) in IT with option in Cyber Security and Information Assurance. The goal of the program is to educate students with appropriate skills and knowledge, with desire to be life-long learners in order to contribute to a rapidly changing world [3].

The requirements for being a Center of Academic Excellence help the IT program to pursue its goal, the University Mission and University Plan 2013: the Center of Academic Excellence, as its name implies, will provide to students access to quality educational opportunities, experiential learning, will prepare them for professional success, responsible citizenship and meaningful contributors to the priorities of our Nation [1].

The University core values of excellence in teaching, scholarship, service, and mentoring; leadership in our community and region, forming partnerships and collaborative relationships, as well as serving the community and region by sharing our expertise; the goals of building distinction in the area of Information Technology, of strengthen partnerships with Fort Sill to provide educational opportunities for soldiers, and of increasing educational partnerships with career technology centers and community colleges, will be part of our requirements for becoming a Center of Academic Excellence [1]. This is a great opportunity to fulfill the University mission and to work toward the accomplishment of the CU strategic plan 2013 (more information on CU strategic plan can be found at [1]).

2. Requirements

The National Centers of Academic Excellence are granted to two years programs in Information Assurance (CAE2Y), and four year institutions in Information Assurance Education (CAE/IAE). A letter of intent must be send to the NSA stating the desire to apply for the CAE program, verifying status as a 2-year institution—for the CAE2Y—, and providing evidence of national or regional accreditation. The letter of intent must be on official institution letterhead, signed at an appropriate level (Dean or higher), and mailed to the NSA Program Office prior to the due date for the CAE application.

As a prerequisite the courseware must be certified under the Information Assurance (IA) Courseware Evaluation Program as meeting the Committee on National Security Systems (CNSS) Training Standard [2]. The CNSS Training Standard 4011 is required and at least one additional CNSS Training Standard (4012, 4013, 4014, 4015, 4016 or subsequent standards) is required.

The NSA additionally requires six criteria for the CAE2Y and nine criteria for the CAE/IAE which will be detailed in Section 2.2.

2.1 CNSS Certifications

As stated in section 2 one prerequisite for getting the National Center of Academic Excellence is to meet at least two training standards from the CNSS including the National Training Standard for Information Systems Security (INFOSEC) Professionals *NSTISSI-4011* which includes the minimum course content for the training of INFOSEC professionals in the disciplines of telecommunications security and automated information systems security [**?**]. At least other CNSS Instruction must be chosen from the *CNSSI-4012* National Information Assurance Training Standard for Senior Systems Managers, the *CNSSI-4013* National Information Assurance Training Standard for System Administrators, *CNSSI-4014* Information Assurance Training Standard for Information Systems Security Officers, *CNSSI-4015* National Training Standard for Systems Certifiers, *CNSSI-4016* National Information Assurance Training Standard for Risk Analysis, or a subsequent standards [2].

2.2 NSA Criteria

The requirements that the Computing & Technology (CaT) Department must meet to become a CAE in Information Assurance Education (CAE/IAE) or a CAE in Research (CAE-R) are dictated by the NSA/Central Security Service and the DHS. Such CAE status is granted for a period of five years, after which we have to reapply in order to retain the designation.

The National CAE in Information Assurance may be granted to two years programs (CAE2Y) and four year programs (CAE/IAE). Since CU has a 2 +2 program in Information Technology (IT) with an option in Cyber Security and Information Assurance, CU is eligible to apply for both. The current plan is to meet the CAE2Y requirements initially and then meet the more substantial CAE/IAE requirements in the following two years.

To meet the requirements of the CAE2Y, an Institution must meet and provide evidence of having met each of the following requirements:

- IA Partnerships evidence must be in the form of an articulation agreement.
- IA Student Development Web site list degrees and IA curriculum with course description.
- IA as multidisciplinary subject IA incorporated in various disciplines.
- IA Outreach Strong collaboration with business, industry, government, and local community.
- IA Faculty assigned specifically to teach and/or develop IA courses/curricula/modules.
- Practice of IA encouraged throughout the Institution - The academic program demonstrates how the institution encourages the practice of IA, not merely that IA is taught.

There are nine requirements for the CAE/IAE. These nine requirements subsume the six for the CAE2Y, so as we accomplish the CAE2Y requirements:

- Outreach/Collaboartion.
- IA treated as a Multidisciplinary Science.
- University Encourages the Practice of IA.
- Academic Program Encourages Student Research in IA.
- Faculty Active in Current IA Practice and Reseach.
- IA Resources.
- IA Academic Programs is Robust and Active.
- Declared Center for IA Education.
- Number of IA Faculty and Course Load.

The system used by the NSA to decide if the requirements are met involves a point system. Points are given for meeting various goals and when the required total points have been reached, the CAE is awarded (See [4]). For example, an articulation agreement with a community college is worth points as is an external speaker coming to campus to discuss Cyber Security issues. CU has already been working toward meeting these goals and has already amassed significant points but there are yet significant points to obtain before the CAE are awarded.

3. Cameron University Implementation Plan.

The six requirements of the CAE2Y are embedded in some of the nine requirements of the CAE/IAE (See Section 2.2). This suggest the general plan of working for getting first the CAE2Y that will benefit initially the student body that is pursuing the AAS, and after getting the CAE2Y, apply for the CAE/IAE that will benefit students in the BS program.

After some discussion, the CaT Department at CU decided that pursuit of a CAE with a target completion time of three years would be in the best interest of the department in spite of the fact that it is a very large undertaking. The Chair and lead faculty immediately started moving forward with the planning and, so far, CaT has been relatively successful overcoming obstacles.

The implementation of the goal to create a CAE at CU heavily involves utilizing the resources provided by the Cyber Security Educational Consortium. The goals of CSEC involve building an information security workforce who will play a critical role in implementing the national strategy to secure cyberspace. Synergies with and resources provided by CSEC are critical to the implementation of a CAE at CU.

The CaT Department is currently working on mapping the CNSS standards 4011 and 4012 to the actual curricula; in articulation agreements with Great Plains Technology Center and Rose State College, reaching out to the community especially at Ft. Sill, incorporating IA and Cyber Security into all of our various programs, developing multiple faculty in the IA area, beginning to work on a web site in support of our Cyber Security and IA programs, and organizing more external speakers to speak to our students and the Cameron community. Success in any one of these items will award us points to be applied to our CAE pursuit. It will take quite a bit of success to achieve our final goals.

3.1 CNSS 4011 and 4012 Course Mapping

The IT program has a common core that includes the following courses: Intro to Computer Information Systems, Intro to Networking, Programming Logic, Intro to Information Assurance/Security, and Fundamentals of Systems Analysis and Design. The program also includes three specialized options: Computer Information Systems, Management Information Systems, and Cyber Security and Information Assurance. The third option was chosen for integration of CNSS 4011 and 4012 course mapping, due to the inclusion of five related courses: Computer Forensics, Information Assurance Networking Fundamentals, E-commerce and Web Security, Security Architecture and Design, and Current Topics in Information Assurance and Network Security.

As the primary goal of the CaT Department was to pursue the CAE2Y, courses chosen for mapping were those taken by students in the AAS degree program. The majority of CNSS 4011 content was already covered in Intro to Information Assurance/Security. However, mapping curriculum for that course to CNSS 4011 helped to ensure a relevant textbook would be adopted, and that support content would be included in Intro to Computer Information Systems, Intro to Networking, and Computer Forensics. Mapping of CNSS 4012 was assigned in the Security Architecture and Design course (See Section 4).

Faculty training was a key component as well. At the beginning of the program, there was only one faculty member responsible for the Cyber Security and Information Assurance option. Currently, there are three faculty members that have received training with support of the Cyber Security Education Consortium and Cameron University. Five core areas proposed by CSEC were part of the training, as well as more specialized training courses in Intrusion Detection Systems, Penetration Testing, and Operating Systems Hardening. As a result of this training, faculty members began to pursue certifications.

3.2 Resources

For the CAE in Information Assurance Education to be successful, it will require CU to provide at least a minimal amount of resources. This includes faculty and staff resources and the use of space and equipment for labs.

A significant amount of the lab facilities are already available. Since the CAE will support the CaT Department and the Information Technology degree programs, funds from what was previously called the Brain Gain Grant are eligible to be used for equipment purchases. Also, it is planned for significant funding to be received via our relationship with CSEC. We believe that we will not need a large infusion of university funds to create required labs, but some funding may be required, especially for maintenance of those labs.

The CaT Department will request that CU provides $1/2$ time release for the Director of the CAE. The staffing request will also include secretarial support and the support of a web/computer/network technician. It is possible that the technician resource can be filled by senior students in IT fulfilling the requirements of an internship.

Initially, the Director will have the responsibility to oversee the pursuit of the fulfillment of the criteria needed for being a CAE. After CU has been awarded the CAE, the responsibility will include the maintenance of such criteria in order to reapply for accreditation when needed. This task will involve keeping the cyber security and information assurance option syllabi up to date, faculty properly trained in bleeding edge Cyber Security issues, partnerships and program agreements, student development, information assurance outreach, research, and publicity. Also, once the CAE2Y is obtained, the director will be responsible for ensuring that the department moves reasonably quickly toward the baccalaureate level CAE/IAE.

Part time secretarial/receptionist support is also requested. This secretary/receptionist will answer the phone, make appointments, and similar activities along with other duties such as typing and filing.

The support of a technician is also requested. The technician would assist the director and faculty in the upkeep of the Cyber Security labs including the computers, networks, and etc. It is also hoped that the technician would be able to maintain the Cyber Security web site. It is possible that we could use one or more student interns to fulfill this role.

A budget specifying previous resources was elaborated.

4. Student Perceptions

In the fall of 2011, the Cameron University Department of Computing and Technology course *IAS 3263 Security Architecture and Design* was structured with a focus of including *CNSSI 4012* elements in the curriculum. The students were made aware of this in the course syllabus and by the instructor. Discussions also ensued on the possibility of the department working toward becoming an NSA Center of Academic Excellence. In a course assignment, students were given the task of conducting research on *CNSSI 4012* and Academic Center of Excellence guidelines. They then had to provide their perceptions, related to possible implementation in the course and in the department. Below are a few of the student responses taken from the assignment.

- Overall, accreditation from the NSA is something that should be highly sought after — the accreditation means that the university is able to provide a wide range of information and resources dealing with IA, while still holding the guidelines that are used by professionals in the workplace. On top of that, the centers that are accredited, are also considered models for the nation, in terms of education of IA.
- Following research of this topic, I have come to learn that becoming a Center of Academic Excellence is a top priority of the Department of Computing and Technology, as well it should be. This program was set up by various government agencies in order to educate and train the next generation of network security administrators to protect our country from foreign cyber threats. The importance of this training could give our country the upper hand in any conflict. Being able to protect important information about our military, government, and economy is of the utmost importance and should not be taken lightly.
- From my experience so far at Cameron University, most of our assignments are writing papers, which is good, but a lot of times I feel like we should be doing more practical assignments to better understand how what we read can be applied in the real world. With that said, we can be given information security case scenarios, and then use the guidelines from *CNSSI 4012* to address the issues. For example, *CNSSI 4012* comprises of ten functions. A scenario can be given in which network vulnerabilities have been found, and we would have to refer to function ten (assess network security) to address the issue.
- Based on the functions which are similar to *IAS 3263*, some possible ways that the *CNSSI 4012* can be used in this *IAS 3263* course are: furthering awareness and knowledge of countermeasures, defenselessness or vulnerability with security systems, risk management and policy direction, acquisition and life cycle management, assigned responsibilities, allocation of resources, and others. These areas can be addressed and explained in more detail, so that new IA professionals can be better equipped in outside workplace settings to help improve systems infrastructures.

The authors believe student understanding and perception to be an important factor to consider in becoming a Center of Academic Excellence. This particular exercise in *IAS 3263* also seems to have helped the students to take ownership of their cyber security education, while providing a sense of realism beyond the classroom.

5. Conclusions and Future Work

This paper presented the process of the creation of a Center of Academic Excellence in Cyber Security and Information Assurance at Cameron University, as an experience that can be taken by other Institutions in their pursue of excellence. A CAE will help Departments in their efforts to fulfill standards in IA that will benefit academia in general and Students in particular. In the case of Cameron University, the CAE will help it solidify its position as the University of Choice by attracting high-quality students. These students will benefit from the high quality of teaching provided by outstanding faculty accompanied by significant opportunities for experiential learning. Students will have the opportunity to receive significant scholarship funding from the DoD and other government scholarship programs. Finally, the department will reach out to the community creating new partnerships with technology centers, other colleges, and forge and/or cement partnerships with the industrial community including the local military industrial complex and local companies.

It is the opinion of the CaT Department that the creation of the proposed CAE in Cyber Security and Information Assurance would be a very positive step for Cameron and be very beneficial to the university, the faculty, and to the students. This is a great opportunity to fulfill the University mission and to work toward accomplishing the goals of the CU's strategic plan 2013.

References

[1] Cameron University, "PLAN 2013. Choices for the Second Century," 2008, accessed June 2011. [Online]. Available: http://www.cameron.edu/uploads/If/Ee/IfEegoinf7abnI5_ItLG5g/Plan-2013.pdf

[2] Committee on National Security Systems (CNSS), "Instructions," accessed June 2011. [Online]. Available: http://www.cnss.gov/instructions.html

[3] P. A. Diaz-Gomez, M. Estep, J. Carroll, and R. Kamali, "Information Technolgy Program Assessment at Cameron University," in *Proceedings of the 2009 on Enterprise Information Systems and Web Technologies*, 2009.

[4] National Security Agency, "National Centers of Academic Excellence." 2010, accessed June 2011. [Online]. Available: http://www.nsa.gov/ia/academic_outreach/nat_cae/index.shtml

[5] National Security telecommunications and Information Systems Security (NSTISS), "National Training Standard for Information Systems Security (INFOSEC) professionals," 1994, accessed June 2011. [Online]. Available: http://www.cnss.gov/assests/pdf/nstissi_4011.pdf

Comparative study between Professional Competences Developed in Engineering and Social degrees

R. Lacuesta-Gilaberte[1], G. Palacios-Navarro[2] and P. Ramos-Lorente[2]
[1]Dept. of Systems Engineering and Computing, University of Zaragoza, Zaragoza, Spain
[2]Dept. of Electronic Engineering and Communications, University of Zaragoza, Zaragoza, Spain

Abstract – *After several studies that we have carried out on engineering about competences acquisition in Computing Engineering, we have developed a pilot experience analyzing science and social degrees competences acquisition. We have compared these data with previous results. The analysis offers a qualitative perspective, comparing the development of these competences in engineering students with other areas. The comparison of competences development among different degrees let us know the way they are developed taking into account the type of degree. The work tries to gather student opinions of different knowledge areas. Obtained results and conclusions will serve us to improve our processes as well as to implement the necessary changes involving the convergence towards the European Space for Higher Education (ESHE) in the best conditions.*

Keywords: computer engineering curriculum, professional competences, pilot study, teaching methodologies, tuning project.

1 Introduction

The general directives to the adaptation to the European Space for Higher Education (ESHE) [1] are related to the competitiveness of the European Higher Education Area (EHEA), as well as to the incorporation of students in the industrial and business world. The vertiginous technological advance of our society entails continuous learning and this is generating a necessity of using educational models that allow students to "learn to learn". On the occasion of the transition to ESHE, the debate about the role of competences in university graduates has been reopened, especially concerning generic competences or transferrable skills.

The competence-based training approach provides a common language, at an international level, to define and express optimal academic and professional profiles, together with the configuration of the new educational programs in accordance with the principles of the construction of the European Space for Higher Education.

The incorporation of competences into educational programs is a basic element for student education, in the framework of a society that reformulates its demands in a constant way, aimed at bringing university formation closer to the labour market. For this reason, current and future educational programs must be designed from the point of view of professional competences acquisition, in such a way that graduates turn into competent people at their jobs. To perform that, we believe that the main change must take place in the teaching methodology.

The Teruel Polytechnic School of Engineering has been developing actions of innovation and educational improvement for several years. Actions carried out by the different management boards and by teaching committees of our university have tried to involve both lectures and students. These actions have achieved an improvement in students' academic performance.

After several years of study we have developed a deep analysis about generic competences aimed at obtaining a valuation to the subsequent analysis and reflection. On this paper we go further and we try to compare the development of these competences in engineering students with other areas. We do it throw the participation and opinion of students and lecturers belonging to the different degrees. The present paper shows the global study fulfilled for the degrees: Fine Arts, Teacher Training, Labor Relations, Computing Engineering, and Telecommunication Engineering.

Several sources have propounded the advantages offered by considering different methodologies in order to acquire the skills and competences students will need in their future jobs. Interpersonal communication, teamwork, group problem-solving, leadership, negotiation and time management [2-5] are examples of the new competences. In addition, positive effects have been shown in students' academic performance, in motivation and their attitudes towards learning [6].

Some of these advantages have also been underlined by students, who consider group activities and active methodologies to be more interesting and entertaining than traditional teaching [7].

This paper is presented as follows. Section 2 introduces generic competences. Section 3 analyzes the survey results, where we focus our attention on generic or transferable competences and section 4 deals with the main conclusions.

2 Generic Competences in Engineering Education

The Spanish Organic Law of Universities [8], in its first article establishes "The creation, development, transmission and criticism of Science, Technology and Culture" as University first function. Consequently, students must develop intellectual, technical, artistic, social and personal abilities. These abilities or competences will encourage creativity, problem solving and autonomous learning through all their life.

Competences represent a dynamic combination of knowledge, understanding, skills and abilities. Fostering competences is the object of educational programs. Competences will be formed in various course units and assessed at different stages and can be distinguished between subject specific and generic ones. Tuning [9] project has highlighted the fact that time and attention should also be devoted to the development of generic competences or transferable skills. This last component is becoming more and more relevant for preparing students well for their future role in society in terms of employability and citizenship. Tuning [9] proposes a total of 30 generic competences classified into three types:

- Instrumental competences: cognitive abilities, methodological abilities, technological abilities and linguistic abilities.
- Interpersonal competences: individual abilities relating to the capacity to express one's feelings, critical and self-critical abilities. Social skills relating to interpersonal skills or team-work or the expression of social or ethical commitment. These tend to facilitate processes of social interaction and of co-operation.
- Systemic competences: abilities and skills concerning whole systems (combination of understanding, sensibility and knowledge; prior acquisition of instrumental and interpersonal competences required).

We have considered these generic competences to develop our study. that were specified in the guide proposed by ANECA [10], where the importance of each of them was valued starting from the results analysis of surveys carried out to company communities, qualified people and lecturers.

2.1 Instrumental competences

Having an instrumental function, the main objective of these competences is to provide students with means and methods allowing them to use put knowledge in practice in the work environment. They include:

- Cognitive abilities, capacity to understand and manipulate ideas and thoughts.
- Methodological capacities to manipulate the environment: organizing time and strategies of learning, making decisions or solving problems.
- Technological skills related to use of technological devices, computing and information and management skills
- Linguistic skills such as oral and written communication, or second language knowledge.

According to Tuning, cognitive and methodological abilities are fundamental, as they will provide the student with the leadership nature.

Competences taking into account in our study include:

- Capacity for understanding and interpreting written information in a critical way (C1).
- Capacity for organization and planning (C2).
- Oral communication in your native language (C3).
- Ability to communicate ideas in different contexts (C4).
- Capacity for understanding and interpreting written information in a second language (C5).
- Written communication in your native language (C6).
- Knowledge of a second language (C7).
- Elementary computing skills (C8).
- Information management skills (ability to retrieve and analyze information form different sources) (C9).
- Capacity for analysis and synthesis (C10).
- Problem solving (C11).
- Decision-making (C12).

2.2 Interpersonal competences

Abilities to communicate and teamwork are fundamental in the success, increasing possibilities of promotion. Communicating solutions to workgroups (social competences) is one of the main goals demanded nowadays.

We may remark ethical commitment and interpersonal skills as essential to manage groups of heterogeneous people. The competences considered are the following:

- Ability to work in a team with responsibility and flexibility (teamwork) (C13).
- Ability to work in an interdisciplinary team (C14).
- Ability to work in an international context (C15).
- Interpersonal skills (C16).
- Appreciation of diversity and multiculturality (C17).
- Ethical commitment (C18).
- Critical and self-critical abilities (C19).

2.3 Systemic competences

Students should have the ability to understand the social and economic context belonging to the sector in which they develop theirs activities. For this reason, it is necessary for them to show an enterprising attitude to find possibilities and to define objectives, together with the managing and teamwork abilities. We consider the following systemic competences:

- Ability to work autonomously (C20).
- Capacity to adapt to new situations (C21).
- Capacity for generating new ideas (creativity) (C22).
- Leadership (C23).
- Understanding of cultures and customs of other countries (C24).
- Initiative and enterprising spirit (C25).
- Concern for quality (C26).
- Sensibility to environmental matters (C27).

3 Practical Experience. Results

During the academic year 2010/11 we designed a survey to be completed by students belonging to the degrees previously presented. We have centred our study on the analysis of generic competences. They survey was completed by more than six hundred students belonging to the aforementioned degrees. We show in this section several comparative analyses with the collected data at the end of the second term.

In order to gather opinions we decided to carry out a quantitative analysis. The so chosen type of survey in this case was delivered to the different classrooms belonging to the aforementioned degrees.

The questions are shared by all the degrees, since we have seen there are 27 questions related to competences (1 to 12 instrumental, 13 to 19 personal, and the rest belong to systemic competences.

In addition, to be able to value the mentioned aspects in the purpose of the survey, it is divided in three columns related to the importance that every student/professor gives to each competence. In this point, the importance in the labour market and the way they have been developed at classroom are taking into account. On the other hand, we have to remark that the scale of valuations varies from 1 to 5, in order to get data having enough meaning.

Fig. 1 shows the development of generic competences within the current year for the aforementioned five degrees, having marks above 3. Social degrees focus on the development of generic competences (to a greater extent) whereas engineering degrees focus on the development of professional competences. We separately analyze every type of competence to guess which are more developed taking into account the degree. We will begin our analysis with instrumental competences, after them, personal ones and finally the systemic ones.

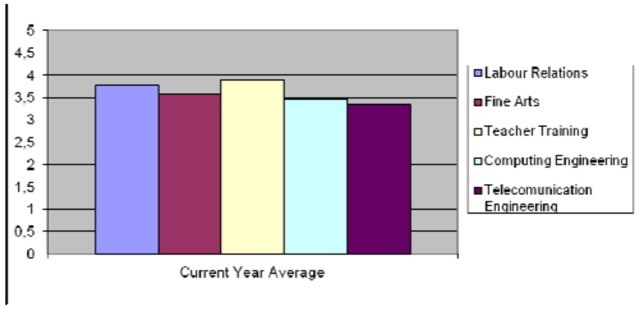


Figure 1. Average of **c**ompetences valuation.

Fig. 2 shows the existing discrepancy among the importance given by students at both personal level and labour market in comparison with the way they have been developed in the current year (according to students' perception). The same conclusion will be observed in the following figures.

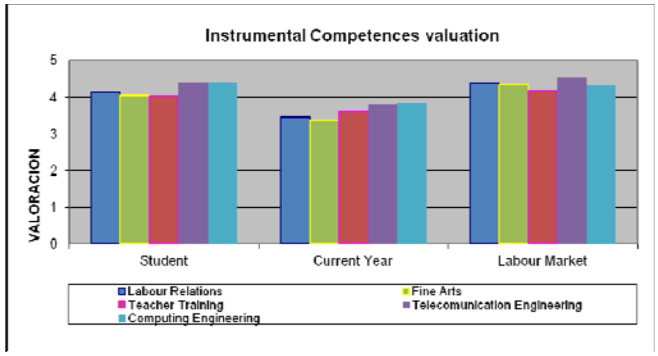


Figure 2. Instrumental Competences valuation.

On the other hand, if we focus the analysis on the current year, we also observe that in spite of the fact that engineering degrees develop to a lesser extent generic competences (in general), these technical degrees instrumental competences develop to a greater extent, such us: Information management skills (ability to retrieve and analyze information form different sources) (C9), Capacity for analysis and synthesis (C10), Problem solving (C11), among others.

Fig. 3 shows that personal competences are developed to a greater extent in social degrees, in comparison with engineering degrees. Remarkable are the following: Ability to work in an interdisciplinary team (C14), Appreciation of diversity and multiculturality (C17), Critical and self-critical abilities (C19) among others.

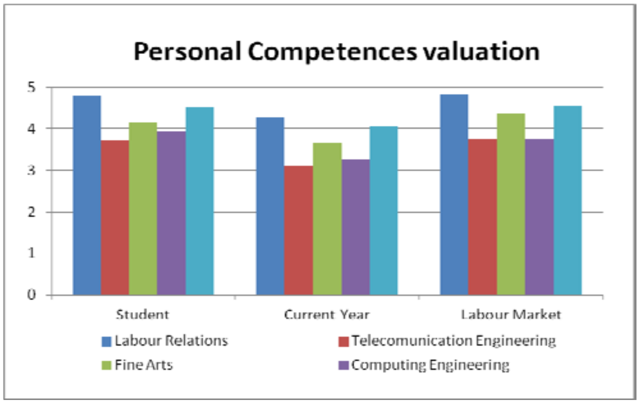


Figure 3. Personal **c**ompetences valuation.

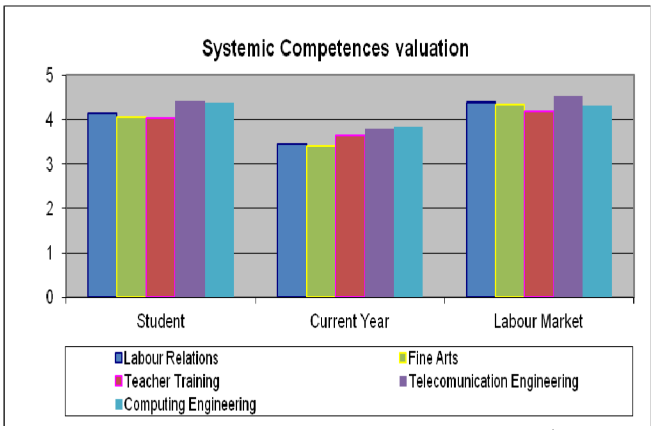


Figure 4. Systemic competences valuation for 1st year of Computing Engineering degree.

Fig. 4 shows the same feature (systemic competences are developed to a greater extent in social degrees. Remarkable are the following: Ability to work autonomously (C20), Capacity to adapt to new situations (C21), Concern for quality (C26).
We will centre our analysis in every single degree, seeking the strongest points of every single one.

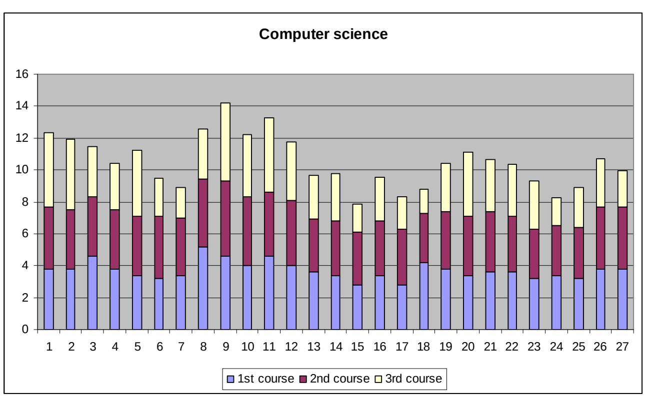


Figure 5. Computing Engineering competences valuation.

We can observe in fig. 5 competence development belonging to the Computing Engineering degree. The development of instrumental competences is remarkable, especially 9 and 11 competences: Information management skills (ability to retrieve and analyze information form different sources) (C9) y Problem solving (C11).

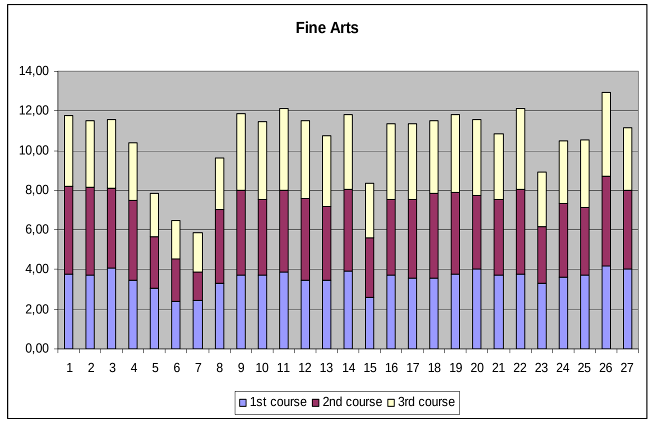


Figure 6. Fine Arts **c**ompetences valuation

Figure 6 shows competence development in the Fine Arts degree. Development of systemic competences is remarkable, especially: Capacity for generating new ideas (creativity) (C22), Concern for quality (C26), as well as Instrumental Problem solving (C11).

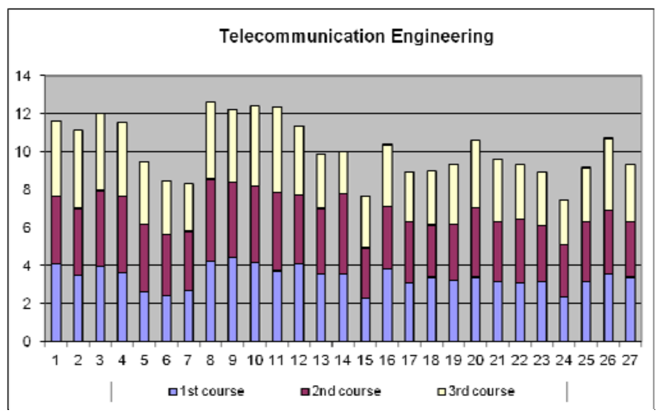


Figure 7. Telecommunications engineering competences valuation.

Fig. 7 shows that the most developed competences in the Telecommunication degree are instrumental: Elementary computing skills (C8), Information management skills (ability to retrieve and analyze information form different sources) (C9), Capacity for analysis and synthesis (C10) and Problem solving (C11).

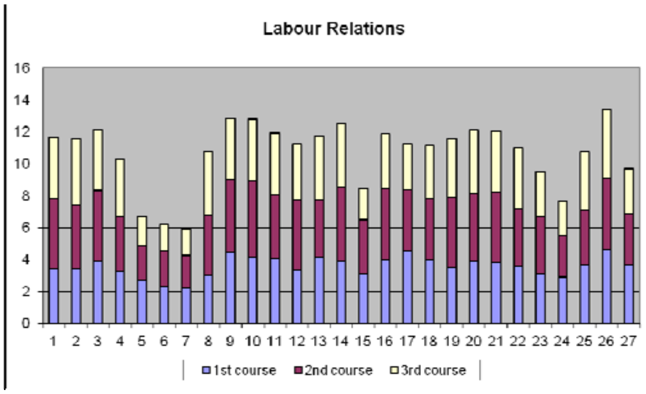


Figure 8. Labour Relations **c**ompetences valuation.

According to the Labour Relations degree (see fig. 8), systemic competence 'Concern for quality (C26)' and instrumental competence 'Information management skills (C9)' are valued above others.

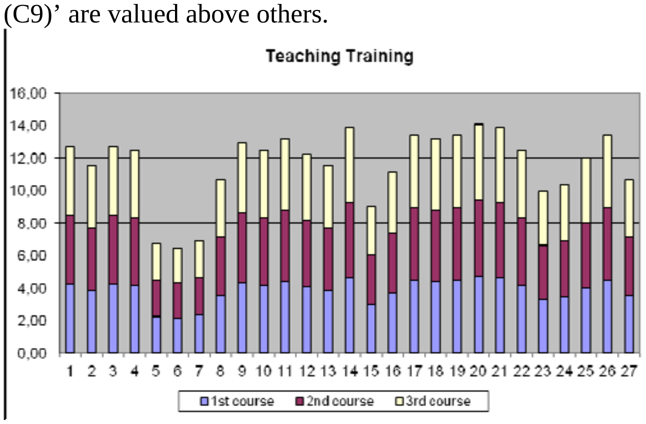


Figure 9. Teaching Training competences valuation.

Within the Teacher Training degree (see fig. 10), the most valued competence is a systemic one (Concern for quality (C26)), followed of Information management skills (ability to retrieve and analyze information form different sources) (C9), Capacity for analysis and synthesis (C10) and Ability to work in an interdisciplinary team (C14).

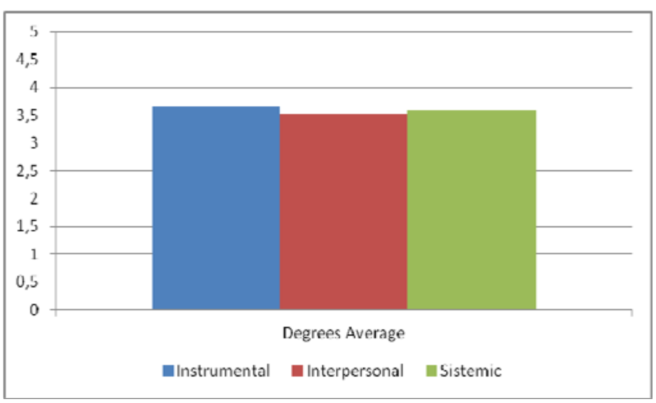


Figure 10. General Competences valuation.

An overall view of the situation (fig. 10) shows that instrumental competences are most valued on average. Nevertheless, values are closer each other.

4 Conclusions

In this work, we have dealt with the importance of competences acquisition; in particular, we have focused our analysis in the degrees taught on Teruel Campus. Due to the forthcoming implementation of the new educational model focused on the student, teaching/learning activities should be carefully designed and schedule to reach the educational goals as far as competence acquisition is concerned. Results obtained in practice will let us correct the tested methodological strategies and work on their improvement. Taking into account competences demanded by companies, we may conclude that, in general, competences most valued by students are really useful in the labour market.

A general lack has been observed in the valuation of competences such us aptitude to communicate (oral and written) in another language, or the knowledge of foreign cultures, for instance. In this respect, lecturers will have to emphasize their importance, trying to develop them at the classroom.

They are slightly differences among valuations, pointing out that mean values are above 3 (out of 5), showing a suitable interest in all the areas. Nevertheless, competences development in the current year is below expectative of importance. Getting the right balance between both aspects is crucial.

5 Acknowledgement

This work has been funded with support of the University of Zaragoza under "PIECyT_11_1_228" project.

6 References

[1] Joint declaration of the European Ministers of Education Convened in Bologna on the 19th of June 1999, European Ministers of Education, 2004 [Online]. Available: http://www-en.us.es/us/temasuniv/espacio-euro.

[2] M. K. Bolton, "The Role of Coaching in Student Teams: A "Just-in-Time" Approach to Learning", *Journal of Management Education*, Vol.23, No.3, pp. 233-250. 1999.

[3] A. P. Christoforou et al., "Improving engineering education at Kuwait University through continuous assessment", *International Journal of Engineering Education*, Vol.19, No.6, pp. 818-827. 2003.

[4] R. Fruchter, “Dimensions of teamwork education, International”, *Journal of Engineering Education*, Vol.17, No.4-5, pp. 426-430. 2001.

[5] K. Sheppard, P. Dominick and Z. Aronson, “Preparing engineering students for the new business paradigm of international teamwork and global orientation”, *International Journal of Engineering Education*, Vol.20, No.3, pp. 475-483. 2004.

[6] C.W. Holtham, R. R. Melville and M. S. Sodhi, “Designing Student Groupwork in Management Education: Widening the Palette of Options2, *Journal of Management Education*, Vol.30, No.6, pp. 809-817. 2006.

[7] D. R. Bacon, K. A. Stewart and W. S. Silver, “Lessons From the Best and Worst Student Team Experiences: How a Teacher Can Make the Difference”, *Journal of Management Education*, Vol.23, No.5, pp. 467-488. 1999.

[8] Cortes Generales, ‘Ley Orgánica de Universidades’, Boletín Oficial de las Cortes Generales, vol 45-13, 26th December 2001, pp.463 – 495 (‘Organic Law of Universities’, Official publication of the Spanish Congress).

[9] http://tuning.unideusto.org/tuningeu/ Website of Deusto University. Last visit: October 2009.

[10] ANECA (Spanish National Agency for Quality Assessment and Accreditation). Website: http://www.aneca.es/activin/docs/. Last access: January 2010.

[11] F. Sánchez et al, “Professional Competences for the Computer Engineering degree”. Polytechnic University of Cataluña. 2009.

Using Crowdsourcing to Enhance Crisis Management

Alfred C. Weaver and Joseph P. Boyle
Department of Computer Science, University of Virginia, Charlottesville, VA 22901, USA

Abstract - *In the event that a natural disaster or terrorist attack strikes, it would be valuable for both individuals and officials overseeing the response to have access to the best possible information regarding what has happened, what is happening currently, what could possibly happen in the future, and where these events are occurring. "Crowdsourcing," the use of a group of people or community to achieve a desired goal, is a potential method for gathering this information. Here we present the design and implementation of a crowdsourcing application suite that facilitates the gathering and display of information from individual citizens during a crisis. Our goal is to provide "ground truth" (multiple citizen observations) to enable a citizen to make an informed decision about his or her health and welfare (e.g., shelter-in-place vs. flight). We explore how the system is used and potential improvements for the future.*

Keywords: crowdsourcing, disaster management, mobile computing, geolocation, intelligence gathering.

1 Introduction

In the event of a bioterrorism attack, the government is unlikely to be adequately prepared or equipped to gather information and disseminate it to the public [2]. Yet the growing use of mobile phones and social media is responsible for the generation of masses of information every day [7]. If we can use these avenues to collect important information, we would have a valuable real-time tool for managing and understanding situations as they unfold, rather than relying solely on retrospective analysis.

In some applications, crowdsourcing has improved efficiency [11] and businesses are increasingly relying on crowdsourcing as a method for solving problems and making predictions [4]. The usefulness of this approach resides in the sheer number of participants and the vast amount of information that can be generated quickly [1]. Along with partners in the UVa School of Medicine, we propose to apply this methodology to crisis response. Figure 1 shows a map that identifies the location of reports received from the "crowd."

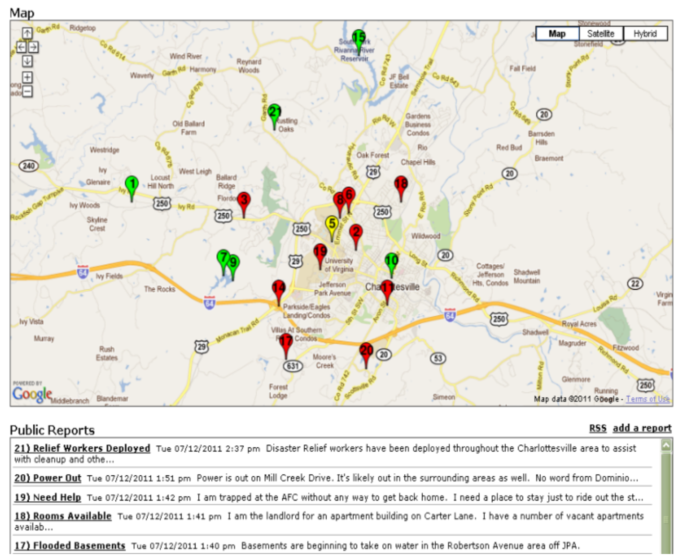


Figure 1. This is an example of report geolocation. Reported locations are converted to GPS locations and plotted on a Google map. Tag colors correspond to the reliability levels of the reports. Numbers correspond to report numbers in the reports list below the map.

2 Methodology

To approach this problem we sought to develop a web-based application that would provide users with information submission and viewing functionality. Because of the widespread availability and use of the Internet, this would be the most effective way to reach a large audience of potential contributors. We began by conducting a survey of existing open-source platforms for crowdsourcing. For this project, we identified three major software systems as contenders for our application base and the features and capabilities of each were compared.

2.1 Ushahidi

The Ushahidi platform was developed in Kenya following the post-election fallout of 2007. It was intended as a way to monitor both violence and peace efforts throughout the country to provide up-to-date information to interested parties. Ushahidi has since become a major player in the crowdsourcing market, with applications in India, Mexico, Lebanon, Afghanistan, the Democratic Republic of Congo, Zambia, and the Philippines, ranging from monitoring elections to tracking medicine supplies [10]. Recently, it has even been deployed in the United States in response to the North American Blizzards of 2010 to help identify areas in need of assistance and help individuals connect with each other to assist in relief efforts [5].

The platform can handle reports from multiple data streams including text messages, email, twitter, and web forms. It has an interactive mapping capability as well as a dynamic timeline tool to track events. Additionally it can integrate with SwiftRiver [9] for information processing and analysis and with Frontline SMS [3] for sending alerts to participants.

2.2 Sahana

Sahana was developed specifically with disaster management in mind. It was built to assist those suffering during the Asian Tsunami in 2004 by volunteers in the Sri Lankan FOSS (Free and Open-Source Software) community. It has since been deployed to assist in the response to the earthquakes in Haiti and Chili in 2010, as well as in the 2011 earthquake and tsunami in Japan, among others [8].

As with Ushahidi, Sahana provides basic reporting and display functionalities based on geospatial analysis. In addition, it enables coordinating and managing aid, as well as tracking shelters. Sahana also has a missing person registry, methods for communications and alerts, and standalone applications for iPhone and Android.

2.3 Crows

The third system we considered was significantly smaller in scope than the previous two. Crows was developed by the Pittsburgh Independent Media Center for coverage of the G20 summit and subsequent protests in Pittsburgh in September 2009 [6]. Crows has not seen widespread use as a crowdsourcing platform, possibly due to its limited functionality.

Crows provides the very basics of crowdsourcing: the ability to submit reports and view them on a Google map. It also includes widgets for Twitter, Flickr, YouTube, and Podcast integration as well as a smartphone-compatible mobile version of the website.

Ultimately, we decided that the Ushahidi and Sahana systems were too large for us to manage given our limited resources. Thus we chose Crows, whose smaller size made management more feasible for this project. Additionally, the lack of certain features enabled us to customize the platform to our own particular needs and develop a more targeted application.

3 System Design

We began development by identifying some of the features that we felt would be the most valuable to our application (including some present in the alternative platforms) and implementing them in our selection. Our goal has been to try to implement a new feature every week over the course of the year. We also have had several meetings with representatives from the American Red Cross and FEMA (Federal Emergency Management Agency). These meetings have been helpful in identifying new useful features to add as well as gaining visibility for the system we are developing and bringing attention to the possible use of crowdsourcing for crisis response.

At this point, our development has resulted in a system capable of using the crowdsourcing model to collect and display information. This system includes a desktop web application, a mobile web application, and a standalone application for the Android platform. Together, these components provide a number of key functionalities and features.

3.1 Submitting Reports

First, the system allows users to submit reports of events through our website (desktop and mobile version) as well as through email. The format of the report includes a headline, the reporter's name (optional), the location (optional), report classification (e.g., traffic, weather), the content of the report, and links to relevant content elsewhere. These reports can be made either anonymously or by registering with the site and logging in. Users can also add links to relevant information (including photos, videos, and news articles) to their reports. Individuals may also upload their own photos (hosted on our server) and link to them in a report.

3.2 Viewing Reports

Second, our site allows all users to view reports that have been made. These reports are shown on an embedded Google map with colored tags identifying each report as well as in a list format below the map (figure 2). Colored flags correspond to the different reliability levels associated with a report (green is highly reliable, yellow is moderate, and red is low or unknown). These flags are also numbered to correspond with the numbered reports in the list below. The map also has the ability to be filtered. Users can choose to see only certain reports by selecting the desired reliability level and report classes. The site

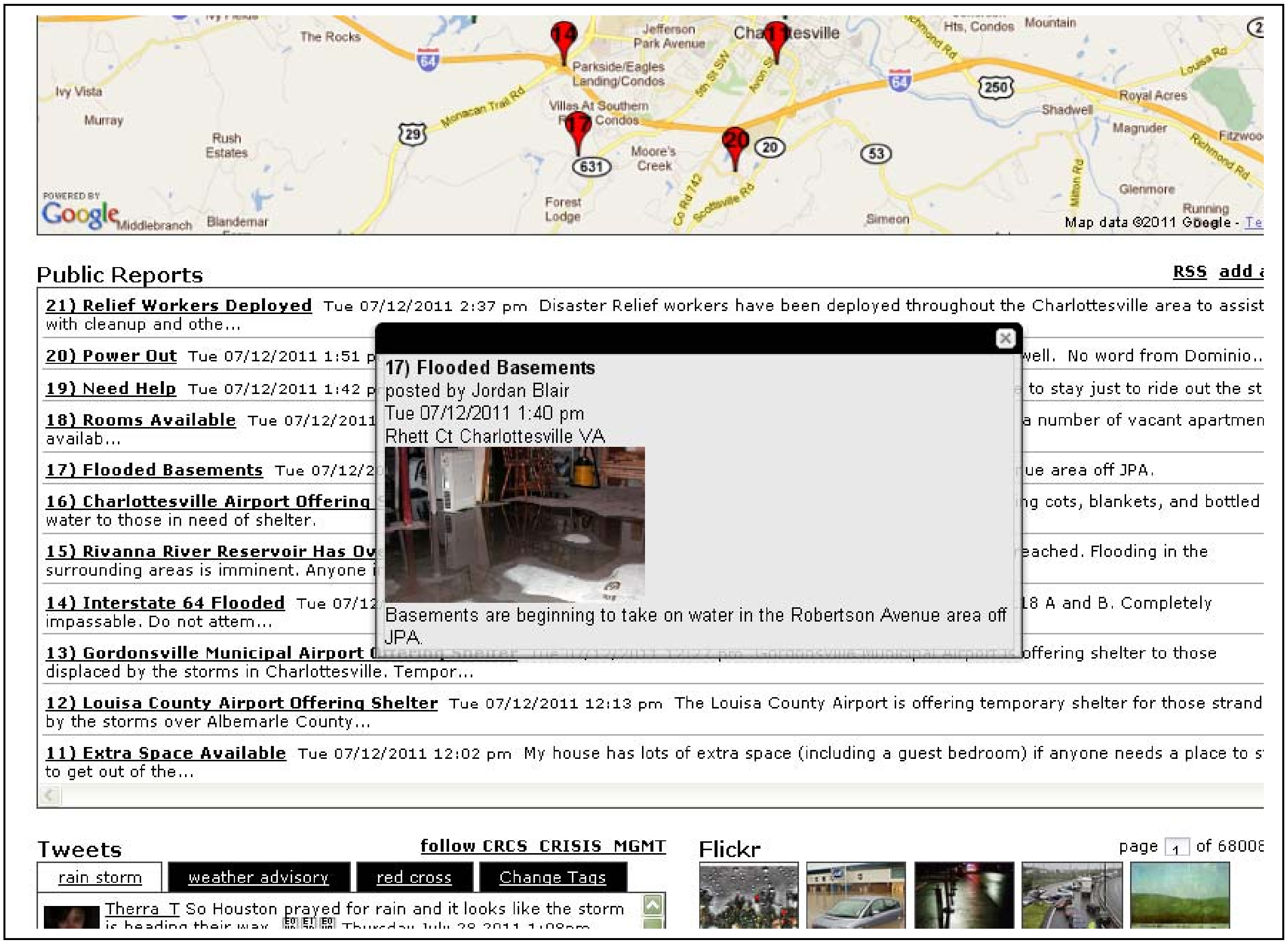


Figure 2. Sample report accessed from the report list. Clicking a report headline brings up a window with all of the report contents.

also includes embedded Twitter and Flickr widgets, showing content from these sites that may be relevant to a particular situation (identified by supplied keywords). These media can be directly included in a report through these widgets.

3.3 Registration

Next, we have the ability for our users to register with our site to gain access to more features. Users register with an email address and have the option to register their mobile phone to receive alerts. A text message is sent to users who elect to register their phone number to confirm the registration. Registering with the site also allows participants to have access to the higher levels of reliability.

3.4 Group Membership and Reliability

Report reliability is determined by group membership. The site allows users to create and join different groups, and these groups may request elevated reliability levels from a system administrator. Access to different groups is controlled by a group administrator, who is responsible for giving individuals the required password to join the group. An individual user's reliability level is the highest reliability level of all groups of which the individual is a member. This allows a user to be a member of many different groups without sacrificing reliability.

3.5 Personal Profile

When a user registers with the site, he also has access to a personal profile page. This page provides an interface for the individual to view and modify information about him. Here, personal information that was supplied at registration can be updated. Users can also view and modify their group memberships as well as see all of the photos they have uploaded to the site (with links provided so that they may be included in future reports).

3.6 Alerts and Timelines

A critical component of the system is the ability to send mass alerts to participating individuals. At any time, a system administrator may send out an alert message to individuals who are registered with the system. The content of this alert is sent in an email (so that every registered person receives the information) as well as via text message to those users with mobile phones registered in the system (figure 3).

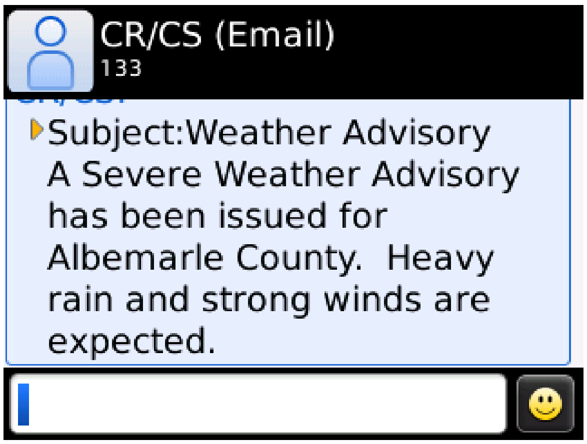


Figure 3. Sample alert text message as viewed on a Blackberry smartphone.

Finally, the system allows users to view a timeline of the event to see a chronological depiction of how the event played out and how reports were made. This tool is useful for identifying how much time passed between an occurrence and subsequent reports, as well as between a report and the time that report was resolved. The timeline is not updated in real time. Instead it requires the content of the database to be exported and parsed before the information is viewable. As such, it serves more as a retrospective analysis tool.

3.7 Mobile Web and Android Applications

The mobile website possesses a subset of these functionalities, including the ability to submit and view reports (in text and on a Google map, see figure 4), as well as upload photos. The website can be accessed from any mobile device with Internet capabilities and the map can be viewed if the mobile browser supports JavaScript.

The Android application possesses all of the core functionalities of the system (viewing reports, submitting reports, and uploading photos) just as the mobile website does. Additionally, it supports logging in and logging out. The application also takes advantage of the GPS device inside of the phone to provide more advanced location features. Specifically, it allows the user to filter reports in the list based on their distance from the user's current location. Similarly, the map provides the option to center on the user's current location, eliminating the need to scroll and zoom manually to find nearby reports. As another convenience, the application allows the user to press a button to automatically fill in her current location and also automatically fills in the user's name if she is logged in. When users submit reports, they can also connect directly to the device camera, if one exists, to include photos. See figure 5.

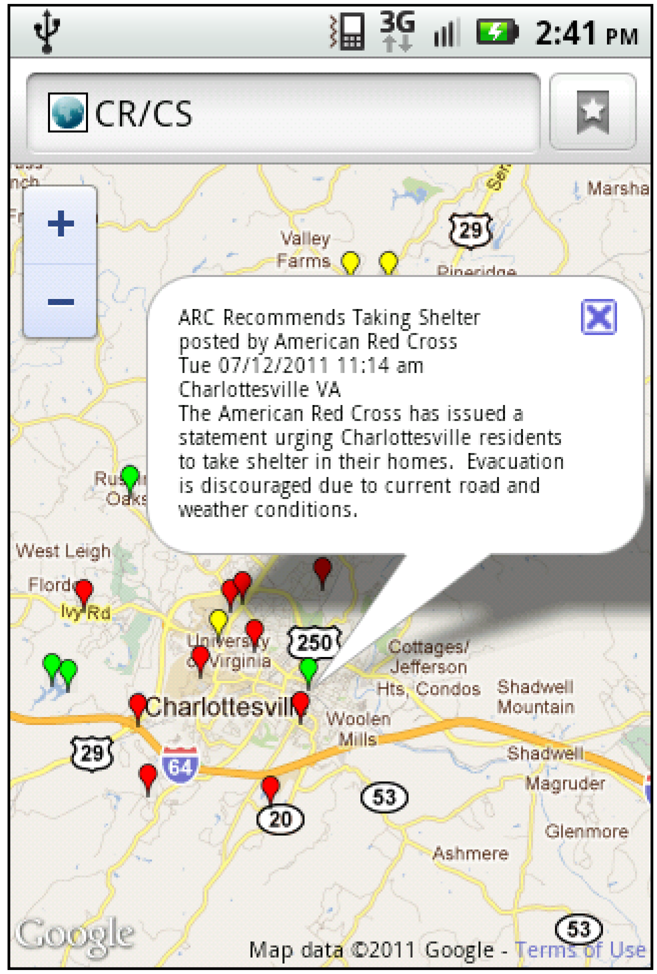


Figure 4. Mobile website map as seen on an Android smartphone with report tag clicked. A popup dialog appears on the click to display the contents of the report.

4 Summary and Future Work

In summary, we have successfully developed a suite of applications (desktop web, mobile web, and standalone mobile) that work together and can be used for crowdsourcing information. We have the critical ability to gather and display information from participating users as well as supplemental features that make the system a more pertinent application to aid crisis and emergency response.

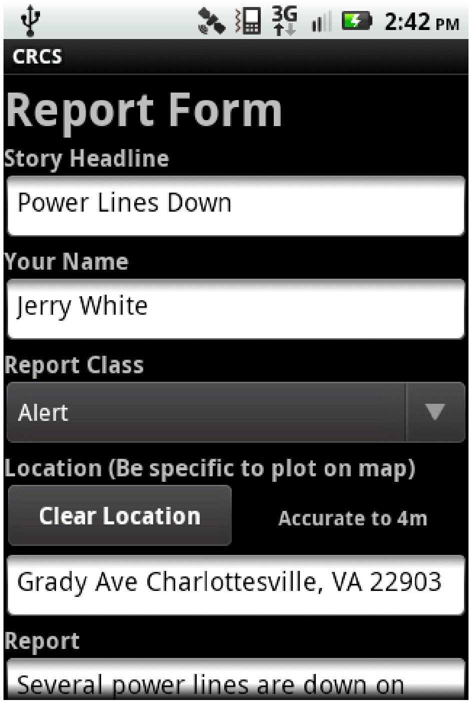


Figure 5. Report form on the standalone Android application populated with sample data.

We continue to work on improving the website functionality (both the desktop and the web versions). In particular, we hope to expand the web version of the system to include an improved timeline tool. We would like to have the timeline update in real time as well as show similar color and number identifiers as are seen on the Google map. Additionally, a valuable new feature that we would like to add is the ability to modify reports or mark them as completed. For example, if a report identified that power is out in a certain area, we would like to be able to annotate that report to show that is has been resolved once the power has been restored to maintain the integrity of the reports.

A major area of improvement we are focused on is the addition of an improved reliability model. We are currently working to develop a reputation system that will integrate with our tools. With this system, users would be able to interact with reports by giving them ratings are well as a general agreement or disagreement response. The model will also allow users to submit comments on reports and see the comments left by others. Based on these inputs, reports will be scored and this score then becomes the basis of reliability rather than relying on administrator assigned values. This would allow the integrity of a report to change over time, which is currently not possible unless a member's group membership changes. Users themselves will also have a rating based on how they interact within the system, allowing them to earn a higher reputation by confirming other reports, identifying incorrect ones, etc. We are also working to improve the functionality of the mobile site to support the use of all web-capable phones.

In addition to improving the functionalities of the system, we are working jointly with members of the University of Virginia Office of Emergency Preparedness to develop an experimental procedure to test our system in a real scenario. The results of this experiment will be extremely important in determining the success of our system.

5 Acknowledgments

We gratefully acknowledge the work of our colleagues in the University of Virginia School of Medicine, including Greg Saathoff, M.D., and Chris Holstege, M.D., as well as our partner in the Department of Defense, Ms. Diane Kotras. We also thank the American Red Cross and FEMA for their interest and assistance throughout the course of this project. Funding was provided by the United States Department of Defense Energy Support Center under grant "Community Shielding – Year 1," subcontract SP0600-08-C-583.

6 References

[1] Brabham, D. Crowdsourcing the Public Participation Process for Planning Projects. Planning Theory, 8(3), 242-262, 2009.

[2] Commission on the Prevention of Weapons of Mass Destruction Proliferation and Terrorism. World at Risk: The Report of the Commission on the Prevention of WMD Proliferation and Terrorism. *Report to Congress*, 2008.

[3] FrontlineSMS. http://www.frontlinesms .com/.

[4] Giles, J. Wisdom of the Crowd. *Nature, 438*(7066):281, 2005.

[5] PICnet. Who's Behind This. *Washington Snow Cleanup*. http://snowmageddoncleanup.com/page/index/2.

[6] Pittsburgh Independent Media Center. The Pittsburgh Independent Media Center Releases Crows 1.0. http://crowsne.st/pr/Crows_Press_Release.pdf.

[7] Raento, M., Oulasvirta, A., Eagle, N. Smartphones: An Emerging Tool for Social Scientists. *Sociological Methods and Research, 37*(3):426-454, 2009.

[8] Sahana Software Foundation. About the Sahana Software Foundation. *Sahana Home of the Free and Open Source Disaster Management System*. http://www.sahanafoundation.org/about.

[9] Ushahidi. SwiftRiver: A Ushahidi Initiative. http://www.ushahidi.com/-/docs/SwiftRiver_1-Pager.pdf.

[10] Ushahidi. http://www.ushahidi.com/docs/Ushahidi_1-Pager.pdf.

[11] Vukovic, M., Laredo, J., & Rajagopal, S. Challenges and Experiences in Deploying Enterprise Crowdsourcing. Lecture Notes in Computer Science, 6189, 460-467, 2010.

Building Mobile Mashup Applications. Some Challenges Encountered in Computer Science Degrees

Javier Díaz[1], **Claudia A. Queiruga**[1], **Pablo J. Iuliano**[1], **Jorge H. Rosso**[1], **Isabel Kimura**[1], **Matías Brown Barnetche**[1]
[1]LINTI, Computer Science School, Universidad Nacional de La Plata, La Plata, Buenos Aires, Argentina

Abstract - the utilization and development of innovative technologies are suitable elements to take into account in Computer Science degrees. Modern technologies must be incorporated to portray the fundamental concepts and, in this way, to familiarize students with their future workplaces. The development of mashups for mobile devices turns out to be an appealing topic of great interest for students that attend technology and computer science courses. It represents a challenge when it comes to develop software for platforms with peculiarities such as screen and hardware restrictions, and with new advantageous features like sensors of movement, GPS and camera. In this article, it is described the experience in the development of mobile mashups applications written in JAVA in its two platforms: JME and Android. This work is the result of an articulation between a research project on a mobile application testbed and a Computer Science School subject in which the students obtain new skills in the development of software.

Keywords: mashups, Android, JME, education in computer science

1 Introduction

A mashup application uses and combines data, presentations and functions from more than one source in order to create new services. The term mashup implies easy and fast integration, often using open APIs and data sources to produce enriched results different from the application's original purpose. This kind of application is predominantly user oriented, as it focuses mainly on user-friendly features, simplicity and usability [1]. Likewise, it is more oriented towards integration with existing applications, than to the software developing process [2].

In the software development process, integration plays a key role. Aside from functional requirements, developers must take into account requirements related to communication and services their tool could offer. Facebook is one of the most popular social networks, allowing users to share photos, personal information and preferences. It offers an open and easily accessible API which allows for the creation of integrated applications and mashup applications that consume the services Facebook provides. Google Maps is another example of a system that offers the possibility of creating enriched applications with maps through an extensive API that is adapted to mobile devices, as well as desktop and web applications.

Today, mobile technology makes it very easy to collect and share information about ourselves and our environment. This, together with uninterrupted Internet access, the advent of the semantic web, 2.0 web and service-oriented web, has resulted in a reformulation of the way in which we communicate, interact and socialize.

This paper describes our experience in the development of mobile mashup applications written in Java, to address local issues. This work is the result of an articulation between a research project on a mobile application testbed and a Computer Science School subject, in which the students acquire new skills in software development, by taking into consideration matters such as user-oriented design, mobility and integration.

2 Motivation

Teaching a new mobile software development paradigm based on open APIs and third-party service integration is a great challenge to Computer Science teachers. The topic is motivating and state-of-the-art. It poses interesting challenges, such as entirely different development and execution platforms, and requires mobile device emulation tools. It is fundamental for the developer to be familiar with the specific characteristics of each device (operating system, type of keyboard, camera, GPS, Internet access). Manufacturers releases SDKs (Software Development Kits) for developers to use.

There is a widespread assumption that developing mobile applications is equivalent to developing traditional, but "smaller", applications. The real challenge posed by taking a project to a mobile device is providing a valuable experience in a generally small interface that allows an entirely different kind of interaction from desktop applications.

When developing mobile applications, it is necessary to establish whether they will be aimed at low-end, mid-range or high-end devices, and what operating system they will feature, since operating capacity, display size and a number of other features play important roles in the design.

Important changes in today's society force us to think of new ways to "reach out" to users. Web access is increasingly aimed at obtaining information about recent events (online news, Craigslist), expressing personal opinions (Twitter and blogs), gaining access to media files (YouTube and Flickr) and creating communities and social networks (MySpace, Facebook, LinkedIn) [3]. Although all these activities require a computer with Internet access, they are being increasingly accessed through mobile devices, which are always on. Worldwide, the ubiquity of these devices makes them the main, and sometimes only, interaction with computers.

Integration is now quite common in software development, allowing projects to generate or gather data from open services such as Facebook, Twitter, YouTube, Vimeo, Flickr, Google Maps, Google Search, Google Analytics and others. Mashups allow for fast data and process integration. If we also integrate the services in a mobile application, we can obtain applications that "go everywhere" with the user, accessible to everyone and as rich as those found on the web. A combination of the information obtained from different sources with the information obtained from the context through sensors and other components present in current mobile device generations provide a richer basis for the development of mashup applications.

3 Why Teach Mashups?

The relevance acquired in the last few years by mashups and other web 2.0 technologies, together with the fact that our students are digital natives and users of social networks such as Facebook, MySpace, YouTube, and Twitter make in necessary to modify the syllabus of computer science degrees to incorporate these innovations [4]. Likewise, the popularization of computing solutions that collaborate with each other, which is what makes mashups possible, allows teachers to orient tasks towards increasing interaction with established web technologies such as standards related to the development of applications that are highly dependent to others in production or legacy (maximum cohesion) and that require applications to be independent from each other (minimum coupling). In the framework of web technologies used to provide integration and minimum coupling, web services and XML are the most widespread. Thus, our students develop mashup applications while incorporating useful techniques for managing integration between new and legacy applications that must be autonomous. At the same time, using XML as a communication protocol between services and the mashup, allows our students to acquire problem-solving mechanisms for the establishment of a "common language" between or among applications for their interaction.

Teachers also consider that the development of collaborative applications is entirely beneficial, if collaborative development is understood as applications that interrelate and benefit from the services they provide to each other. When students code applications that use data gathered from other applications, they see the resulting added value to the information. This has a positive impact in their motivation, one of the students participating in the *mAvatar4Moodle* project proposed and implemented additional features to share avatars through bluetooth and email.

All the preceding shows the pedagogical benefits of teaching mashups, mainly as regards motivation, as mashups extends learning beyond the limits of each kind of development, and in some cases, even the limits of computer science, as some applications have been taken to the social sciences.

4 Architectures for Mashups

There are many software architectures that can be used in the construction of mashup applications: server-based [5], client-based [6], and mobile [7].

Although we could divide a mashup application into several software layers for a detailed analysis, we generally find three main layers that allow us to understand the choice of architecture among those presented earlier.

Content provider or data source layer: data are available by means of an API and web protocols such as RSS, REST and Web Service.

Mashup layer: where the new service is constructed, nurturing from different sources to generate new information (which it does not own) and resulting in added value for the user.

Mashup presentation layer : mashup user interface.

Figure 1 shows the server-based architecture: the mashup layer is in the server and the presentation layer, in the client.

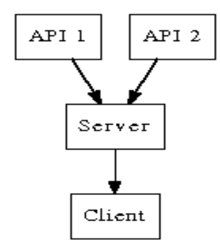


Figure 1. Server-based architecture

Figure 2 shows a client-based architecture: both the mashup and the presentation layers are in the client, which is an application in a browser.

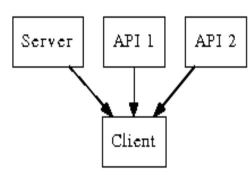


Figure 2. Client-based architecture

Figure 3 shows a mobile architecture: both the mashup and the presentation layers are in the client, which is a native mobile application.

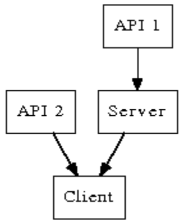


Figure 3. Mobile architecture

Native mobile applications offer maximum experience for mobile device users, as they allow for integration on a deeper level than a web browser. For a native mobile application, users have direct access to device features such as the GPS, accelerometer, camera and storage, and offers innovative interfaces based on gestures, directions, voice, etc. On the other hand, the development of native mobile applications available in a wide range of devices currently requires the development of as many applications as there are mobile operating systems.

5 Cases

In the Software Laboratory course, which is included in the fourth year of the B.S. in Computer Science and the B.S. in Systems at the Computer Science School (UNLP), students were guided by their teachers through development of native mashup applications for smartphones using mainly Java-based mobile device technologies, with Java Microedition (JME) and Android platforms. A mobile web widget was developed using HTML5 technologies and JavaScript on the Opera Widgets platform.

The projects presented are articulated with a School project related to mobile application development. For application development and testing, several mobile devices were acquired by the School. The devices used for this purpose were:

- HTC Touch HD™ T8282, with Windows Mobile® 6.1 Professional OS.

- Nokia N900 with Maemo 5 and Nitdroid (Nitroid is a free software project consisting in taking Android to Nokia Internet Tablet N900)) double booting.

- HTC Nexus One withAndroid 2.2 and 2.3.

The tested mashup applications were aimed at a mid-range/high-end device with touchscreen features.

JME-based projects were tested on the HTC Touch HD™ device with Windows Mobile 6. Because Windows Mobile does not natively support JME applications, it was necessary to install several emulators to provide a controlled environment for application execution. The HTC Touch HD™ device has a JME emulator, JBlend; however, it was impossible to emulate developed applications on JBlend. The teachers thus tested a series of emulators, which included IBM, PhoMe and JBed. All the applications were successfully emulated using these products.

Android-based projects were tested on the Nokia N900 device with double-booting and in the HTC Nexus One devices by Google, with Android 2.2 and 2.3. These devices allowed for native application installation, and there were great advantages in terms of performance and usability.

The experience in development and setup of JME applications raised some issues in terms of lack of available memory for the emulator to dynamically render maps that had to roll because of user interaction. To solve this problem, it was necessary to optimize resource allocation, which made it necessary to lose the portions of the map that were not rendered. This decision goes against efficiency, given that it makes it impossible to have a memory cache. Emulators restricted permissions as well.

Another topic relevant to mobile application development with JME are limitations in terms of user graphic interfaces. The user interface controls provided by the platform are basically text-oriented (text fields, buttons, etc) and not extensible or customizable.

With the goal of promoting collaborative work and giving the students a taste of a professional development environment, the teachers have created a wiki [http://wiki.labmovil.linti.unlp.edu.ar] that registers the experiences in the development of mobile applications. In this shared repository, the developed applications are presented by their authors, who share problems faced during development, noteworthy features, and the architecture and tools used. Screenshots and executable files are also shared, so that visitors can use and test the applications.

5.1 Mashup based on GoogleMaps

GoogleMaps is a free web service that provides interactive maps, where users can make annotations and mark locations.

Mashup applications based on maps developed in the framework of this project were aimed at the JME platform an allow displaying, consulting and adding information on a set of shared restaurant guides, which hold information on related places. These guides are shown by GoogleMaps in a map with markers in the location of the restaurants. Information can be stored about each location.

Many mobile device emulators were evaluated (JBed, PhoneMe and JBlend) to execute these applications, out of which only IBM gave the expected results.

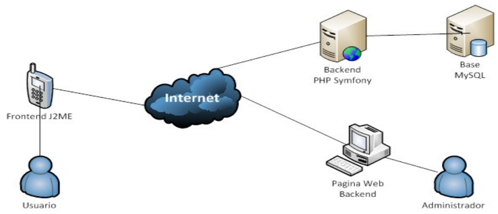


Figure 4. Architecture of a mashup based on GoogleMaps

The designed of the developed applications required a special user interface control to display the map in the screen of the device and be able to roll it in the four cardinal points (North, East, South and West), and read the information on a particular restaurant, among other features. As mentioned before, JME has very little capacity to build user interfaces, which made it necessary to develop our own user interface canvas with the required features.

The developed GoogleMaps-based mashup applications respond to a mobile architecture. As shown in Figure 4, a service was created in the web to register the information entered from the mobile application. The service implements an interface that is similar to REST to access and modify the information resources. Data are communicated through JSON.

5.2 Mashup based on Flickr

Flickr is an image storage website that allows users to organize their collection of images and share them. Using their API, content can be shared across websites, creating mashup applications.

The applications developed in this project allow managing Flickr user photostreams, creating and modifying galleries, uploading and deleting photos, modifying the information related to each photo and searching the entire site. In this project, some students chose to implement with JME and others with Android.

Flickr does not provide an API developed exclusively for mashups, so that data are obtained and processed manually.

The developed Flickr applications have a particular feature that can only be used through a mobile device, which is taking a picture with a cellphone camera and uploading it to a Flickr account from the application itself. In these development, it is worth noting the advantage of using native mobile applications with respect to desktop or browser-based applications.

Projects done on JME had some memory problems owing to the need to execute on an emulator and not natively in the device. The service that Flickr provides has authentication tokens for applications and users. Out of all the emulators evaluated, only PhoneMe allowed these tokens successfully.

Projects developed on Android had no functional problems due to native support. Likewise, very friendly and easy to program user interface were achieved.

The architecture of these applications is mobile, with a native mobile application as mashup and presentation layers, from which pictures can be taken and interaction achieved with modifications that are communicated directly to the Flickr service.

5.3 Mashups based on Facebook

Facebook is a currently booming social network, where people can share interests, tastes and images, and stay connected with friends all over the world. One way to meet a Facebook user is through their profile picture.

AvatarFacedGet is an application for creating and sharing avatars through Facebook, or storing them in a mobile device. It provides thousands of combinations out of a limited set of elements.

AvatarFacedGet is a widget developed for the Opera Widget platform [8] with standard web technologies such as HTML5, CSS3 and JavaScript, executed in the widget mobile manager, with multiplatform and multidevice features (from desktop to mobile to TV applications). This widget can be executed in a number of operating systems in mobile devices that include the widget mobile manager.

Facebook offers an API for developers that is very well documented, allowing to create applications inside Facebook or communicate with it through REST.

As is the case with Twitter and Flickr, Facebook uses application authentication keys. They can be obtained by filling a form in Facebook where the characteristics of the developed application are to be specified.

The architecture of AvatarFacedGet corresponds to a mobile mashup architecture, since the mobile client can only create avatars and the Facebook service is in charge of publishing albums and profile image.

5.4 Mashups based on Moodle

Moodle is an open source Course Management System (CMS), known also as a Learning Management System, (LMS) or as a Virtual Learning Environment (VLE). It is very popular amongst educators around the world as a tool for the creation of dynamic online websites aimed at students [9].

The Computer Science School uses a Moodle-based platform for its courses as a complement of in-class activities and for communication with students, located at http://catedras.info.unlp.edu.ar. The students of the School have a user in the platform that is associated with each course they are taking.

mAvatar4Moodle is an application aimed at mobile devices and used to build avatars, dress them up, give them expression and accessories, build avatar galleries and publish them as user profiles in the platform of Computer Science courses.

Students in the Software Laboratory subject have developed different versions of *mAvatar4Moodle* for the Android platform, and each of these experiences posed a challenge. One of these challenges was communicating with the Moodle service of the School, which implied managing SSL certificates through the application. For this purpose, it was necessary to add the certificate to the application for Moodle to trust its authenticity. It was also necessary to cover the need to use multipart forms to send images through the URL. Another challenge is related to avatar editor user interface programming, so that characters could be built on the basis of images representing their parts. Android offers a very comprehensive API for the development of user interfaces that facilitated the completion of this task.

The developed *mAvatar4Moodle* applications communicate directly with the Moodle system using the device's network layer. For this, a SSL certificate was added and a direct connection with the system was implemented. Still, the mobile architecture remains dominant, since avatars are published on user profiles through the Moodle server.

6 Conclusions

The use and development of innovative technologies cosnstitute appropriate elements to be incorporated in computer science degrees. However, the main goal in university-level computing education is to acquire concepts with lasting relevance. Modern technologies should be incorporated to illustrate these fundamental concepts and place the students in a working environment that closely resembles the real one.

Mashup development for mobile devices is a motivating, innovative topic that students are very interested in. It is a challenge when developing applications for platforms with such characteristics as display and hardware limitations and new features such as movement sensors, temperature sensors, GPS and camera, among others. This poses new ways of thinking of applications.

From the experience, it is worth noting that students who chose JME as their development platform faced greater difficulties, from adapting resource allocation to limitations in the construction of friendly and custom user interfaces. Students who opted for Android as their development platform encountered simpler solutions with a consistent interaction throughout the device and features that allowed them to easily extend the functionality of the device incorporating sensors.

7 References

[1] Agnes Koschmider, et. all, "Elucidating the Mashup Hype: Definition, Challenges, Methodical Guide and Tools for Mashups", Proceeding of 2nd Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web (MEM 2009), April 2009.

[2] M. Xenos, D. Stavrinoudis and D. Christodoulakis. "The correlation between developer-oriented and user-oriented software quality measurements (a case study)". Proceedings of the 5th European Conference on Software Quality, Sponsored by European Organization for Quality − Software Committee (EOQ-SC), Dublin Ireland, pp. 267-275, 1996 .

[3] E. Michael Maximilien, "Mobile Mashups: Thoughts, Directions, and Challenges", 2008 IEEE International Conference on Semantic Computing, pp.597-600.

[4] A. Easton and G. Easton. "Demystifying Mashups". Proceedings of Informing Science & IT Education Conference (InSITE) 2010, pp. 479-486.

[5] E. Ort, S. Brydon, and M. Basler. (2007, May) Mashup styles, part 1: Server-side mashups. [Online]. Available: http://java.sun.com/developer/technicalArticles/J2EE/mashup1/index..html

[6] E. Ort, S. Brydon, and M. Basler.(2007, August) Mashup styles, part 2: Client-side mashups. [Online]. Available: http://java.sun.com/developer/technicalArticles/J2EE/mashup_2/index.html

[7] A. Brodt and D. Nicklas, “The telar mobile mashup platform for nokia internet tablet”. Proceedings of the 11th international conference on Extending Database Technology (EDBT 08), ACM, 2008, pp. 700–704.

[8] Opera Widget SDK: http://dev.opera.com/sdk/

Role of Game Programmers for Serious Games in Academia: Colleague, Collaborator AND Client

Kris Stewart[1], **Mark Thompson, Jr.**[1], **Abhishek Sood**[1], **Sathyanarayan Chandrashekar**[1], **Megha Shaseendran**[1], **Mark Siprut**[2], **Khaled Morsi**[3], **Marilee Bresciani**[4]
[1]Computer Science Department, College of Sciences;
[2]School of Art Design and Art History, College of Professional Studies and Fine Arts;
[3]Mechanical Engineering Department, College of Engineering;
[4]Administration, Rehabilitation, and Postsecondary Education Interwork Institute, College of Education; San Diego State University

Abstract *- This paper presents a unique collaboration among faculty and students from four distinct disciplines united to create a game-based simulation of a Materials Science laboratory experience for non-Mechanical Engineering students in a lecture-only course. The benefits to each of the development disciplines is unique, and has been sustained over a lengthy development cycle. Student feedback shows the value of the software tool to engineering students, in addition to the highly enriching experience gained by multi-disciplinary student engagement in its development.*

Keywords: Materials Science, Interactive, Simulated Laboratory

1 Introduction

A typical situation at institutions of higher learning is that faculty in different colleges (or departments) are unlikely to find common ground to collaborate. But the presumed **Colleague**-role will easily allow a faculty with a need (computer enhanced education in Mechanical Engineering) to reach out to another. In our present case a large portion of undergraduate students in the College of Engineering were each required to take a lecture course in Materials Science, however, only students in the Mechanical Engineering (ME) major received the hands-on experience of working with the impact testing and tensile testing machines in the Engineering Materials Lab. A team of investigators with multi-disciplinary backgrounds covering materials engineering, 3D Game Programming [1], digital arts & design and educational assessment assembled to provide a solution for students who are deprived of hands-on experiences in materials science. The present paper is a brief presentation of the activities behind the development of the Materials-ISLE (Materials Interactive Simulated Laboratory Experience) [2] software, with a focus on the role of programmers as colleague, collaborator and client.

2 Serving the different disciplines

The goal of this project was to create an interactive computer simulation of the Materials Science laboratory experience that students in the lecture-only course could use to gain some level of insight to the field. The software provides students with experimental experiences not achievable through textbooks or regular simulations. As such, within the developed simulations, experimental roadblocks were intentionally placed for students to overcome. In addition, a direct benefit was to provide students enrolled in the 3D Game programming course (CS 583) the experience of working outside their discipline with a Client from mechanical engineering as well as develop and produce a product with art and design students. In general most CS programmers do not have the training in art to allow them to effectively create the 3D objects that would be needed to simulate the actual engineering laboratory. Therefore a realistic virtual lab with furniture, machines and relevant objects was produced by digital arts experts and a systematic assessment plan implemented by learning assessment experts within the software game.

2.1 Computer Science game programming

The game-based simulation code was written in Microsoft XNA [3] based on the software ability to simulate the engineering lab in 3D, strong education support from the online community for developers, advantageous licensing fees (free) for non-commercial use and this was the software currently used in the CS 583 course.

The NSF OCI "EPIC - Engaging People in Cyberinfrastructure" [4], using the Torque Game Engine [5] led to establishing the SDSU course in 2006. Further studies showed how game programming can be used as service-learning for computer science students [6]. Channeling CS programming student interest towards the Serious

Games [7] field adds depth to the student programming experience. Over time much additional evidence tracks the growing use of game technology in a variety of applications [8]. There are applications in military training [9] and NASA (National Aeronautics and Space Administration) [10] as well as Science and Engineering Visualization [11].

2.2 Art Design

The artists' first choice of development tool was Maya [12] from Autodesk, due to its capability and availability in the Art Labs through cite-license with SDSU. This software is covered in various arts courses taught at SDSU which would allow the participation of arts students after taking their courses.

3 Evolution of the student team

Within the game programming side of development, Stewart was able to engage a sequence of students, after completing the upper-division CS elective class CS 583, to join our team. The first student was Mark Thompson, Jr., in January 2009. Mark worked closely with the students from ME who used Pro/Engineer [13] to create fine-grained prototypes of the devices in the Materials Science laboratory. Mark was key to the project in finding the way to smooth the process of taking prototypes from Pro/Engineer, which were refined by the art lab students using Maya and eventually recast using 3D Studio Max, software which is now also available from Autodesk [14]. The compromise between the realistic detail from the engineering CAD models, then made believable as 3D models on the screen by artists using Maya, then simplified to allow effective representation within the 3D world of the game environment without losing any needed detail for an efficient game was essential to the development and is a continual tradeoff.

Our second game programmer, Abhishek Sood, took on the task of providing much of the text-manipulation needed in the lab to explain the requirements of the lab and the materials science content being explored as well as capture the results of students using the lab. As a preliminary presentation on campus, both Thompson and Sood participated in the SDSU Student Research Symposium, 2010, with their poster [15] created by Claudia Faulk in the Art Lab. Abhishek Sood defended his Masters Thesis [16] at SDSU Spring 2012. A co-game-programmer with Sood, Sathyanarayan Chandrashekar, also successfully defended his CS Masters thesis in Fall 2011, on a topic distinct from Materials-ISLE, "3D Visualization of Conic Sections in XNA Game Programming Framework" [17]. Our current game programmer, Megha Shaseendran, has entered the Microsoft Imagine Cup with her programming of the interactive, virtual 3D Lattice Voyage [18].

4 Assessment

Continuous assessment of student learning outcomes (direct and indirect) was conducted over the period of the development of the software. We have shared the design of the assessment plan [19] for the learning outcomes of the students from engineering who have used the software game. Once students in this spring 2012 semester complete their coursework, and provide additional assessment data, we anticipate sharing further results.

5 Our programming outcome

Our game programming students and art students have worked side by side through the development process for several years now and though the individuals have changed, the project is now demonstrating believable detail and effective game-play. The development students have also had the opportunity, in our weekly team meetings, to observe some discipline-specific approaches expressed by the faculty. Talking with the game programmers afterwards, they have uniformly told Stewart that the experience was unique and useful. Programming to deadlines is always challenging. Being the Client for the artists who were charged with creating the 3D models was illuminating. The game programmers have been client, collaborator and colleague with an outcome of three separate Material Sciences Labs that allow 3D simulation of the three labs, pictured below in Figures 1-6. The engineering students have expressed they value the interactive labs and further formal assessment results are expected soon.

We have a first person point of view game that runs on Windows platforms that provides an interactive, non-linear and dynamic education experience, as recommended by Jackson [8]. Students take initial quizzes testing their understanding of lab safety before they enter the virtual materials science laboratory. There are additional quizzes, tailored for the individual labs, that allow students to demonstrate their mastering the actual content of the materials science laboratory. The large class of engineering students, roughly 100 to 140 per semester (due to recent spike in student enrollment), have their results captured within the XNA game and automatically sent to a MySQL database running on another campus computer. The several quiz results are available to the engineering faculty to include in the scoring for students in the lecture-only class as well as the assessment faculty to monitor student learning outcomes. Anticipating a need to easily modify quiz questions from class to class, the NEEV engine [16] provides an interface to modify text and summarize results of actual game use, such as time to complete and number of attempts of quiz questions. The NEEV engine also ensures student logins and results are maintain through a secure registrations of students.

Figures 1-6 show images from Materials-ISLE game play. Detailed presentations and discussions of the individual labs will be the subject of separate publications. Online resources such as Multimedia Educational Resource for Learning and Online Teaching (MERLOT) [20] will also be targeted as dissemination routes for Materials-ISLE.

Tensile-Testing Lab:

Figure 1 The Samples on lab table with Task List for this lab

Figure 2 The Virtual Tensile Testing Machine

Impact Testing Lab:

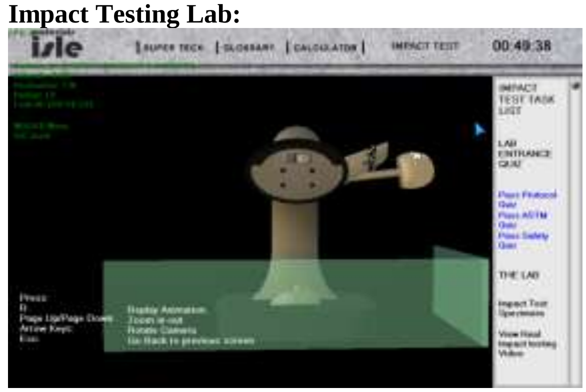


Figure 3 Safety screen to protect users during impact testing

Figure 4 Actual high-speed video of Impact Test Machine, included in Materials-ISLE

Lattice Voyage:

Figure 5 Virtual image of teleporter in the virtual lab environment

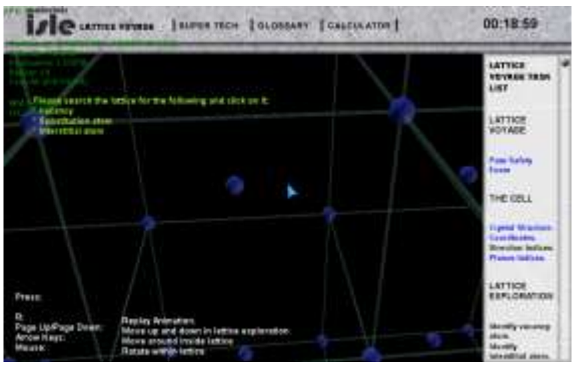


Figure 6 Student will search 3D space to identify a vacancy, a substiubstitution atom and an interstitial atom

6 Acknowledgement

We wish to acknowledge the artists who have contributed to this project: Christina Bertrang, Claudia Faulk, Julius Santos, Jessica Knight and Christine Creber; along with the assessment researchers Dr. Mark Tucker and Allison Duncan and the mechanical engineering students Kunal Aher, Jimmy Patel and Abinash Numula. This work was supported by National Science Foundation Division of Undergraduate Education, #0837162.

7 References

[1] 2011-2012 SDSU General Catalog, http://arweb.sdsu.edu/es/catalog/quickref.html
Please search for CS 583 3D Game Programming

[2] Web Page of artifacts from our Game Development project can be viewed at http://www-rohan.sdsu.edu/~stewart/MatsISLE

[3] Microsoft XNA 3.1 and Visual Studio 2008 software, available free to academia for non-commercial development of Windows, Windows Phone and XBOX 360 games http://create.msdn.com/en-US/education/catalog/

[4] Engaging People in CyberInfrastructure (EPIC) NSF Grant to pursue 3d Game Engines as a curriculum tool, Date: 15April2005. NSF OCI 0520146, subcontract to Boston University and then to SDSU. http://www.nsf.gov/awardsearch/showAward.do?AwardNumber=0520146

[5] The Torque Game Engine, GarageGames, Eugene OR http://www.garagegames.com/education/3d

[6] Stewart, Kris, 3d Game Programming as Service-Learning for CS Students, J. Comput. Small Coll. 24, 4 (April 2009), 246-251.

[7] Harz, Chris, GDC 2009: Serious Games Opportunities, Game Developers Conference Serious Games Summit, http://www.awn.com/blogs/animated-travels/gdc-2009-serious-games-opportunities

[8] Jackson, Melinda, Making visible: Using simulation and game environments across disciplines, contained in Beyond Fun: Serious Game and Media, Davidson, Drew et al, 2008, ETC Press, available as free download http://www.etc.cmu.edu/etcpress/content/beyond-fun-serious-games-and-media

[9] Macedonia, Michael, Games, Simulation, and the Military Education Dilemna, http://www.educause.edu/ir/library/pdf/ffpiu018.pdf

[10] NASA Ames Simulation Laboratories, http://www.simlabs.arc.nasa.gov/

[11] Science Magazine's Science and Engineering Visualization Challenge http://www.sciencemag.org/feature/data/vis2003/ssintro.html

[12] Autodesk Maya, http://usa.autodesk.com/maya/

[13] Pro/Engineer, now Creo Parametric for 3D CAD http://nxrev.com/

[14] Autodesk 3ds Max, http://usa.autodesk.com/3ds-max/

[15] Thompson, Mark A., and Sood, Abhisheek, Utilizing 3D Video Game Technology for an Immersive Laboratory Experience, SDSU Student Research Symposium, 2010 http://www-rohan.sdsu.edu/~stewart/MatsISLE/SRS2010_posterWideBClaudiaFaulk.pdf

[16] Sood, Abhishek, NEEVENGINE: Using XNA Game Studio for Serious Game Design and Development, Computer Science Masters Thesis, San Diego State University, 2012.

[17] Chandrashekar, Sathyanarayan, 3D Visualization of Conic Sections in XNA Game Programming Framework, Computer Science Masters Thesis, San Diego State University, 2011.

[18] Imagine Cup http://www.imaginecup.com/CompetitionsContent/GetStarted.aspx

[19] Exploring the Challenges in Designing and Implementing an Assessment Plan for a Virtual Engineering Materials Lab, Marilee J. Bresciani, Khaled Morsi, Allison Duncan, Mark Tucker, Mark Siprut, Kris Stewart, Eludamos. Journal for Computer Game Culture [EJCGC]. 2010; 4 (2), p. 277-285 http://www.eludamos.org

[20] http://www.merlot.org/merlot/index.htm MERLOT is a free and open online community of resources designed primarily for faculty, staff and students of higher education from around the world to share their learning materials and pedagogy.

A Heuristic Algorithm for Solving the Faculty Assignment Problem

Manar I. Hosny[1]
[1]College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia
mifawzi@ksu.edu.sa

Abstract— *In this paper we describe a heuristic-based algorithm for the faculty assignment problem. The algorithm was designed and implemented as part of a graduation project in the IT department of King Saud University, Riyadh, Saudi Arabia. The purpose of the project was to automate the manual tedious process of assigning TAs (Teaching Assistants) to suitable lab sessions; a task that needs to be done every semester in the IT department. The algorithm was designed to fulfill the assignment task while taking into consideration a number of constraints pertaining to both the TAs (e.g. personal preference and available hours) and the IT department (e.g. maximum allowed workload). The problem definition, the algorithm, and the experimental results are thoroughly explained in this paper. The computational experimentation indicate the efficiency of the proposed algorithm in assigning courses that conform to the preferences of the TAs and the rules set forth by the IT department.*

Keywords: The Faculty Assignment Problem, Scheduling, Timetabling, Heuristics, Meta-heuristics

1. Introduction

The faculty assignment problem is one type of scheduling problems that almost every educational institution needs to handle periodically. The problem is concerned with assigning courses to staff members, while a number of constraints are adhered with. For example, the assignment should take into consideration the preferences of staff members, their availability, and their permitted workload. Many institutions still perform this task manually, which often requires a lot of time and effort. In addition, the resulting staff schedules are usually inefficient in terms of not satisfying the preferences of all staff members, or the uneven distribution of workload among them. Automating the planning and scheduling tasks has great benefits in reducing the workload of the planner. In addition, creating balanced and fair schedules help to satisfy faculty members and reduce the stress among them, which may encourage them to stay in the academic career.

The faculty assignment problem is a special case of a wider class of scheduling problems that take many shapes and forms. For example, there are exams scheduling, courses scheduling, nurse rostering, scheduling of sports and business events, etc [1]. Scheduling problems belong to the class of NP-hard problems [2] for which there is no polynomial time algorithm that exists to solve the problem to optimality. In addition, the presence of a number of constraints makes the problem even more difficult to solve. In general constraints are of two types: *hard constraints* and *soft constraints* [3] [4]. Hard constraints are those that must be strictly enforced in the solution, otherwise the solution will be infeasible. For instance, the condition that a faculty member cannot teach two sessions at the same time is a hard constraint. Some hard constraints may be specific to the institution. For example, one institution may probably require that a staff member cannot teach more than three sections a day. Soft constraints, on the other hand, are desirable but they do not make the solution infeasible. For example, giving staff members courses within their wish list is a soft constraint. Given the difficulty of solving this problem, due to the often conflicting constraints, heuristic and meta-heuristic algorithms are usually used to provide a good problem solution in a reasonable amount of time.

In this research we consider the faculty assignment problem that is specific to the Information Technology (IT) department, of the College of Computer and Information Sciences in King Saud University (KSU), Riyadh, Saudi Arabia. We developed a heuristic algorithm to solve this problem based on the constraints specified by the IT department. The algorithm was integrated with a web-based tool having a centralize database. The system as a whole was a 2-semester graduation project of five students in the IT department. The preliminary system analysis and design phases of the project, prior to the implementation phase, were described in a virtual brief paper [5]. In this paper, we mainly focus on the heuristic algorithm, which is the core of the implemented tool, describing and analyzing its results and performance after the end of the implementation phase of this project.

The rest of this paper is organized as follows: Section 2 is a brief description of some related work. Section 3 explains the problem that we are trying to handle in this research, by describing the context in which it is applied and the problem constraints that should be adhered with. Section 4 describes in detail the heuristic algorithm used in solving the underlying problem. Section 5 presents and discusses the experimental results of the algorithm. Finally, Section 6 provides some concluding remarks and some thoughts for future directions.

2. Related Work

Solutions to the faculty assignment problem are based on the specific problem under consideration. Every institution has certain constraints that should be fulfilled when assigning faculty members to courses. Thus, the quality of the solution is very much subjective: a good assignment for one institution may be a poor one relative to another institution. Accordingly, techniques for solving this problem were mostly developed to fit the needs of specific institutions, under clearly pre-determined conditions.

[6] presents a preference-based decision making approach for the faculty assignment problem, which takes into consideration faculty preferences, students preferences and adminstration preferences regarding a particular subject. A decision making tool called "Analytic Hierarchy Process" (AHP) has been proposed in this paper. The AHP technique takes into consideration both subjective and objective factors in the decision making process. The methodology proceeds in several steps and depends on assigning weights to: (1) the preferences of faculty members over each subject; (2) a pairwise comparison by students of the faculty available to each subject; (3) a pairwise comparison between the faculty in terms of their research work and other activities (i.e., faculty having more research work are given less workload); and (4) a management's assigned weights to each of the above criteria. A decision matrix is formed based on the preferences values obtained, and a table of faculty members vs. subjects is formed. Using this table, faculty members who have higher weights are given priority in assignment, unless their workload is exceeded.

[7] considers the problem of faculty-course-time slot assignment in a single stage. The model includes the following constraints: (1) each course must be assigned to only one instructor; (2) there is a certain weekly limit for the instructor's load; (3) there is a limited number of courses that can be given during the same time slot; and (4) there is a limited number of same-time slots that can be assigned to each instructor, since some time slots may be more preferred than others by all instructors. The objective function is to maximize the preferences of the instructors for both courses and time slots, and also maximize the preferences of the administration regarding instructor-course assignment, and course-time slot assignments. A multi-objective linear 0-1 model is developed for solving this problem that depends on scalarization of the given problem (i.e., combining different objectives into a single objective). The Analytic Hierarchy Process (AHP) and Analytic Network Process (ANP) are used to assign weights to the different and conflicting objectives. After this, the scalarized problem was solved to optimality using the GAMS/CPLEX solver.

A two-stage heuristic approach for the class-course-faculty assigning problem is described in [8]. The problem was formulated to fit the needs of the Electronic Engineering Department of Kao Yuan University, Taiwan. The model considers teachers' specialities and preferences on specific teaching periods. Also, students' need for taking certain courses, in order not to delay graduation, is taken into consideration. Other organizational constraints, such as available facilities and classroom resources are also considered in this model. The optimization process is divided into two stages: class-course scheduling and class-faculty assignment. Two fitness functions are designed to evaluate the satisfaction of the teachers and the students. The heuristic approach is based on iterative mutation, which either changes the time slot assigned to a certain course, or changes the teacher assigned to a certain course.

In the following sections, we explain in detail the research conducted in the project under consideration.

3. Problem Definition

As previously mentioned, the problem handled in this research was specifically defined according to the needs of the IT department at KSU. Specifically, we were concerned with assigning TAs (Teaching Assistants) to suitable lab sessions, taking into consideration a number of constraints to be explained shortly. In the IT department, there are approximately 12 courses which are associated with lab sessions. Each course has a number of sections (usually ranging from 1-6), and each section has its own lab session, whose time and date has been pre-determined by the department. Weekdays are from Saturday to Wednesday. Each lab sessions runs for 2 hours, and there are 3 possible lab time slots per day: 8-10, 10-12, and 1-3. Due the often large number of students enrolled, each lab session is assigned to two TAs, who should both attend the lab session concurrently. The TAs in the department are of two types:

1) **Full-Time TAs:** whose maximum permitted workload is 16 hours per week.
2) **Masters Students TAs:** whose maximum permitted workload is 8 hours per week. In addition, those TAs have to attend their Masters lectures, i.e., they have a limited availability that should be considered when lab sessions are assigned to them.

The TA Assignment Committee (TAAC) of the IT department is responsible for assigning suitable TAs to all lab sessions. To do this, the TAAC first asks all TAs in the department to fill in a "wish list", in which they indicate three courses that they like to teach in order of preference. In addition, Masters students TAs should indicate their busy hours (i.e. Masters lectures), during which they cannot teach lab session.

We can define the constraints of the above problem as follows:

- **Hard Constraints:**
    1) To assign a lab session to a certin TA, the lab's date and time must not conflict with the TA's availability, i.e., she should not be busy during the time slot of the lab (e.g. taking a Masters lecture or teaching another lab).

2) Each lab session should be assigned a maximum of two TAs.
3) The maximum permitted workload of the TA cannot be exceeded.

- S**oft Constraints:**
  1) The preferred courses for the TA should be taken into consideration, in order.
  2) Each lab session should be assigned to at least one TA.
  3) The distribution of workload should be as fair as possible.

The algorithm developed in this research will help facilitate the task of the manual generation of schedules for the TAAC in the IT department. The algorithm will generate schedules for the TAs automatically, based on the information and the constraints provided by both the TAs and the TAAC. The steps of the algorithm are described in detail in the following section.

4. The Faculty Assignment Algorithm

The faculty assignment heuristic algorithm will try to assign courses[1] that conform to the preferences of the TAs, their available times, and their permitted workloads. The assignment process will be as fair as possible in terms of the distribution of the workload. Specifically, older Masters students (based on the year they are currently enrolled in) are given higher priority in the assignment of their preferred courses. Second priority is given to Masters students with higher GPA. For the full-time TAs, the older ones (according to their employment date) are given priority in assignment. However, during the processing of the algorithm, priorities of assignment change adaptively, depending on the currently assigned working hours of each TA. This is intended to increase the number of assigned working hours for under-utilized TAs.

The input to the algorithm will be information related to a number of TAs and a number of lab sessions:

- For each TA: a list of 3 preferred courses, available working hours, maximum allowed working hours, and the number of assigned working hours (initially 0).
- For each course: course name, section number, day, time, number of assigned teachers (maximum 2, initially 0), and the list of assigned teachers (initially empty).

The output of the algorithm will be the automatically generated labs schedule after the assignment of each lab session to two TAs, if possible, in addition to the total working hours assigned to each TA by the algorithm. If no solution can be found, i.e., the algorithm cannot assign all lab session to suitable TAs, the algorithm will display a message informing the TAAC that some labs were not assigned to two TAs, in which case the schedule must be adjusted manually[2]. Thus the main objective of the algorithm is to maximize the number of assignments of TAs to courses. The objective is fulfilled when each lab section is assigned to two TAs, i.e., the maximum number of assignments = $2 \times NumberOfSections$.

The algorithm will first sort the TAs according to their studentship status, i.e., Masters students are placed at the top of the list to give them priority in assignment. Within the list of Masters students, older students are preferred, followed by high GPA students. Regarding full-time TAs, they are sorted according to employment date, such that older TAs in the department are given priority.

The algorithm then proceeds in two phases:

Phase 1: will iterate 3 times (once for each preference). In each iteration, the algorithm will try to assign to every TA in the sorted list the current preference under consideration, subject to the following:

- Her preferred lab is not already assigned to two TAs;
- The lab day and time does not conflict with her availability;
- And her workload will not be exceeded after assigning the lab to her.

At the end of each iteration (i.e., before progressing to the next preference in order), the sorted list of TAs will be reordered by giving priority to TAs with the least number of assigned working hours. If there are still lab sessions that are not assigned to two TAs each, the second phase will start.

Phase 2: This phase will start by first initializing the best so far schedule to the schedule generated in phase 1. It will then use a Hill Climbing (HC) optimization approach to improve this schedule. First, the list of TAs will be reordered, according to the ratio of assigned working hours to the permitted workload. Thus, priority is given to under-utilized TAs. Then, the algorithm will proceed similar to phase 1, but instead of assigning each TA to her preferred lab session, it will try to assign to each TA a random non-assigned lab session, subject to the same conditions mentioned above. This process will be repeated until one of two conditions are satisfied: (1) all courses have been assigned to two TAs; (2) a maximum number of trials (e.g. 10) has been attempted.

During the HC process, each new schedule generated will be compared with the best so far schedule. If the new schedule is better, in terms of reducing the number of unassigned courses, it will replace the best so far schedule. The final best schedule will be returned as the output of the algorithm. In case there are still lab sessions that could not

[1] We will use the terms courses and labs interchangeably since we only consider assignment of lab sessions associated with courses in this research.

[2] Manual adjustment is allowed through the interface of the web-based tool.

be assigned after the end of the optimization process, the algorithm will display a warning message so that the TAAC will take an appropriate action.

The steps of Phase 1 of the algorithm is shown in Algorithm 1, while the steps of Phase 2 are shown in Algorithm 2.

Algorithm 1 Faculty Assignment Algorithm - Phase 1

create $TAList$, the sorted list of all TAs
{Sort TAs by type: Master students first then full-time TAs}
{Sort Masters students in descending order of their current Master year. Students in the same year are sorted in descending order of their GPA}
{Non-Masters students are sorted by employment date (oldest first)}
$totalAssignments = numSections * 2$
$numAssignments = 0$
$prefIndex = 0$
$currentSchedule = empty$
while ($numAssignments < totalAssignments$) AND ($prefIndex < 3$) **do**
  $prefIndex++$
  for ($curTA \in TAList$) **do**
    $prefLab \leftarrow Labs[prefIndex]$
    if ($prefLab$ not already assigned to two TAs) AND ($prefLab$ time slot does not conflict with $curTA$'s availability) AND ($curTA$'s workload will not be exceeded after assigning $prefLab$ to her) **then**
      Assign $curTA$ to $prefLab$
      $numAssignements++$
      Increment number of TAs assigned to $prefLab$
      Update availability of $curTA$ to reflect the new assignment
      Increase assigned working hours of $curTA$
      Update $currentSchedule$ to reflect the assignment
  Re-sort $TAList$ in ascending order of assigned working hours

Algorithm 2 Faculty Assignment Algorithm - Phase 2

if ($numAssignements < totalAssignments$) **then**
  $numAttempts = 0$
  $MAXATTEMPTS = 10$
  $bestSchedule = currentSchedule$
  repeat
    $numAttempts++$
    create $newTAList$, the new sorted list of all TAs {sort TAs according to the ratio $workingHours/workLoad$, lowest ratio first}
    $Stop = False$ {The loop will stop if no successful assignments can be found for all TAs in the last pass}
    while ($numAssignments < totalAssignments$) AND ($Stop == false$) **do**
      for ($curTA \in newTAList$) **do**
        $randomLab \leftarrow Labs[randomIndex]$
        if ($randomLab$ not already assigned to two TAs) AND ($randomLab$ time slot does not conflict with $curTA$'s availability) AND ($curTA$'s workload will not be exceeded after assigning $randomLab$ to her) **then**
          Assign $curTA$ to $randomLab$
          $numAssignements++$
          Increment number of TAs assigned to $randomLab$
          Update availability of $curTA$ to reflect the new assignment
          Increase assigned working hours of $curTA$
          Update $currentSchedule$ to reflect the assignment
          $Stop = False$ {found a successful assignment this iteration}
        else
          $Stop = true$
    if ($currentSchedule$ is better than $bestSchedule$) **then**
      $bestSchedule = currentSchedule$
  until ($numAttempts == MAXATTEMPTS$) OR ($numAssignements == totalAssignments$)
Return $bestSchedule$
if ($numAssignements < totalAssignments$) **then**
  display warning message

5. Results and Discussion

The algorithm was implemented using the Ruby language on an Intel(R) 1.73 GHz processor, and a 2.00 GB RAM. It was tested on 10 test cases using data obtained from the IT department. In 8 out of the 10 test cases there are 30 lab sections and 15 TAs (10 Masters and 5 full-time). To test the performance of the algorithm in some special circumstances, where there is a fewer number of sections or a fewer number of TAs, test case 9 had only 10 lab sections, and test case 10 had only 10 TAs (6 Masters and 4 full-time). In addition, in test cases 2 and 3, we tried the case where some TAs (4 out of 15) did not choose any preferences.

To measure the performance of the algorithm, we recorded the number of times the algorithm was able to assign to each TA a course among her wish list, in addition to the total number of assigned working hours. We also calculated the total number of assignments of TAs to sections (remember that our goal is to assign each section to two TAs under the given constraints). Table 1 shows a summary of the results of the 10 test cases. The first column indicates the number of the test case. Columns 2-5 show the percentage of preference 1, preference 2, preference 3, and non-preference (preference 0) assignments among all TAs in the corresponding test case[3]. Columns 6-7 show the average assigned working hours for Masters and full-time (regular) TAs respectively. Finally column 8 shows the average processing time of each test case. The final row of the table calculates the overall average of each corresponding column. Test cases 2, 3, 9 and 10, which have some special conditions as explained above, are shown in italic.

Fig. 1 shows the percentage of assignments achieved by the algorithms compared to the total number of assignments required for the 10 test cases. For example in case 1, there

[3]The percentages in columns 2-5 do not add to 100% because each TA is assigned one or more lab sections; i.e., she could be assigned a mixture of preferred and non-preferred labs.

Table 1: Algorithm Results

Case	Pref 1	Pref 2	Pref 3	Pref 0	Avg MSc Hrs	Avg Reg Hrs	Proc Time
Case 1	73%	53%	60%	27%	7.2	8.4	0.76
Case 2	*67%*	*33%*	*47%*	*87%*	*6.6*	*9.2*	*0.98*
Case 3	*60%*	*40%*	*53%*	*73%*	*7*	*9.2*	*0.74*
Case 4	67%	47%	53%	13%	6.6	9.6	0.9
Case 5	67%	53%	33%	47%	6.2	10	0.84
Case 6	73%	33%	47%	20%	6.6	9.6	0.9
Case 7	67%	67%	33%	47%	7.2	8.4	0.67
Case 8	67%	53%	40%	40%	6.6	9.6	0.93
Case 9	*53%*	*40%*	*27%*	*13%*	*2.8*	*2.4*	*0.38*
Case 10	*70%*	*50%*	*40%*	*0%*	*7*	*12.5*	*0.97*
Average	**66%**	**47%**	**43%**	**37%**	**6.38**	**8.89**	**0.81**

were 30 sections that need to be assigned to two TAs each, so the total number of required assignments is 60. The algorithm was able to make 57 successful assignments while adhering to the problem constraints, which accounts for 95% of the required number of assignments.

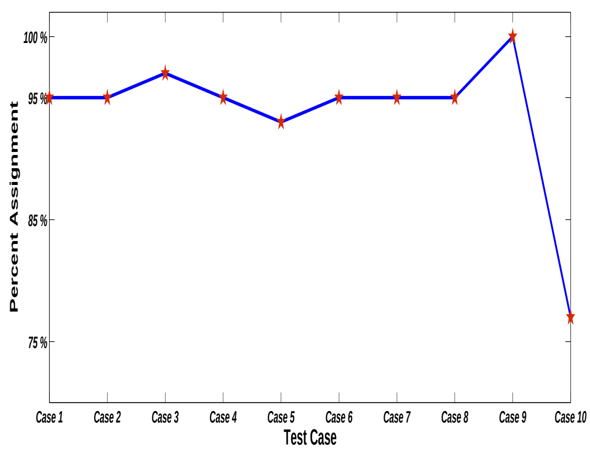


Fig. 1: Percentage of Sections Assignment

The following can be deduced from Table 1 and Fig. 1:

- The algorithm was very successful in satisfying the preferences of the TAs. This can been seen from the high percentage of assignment of preferred courses, which is specially noticed in preference 1 (i.e., the most desired course). In fact, the only two cases where there was a large percentage of assignment of non-preferred courses (preference 0) were cases 2 and 3, in which some of the TAs did not specify any preferences. If we exclude these two cases from the calculation of the overall average of preference 0, the average will be reduced from 37% to only 26%. We can also deduce from the algorithm that there can be three possible reasons for assigning a non-preferred course to the TA.
  1) The preferred courses may be conflicting with her availability.
  2) She may have been assigned one or more of her preferred courses, but her total number of assigned hours is still small compared to other TAs.
  3) Her preferred course(s) may have been already assigned to other TAs who have a higher priority in the TAs' list (e.g a graduating Masters student, or an older TA).
- With the exception of the special cases 9 and 10, those having a fewer number of sections and a fewer number of TAs respectively, the algorithm was very stable in the number of hours assigned to the TAs. For Masters students TAs, the average assigned hours ranged from 6.2 to 7.2. For full-time TAs, the average assigned hours ranged from 8.4 to 10.
- Again, with the exception of cases 9 and 10, full-time TAs were assigned, on average, approximately 27% more working hours than Masters students TA. This is a reasonable assignment, given that the number of sections tried in this experiment is not very large. We expect that if there was a larger number of sections, the assigned working hours of full-time TAs could be as much as 50% more compared to Masters students TAs. This assignment would then conform to the conditions of the maximum working hours mentioned in Sect. 3.
- The processing time of the algorithm was very fast. In all test cases the resulting schedule was produced in less than 1 second. The average processing time is 0.8 seconds.
- We can observe from Fig. 1 that the algorithm was able to achieve a very high percentage of fulfilling the required sections assignments. With the exception of test case 10, in which there was only 10 available TAs, the percentage of assignments ranged from 95%-100%. The low percentage of assignment in test case 10 is expected, since there is a shortage in the number of TAs that can be assigned to the available courses. On the other hand, the 100% assignment rate in test case 9 was also expected since the number of courses in this test case was fewer than the other test cases. Having said this, it is our conjecture that the algorithm can even achieve higher assignment rates, if we increase the maximum number of attempts (currently set at only 10). This can be easily done since the algorithm is quite fast and a small increase in processing time will not degrade its performance.

6. Conclusions

This research is based on a BSc graduation project in the IT department of King Saud University, Riyadh, Saudi Arabia. The project is a web-based tool with an integrated database designed to help the TA assignment committee in the IT department to automatically assign TAs to lab sessions, while fulfilling a number of difficult constraints. The constraints included the preferences of the TAs, their availability and their maximum allowed workload. The core of the tool, which is the theme of this paper, is a 2-phase heuristic algorithm that deals with the above constraints and tries to achieve balanced schedules by dynamically changing the priorities in assignment, based on the current progress of the algorithm.

The experimental results indicate that the algorithm was very successful in producing the anticipated results in terms of satisfying the preferences of the TAs and balancing the workload among them. In addition, the algorithm was very fast, since the schedule was always generated in less than 1 second of time.

Although the algorithm in this project was specifically designed to meet the requirements of the IT department, it is a robust algorithm that can be applied in several other domains. For example, with just minor modifications in the problem settings, the algorithm can be used to assign courses to different faculty members, not only TAs. The framework of the algorithm is also general enough to allow the modification of constraints, based on the specific needs of a department or institution, without changing the whole algorithm.

It remains to test the tool in realistic settings to measure the satisfaction of users and decide whether it can completely or partially replace the tedious manual planning process.

Acknowledgment

Many thanks to Wjdan Al-Dawood, Ashwaq Al-Rudyan, Hind Al-Harbi, Sara Al Hayash and Noorh Al-Sfyan for their effort in designing the web-based tool and implementing and testing the algorithm described in this paper.

References

[1] S. Petrovic, “Towards the benchmarks for scheduling problems,” in *Workshop "Scheduling a Scheduling Competition"*, 2007, held in conjunction with the 17th International Conference on Automated Planning & Scheduling (ICAPS ’07).

[2] M. R. Garey and D. S. Jhonson, *Computers and Intractability: A Guide to the theory of NP-Completeness*. W. H. Freeman & Co, 1997.

[3] S. Yang and S. Jat, “Genetic algorithms and local serach strategies for university course timetabling,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, no. 1, pp. 93–106, 2011.

[4] M. Hosny and S. Fatima, “A survey of genetic algorithms for the university timetabling problem,” in *International Proceedings of Computer Science and Information Technology*, vol. 13, 2011.

[5] M. I. Hosny, “TACO: An automatic TAs to courses assignment tool",” in *EdMedia 2012 - World Conference on Educational Media and Technology*. AACE: Association for the Advancement of Computing in Education, June 2012, *to appear*.

[6] P. Parthiban, K. Ganesh, S. Narayanan, and R. Dhanalakshmi, “preference based decision making model (PDM) for faculty course assignment problem,” in *International Engineering Management Conference*. IEEE, 2004.

[7] N. A. Ismayilova, M. Sagir, and N. Gasimov, “A multiobjective faculty-course-time slot assignment problem,” *Mathematical and Computer Modelling*, vol. 46, pp. 1017–1029, 2007.

[8] C.-M. Hsu and H.-M. Chao, “A two-stage heuristic based class-course faculty assigning model for increasing department-education performance,” in *International Conference on New Trends in Information and Service Science*. IEEE Computer Society, 2009.

Reforming Introductory Computer Science Classes to included Inquiry Based Projects

Yenumula B Reddy
Department of Computer Science, Grambling State University, Grambling, LA 71245
ybreddy@gram.edu (http://www.symbolicscience.com)

Abstract - *Inquiry based learning is a new educational method that students explore the concepts and problem solving by participating in the complete process of development. The process involves new ideas, questions, suggestions, and criticism during the developmental process. The inquiry based project implementation was introduced in computer science program as part of Center for Mathematical Achievement in Science and Technology to retain and prepare the computer science students for 21st century work force. The initial implementation of the project was introduced in higher level courses for practice and brought the idea in introductory courses. The methodology involves the students in the learning process, improved the learning, and retains the information. The process further helps students to meet the prerequisites for the next level courses.*

Keywords: Inquiry based learning; problem based learning; retention; mini-projects; team projects;

1 Introduction

The word "reforming" shakes the community all the times. The reason is that the existing techniques and technology used need to be modified. The people learned and following the traditional techniques need to be changed by incorporating the new techniques as a society or participant group requires. The axiom "`Tell me and I forget, show me and I remember,` ***involve me and I understand***" is the source to Inquiry-Based Projects (IBP).

The students are changing from computer science major due to fear of programming. This made us thinking to introduce a new method of teaching using problem based learning (PBL) and IBP in introductory classes that help to build skills of programming and problem solving techniques. The team projects at the introductory level were impressed and encouraged by the ABET (*Accreditation Board for Engineering and Technology, Inc*) team of Computing Accreditation Commission (CAC) visited the computer science program at Grambling State University (GSU). Further, IBP in introductory classes helped us in retaining the students. The PBL at the introductory level is supported by the CMAST (Center for Mathematical Achievement in Science and Technology) program funded by National Science Foundation (NSF) at Grambling State University (GSU).

The traditional methods of teaching the introductory computer science courses use problem solving approach, write algorithm, flow diagram, and then develop a program for compilation. In this process, the instructor discusses the problem solving approach, explains the pseudo code (algorithm), and then enters in programming development. Recently, the methodology was simplified. The simplified method involves the explanation of the problem and then write program (assume the basic elements of the program were taught). The current generation forgets the information after completing the current instruction process because there is no direct involvement of students in developing the program. The instructor gets frustration to teach the same content more than one time. The students get the feeling that the subject is very difficult to learn and change the major instead of wasting time in learning the computer programming. The main reason was the student involvement was missing at every stage (step-by-step) of program development. New ideas, group discussions, develop the simple program or mini-project with complete involvement of students in the program development process.

The PBL is built in the process of observing, questioning, predicting, planning, communication, requirements analysis, and suggesting alternative methods. In the question answer process, the discussions of instructor and students lead to the answer. The process helps to develop concepts, new ideas, discussions, integration of ideas to solution. The PBL focusses on participation and involvement in the learning process unlike traditional methods concentrated on teacher centered explanation and mastering the subject (not necessary understand everything).

The traditional methods deliver the subject to make students understand the subject with no guarantee of retention. The IBP process clearly shows the complete involvement of each student in the problem understanding and step by step development of the project. Further, the instructor has a clear idea about the capabilities and strength of the student in the developmental process. The process helps the instructor to modify the teaching methodology and introduce the new approaches with student involvement. Further, participation of students throughout the developmental process maximizes the understanding and leads to create small team projects.

In the PBL process, students use the resources beyond the classroom. The resources beyond the classroom involve the students to connect local and world communities. The broad involvement encourages students to ask more questions and participation in problem solving. In this paper, we discuss two examples (due to space limitation) to develop simple program to mini-project.

2 The Real Purpose of Academic Project

It is worth spending time on projects (mini and large) as part of the degree program. The reason is that the student gets both theoretical and hands on experience. Further, there are many skills that cannot be taught in the classroom. Learning and strengthening the skills only possible through practical experience. We identified the following reasons to work on projects.

Working effectively as part of the team: In team work, the participants divide the project into tasks, monitor the tasks by the team leader, integrate, and test these tasks after completion. The weekly or by-weekly meeting helps the progress of the project work.

Interacting with users: Instructor assigns the classroom projects and told to solve it. In the team projects, the team is asked to develop a project through meetings and discussions.

Developing requirements analysis and design: In the classroom students learn specifications, analysis, and design of a project. The concepts can be applied to mini and large projects. Mini projects improve the basic skills whereas large projects involve complex package requirements and software development tools.

Developing prototypes: students normally see the working prototype before they start the project. In a semester time, developing a prototype and implementing the project within a semester is not possible. The mini-project experience helps the team to select large project and can complete on-time.

Writing and oral presentation skills: The projects help students to interact, communicate, written, and oral presentation skills.

Therefore, the project work is not just writing code, but involves team work, selection of problem, time-lines, discussions, and complete development process. Mini-projects are more helpful to strengthen the student's skills and confidence in developing the projects.

3 First Experimental project

ABET accredited computer science program at GSU identified the following courses as core courses.

CS 110	Introduction to Computer science I
CS 120	Introduction to Computer science II
CS 210	Discrete Structures
CS 225	Computer Organization and Assembly Language
CS 235	Data Structures and Algorithms
CS 300	Computer Science Seminar I
CS 310	Software Engineering

The remaining courses are built on these courses. The IBP is practiced in CS 110, CS 120, and CS 235 courses. Initially, inquire based projects were tested in CS 310, CS 414 (computer Architecture), CS 424 (Advanced Programming Techniques), and CS 456 (Special Topics in Computer Science). As part of the IBP, the 3 member teams were formed for selection of projects. The selection of the project, feasibility study, and components to include in respective projects were discussed and prepared the time line to complete the projects. The brain storm took at least 5 meetings in two weeks. We created the blackboard approach for open discussions, and started writing ideas to implement the project. The teams completed the requirements analysis and design with group effort and created the algorithms to complete the project. The team participation helped us to complete the project in the scheduled time. The IBP methodology encourages all students particularly students who are not confident to complete a project within the semester time. The students learned to work on team projects, understood the work, and had good grades.

4 Experiments at Introductory Level courses

The IBP model completed with small teams was brought to introductory level courses CS 110, CS 120, and CS 235, since the retention of the students in computer science is based on these three courses. The retention process starts in the teaching methodology in the first course (CS 110). The process to introduce Java language can be done in two ways. The teaching experience shows that teach the basics through Java program examples or first teach basics and then introduce program writing. Most of the times participants involved more in the first method (starting with an example and slowly introduce language requirements: comments, declarations, expressions, statements, garbage collection, syntax, and semantics). Once the language concepts are clear through examples, the instructor can successfully introduce the small program projects (IBP) through PBL process.

In this document, we discuss two examples due to space limitations. The first example project discusses the problem based learning and second project inquiry based project. In the first program example, we assume that students have the concepts of iterations (for loop and while loop). The development of the first program assumes that the students understand the class concepts and methods with input/output statements.

Example 1: Participating in the first project. Problem based learning

For example, develop a program to add 10 numbers. Assume that stunts have completed the iterations in the classroom. The following steps are involved in the initial level of learning.

Instructor: what is the method of adding the first 10 numbers?

Student1: 1+2+3+…+10;
Student2: add first 10 natural numbers;

Instructor: Where do you get these numbers to add?

Student1: read one number at a time and add;
Student2: read all numbers at one time and add

Student3: Since we know these are first 10 natural numbers we can add them directly without reading (student1 – first step). It saves time.

Instructor: All methods are correct. But we need to know the best method among these. If we have to add first 10 natural numbers, use the following method.

```
n=1; sum =0;
repeat the following step until n less than or equal to 10
   add n to sum and increase n by 1
```

The simple method is, take the first number and add to the current sum (let current sum be initialized to 0). Increase the number by 1 to get the next number and add to the previous sum. By doing this, you do not need to save all the numbers in memory or type ten numbers on the screen.

In the next step, we need to generalize the method. Therefore, the method must apply to add any number of natural numbers. The method must have the facility of selecting starting number and ending number. We will rewrite the above method as below.

```
n=starting_Number;  m=ending_Number;  sum =0;
repeat the following step until n less than or equal to m
   add n to sum and increase n by 1
```

The above method is general and applies to all similar problems (irrespective of any starting and ending number). The best method means that it should save time, program steps, and computer memory space. In addition, the method must apply for a different set of numbers (may be ten thousand).

It is easy to explain the program steps with team contribution using problem solving techniques. The instructor supervises and helps to complete the implementation of the program on computer. Further, the instructor creates similar problems to strengthen the concepts of problem solving and implementation. Some of the examples are given below that the students will get clarity to understand problem solving process and implementation on a computer.

- Sum of even numbers less than 20
- Sum of odd numbers less than 20
- Product of first 20 numbers
- Product of even numbers less than 20
- Product of odd numbers less than 20

Therefore, an instructor can create many similar problems to get a clear understanding of concepts and solve the problems in the PBL classroom.

Example 2: Inquiry-based project after completion of 2-D arrays (Matrix)

Example 2 assumes the basic knowledge to develop a mini-project. We divide the students into 3 to 5 member groups in the classroom depending upon the class size. Each group develops the same project and participates in the classroom discussions. The problem selected was Magic Square.

Objective:

Students will work cooperatively in a Team and analyze the magic square problem

Rules:

1. Students work in Teams
2. Students discuss strategies to create programmable puzzle

Procedure:

1. The teams were given a start point of the puzzle and the placement of the first 3 numbers
2. All game rules to create the magic square were explained. Each student was encouraged to ask questions while developing the algorithm
3. Starting position of the first number can be changed and other numbers depends upon the change (but make certain that a magic square will be achievable).

Questions considered by the whole group:

1. What are the strategies available to place the numbers on the square with minimum effort?
2. Is there any reason that first number to put in a specific position?
3. Verify the sum of each row, column, and diagonal of the magic square? The sum must be equal in each case.
4. If the first three numbers 1, 2, and 3 are placed as given in the Figure 1, is there another unique solution to the puzzle? Find all possible solutions and follow one to implement that is more suited to implement in Java programming?

Procedure to place the numbers

Initially students were told the procedure to start the 3x3 square and place the first number in the middle of the first row. Students have many questions about the placement. I told them that the placement and staring number can vary. To achieve the solution, we have to select certain strategy and understand the process. Later different methods can be used to achieve the solution. In Java language, indexes start from 0. The index value of the first element of an array is 0. The index value of the first element of a matrix is (0, 0). The matrix must be initialized before you start the process. Test each cell before place the next number. The rules of placement of numbers are explained as below.

- The rows and columns outside the colored part of the square have special meaning. The row above the first row is replication of the last row. The row at the bottom (below the last colored row) is the replication of the first row. Similarly, the right side non-colored column is replication of the first column and left side non-colored column is replication of the last column (rightmost colored column).
- Placement of the next number (2) will be rightmost top corner of the square. Thinking of program design steps, we have to subtract 1 from row and increase column by 1. The process conditions are given below.
  - ➢ If the row is less than 0 replace row to maximum (row=2). If the row is greater than 2 (maximum size 2) then set row as 0. If the column index is greater than 2, then set column index as 0. If the column index is less than 0 then set column index as 2. In the current strategy, row will not be more than 2 and column will not be less than 0.
  - ➢ If the number is at the rightmost corner, then increase the row by 1 and do not change column (this happens when both row and column crosses boundaries).
  - ➢ If the next position is filled by decreasing the row and increasing the column, then row will be increased from current value without changing the column value. For example, after the number 3 was placed the next cell is to be filled by decreasing row and increasing column. In that position, already 1 was placed (occupied). In such situations increase row and no action on column. The number 4 will be placed in the new indexed position.

The procedure to place the numbers in the square was explained more than once to make the students understand. Once they understood the procedure, they completed the magic square on the board. After students understand the problem solving method, I explained the procedure (algorithm) to use in development of the program. The algorithm design completed with many questions related to decreasing row, increasing column, corner cell, filled cell process, and check the row/column boundaries.
Students then tested with any natural number as starting number and larger size of the magic square. Once they understood the process, I explained the

generalization process that we can do any odd number size magic square. The students surfed the literature from Web and showed me the various procedures and strategies to develop the magic square. They found various starting points, various size squares, and placement of numbers in different directions. They implemented the program by moving the next position in different directions. The program is large but using the PBL process students could do this mini-project successfully. There are other mini-projects completed using inquiry based project concepts. The modules will be made available through our tutorials.

CS 235 (Data Structures and algorithms) is a decision level course for computer science students. Change of major after completion of CS 235 (Data Structures and Algorithms) is negligible. Therefore, IBP is very important in introductory courses (CS 110, VS 120, and CS 235).

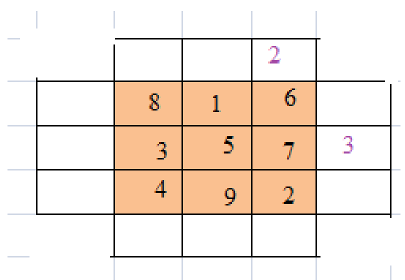


Figure 1: Magic Square

5 Lessons Learned

Introduction of mini-projects is a challenging task at the introductory level courses. The team formation and dividing the tasks and explain how to divide the project into tasks, responsibilities of each team member and team leader is another problem in the first time. Students understood the process after the first time.

Answering the questions, explain the information sources, and collecting the needed information is another important task in the process. The fusibility study of the project and time-line is another important task. Involving the team members and monitoring each team member to complete the scheduled work on-time is a challenging task.

6 Conclusions

The mini projects at the introductory level make a lot of difference in acquiring the depth of knowledge and learning process. Mini-projects are challenging in introductory courses and participation of students in the project development was impressive. The problem understanding and attempting another new project was improved. Using problem based learning and inquiry based project implementation the computer science faculty members are confident to achieve the projected retention in the CMAST program.

Acknowledgement

The author wishes to express appreciation to Dr. Connie Walton, Provost & Vice President of Academic Affairs, Grambling State University for her continuous support for "Inquiry-based Projects in introductory computer science classes". Thanks to Dr. Wynn, Dean, Dr. Hubbard, Associate Dean and the computer science faculty Dr. Gassant and Dr. Mesit for their valuable comments and discussions. The author is also thankful to CMAST (Center for Mathematical Achievement in Science and Technology) program funded by National Science Foundation through grant# HRD-1137590 for the support of IBP where PBL.

7 References Cited

1. Center for Inquiry-Based Learning: http://www.ciblearning.org.
2. The Exploratorium Institute for Inquiry: http://www.exploratorium.edu/ifi.
3. Problem Based Learning, Michael Wald (Editor-in-Chief), SPECIAL ISSUE, Volume 19 Number 5, Volume 19 Number 5 2003 ISSN 0949-149X.
4. Barrows, Howard S.., "Practice-Based Learning: Problem-Based Learning Applied to Medical Education", Southern Illinois University, School of Medicine, IL 62794-9230. ISBN-0-931369-27-4, 1994.
5. Bridges, Edwin M., "Problem Based Learning for Administrators", ERIC Clearinghouse on Educational Management, University of Oregon, ISBN-0-86552-117-4, 1992
6. Geoffrey R Norman, Henk G Schmidt., "Effectiveness of problem-based learning curricula: theory, practice and paper darts", http://onlinelibrary.wiley.com/doi/10.1046/j.1365-2923.2000.00749.x/full
7. H. S. BARROWS., "A taxonomy of problem-based learning methods", http://onlinelibrary.wiley.com/doi/10.1111/j.1365-2923.1986.tb01386.x/full

Kinect Academy

Stuart Anderson, Vu Pham, Monish Parajuli, Anh Pham, Majid Khan, Dat Tran and Dharmendra Sharma
Faculty of Information Sciences and Engineering
University of Canberra, ACT 2601, Australia

Abstract - *Microsoft Kinect is a low cost natural user interface (NUI) device compared to other NUI solutions. Inspired by the Kinect Effect while keeping in mind the lack of educational games for kids on Kinect and lack of physical activities on other platforms, we propose Kinect Academy—a research project that fills the gap by providing educational application for children which is filled with activities. This project uses Kinect's NUI capabilities to help children learn by letting them have fun while learning. This projects aims to be a tool for educators to teach children while keeping them physically active by being to interact with educational materials in a new and fun way. Kinect Academy tries to achieve our aim by using Kinect to make learning fun and linking it with an open and collaborative web platform for education.*

Keywords: Microsoft Kinect, Kinect Academy, Natural User Interface

1 Introduction

Most of skills that children are going to use for the rest of their lives are developed during their child's formative years. During this time when they are developing their language and comprehension skills, it is of the utmost importance that they have a solid foundation of literacy and numeracy upon which they can build their further knowledge. The research of Ainley and Fleming (2000) shows that learning to read in the early primary years helps develop children's literacy skills. Also, The William and Flora Hewlett Foundation, which works towards Quality Education in Developing Countries explains the importance of education in economic developing countries by mentioning that literate and educated citizen have better economic opportunities, better productivity and better health (Global Development and Population Program). However, many children cannot access to education at this age, especially children from the poor countries due to lack of infrastructure.

To support children from poor countries to have access to education, International Monetary Fund (IMF) has been providing monetary assistance to developing countries of the world towards achieving United Nation's Millennium Development Goals (Factsheet: The IMF and the Millennium Development Goals, 2011). According to IMF's report, in developing countries enrollment of children in primary school has generally improved and the percentage of students completing primary education has also improved compared to 1990's. However, surveys conducted by IMF have found that completion of primary school does not confirm students learning basic academic skills. The surveys have found that in many low-income countries that students receiving some schooling still cannot read, write or count. (Arye L. Hillman, 2004).

In this research project, we intend to prevent open ended problems by focusing mainly on children from four to eleven years of age, which is the age of primary grade in many countries. This is because there have been studies such as Brotherson (2005) and Live and Learn (2006) showing that this age group is a critical phase for children to develop. Also, it is in line with one of the United Nation's Millennium Development Goals which is to achieve the universal primary education for children around the world. We aim not only to provide open access to learning opportunities, but also to make engaging with these learning opportunities as enjoyable as possible, so as to encourage their use. This can be supported by studies showing that children learn more affectively, even with the complex concepts, while having fun.

2 Current Work

Although the concept of helping children develop skills and assist learning using computers is not new, there are still many limitations with the existing solutions. There exists plethora of computer programs like ClickN Read, ClickNSpell, Number Munchers, Math Blasters and Tux math. There also have been several applications in gaming platforms like Body and Brain Connection, Brain challenge, Buku Sudoku, Pictionary, Little Big Planet, Brain Training, My Word Coach in PlayStation, Nintendo Ds, Wii, Microsoft XBOX as well as Microsoft Kinect. However, there is common issue in all the above mentioned applications that the applications provided on the gaming platforms lack strong focus on the educational aspect, while computer applications are not very interactive and/or lack the physical interactions provided by the gaming platforms.

Still most of those applications and their implementation in an educational setting would require funding which is scarce in developing countries. It is not just the issue of cost, providing such facilities would need various infrastructures (like physical infrastructure for storage, electricity and internet connectivity).

3 Kinect Academy

3.1 Kinect Device

The Kinect device gets 3D scene information from its 3D depth sensors. This sensor consists of an infrared laser projector combined with a monochrome CMOS sensor, which captures video data in 3D under any ambient light conditions. The device also has a RGB camera and a multi-array microphone for speech recognition.

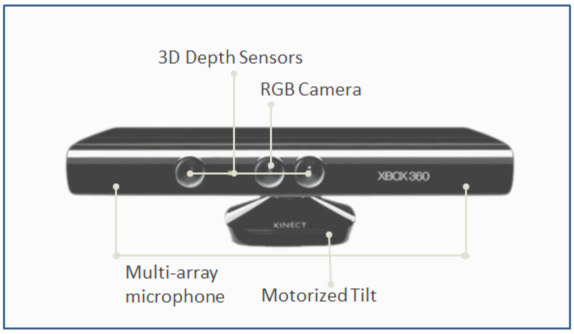


Figure 1. The Kinect device

3.2 Motivation

There has been a trend called the "Kinect Effect" where independent individuals and groups have made applications using the Kinect device using Microsoft Kinect SDK as well as open source SDK that use the Kinect device in novel contexts. Some examples of such applications in education are KineSis, Kinect Math and Kinect Paint from KinectEducation. It is noteworthy to mention Microsoft's effort to used Kinect to help teaching kids in Africa.

Following the spirit of the "Kinect Effect" while keeping in mind the lack of educational games for kids in Kinect and lack of physical activities on other platforms, we develop our project Kinect Academy to fill the gap by providing educational game for children which is filled with activities. This research project intends to use Kinect's NUI capabilities to help children learn by letting them have fun while learning. This projects aims to be a tool for educators to teach children while keeping them physically active by being to interact with educational materials in a new and fun way.

3.3 Human-Machine Interface of Kinect Academy

As the project concentrates on assisting children in learning and making learning fun for them, usability is one of the priorities. Indeed, the goals for this project, apart from the functional goals, include making the application:

- **Intuitive**: activities such as waving hands or using voices to communicate were learnt by individuals at very early age. By using Kinect which enables users to use Natural User Interface, we aim to provide users methods which they can easily intuit to interact with the application. Also, by logically laying out objects from left to right, top to bottom, use readable fonts and avoiding using technical words, we aim to make the application easy to learn to use for users. Therefore, even with children and individuals that have limited background of technology, they can easily learn to use the application without having to go through the process of trial-and-error.
- **User friendly**: the application was developed so that it provides ease of use for both children and educators by applying Neilson's 10 usability heuristics. This aims to reduce users' efforts in terms of navigation, recognition, error prevention and error recovery. Also, the color scheme was designed by applying the principal from Web Content Accessibility Guidelines 2.0 to create the distinction between objects and prevent users from feeling uncomfortable when working/using with the application for a long period.

3.4 Kinect Academy System

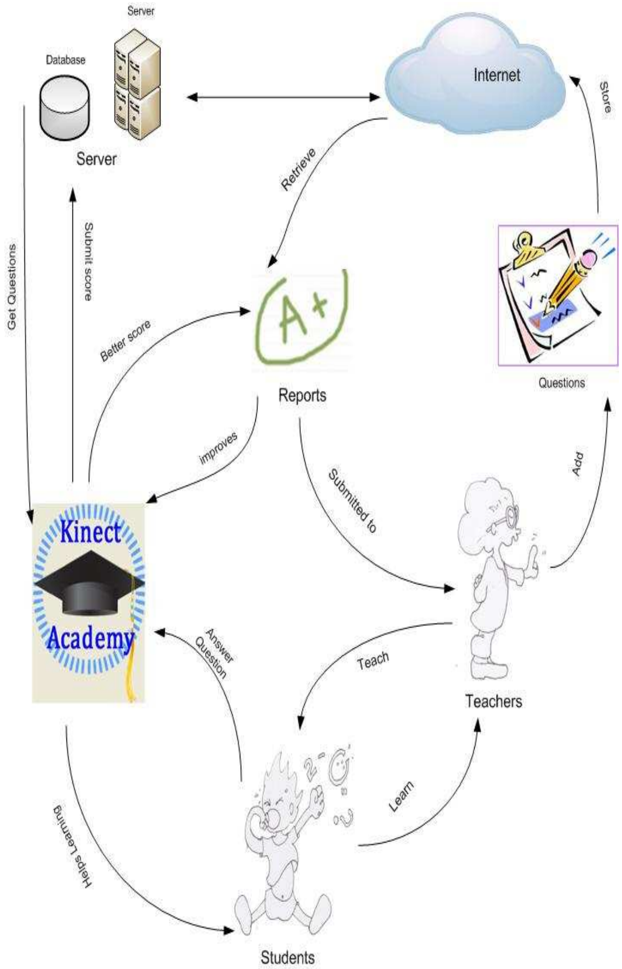


Figure 2. System architecture of Kinect Academy

Kinect Academy fully supports presence of fresh and updated data. It allows educators to collaborate, so they can modify the questions and answers to fit their purposes. This is done via the website and the changes will be visible at the application level immediately. Two-tier architecture was implemented.

Kinect Academy is focusing on two things. Firstly, using Kinect to make learning fun, and secondly, using the sharing of information to create an open, free and ever expanding platform for education. Kinect Academy allows users to provide answers to sets of quiz questions covering any topic imaginable, with the aim that using body movement and voice commands will be more engaging than traditional learning methods. Information sharing will be facilitated by the application accessing a remote data server to retrieve new question sets, organized according to category and complexity, and users will be able to add new question sets or update existing sets via a web site.

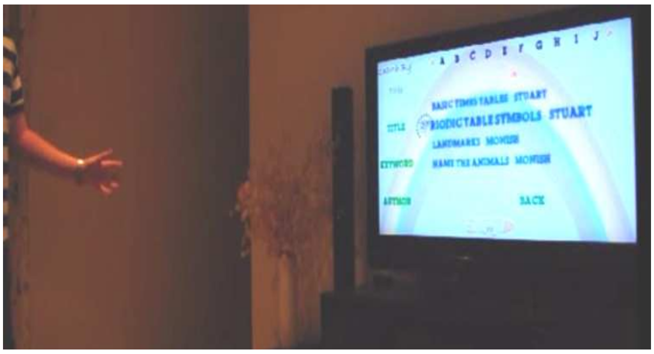


Figure 3. Kinect enables users to use Natural User Interface, waving hands or using voices to communicate.

Kinect Academy will make use of skeletal tracking as its primary form of input, tracking users' hands to interact with the application. Hands will be used to navigate the application by hovering over icons representing menu commands or answer possibilities. Skeletal tracking will also be used to implement gesture recognition for pre-conceived body gestures which will be used to activate certain functions within the application, for example to bring up the menu overlay interface or pass on a question. Wherever possible, voice recognition will augment skeleton tracking, with the menu accessible by voice commands and all icons having an associated voice command, providing multiple input options both in selecting answers and navigating the application. This dual control option gives the user flexibility, variety, and freedom.

In general use, the user will be presented with a full screen play area consisting of floating icons. On menu screens, these icons will be used to navigate between various screens in the application, sometimes accompanied by a title of brief instructional prompt. On play screens, icons will be text, polygon or image based and will represent possible answers corresponding to an on screen textual/visual question prompt. A partially transparent menu overlay will provide access to additional GUI icons on most screens. Basic use of Kinect Academy will consist of a user selecting a question set to attempt, completing the question set, and reviewing their performance.

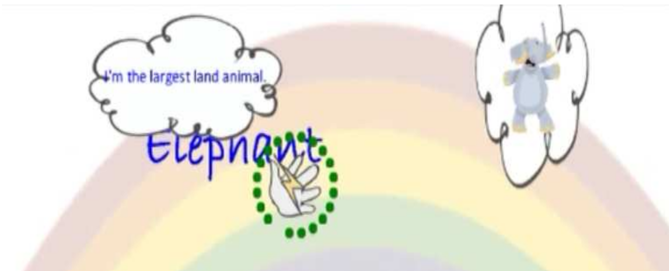


Figure 4. User friendly interface design: ease of use for both children and educators.

Kinect Academy will make use of the Gadget Accelerator Kit along with custom game engine along purpose built to provide some of the functionality required by Kinect Academy, namely per pixel collision detection, random 2D shape generation, and conversion of polygons and text to texture formats compatible with the collision detection implementation.

3.5 Benefit

Although learners, especially children from four to ten years of age, are the target group, educators can also benefit from this. Indeed, the educators will be able to quickly and easily provide for them a set of categorical questions tailored to the learning needs of their learners and presented so as to maximize fun. In an open framework for learning, these questions will then be available for free for the benefit of anyone who has access to the application, allowing children not in school the opportunity to engage their minds, learn new things and have fun while doing so. People with expert knowledge will benefit from a platform via which they can share their knowledge, and older, post-school people will benefit from a place to learn new things or test themselves in an area of interest.

While a broad group of people will benefit, the impact will differ from demographic to demographic. While existing students will have their learning experience enriched, e.g. by creating a positive learning space or a less daunting testing environment, students with no previous access to education will be offered the chance to start learning without a teacher and get a taste for education, developing a thirst for knowledge and a positive attitude towards learning. Adults will get the benefit of optional self-education in areas of their own choosing.

3.6 Application

We believe that the solution can become realistic in the near future since it is relatively cheaper than most of the existing solutions. Indeed, all we need are a laptop, a Kinect and a couple of dependencies such as .Net framework 4.0 and XNA framework. The dependencies are free and provided by Microsoft. The cost will possibly be reduced remarkably as One Laptop Per Child (OLPC) Foundation is tackling the huge issue of providing computer access to every child in developing countries.

Although the project uses framework, it can be commercialized by leveraging the popularity Kinect devices to assist educators (parents and teachers) teach and children learn. The lack of educational software using natural user interface has been observed and there is a great potential for this application.

4 Conclusion

This research project manages to address the lack of educational software using natural user interface and making learning fun from anecdotal evidence from children who are the potential educators. However there are rooms for improvements in terms of user interface, usability and features. The project is technically open for future upgrades and collaboration and open participation. XML is implemented so that the application can communicate, retrieve and update data while the website was built to allow users to perform editing data to the server. This not only has enable users to add, edit and delete data using other device such as phones and tablets but has also made Kinect Academy platform independent which means it can be extended to other platforms in the future.

5 References

[1] Ainley, J. & Fleming, M. (2000). Learning to Read in the Early Primary Years. Rertieved 7 September 2011 from http://www.dest.gov.au/NR/rdonlyres/74CF524B-6E4F-45B8-AF7D-41B1C09AE29F/1577/LearningToRead.pdf

[2] Arye, L. & Hillman, E. J. (2004). Educating Children in Poor Countries. Economic Issues . International Monetary Fund.

[3] Brotherson, S. (2005). Understanding Brain Development in Young Children. Retrieved from North Dakota State University Website http://www.ag.ndsu.edu/pubs/yf/famsci/fs609w.htm

[4] Buzzle.com. Children: Learning by Playing. Retrieved from http://www.buzzle.com/articles/children-learning-by-playing.html

[5] International Monetary Fund. (2011). Factsheet: The IMF and the Millennium Development Goals. Retrieved 10 07, 2011, from International Monetary Fund Website: http://www.imf.org/external/np/exr/facts/mdg.htm

[6] Live and Learn. Developmental Activites For Children From Age 4 To Age 5. Retrieved from http://www.liveandlearn.com/4to5.html

[7] Willian and Flora Hewlett Foundation. Global Development and Population Program. Retrieved 03 01, 2012, from The: http://www.hewlett.org/programs/global-development-and-population-program

[8] Kinect for windows SDK beta. Retrieved 12/08, 2011, from http://research.microsoft.com/en-us/collaboration/focus/nui/kinect-windows.aspx

[9] Guinness World Records. Fastest-selling gaming peripheral. Retrieved 12/14, 2011, from http://www.guinnessworldrecords.com/records-9000/fastest-selling-gaming-peripheral/.

[10] M. Fisher, Kinect. http://graphics.stanford.edu/~mdfisher/Kinect.html

Exploring the Use of Peer Mentors in Calculus to Support Retention Efforts in Engineering

Deborah L. Worley[1], **Jeremiah Neubert**[2], **Naima Kaabouch**[3], **and Mohammad Khavanin**[4]
[1]Department of Educational Leadership, University of North Dakota, Grand Forks, ND, USA
[2]Department of Mechanical Engineering, University of North Dakota, Grand Forks, ND, USA
[3]Department of Electrical Engineering, University of North Dakota, Grand Forks, ND, USA
[4]Department of Mathematics, University of North Dakota, Grand Forks, ND, USA

Abstract – *This paper describes the preliminary results of using advanced engineering students as peer mentors in delivering calculus course content to students who express interest in engineering as an academic major. The data were used to inform the design and delivery of supplemental study sessions for calculus courses that are intended to improve retention rates in engineering. The overall study is funded by a National Science Foundation (NSF) Course Curriculum and Laboratory Improvement (CCLI) grant.*

Keywords: engineering peer mentors, student success, retention

1 Introduction

Research on retention in science, technology, engineering, and mathematics (STEM) fields emphasizes that students are often academically unprepared or underprepared for the rigors of engineering [1]. Many of the students that switch away from engineering cite math, specifically calculus, as one of the most influential factors in their decision [2]. Students who switch to another major may have deficiencies in their mathematical background or study skills and are not prepared for the rigors of college calculus. Additionally, students may not understand the importance of calculus as an engineering tool.

We seek to provide ways for institutions to capitalize on opportunities to educate the next generation of engineers. A traditional engineering curriculum requires students to endure a year of math and science courses before being immersed in engineering. Others have identified the need to include engineering content and equations in calculus courses [2][3][4]. Given what is known about the calculus barrier to engineering study, we focused our efforts on embedding real-world engineering problems into calculus courses.

A significant obstacle to teaching a calculus course with engineering content is having sufficient support for both the mathematics and engineering content. One solution to this problem is to have a mathematics professor teach the lecture portion of the calculus course and use upperclass-level mentors to help the students learn the engineering content. To that end, the purpose of this project is to explore the use of advanced engineering students as peer mentors in delivering calculus course content to students who express interest in engineering as an academic major. This study is a work in progress, and part of a larger study where we design and deliver supplemental study sessions for calculus courses that are intended to improve retention rates in engineering. The overall study is funded by a National Science Foundation (NSF) Course Curriculum and Laboratory Improvement (CCLI) grant (Project DUE-0942270).

2 Peer Mentoring and Reciprocal Peer Coaching

Peer mentoring and reciprocal peer coaching are integral components of our project; we expect the engineering peer mentors to provide real-time, instructive feedback as well as guidance on efficient study habits to students enrolled in calculus and considering engineering as an academic major. Moreover, we use peer mentors to connect students to campus resources as well as to each other. Peer mentoring is regarded as a successful intervention to address issues of student retention in academic programs [5]. Peer mentoring is defined as "a helping relationship in which two individuals of similar age and/or experience come together … in the pursuit of fulfilling some combination of functions that are career-related and psychosocial" [6]. For students in engineering, the act of pairing incoming students with peer mentors benefits students in a variety of ways, including reduction of feeling of isolation, improved academic performance, and an increase in minority student persistence [7][8].

Reciprocal peer coaching, a form of peer-assisted learning, shares many characteristics with peer mentoring, as it "encourages individual students in small groups to coach each other … so that the outcome of the process is more rounded understanding and a more skillful execution of the task in hand than if the student was learning in isolation" [9]. Reciprocal peer coaching encourages and motivates students to work toward a common goal; there is also an opportunity for students to give and receive feedback in a small group setting, a learning environment that is less "risky" than the traditional classroom setting.

The students who served as peer mentors in this project participated in training exercises prior to facilitating group sessions. The mentors received information on group facilitation and leadership, in addition to instructions as to how to distribute the module problems and schedule the small group meetings. The mentors were also guided by a member of the research team on how to create a safe learning environment and meet the needs of individuals with different learning styles. A summary of mentor responsibilities includes:

- Schedule mentor sessions with students.
- Facilitate group discussions
- Provide general support for learning engineering content
- Serve as a resource to help students connect with each other, and to engineering as a discipline
- Provide feedback to project faculty on students' levels of participation in sessions
- Provide process-related feedback to project faculty on the modules and mentor sessions

At each session, the mentor was instructed to:

- Create an atmosphere that is comfortable for learning
- Select a student to share her/his process for solving the problems in the module.
- Encourage other students in attendance to comment in a positive manner and ask questions as they go through the process.
- Guide the students in a discussion that will help them find the most efficient, correct method for solving the problem.

3 Methodology

In this project, students use real-world engineering examples, provided in learning modules, to learn calculus concepts. Students work on the problems with experienced, upper-level engineering students as mentors in specified study sessions that take place outside of class. The full study began in the fall 2011 semester with students enrolled in Calculus I.

Students were informed of the opportunity to participate in the study prior to enrolling in Calculus I. To encourage participation in the study, we highlighted the fact that real-world engineering problems would be incorporated into the class as well as the opportunity to work with an experienced, upper-level engineering student in specified mentor-led study sessions. We offered students an opportunity to use their work at the mentor-led study sessions to count toward earning extra homework-related points. Students who enrolled in the course but chose not to participate were still invited to the mentor study sessions so as to avoid bias or penalty for choosing not to participate in the study.

The mentor-led study sessions were small group discussions of engineering problems that can be solved by applying concepts from Calculus I. Students in the study were organized into small groups to work on the module problems. Groups were constructed using the Comprehensive Assessment for Team-Member Effectiveness (CATME) Team-maker tool, developed and maintained with support from the National Science Foundation [10]. The problems were organized into learning modules that were distributed by the peer mentor. There were 8 modules and corresponding mentoring sessions, spaced evenly throughout the semester. Each mentoring session lasted approximately one to one and a half hours; no session exceeded two hours.

Members of each small group met with the mentor to discuss the modules within two weeks of receiving the modules. At these meetings, group members took turns presenting solutions to the module problems. When not presenting, students were strongly encouraged to engage in discussion. Students were expected to attend a minimum of five mentoring sessions. Participants in the study were asked to provide feedback on their experiences at the end of the semester by completing the Modules and Mentors (MM) survey.

The Modules and Mentors (MM) Survey was designed to learn about students' experiences in their calculus classes related to the engineering concept problems and working with their engineering peer mentors. There were 18 total questions on the survey, nine of which concerned students' feedback on working with the peer mentor and their interactions with other students in the peer-led study sessions. In addition, participants responded to questions that focused on their self-reported comfort and confidence levels with calculus, as well as on connecting calculus knowledge to the study of engineering.

The Modules and Mentors survey instrument was reviewed and approved by the Institutional Review Board (IRB) at the University where it was administered. Upon certification that the study met the human research subject protections and obligations as required by law and University policy, the survey was distributed online to students who were enrolled in Calculus I in the fall 2011 semester and participated in the mentor-led supplemental study sessions at a large research university located in the Midwest (n=46). Students' completion of the survey was voluntary. As such, the response rate was approximately 21.7% (n=10).

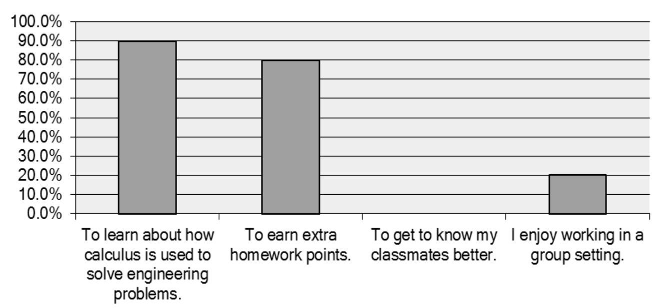
4 Results

The results of the content-based questions on the Modules and Mentors survey are organized into three sections: 1) motivations; 2) perceptions on the human components and; 3) confidence levels. The first section concerning motivations includes participant self-reporting of why they chose to attend the mentor-led study sessions. The second section, perceptions on the human components, asked students to report information about the degree of helpfulness they experienced by working with a peer mentor as well as the feeling of connectedness to other math and engineering students. The third section addresses participant self-reported confidence levels in mathematical abilities, engineering, and their comfort, as well as students' perceptions in using calculus to solve engineering problems.

4.1 Motivations

All of the students who responded to the survey attended six to eight (75.0-100.0%) of the mentor-led study sessions. Nine out of 10 respondents indicated that they attended to "learn how calculus is used to solve engineering problems". The majority of students (80.0%) were also motivated to attend by the opportunity to "earn extra homework points" for class. Although two students noted that they preferred to learn in a group setting, no students indicated their reason for attending was to "get to know [their] classmates better". See Table 1 for a graphic representation of student motivations for attending the mentor-led study sessions.

Table 1
Student Motivations for Attending Mentor-led Study Sessions



4.2 Perspectives on human components

For purposes of this paper, we were primarily interested in learning how students were impacted by the human components of the study: working with engineering peer mentors. As previously stated, the peer mentor led each study session and facilitated group work toward exploring and finding solutions to the engineering-related calculus problems. In the fall 2011 study, participating students felt that working with the engineering peer mentor was "helpful" (40.0%), "very helpful" (20.0%), or "somewhat helpful" (20.0%) in learning calculus (see Table 2).

Table 2
Student Perceptions of Helpfulness of Working with an Engineering Peer Mentor

	%
Not at all helpful	10.0%
Somewhat helpful	20.0%
Helpful	40.0%
Very helpful	20.0%
Extremely helpful	10.0%

In the same semester, 40.0% of participating students felt "connected" to other students in math and engineering after taking Calculus I and participating in the mentor-led study sessions (see Table 3).

Table 3
Student Perceptions of Connectedness to Other Students in Math and Engineering

	%
Not at all connected	20.0%
Somewhat connected	40.0%
Connected	40.0%

4.3 Confidence levels

Students remained "confident" (50.0%) and "somewhat confident" (30.0%) in their mathematical abilities following completion of their participation in the mentor-led study sessions (see Table 4). Concerning their confidence in engineering subject matter, 40.0% of students expressed they were "somewhat confident" and 20.0% felt confident in their ability to succeed in engineering (see Table 5). Table 6 further illustrates students' comfort and confidence levels in using calculus to solve engineering problems. Table 7 highlights perceptions of using calculus to solve engineering problems, specifically concerning the frequency with which students felt calculus is necessary to solve engineering problems. Responses to this question indicate that the majority of participants (60%) felt that calculus is always required to solve engineering problems. No participants reported that they did not see a connection at all between calculus and engineering.

Table 4
Self-reported Student Confidence Levels in Mathematical Abilities

	%
Not at all confident	0.0%
Somewhat confident	10.0%
Confident	50.0%
Very confident	10.0%
Extremely confident	30.0%

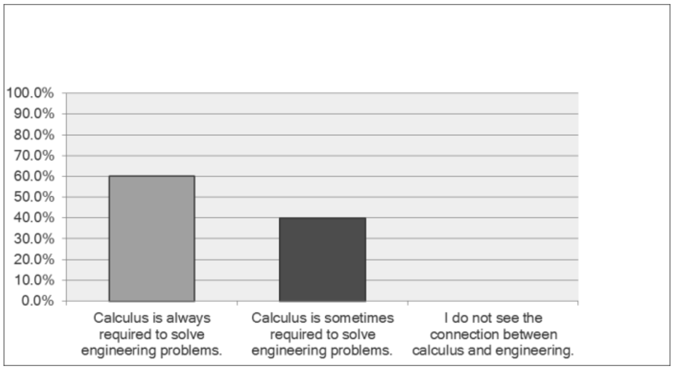
Table 5
Self-reported Confidence Levels in Engineering

	%
Not at all confident	10.0%
Somewhat confident	40.0%
Confident	20.0%
Very confident	20.0%
Extremely confident	10.0%

Table 6
Student Comfort Levels in Using Calculus to Solve Engineering Problems

	%
Not at all comfortable	0.0%
Somewhat comfortable	20.0%
Comfortable	60.0%
Very comfortable	10.0%
Extremely comfortable	10.0%

Table 7
Student Perceptions of Using Calculus to Solve Engineering Problems



5 Discussion

The opportunity for students to participate in reciprocal peer coaching and to get direct instructive feedback on their problem-solving methodology is not always possible in a traditional calculus course. Working in a small group setting seems to be of some benefit to students in our study, but the perception of helpfulness of working with a peer mentor was not as strong as expected. Student performance and attitudes about calculus and the study of engineering may benefit from a revision to the peer mentor training seminar where the role of mentors in the student learning process will be emphasized.

Less than half of the students who participated in the fall 2011 study indicated a stronger connection with other students in math and/or engineering after completing the Calculus I course and participating in the mentor-led study sessions. Combined with a lack of respondents who indicated a desire to meet others as impetus for attending the study sessions, it does not appear that peer connections are a priority for students in this iteration of the study. Peer connections should be emphasized as an important component in the pursuit of engineering student success in all academic areas, particularly in calculus. There may be an opportunity to better facilitate peer interaction in the small groups setting, led by the engineering peer mentor, by considering group construction. Pairing mentors with a compatible group of mentees that share commonalities has been shown to be very beneficial [11]. In future semesters, we will strive to assign each group a mentor from their specific engineering discipline. In addition, special efforts will be made to match mentees of an underrepresented group with a mentor of a similar group.

Participating students felt more confident in their mathematical skills as well as in their abilities to succeed in engineering. Moreover, students made a connection between calculus and engineering. This was an important outcome for those who took part in this study, specifically from their interactions with upperclass-level engineering students, and participation in the study sessions. We anticipate expanding the data analysis portion of this project once the full study is complete, to include determination of correlation coefficients and some predictive analysis of students' likelihood to remain in engineering.

6 Conclusions

The use of engineering peer mentors is a key component in supporting students' comfort, confidence, and acquisition of the requisite calculus knowledge, skills, and abilities. Moreover, the program's goal is student persistence and improved institutional efforts concerning retention of STEM students. As consideration of the mentor-led study sessions continues, so does the pursuit of creating a framework for best practices of peer mentoring programs that may be used by other institutions.

7 References

[1] D.F. Whalen, M.C. Shelley III, "Academic Success for STEM and Non-STEM Majors," *Journal of STEM Education*, 2010, pp. 45-60.

[2] R. Hensel, J. Sigler , A. Lowery, "Breaking The Cycle Of Calculus Failure: Models of Early Math Intervention to Enhance Engineering Retention." *Proceedings of the American Society for Engineering Education Annual Conference & Exposition*, 2008.

[3] N. Klingbeil, R. Mercer, K. Rattan, M. Raymer, D. Reynolds, "Rethinking Engineering Mathematics Education: A Model for Increased Retention, Motivation and Success in Engineering," *Proceedings of the American Society for Engineering Education Annual Conference & Exposition*, June 2004, pp. 12169-12180.

[4] D. Sathianathan, P. Yaeger, S. Tavener, S. Armentrout, R. Marra, "Using Applied Engineering Problems in Calculus Classes to Promote Learning," *Proceedings of the Frontiers in Education Conference,* November 1999, pp. 12D5-14 – 12D5-18.

[5] K. Royal, & A. Tabor, "Theories of Student Success: Evaluating the Effectiveness of an Intervention Strategy," Presented at the annual meeting of the *Mid-Western Educational Research Association*, 2008.

[6] J. Terrion, D. Leonard, "A Taxonomy of the Characteristics of Student Peer Mentors in Higher Education: Findings From a Literature Review," *Mentoring & Tutoring*, 2007, pp. 149-164.

[7] S. Brainard, S. Laurich-Mcintyre, L. Carlin, "Retaining Female and Undergraduate Students in Engineering and Science: 1995 Annual Report to the Alfred P. Sloan Foundation," *Journal of Women and Minorities in Science and Engineering*, 1995, pp. 255-267.

[8] D. Farver, C. Gattis, "Development and Implementation of a Peer Mentoring Program for Women in Engineering," *Proceedings of the American Society for Engineering Education Annual Conference & Exposition*, Chicago, IL, 2006.

[9] A. Asghar, "Reciprocal Peer Coaching and its Use as a Formative Assessment Strategy for First-year Students." *Assessment & Evaluation in Higher Education*, 2010, pp. 403-417.

[10] *Comprehensive Assessment for Team-Member Effectiveness (CATME) Team-Maker*. (n.d.). Retrieved from https://engineering.purdue.edu/CATME

[11] B. Koehler, S. Matney, J. Lavelle, M. Robbins, "MENTOR: Motivating Engineers Through Organized Relationships", Proceedings of the American Society for Engineering Education Annual Conference & Exposition, Honolulu, HI, 2007.

Study on Operating System Designing in Computer-integrated Education

Liang Shi, Duoqian Miao
Tongji University, Shanghai, China

Abstract - *In order to enhance students' understanding of computer composition principles and encourage them to compile operating system and compiler with the 32-bit MIPS based instruction subset processor designed by themselves, this paper will introduce main process and issues of Linux transplanting and discuss on possible solutions for transplanting integrity instruction set, besides, analyze their merits and disadvantages. In addition, it will fully introduce modification and reservation methods for exceptional instructions through the research of Linux source code. It will also capsule how to modify GCC complier so as to provide solid theoretical ground for the follow-up practical experiment, by which similar theories based on Linux system transplanting can be further perfected. This work can be utilized as an instrumental reference for studies.*

Keywords: MIPS32, Operating System, Experimental Reform

1 Introduction

With the rapid development of computer science, communication technology and network technology, embedded system has become an indispensable part in our daily life. Not like Windows, Linux has become the first choice for the developers in the embedded system filed for its good performances, clear structure and its open-source code. Linux has support for X86, ARM, MIPS, MIPS64, Sun SPARC, POWER PC, Motorola 68K, IBM S/390, Alpha, IA64, cris, parisc, sh and some other architectures.

Computer is a subject combined with science and engineering. Theoretical knowledge of computer science and practical use of computer engineering supplement each other, which take a giant stride forward in the subject and even the human society. In order to change the current phenomenon that students paid too much attention on books and they lacked of practice by their own hands, this paper expect to make an embedded Linux in a processor based on 32-bit MIPS instruction set, which provide a new idea in the reform of experimental practice in subjects like the operating system and the compiler theory. [1]

The development platform in this paper is an achieved processors based on 32-bit MIPS instruction set in the former part of integrated computer experiment. For some reason in designing, the instruction set is not the standard MIPS instruction set but its subset. The processor should be the same in logic after transplanting. [2]

2 Cross-Compilation Environment Build for Linux and Kernel Configuration and Compilation

Linux build is mainly including the cross-compiling environment build, editing codes on the host computer, compiling of programs, files and libs link and remote debug from host computer by a debug agent. We put our program into the ROM of development board to accomplish the experiment.

It is not feasible and practical to build a development environment on the development board because of its limitation of hardware. So we should build two platforms before transplanting the system. One is the host PC as development platform. The other is running platform on the Development Board. The relationship of them is shown in figure 1.

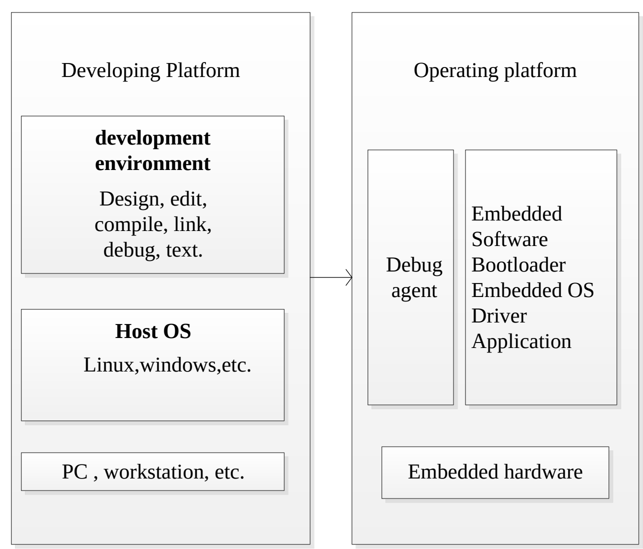


Figure 1 Software development platform

The Development process of software can be divided into three steps:

1 Source code writing.
2 Object modules building by cross compiling.
3 Link to modules and relative libs.

In order to generate binary codes which are suitable for running on the target platform, the codes should be compiled by the cross compiler between the platforms, which is different from developing software on a personal computer. In order to link the program codes with the correct data and libs from the memory in the target computer, we should use cross linker. Therefore, establishing a correct cross compiler is indispensable in the process of writing embedded software.

The main tools to establish a cross-compiler environment on MIPS instruction set including GCC, glibc, binutils and gdb as debugger.

The GNU Compiler Collection (GCC) is a compiler system produced by the GNU Project supporting various programming languages. GCC is a key component of the GNU tool chain.

Glibc is the link library and runtime of the program. For the reason that this link library and runtime is not based on development platform but the target platform, it should be compiled by the cross-compiler environment before being used.

Linux supports several popular computer architectures, which makes us choose the right configuration before cross compiling including processor architectures, file system types, board-level supports and device driver.[3]

3 Ways to Solve the Problem With Integrity MIPS Instruction Set

Due to the students' ability, the design of processor should not be too complicated in the former procedure of writing the processor in the integrated computer experiment. So we removed some less-frequently-used instructions and some complex instructions in coding. The whole processor is based on a subset of the standard MIPS instruction set. [4] The main problem in the transplantation is how to keep the logical equivalence and how to make sure the program will be executed correctly.

After the kernel is compiled, if the corresponding instruction is not found in the target instruction set, the processor will throw out a reserved instruction exception. There are two ways to solve this question:

1) Change the kernel codes
2) Change the compiler.

3.1 Change the kernel codes

While the processor is running, if the next instruction does not exist, the system will throw out a reserved instruction exception and jump into the corresponding interrupt handler. we can achieve these instruction logically by adding actual operations for these losing instructions in the interrupt handler. That's the principle and the procedure of how to change the kernel.

For the MIPS instruction set in the kernel, the reserved instruction exception initialization is located in the */arch/mips/kernel/traps.c*.[5] The function *asmlinkage void do_ri(struct pt_regs *regs)* is the function which handle the reserve instruction exception. Its main structure is as follows: [6]

```
asmlinkage void do_ri(struct pt_regs
*regs)
{
 unsigned int __user *epc = (unsigned int
__user *)exception_epc(regs);
 unsigned long old_epc = regs->cp0_epc;
 unsigned int opcode = 0;
 int status = -1;

 if (notify_die(DIE_RI, "RI Fault", regs,
0,   regs_to_trapnr(regs),   SIGILL)   ==
NOTIFY_STOP)
   return;

 die_if_kernel("Reserved  instruction  in
kernel code", regs);

 if (unlikely(compute_return_epc(regs)<0))
   return;

 if (unlikely(get_user(opcode, epc)<0))
   status = SIGSEGV;

 if (!cpu_has_llsc && status < 0)
   status = simulate_llsc(regs, opcode);

 if(status < 0)
   status = simulate_rdhwr(regs, opcode);

 if(status < 0)
   status = simulate_sync(regs, opcode);

 if(status < 0)
   status = SIGILL;

 if(unlikely(status > 0))
 {
   reg->cp0_epc = old_epc;
   force_sig(status, current);
 }
}
```

We can add our own functions into exception handler for unrealized instructions here, the code is structured as follows:

```
int simulate_otherOP(struct pt_regs *regp,
unsigned int inst)
{
 register unsigned long rs;
 register unsigned long rt;
 unsigned long va;
 unsigned long mem;
 unsigned int byte;
 rs = _GPR_STKOFFSET(_RS_(inst));
 rt = _GPR_STKOFFSET(_RS_(inst));
 va = rs + (unsigned
long)((short)_OFFSET_(inst));
 byte =va & 3;
 switch(_OP_(inst))
 {
  case 0x33:
   Return 1;
  default:
   return 0;
 }
}
```

_OP_ is a macro operation. Its function is to make out the foundation for the switch statement by fetch the op code from 32-bit machine code. In the different case statement, we add corresponding appropriate handler by the fetched op code to achieve these instructions that will caused reserved instruction exception.

For example, in MIPS instruction set, the op code 0x22 corresponds to the LWL instruction. We add the appropriate command processing codes in the case statement to calculate the offset of non-aligned instruction then return from the function. The codes are shown as below:

```
Case 0x22:
 mem = *(unsigned long*)(va-byte));
 mem = mem << (3 - byte) * 8;
 rt = (rt & ~(-1UL >> byte * 8) | mem;
 _GRP_STKOFFSET(_RT_(inst)) = rt;
break;
```

At last we add our custom handler into *do_ri()* function.

```
If(simulate_otherOP(reg, opcode))
{
 Compute_return_epc(regs) ;
}
```

After the custom function finished and retuned 1, the Linux system will invoke *compute_return_epc()* to assign the value back into pc register from epc register. If not, the system will continue processing other exception handler.

We can change the Linux source code based on C language directly in this way. It is simple to starter and easy to master. The equivalence can be easily ensured after being modified so that this method has high practical value. However, this method is based on the interrupt handler of reserved instruction exception. If the code contains a huge amount of unachieved instructions, the system performance will be reduced. This factor must be considered before the medthod is invoked.

3.2 Change the compiler

Changing the Linux source code to achieve the custom operation in reserved instruction exception handler may make the system have low efficiency in the practice. Therefore, modifying the GCC compiler directly may also have its impact.

The main compilation process of GCC is shown in Figure 2[7]:

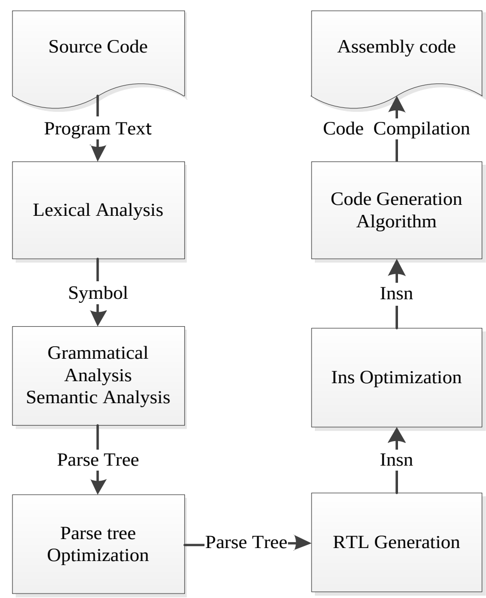


Figure 2 compilation process of GCC

4 Conclusions

During the instruction generation, we can change part of the GCC compiler source code files. While compiling, we can use achieved equivalent instructions sequences to instead of unachieved instructions which may be generated to solve this problem. The main GCC compile file includes inst-emit.c, inst-flags.h, inst-config.h, inst-code.h and so on. This method can solve the problem of system transplantation with the uncompleted instruction set in an efficient way. However, due to the high threshold and the efficiency of the GCC compiler, the experimental cycle of computer experiment will be extended greatly while the system efficiency is greatly improved. This section need to be further improved and followed up in the future work.

5 References

[1] Yang Sheng-Quan, ”Integrated Curriculum Design of Computer Systems”, 2008, Tsinghua University Press.

[2] MipsMIPS32™ Architecture for Programmers Volume II: The MIPS32™ Instruction Set,, MIPS Technologies, Inc.

[3] Karim Yaghmour, “Building Embedded Linux Systems”, O’REILLY, 2003-O4.

[4] Dominic Sweetman, Zhao Junliang, Zhang Fuxin, Tao Pin., “MIPS Processor Design Perspective”, 2005, BeiHang University Press.

[5] Ni Jili, “Analysis and Programming of the Linux Kernel”, 2005 , Tsinghua University Press.

[6] Li Shanping, Liu Wenfeng, Wang Huanlong, “Linux and Embedded Systems”, 2003, Tsinghua University Press.

[7] Arthur Griffith, Hu Enhua, “GCC Technical Reference”, 2004, Tsinghua University Press.

The use of ICT in Education: *a comparison of traditional pedagogy and emerging pedagogy enabled by ICT's*

Mbodila, Munienge[1] and Kikunga Muhandji[2]

1. Science Foundation Unit - Computer Science, University of Venda, Private Bag X5050, Thohoyandou 0950, South Africa.
2. Computer Science, University of Venda, Private Bag X5050, Thohoyandou 0950, South Africa

(*munienge.mbodila@univen.ac.za , bkikunga@hotmail.com)*

Abstract

Improving excellence in education is a critical issue; mainly at this time of everything is becoming globalized. The traditional pedagogy approach to teaching and learning, as ancient as formal teaching itself, involves the directed flow of information from teacher as sage to student as container. This method is characteristically based on pre-packaged learning materials, fixed deadlines, assessment tasks and criteria are defined by teachers. But research has shown that the appropriate use of ICTs can catalyze the paradigmatic shift in both content and pedagogy that is the heart of education reform in the 21st century and promote problem based learning. This paper gives an overview of the traditional ways of teaching and some of its limitations; also discussed the emerging ways of teaching enabled by the use of Information Communication Technology (ICTs), furthermore organizes a variety of approaches found in ICTs uses in Education.

Keywords: ICT's, traditional pedagogy, pedagogy enabled by ICT's, learned-centered

1. Introduction

Education is an engine for the development and improvement of any society. It does not just impart knowledge and skills, but it is also responsible for building human capital which breeds, drives and sets technological innovation and economic growth (*Innovative Methods of Teaching, Dr. Damodharan V. S. ACCA, AICWA and Mr. Rengarajan.V AICWA).* The strategy behind our teaching methodology and the focus of our teaching is called pedagogy. Pedagogy is also about learning; one cannot teach without being a learner. To be able to comprehend what is involved in the process of being a learner is part of pedagogy.

Today, in the 21st century understanding is more important than ever. That is why it is vital for educators to use methods or techniques in teaching that will enable learners to use their knowledge efficiently to solve problem in their daily lives. In this paper we compared two techniques that are used in education for teaching and learning; we discussed their effectiveness approach, innovation, we also identified the limitation and significance of each pedagogy for quality teaching and success in the process of teaching and learning in 21st century.

2. What is Traditional Pedagogy?

Before starting to explain what traditional pedagogy is, it will be important to define the term pedagogy itself. Pedagogy is the technique we use to teach, the approaches behind our teaching and the centre of our teaching. When we talk about teaching and learning outcomes, there are techniques that must be applied in order to reach those outcomes. The pedagogy used in teaching and learning will determine the type of students that will be produced in the institutions. Pedagogy is also about learning; which means one cannot teach without being a learner. To be able to understand what is involved in the process of being a learner is part of pedagogy.

When we talk about traditional pedagogy approach to teaching and learning, as ancient as formal, it involves the directed flow of information from teacher as sage to student as receiver. The traditional pedagogy can be also defined as a pre-technology education context in which the teacher is the sender or the source, the educational material is the information or message, and the student is the receiver of the information. Typically it is based on pre-packaged learning materials, fixed deadlines, assessment tasks and criteria determined by teachers. In terms of the delivery medium, the educator can deliver the message via the "chalk-and- talk, marker-and-white board" method and overhead projector (OHP) transparencies (*Innovative Methods of Teaching, Dr. Damodharan V. S. ACCA, AICWA and Mr. Rengarajan.V AICWA*). The method has its foundations embedded in the behavioral learning perspective (*Skinner, 1938*) and it is a fashionable practice, which has been used for decades as an educational approach in all institutions of teaching and learning. In this type of teaching methods the teacher as the monopoly of prescribing the activities, students are there only to listen and follow what is asked for them to do in the classroom. The learners depend on the teacher who directs what, when, how a subject is learned and tests what has been learned. In this method the learner's skill, knowledge and practice is of little value. Therefore teaching methods are educational and people learn what society expects from them. So the curriculum is homogeneous. It has been found by many teachers and students in most institutions that the conventional lecture approach (traditional pedagogy) in classroom is of limited effectiveness in both teaching and learning.

3 Limitations of Traditional Pedagogy

There are many limitations of using the traditional pedagogy in teaching and learning. This technique of teaching is a one way flow of information in which the teacher often continuously talk for an hour or more expecting that when he asks a question, the students will able to reproduce the same thing that he was talking about. Below are some of the limitations indentifying for traditional pedagogy:

- Teaching and learning are concentrated on theoretical method rather than practical aspects
- There is not enough interaction with students in classroom
- There is less activities in the classroom, teacher decide what to do, when and how
- No creativity, learners reproduced what the teacher told them over and over again
- Less integration for students, the teacher is the only individual who talks and gives command
- More emphasis has been given on theory without any practice and real life time situation

In 21st century, ICTs has offered different techniques that are available to everybody to use. Therefore traditional pedagogy has less impact in teaching and learning. In the following paragraph explained the implication of ICT in education and diverse approaches that ICTs uses to make the teaching and learning environment to be a learner centered environment and encourage problem based learning.

4 The Significance of ICTs in Education

Many researchers argue that the use of new pedagogy enabled by ICTs in teaching and learning is indispensable for providing opportunities for students to learn; to operate in today's e-society. As Yelland (2001) argued, the traditional educational environments do not appear to be suitable for preparing learners to function or be productive in the workplaces of today's society. She also claimed that organizations that do not fit in the use of new technologies provided by ICT's cannot significantly claim to prepare their students for life in the 21st century. This argument was supported by Grimus (2000), who pointed out that "by teaching ICT skills students are prepared to face future development based on proper understanding" (p.362).

Many researches and theorists emphasize that the use of computers can help students to become knowledgeable, decrease the amount of direct instruction given to them, and give teachers an opportunity to help those students with particular needs (*Iding, Crosby, & Speited, 2002; shamatha, Peressini, & Meymaris 2004; Romeo, 2006*). While new technologies can help teachers enhance their pedagogies practice, they can also assist students in their learning process. According to Grabe (2007), technologies can play a crucial role in student skills, motivation, and knowledge. In their argument, they also claim that ICTs can be used to present information to students and help them complete learning tasks. A very important aspect of learning enhanced by using ICTs will facilitate a more smoothed and compound view of abstract concepts. In this it is significant to understand different approaches that ICTs use in education to fulfill the teaching and learning outcomes.

5 Different Approaches of the use of ICTs in Education

It is true that teaching 21st century learners requires strong pedagogy. Traditional Pedagogy approaches have resulted in a divergence between what is taught to the students and what the industry needs. As such, many institutions are moving towards problem based learning as a solution to producing graduates who are creative; think critically and analytically, to solve problems. The pedagogy enabled by ICT provides problem based learning and enable students to be independent, have a critical thinking. It is obvious that by the use of ICT's, instructors will develop strategies that will promote deep learning and change the learning environment into the learner-centered environment. The impact of ICTs on learning can be approached in various ways to meet the need of learners. Differently from traditional pedagogy, that has one particular way of teaching (direct flow of information from teacher to the students) ICT's offer diversities of models in teaching and learning.

Diverse types of ICTs uses can be visualized as: computer assisted learning, web-learning, computer-classes, online training, distance education, visualization software, eLearning, virtual learning, digital training, etc. All these techniques used by ICTs shift in the role of a teacher to that of a facilitator. The existence of ICTs can enable teachers to transform their teaching practices. Research has shown that the use of different approaches offered by ICTs will transform the learning environment into the one that is learner-centered and promote deep learning. The following are example of some approaches enabled by the use of ICTs in teaching and learning process.

5.1 Active Teaching and Learning Approaches

The use of ICTs offers a platform for student to question, investigation and construction of new information in the process of teaching and learning. This is one aspect which is different from the traditional pedagogy in which learners learn as they practice and, whenever appropriate, work on real-life problem in depth. The teaching and learning process become less abstract but more relevant to the learner's life.

5.2 Collaborative Teaching and Learning Approach

The use of ICTs creates collaboration, interaction among students and teachers regardless of where they are. This limitation of traditional pedagogy has been overcome by the use of ICTs. Beside from modeling real-world interactions, ICT-supported learning provides learners with the opportunity to work with people of different cultures, different background. This has helping by improving learner's teaming and communicative skills as well as their global awareness.

5.3 Creative Teaching and Learning Approach

The use of ICTs promotes the manipulation of existing information and creation of real-world products rather than the regulation of received information, which is different from reproductive learning used in traditional pedagogy way of teaching.

5.4 Integrative Teaching and Learning Approach

ICTs improve learning by integrative approach to teaching / learning. This eliminates the artificial separation between the different disciplines and between theory and practice that characterizes the traditional pedagogy approach.

5.5 Evaluation Teaching and Learning Approach

Unlike traditional pedagogy, ICTs recognizes many different learning pathway and different articulations of knowledge as said above that ICT does not focus in only one type of teaching or learning approach like traditional pedagogy. The pedagogy enabled by ICTs allows learners to explore and discover rather purely pay attention and keep in mind.

6 Overview of Traditional and Emerging Pedagogy Enabled by ICTs

Table 1: Source ICT in Education by Victoria L.Tinio

Traditional pedagogy	Emerging pedagogy enabled by ICTs
• Activities prescribed by teacher • Individual • Apply known solution to problems • No link between theory and practice • Teacher-directed • Summative	• Activities determined by learners • Working in teams • Find new solutions to problems • Integrating theory and practice • Student-directed • Diagnostic

The above table gives an overview of traditional pedagogy and emerging pedagogy enabled by ICTs, it is obvious that their aspects differ from one another in term of promoting good teaching and learning environment. The table gives a summary of these two pedagogy in the classroom; this also defines the type of environment that each one of them can create in process of teaching and learning. Therefore it is essential for instructors in the 21st century to make use of pedagogy that will promote student autonomy and pay careful attention to the knowledge, skills, attitudes, and beliefs that learners bring with them to the classroom.

7 Conclusions

The researchers believe hat the core objective of teaching is passing on the information or knowledge to the brains of the students. To achieve this there are many strategies behind the teaching called pedagogy that can be use during this process. This paper gives a comparison of traditional pedagogy and the emerging pedagogy enabled by the use of ICTs in education. The paper also highlighted some limitations of the use of traditional pedagogy in teaching / learning and the significance of using emerging pedagogy enabled by ICT in education. As teaching and learning shifts from the teacher-centered model to a learner-centered model, the teacher becomes less the one and only voice authority but more the facilitator, mentor and coach; from stage on stage to guide on the side. As such, many institutions in the 21st century are moving towards new techniques of teaching enhanced by ICTs to promote problem based learning as a solution to settings where the teacher is the only individual who talks and gives command and less integration for students.

8 References

1. Making Sense of Teaching Methods in Computing Education, Kris D. Powers

2. Innovative Methods of Teaching, Dr. Damodharan V. S. ACCA, AICWA and Mr. Rengarajan.V AICWA

3. Boud, D. & Feletti, G. (1999). The Challenge of Problem-Based Learning, (2nd Ed.), London: Kogan Page.

4. Agnew, P. W., Kellerman, A. S. & Meyer, J. (1996). Multimedia in the Classroom, Boston: Allyn and Bacon

5. Anderson, P. (2007) "What is Web 2.0? Ideas, technologies and implications for education. JISC Technology and Standards Watch".

6. New technologies for teaching and learning: Challenges for higher learning institutions in developing country, International Journal of Education and Development using Information and Communication Technology (IJEDICT), 2007, Vol. 3, Issue 2, pp. 57-67

7. Teacher perspectives on integrating ICT into subject teaching: Commitment, constraints, caution and change, Sara Hennessy, Kenneth Ruthven and Sue Brindley

8. Making Sense of Teaching Methods in Computing Education, Kris D. Powers and Daniel T. Powers

CPU Design for Computer Integrated Experiment

Shan Lu, Guangyao Li, Yijianan Wang
CEIE, Tongji University, Shanghai, China

Abstract - *Considering the necessity and difficulty of designing a CPU for students, we simplify the MIPS instruction set for a MIPS architecture processor. Besides this, data path is given and all internal modules are compiled such as ALU module, controller module, bus module and so on. After testing and validation, the difficulty and time consumption of designing such a CPU are proved possible for students. As a result, a suit of experiment platforms based on CPU is born. It not only provides students a chance to have their own processor, but also provides us with a platform for further experiment, such as OS and fundamentals of compilation.*

Key words: MIPS, processor, integrated experiment

1 Introduction

Nowadays, most students whose major are CS have little understanding of the principle of CPU works, and have little practice. So these students can hardly grasp the main point of principle computer organization. To solve this problem, a simple student-oriented CPU is born. This design is used to help students to learn more about the computer configuration in practice. What's more, this CPU also provides a platform for the following course of OS and fundamentals of compilation.

Thanks to the open instruction model, MIPS instruction set[1] is chosen. In this model, subscribers are encouraged to design your own CPU in accordance with their own demands. Besides this, MIPS belongs to the RISC architecture, and the instructions are simple and few. So it is a good idea to choose MIPS instruction to structure our CPU. It is easy for students to see the achievement in a short term.

Verilog[2] is chosen to design this CPU because it is similar to C, and majority of students have C programming foundation. What's more, there are enough choices for you to design, such as ALU module, controller module, I/O module[3], storage module and so on. In this article, the designs and verification of all modules are involved. As a result, this CPU design has achieved the expected effect.

2 The Design of MIPS Processor

2.1 MIPS Instruction Set

According to Harvard structure, data are stored separately from instruction. Figure 1 shows the five steps of execute a MIPS instruction in a single cycle. The five steps are IF, ID, EX, MEM, WB. (1) IF step, instruction address is taken from PC Register(PC_Reg) to identify the current execution instruction and sent to Instruction Memory(Instr_Mem) to get the binary code of the instruction through Address Bus. (2) ID step, instruction code is decoded by Controller to get the control signal. According to the different instruction type, the corresponding register value is obtained from the Register File(Reg File). (3) EX step, the operand and the control signal are sent to the ALU for arithmetical or logical operation or the BS module for shifting. (4) MEM step, according to different MIPS instruction, the ALU result will be sent to the bus controller for arbitration. The bus controller will decide the destination of data such as Data Memory(Data_Mem) or GPIO. GPIO module realizes the responsibility of cache to solve the speed relation between internal unit and peripheral unit. (5) WB, according to the control signal, write-back mux module will decide whether the data writes back to the Register File(Reg_File).

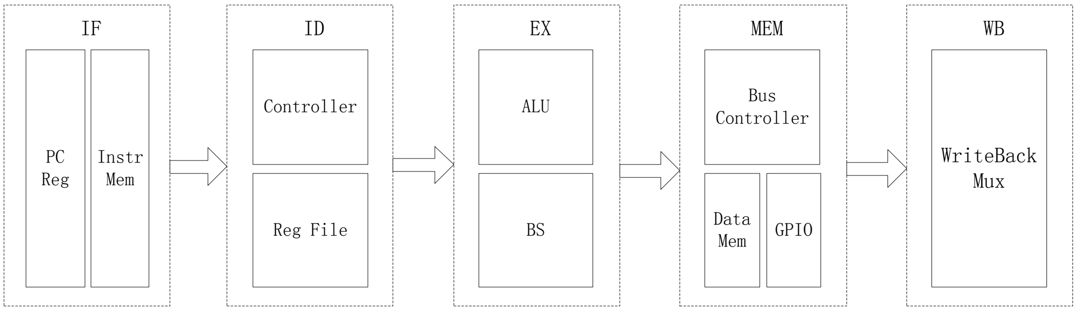


Figure 1 process and modules of MIPS

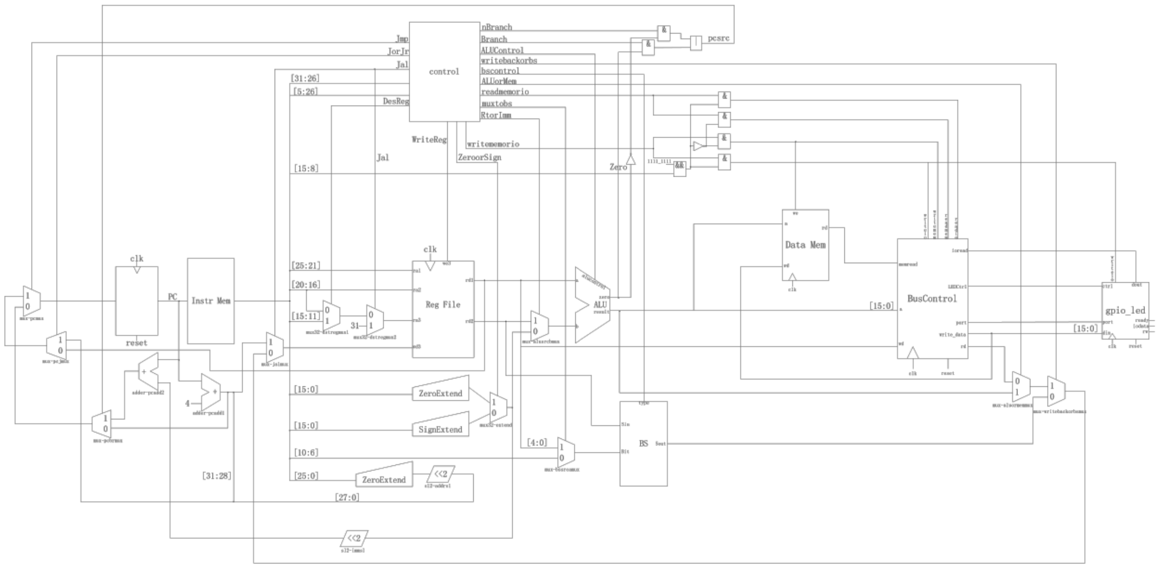


Figure 2 MIPS processor data path

The 30 most common used instructions as shown in Table 1 include the arithmetical operation, logical operation, branch, store instruction and so on. This set meets all the demands we need in the following design. MIPS instruction sets include three types: R-type, I-type and J-type. (1) R-type means register instructions. Two operands are taken from Register File, and the result is also sent to the Register File. (2) I-type means immediate instructions. These instructions fetch 16-bit immediate as an operand. (3) J-type means jump instructions. These instructions fetch 26-bit immediate as the destination address which will be stored into PC Register.

2.2 MIPS CPU Design

2.2.1 Data Path Design

After analyzing the MIPS instruction format and its classification, referring to the relevant CPU design[4, 5] and the MIPS modules defined in Figure 1, the data path of MIPS processor is designed as Figure 2. Pin information and wiring information is designed as well according to the statement in section 2.1.

2.2.2 ALU design

ALU module typically handles logical and arithmetical operation. The operation between operand A and B is controlled by ALU control signal(alucontrol). Afterwards ALU result and zero flag can be fetched from the output pins. Figure 3 shows the pin signal and Table 2 shows the detailed information of pins.

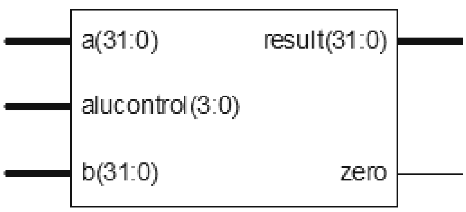


Figure 3 ALU pin signal

Table 1 MIPS instruction set and its format

Mnemonic Symbol	Format						Sample
Bit #	**31..26**	**25..21**	**20..16**	**15..11**	**10..6**	**5..0**	
R-type	**op**	**rs**	**rt**	**rd**	**shamt**	**func**	
add	000000	rs	rt	rd	0	100000	add $1,$2,$3
addu	000000	rs	rt	rd	0	100001	addu $1,$2,$3
sub	000000	rs	rt	rd	0	100010	sub $1,$2,$3
subu	000000	rs	rt	rd	0	100011	subu $1,$2,$3
and	000000	rs	rt	rd	0	100100	and $1,$2,$3
or	000000	rs	rt	rd	0	100101	or $1,$2,$3
xor	000000	rs	rt	rd	0	100110	xor $1,$2,$3
nor	000000	rs	rt	rd	0	100111	nor $1,$2,$3
slt	000000	rs	rt	rd	0	101010	slt $1,$2,$3
sltu	000000	rs	rt	rd	0	101011	sltu $1,$2,$3
sll	000000	0	rt	rd	shamt	000000	sll $1,$2,10
srl	000000	0	rt	rd	shamt	000010	srl $1,$2,10
sra	000000	0	rt	rd	shamt	000011	sra $1,$2,10
sllv	000000	rs	rt	rd	0	000100	sllv $1,$2,$3
srlv	000000	rs	rt	rd	0	000110	srlv $1,$2,$3
srav	000000	rs	rt	rd	0	000111	srav $1,$2,$3
jr	000000	rs	0	0	0	001000	jr $31
Bit #	**31..26**	**25..21**	**20..16**	**15..0**			
I-type	**op**	**rs**	**rt**	**immediate**			
addi	001000	rs	rt	immediate			addi $1,$2,100
addiu	001001	rs	rt	immediate			addiu $1,$2,100
andi	001100	rs	rt	immediate			andi $1,$2,10
ori	001101	rs	rt	immediate			andi $1,$2,10
xori	001110	rs	rt	immediate			andi $1,$2,10
lw	100011	rs	rt	immediate			lw $1,10($2)
sw	101011	rs	rt	immediate			sw $1,10($2)
beq	000100	rs	rt	immediate			beq $1,$2,10
bne	000101	rs	rt	immediate			bne $1,$2,10
slti	001010	rs	rt	immediate			slti $1,$2,10
sltiu	001011	rs	rt	immediate			sltiu $1,$2,10
Bit #	**31..26**	**25..0**					
J-type	**op**	**address**					
j	000010	address					j 10000
jal	000011	address					jal 10000

Table 2 ALU pin signal and function

pin name	pin meaning
a(31:0)	operand a of arithmetical or logical operation
b(31:0)	operand b of arithmetical or logical operation
alucontrol(3:0)	control signal, operation type
result(31:0)	ALU result
zero	zero flag, if it is effective, ALU result is zero

2.2.3 Controller Design

During the five steps of implementing MIPS instructions, signals from controller control the data flow in EX step, Mem step and WB step. Controller receives the machine code from instruction register, and then decodes it into control signal to control others modules, such as ALU module, BS module and so on. Figure 4 shows the pin signal and Table 3 shows the detailed information of pins.

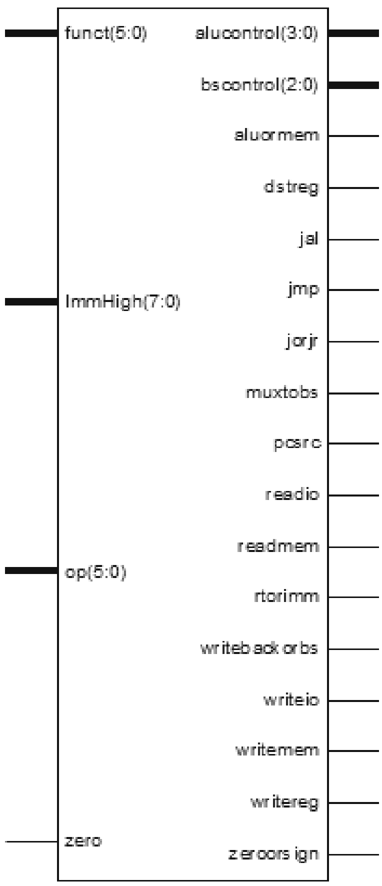


Figure 4 controller pin signal

Table 3 controller pin signal and function

pin name	pin meaning
funct(5:0)	funct filed of instruction
ImmHigh(7:0)	data bus, controller get the high 8-bit of immediate to identify the data direction, to I/O or memory
zero	zero flag
alucontrol(3:0)	ALU control signal
bscontrol(2:0)	BS control signal
aluormem	data strobe signal, if it is effective, ALU result is sent to register, else data from memory is sent to register
dstreg	address strobe signal, if it is effective, $Instr_{20..16}$ are set as destination register address, else 15-11 bits of instructions are set as destination register address
jal	data or address strobe signal, when signal is effective, the value of PC+4 is sent to the NO.31 register
jmp	PC strobe signal, when signal is effective, the jump location is sent to PC register
jorjr	PC strobe signal, when signal is effective, the jump location is sent to PC register
muxtobs	bit strobe signal, if it is effective, the bit is the value which is stored in the register, which address is 25-21 bits of instructions, else the immediate in the 10-6 bits of instruction is sent to BS as the bit to be shifted
pcsrc	branch signal
readio	I/O read signal
readmem	memory read signal
rtorimm	data strobe signal, if it is effective, ALU operand is the value from register, else 15-0 bits of instruction
writebackorbs	data strobe signal, if it is effective, BS result is written back
writeio	I/O write signal
writemem	memory write signal
writereg	register write signal
zeroorsign	extend control signal, if it is effective, the 15-0 bits will extend by zero, else the 15-0 bits will extend by symbol

2.2.4 BS Module Design

This CPU is a single cycle processor. In this type of process, shift instruction is demanded to complete in a single cycle. Nowadays, there are three popular encoding. After analyzing these decoding, full-encoding[6] is chosen to realize the BS module. For 32-bit barrel shifter, it demands 5-bit control signal to complete logical shift, arithmetical shift and cyclic shift. Figure 5 shows the pin signal and Table 4 shows the detailed information of pins.

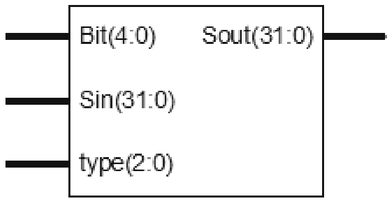


Figure 5 barrel shifter pin signal

Table 4 barrel shifter pin signal and function

pin name	pin meaning
Bit（4:0）	bit to shift
Sin（31:0）	operand
type（2:0）	type
Sout（31:0）	BS result

2.2.5 Bus Module

Bus module provides the unified writing and reading management of I/O and memory. The data can be distributed and I/O selected signal and port address can be given by the I/O or memory read-write signal. Figure 6 shows the pin signal and Table 5 shows the detailed information of pins.

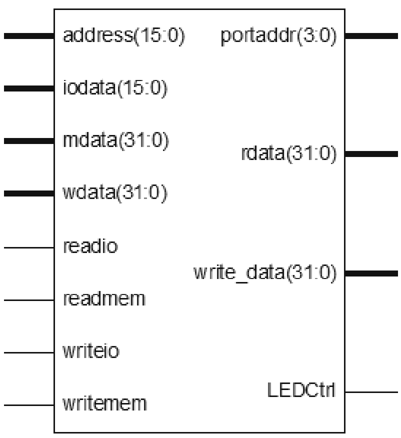


Figure 6 bus module pin signal

Table 5 bus module pin signal and function

pin name	pin meaning
address(15:0)	address bus, the low 16-bit of ALU result is set as the address of memory or I/O
iodate(15:0)	data bus, the data from I/O is sent to bus controller
mdata(31:0)	data bus, data from memory is sent to bus controller
wdata(31:0)	data bus, data from register is sent to bus controller
readio	I/O read signal
readmem	memory read signal
writeio	I/O write signal
writemem	memory write signal
portaddr(3:0)	port address, the low 4-bit of I/O address is set as port address
rdata(31:0)	data bus, I/O or memory read signal decide to read I/O data or memory data
write_data(31:0)	data bus, I/O or memory write signal decide to write data to I/O or memory
LEDCtrl	LED strobe signal

2.2.6 Register Module

Register module(Figure 7) consists of 32 registers. Among those registers, No.31 register is used to store return address, the other registers are general-purpose register. During the five steps of implementing MIPS instructions, register module typically handles EX step and WB step. During EX step, operand is fetched from register and sent to ALU module. During WB step, the data is written to the register by the write register signal.

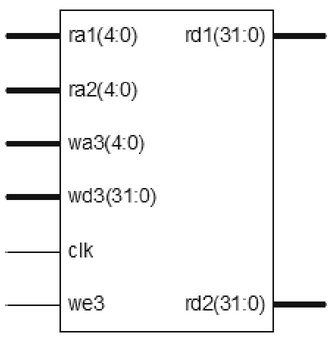


Figure 7 register module pin signal

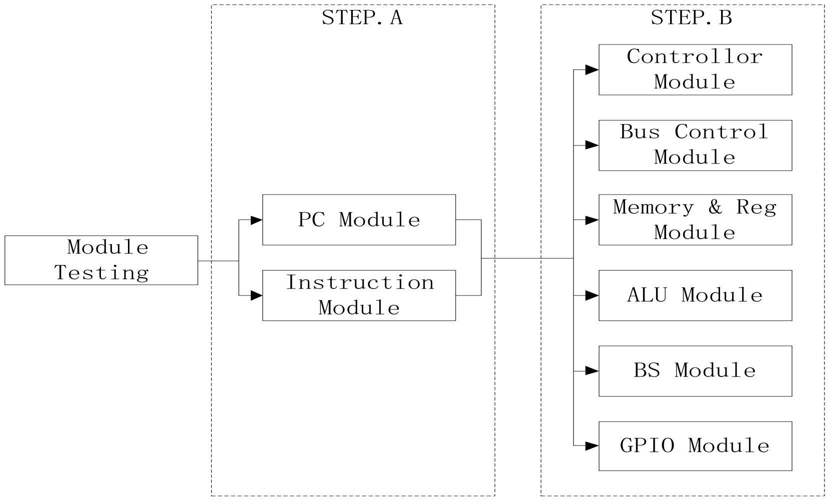


Figure 8 module testing process

```
[00400024] 20020005  addi $2, $0, 5          ; 2: addi $2,$0,5
[00400028] 2003fff9  addi $3, $0, -7         ; 3: addi $3,$0,-7
[0040002c] 14430002  bne $2, $3, 8 [temp1-0x0040002c]
[00400030] 2404000d  addiu $4, $0, 13        ; 5: addiu$4,$0,13
[00400034] 30450007  andi $5, $2, 7          ; 7: andi $5,$2,7
[00400038] 10430002  beq $2, $3, 8 [temp2-0x00400038]
[0040003c] 20460002  addi $6, $2, 2          ; 9: addi $6,$2,2
[00400040] 20670004  addi $7, $3, 4          ; 11: addi $7,$3,4
[00400044] 20080000  addi $8, $0, 0          ; 12: addi $8,$0,0
[00400048] 0c100015  jal 0x00400054 [temp3]  ; 13: jal temp3
[0040004c] 2009000b  addi $9, $0, 11         ; 14: addi $9,$0,11
[00400050] 08100018  j 0x00400060 [temp5]    ; 15: j temp5
[00400054] 08100017  j 0x0040005c [temp4]    ; 17: j temp4
[00400058] 200a0016  addi $10, $0, 22        ; 18: addi $10,$0,22
[0040005c] 03e00008  jr $31                  ; 20: jr $31
[00400060] 200b0021  addi $11, $0, 33        ; 22: addi $11,$0,33
```

Figure 9 TestBench code

Table 6 register module pin signal and function

pin name	pin meaning
ra1(4:0)	address bus, the 25-21 bits of instruction is send to register as register address
ra2(4:0)	address bus, the 20-16 bits of instruction is send to register as register address
wa3(4:0)	address bus, the data written back is sent to register as register address
wd3(31:0)	data bus, the data written back is sent to register as register data
clk	clock signal
we3	register write signal
rd1(31:0)	data bus, the data is read from registers according to the ra1 address
rd2(31:0)	data bus, the data is read from registers according to the ra2 address

3 Testing and Verification

RTL simulation consists of two parts, module testing and system testing. This simulation needs complete testbench, and output response for observation. You can determine whether the design reach the expected function in accordance with these information

3.1 Module Testing

This testing is divided into two parts as shown in Figure 8. Step A verifies the correctness of PC to fetch the current instruction and next instruction. That is, to verify whether the MIPS processor fetches the machine code which PC points to, correctly parses the func filed, op filed and ImmHigh field of instruction and sends the binary code into right pin. Step B use QtSpim to generate the machine code of test program. The test programs are required to cover the routine testing and marginal testing.

All test programs are run in Modelsim. You can determine whether the design reach the desired function with the help of output pin signal.

3.2 System Testing

Write testbench to test the MIPS processor by black box testing. The test which involves all instructions includes logical operation test, arithmetical operation test, GPIO test and so on.

Figure 9 shows the testbench code of jump instruction which contains all J-type instruction, jr instruction and parts of I-type instruction such as beq instruction. With the help of instruction execute order and the registers data generated by Modelsim, we can determine whether the design reach the desired function or not.

4 Conclusion

A student-oriented MIPS processor is designed to recognize and execute MIPS instruction set correctly. Besides this, as supplements, interface information and function explanation are given. Tests show that the difficulty and time factor are considered to be possible for students to design such a CPU. What's more, the implement of this MIPS processor provide the computer integrated experiment a visual testbed. In the process of this design, students can learn more about the computer configuration in practice. Up till now, we have completed the design and implementation of MIPS processor. The future work is to improve and enlarge the processor, and to program some software such as BIOS, Mini OS or Compiler beyond it.

5 Reference

[1] MipsMIPS32™ Architecture For Programmers Volume II: The MIPS32™ Instruction Set, , MIPS Technologies, Inc.

[2] Yuwen Xia, Verilog Digital System Design Tutorial. 2008, Beijing University of Aeronautics and Astronautics Press.

[3] XILINX, XPS General Purpose Input/Output(GPIO) (v2.00a).

[4] David Money Harris, Sarah L. Harris, Digital Design and Computer Architecture.

[5] Quansheng Yang, The Computer System Comprehensive Design Course. 2008, Tsinghua University Press.

[6] Field Programmable Gate Array. 1999, Atmel Corporation.

An Approach for Making a Class Seem Like the Real World

A. Chen
Computer Science and Information Systems, Minnesota State University Moorhead, Moorhead, Minnesota, USA

Abstract— *Today, many students may perceive a disconnect between the classroom and life outside academia. We outline an approach to class construction which holds promise for eliminating this disconnect in the students' minds. This approach was used to teach a game development class in which students from various disciplines enrolled. We present the rationale for how the class was structured. We also present some details regarding the planning process, including guiding principles which were found to be beneficial. This should be sufficient information for those who wish to apply the general idea behind this approach to other similar interdisciplinary project-based classes, without being hindered by the exact details of how this class had been conducted.*

Keywords: Team, Project, Education, Collaborative Learning, Pedagogy

1. Introduction

Many students today may find a disconnect between classroom activities and their perceptions of life outside of academia. In this paper, we outline an approach to class construction that has been tried once and which we believe holds promise for helping eliminate this apparent disconnect in the minds of the students. The approach described in this paper was used to teach a game development class in which students from various disciplines, including computing, graphics, music, film, psychology, history, art, and others, were encouraged to enroll. This paper will consist mainly of rationale for how the class was structured, along with some of the planning process. This should be sufficient information for those who wish to apply this approach to other similar interdisciplinary project-based classes to consider doing so.

2. Classroom Disconnect

In the experience of the author, many students today find a disconnect between classroom activities and their perceptions of life outside of academia. For one example source of information about life outside of academia, let us consider select salient points of so-called "reality television shows":

- Competition, including winning and losing, is regular and public
- The number of survivors is whittled down over time
- There is a big prize at the end

While it may be arguable regarding the extent to which individual "reality television shows" impact the views and opinions regarding students towards that particular aspect of "reality" that is portrayed (such as surviving on an island, being a teenage mother, or becoming a popular vocalist), the above common characteristics are not often challenged or critiqued, suggesting that students see life as consisting of these. The coverage of the news of corporate mergers and political candidate selection and election also mirrors a similar process, whether it being consolidation in business markets that have had recent successes of entrepreneurial endeavors or the choice of presidential candidates in primary elections.

Thus, it seemed relevant to try to design a class in which these attributes were present.

3. Class Design

This was tried in a game development class, but could be applicable to any other team production class. To ensure the students were properly motivated, both student egos and grades were used as motivating factors.

3.1 Class Mechanics

1) Students each pitched a game concept to the class.
2) The class as a whole voted.
3) Those with higher scores had their concept survive, and those with lower scores became available for selection as partners to work with on the surviving game concepts.
4) Thus, one round consisted of voting, elimination, and selection of additional partners.
5) This process repeated through several rounds, with each team on average gaining one additional partner per round.
6) The number of rounds that students were on a game concept that survived to the next round was the single largest part (although not a majority) of their grade in the class.

3.2 Class Design Criteria Verification

Through the class mechanics, competition, including winning and losing, was regular and public (voting and hiring alternated weeks). The public aspect of this contributed to their ego investment.

The voting rounds whittled down the number of survivors over time.

The big prize at the end consisted of having the game that their team finished be played in class by all of their classmates. The entire class also voted on how good any final game was, and this vote contributed to the grade. If a team had just done a good job of talking, but failed to deliver a quality game, this would be reflected in the vote and in the grade. More likely than the grade, the potential ego damage from classmates making potentially negative remarks about that game during that (final and only) playing session would be a powerful motivating factor as well.

4. Planning Process

Many iterations of this concept were considered. Among them were variations such as having students that lost not be hired, but instead have to write a paper, or alternating the competitive rounds with cooperative rounds (after a hiring and before a voting) in which teams would be required to present to the class "lessons learned" or similar knowledge gained. Those variations were not chosen. This variation was decided upon instead of those because it satisfied the following criteria:

- It kept true to the previously-mentioned salient properties of "reality television shows".
- It kept all students involved in some project (possibly shifting projects throughout time).
- It ensured that as projects developed, the resources allocated to them could increase (through more hires).
- It had an embedded and simple grading component (number of rounds a student is on a team that survives) which did not require extensive amount of instructor work or justification (since it was the student vote that decided it, any justification would have to be done by the students talking to each other).
- Once the students grasped this unusual classroom approach, they didn't have to keep on re-adjusting to something else that was different.

5. Guiding Principles

Furthermore, there were some guiding principles which the instructor held to which simplified the conduct of the class. First among them was that it was the students, not the instructor, who decided which teams survived and which teams did not. In no way did the instructor hold any right to veto or otherwise over-rule the results of the voting. The sole area of discretion that the instructor claimed was that of determining the cutoff point regarding how high of a score was necessary to survive, and this was done primarily because, as different teams had different sizes, and the hiring round was only to have enough people for every team to make at most one hire, the number of teams that could be cut was dependent upon how many people were in those teams that would be cut.

Second amongst these guiding principles was humility on the part of the instructor. Acknowledgement that the instructor was lacking in professional experience in actual commercial game development was a way of both legitimizing to the students that the instructor was reasonable in letting them vote, as well as in ensuring that the students understood that in real life, as in "reality television shows", market forces (or audience votes) hold far more sway than that of a single instructor (or boss or other superior).

Thirdly amongst these guiding principles was that of working with student concerns rather than against them. This was made manifest in having a portion of the grade being student evaluations of each other, thus providing grade incentive to be valuable team members rather than free-riders that loaf off of the team's contributions.

Fourthly amongst these guiding principles is the realization that this is a labor market and so involves everything appropriate for a labor market, including firing, quitting, teams giving up (analogous to going bankrupt), and so on.

6. Research Directions

The process of learning from and articulating all of the lessons of this first teaching of this class is a long one. While there are many possible opportunities for improvement, there are also many good justifications for collecting data before and after the improvement so as to have a data-based record of the result of these experiments in class structuring. Weighing the pros and cons of improvement based on gut instinct versus the careful, slow, and methodical collection of data with regard to the impact on the quality of education for the students and the potential impact on the instructor's career is a delicate matter. Other work ongoing includes a more detailed overview of the procedure followed in the class, as well as the situation of this work in a variety of other different contexts, including as an example of interdisciplinary work and as an example of the application of concepts from the field of critical game studies.

Situating this work as being an example of application of concepts from the field of critical game studies requires some explanation, but through this, some much deeper theoretical results can be obtained (in the sense of pedagogical theory).

In attempting to think of the structure of this class in a larger pedagogical framework, a reasonable analogue that comes to mind, from the realm of critical game studies, is that of the narratology versus ludology debate. In narratological terms, games are to be seen as consisting primarily of a story, which makes them amenable to the same sort of analysis as other media that tell a story, such as film, literature, and so on. In ludological terms, games are to be seen as consisting primarily of play and the elements of play (ludic elements), and as such, a focus on the elements that contribute to the interactive feedback loop that facilitates play becomes central.

In many ways, classes can be seen as games, with the grade in the class being the score that one receives in the game.

The approach described in this paper can be seen as modeling the ludological approach, since there is no predefined story for the students to progress through, and they could potentially have such wildly divergent stories as having their original concept survive until the end versus consistently being hired by losing teams throughout the semester, and thus is clearly not narratological in nature (furthermore, the nature of the choices made in the production of the game also contributes to the essence of the story that unfolds as well - a story that is different for every team that ever was within the class). The feedback loop of present, vote, eliminate teams, hire from the eliminated teams, and repeat, provides a clearly ludic function throughout the duration of the class.

In the future, we expect to research and further develop a more thoroughly articulated ludologically informed pedagogical viewpoint that should be amenable to application in ways other than the specific form that can be found in the central focus of this paper.

We also expect to gain additional experience on the application of the approach presented here, and to report on that in future research.

VLSI Implementation of Residue Adder and Subtractor

Omar Dajani, Gurinder Bawa, Harpreet Singh
Department of Electrical & Computer Engineering
Wayne State University

Abstract: *In the present world there is always a demand for faster algorithms and techniques that could boost up the speed of the computations. With the help of VLSI fabrication techniques and using residue number system (RNS) arithmetic we can achieve the faster speeds. In this paper we have designed an RNS adder and subtractor; and then we have used VLSI techniques, XILINX and Cadence tools for the simulation as well as implementation of the design.*

1. Introduction

In the last decade RNS arithmetic has become attractive design option [1-3] for real time application fields such as signal processing, image processing and computer graphics. The merits of using RNS arithmetic lies in its capability of performing addition, subtraction and multiplication without the generation of carry propagation. Also RNS arithmetic has the capability of being designed and fabricated using VLSI techniques. This characteristic of RNS makes it most suitable for digital signal processor hardware.

In RNS the large numbers are represented by an *n*-tuple of smaller numbers that are independent of each other, where *n* is the number of modulus in the moduli set. Hence, number operation can be done on these smaller numbers rather than the original number. Furthermore, because the numbers in the *n*-tuple are independent of each other, the operations can be done in parallel. The RNS is defined by a set of moduli $(m_{1,}\ m_{2,\ \ldots\ ,}\ m_n)$ that are pairwise relatively prime positive integers. It can be shown that there is a unique representation for each number in the range of $0 \leq X < M$ Where $M = \prod_{i=1}^{n} m_i$ and is called dynamic range [4]. Each integer X can be represented by an *n*-tuple of residues $X = (x_{1,}\ x_{2,\ \ldots\ ,}\ x_n)$,

where $x_i = X \bmod m_i$ or it can be written as $x_i = |X|_{m_i}$. The RNS can represent any number within the range of [0, M) for unsigned numbers or $\left[-\frac{M}{2}, \frac{M}{2} - 1\right]$ for signed numbers. In RNS, the binary operations { +, - , * } are defined as follow If $Z = A \square B$ then $(z_{1,}\ z_{2,\ \ldots\ ,}\ z_n) = (a_{1,}\ a_{2,\ \ldots\ ,}\ a_n) \square (b_{1,}\ b_{2,\ \ldots\ ,}\ b_n)$ where $z_i = (a_i \square\ b_i) \bmod m_i$. Each residue digit can be computed independently of the others allowing fast data processing in *n* parallel independent channels.

VLSI implementations for adder and subtractor have been realized by many researchers. Bayoumi [5] used three approaches (look up table approach, binary adder approach and hybrid implementation approach). Taylor [6] realized implement-tations using various approaches. Banerji [7] and Soderstand [8] have their contributions as well for implementation of binary adders using VLSI techniques. In our work we have implemented an adder and subtractor. We have used Cadence (Virtuoso and Encounter) for the layout and XILINX for the simulation results of the design.

2. Residue Adder and Subtractor

If we have two integers A and B of modulo m, then their addition and subtraction is expressed as sum and subtraction of A $\pm$ B mod m. The operation addition and subtraction can be described as below in equation 1 and 2 respectively.

$$|A+B|_m = \begin{cases} A+B & \text{if } A+B<m \\ A+B-m & \text{if } A+B \geq m \end{cases} \quad (1)$$

$$|A-B|_m = \begin{cases} A-B & \text{if } A-B \geq 0 \\ A-B+m & \text{if } A-B<0 \end{cases} \quad (2)$$

Figure 1 shows the implementation of RNS adder /subtractor based on above equations.

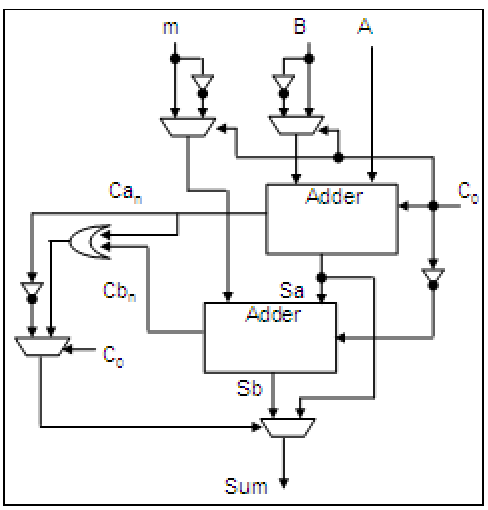


Figure 1: RNS adder & subtractor

The operation addition or subtraction is decided by the select line C_0. When C_0 is 0, the RNS operation is addition and when C_0 is 1, the RNS operation is subtraction.

For n-bit RNS adder the select line C_0 is 0 and it will simply perform as an adder. Inputs to adder are A, B and m. Figure 1 shows all the components that are used in RNS adder and subtractor. There are two full adders, four 2x1 multiplexers and one OR gate are used. As already mentioned for Adder C_0 is zero, therefore inputs A, B and C_0 are fed to the first full adder, which in turn will yield carryout Ca_n and sum Sa. In next step m, $\overline{C_0}$ = 1 and sum of previous full adder Sa goes to the next full adder in the hierarchy which in turn will give us sum Sb, Cb_n. Then Ca_n and Cb_n are added and are sent to third 2x1 multiplexer along with signal Ca_n. The output of the third 2x1 multiplexer is used as a control signal to the final 2x1 multiplexer which generates the final output Sum based on equation 1. For subtractor the scenario is same and equation 2 is implemented, but the control signal C_0 in this case is 1.

4. VLSI Implementation

In this section we implemented the adder and subtractor using VLSI techniques such as XILINX and Cadence tools (virtuoso, Encounter). For simulation we have used XILINX and for layout we have used encounter and virtuoso.

a) XILINX is a platform where we can generate the schematic using the verilog code of design, and at the same time we can test if the desired schematic or design is correct by looking at the behavioral simulation. Once we have created a verilog code for the adder and subtractor, we used the Xilinx tools to synthesize the schematic. Eventually for the verification of design, we built a testbench and run the simulation to see the correct desired output. Figure 2 shows the result of RNS adder and subtractor.

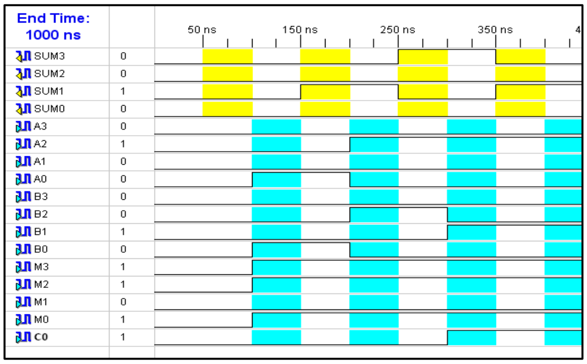


Figure 2: Waveform of RNS adder and Subtractor

b) Encounter is a technique where we can generate a layout of our design. Ambit Buildgates is used to generate the netlist and in encounter we upload the same. Further by assigning floor plans, appropriate layers and nano routing we can get the design in .gds and .def format. We can also check for the design if it is flawless or not in terms of connectivity, density etc. Figure 3 shows the encounter part of our design.

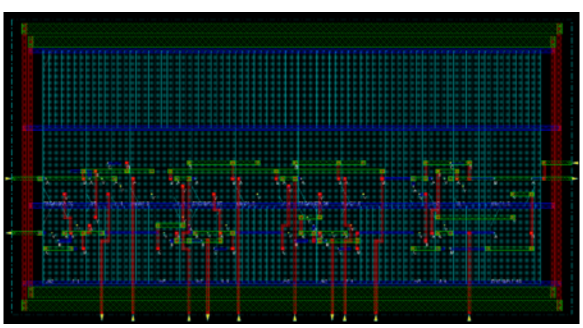


Figure 3: Encounter Part of RNS adder and Subtractor

c) Virtuoso is tool through which we can send our layout for the chip fabrication. We can pursue virtuoso in two ways, one is to build the design from basic building blocks, such as from transistor level (pmos, nmos etc.) and the second way is to import the design from encounter with the help of certain technology libraries. In our work we have used the later method where we have imported the design from encounter and the same was padframed in order to send it to MOSIS for chip fabrication. The complete Layout along with the connections is shown in figure 4 below.

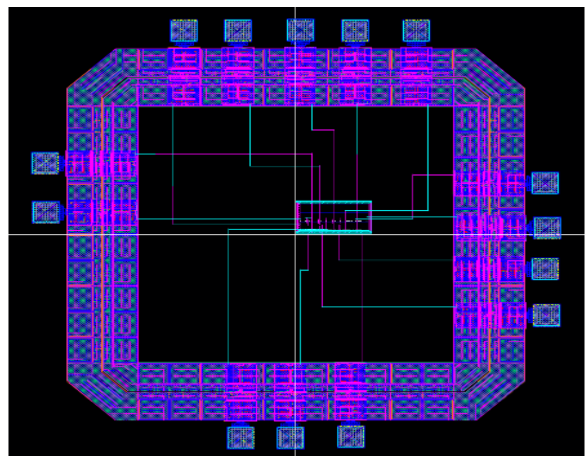


Figure 4: Virtuoso Part of RNS adder and Subtractor

5. Conclusion

In this paper we have discussed the VLSI approach for the realization of RNS adder and subtractor. XILINX is used to get the behavioral simulation of the design with the help of which we were able to verify that our design is performing as desired. Cadence encounter is used to build the layout of the design. Design layout along with the technology libraries and layers is exported to virtuoso. Furthermore padframing was performed in order to send the design to MOSIS for fabrication

References

[1] M.A. Bayoumi, "Digital filter VLSI systolic arrays over finite fields for DSP applications," in Proc. 6th IEEE annual Phoenix Conf. Computers and Communications, pp 194-199, Feb 1987.

[2] M. A. Soderstrand et al., Eds., "Residue Number System Arithmetic: Modern Applications in Digital Signal Processing" New York: IEEE Press, 1986

[3] K. Konstantinides and V. Bhaskaran, “Monolithic architectures for image processing and compression,”IEEE Computer Graphics & Applications, pp. 75-86, Nov. 1992

[4] N. S. Szabo and R. I. Tanaka, "Residue Arithmetic and Its Application to Computer Technology," New York: McGraw-Hill, 1967

[5] M. A.Bayoumi, G. A. Jullien, and W. C. Miller, “A VLSI implementation of residue adders,” IEEE Trans. Circuits Syst., vol. CAS-34, pp. 284-288, Mar. 1987.

[6] F. J. Taylor, “A single modulus complex ALU for signal processing,”IEEE Trans. Acoust., Speech Signal Processing, vol. ASSP-33, Oct. 1985.

[7] D. K.Banerji, “A novel implementation method for addition and subtraction in residue number systems,” IEEE Trans. Comput., vol. C-23, pp. 106-109, Jan. 1974

[8] M. A. Soderstrand, “A new hardware implementation of modulo adders for residue number systems,” in Proc. IEEE 26th Midwest Symp. Circuits and Systems, pp. 412-415, Aug. 1983.

SESSION

EDUCATIONAL ENVIRONMENTS, EVALUATION AND ASSESSMENT METHODS, VIRTUAL LABORATORIES, ANALYSIS TECHNIQUES AND WEB TECHNOLOGIES

Chair(s)

Prof. Hamid Arabnia

Google and IBM Clouds Make Enterprise Computing Available to all Undergraduate CS Majors

David J. Powell and Joel Hollingsworth
Department of Computing Sciences, Elon University, Elon, NC, USA

Abstract - *Cloud platform as service offerings (PAAS) from Google and IBM make it possible for all universities (large and small) to offer state of the art curriculum for enterprise level, server side software development. Google and IBM provide free access to the same hardware and software used by their corporate customers for software development and hosting of server side applications. Universities can easily and safely leverage the cloud to offer courses on enterprise computing to make their graduates more competitive in the market place. The Google cloud (App Engine) has been successfully used at Elon University for two years and the IBM cloud (SmartCloud Enterprise) for one year to teach Java Enterprise Level Programming of Servlets, JSP and JSF. This paper describes the advantages of using the IBM Smart Enterprise cloud as a platform for a course on Servlets 3.0 and JSP 2.2.*

Keywords: Cloud Computing, Enterprise Programming, Server-side Programming

1 Introduction

The computing field is constantly changing. Computer Science departments are challenged to continually analyze changes in the field to provide students with a curriculum that provides the key foundational concepts to support a career of lifelong learning while using state of the art software tools, frameworks and packages for the students to be competitive in the job market for internships and jobs after graduation. The rapid emergence and widespread use of mobile and cloud based computing where information is available to anyone, anywhere and at any time is changing the way we live, work, play and learn. Graduating students need to have a solid foundation in mobile and cloud computing to be competitive in today's marketplace [1]. Elon University made cloud based, server side programming, a part of the core curriculum in 2010. For the first two years (2010 and 2011), the department offered a core and an elective course in enterprise level computing using the Google cloud (Google App Engine). In 2012, the Google cloud was replaced with the IBM cloud (SmartCloud Enterprise). The enterprise level programming courses have enhanced student resumes and made the graduates competitive in the marketplace. Every graduate for the past three years has received at least one job offer and all rising seniors have accepted job internships for the summer.

The Google and IBM clouds are easy to use for teaching enterprise computing, offer students a competitive advantage in the job market and can be immediately integrated into an undergraduate curriculum of any size. This paper supports this hypothesis in four sections. The first section identifies the skills needed by students in enterprise computing and the challenges faced by a small university in offering a course to provide these skills. The second section describes the advantages and disadvantages of using the Google cloud as the platform for the course. The third section describes the configuration and use of the IBM cloud to provide an improved platform over an already excellent Google cloud. The fourth section provides details of the course delivery at Elon to provide readers a benchmark to use for developing a similar course. Finally, the paper concludes and identifies some future directions.

2 Curriculum

Ten years ago, most software development was for stand-alone applications running on a desktop machine with a single processor. There were many jobs for a software developer that was well skilled in a single language such as Java, C#, C++ or Visual Basic. Core computer curriculums were structured to support this single machine model. Students learned how to design, develop, debug, test, and maintain software for a single machine. However, during the past decade, the web platform has become the dominant development platform [2]. Developing for a web platform requires developers to be skilled in software development for multiple computers, multiple processes and multiple languages. The student needs to understand how to design, develop, debug, deploy and test an internationalized application that involves the client (either a mobile app or browser application), the web application server and the database server.

The number and types of languages and frameworks for enterprise programming is rapidly changing and will continue to change. At Elon University, we have selected mainstream, industry standards and software packages that implement the standards to provide the student the background to quickly learn other languages and frameworks while making them attractive to potential employers in the job market. We have selected Java as the foundational language for computer science majors and Eclipse as the primary IDE. As a minimum, we expect every graduate to have core skills for the client browser of HTML 5, CSS and JavaScript. For the web application server, we have selected Java EE 6 which is used to develop core skills in Servlets, JavaServer Faces, JavaServer Pages, JPA and JDBC. For the database tier, we expect all graduates to have core skills in SQL.

The field of Computing Sciences continually changes at a rapid pace. Computer Science Departments cannot simply add an additional course on cloud computing and purchase the hardware, software and technical support needed to support it. Departments have a fixed set of resources and constraints that must be optimized to provide maximum benefit to the student both in the short and long term. Some of the fixed set of resources and constraints are curriculum guidelines, budget and availability. Each of these constraints will be briefly discussed along with how Elon has made tradeoffs to add cloud computing into the curriculum.

2.1 Curriculum guidelines

Currently, there are over 265 ABET accredited undergraduate computer science programs. Each of these programs has met the accreditation constraints of having at least 40 semester hours in computer science and 30 semester hours of science and mathematics [3]. Within the 40 semester hours of computer science courses, programs are required to have coverage of algorithms, data structures, design, programming languages and computer organization and architecture. The total of 70 semester hours is a lot of hours and cannot be simply increased without impacting the students' ability to satisfy general education requirements at the university as well as pursue additional minors or majors. If a department is going to add cloud computing then it will come at the expense of removing other important topics from the 40 semester hour of computer science courses. Every item in the curriculum is very important and making relative tradeoffs based on perceived importance is difficult and is made based on the educational objectives for the graduates from each university.

Elon University is on a 4 credit hour system and each Computer Science major must complete 44 hours (11 courses) of Computer Science courses to graduate. Eight of these eleven courses are designated as core courses that each student must take to graduate. Elon considers enterprise level, server side programming to be a fundamental core course required of all students and has made the course called Computer Science III part of the core eight courses. The Computer Science III course has a prerequisite of Computer Science I and II where the student learns problem solving using Object Oriented Design and Programming with Java SE in an Eclipse environment. The Computer Science III course is positioned to be taken during the fall semester of the junior year to provide students the knowledge to be competitive and productive in an internship during the summer between their junior and senior years. The Computer Science III course is a standards based course and is based on Java EE 6. The course content covers client, server and database machine basics. After Computer Science III, students can design, develop, debug, deploy and maintain a multiple tier web based application with:

- a client browser using HTML 5, CSS, JavaScript
- a web application server using servlets 3.0, JSP 2.2, JSF 2.0, JPA 2.0 and JDBC
- a database server using an SQL database.

2.2 Budget

The economic crisis has frozen or reduced the department budgets at many universities. There are many competing requests for the fixed budget. The chances of submitting a successful request for server hardware, server software, database software and system administrative personnel to support cloud computing would be difficult. Moreover, even if the request is funded then most likely it would not be at level to match the free PAAS offerings on the Google and IBM clouds. Both of these clouds provide universities with free hardware, free software with no maintenance fees and free system administration. The free hardware is accessible 24 x 7 from anywhere on or off campus, has no firewall restrictions, has ample storage and excellent performance. The software is upgraded on a routine basis and the faculty member can easily apply/use the upgrades at a convenient time either during or between semesters. Both clouds allow users to be added with no need for a third party to create user ids and passwords. Google allows the user to directly request a user account and receive it within seconds via a confirmation password on their mobile phone. IBM gives the faculty member a GUI application to easily request a virtual machine instance with a username and password. The instance is then queued for creation and available within minutes.

2.3 Availability

Students have different class schedules and a variety of different constraints from on campus and off campus activities such as internships, sororities, student organizations and sports. An ideal computing environment would be available to the students at any time from any place. Students should not need to be within the university firewall or in a certain lab to develop their cloud applications. If a server goes down during the week or weekend then personnel should be available to immediately address the problem. At a small campus where system administrators have multiple job functions and support multiple departments, 24 x 7 access, service support and uptime is not pragmatic. However, both Google and IBM offer

24 x 7 access and have no firewall restrictions. In three years of using the Google cloud and one semester of using the IBM cloud there has been no measurable downtime.

3 Google cloud (Google App Engine)

Elon has used the Google cloud to teach the Computer Science III course on Enterprise Computing in the Fall 2010 and Fall 2011 semesters. In addition, the Google cloud was used to teach an Elon elective course on Servlets 2.5 and JSP in Spring 2010 and a SIGCSE mobile computing - cloud workshop in Spring 2011. The Google cloud has proven to be a superb cloud computing platform for the Elon course offering with many advantages. It is free, easily accessible through an Eclipse plugin for both Apple and Window machines, has generous quotas, requires no server system administration, has automatic backups, provides database connections with JPA or JDO, supports most of the servlet, JSP and JSF standards and has both an active development and user group. Until the recent release of pre-configured images of the IBM cloud, it was the platform of choice for all Elon cloud computing. For details of the course and the use of the Google cloud in the courses please see [4, 5].

As with any platform, there are some disadvantages or issues that you would like to see improved on. The primary disadvantage of Google App Engine is that while standards based it does not implement the full Java EE 6 set of standards released in fall 2009 and limits some standard Java SE functionality. By not fully implementing all Java EE standards for Servlets, JavaServer Pages, JavaServer Faces, JPA and JDBC, students get frustrated and spend many hours trying to get a feature to work that either is not available or is partially implemented and does not work as defined. Server side programming is hard to debug for beginners and these additional unexpected limitations adds to the complexity of teaching cloud computing.

Java EE 6 is made up of 28 specifications. Some of the key specifications which the Google cloud does not fully support are Servlets 3.0, JSF 2.0, JPA 2.0 and Java SE 6. The Google cloud does not support Servlets 3.0. It is still on Servlets 2.5 which does not support annotations. The Google cloud does not directly support JSF 2. However, with the addition of two additional jar files and some web.xml configuration settings, Google will support most of JSF 2.0. A major JSF limitation is the use of an earlier version of the expression library (EL 1.1) which does not allow parameters to be passed to actions. The Google cloud does not recognize the El 2.2 library. The Google cloud does not fully support JPA 2.0. Their partial implementation introduces major restrictions on the storage and retrieval of objects as the underlying BigTable Datastore is not based on SQL. Google does not support the full Java SE 6. The Google cloud has a well-defined whitelist that enumerates the Java SE 6 features not supported for performance limitations in making a highly scalable cloud. The key whitelist limitations are: no jdbc, no writing of files and no threads. The Elon faculty feel that the failure to support JDBC and to provide a freely available SQL database is a major shortcoming to students.

4 IBM cloud (SmartCloud Enterprise)

IBM has an Academic Initiative program “that facilitates the collaboration between IBM and educators to teach students the information technology skills they need to be competitive and keep pace with changes in the workplace” [6]. During the summer of 2011, Elon University faculty worked closely with their IBM academic alliance relationship manager with the goal of having preconfigured cloud images developed by IBM that would have all of the advantages of the Google cloud and none of the disadvantages. IBM developed two images that supported the full Java EE 6 specifications with no whitelist, required no software installation by the student and allowed each student to have both an individual virtual machine for development and a separate virtual machine for deployment. In addition to having all of the advantages and none of the disadvantages of the Google cloud, WebSphere expertise is more desirable in the marketplace for internships and full time positions after graduation.

4.1 Development and deployment images

IBM created a development image and a deployment image. Both images use the windows operating system which Elon students are familiar with and required no learning curve. The development cloud image had Derby, WebSphere V8 (WAS) and Rational Application Developer V8 (RAD) installed and pre-configured. The pre-configuration was modeled after the ease and use of the Google cloud. The goal was to allow a beginner student in server side application development to develop their application in an Eclipse environment and to deploy it to an application server with a single mouse click. The pre-configuration involved the following key features:

1. The Derby database was installed with two shortcuts on the desktop. The first shortcut started the derby server as a network database server accepting all connections on port 1527. The student just had to double click on the shortcut to start the database. The second shortcut was to shut down the database.

2. A WebSphere profile was defined with a DataSource called ElonDB. The named DataSource made it easy for Servlets, JavaServer Pages and JavaServer Faces to connect to it and provided a nice abstraction to hide the underlying details of connecting to the external Derby database. Two shortcuts were created on the desktop to start up and shutdown WebSphere with the defined profile.

3. RAD had a workspace preconfigured with a Data Perspective and two servers. This section will first discuss the Data Perspective in detail before discussing the two servers.

When the student started RAD, RAD automatically came up on the predefined workspace. The Data Perspective had two database connections profiles defined. One connection called “ElonDB – localtest”, shown in Figure 1, connected to a predefined Derby database on the Development image. The other predefined connection called “ElonDB – remote”, connected to the predefined Derby database on the deployment machine.

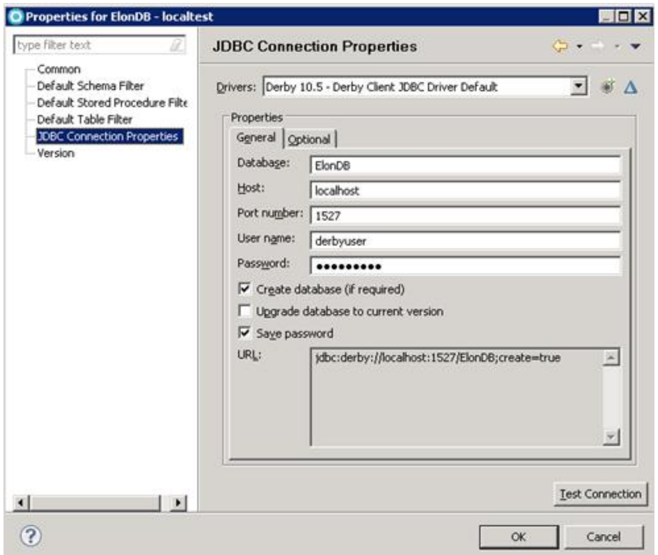


Figure 1: Eclipse configured database connection

The Database Connections in the Data Source Explorer View in the Data Perspective allowed the student to use the Data Source Explorer View wizards and GUI functionality to easily create, review, update and delete schemas, tables and data. The wizards and GUI functionality proved to be of great value when learning SQL and verifying that executing SQL statements through JDBC worked properly and to interactively develop the SQL to eventually call through JDBC.

Two WebSphere server instances are preconfigured in the Server Tab of the Java EE Perspective. The WebSphere instance running on the Development machine is called “WebSphere Application Server v8.0 at localhost with Derby”. The WebSphere instance running on the Deployment machine is called “WebSphere Application Server V8.0 with Derby”. The student right clicks the server instance that the student would like to publish an Enterprise Application on and selects the menu option to “Add” or “Remove”. An Add or Remove Wizard is displayed for the user to select the Enterprise Application. Once added then the student can invoke the desired annotated servlet or JSP page by supplying the full URL in a browser.

4. RAD is based on Eclipse and required no learning curve for the Elon students. RAD provided additional wizards and tools that fully supported creation of Java EE 6 server side applications.

4.2 No software installation

The image instances required no software to be installed by the student. Both instances were directly accessible using the Microsoft Remote Desktop Connection from any Windows machine on or off campus or any Mac machine running Microsoft Office. The student only needed to supply the IP address of the image instance and the remote machine console was displayed as if the student were seated at the remote machine. The student could easily access and transfer files between machines using cut and paste or drag and drop.

4.3 Two virtual machines per user

The IBM SmartCloud Enterprise GUI allows the instructor to click on an image to request the naming and creation of an instance. The instructor created two individual virtual machines for each student using a login of the student’s first and lastname and a password based on the student’s unique and private student identification number. By using a standard naming convention, each student was guaranteed privacy from other students while providing the instructor the ability to remotely connect to a student machine to grade or answer questions. One machine had an instance of the development image and the other machine had an instance of the deployment image. With both instances, the student had full access to the WebSphere Application Server GUI administrative console. This allowed the student to get a sense for the tasks involved by a system administrator in adding user roles for application security, defining database schemas and tables for an application and installing an application into an environment with other running applications.

4.4 WebSphere job opportunities

The Google and IBM clouds are easy to use, extremely reliable and provide a standards based PAAS for use at all universities to teach Enterprise Computing. The IBM cloud with WebSphere has the added advantage of providing increased internships and job opportunities for graduates. Indeed.com and Dice.com are recognized computer science job search sites. Both are easily searchable to determine the number of currently open jobs looking for experience with the Java EE 6 servers for WebSphere from IBM, Weblogic from Oracle, JBoss from Redhat and App Engine from Google. Figure 2 shows the number for job openings posted by Indeed.com on May 22, 2012. The clear leader is WebSphere followed by Weblogic, Redhat and App Engine. There are significantly more jobs for WebSphere than App Engine. A similar query was run on Dice.com with results showing a similar relative ranking of application servers. There were 3209 WebSphere postings, 2156 Weblogic postings, 1856 JBoss postings and 155 App Engine postings. Eleven Elon students took the Spring 2012 enterprise computing course on the IBM Cloud. Four received internships or full time positions using WebSphere and one received a full time position using JBoss.

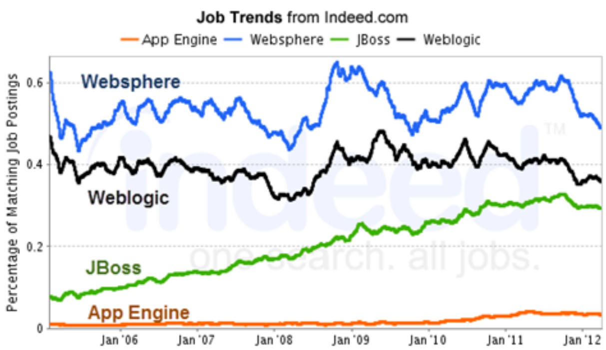


Figure 2: Current job postings for Java EE server skills

5 Course content

The Spring 2012 Elon course on enterprise computing focused on teaching server side programming using Servlets 3.0 and JSP 2.2 with an external SQL compliant database. When HTML was generated by servlets or JavaServer Pages, it was HTML 5. The IBM cloud discussed in section 4 was used for all development and deployment. This section describes the course materials, course format and course assessment.

5.1 Course materials

The course had two required text books and was supplemented by online materials. The required text books were Core Servlets and JavaServer Pages Volume 1 and 2 [7][8]. The author of these textbooks, Marty Hall, is president of a training and consulting company, coreservlets.com, that focuses on server-side technology and is an adjunct professor at John Hopkins University. He has extensive experience teaching Servlets, JavaServer Pages and JavaServer Faces at major corporations around the world. He freely supplies all of his course materials to faculty [9]. The materials include PowerPoint slides, example code, class exercises and class exercise solutions. His code is extremely well written and both uses and reinforces good programming practices. Every example that he provided worked without modification on the IBM cloud. This is not the case when using the Google cloud due to its whitelist limitations. Marty Hall updates and improves course materials after each of his courses based on feedback from corporate customers and students. The course materials complement his textbooks and are updated to the latest release of Servlets 3.0 and JavaServer Pages. This is critical. With the exception of one text book, all of the leading Servlet and JavaServer Page texts are based on Servlets 2.5.

Online materials were used to supplement the books and provided additional information on servlets 3.0, databases and HTML 5. For additional information on servlets 3.0, the students used the Servlet 3.0 Cookbook [10]. The site provided a nice overview of the new annotations used in Servlets 3.0 and provided illustrative examples. For learning SQL and the Derby database, the students were directed to read a 14 article series on IBM developerworks by Robert Brunner [11]. The hands on series starts by introducing the student to the Derby database by building a schema and table using SQL interactively with the Derby IJ Tool. The next seven articles cover the SQL data definition and data manipulation language. The last six articles cover JDBC. HTML 5 was used primarily for the new form capabilities. W3Schools provided a good online resource to introduce the new form attributes and elements [12].

5.2 Course format

The Spring 4 semester hour course met twenty eight times over 14 weeks. Each class session was 100 minutes in length and held in a lecture/lab setting. Each class typically consisted of a lecture followed by an in-class computer based exercise on the covered material using the IBM cloud. With the exception of database coverage, the course primarily followed the chapter sequence in the course texts and had four major parts. The first part of the course covered basic servlets, servlet annotations, request and response headers, cookies, sessions and file uploads. The second part of the course had a database focus. The Derby database was introduced with coverage of SQL, table design, normalization, JDBC and Data Access Objects. Exercises were assigned to have Servlets perform create, read, update and delete operations (CRUD) from Derby databases created by the students. The third part of the course covered JavaServer Pages, scriptlets, Java Beans, the JSP 2.0 expression language and using MVC for application design. Most exercises in this part required database CRUD operations to reinforce the heavy use of a database in server side applications. The fourth part of the course focused on advanced technologies covered in CoreServlets and JavaServer Pages Volume 2 for deployment descriptors, declarative and programmatic security, listeners, filters and JSTL. Exercises in this part required students to use their development machine for developing and testing the web application and database and to then deploy both the database tables and the web application to their deployment machine. Students had a chance to use the Administrative Console of WebSphere to add user names, passwords and groups and to associate these names against security roles defined in the web application to control resource access.

5.3 Student assessment

Students received frequent and continual assessment throughout the course with quizzes, homeworks and a final exam. The students had three major quizzes during the course. There was one quiz after part 1, one quiz after part 2 and one quiz after part 4. Each quiz had questions focused only on the material for the part of the course that it was assessing. The questions were taken from a test bank of questions for the concept being tested with a difficulty similar to that found on the Oracle Java EE Web Component Developer Certified Expert Exam. Students were provided links to various mock certification tests to prepare for each quiz. The goal of these quizzes was to have the student supplement the book reading with an in depth study of appropriate APIs in preparing for each quiz.

Individual homework programming assignments on the IBM cloud were assigned weekly. Each assignment typically took 2 – 5 hours to complete. During part one of the course, each assignment focused on the topic covered for the week. During parts three and four of the course, each homework assignment built on the previous assignment. The building block approach allowed each student to build a large, working web application that mimicked the play.google.com site for maintaining a list of Android Applications. The increasing complexity of each homework enabled students to understand the importance of the Model View Controller architecture, security restrictions and data access objects. The overall high average of 91 on assignments indicated that students had developed the ability to design, develop, debug, test and deploy an enterprise level application.

The final exam was a complete Java EE 6 Web Component Developer Certified Expert Exam from enthuware.com. The students were given the same number of questions (57) and the same time limit (120 minutes) as the actual exam. The use of the three earlier quizzes helped the students prepare for this exam as they understood the level of difficulty of the questions and the format of the exam.

6 Conclusions

The web platform is the dominant platform of our time. Elon University believes that enterprise computing should be a required course for all CS majors and has it made it a core course in its curriculum. The Google cloud and the IBM cloud have been successfully used at Elon and have provided students free, 24 x 7 access, with high reliability. The recently adopted IBM cloud with pre-configured images has proven to be as easy to use as the Google cloud while offering the advantages of being a fully Java EE compliant environment. These clouds are available to all universities and provide an ideal PAAS for teaching enterprise skills and make the student more competitive in the job market for internships and after graduation.

7 References

[1] Rosenberg, J. 2012. Skype Chief Technologist, April 16, 2012, Presentation to Computer Science Forum at Elon University.

[2] Gundotra, V., Picai, D., Google IO 2010 - Keynote Day1. http://www.youtube.com/watch?tv=mbVVDDu8f9k&feature=related

[3] Criteria for Accrediting Computing Programs 2012-2013. http://www.abet.org/computing-criteria-2012-2013/

[4] Hollingsworth, J. and Powell, D. 2011. Requiring Web-based Cloud and Mobile Computing in a Computer Science Undergraduate Curriculum. Proceedings of the 49th ACM Southeast Conference.

[5] Hollingsworth, J. and Powell, D. 2010. Teaching Web Programming Using the Google Cloud. Proceedings of the 48th ACM Southeast Conference.

[6] IBM Academic Initiative. https://www.ibm.com/developerworks/university/academicinitiative/

[7] Hall, M. and Brown, L., Core Servlets and JavaServer Pages: Core Technologies, Vol. 1 (2nd Edition), Prentice Hall, 2003.

[8] Hall, M., Core Servlets and JavaServer Pages: Advanced Technologies, Vol. 2 (2nd Edition), Prentice Hall, 2007.

[9] Hall, M., Intermediate Servlets & JSP Tutorials, http://www.corservlets.com/Course-Materials/csajsp2.html

[10] Servlet 3.0 Cook Book. http://www.servletworld.com/servlet-tutorials/servlet3/cook-book.html

[11] Brunner, R. 2006. Developing with Apache Derby. http://www.ibm.com/developerworks/library/os-ad-trifecta1/index.html

[12] HTML 5 Tutorial. http://www.w3schools.com/html5/

Standard Evaluation of Interventions for Those with Autism Spectrum Disorder

Justin Ehrlich
School of Computer Sciences, Western Illinois University
Stipes Hall 447 / 1 University Circle
Macomb, Illinois, US

Abstract *– Increasingly, Virtual Reality (VR) interventions are being researched as tools to treat those with Autism Spectrum Disorders (ASD). These studies require a standardized toolset to effectively test both the efficacy of VR interventions and the effectiveness of interventions compared to traditional interventions. Therefore, this paper presents a method for testing VR interventions against traditional interventions. Further, this paper presents a method for testing improvements to VR-based intervention to determine any benefits gained from the improvements. The methods presented here has been successfully used in both a previous study (24 subjects) involving the effects of illumination realism and a pilot study to determine the efficacy of a VR-based intervention. Also, the methods are being used in a current study (400 subjects) testing the efficacy of a VR simulation for use within schools compared with traditional interventions.*

Keywords: virtual reality, autism, intervention, social skills, education

1 Background

Virtual Reality (VR) interventions have been established in the special education literature as an effective evidence-based practice [1] and there have been numerous successful studies using virtual reality as interventions for individuals with ASD [2-4]. In a study conducted by Herrera and colleagues [2], an argument is made attributing VR to improvements in the understanding and manipulation of objects within the imagination. The inability to generalize symbols in the imagination lies at the heart of the ASD disability [5]. VR encourages the students to imagine virtual scenes as actual scenes and avatars as real people. Herrera, citing prior research conducted by Sherratt et al. [6], suggests that symbolic play can be induced through "use of a clear structure, the repetition of familiar routines, and the degree of affective treatment", all of which VR can provide. If a VE can encourage this symbolic generalization in the imagination, not only will the particular skill-set be learned, but also the generalization ability of the subject may be improved and the student’s perspective-taking ability may be made more effective. Indeed, in his 2008 study [2], Herrera did find that generalization does occur by using VR. Typically, the reason behind these studies is to develop an intervention that can improve the perspective-taking ability for individuals with ASD, which as described above, allows people to make conscience decisions pertaining to behavior. Individuals with ASD lack the ability to develop an effective perspective-taking ability on their own. Traditionally, interventions have involved teaching a set of rules to enhance the individual’s ability to do this. In a 2007 study [3], Mitchell and colleagues make the argument that teaching rules by themselves is not sufficient. While the external actions of social behavior appear to be rule based, the execution of one's understanding of a situation involves "imaginatively adopting other people’s conceptual perspectives." Pretense is needed, but children with ASD lack some of this ability. VEs offer a unique opportunity in that role-play is necessary to traverse an environment. Mitchell comments that the more realistic the virtualized environment, the greater the chance of generalizing the role-playing skill-set induced by the VE through imagining other people’s perspectives.

The issue now is how to evaluate these VR-based interventions in a standard method. It is difficult to find the best interventions and reasons for VR-based intervention effectiveness by scanning the literature since there are currently no standard methods for evaluating these studies. Therefore, this work provides three methods of evaluations for three different scenarios of testing. The first scenario is simply evaluating the efficacy of a VR-based intervention. This is typically for pilot studies to determine if the intervention can be used as a tool for treating those with a disability. For this scenario, this work proposes the utilization of a pretest and posttest using videos to describe situations and asking users what the best response would be. The subjects would be kept in one group for a within-group study. If the intervention was at all effective, the subjects would answer more questions correctly during the posttest than pretest.

The second scenario involves comparing a VR-based intervention with a standard intervention. This is to determine if the VR-based intervention is a superior tool than that already being practiced. This work proposes the use of between-group study divided between a group experiencing the VR-based intervention and a control group experiencing a traditional intervention, such as role-playing, discussions, games, and other activities that may instill social rules [7, 8]. A posttest is performed, again based upon videos, showing situations and then asking the user what the best response would be. It is hoped that the group experiencing the VR-based intervention answers statistically significantly more questions correctly than the control group.

The third scenario's goal is to determine how much more effective, if any, improvements made to an existing VR-based intervention are. This scenario is divided into two parts. First, the VR-based intervention should be tested in a similar fashion to the second scenario. For this a between-group study is conducted with one group receiving the standard intervention while the other group gets the improved intervention. A posttest is performed using videos portraying situations and the user is to pick the best response to each situation. For the improved intervention to be more effective, the a statistically significant increase in correct responses should be present in the group experiencing the improved intervention.

For the second part, since VR-based interventions are based upon immersive technologies, a test of presence should be conducted since it is thought that an increased presence increases the generalization in PVEs for not only neurologically typical (NT) individuals [9], but also for those with ASD [2]. A questionnaire can be inserted into the intervention after each scenario or step to determine the presence felt of that particular segment. It is hoped that any improvements to immersive technology should provide a greater sense of presence, and so the group experiencing the improvements will provide more emotionally charged responses than that of the group experiencing the traditional intervention.

2 AViSSS

The proposal is based upon work a number of colleagues and I have done on developing a series of studies in order to develop a VR-based intervention to teach social skills to those with ASD. Researchers at the University of Kansas and I created a virtual intervention that teaches social skills that we call Animated Visual Supports for Social Skills (AViSSS) [10, 11]. Our group wanted to create an intervention that is not invasive, is simple to use, and is relatively inexpensive. Figure 1 is a screenshot from AViSSS depicting a hallway scene.

Figure 1: The Hallway in AViSSS. In this sequence the user is presented with a scene where an avatar bumps into the user. The user should respond with an "excuse me."

To evaluate the effectiveness of a social skill-teaching tool with a VE on a standard desktop display, we needed to answer two questions:

1) Can students with ASD acquire social skills from a virtualized intervention?

2) How much more effective, if any, is a virtualized intervention compared to a traditional intervention?

To answer the first question, we conducted a within-group study for students using our intervention. The evaluation was done with a pretest and posttest that tested social skill ability. If results from the posttest showed a significant improvement over the pretest, the students must of accepted the virtual environment and learned from the intervention. Further, we added questions on the posttest to determine the ability to generalize the specific social skills taught with the intervention.

To answer the second question, we conducted a between-group study comparing social skill ability between two different groups of students with ASD. The control group (CG) experienced a traditional intervention, specifically social narrative videos. The second group (AG) experienced the AViSSS intervention. The social skill ability was measured using the same posttest as in the within-group study.

Both of these questions require a tool that can be used to compare results from both the VR-intervention and a traditional intervention. Traditional interventions such as social narratives with flashcards are normally used by specialists to present situations along with correct responses to individuals with ASD in an attempt to teach social rules to these individuals. Videos have also been used to describe scenarios and are easy to preset on the computer without any confusion. Therefore our tool uses videos to compare the efficacy of both the VR intervention and the standard intervention. This tool can be utilized for both between-group and within-group studies.

Additionally, another study conducted by my team explored illumination realism to understand how much more effective a VR intervention can be with improved illumination. To test this, a tool had to be developed to test the efficacy between two groups – one experiencing a more realistic form of AViSSS and the other group experiencing a less realistic form. This can be used to understand any differences in effectiveness that improvements to VR-based interventions may bring to the subjects.

3 Methodology

To assess whether or not students with ASD can use AViSSS to learn social skills, we used a pretest and posttest. The students experience AViSSS immediately after taking the pretest, and then immediately after that, the posttest is given. For each question, a video of a situation is shown along with a question dialog and narration explaining the situation. Each question's response has multiple choices with either three or four choices for each question. The responses along with the number of correct answers were recorded for each student. Each situation is a video of live actors performing the scene that was virtualized within AViSSS and these situations are shown on both the pretest and posttest. Also included is six situations not seen within AViSSS to test the generalizability of the skills learned.

For the video-based tests, we modeled our experimentation on Mitchell's 2007 study [3] and showed twenty different videos to both groups before exposure to AViSSS. Each video related to a different social situation that AViSSS was attempting to address. After the intervention, the users saw the twenty videos again, along with six different videos of situations not addressed within AViSSS. This second set from the second portion of the videos corresponded to the situations from AViSSS, although the environment changed to show if the student could generalize the responses. The users were asked what they would do in each video situation. For instance, as depicted in Figure 2, the first video set might contain a video of a cafeteria scene where the user must choose a seat. The user gave a response to each scene, which we recorded. We then subjected the user to AViSSS, which included a scene that guided the user in choosing a seat in the cafeteria. After the user completed AViSSS, we showed the second set of videos. The first part of this second series was exactly the same as the first and included the entire cafeteria scene. This demonstrated the effectiveness of AViSSS at teaching social skills at the particular situations in the cafeteria virtualized within AViSSS. The second part was a seat-choosing scene in the school library, which was never addressed by AViSSS. We again asked the user to make a choice for each situation, which we again recorded. This demonstrated the effectiveness of AViSSS at teaching the generalization of picking a seat.

To determine the effectiveness of AViSSS compared to a standard intervention, we used a posttest based upon social skills targeted during the interventions. This posttest was the same as used above but is given for both AViSSS and the standard intervention. The standard intervention consisted of the current offerings that the school was providing the student.

Figure 2: Screenshot of the Cafeteria video. This still image is one frame of a 30-second sequence involving audio and video of professional actors/actresses.

To determine the effectiveness of a more realistic form of AViSSS, a video-based test had to be performed testing user knowledge before and after the intervention. This video-based test was in the same form as that used for the above study. Each group was randomly selected from the sample population. Group GL experienced AViSSS with global illumination and group LL experienced AViSSS in standard graphics mode – without the global illumination. After each of the nine scenarios, AViSSS asked the students to answer one of the reported presence questions. Each question was designed specifically for a corresponding scenario. The answers to the questions were written to a file, which we analyzed.

For the effect of illumination realism on presence, a questionnaire is administered during both groups' experience in their respective VE. The questionnaire during the test asks questions similar to the ones Mania suggests, which were taken from those introduced [12] and justified [13] by Slater et al. and verified by Usoh, et al. [14] to find reported presence. Originally, a 5-point scale was used, similar to Mania's approach. For example, these three questions were to be part of the study:

- Rate your sense of being in the 3D room, where 5 represents your normal experience of being in a place (1=Not at all. 5= very much)?

- To what extent were there times during the experience when the 3D room was reality for you (1=not at all. 5= Most of the time)?

- When you think back about your experience, do you think of the 3D room more as images that you have seen or more somewhere that you visited (1= images. 5= a place visited)?

After a pilot study with individuals that varied in age and placement on the ASD scale, this spectrum was found to be too abstract for them to conceptualize. After interviewing a number of the specialists, including a couple of educators, we came to the conclusion that these individuals think much too discretely to place ideas on a continuous scale. For instance, a midrange response was rarely received; instead the answers were usually extreme. This finding, coupled with the interviews, led us to use discrete multiple choice answers to the questions instead.

While converting the spectrum questions into discrete multiple-choice questions, we carefully retained the nature of the questions, based on the recommendations from past studies by Mania et al. [9, 15-17] and Slater et al. [12, 13, 18, 19]. To quantify the choice, each response was given a value from 0-3, with 0 defining no presence and 3 defining the most presence. The following is one such question from the hallway scenario:

What was your experience like in the hallway?

It felt like I was in the hallway. (Value = 3)

It usually felt like I was in the hallway. (Value = 2)

It usually did NOT feel like I was in the hallway. (Value = 1)

It never felt like I was in the hallway. (Value =0)

4 The Studies

The first study tested the efficacy of AViSSS to demonstrate the VR-based intervention as a proof of concept. We used a within-group model to determine any social skills learned during the intervention. For this, 24 subjects were asked to experience AViSSS and to complete a pretest and post video-based test, as described above, to determine what was learned, if anything, during the study. We found a statistically significant increase in questions answered correctly between the post and pretest.

The first study also sought to evaluate the differences in effectiveness between AViSSS and traditional interventions. In order to compare the interventions, we divided 24 subjects into two groups and conducted a between-group study. In this portion, we conducted a posttest and analyzed the differences between the responses of both groups. To measure the efficacy,

The current study involves 400 students and is testing the efficacy of AViSSS compared to a traditional intervention involving Social Skills Training rules [7, 8] in the form of still images and narrated dialog. This is a between-group study with one group experiencing AViSSS and the other group experiencing the traditional intervention. The control group will be presented with a series of narrated situations along with the best responses. After both groups experience their interventions, a series of video-based situations along with multiple-choice responses are presented to the users. These questions are exactly the same as described for the study comparing two VR-based interventions, except now they're used to evaluate the efficacy of AViSSS compared to a traditional intervention.

The study to determine the effectiveness of improving an intervention was designed to compare the efficacy and generalizability between two different interventions. We improved AViSSS by increasing the immersive technology and compared this to the standard version of AViSSS. To find possible differences of effectives between the two groups, we measured any differences in correct responses that global illumination caused in AViSSS using the same videos as utilized above. We found a significant increase in effectiveness of the improved AViSSS compared to that of the standard AViSSS, which is fully discussed in the publication of the work [20].

To determine the level of presence felt while increasing the realism of AViSSS, we also asked ten questions relating to presence during the intervention, as discussed above. We didn't want the user to forget the each experience, so we asked the question immediately following each ten of the situations. We found a statistically significant increase in presence felt using the improved version of AViSSS compared to the standard AViSSS.

5 Discussion

The results from both studies demonstrate that this video-based test is an effective tool to evaluate both the efficacy of a VR intervention and to compare VR interventions. While the results from the efficacy study are not yet available, a pilot study demonstrates a statistical difference between the efficacy of AViSSS and a traditional video-based intervention. This study depends on a valid evaluation of its results, so an objective test is pertinent to its results. In a pilot study, we have found a statistical difference between AViSSS and a traditional video-based intervention, with AViSSS being more effective.

The study comparing the effectiveness between two versions of AViSSS successfully used the video-based tests to compare and a statistical difference was found. This implies that the video-based test can be used to compare two different interventions within a between-group study by conducting only a posttest that determines the efficacy of each intervention. The results can be compared between the two groups with statistical significance. The outcome of this study demonstrated that the more illumination-realistic VE was able to outperform the lesser illumination-realistic VE in terms of efficacy. This provides evidence that the video-based test discussed here is a valid tool for evaluating different VR-based interventions.

Additionally, the inner-AViSSS questionnaire was effective at determining the felt presence between the two versions of AViSSS, with the more realistic version providing a deeper feeling of presence. This is an important because presence allows the user to make stronger connections between the virtual and real world. The more presence the virtual environment induces automatically, the more the child imagines subconsciously. Therefore, the child requires less intentional imagination (mental imagery [21]) to think of the environment as real. If we can induce pretense while the user is experiencing strong presence, the user can learn as a NT individual does since it’s similar to using pretense to learn in the real-world, including the same emotions, except for the fear of failure.

6 Conclusion

In conclusion, the presented video-based test is an effective tool to not only evaluate the efficacy of VR-based interventions for those with ASD, but also at comparing VR-based interventions. Therefore this test can be used to evaluate VR-based interventions against each other or testing the efficacy of interventions against traditional interventions. This test was successfully used for two studies targeting those with ASD and should be able to be extended for use with any interventions involving VR.

This work is pertinent for any future work involving VR interventions, for a standard toolset for evaluation is needed to effectively evaluate these VR interventions. Without a standardized tool, the comparison between interventions would be subjective upon the interpretations of the researchers, and not of the experiences of the users.

It is suggested that researchers’ use professional actors to create the videos for the tests. In the study, we used actors/actresses from the University of Kansas’ Drama Department, which made the videos much more realistic than using amateur actors/actresses. Further, the questions much be carefully selected to match the age groups of the targeted users. If not, the difference of the results will be too small to be statistically significant.

For future work, we would like to evaluate other forms of testing, such as virtualized flash cards, which are also a popular form of intervention. Perhaps this along with video interventions would allow for a more precise form of testing interventions. Also, this may cause VR interventions to be taken more serious by traditional caregivers, which many are hesitant to use new forms of interventions to treat their students.

7 References

[1] S. Smith and H. J. Lee, "Virtual Environments for individuals with ASD," 2009.

[2] G. Herrera, F. Alcantud, R. Jordan, A. Blanquer, G. Labajo, and C. De Pablo, "Development of symbolic play through the use of virtual reality tools in children with autistic spectrum disorders: Two case studies," *Autism: The International Journal of Research & Practice,* vol. 12, pp. 143-157, 2008.

[3] P. Mitchell, S. Parsons, and A. Leonard, "Using Virtual Environments for Teaching Social Understanding to 6 Adolescents with Autistic Spectrum Disorders," *Journal of Autism & Developmental Disorders,* vol. 37, pp. 589-600, 2007.

[4] D. Strickland and L. M. Marcus, "Brief report: Two case studies using virtual reality as a learning tool for autistic children," *Journal of Autism & Developmental Disorders,* vol. 26, pp. 651-659, 1996.

[5] A. M. Leslie, "Pretense and Representation: The Origins of "Theory of Mind"," *Psychological Review,* vol. 94, pp. 412-426, 1987.

[6] D. Sherratt and M. Peter, Developing Play and Drama in Children with Autistic Spectrum Disorders. London: Fulton, 2002.

[7] K. S. Exkorn, The autism sourcebook : everything you need to know about diagnosis, treatment, coping, and healing. New York: ReaganBooks, 2005.

[8] C. Sicile-Kira, Autism spectrum disorders : the complete guide to understanding autism, Asperger's syndrome,

pervasive developmental disorder, and other ASDs, 1st Perigee pbk. ed. New York: Berkeley Pub. Group, 2004.

[9] K. Mania, D. Wooldridge, M. Coxon, and A. Robinson, "The Effect of Visual and Interaction Fidelity on Spatial Cognition in Immersive Virtual Environments," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, pp. 396-404, 2006.

[10] J. A. Ehrlich and J. R. Miller, "A Virtual Environment for Teaching Social Skills: AViSSS," *IEEE Comput. Graph. Appl.,* vol. 29, pp. 10-16, 2009.

[11] J. A. Ehrlich, "The effect of desktop illumination realism on a user's sense of presence in a virtual learning environment," Ph.D. dissertation, University of Kansas, Lawrence, KS, 2010.

[12] M. Slater, A. Steed, J. McCarthy, and F. Maringelli, "The influence of body movement on subjective presence in virtual environments.(Special Section: Virtual Environments: Models, Methodology, and Empirical Studies)," *Human Factors,* vol. 40, p. 469(9), 1998.

[13] M. Slater, "Measuring Presence: A Response to the Witmer and Singer Presence Questionnaire," *Presence: Teleoper. Virtual Environ.,* vol. 8, pp. 560-565, 1999.

[14] M. Usoh, E. Catena, S. Arman, and M. Slater, "Using Presence Questionnaires in Reality," *Presence: Teleoperators & Virtual Environments,* vol. 9, pp. 497-503, 2000.

[15] K. Mania, "Connections between lighting impressions and presence in real and virtual environments: an experimental study," presented at the Proceedings of the 1st international conference on Computer graphics, virtual reality and visualisation, Camps Bay, Cape Town, South Africa, 2001.

[16] K. Mania and A. Chalmers, "The Effects of Levels of Immersion on Memory and Presence in Virtual Environments: A Reality Centered Approach," *CyberPsychology & Behavior,* vol. 4, pp. 247-264, 2001.

[17] K. Mania and A. Robinson, "The effect of quality of rendering on user lighting impressions and presence in virtual environments," presented at the Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry, Singapore, 2004.

[18] M. Slater, P. Khanna, J. Mortensen, and I. Yu, "Visual Realism Enhances Realistic Response in an Immersive Virtual Environment," *IEEE Computer Graphics and Applications,* vol. 29, pp. 76-84, May/June 2009.

[19] M. Slater, M. Usoh, and Y. Chrysanthou, "The influence of dynamic shadows on presence in immersive virtual environments," presented at the Selected papers of the Eurographics workshops on Virtual environments '95, Barcelona, Spain, 1995.

[20] J. A. Ehrlich, "The Effect of Global Illumination on Presence in a Virtual Environment for those with Autism Spectrum Disorder," in *CGVR 2011*, Las Vegas, NV, 2011, pp. 65-70.

[21] Stanford. (2008, 2/12/2010). *Mental Imagery*. Available: http://plato.stanford.edu/entries/mental-imagery/

Design of a Reconfigurable Virtual Computing Laboratory

S. Baev[1], **B. Peltsverger**[1], **and A. Shah**[1]
[1]Department of Computer Science, Georgia Southwestern State University, Americus, GA, USA

Abstract— *A growing number of online students and complexity of programming projects dictate the necessity of designing/deploying a computing infrastructure which must satisfy contradictory requirements. From one stand point, such infrastructure is expected to grant administrative privilege to students, working on a variety of programming projects, for instance, web programming for cloud computing. From another perspective, granting administrative privilege will not only allow students to access all files on the system (including other students' work) without legal permission but also may result in a system crash. The simplest workaround in this case would be to setup dedicated workstations to individual students but this solution doesn't seem to be scalable, reusable, and convenient to run distance learning process.*

Keywords: Infrastructure, networking, virtualization, hypervisor.

1. Proposed Solution

The authors propose a design of *virtual computing laboratory* which not only capable of addressing the mentioned problem but also has many other useful features: *scalability*, *improved security measures*, *convenient administration*, and *reduced cost*. The proposed design is sketched on Figure 1.

1.1 Outline of the Infrastructure

The design of the computing infrastructure is *hybrid* so it has a mixture of *physical* (hardware) and *virtualized* (software) components. In particular, the virtual part of the infrastructure exploits *hypervisor* technology in two most commonly used implementations: *Citrix XenServer* and *VMWare vSphere ESXi* server. Both implementations allow hosting multiple virtual machines on a *bare-metal* platform (hardware server running hypervisor software). Virtual machines are interconnected into a *hybrid network* supported by several *physical* and *virtual* network *switches*. The last crucial component of the system is a set of *thin terminals* which are *inexpensive*, *diskless* endpoints of the user interface provided by virtual machines. Such an interface, (a.k.a. *virtual desktop*) is delivered to end-users by means of either of the following approaches: Microsoft Windows Terminal Service (RDP protocol) or Linux Terminal Server Project.

From the perspective of the offered services, the infrastructure has the following components:

1) **Centralized gateway/firewall/access controller** implemented on the basis of pfSense appliance
   a) **Firewall / NAT router**: (i) filters network traffic; (ii) supports translation of private IP addresses; (iii) provides access to private servers from the Internet.
   b) **OpenVPN server**: provides controlled access to the private network from the Internet via a variety of OpenVPN clients (Windows, Linux, Mac, Android).
   c) **DHCP/DNS servers for private network**: supports high level of flexibility and scalability of the infrastructure.
   d) **Content filter**: processes HTTP/HTTPS traffic with respect to class schedule and pre-configured access list of allowed/disallowed Internet resources.
2) **Intrusion detection/protection system (IDS/IPS)** implemented on the basis of Security Onion appliance (snort, snorby, sguil, squert, and others)
3) **Network Attached Storage (NAS)**: allows for centralized, convenient, configurable storage of user and configuration data.
4) **Directory Server** (OpenLDAP project): centralized storage of user credentials used for login and access control purposes.
5) **Terminal servers (Windows and Linux)** allowing for the delivery of virtual desktops to a set of diskless thin terminals (clients)
6) **Thin terminals (clients)** are robust, reliable, and cost efficient computing devices âĂŞ user interface of the infrastructure
7) A variety of **application servers** to support teaching and research processes
   a) *Document management server* (LetoDMS project): combines standard document manipulation functionality with a nice-looking web-interface.
   b) *Wiki* (MediaWiki project): supports publishing supplementary materials used in a variety of classes and research projects.
   c) *Version control system* (SVN repository): allows for centralized storage of students' project files with proper backups, access control, and convenient web-interface.

1.2 Functionality

The introduced computing laboratory is also designed to be *highly reconfigurable* in terms of *structure* and provided

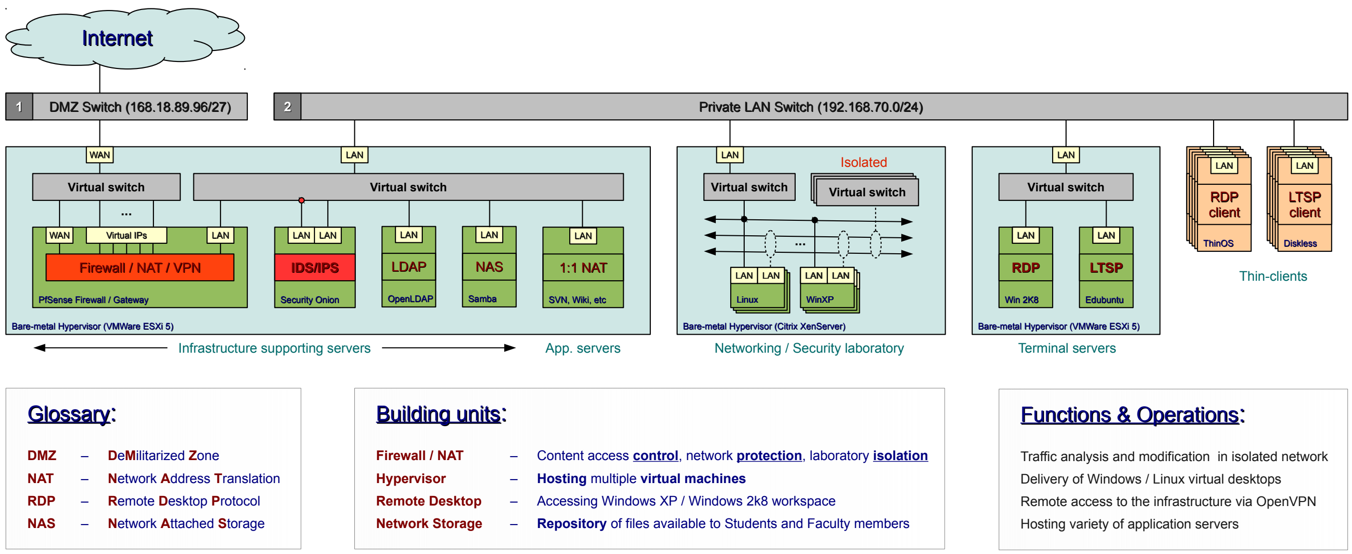


Fig. 1: Outline of the Reconfigurable Infrastructure

content. Having basic set components intact a *new* set of *servers* (physical or virtual), *workstations* and *thin terminals* can easily be added to extend the scope of laboratory applications. Moreover, a library of virtual machines images can be made available to allow *on-demand reconfiguration* of the computing environment in case of shared classroom facilities. A typical scenario can be recognized by many faculties: one same computer classroom is used throughout the day to offer a variety of classes. Say, at 8:00 AM it is a UNIX class which requires nothing more than a shell console to be available – any other software (including web browsers) is disabled to help students gaining certain skills and habits. Once the class time is over, say from 9:30 AM to 11:00 AM, the laboratory is reconfigured for *assessment purposes* (testing center), for instance, each workstation is locked-down to web-browsing software while enforcing limited access to the Internet. Then, say at 3:00 PM, the laboratory is used to teach a MATH class based on the Pearson's MyMathLab web-application. All such laboratory reconfigurations could be *automatically* performed with respect to some schedule by means of reassigning different set of virtual machines to the same set of thin terminals located in the classroom.

1.3 Technology

The infrastructure is implemented on *HP Proliant ML350* hardware platform (three servers) and two types of thin-terminals: *Wyse V10L* and *LTSP Term 1720*. From the perspective the virtualization technology, the infrastructure is based upon two hypervisor implementations: Citrix XenServer and VMWare ESXi 5. The majority of servers are Linux (Debian and Ubuntu) based ones. The OpenSource ideology is preferred throughout the design against any proprietary analog.

1.4 Licenses

The Department of Computer Science of Georgia Southwestern State University is a member of two academic alliances: MSDN (Microsoft) and VMAP (VMWare). Membership in these alliances helps reducing the cost of the infrastructure while providing convenient and hassle-free access to a variety of the software for both student and faculty members of the department.

2. Conclusions

The proposed solution has partially been implemented at the School of Computing and Mathematics in Georgia Southwestern State University. The work still in progress but several results can be outlined as accomplished:

1) A *Cloud Computing* course was successfully taught in Spring 2012 term. Students were individually working in the Microsoft Azure cloud computing environment while having *administrative privilege* on the Windows Server 2008 R2 machine and *independent account-specific storages* for project files.
2) A *Computer Security* course, in particular Network Security laboratory, was successfully conducted in Summer 2012 term. Students were divided into three groups so there were three *independent virtual networks* with three corresponding sets of victim, attacker, and router workstations.
3) The School Document Management Server was deployed back in Spring 2011 and successfully functions since that time. Its database along with document data storage is getting periodically backed up making last 30 days of the archive available on demand.

Understanding Academic Assessment Mechanisms:

A Case Study on the Impact of Western Education Influences in Asia

Christine Amaldas[1], **Avinash Shankaranarayanan**[2], **and Prof Kiminori Gemba**[1]
[1]Graduate School of Technology Management, Ritsumeikan University, Japan
[2]Aries Greenergie Enterprise (P) Ltd., India

Abstract— *The research focuses on various factors influencing intrinsic and extrinsic motivational levels in international tenured academics and contractual teaching staff in Management and Business departments towards the adoption of Management of Technology (MOT) related methodologies. A set of hypotheses were defined to deduce the relationship between teaching and adoption of MOT as a framework. This research implies that job performance of International academics strongly depends on various motivational levels. The study was conducted using the interaction survey method with in-depth personal interviews consisting of open ended questions with 250 International academics (respondents consisting of Japanese and Foreign teaching staff) chosen for the study based in Japan. Hence, policy recommendations and decision making should be dealt with prudence and pragmatism.*

Keywords: Academics, Extrinsic, Intrinsic, Japanese, Management of Technology.

1. Introduction

Education is considered as one of the most important developmental issues facing Asiatic countries. The youth of these countries are dependent on education as their future resources towards development of their respective societies. Education obtained major attention and big budgets from all walks of life as an artificial scenario created to make education available to every individual‘s requirements and beliefs towards societal development through economic incentives. Utilizing this so called ‘educational system’, students and professionals were expected to develop themselves to ensure their future along with the future of their families and their surrounding societies. In conjunction to such an aspiration, it is believed that management studies related to students‘ academic excellence and professional achievements utilizes a process of systematic teaching systems imposing numerous examples from relevant business experiences. Supporting materials enabled the mastery of students in such a way leading to business pathways taking it to the next level. Due to this, many ideas and initiatives (such as the use of Six Sigma in businesses) have been generated to improve the quality of the education, thus, ensuring the quality of the teaching and learning processes to be competent with the latest development and research findings. In terms of policy making, all educational policy makers must plan various strategies besides providing sufficient learning facilities (technological infrastructures) and trained academics (faculty support) within institutions equally.

This study was conducted using the interaction survey method with in-depth personal interviews consisting of open ended questions with 250 International academics (respondents consisting of Japanese and Foreign teaching staff) chosen for the study based in Japan. The instruments used in the study were specifically developed to encompass all tasks and duties based on teaching specifications utilizing the Management of Technology (MOT) framework used at the department of Business and Management Studies and the department of Information and Communications Technology (ICT). Specific elements explored were factors influencing MOT adoption and usage in teaching environments including: Gender, Skillset, Age, Class Handling, Teaching Experience and Intrinsic and Extrinsic motivation levels such as Competitiveness, Popularity of students, Passing rates, Job Satisfaction, Teaching Interests and Workload. The discussions in this paper focus on the factors that influence the intrinsic and extrinsic motivation levels in International tenured academics and contractual teaching staff in Management and Business departments with respect to the adoption of MOT related methodologies. The studies focal point is the assessment of the influence on learning and teaching mechanisms of academics with respect to the following variables: Gender, Skillset and Age which is evaluated in this research. The research question was defined to test the variables proposed to deduce the relationship between teaching and adoption of MOT as a framework in Classroom Based Learning Environments (CBLE).

This research implies that the job performance of the international academics strongly depends on various motivational levels with respective to the roles of international teaching staff that were inter-connected to each other. The introduction served as an example of current teaching standards being presently deployed in various international institutions; Section 2 relates to the review of literature discussing about the penetration of MOT as a framework, digital technologies and ICT related services responsible for the Management and training of Business students who are absorbed into the industries with this know-how (tacit knowledge); Section 3 defines the research question, objectives and significance of the study; Section 4 defines the hypotheses related to the variables tested for correctness in this study; the rest of the paper statistically explores the realms of MOT influence and adoption in various streams of teaching in various departments followed by verification of primary and secondary data collected using the instruments defined. This is followed by the findings, limitations, delimitation's and conclusion of the study.

1.1 Literature

MOT can be considered at two levels: Managing at the macro-level of countries or at the micro-level of companies. At the macro-level, it is "concerned with the setting and implementation of policies to deal with technological development and utilization, and the impact of technology on society, organizations, individuals and nature. It aims to stimulate innovation, create economic growth, and to foster responsible use of technology for the benefit of humankind" (Khalil, 1993). At the micro-level of the firm, it is "concerned with the planning, development, and implementation of technological capabilities to shape and accomplish the operational and strategic objectives of an organization" (NRC, 1987). The proper Management of Technology requires progressive policies in support of technological development. It is a precursor for sustainable economic growth (Berman & Khalil, 1992).

Kanz and Lam (1996) define MOT at a more simplistic societal leveling linking the underlying use of technologies working together with people to achieve systematic methods towards applying knowledge in order to produce goods and services. On one hand, Kanz and Lam (1996) take the role of defining MOT using a more practical viewpoint; and on the other hand, Tabbada (2000) takes on a higher tier of academic perspective defining MOT as an educational research tool fabricated to manage the technology components of individual product life cycles, capitalize on process technology to gain competitive advantage, and integrate product and process technologies. In this paper, MOT is defined as an educational framework that is utilized in Science, Engineering and Management departments towards acquiring knowledge for effective standardization and delivery of industry recognized courses as the core components in tertiary institutions. Utilizing this framework, the researcher intends to provide the necessary framework for multinational organizations and educational institutions to gain competitive training advantages towards optimizing desired levels of growth and performance on its students as well as the future employees.

Proper MOT is what creates wealth and prosperity for nations, companies, and individuals. It forms the foundation for economic growth and determines national and organizational competitiveness. Multinational corporations and all successful enterprises realize that business competitiveness is no longer a matter of luxury but a matter of survival in an increasingly global, fiercely competitive marketplace. The distinction between developed and developing economies lies primarily in the ability to effectively manage resources and technological assets (Khalil & Ezzat, 2001). It demonstrates the impact of technology on fostering national development and improving competitiveness. It contrasts the wealth created in industrialized countries and their companies with the wealth created in less industrialized countries. It also defines the necessary ingredients for enterprise competitiveness in the current environment (Khalil & Ezzat, 2001). The wealth creating capacity of nations as well as companies does not only depend on the acquisition of technology but more importantly on the ability to manage their resources and technological assets. It is only when technology reaches the marketplace and is accepted and paid for by customers that wealth is created. This is the essence of the relatively new MOT discipline. The challenge that faces every country and every company today is how to harness technology for wealth creation in this era of exponential growth in technology. In this context, the definition of wealth creation includes the well being of government and non-profit organizations, the quality of life, and environmental concerns. It is important that the economic growth achieved through technological progress be sustainable on the economic, social, and environmental dimensions, (Khalil & Ezzat, 2001).

Nations around the world have started focusing on economic growth, as evidenced by the number of countries that have signed up for the General Agreement on Tariff and Trade or applied to the World Trade Organization. Since technology and the management of technology are recognized as major drivers of economic growth, the economic development of the less-developed countries and the issue of wealth distribution have become a global problem; integral to that is how to manage technology in a global economy, (Khalil & Ezzat, 2001).

In a series of workshops held under the auspices of the National Science Foundation (NSF) and the International Association for MOT (Khalil 1998, 2001), primary drivers of change in the twenty-first century were identified which can be classified under seven main technological areas: (1) Technology; (2) Changes in Business Environment;(3) Communication, Integration, and Collaboration; (4) Strategic Directions of Industry; (5) Changes in Organizational Structure; (6) Financial Sector Structure; and (7) Education and Training. The intensity of changes in technology is projected to continue into the next century. The rapid communication and diffusion capabilities of information will grow, resulting from technological advancements. The Internet, for example, is an avenue for the provision of extensive amounts of raw information, accessible by many people. Through it, interactions can be two-way or multi-way and will foster collaboration in departments within organizations as well as between organizations. In addition to information technologies, another technical area where major and rapid change will continue is molecular biology. Industries such as healthcare, agriculture, and genetic-based corporations are likely to be privy to great changes in technology within genetic engineering, biotechnology, and connecting to nanotechnology (Khalil & Ezzat, 2001). The millennium has opened its doors to the numerous technological arenas. Human beings must be motivated to learn and use these technologies. The venue for fostering these will be a tertiary institution where motivation should be in every individual's bloodstream. An institution is viewed by the researcher as a temple of knowledge where academics are the pillars to strengthen the temple. If the pillars are weak, the temple would collapse sooner or later.

Motivation at the workplace is determined by three principles namely,intrinsic motivation, extrinsic motivation and basic psychological need (Hackman J.R. & Oldham G.R, 1976). Motivation refers to action, direction, intensity and behavioral persistence (Gordon P.H. et al, 1983). The Two-Factor theory explored by (Herzberg, 1968, 1966) puts stress on the importance of intrinsic and extrinsic motivations levels wherein psychologists indicate that intrinsic motivation is the direct result relationship between employees and their work which is usually self-instigating. Extrinsic motivation is formed based on the external environment of the workplace usually controlled by several variables directly or indirectly by several levels of stakeholders.

The relationship between extrinsic and intrinsic motivation levels indicates that when both results of intrinsic and extrinsic are attractive, it will contribute positively to the motivation levels of the individual. In this context, career planning is directly affected by the employee's motivation levels that is influenced by his or her work environment (Holden 1990, 1991). Employees, among others, are influenced by several factors, like environment and ecology, perception, memory, cognitive development, emotion, behavior and personality (Huitt, 2001). Rewards especially, can stimulate motivation and generate the actions required to do an activity (Enabou & Tirole, 2003). At the same time, emotional and material support enable staff to become more committed (Brandt, 1995) in doing an activity. Besides that, good and close relationships between staff also influence the commitment within staff (Nordin, 1997). In relation to that, training's given to teaching staff also improves commitment and job performance within them (Sahandri, 2008). Good communication in organizations also encourage job satisfaction, job performance and job commitment and these can be considered as assets in an organization (Joni & Balan, 2001).

1.2 Problem Statement

The paper looks directly at the motivation levels of International academic staff and the level at which they are comfortable in adopting and utilizing MOT framework in their teaching and curriculum development. Hence, the research question is formulated based on the field work under taken as follows:

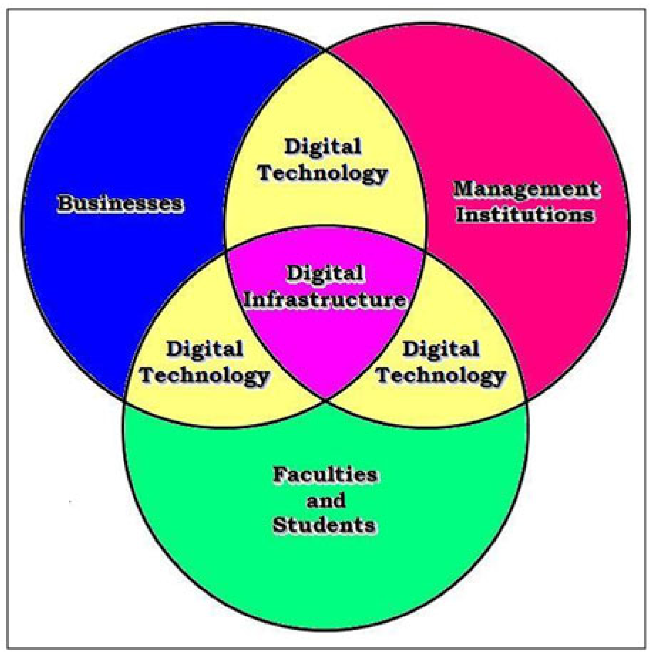
> *"Does the adoption of MOT as a framework have any effect on the motivation levels of International academics in Management related departments?"*

To answer this question the following assumptions and objectives are formulated in the following sections. The rest of the paper investigates the proposed research question using empirical instruments for evaluation and validation.

2. Conceptual Framework

This paper is specifically linked to finding out the MOT based motivation and training awareness among International and Local academics or teaching staff in Management departments and tertiary institutions at a well known International University in Japan. This study considers the various motivational factors and theories related with the function and roles of International academics and staff with the confluence of MOT and its related technologies as shown in Figure 1.

Fig. 1: Hypothetical Conceptual Framework



Source: Designed by the Author

Based on the Conceptual framework in Figure 1, the roles and tasks done by the tenured and contractual staff consists of personal factors influencing the work culture and professional ethics of the selected organization. The relationship between individual and the institution needs be illustrated using the motivational theories and expectations directly influencing the performance of the individuals in the selected institutions. The whole interaction between the formulated research question and the Conceptual framework above will be elaborated by the determined variables in the instruments of the study. The resulting model contains four main modules encapsulating the system proposed: (1) Businesses, (2) Management institutions, (3) Faculties and (4) Students. The cornerstone of all changes in assessments of international students is the academics (grassroots of the institutions) at universities. It may be argued that teaching, learning and assessment strategies will not be relevant to internationalization if they do not reflect the generic principles of good practice in higher education. This implies that the development of inclusive learning, teaching and assessment strategies among teaching staff require them to develop new skills, knowledge, attitudes and values. In any academic environment, academics may have legitimate concern that there is no space in their subject for a holistic approach to internationalization. Institutions like University of South Australia have responded to such concerns with what is known as the infusion approach. Using its graduate qualities as a framework for curriculum development, a team-based approach to international teaching has been adopted. It provides clarification of what internationalization means in different subjects within a discipline. The model proposed here enables the interaction between the various components using the introductory information systems that are transformed into knowledge by student activities and this in turn feeds various Internet Rich Applications (IRAs) with the acquired knowledge through project activities leading to production of new knowledge enabled by motivational factors sustained by communication from other participants, students or faculty.

3. Methodology

In social research, countless number of methodologies can be employed towards primary and secondary data collection. This includes methods such as experimental research, survey research, ethnography, phenomenological research, grounded theory, heuristic inquiry, action research, discourse analysis, feminist standpoint research, and many others (Crotty, 1998:5). However, research in the education domain tend to make use of a selection of common methodologies, namely Experiment, Survey, Case study, Grounded theory, Ethnography and Action research, each suiting specific studies.

3.1 Design of Study

This study has been carried out at an International University in Japan at the Center for Business and Management, during the winter semester of 2011. The study design encompasses both quantitative and qualitative approaches. For the quantitative approach, the survey descriptive design is used. Descriptive research, also known as statistical research, describes data and characteristics about the population or phenomenon being studied. Descriptive research answers the questions who, what, where, when, why and how...(Babbie, 1989). Babbie identifies exploration, description and explanation as the three purposes of social science research. Descriptive research classifies phenomena by providing an illustration of a variable that is to be observed using statistical determiners to perceive the distribution, pole and the data tendency. Two sets of instruments were used here: (i) Instruments for tenured academics and contractual teaching staffs and (ii) The academics response instruments for the implementation and usage of MOT as a framework. The quantitative variables measured in this study are demographic backgrounds, motivation levels, roles, personal factors and environmental factors which influence the respondents‘ motivation.

3.2 Instruments used

The instruments of the study consists of the Motivation Instrument and the Role of International academics in the adoption and usage of MOT and related technologies. The Motivation Instrument (which is based on our Conceptual framework) and the roles of International academics are discussed: demographic information, age, experience, position, salary and awards received and intrinsic and extrinsic motivation levels related to summarizing the roles of the respondents. Both of these parts will form the cumulative score to view the performance of the selected respondents.

4. Findings and Discussions

The survey has been conducted in several stages. First, the media department was contacted to access the list of possible academic staff and their respective contact details were attained. Next, individual appointments with the available tenured staff and the contractual staff were calendered. At the appointment, the survey took place in two steps: (1) The participant‘s agreement to undertake the interview with voice-recording was confirmed as the first step; (2) The interview took place on the spot, that is, the section of the office or the staff room, and the questionnaire was delivered and collected after fifteen minutes of window interval. The response in the interview was taken down on a voice-recorder and a sheet of paper with pre-set questions by the interviewer. The interviewee could have a copy made available to them upon their request. Finally, the information collected was processed and analyzed at various levels.

Around 250 participants of the two groups were initially expected to be the respondents in the survey. There were many barriers towards the collection and completion of the data collection. The participants were most likely agreeable to participate after lecture hours for the research.

The data collection methodology applied includes Personal Interviews, Surveys and Questionnaires employed during the author‘s involvement as an academic staff member in the respective institutions. The names of these institutions are not being disclosed here due to contractual obligations when undertaking this research. Hence, for simplicity the tertiary institution (International University in Japan) herein will be referred to as Institution A.

Institution A, comprises of 250 respondents (134 permanent teaching staff and 116 contractual teaching staff as our respondents out of approximately 400 teaching staff) playing an important role in internationalizing the curriculum and bringing International university degree programs and related content to students in the selected departments. The institution is affiliated to the Ministry of Education in Japan and has its ties to numerous International University linkages such as Australia, Singapore, India, China and Malaysia inculcating international management perspectives and technological superiority. Thus, programs and activities of all Management related institutions depended a lot on commitment, work culture and satisfaction, competency, attitude and motivation of all its staff and students involved. For analyzing the data, the 't-test' and one way ANOVA statistical methods were applied for evaluating the gathered data.

4.1 Significance of MOT - Age

For a given set of faculty of Management in an International university, we were interested in observing if there was really a statistically significant relationship and difference between the Age groupings and the MOT adoption. The results obtained are based on Age groupings and their opinions obtained from Table 1 wherein independent samples of t-test were carried out and the results are shown in Table 1. To test our research question a one way ANOVA was used to obtain the results along with the Mean, Standard Deviation (S.D.) and Percent values as shown in Tables 1,and 2.

Table 1: Mean, S.D. Based on Age Groupings

Age Groups	N	Mean	SD.
25 - 30	61	55.619	9.478
30 - 35	77	57.479	10.435
35 - 40	66	59.595	10.208
40 and Above	46	61.197	9.835

Source: As calculated by the Authors

Table 2: Test for Significance with Respect to Age

Variables	Sum of Sqs	MeanSq.	F-value	0.005 level
Between Groups	986.37	328.79	3.26	0.022 S
Within Groups	24789.26	100.76	3.26	0.022 S
Total	25775.63	249		

Source: As calculated by the Authors

Based on the results obtained as disclosed in Table 2, a statistically significant difference between Age Groupings were observed on in adopting MOT based technologies when teaching in Japan. The F-value from the Table 2, is 3.263 is found to be significant at 0.05 levels. It indicates that there is a significant difference among the academics when MOT based motivation and training awareness is considered with respect to their age. It indicates academics with above 40 years of age (M = 61.197) have greater MOT based motivation and training awareness compared to other age groups of academics. Similarly, academics within the age groups of 25 - 30 years (M = 55.619), 30 - 35 years (M = 57.479) and 35 - 40 years (m=59.595) seems to be less inclined towards MOT adoption. From Table 2, the F-value (3.263) is found to be significant and hence, the reason for above 40 years age academics having greater MOT based motivation and training awareness is that they are attending educational workshops, in service-training, refresher and orientation courses conducted by third party trainers and institutional level linkages with field experience gaining wider MOT adoption.

4.2 Significance of MOT - Gender

Based on the results obtained in Table 3, there is a statistically significance and differences between Gender and MOT adoption. To test the proposed research question the Mean, Standard Deviation (S.D.) and 't' values were calculated and given as follows:

Table 3: Test for Significance with Respect to Gender

Gender	N	Mean	SD.	t-value	0.005 level
Male	129	55.494	9.288	4.619	S
Female	121	61.212	10.285	4.619	S
ICT	134	57.355	9.793	1.533	NS
Social Science	116	59.328	10.544	1.533	NS

Source: As calculated by the Authors

From Table 3, the t value is calculated as 4.620 greater than the table value (1.96). It is found that there is a significant difference between male and female academics in MOT based motivation and training awareness at 0.05 of significance. From Table 3 based on Skillset,the 't' value was calculated as 1.532 less than the table value 1.96. It is found that there was no significant difference between ICT and Social Science academics in MOT based motivation and training awareness at 0.05 level of significance. Many studies include that the women are as successful as the men in most cases, which is clearly not the case in our study as the results indicate that women are less susceptible to be technology orientation than men in the case of Japan inside classroom based teaching environments.

5. Conclusion

Asiatic regions such as Japan with respect to International education still have scope for improving their adoption and utilization of technologies and content delivery in terms of technological development. MOT is a process requiring a change in mindset and outlook shifting towards a more strategic mindset starting with the admissions process where there is a need to improve and effectively manage technology at the policy level. Integrating technology strategy and business strategy are essential ingredients in proper management of technology (Khalil & Ezzat, 2001).

While students and faculty in the Asia Pacific may value the cultural capital that is gained by creating knowledge transfer at international and foreign institutions, the educational environment is highly volatile. Western influences in Asia is picking up by adopting foreign capital based educational enterprises that may open shop in Asia, by offering better incentives like migration to countries of host universities and planned career choices. In such a milieu, instead of the individual fitting into the organization, the organization would have to compete with other institutions for the same human capital.

Unfortunately, not all sectors of the economy would appreciate this reality with the same level of enthusiasm. High-performing enterprises and institutions would most likely rise up to the challenge and move along in accordance with the MOT framework with ease. Sadly, majority of our enterprises and institutions are still rigidly closed to the idea of changing and adapting to new paradigms which impair adoption of such frameworks at the policy level. On one hand, much work needs to be done and the pressure remains with the government to create the right environment for effective MOT adoption among all organizations. The challenge to academe, on the other hand, is to initiate curricular reforms that would address the MOT needs of Japan in this case. Meanwhile, businesses should be more open to collaboration where immediate monetary benefits may not be apparent and training along with international exposure from a university environment might be some sort of a luxury for its future employees.

Acknowledgments

First and foremost, I (Christine) would like to express the deepest pleasure to The Almighty God for keeping me mentally and physically sound to demonstrate my concern for global affairs and modern technology. I take this opportunity to extend my gratitude to the people who have been instrumental in the successful completion of this research. I would like to show my greatest appreciation to Professor Kiminori Gemba, my supervisor and mentor for his scholarly advice and encouragement with regards to research and teaching. Next, I would like to thank Professor Shuichi Ishida who introduced me to the Empirical Methods; Professor Tetsuaki Oda who guided me in Research Methods; and Professor Hamid R. Arabnia, WORLDCOMP'12 Congress. I would like to thank Ritsumeikan University, Gakkai Grant and ASTER Scholarship towards financial support rendered for the research work.

References

[1] Abiad, V.G., (1997, February). Innovatronix, Inc. ASEAN business case studies (Case study No.6). Antwerp: Centre for ASEAN Studies and the Centre for International Managementand Development.

[2] Babbie, E., (1989). The Practice of Social Research. 5th edition. Belmont CA: Wadsworth.

[3] Berman, E.M., & Khalil, T.M., 1992. "Technological Competitiveness in the Global Economy. A survey." International Journal of Technology Management, Vol 7, No. 415, pp. 347-358.

[4] Brandt, R., (1995),Punished by rewards? A conversation with Alfie Kohn. Educational Leadership, 53, 13-16.

[5] Balboa, L.P. & Lutanco-Chua, J. (2006, January-February). Time to export. Entrepreneur, Retrieved 2012/01/12 from http://www.entrepreneur.com.ph.

[6] Crotty, M., 1998. The Foundations of Social Research, Allen & Unwin, NSW. DOST, 2009.DOSTâĂŹs funded P114-M SET-UP projects in 2009. Retrieved on 2011/07/28 from http://balita.ph/2010/02/08dosts-funded-p114-m-set-up-projects-in-2009/.

[7] Department of Science & Technology (2004). Canning greener laing. DOST-IPCT PostCPSuccess Stories, 1, Last Accessed on: 2011/06/28 from http://cptech.dost.gov.ph/success%20stories.php.

[8] Department of Trade & Industry (2007), Last Accessed on: 2011/08/18 from http://www.dti.gov.ph.

[9] Digital Infostructure Live (2007), Last Accessed on: 2011/11/08 from http://eenterprise.net/ibose/templates/themes/digitalinfo2007.

[10] Enabou, R.B. & Tirole, J., (2003). Intrinsic and extrinsic motivation. Review of Economic Studies, 70, 489-520.

[11] EITM,2007, European Institute of Technology Management, Last Accessed on: 2011/08/10 from http://www.mmd.eng.cam.ac.uk/ctm/eitm/index.html.

[12] Gordon, P.H. et. al., (1983). Is Teaching a profession. Bedford Way Papers, University of London, U.K.

[13] Hackman, J.R. & Oldham G.R. (1976). Motivation through the design of work: Test of a theory.Organizational Behavior & Human performance, 16, 250-279.

[14] Herzberg, F., 1968. "One more time: how do you motivate employees?", Harvard Business Review, vol. 46, issue 1, pp. 53âĂŞ62.

[15] Hertzberg, F., (1966). Work and Nature of Man. Cleveland: World Publishing.

[16] Huitt, W., (2001). Motivation to learn: An overview. Educational Psychology Interactive.

[17] Valdosta, GA: Valdosta State University, Last Accessed on: 2011/09/09 from http://chiron.valdosta.edu/whuitt/col/motivation/motivate.html.

[18] Innovatronix, Inc. (2007), Last Accessed on: 2012/02/12 from <http://www.innovatronix.com>

[19] Isaac, S. & Michael, W.B., (1979). Handbook in research and evaluation: A collection of principles, methods, and strategies useful in planning design and evaluation of sciences.Cetakulang.Pub. San Diago.

[20] Jenders, S., (2010, March 31). Evaluation of the SMED Plan 2004-2010.Department of Tradeand Industry and German Technical Cooperation.

[21] Kanz, J. & Lam, L. (1996). Technology, strategy and competitiveness: An institutional managerial perspective. In. G. Gaynor (Ed.), Handbook of technology management. NewYork: McGraw-Hill.

[22] Khalil, T. (2000). Management of technology, the key to competitiveness and wealth creation.Boston, Mass.: McGraw-Hill.

[23] Boon, L., (1983). The relationship between stress, satisfaction and performance within employees in a timber company. Latihan Ilmiah Bangi, UKM.

[24] Nordin, A., (1997). Practicum guidance and supervision by lecturers and academics of hope and reality guidance: an empirical study. Academics Education Practicum seminar working paper 1997: Century Hotel, Melaka.

[25] Manalo, G.R., (2006, September). The art of risk-taking.Entrepreneur, Last Accessed on: 12/02/2012 from http://www.entrepreneur.com.ph.

[26] Maneja, R., Poso, N. & Ramoso, C. (2004, March). A case study in TM 241: Technology assimilation in digital infrastructure systems and consulting corporation (Unpublished Case Study). Paper submitted in partial fulfillment of the requirements of Technology Management 241,

a subject under the Master of Technology Management program of the Technology Management Center of the University of the Philippines.
[27] MoonDish (2007), Last Accessed on: on 2012/02/12 from http://web.295.ca/~juluism/moondish/About%20Us.html.
[28] Merriam, S.B., (1997). Qualitative Research and Case Study in Education, San Francisco: Jossey-Bass.
[29] Merriam, S.B., (1998). Qualitative Research and Case Study Applications in Education, San Francisco: Jossey-Bass.
[30] Merriam, S.B., (2002). Qualitative Research in Practice,San Francisco: Jossey-Bass.
[31] NJ Open City (2007), Last Accessed on: 2012/01/04 from http://www.njopencity.com
National Research Council, 1987. "Management of Technology: The Hidden Competitive Advantage." NRC press, Washington, DC.
[32] Omrod, J.E., (2003). Educational Psychology: Developing learners. Edisi Keempat. New York:Merrill Prentice Hall.
[33] Poso, N.N., (2004, February). Philippine innovation success stories #2: Infostructure, Info Journal on eLearning, Manila, Philippines, Last Accessed on: 24/02/2012 from http://203.131.182.35/ibose/images/isc/infojournal
[34] Reyes, R.R., (2007, July). Businessmen highlight the role of innovation to boost growth. SME News, Last Accessed on: 08/02/2012 from http://www.sme.com.ph
[35] Salazar, T., (2007, February 18). Firms come up with innovations. Inquirer Opinion. Last Accessed on: 08/02/2012 from http://services.inquirer.net
[36] Sharifah, Balan, J., (2001). The Relationship between Communication in an Organization with Job Satisfaction, Work Performance and Work Commitment, Universiti Kebangsaan Malaysia (UKM) Bangi.
[37] Sahandri, M.G.H., (1998). An Analysis of Trainee Academics Commitment Pattern, Uni. Kebangsaan Malaysia (UKM), Bangi.
[38] Sahandri, M.G.H., (2008). Academics Activity Centre (TAC) A Catalyst of Excellence. First Speech of National TAC Conference, Seremban.
[39] Setup, (2006).Small enterprises technology upgrading program (SETUP). Last Accessed on: 08/02/2012 from http://setup.dost.gov.ph
[40] SMED, (2010). Small and Medium Enterprise Development (SMED) Council, SME Development Plan 2004-2010.
[41] Tabbada, Jose P., (2000). Management of technology as a field of study. In E. Patalinghug, J.Tabbada & E. Zamora, Managing technology for global competitiveness, pp. 254-265. Quezon City: UP Press.
Khalil & Ezzat, (2001). International Journal of Technology Management, Management of technology and responsive policies in a new economy, Vol 32, Issue 1, pp. 88 - 111.
Khalil, T.M., 1993. "Management of Technology Education for the 21st Century." In Sumanth, D., et. al. (Eds), Productivity and Quality Frontiers, IIE, Norcross, GA.
Khalil, T.M. & Ezzat, H.A., 2001, Emerging New Economy - Responsive Policies, Vienna Global Forum, Vienna International Centre, Austria, 29 - 30 May 2001.

Christine Amaldas has been an active researcher and academic at various International tertiary institutions. She has been the author of several Book Chapters, Conferences and Journal periodicals spanning the Asia Pacific region. She has chaired and presided over a number of conferences and has given numerous keynote speeches and guest talks in Singapore, India, Japan, USA, Vietnam and Australia. She specializes in Asia Pacific Studies, Education, IT Governance, Security and Fraudulence, Ethics and Ethical Governance in ICT, Holistic Medicine and Energy Healing and Governance (Corporate, Public, IT and Dynamic). In 2010, she received the 2010 Teaching and Research Award from the Royal Melbourne Institute of Technology International University for her excellent teaching methods in the institution. She was also the recipient of the prestigious Best Paper Award at the ANNA World Congress 2011, Chennai, India. She is currently a doctoral student and researcher at the Ritsumeikan University, Japan and the Managing Director of The Journal, Singapore.

Avinash Shankaranarayanan has been actively involved in publishing several Books, Book chapters, research papers, reviews and periodicals in International Conferences and Journals of high standards such as the IEEE and ACM. He has presented and presided as Program Committee Member and Program / Conference Chair over a number of International Conferences and has been invited as a Guest Speaker in various countries such as Singapore, India, Australia, USA, Japan, Mauritius, Europe, the United Kingdom and many parts of South Asia. His research areas includes: High performance Grid Computing, Higher Education, Bioinformatics, Material Flow Management, and Renewable Energy Systems. He is currently the Director of Aries Group of Companies, India and has been freelancing as a Consultant for numerous companies overseas.

Professor Kiminori Gemba is a leading researcher in Japan and his major is innovation theory and strategic management. His research areas include analyzing the innovation strategy of a "service innovation", with the main object of his research being to determine the source of the competitiveness of service innovations. He received his Master’s degree in Engineering from the University of Tokyo followed by a PhD also from the University of Tokyo, where his doctoral thesis was entitled "The Dynamics of Diversification on Japanese Industries". He is currently a Professor at the Graduate School of Technology Management, Ritsumeikan University.

Japanese Sojourners in Higher Education Reform for Internationalization in Contemporary Japan

Marystella Amaldas[1], [2] **Christine Amaldas, and Avinash Shankaranarayanan**[3]
[1]Graduate School of Technology Management, Ritsumeikan University, Japan
[2]Ritsumeikan Asia Pacific University, Beppushi, Japan
[3]Aries Greenergie Enterprise (P) Ltd., India

Abstract— *Internationalization of higher education is a strategic theme in the contemporary educational reforms of Japan. Both at national and university levels, many policies and projects have been introduced in Japan. The internationalization of the Japanese students has been pursued by four distinct activities: (a) internationalizing the curriculum at Japan's universities; (b) introduction of foreign languages; (c) sending Japanese students for short and long term overseas courses; (d) the welcome extended by Japan to more than a hundred thousand international students. All these activities are often carried out either fully or partially depending on the emphasis placed on internationalization at various Japanese universities. The paper utilizes secondary data to describe the expected outcomes providing the theoretical framework for providing important insights into reforming study abroad preparations; expected outcomes and the policies associated with study abroad projects at Japanese universities.*

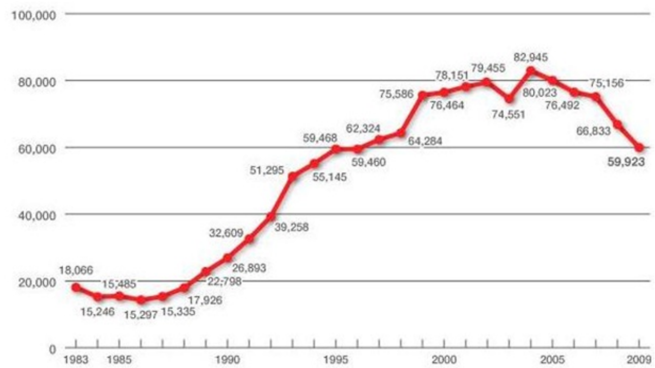
Keywords: Internationalization, Globalization, Higher Education, Study Abroad, Japanese students, Sojourners.

1. Introduction

In order to promote sustainable development, within the global knowledge-based society, the Japanese government is urging the nation's higher education institutions to internationalize. Although a selected number of Japanese universities are highly regarded in the world university rankings, many observers argue that the majority of Japanese higher education institutions are not sufficiently internationalized compared with those of other industrialized countries. However, the dynamics and structure of internationalization is highly diverse and well considered policy initiatives are required to guide and foster the process in respective institutions (Yonezawa, A., Akiba, H. & Hirouchi, D., 2009).

Between 2004 and 2005 the total number of Japanese students studying at U.S. universities and colleges was 42,215. Although there was a significant decline by 11.2 percent between 2003 - 2004, the number of Japanese students in 2004 - 2005 increased by 3.4 percent from the previous year. Based on a comprehensive survey of reviews of internationalization in Japanese universities, the perceptions of Japanese university leaders in terms of the

Fig. 1: Number of Japanese Students Studying Overseas, 1983 - 2009



Source: Newspapers in Japan, 2012

extent to which they reflect an "international" character. The present context and possible roles of government and institutions for internationalization in Japanese higher education are also explored (Yonezawa, A., Akiba, H. & Hirouchi, D., 2009). A paradigm for rigorous scientific assessment of study abroad programs, with the focus being on how study abroad experiences affect psychological constructs as opposed to looking solely at study-abroad-related outcomes has been proposed by Mark McLeod and Philip Wainwright, (2009).

The need for autonomous initiatives on the part of universities and academics themselves for the internationalization of higher education need to be emphasized. The Japanese government and the nation's higher education institutions are still in the process of discovering their identities amidst rapidly changing regional circumstances. Consequently, the direction of national and institutional strategies has often been observed to be unsettled and inconsistent. In order to sustain a continuous internationalization process, dynamic initiatives by academics and universities to enhance knowledge creation and exchange are indispensable.

In the case of Japan, adding to a strong identity based on an advanced higher education system, and continuous development of neighboring countries in terms of their science and technology sectors as well as their industrial and service economies is having a significant impact on policy direction and the actual internationalization process. (Akiyoshi, Y., 2009). This paper examines the efforts undertaken by Japanese institutions to internationalize the minds of university students by sending them abroad for a period of time as part of the university curriculum. The first section of the paper examines the learning that takes place amongst Japanese students when they go abroad to learn English or subjects taught in English at overseas universities. The following section examines objectives such as the extra-linguistic factors, the pre and post-sojourn reflections, the impact of different locations on learning and the knowledge that Japanese students gather in terms of societal culture and expectations. The final section examines the effects of gender and the different overseas contexts upon the learning of Japanese students locally and globally.

2. Purpose of the Study

In the last two decades, there has been considerable research on study abroad and its effects on second language acquisition, cross-cultural psychology and intercultural/international relations, and identity expansion (See Freed, 1995, 1998; Collentine & Freed, 2004 for reviews; Coleman, 1998 for a review of European research on study abroad). As a result, it is widely accepted that language acquisition in a study abroad setting depends on a myriad of factors, one of the most important being the context in which the learning occurs. There is a general assumption that formal language learning paired with immersion in a natural speech community provides the best environment for language learning (Freed 1995, 1998). Freed (1995) explains that there has been an assumption held by students, teachers, parents and administrators that students who study abroad learn languages better than those who study in the classroom or those who are immersed in the second language without receiving formal instruction. On the basis of this assumption, a large and ever increasing number of undergraduate students seem to travel abroad using overseas exchange programs involving one semester (spanning 3 - 4 months overseas) to one year exchange programs abroad. This assumption, however, has never been convincingly supported by empirical research.

Miller and Ginsberg (1995) characterized learner and facilitator beliefs about the benefits of study abroad and language learning in the study abroad context as 'folk linguistic'. He notes that they often do not correlate and sometimes even contradict current understanding of language acquisition. While interest in the role of context in language acquisition has increased in recent years, many empirical questions about the nature of language learning in study abroad remain unaddressed (Collentine & Freed; 2004). Study abroad has been shown to impact learners psychologically, culturally and linguistically. Linguistically, study abroad has most often been implicated in the learning of oral communication proficiency (Ikeguchi, 1996; Segalowitz & Freed, 2004). In Ikeguchi‘s study, learners‘ self-rating of their oral communication proficiency gains closely matched that of their teachers. This suggests not only that study abroad facilitates the learning of oral communication, but that self-assessment may be an accurate means of measuring this language learning. Segalowitz and Freed (2004) found that study abroad learners made greater oral proficiency than learners who remained at home. Subcomponents of oral proficiency have also been shown to be influenced by study abroad. Fluency is one sub-area of oral communication that has been extensively studied in study abroad contexts. Freed et al. (2004) also found that learners who studied both in study abroad and in immersion contexts increased their fluency compared to learners in traditional programs. Study abroad has also been implicated in the development of narrative abilities and the production of more lexically dense language (Collentine, 2004).

Empirical research of study abroad has linked successful language learning in study abroad to individual factors including pre-study abroad proficiency. However, more proficient learners have also demonstrated learning language in study abroad (Lennon, 1990). Additional individual factors, including strategy use, might also mediate the effectiveness of study abroad for language learning. Research has shown that immersion in the target culture is of great value to learner‘s second language acquisition (SLA), especially in improving oral production ability (Collentine, 2004; Freed et. al., 2004; Isabelli-Garcia, 2003; Segalowitz & Freed, 2004). Besides the immersion in English language classroom, the amount of contact with native speakers is an important factor in the acquisition of socio-linguistic and socio-cultural knowledge (Lafford, 1995; Lapkin et. al., 1995). There are, however, inconsistencies in study abroad research since claims are made based on different acquisition aspects and distinct amounts of time spent abroad and the type of interaction between learners and native speakers is frequently not specified. Research on study abroad does not account for socio-linguistic dimensions and prevents insights into the nature of learner‘s language (Firth & Wagner, 1997). Despite the abundance of research on different aspects of formal classroom and out-of-class contact in study abroad settings, many questions remain as to what specific types of interactions students participate in both contexts and how this affects second language acquisition. The present study will focus on how students and their native speaker interlocutors engage in negotiations to deal with actual communication problems and how the process of negotiation itself affects students‘ opportunities for English language acquisition. An example of this can be observed in the following statements of (Pica, 1994: 494),

> "...the modification and restructuring of interaction that occurs when learners and their inter-

locutors anticipate, perceive, or express difficulties in message comprehensibility." - *Pica, 1994: 494*

When language learners travel overseas, they enter a very specialized learning environment that has been specifically constructed by a variety of locally negotiated social, cultural and political dimensions that affect their place in the host classroom and the degrees to which they participate in it. These local dynamics arise within macro-level deliberation between administrators and faculty during pre-program planning. The study will attempt to investigate how dynamics at both the micro and macro level shape student opportunities for the development of bilingual (multilingual) and bicultural (multicultural) identities across time and space.

3. Research Questions

The following observations are made based on macro and micro aspects of English language learning using primary and secondary data from field trips undertaken by the authors. The following research questions are raised in the study for validation and evaluation:

- A) What individual extra-linguistic factors (such as motivation, contact with the host culture outside the classroom and attitudes towards the host culture) can be related to development of oral English communication skills and accuracy?
- B) How do Japanese students negotiate with native speakers in their study abroad classroom and host country settings?

The research questions were formulated based on a series of interviews conducted at three sites namely the University of Hawaii(UH), University of British Columbia (UBC) and Singapore Management University (SMU) that answers the research questions shown above. The interviews described in the next section clearly indicates that "individual extra-linguistic factors" are related to the development of oral English communication skills and accuracy. It is also evident that students negotiate with native speakers in their study abroad context inside the classroom and within the host countries environmental settings. The same series of interview questions were repeated to a set of students to also answer the second research question from the aforementioned study sites.

4. Methodology

The methodology of research would be as follows: (a) three study-abroad contexts have been chosen for the observation of Japanese students. These are the Singapore Management University, the University of Hawaii at Manoa, and the University of British Columbia; (b) The researcher spent six months at the three study-abroad contexts; (c) Four focus group interviews were conducted with a total of 21 Japanese students at the three sites; (d) About twenty-three students, both males and females, were chosen for in-depth study of pre-sojourn, sojourn and post-sojourn experiences; and (d) Seven interviews were conducted which included the professors who teach the sojourning Japanese students, program managers, coordinators of programs, counsellor and the General Manager of the host families agency were interviewed in-depth to obtain additional perspectives on the research theme. Language learning strategies have been defined as "operations used by learners to aid the acquisition, storage and retrieval of information" (Oxford & Nyikos, 1989), or as "techniques, approaches, or deliberate actions that students take in order to facilitate the learning and recall of both linguistic and content as information" (Chamot, 1987:71). They can be seen as the tools that learners consciously employ in the cognitive process of acquiring new languages. Language learning strategies can be automatic or learned (Oxford, 1990) and are seen as integral to the process of learning a language. Research indicates that all language learners employ strategies. However, they differ in the number, type and frequency of strategies used. The current research, while using the existing literature, plans to go beyond and explore how Japanese students experience their international learning and what they perceive as having met their expectations.

5. Findings and Discussions

The study takes up a holistic approach to the observation of English language learners in the study abroad contexts of Japanese students. It looks at both linguistic factors as well as extra-linguistic factors that influences their identities in their English language acquisition process. Studies that examine learner‘s attitude, motivation and behavior in the host environment, linking these factors directly to linguistic development can show that learners may not magically become fluent speakers simply by being surrounded by the target language. One aspect of the host environment is the informal relationships contracted by the individual learner, which have been referred to as 'social networks' a term coined by Milroy (1987).

5.1 Outcomes of the Research Question (1)

One of the aims of the study is to fill the gap that exists in the acquisition of English language in examining the effects of extra-linguistic influences. Specifically, this study will ask the following research question which is reiterated here from our previous section: (A) What individual extra-linguistic factors (such as motivation, contact with the host culture outside of the classroom and attitudes towards the host culture) can be related to the development of oral communication skills and accuracy? Let us take the following series of interviews conducted at the selected sites given to be self explanatory with references to the real world experiences elucidated by overseas Japanese students:

Scenario 1: *"I want to be a global person. My father is my major influence to go for the exchange* *program. He was a businessman in the Fujitsu Japanese Company. I want to communication with lots of people." [E San 22 years old Male, Osaka, Japan. SMU].*

Scenario 2:*"Improve my English. I wanted to be independent from my family and friends. Job opportunities. The first reason was I wanted learn specific field. I am Social Science student but I was learning many things like environment, tourism, cultural studies. I was not sure what I was studying so I wanted to focus on something. I went to Hawaii and took specific field...women studies. As a whole I want to improve my English." [C San 21 years old female, Kumamoto, Japan. UH].*

Scenario 3:*"Mother is working in international bank. I started learning English at 8 or 9 years old but just hobby. My mother can speak English well. When I was a child, my mother teach some English words. My father can speak English. My brother can speak English. "You go...go...go..." Father said go. My brother also said go. I want to speak English. Good job and my mother. I am interested in Media...International Journalism. One of the main reason is I want to speak English. The university in Japan, the students don't study hard but international university like UBC, most students study hard so I also want to study like international students...so I went to exchange. Main reason for coming is get good job." [M San 20 years old female, Itabashiku, Japan. UBC].*

Scenario 4:*"My major is International Business Administration. My specialization is Business. I used to want to be a flight attendant but now I want to do something using English. I want to do something using English. My mother and grandmother supported me financially. My father is not living with us anymore. My teacher motivated me to join this program." [N San 20 years old female, Hokaido Japan. UBC].*

5.2 Outcomes of the Research Question (2)

In native speaker (NS) - non-native speaker (NNS) interactions, interlocutors are frequently faced with problems in communications that stem from the learner‘s imperfect command of the English language. These problems often must be negotiated in order to achieve a satisfactory level of understanding. Reiterating from our previous section: B) How do Japanese students negotiate with native speakers in their study abroad classroom and host country settings? The following series of interviews conducted at the selected sites are self explanatory with references to the real world experiences elucidated by overseas Japanese students as observed by one of the researchers (Marystella):

Scenario 1:*"Actually before I applied to the exchange program. I joined a volunteer in Singapore. It‘s called the "Revitalizing" Actually I won the scholarship at the Temasek Foundation I have to join some volunteer in Singapore so "Revitalizing" is one of them. Temasek Foundation provide me accommodation also. We clean up the local people‘s house or room. They are very old people and they cannot speak English only Chinese. The room was very messy and very dirty so I tried to do good to them so once they know that I am Japanese, they recalled the history...the Japanese Occupation in Singapore so they treat me not very good. That was a bit very sad moment." [E San 22 years old male, Osaka, Japan. SMU].*

Scenario 2 (Day 1):*"During discussion the group member maybe did not like me because the discussion could not go smooth due to my English problem...so I had to try to talk and do my best. They were not upset but a girl in my group was really mean to me. Maybe she did not like me. She was from mainland. Hawaiian bus system is really nice. Some people on the wheelchair have difficulty come on the bus....I did not know that I was just sitting there like in the front behind the driver. I was suppose to move behind so some people stared at me and said that I should move in to give way. Some I went to Waikiki or Ala Moana to eat out I understand the American Hawaiian culture in Hawaii so it‘s ok."*

Scenario 2 (Day 2):*"I learnt a lot of academic things. The positive thing was I took so many women‘s studies there I got the confidence that I learnt one specific field, improved my English, I got many friends, I felt independent, got good exposure and I happy about my many experiences. I was living in Waikiki, the owner was Colombian, and other woman was there...she was Mexican. The owner was a lesbian. She had a girlfriend. The girlfriend did not like me so I had to move to another place. The family was Filipino family. They were living upstairs. Downstairs, I and an American girl who was a UH student too lived there. We were sharing the kitchen and bathroom and toilet and we had our own room. We cooked together but never shared with our owner. I make 'Okonomikake' and shared it with the American girl. She ate it and she made something like cheesy. We sometimes had party together. I am going to China to see her because she is learning Chinese in China now during summer vacation. The owner celebrates Christmas party. Two owners, son was in high school student. The other girl she was university student same age as me. She went to* *university in mainland. She came home during the winter breaks. The children are nice to me." [C San 21 years old female, Kumamoto, Japan UH]*

Scenario 3 (Day 1):*"Starbucks when I go to buy something the staff is rude...especially when the shop is crowded... She cannot understand my English...She said,"What...what...she shouted." [M San 20 years old Female, Itabashiku, Japan. UBC]*

Scenario 4 (Day 1):*"I came to September 1. I need to go out and see my Professor September 6. But I had to go by own. I missed the UBC bus route and I am looking for other bus but one student, I think he is Korean, he asked me, "What are you doing?"...I missing the bus so he tells me which bus I going there and then I could ride on the bus and get off and I can meet my Professor. After meeting, I lost my way again for go back UBC and I ask one man. And I ask him, "Do you know how to go to UBC and he said, "Yes". And he pick me to bus-stop also he telling other people who goes to UBC and two women goes to UBC so I follow them. I follow the man and two girls...I could go to UBC. Canadian people."*

Scenario 4: (Day 2)*"Supermarket. I bought a hot pan cake mix. That was box so I did not know that it is opened already. It is direct box so I did not know that so...."Can I change?" but she does not tell me that it's already opened... she said, only I cannot change...a little bit angry. I did not know that it's opened. But that is not miss. I did not know that the box is sold by open...directly box. Actually in Japan box and something.... But in Canada directly in box..in pan cake mix so I did not know that so I told the shop woman that it is already opened can I change this?" but she does not tell the reason why this is opened. Just say, I cannot change this." [N San 20 years old female, Hokaido Japan. UBC]*

6. Conclusion

Understanding the various factors influencing the Japanese in a study abroad context involves many variables. The so called "International exposure" experienced by these Japanese as depicted in the interviews conducted (a total of 28 interviews were observed) at the three sites: Singapore Management University (SMU), the University of Hawaii (UH) at Manoa, and the University of British Columbia (UBC), Vancouver, Canada had many similarities within the aspects of a dynamic host environment (Multi-dimensional). The study concludes the paper with the following findings from the field study undertaken at the three sites:

- The research adds to the current literature on what is known about how interaction takes place within the study abroad context with respect to motivation (Ushioda, 2001) and vice versa.
- The research provides us with an understanding of the complex relationship between motivation, acculturation and the development of social networks that ultimately provides opportunities for exposure to the target language and extends interactions that may be the driving force behind language acquisition in the English learning contexts.
- The study examines the 'overseas cultural dimensions' experienced by Japanese students as they encountered variations in the educational context (Training for example) as observed in the interviews expressed by some of the overseas Japanese students.
- The study would pedagogically help in determining what experiential learning strategies would be useful to Japanese students in gaining proficiency in speaking English as a foreign language.

Due to space constraints only the most important interviews are placed here in the paper. The study abroad sojourning of Japanese students was placed as an aspect of clearly crafted strategy of the Japanese government and universities as opposed to alternative learning experiences as clearly observed in the interviews conducted in this research.

Acknowledgements

I (Marystella) would like to take this opportunity to thank The Setsutaro Kobayashi Memorial Fund (FUJI XEROX) for their ïňĄnancial support with regards to the research undertaken. Next, I would like to give my special thanks to my supervisor and mentor, Professor Ko Yiu Chung for his esteemed help and kind support towards enabling me to travel and present my ïňĄndings at an International Conference at Las Vegas, Nevada, the United States of America. I would also like to extend my gratitude to all the Academics, Staff members and Students at all the three study sites, namely, University of Hawaii at Manoa (UH), University of British Columbia (UBC) and Singapore Management University (SMU) for their time and support rendered towards the research work undertaken. Next, I would like to take this opportunity to personally thank my fellow authors: Christine Amaldas and Avinash Shankaranarayanan for all their support and help with my area of research. Finally all my successes are due to the infinite ONE without whom nothing is possible.

References

[1] Chamot, A.U., (1987). 'The learning strategies of ESL students'. In A. Wenden and R. Rubin (eds), Learner Strategies in Language Learning, Englewood Cliffs, New Jersey: Prentice Hall, pp. 71-83.

[2] Collentine, J. & Freed, B.F. (2004). 'Learning context and its effects on second language acquisition'. Studies in Second Language Acquisition 26 (2), pp. 153-71.

[3] Coleman, J.A. (1998). ‘Language learning and study abroad: the European perspective’. Frontiers: The Interdisciplinary Journal of Study Abroad, IV, 167-203. Last Accessed on: 21 June, 2012. http://www.frontiersjournal.com/issues/vol4/vol4-07_Coleman.pdf.

[4] Freed, B.F. (1998). ‘An overview of issues and research in language learning in a study abroad setting’. Frontiers: The Interdisciplinary Journal of Study Abroad, IV, 31-60. Last Accessed on: 21 June, 2012. http://www.frontiersjournal.com/issues/vol4/vol4-02_Freed.pdf.

[5] Freed, B.F. (ed.). (1995). ‘Language learning and study abroad’: In B.F. Freed (ed.), Second Language Acquisition in a Study Abroad Context, Amsterdam: John Benjamins.

[6] Firth, A. and Wagner, J. (1997). ‘On discourse, communication and (some) fundamental concepts in SLA research’. The Modern Language Journal, 81 (3), pp. 285-300.

[7] Isabelli-Garcia, C.L. (2003). ‘Development of oral communication skills abroad’. Frontiers: The Interdisciplinary Journal of Study Abroad, 9, 149-173. Last Accessed on: 21 June, 2012. http://www.frontiersjournal.com/issues/vol9/vol9-07_isabelligarcia.htm.

[8] Ikeguchi, C.B. 1996. Self assessment and ESL competence of Japanese returnees. (ERIC Document Reproduction Service, No. ED 399 798).

[9] Lennon, P. (1990). ‘The advanced learner at large in the L2 community: Developments in spoken performance’. IRAL 28 (4), pp. 309-324.

[10] Lafford, B.A. (1995). ‘Getting into, through and out of a survival situation: A comparison of communicative strategies used by students studying Spanish abroad and at home’. In B.F. Freed (ed.), Second Language Acquisition in a Study Abroad Context. Philadelphia: John Benjamins, pp. 97-121.

[11] Lapkins, S., Hart, D. and Swain, M. (1995). ‘A Canadian interprovincial exchange: Evaluating the linguistic impact of a three-month stay in Quebec’. In B.F. Freed (ed.), Second Language Acquisition in a Study Abroad Context. Amsterdam: John Benjamins, pp. 67-94.

[12] McLeod, M. & Wainwright, P., (2009). ‘Researching the Study Abroad Experience’, Journal of Studies in International Education vol. 13, No. 1, pp. 66-71.

[13] Milroy, L. (1987). Language and Social Networks. Oxford: Blackwell.

[14] Miller, L. & Ginsbery, R.B. (1995). ‘Folk linguistic theories of language learning’: In B.F. Freed (ed.), Second Language Acquisition in a Study Abroad Context. Amsterdam: John Benjamins, pp. 293-315.

[15] Newspapers in Japan, 2012, Last Accessed on: 06 July, 2012. http://www.nippon.com/en/genre/economy/l00028

[16] Oxford, R.L. & Nyikos, M. (1989). ‘Variables affecting choice of language learning strategies by university students’. The Modern Language Journal 73, pp. 291-300.

[17] Oxford, R.L. (1990). ‘Language Learning Strategies: What Every Teacher Should Know’. New York: Newbury House.

[18] Pica, T. (1994). ‘Review article: Research on negotiation: What does it reveal about second language learning conditions, processes and outcomes?’. Language Learning, 44 (3), pp. 493-527.

[19] Segalowitz, N. & Freed B.F. (2004). ‘Context, contact and cognition in oral fluency acquisition: Learning Spanish at homeâĂİ and âĂIJstudy abroadâĂİ context’. Studies in Second Language Acquisition, 26 (2), pp. 173-199.

[20] Ushioda, Ema, 2001 ‘Motivation, language and identities and the L2 self: A theoretical overview’ in Zoltan Dornyei and Ema Ushioda (editors), Motivation, language and identities and the L2 self, Ontario, Canada: Multilingual Matters, pp. 1-8.

[21] Yonezawa Akiyoshi. (2009). ‘The Internationalization of Japanese Higher Education: Policy Debates and Realities’, Center for the Advancement of Higher Education, Tohoku University, No.9, pp. 199-219.

[22] Yonezawa, A., Akiba, H. & Hirouchi, D., (2009). ‘Japanese University Leader‘s Perceptions of Internationalization: The Role of Government in Review and Support’. Journal of Studies in International Education, June 2009; vol. 13, 2: pp. 125-142.

Marystella Amaldas is a doctoral student at the Graduate School of Asia Pacific Studies, Ritsumeikan Asia Pacific University (APU). Her research interests include early childhood education, globalization, Chinese student migration, study abroad experiences of Japanese students, and migrants from former French territories in India and many others. Some of her research publications include: International family migration; Management of globalization; and The impact of regional differences in English language learning experiences amongst Chinese students. Recently she was awarded the Best Paper Award for her work on “English Language Learning Experiences in China” at the ANNA World Congress 2011 in Chennai, India.

Christine Amaldas has been an active researcher and academic at various international tertiary institutions. She has been the author of several Book Chapters, Conferences and Journal periodicals spanning the Asia Pacific region. She has chaired and presided over a number of conferences and has given numerous keynote speeches and guest talks in Singapore, India, Japan, USA, Vietnam and Australia. She specializes in Asia Pacific Studies, Education, IT Governance, Security and Fraudulence, Ethics and Ethical Governance in ICT, Holistic Medicine and Energy Healing and Governance (Corporate, Public, IT and Dynamic). In 2010, she received the 2010 Teaching and Research Award from the Royal Melbourne Institute of Technology International University for her excellent teaching methods in the institution. She was also the recipient of the prestigious Best Paper Award at the ANNA World Congress 2011, Chennai, India. She is currently a doctoral student and researcher at the Ritsumeikan University, Japan and the Managing Director of The Journal, Singapore.

Avinash Shankaranarayanan has been actively involved in publishing several Books, Book chapters, research papers, reviews and periodicals in International Conferences and Journals of high standards such as the IEEE and ACM. He has presented and presided as Program Committee Member and Program / Conference Chair over a number of International Conferences and has been invited as a Guest Speaker in various countries such as Singapore, India, Australia, USA, Japan, Mauritius, Europe, the United Kingdom and many parts of South Asia. His research areas includes: High performance Grid Computing, Higher Education, Bioinformatics, Material Flow Management, and Renewable Energy Systems. He is currently the Director of Aries Group of Companies, India and has been freelancing as a consultant for numerous companies overseas.

Using the N-Body Problem to Engage Undergraduates in Parallel Programming

Ben White, Miriam Robinson, Chris Mitchell, Jens Mache
Department of Mathematical Sciences, Lewis & Clark College, Portland, Oregon, USA
{benwhite, miriamr, chrism, jmache}@lclark.edu

Abstract— *With the rise of multicore hardware, it is increasingly apparent that parallel programming is overdue for integration into undergraduate curricula. While some institutions are trying this, their assignments are often not interesting or not engaging, and the difficulty of low-level parallel languages might be overlooked. We have created a fun and compelling problem for students to parallelize: an N-body simulation. Although our example is under development, we suggest that more undergraduate institutions implement similar examples in the classroom.*

Keywords: parallel computing, n-body simulation, computer science education

1. Introduction

Parallel programming is often given cursory treatment in current undergraduate curricula, even though it is now necessary for any computer scientist who wishes to write programs that properly utilize multicore machines. The subject of parallel programming has been traditionally reserved for students pursuing graduate-level studies, but the rising importance of parallelism brings the need to introduce parallel concepts earlier on in computer science education. We aim to do this by creating exercises and materials that are targeted for undergraduate students.

Background research on recent efforts for bringing parallelism to the undergraduate classroom revealed two trends. First, assigned programming exercises were not very compelling. Often, these assignments included overused problems like matrix multiplication which are unlikely to catch student's interests and do not encourage experimentation. A student is likely to just "try to get the assignment done" if they do not care about it. Second, we noticed that many introductions to parallel programming involved lower-level abstractions like POSIX threads (Pthreads) [1] or Message Passing Interface (MPI) [2], which may obfuscate the higher-level parallelism concepts with a swath of low-level details. Students' fundamental understanding of parallelism may suffer as a result of being overwhelmed by these details.

This paper describes an exercise in which students use OpenMP to parallelize a graphical implementation of the N-body problem written in C. The N-body problem is an $O(N^2)$ algorithm for computing the motion of N bodies under the influence of physical forces. In addition to being visually interesting program, the naive algorithm for computing the forces is easy to understand and computationally expensive, making it a good fit for teaching parallel concepts to undergraduates. OpenMP [3] is an industry standard API for shared-memory parallel programming that is high-level and easy to grasp. Adding OpenMP directives to one's code causes the compiler to emit programs that will make use of threads to complete their work.

2. Related Work

One college or university that has already introduced parallel programming to undergraduates is the University of San Francisco, which has a lower-division elective course in the subject [4]. Students are taught the C programming language during the first month of the course, and then are taught parallel programming using MPI, Pthreads, and OpenMP. The course also uses an N-body simulation, but devotes a large amount of time to replacing the naive $O(N^2)$ implementation with a faster $O(N \log N)$ implementation that uses the Barnes-Hut algorithm [5]. Using Barnes-Hut is a valuable lesson in algorithm optimization, but not particularly useful for teaching parallel concepts to students.

At the University of Washington, a three-week introduction to parallelism and concurrency is taught in a sophomore level data structures course [6]. The students in this course learn about parallelism with Java's built-in Thread library. Since these threads have a large amount of overhead, they are ill-suited for small computations and do not offer much speedup for the types of programs that students are likely to write. Even though C and C++ offer fast threading that may provide students with a more satisfying speedup, at the University of Washington, students are already familiar with Java, so teaching the course in Java enables more time to be spent teaching parallelism concepts instead of a programming language.

A breadth-first course in parallel programming for undergraduates was introduced at Sonoma State University using three high-level languages: OpenMP, Intel Threading Building Blocks (TBB) [7], and CUDA [8], [9]. The labs and assignments used in this course were standard problems such as finding a maximum value in an array or incrementing all the elements of an array. While these are simple and easy to understand problems, they may be better suited for during a lecture instead of as a homework assignment.

At Lewis & Clark College, Mitchell *et al.* created a lab exercise for introducing the CUDA parallel programming language [10]. The exercise asks students to use CUDA to parallelize an animated implementation of John Conway's Game of Life. The problem was chosen for its strength in providing visual feedback and for its more recreational feel when compared to typical numerical problems. Because CUDA requires an NVIDIA graphics processing unit (GPU) to run, it may be impractical to teach this type of parallelism in an undergraduate environment that is not equipped with NVIDIA GPUs.

This paper builds on these works by combining two qualities that seem to be useful for teaching this subject: interesting and engaging examples, and languages that have high-level and easy to learn abstractions for parallelism.

3. The N-body Problem

The N-body problem simulates the movement of a collection of N bodies under the influence of physical forces. In our case, we simulate gravity in two dimensions. Each body in the system moves according to the net force exerted on it by the other $N-1$ bodies, yielding a computational complexity of $O(N^2)$. Our serial approach to the n-body problem uses numerical integration to estimate the positions of the bodies at regular time steps throughout the simulation. To turn the n-body problem into an assignment, after learning some OpenMP, students may be given our serial implementation and challenged to parallelize it with OpenMP directives to make it run as fast as possible. At a very high level, our serial implementation works like this:

```
 1 For each step:
 2    draw the bodies
 3    move the bodies one time step
```

Updating the position of the bodies for each time step involves using the positions and masses of the bodies to calculate the gravitational forces they exert on each other, and then using these net forces and body positions and body velocities to update the bodies with new positions and velocities. Pseudocode for progressing the simulation one time step looks like:

```
 4 For each body q:
 5    Initialize net force on q to zero
 6    For every body k besides q:
 7        Calculate the force k exerts
              on q
 8        Add this force to the net
              force on q
 9 For each body q:
10    Update q's position and velocity
```

Note the doubly-nested loop that calculates the net forces in the system and gives the algorithm an $O(N^2)$ runtime. The gravitational force exerted on a body q by another body k is given by:

$$F_{qk} = \frac{G m_q m_k}{r_{qk}} \tag{1}$$

where G is the gravitational constant, m_q and m_k are the masses of bodies q and k, respectively, and r_{qk} is the distance between the two bodies.

4. N-Body as an Assignment

We propose providing finished serial code and challenging the students to add OpenMP directives to gain the best speedup they can. Providing a serial implementation allows students to focus on the parallel task at hand and not having to worry about implementing the physics. Instructors could also challenge students to produce the fastest version in the class.

There are a number of ways a student might approach parallelizing the code.

The most obvious section to parallelize is the nested loop which computes the forces on each particle. This is done by adding a compiler directive for OpenMP in front of the loop on line 4 (missing proper private/shared clauses):

```
# pragma omp parallel for
    num_threads(n)
```

Here `n` is an integer indicating the number of threads in the thread pool. Using the `num_threads` clause allows the user to vary the number of threads to test the performance of the program.

The exercise affords a little bit of experimentation because there are a few ways to do this parallelism incorrectly and also correctly. For a complete serial implementation and for more information on parallelizing this code, please refer to our website [11].

5. Discussion

5.1 Potential Difficulties

Despite the apparent simplicity of inserting OpenMP directives to add parallelism, a careless approach will cause the student to make correctness errors. These mistakes will encourage them to reason about race conditions and variable scope (private/shared) to understand why their program has failed and to determine what works and what doesn't.

The first potential problem is that of shared and private variables. For example, a naive approach to the exercise is to simply insert a `parallel for` directive before the outer loop. However, since OpenMP automatically treats any variables declared outside of an OpenMP directive as shared, those variables will be visible and accessible by all threads. This can lead to race conditions between iterations of the loop and cause incorrect answers. A simple fix is to declare all critical variables as private. For example, if we have a variable `x` used inside the for loop, but it was declared outside the loop, we can make it private by adding the `private` clause:

```
# pragma omp parallel for private(x)
```

Another way to avoid this type of race condition is to implicitly set variables as private for OpenMP by placing their declaration inside the for loop.

A second problem students may come across is the issue of handling output. Since I/O is not inherently thread-safe, different threads attempting to print simultaneously can cause unexpected output. For example, two threads participating in a parallel loop with each thread trying to print `Answer from Thread N\n` might produce this jumble:

```
> Answer fromAnswer from Thread 2
>  Thread 1
>
```

In order to avoid this, students should ensure the print commands are serialized by placing them in code blocks with the `# pragma omp critical` directive. The `critical` directive ensures only one thread is executing the code in the block at any time, preventing simultaneous printing to the screen.

5.2 Expected Performance

We timed the N-Body program on an 8-core Sun server (with 64 hardware threads) with varying numbers of threads, 1000 bodies, and with termination occurring after 100 frames. The data we gathered should serve as a guide for the kind of performance increase students can expect to see. Table 1 shows the speedup of the parallelized force calculation using the OpenMP `parallel for` pragma. Speedup was defined as

$$S = \frac{T_1}{T_p}, \tag{2}$$

where T_1 is the execution time of the sequential program (one thread) and T_p is the execution time of the parallel program. The number of threads was controlled by varying the `num_threads` clause, and the runtimes were gathered by using the UNIX `time` command. The loops equivalent to those in the pseudocode on lines 4 and 9 were parallelized with a `parallel for` pragma.

6. Conclusion

While we have not classroom-tested this assignment yet, we anticipate that this example will be engaging and useful for students not interested in typical problems such as matrix multiplication. In addition to being a fun example, it is also useful in situations where students have diverse backgrounds and interests that may not include the usual computational mathematics problems. The usefulness of this paper should not be limited to implementing just the N-body problem either; we hope that other instructors will create their own interactive and interesting examples for their students.

Table 1: Speedup of the parallelized force calculation using the OpenMP `parallel for` pragma.

Number of Threads	Speedup
2	2.00
4	3.99
8	7.93
16	14.24
32	21.80
64	26.72

7. Acknowledgements

Partial support for this work was provided by the National Science Foundation's "Transforming Undergraduate Education in Science, Technology, Engineering and Mathematics (TUES)" program under Award No. 1044932, by the John S. Rogers Science Research Program and by the James F. and Marion L. Miller Foundation. We would also like to thank Quentin Froemke and Jingke Li.

References

[1] B. Barney. (2012) POSIX Threads Programming. [Online]. Available: https://computing.llnl.gov/tutorials/pthreads/

[2] (2012) The Message Passing Interface (MPI) standard. [Online]. Available: http://www.mcs.anl.gov/research/projects/mpi/

[3] L. Dagum and R. Menon, "OpenMP: an industry standard API for shared-memory programming," *Computational Science Engineering, IEEE*, vol. 5, no. 1, pp. 46 –55, 1998.

[4] P. Pacheco. (2012) Computer Science 202, Introduction to Parallel Computing. [Online]. Available: http://cs.usfca.edu/∼peter/cs220/

[5] J. Barnes and P. Hut, "A heirarchical O(NlogN) force-calculation algorithm," *Nature*, vol. 324, no. 6096, pp. 446–449, 1986.

[6] D. Grossman. (2012) Sophomoric Parallelism and Concurrency. [Online]. Available: http://www.cs.washington.edu/homes/djg/teachingMaterials/spac/

[7] (2012) Intel Threading Building Blocks (TBB). [Online]. Available: http://threadingbuildingblocks.org

[8] (2012) CUDA Zone. [Online]. Available: http://developer.NVIDIA.com/category/zone/cuda-zone

[9] S. Rivoire, "A breadth-first course in multicore and manycore programming," in *Proceedings of the 41st ACM technical symposium on Computer science education*, ser. SIGCSE '10. New York, NY, USA: ACM, 2010, pp. 214–218.

[10] C. T. Mitchell, J. Mache, and K. L. Karavanic, "Learning CUDA: lab exercises and experiences, part 2," in *Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion*, ser. SPLASH '11. New York, NY, USA: ACM, 2011, pp. 201–202.

[11] J. Mache. (2012) Teaching parallel computing with higher-level languages and activity-based laboratories. [Online]. Available: http://lclark.edu/ jmache/parallel

Analysis of online and blended course delivery and their future in academia

S. Srinivasan
Division of International Business and Technology Studies
Texas A and M International University
Laredo, TX 78041, USA

Abstract: *Academic institutions are facing a new kind of challenge in the course delivery mechanisms. Several students want flexibility in the way courses are delivered because of their lifestyle changes. Two such methods that are popular with students are online and blended courses. Online courses have become more popular because of advances in technology that make video communication using the internet highly reliable. In meeting the institutional expectations for quality, online courses offer some unique challenges. Blended courses satisfy some of the constraints experienced in online courses. In the future there is going to be greater need to try different methods of course delivery since the learners have become adept in using technology in ways that will help simulate a physical presence in a classroom setting. In this talk we address these two course delivery mechanisms and how institutions could cope up with the demand while maintaining quality.*

Keywords: Online course, blended course, technology, quality, course delivery

1 Introduction

Higher education is constantly facing new challenges in the form of student expectations. Today's students are technologically savvy. Many students have access to higher-end applications that facilitate video communication. Students try to balance their educational goals with that of being independent and self-sufficient. In order to meet the latter goals they need flexibility in meeting their educational obligations. Traditionally courses are offered at a physical location where the students congregate. With the advancements in technology many educational institutions are trying to diversify their offerings. The choices available for course offerings range from fully face-to-face in a classroom setting to fully online. The additional choices available are a mix of face-to-face and online, called by the name hybrid or blended offering [1]. Instructors feel comfortable in the traditional instructional method of face-to-face teaching because they feel a link to the students whom they can see in person and assess them on multiple factors. Moreover, this method gives them a sense of control and satisfaction on the way they conduct their teaching and assessment. They will be able to individually assess each student's level of interest in learning and participation in discussions.

Online education has received the attention of many educators over the years. In 1997, Peter Drucker, the well known Management professor, predicted that traditional universities would become "wastelands" in the early part of the 21st century [2]. Others perceived that online education would lead to virtualization in educational delivery. Consequently, the general feeling in early 1990s was the replacement of "physical processes with new processes that can be accomplished over networks" [3]. Over the past two decades technology has improved to a level that high quality video could be delivered to many devices, including mobile devices. With this advantage today many higher education institutions are focusing more on online delivery of courses. In May 2012, John Hennessy, President of Stanford University, predicted that online education is "going to change the world and it's going to change the way we think about education" [4]. His observation is that the traditional lecture halls "will play a much smaller role" in the future. His statements are borne out by the fact that more than 100,000 students from many parts of the world took three engineering courses in Fall 2011 that Stanford University offered online. No institution around the world has the physical infrastructure to handle that many students in a classroom setting.

Massachusetts Institute of Technology (MIT) announced in December 2011 the MITx project to offer Massively Open Online Courses (MOOCs). When they opened up the first MOOC in March 2012 on Circuits and Electronics, the course enrollment was 120000. Harvard University joined MIT in this venture to create the edX project in April 2012 to offer free online classes. This edX project is unique in that two of the highly respected institutions in the world have joined forces to offer free online courses but also offer certificates to the participants who successfully complete the courses. Other highly reputed U.S. universities such as the University of Pennsylvania, University of Michigan, Stanford University and Princeton University formed a consortium called Coursera to offer online courses. This opens up another topic to consider, namely credentialing of online programs. Already University of Phoenix has a dominant presence online with over 400000 students worldwide. However, the reputation of University of Phoenix online degrees is not high. The pooling of efforts by well known academic institutions in offering online courses is not new. It is important to note that some such ventures were tried in the past and they failed. For example, University of Michigan, University of Chicago and Columbia University launched a commercial venture called Fathom in 2001 and folded the operation in 2003. Likewise, Stanford University, Yale University and Princeton University collaborated in the venture called AllLearn that collapsed in 2006. The approach by some of these universities this time is in attracting new learners is by offering free online courses. They eventually hope to have some of these learners as students in their institutional online programs even though it is not explicitly stated at this time. Without questioning their motives it is worth noting that this experiment clearly shows the tremendous need for online courses around the world, not just in US. Next we will look at two of the main types of offerings that are becoming popular.

2 Online Course Offerings

The author strongly believes that online education is essential for an academic institution to consider for two reasons: flexibility for students, creativity for faculty. A random web search of 50 institutions in five populous states showed that all institutions offer online courses in varying numbers. People associate online course offerings to asynchronous learning. That is where flexibility comes for the student. From the instructor perspective, the instructor is not tied by a fixed time of a class nor the resources that could be brought in one setting. This is where the creativity of the instructor comes into play. It is essential to note that online offerings need not be totally asynchronous. Given the reach of technology today it would be reasonable to expect that some part of the course time could be devoted to synchronous discussion sessions. This would give an opportunity for all students to interact with others in the class via online chat. Technology allows discussion forums to be moderated by the instructor where the postings could be asynchronous. The added advantage is that everyone could peruse the thread of discussions at any time.

Online course offerings come with several challenges. The first and foremost challenge for the instructor is to map out the entire course with all the details for content and assessment [5]. This is a daunting task for anyone who has not undertaken the responsibility for an online course. Programs such as the University of Maryland Quality Matters offer the necessary planning tool for the instructor [6]. From a student perspective, there should be an allotted time for the course in order to meet the course obligations. Many students lack this discipline and consequently feel quite stressed out during periods of assessment. Traditional in-class assessment models do not work with online courses because the instructor is not there physically to monitor. Further, the students are likely spread over multiple time zones. For this reason the reliance on technology for assessment becomes more critical. There is room for abuse in assessment and so adequate safeguards must be built-in in order to ensure fairness for all students. Use of video technology should be highly encouraged.

The author has firsthand experience offering an online course and facing several challenges to overcome from the technology perspective. Student expectation for an online course is asynchronous learning with no face-to-face meeting requirement. The students should be fully aware of the course requirements such as face-to-face meeting times, synchronous video chat session requirements and assessment expectations prior to choosing an online course. Another important aspect is the necessary technology capabilities required at the student end for a successful completion of the online course. An essential component of any learning experience is the ability to exchange ideas with peers. Discussion Forums provide such a possibility in online learning and the students should be incentivized to participate in such discussion forums. Online courses meet the student expectations for flexibility of time and place. It is up to the instructor to enforce quality. Next we will look at blended course offerings.

3 Blended Course Offerings

Many educators use additional learning tools besides the standard lecture in a classroom supported by a PowerPoint presentation. Even in this approach it has become important to use materials from YouTube, Facebook and Twitter. Because of the need to integrate multiple methods of disseminating knowledge and knowing the various viewpoints on a topic, blended course offerings are becoming important. As the term ‘blended’ clearly implies, we are referring to incorporating multiple methods of learning. Thus, blended learning enables the student to have the best of both – face-to-face setting and external learning [7]. This approach enables the students to have less “seat time” on campus, which is well received by today’s technologically savvy students. As observed above, students should have the opportunity to explore some material related to the course on their own through internet search. Moreover, they should be able to view any related videos on the topic and comment on them. Such comments could be threaded so that the instructor will be able to monitor the contributions of students. Facilitating this type of learning requires the lecture aspect to be reduced from 100 percent to somewhere around 60 percent. Many instructors feel threatened if they do not interact with the students all the time. This attitude should change given the technological capabilities that are available today. The author's experience is that some instructors are willing to try other technologies but only within the classroom context. Blended course offerings clearly require a fundamental shift in attitude from the classroom being the center of all learning related activities [8].

In the educational research it is difficult to find any definitive recommendation on how a blended course should be taught. Based on personal experience and talking with several other colleagues, the author recommends a 60 – 40 split in the classroom activities. The 60 percent refers to in-class activities such as lectures and discussions. The 40 percent refers to course time devoted for students to explore things on their own and also use other technological tools. The instructor should certainly make available some of these additional resources that the students should explore.

The concept of blended education is not new. The Sloan Consortium has done extensive work in this area and published some books as well. The book by Picciano and Dziuban highlight the challenges and best practices for blended learning. The Alliance for Higher Education Survey [9] identifies several success factors for online education. The top 5 in this category in descending order of importance are:

- Executive leadership and support
- Faculty and academic leadership commitment
- Student Services
- Technology Infrastructure
- Course/Instructional quality

This list clearly shows that technology is important but student services to support online or blended education is even more critical. In the next section we will highlight some of the technology support needed to support online and blended education.

4 Technology Support

Virtually all academic institutions in U.S. use one of a handful of e-learning tools available today. These are: Blackboard, Moodle, WebCT, eCollege, Desire2Learn, Angel, Wimba, etc. All these tools provide a variety of ways to communicate between instructor to student and student to student. Participants in online or blended course offerings should take advantage of these tools. Chat sessions can be easily captured and made available for others to review. Discussion forums provide a threaded discussion and so one can review the earlier posts on the topic. In addition to the above e-learning tools, one could also use Adobe Presenter and Adobe Connect to share instructor created video material with the students. These two tools are not free tools and so the institution must make an investment in software and training for the users. From a student perspective the created videos are easy to view as they come as web links from a server. Use of these Adobe tools requires certain level of training and practice as there are minor glitches that need to be taken care of before the student is able to fully benefit from these video postings. Video postings are an excellent alternative for the classroom experience for the student as the instructor is able to share the details through examples and writings [10]. The author's personal experience is that these video creation tools are easy to use once the user experiments with a few simple videos. They enable integration of material from multiple sources in the creation of the video. What was lacking was the ability to edit the videos in order to make them better. This aspect requires significant investment by the institution in both the tools and training of people who could assist the instructors. We have only listed a very few number of simple tools that are available online for use for a variety of specialized uses. These are easy to locate on the web as the need arises.

Future of online course offerings is bright. Many educational institutions, excluding the elite institutions such as MIT, Stanford University, etc, at this time have not invested significantly in online education. For an overwhelming majority of the institutions offering online courses has become a necessity just like banks are expected to offer online banking for their customers. Moreover, the general feeling in academic institutions is that online offerings provide cost savings to the institutions. The demand for online courses is emanating from the students and as customers their voices are heard. In order to meet this need educational institutions have offered minor incentives to faculty to develop and deliver these courses. Often the investment is in the form of a few tools and limited training to some staff members to support the faculty. Another important thing to note is that at this time an extremely small number of institutions alone offer an entire program online. By this we mean the program could be a complete MBA program or a BA program in Art History. It takes lot more effort and resources to offer full online programs than certain courses in a program. That is why many institutions have decided to offer only select courses online. In most cases these courses are the ones that are core courses in the early stages of the curriculum with large enrollments, such as Introduction to English Literature.

The experiments by elite institutions show that when courses are offered online the enrollment in the courses is in the thousands, often exceeding 100,000. Such numbers would overwhelm the IT infrastructure of most educational institutions. Furthermore, online courses require timely feedback to the students just as the face-to-face courses require. In a traditional course on campus when enrollment is large such as 500 students, it is taught by a single instructor in a large classroom with video monitors and then the students are split into groups of 30 to meet with instructor assistants for more personal attention. This is essential for learning. When the enrollment exceeds 100,000 in an online course there is no practical way to provide small group interaction with instructor assistants. For this reason the elite universities have automated the entire process of feedback. As mentioned earlier some of these experiments failed in the past because the technological advancements are not what they are today. We see a bright future for online courses from many smaller institutions, many of them public universities.

5 Summary

We have highlighted the benefits and challenges of online and blended course offerings from the perspectives of the student, the instructor and the institution. We have also pointed out the very high attraction of some online courses from elite institutions where the enrollments are in the hundreds of thousands. This experiment in offering online and blended courses is essential for the growth of academic institutions. However, most institutions have not made the necessary investment in this new approach to curriculum delivery. With the advancements in technology the institutions are now able to support the demand for both online and blended courses. As some of the well known educators have predicted there is going to be a tremendous need for more online courses than are what are offered now in traditional settings.

6 References

[1]. Hamilton, E. & Feenberg, A. (2005). The Technical Codes of Online Education. DLA E-journal, 9 (#1).

[2]. Drucker, P. (1997). Seeing things as they really are. Forbes 159 (#5), 122-129.

[3]. Katz, R. & Oblinger, D. (2000). The 'e' is for everything: E-commerce, e-business and e-learning in the future of higher education. Educause Leadership Strategies, # 2. San Francisco, CA: Jossey-Bass, Educause & Pricewaterhouse Coopers.

[4]. Hennessy, J. (2012). IEEE Medal of Honoree. IEEE Spectrum, 49 (#5), 28-32.

[5]. Abel, R. (2005). Implementing Best Practices in Online Learning. Educause Quarterly, 28 (#3), 75-77.

[6]. QM. (2012). Quality Matters Program. http://www.qmprogram.org

[7]. Picciano, A. and Dziuban, C. (2007). Blended Learning: Research Perspectives. Newburyport, MA: Sloan-C Series.

[8]. Boettcher, J.V. (2011). Ten Best Practices for Teaching Online. http://designingforlearning.info/services/writing/ecoach/tenbest.html

[9]. A-HEC. (2005). Achieving Success in Internet-Supported Learning in Higher Education survey.

[10]. Hartman, J.L., Dziuban, C., & Moskal, P. (2007). Strategic initiatives in the online environment: Opportunities and challenges. On The Horizon, 15 (#3), 157-168.

DEVELOPED AND SECURED WEBSITE FOR NIGERIAN UNIVERSITY.

Dr Okonigene R.E[1]., Dr. John S. N.[2], Ojieabu C[3]., Ihenyen Aigbe[4].

[1,3,4]Department of Electrical & Electronics Engineering, Ambrose Alli University, Ekpoma, Edo State, Nigeria

[2]Department of Electrical and Information Engineering, Covenant University, Ota, Ogun State, Nigeria

Abstract: *Due to lack of available relevant information required to determine a student's performance and progress, in most Nigerian Universities, by parents, guardian or sponsors, and the general public we therefore decided to develop a secured website on a local area network (LAN). Thus, we developed a website for the Department of Electrical and Electronics Engineering, Ambrose Alli University, Ekpoma. The design was implemented with the help of notepad, Dreamweaver, macromedia fireworks and macromedia flash. The combination of this software made it easier to organize templates with animated effects and graphics. Codes were generated as the pages were accessed by using object oriented software and these codes were used to reproduce the site anywhere. The security aspect of the work is not discussed in this paper. The website was made operational over a local area network, tested and confirmed to be functioning properly with IP (Internet Protocol) addresses assigned manually to each computer, fast Ethernet switch (16 port Cisco switch 10/100Mbps) was used for the network to reduce the dependency of the hosts on a single cable. For security purposes the administration login page was not hyperlinked with the main website so as to avoid unauthorized alteration of files.*

Keywords: Website design, Intranet, Internet connection, Web security, star topology

1. Introduction:

The World Wide Web popularly referred to as the internet is the information super highway. It comprises of digital systems connected together via terrestrial satellite, fiber optic and is mainly accessed by people looking for information about a product, company or organization [1,2,3]. Today's world is a global village and require huge amount of information to be communicated. Hence the increased cost and time in effecting this arises the need to sort out ways of reducing the high cost of such communication and increasing the rate at which it reaches its destination with much accuracy [4]. Many parents, guardians, or sponsors of students in most Nigerian Universities would have loved a situation that enable them monitor the activities and performance of their wards. After a careful study of the enabling laws that establishes most Nigerian Universities we decided to solve these problems by developing a secured website. A local area network, LAN is a group of computers located in a relatively small area and connected by a common medium. A LAN interconnects over small distances up to about 10km. each of the computers and other communicating devices on the LAN are called a node. A LAN is characterized by three primary attributes: *topology, medium* and *protocol* [5,6,7,8]. This developed website was implemented on the intranet-a private corporate network that uses Internet software and TCP/IP networking Protocol standards. We used the intranets to make available students data such as registration number, matriculation number, date of registration, sponsors name, student's contact information, etc. An intranet does not always include a permanent connection to the Internet [9,10]. The site is purely an object oriented designed work making use of software that bring out the beauty and easy accessibility of Web Pages. It comprises of Pictures, Multimedia, Animation, information, links to technical sites, and other information obtained from the Faculty handbook, directly from lecturers and Department. The level of security implemented is a server based access one to avoid unauthorized alteration of data stored in site.

2. Methodology

The web pages were developed using software which generates codes automatically and it became easy to manipulate the buttons, images, multimedia, and marquee effect amongst others that are typical of web pages. The star topology, shown in Figure 1, was chosen to implement the intranet network where all the computers are connected to a switch. We employed a 16 port Cisco hub 10/100Mbps fast Ethernet switch for the network. A CAT6 UTP cable with bandwidth up to 100Mbps was used for data transfer (link) in the intranet. The major text editing software used for the design of this project is the macromedia Dreamweaver software. Thus for this work we used the Dreamweaver and the Notepad to generate the codes.

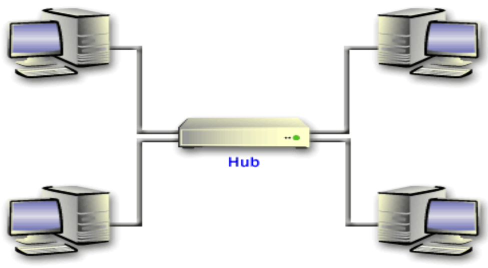


FIGURE 1 Star Topology

3. Results and Analysis

In this present homepage, shown in Figure 2, we splitted the table into several rows and columns, the page properties for the home page was as follows:

For the banner we used a width of 1329 pixel, a height of 195 pixel, which was splitted into two the left hand side being a width of 194 pixel and a height of 195 pixel, the main body was divided into three columns the first from the left hand side having a width of 308 pixel and a height of 313 pixel, the middle having a width of 606 pixel and in the last column we used a width of 407 pixel all having a height of 313 pixel.

The following html codes were used for the home page:

```
<html>
-
-

<! DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html
xmlns="http://www.w3.org/1999/xhtml">
<head>
-
-
  <tr>
   <td height="261"
bgcolor="#FFFFFF"><img
```

```
src="images/school_gate.jpg"
width="309" height="243" /></td>
  </tr>
</table>
</body>
</html>
```

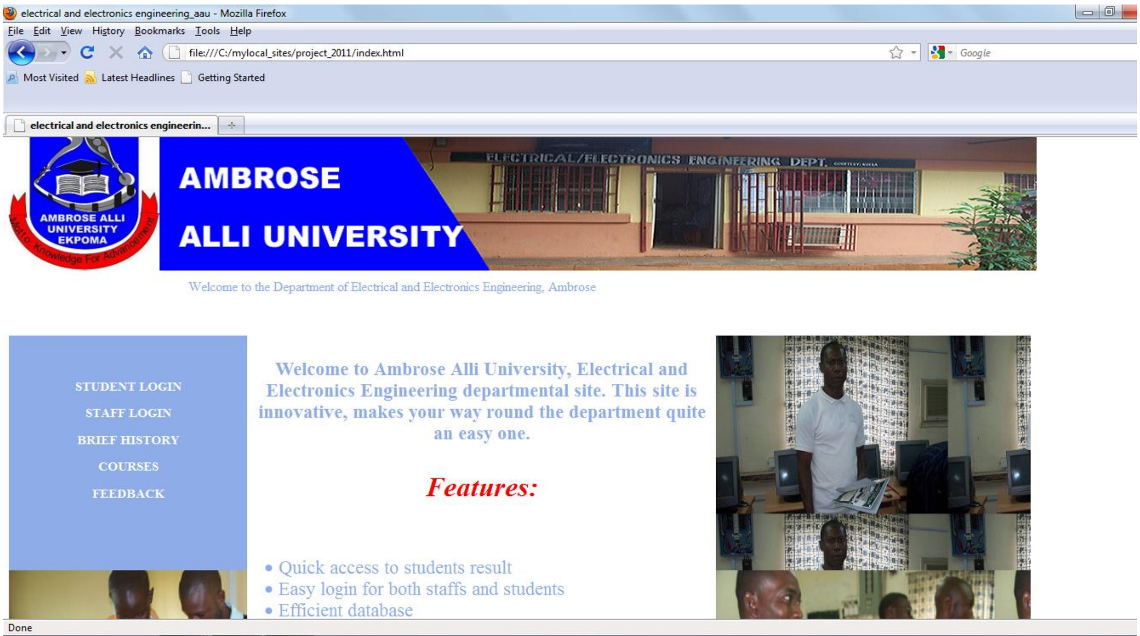


FIGURE 2 Homepage of the website

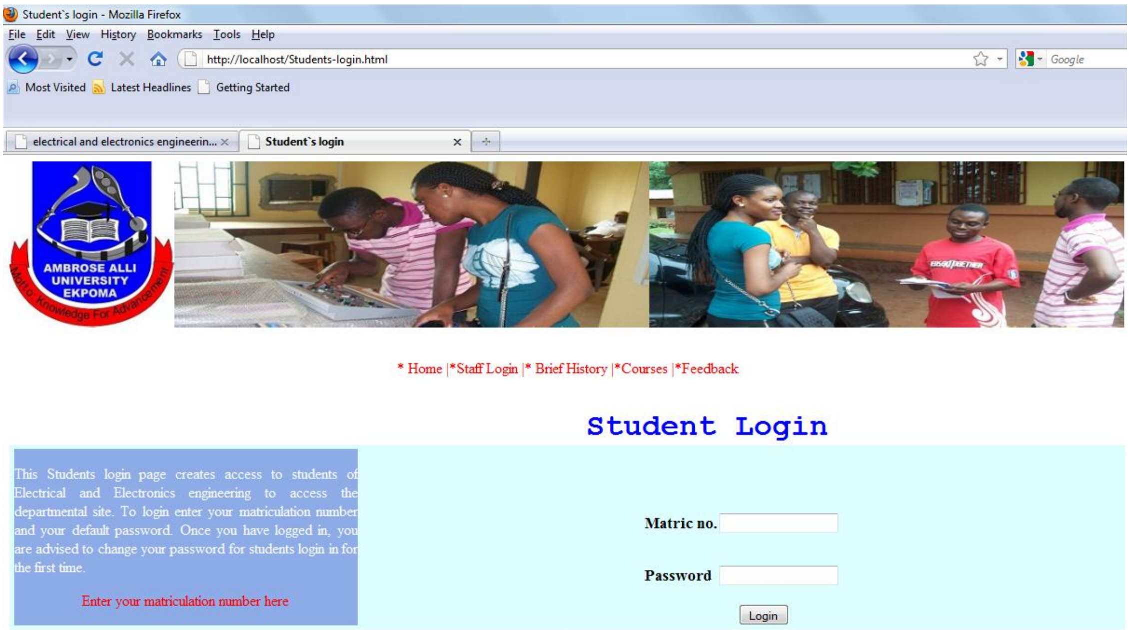


FIGURE 3 Student login page of the website

Figure 3 shows the student login Page. In creating the Student Login Page, we relied on information from the Faculty of Engineering student information book, the students personal data, students score records, and contact address. Here information about the students examination records, diciplinary action, conducts, commendations and other relevant information can be assessed by the corresponding parents, guardian or sponsor.

4. Conclusions

In this project, we have looked at the area of information and communication technology, its impact on education in the 21st century. The magic of packet switching using a sophisticated Cisco switch in addition with CAT5 UTP cables provided useful tool in setting up our intranet (100Mbps fast Ethernet).
Also, in this project, we developed a website for the Department of Electrical and Electronics Engineering Ambrose Alli University, where information about students and staff of the Department can be accessed by parents, guardians, and sponsors.

References

[1]. Kai Chang "RF and Microwave Wireless System" second edition.

[2]. "Encyclopedia Britannica" Student and Home Edition.

[3]. Berners-Lee, Tim; Fischetti, Mark (2000). "Weaving the Web: The Original Design and Ultimate Destiny of the World Wide W eb by Its Inventor". San Francisco: Harper.

[4]. Craig Hunt (1997) "TCP/IP Network Administration", second edition, December 1997.

[5]. M.S OKUNDAMIYA "Principles of Computer Communication Networks" first edition, Stemic Publications, 2009, chapter 5.

[6]. Todd Lammel "Cisco Certified Network Associate" study guide, 6th edition.

[7]. Luke Welling and Laura Thompson [2003] "PHP and MySQL Web Development".

[8]. Public domain. World Wide Web Consortium. 2009-06-09. Retrieved 2009-06-10

[9]. James Robertson "The Difference between Intranet and Website", jamesr@steptv. com.all

[10]. www.w3schools.com