University Ontology: case study Ahlia University

K. Hadjar[1] **and N. Chanane**[1]
[1]Department of Multimedia, Ahlia University, Al Manama, Bahrain

Abstract - *The huge amount of information available on the internet (and intranets) and its unstructured nature is reaching a point that some actions has to be taken in order to ease the use of queries within a web search engine. The introduction of order/organization and structure is necessary for the process of this information. One-step toward this goal is the use of ontologies for specific areas/domains. The word ontology is becoming widespread and its use in organizing the web is gaining momentum. Many scientists are working on semantic webs, which are considered as intelligent and meaningful webs. The lack of university ontology made me develop such one. A case study was developed to validate my ontology: Ahlia University, Bahrain.*

Keywords: Ontology, Abstract Model, University.

1. Introduction

Nowadays the web has shifted into another dimension: the semantics. The language of the web: HTML has embraced semantics into its last version: version 5. Almost all the fields have their own ontology. Since I am in the field of academics since a while, and I have noticed that there is few serious development of university ontology, I have decided to build one.

Several tools and methods have been developed to build ontologies. Rather than focusing all the attention to information, it was all in the core concepts in using ontology and its relationships. The most well-known and widespread tool for editing and developing ontologies is Protégé [10]. Its Graphical User Interface (GUI) lets the developers concentrate on the concept rather than thinking about the syntax of the output language. Protégé has pliable data design and extendible plug-ins.

In this paper, definition of the University concept is clarified through the university ontology. Creating a university ontology using Protégé is the object of this paper. Ahlia University is taken as a case study for the development of the ontology and several phases are outlined e.g., superclass, hierarchy of subclasses, creating subclasses instances, retrieving queries, graphs and visualization views. The case study is limited to few departments and courses, as an example. This implies that since the model works for one university, it will work for other universities, with minor changes.

This paper is organized as follows: in section 2 I discuss the steps required for building the university ontology. Section 3 presents my case study: “Ahlia University”. In section 4, some queries applied on the Ahlia University ontology are exposed. Finally section 5 concludes this paper.

2. University Ontology

There is many data shared between organizations and organization divisions, which can be used in building up the university structure. Yet there did not seem to be a suitable shared terminology for presenting such information in linked data. Based on the need of a clear structure for any organization, I have developed university ontology as the basic structure model to share between all organizations who do not like to start from scratch.

In fact a lightweight, highly reusable ontology, which did not try to model particular organizational structures, is required.

2.1 Building the ontology

To increase the ontology efficiency, I need to ensure that Ontology is defined as a formal specification, explicit and consensual conceptualization of a domain [5]. Definitely, the development and design of Ontology supports people to recognize and answer the doubts of domains [1]. It comprises of a group of concepts related together in an organizational method. In this paper, I focus on specialized Ontology that is domain ontology and task ontology. These are reusable Ontologies within a given domain, but not from one domain to another. While all tasks performed in a given domain are within the ontology [5]. According to Mizoguchi, Vanwelkenhuysen and Ikeda [6], the ontology task is to describe a curtain vocabulary related to a task. The reuse of ontology is critical. I have to build manually the ontology from scratch by following a known methodology.

2.2 Ontology Development Methodology

To ensure the consistency of ontology structure and increasing its efficiency, during development I have followed the guidelines from many sources. First, I have studied how to build ontology by using a guideline from [7]. The guide built using Protégé ontology editor [6], which is the same tool that we have used for the University Ontology development. I have studied a few ontology development methodologies and finally I have decided to follow a recently defined methodology from [4], incorporating with the guide from [5]. This ontology development covers the steps from the initiation phase to the data retrieval phase of ontology. Specification and Conceptualization are the two main steps available in this methodology. Obtaining knowledge about the domain is the objective of the first step. Moreover, organizing, structuring the information using exterior demonstrations independently from the environment and implementation language is the second objective.

2.2.1 Specification

The scope puts boundaries for the ontology; requiring what has to be involved and what has not. This step was suggested for an advance stage in the Ontology Development: A guide to creating your first ontology [5], is included at this stage to minimize the process of analyzing concepts and data, particularly for the range and difficulty of the University Model Ontology. The iterations for following verification, process will be adjusted when needed. I have considered the needs for elaborating the University structure project with theories related to higher education organizations. It is a first prototype and the considered concepts are not related to all divisions in an organization. Therefore, it includes general concepts for the university abstract model.

Previous domain analysis was necessary to be done as the first step. In this work, the presentation for framing the university structure and the relevant documents were collected from a number of organization charts of different universities. Furthermore, advices from management leaders of universities and faculty were put into consideration. Gruninger and Fox Methodology point of view [11] was taken into account. Problems arise when people need information but the systems don’t provide it. The motivation scenarios are followed. In addition, templates have been used in order to define motivation scenarios and link them to the people involved. A set of solutions to all problems is made available whenever the semantic features can be resolved.

2.2.2 Conceptualization

In this step, the terms used in representing the most important entities in the university structure are enumerated as classes shown in Table 1. Definitions of the main classes are listed after the table. All the concepts appearing in the figure mostly focus on the main departments in any University e.g., AcademicAffairs, AdministrativeAffairs, President, Deans, Chairs, Faculty, Student, Courses, Library, Gym, WebSite, BookStore, etc…

Table 1. Key Item List as Class and Subclass

Class	Subclass
Courses	GraduateCourses
	UnderGraduateCourses
Programs	BachelorProgram
	MasterProgram
	PhDProgram
Person	Employee
	Students

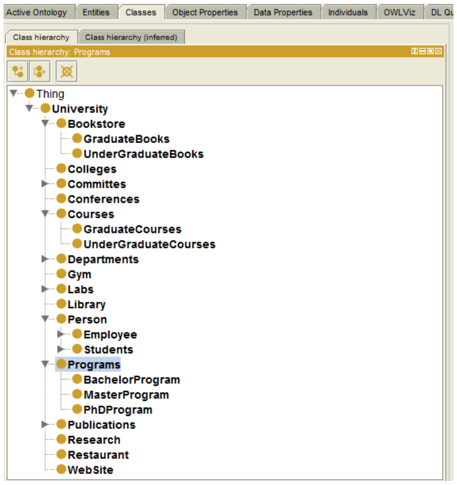
- The class “Courses¨ is defined as the basic two categories of the courses available at most universities, “GraduateCourses” and “UnderGraduateCourses”.
- The class “Programs¨, defined as the possible offered programs by universities at their start or at later stages of their program. And the three subclasses are, “BachelorProgram”, “MasterProgram”, and “PhDProgram”.
- The class “Person¨ divides the people at the university to two types, Employees; staff and faculty, and Students.

The main relations, attributes and properties have been created as shown in Table 2.

Table 2. Relation between the University Model Classes

Class Name	Relation	Class Name	Inverse Relation
Department	hasHead	Chair	isHeadOf
Students	take	Courses	areTakenBy
Faculty	publishes	Articles	isPublishedBy
Classes	attendedBy	Students	attends

Figure 1, contains the object properties according to the relationship, which we want to add between the classes “Professor” “advises¨, and the courses “areOfferedBy¨.



- Thing
  - University
    - Bookstore
      - GraduateBooks
      - UnderGraduateBooks
    - Colleges
    - Committes
    - Conferences
    - Courses
      - GraduateCourses
      - UnderGraduateCourses
    - Departments
    - Gym
    - Labs
    - Library
    - Person
      - Employee
      - Students
    - Programs
      - BachelorProgram
      - MasterProgram
      - PhDProgram
    - Publications
    - Research
    - Restaurant
    - WebSite

Figure 1. University Ontology Classes Hierarchy.

Table 3, illustrates the relationship between individual to data literal, for example the Course has “CRN ¨, “courseName ¨, “creditHours ¨.

Table 3. University Ontology ObjectProperty

Class	Property
Faculty	emailAddress
	mobile
Course	CRN
	courseName
	creditHours

In a general usage, a restriction can be a general form of instructions that sets a limited border defined for a function or a type of process. These relations were captured in a semantic diagram to represent the relations between components.

- Property and Relationship

Since having only classes cannot answer all the enquiries; defining links inside or between the classes is needed (such as properties). I have used property, which shows relationship between individual and individual, Relationships between Individuals at University Ontology Such as property, faculty as advisor of student. I have also defined Object Properties Domain & Ranges, for example:

```
<owl: Object Property rdf: about = advisor>
<rdfs: domain rdf: resource=student/>
<rdfs: range rdf: resource=faculty/>
</owl: object Property>
```

The Top layer of the university ontology there is: “Person”, “course”, “committee”, “AcademicAfairs”, “Admission”, “University” etc… In the Middle layer of the university ontology there is: “AdminStaff”, “Student”, “articles”, “books” and “subject”, “library”, “colleges” and “departments” etc. And the Bottom Layer includes: “Chair (Professor)”, “Teaching Assistant”, “Dean”, “Director”, “Visiting Professor” and “Professor Types”, etc…, for example

```
<owl: object property rdf: about =
TeacherOf/>
<rdfs: domain rdf: resource = Faculty/>
<rdfs: range rdf: resource = Course />
</owl: object Property>.
```

The object Property –TeacherOf; its domain is in Faculty and range in Course. It means that TeacherOf Property value will be only just opposite to the isFaculty property because has Property is always inverse to is Property. The relation of Inclusion (rdfs: subPropetryof), equivalent (owl: equivalentPropetry) and Inverse (owl: inverseOf), and the limitation of function (owl:FunctionalPropetry) and inverse function (owl: InverseFunctionalProperty).

Since the conceptual model of the ontology has been created, the next step is to define related instances. For each instance, I have described: a name, the name of concept it belongs to, and its attribute values.

The instance (individual) is described first; then, the right class was selected, and finally its instances for the class are created.

Use rdf: type to state its class, and one instance can belong to many classes or many class belongs to same instances, for example;

```
</owl: thing rdf: about=
AdvancedDatabaseSystems >
<rdf: type rdf: resource= #course/>
<rdf: type rdf: resource= #student/>
</owl: thing>
```

Here it defines an individual or instance AdvancedDatabaseSystems, which belongs to the class “course and student. In which rdf: type has appeared twice, it shows that this instance belongs to two classes at the same time.

2.2.3 Implementation

I have chosen Protégé 4.1 in order to implement the ontology, due to its extensibility, quick prototyping and application development. Protégé ontologies are easily exported into different formats including RDF Schema, Web Ontology Language (OWL) and Top Braid Composer [9], which we have used at later stages in querying. Particularly, we have implemented the University Ontology in OWL. Structured relations are transformed into bidirectional relations while modeling in OWL. Moreover, only relations that are necessary in answering competence questions were modeled in ontology.

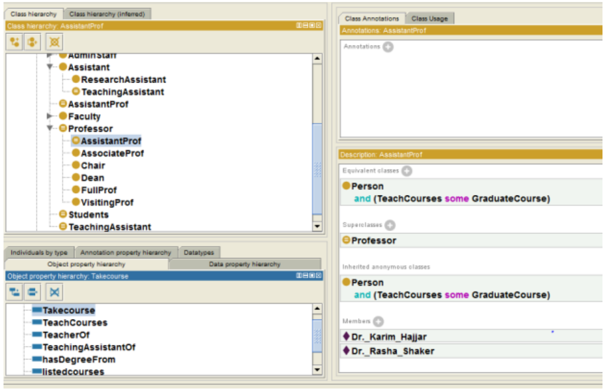
2.2.4 Verification

Consistency validation and classification are verified by using the Reasoner. During the process of charging classes and attributes, we used incremental and continuous verification to avoid future propagation errors. In the Reasoner, any class which is unsatisfiable is shown in red color indicating that error exists. At this point, it is very important to see how classes are defined (disjoint, isSubclassOf, Partial Class, Defined Class, etc…) and how are their restrictions (unionOf, allValuesFrom, etc…). Classification process is either for the whole ontology or for selected subtrees only. When the test is completed, the whole ontology, errors were listed, moving from bottom to upper level class. To compare the ontology execution with its conceptualization, graphs were generated using OWLViz [2] and OntoViz plug-ins [8].

3. Ahlia University Ontology

Ahlia University ontology defines elements to describe Ahlia University and its activities, which can occur between Departments, Faculty and Students. I have built the ontology based on the organization chart available on the University website and all data used for testing my work was also taken from the catalogue available. Since my base of the ontology was the university ontology, I only had to make some changes on the classes following the organization chart. Concepts (classes) such as, Departments, Degrees, Deans, Chairs, Faculty, Student, Courses, Library, CareerCenter, WebSite, ICTCenter, Labs ...etc, More relations (rules) where added between the Classes to show how they are related and linked to each other. The Ahlia University Ontology also includes relationships between classes. For instance, the relationship "teaches/isTaughtBy"is between Faculty class and Courses class. Other relationships are added, such as: hasHead/ isHeadOf, hadMember/isMemberOf, etc...
As shown in figure. 2, some of Ahlia University related classes and subclasses are listed. All the concepts appearing in the figure are mostly focused on the students, faculty and course based.

- The class “AhliaUniversity” is the highest-level class in this domain.
- The class “Assistant” defined as the basic two categories of the position of an Assistant available at most universities, ResearchAssistant and TeachingAssistant.
- The class “Professor”, defined as the rank type of the faculty at the universities. I have listed its subclasses (AssistantProfessor, AssociateProfessor, FullProf and VisitingProf).


Class hierarchy
Class hierarchy (inferred)
Class hierarchy: AssistantProf
Class Annotations
Class Usage
Annotations: AssistantProf
Annotations
Individuals by type
Annotation property hierarchy
Datatypes
Object property hierarchy
Data property hierarchy
Object property hierarchy: Takecourse
Description: AssistantProf
Equivalent classes
Superclasses
Inherited anonymous classes
Members

- AdminStaff
- Assistant
  - ResearchAssistant
  - TeachingAssistant
- AssistantProf
- Faculty
- Professor
  - AssistantProf
  - AssociateProf
  - Chair
  - Dean
  - FullProf
  - VisitingProf
- Students
- TeachingAssistant

- Takecourse
- TeachCourses
- TeacherOf
- TeachingAssistantOf
- hasDegreeFrom
- listedcourses

Equivalent classes: Person and (TeachCourses some GraduateCourse)

Superclasses: Professor

Inherited anonymous classes: Person and (TeachCourses some GraduateCourse)

Members: Dr._Karim_Hajjar, Dr._Rasha_Shaker

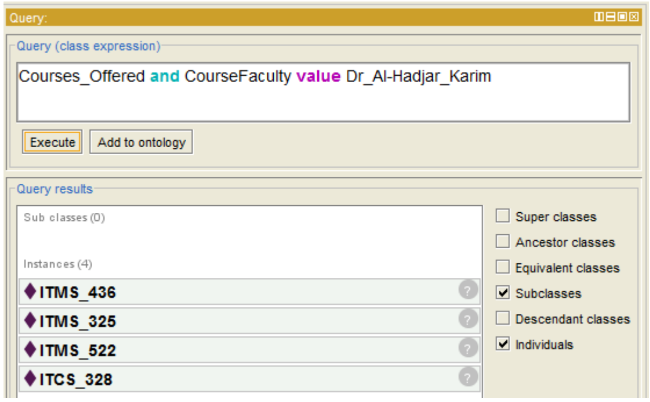
Figure 2. Ahlia University Ontology Class Hierarchy.

4. Query retrieval

A very powerful feature tab is available in Protégé, which is the DL Query. Considered as one of the basic plugins in Protégé 4 and is either available as a tab or a widget. It is based on the Manchester OWL syntax, which is a query language supported by the plugin, and a user-friendly syntax for OWL DL. A frame which fundamentally based on the information is collected about a specific class, individual or a property, into a single construct [7]. Here again, the query retrieval process has gone along the steps depicted in previous sections, and illustrated next figures (figure 3 and 4).

DL Query 1:

- Which courses does Dr. Karim teach?
- Courses_Offered and CourseFaculty value Dr_Al-Hadjar_Karim



Courses_Offered and CourseFaculty value Dr_Al-Hadjar_Karim

Sub classes (0)

Instances (4)

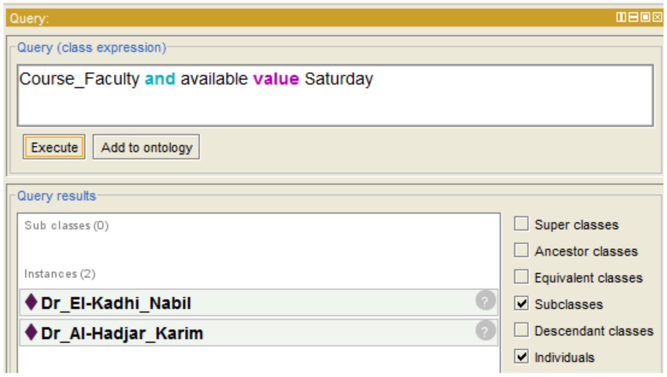
- ITMS_436
- ITMS_325
- ITMS_522
- ITCS_328

☐ Super classes
☐ Ancestor classes
☐ Equivalent classes
☑ Subclasses
☐ Descendant classes
☑ Individuals

Figure. 3. Snapshot of the DL Query 1

DL Query 2:

- The list of available faculty on Saturday
- Course_Faculty and available value Saturday



Course_Faculty and available value Saturday

Sub classes (0)

Instances (2)

- Dr_El-Kadhi_Nabil
- Dr_Al-Hadjar_Karim

☐ Super classes
☐ Ancestor classes
☐ Equivalent classes
☑ Subclasses
☐ Descendant classes
☑ Individuals

Figure. 4. Snapshot of the DL Query 2

Below, I have listed example of the data retrieved from the ontology using SPARQL query editor, available in TopBraid Composer.

Query :.

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX uo:
<http://www.semanticweb.org/ontologies/2011/9/Ontology1319571856122.owl#>
SELECT? X, ?Y, ?Z
WHERE
{?X rdf: type uo:Student
?Y rdf: type uo:Faculty
?Z rdf: type uo:Course
?X uo:advisor ?Y
?Y uo:teacherOf ?Z
?X uo:takesCourse ?Z }
```

Aside the features of Students class and hierarchy of Faculty class, most classes and properties used in this query can characterize it.

5. Conclusion

This paper presents my contribution to create University ontology. All the work done on Ahlia University Ontology was a reuse of the University Ontology that I have developed. Ahlia University Ontology describes all the departments under the University structure and the relationships that exist between them. I have modified the OWL version of the University Ontology and added more classes and restrictions based on the University organization chart of Ahlia University to get the final OWL of the Ontology. The ontology was expressed in OWL starting from creating classes and subclasses to properties, restrictions and instances. Then the owl file of the ontology is imported into TopBraid Composer for more powerful data retrieval software, to get the data needed from the ontology easily with short SPARQL queries. DL query in Protégé is also used for querying.

In ontologies, the focus is on relationships between concepts and not information itself. This work demonstrates the relations of university modules in the form of university ontology.

All the attention was given to the core concepts of using ontology and its relationships rather than information. As a future work in the domain of ontology in higher education, one can consider the following topics. The list may include but is not limited to,

- E-learning Applications Ontology
- Ontology sharing and reuse
- Graphs Ontology
- Enterprise Ontology
- Ontology matching and alignment

6. REFERENCES

[1] Ghorbel H., Bahri A., and Bouaziz R., *Les langages de description des ontologies: RDF & OWL*. Acte des huitièmes journées scientifique des jeunes chercheurs en Génie Electrique et Informatique (GEI), Sousse-Tunisia, (2008).

[2] Graphviz - Graph Visualization Software, Retrieved from http://www.graphviz.org

[3] Gruber T.R., A translation approach to portable ontology specifications, Knowledge Acquisition, 5 (2), pp 199-220, (1993).

[4] Gruninger M. and Fox M. S., *Methodology for the design and Evaluation of Ontologies*, IJCAI Workshop on Basic Ontological in Knowledge sharing, Montreal, Canada, (1995).

[5] Guarino N., *Formal Ontologies and information systems*. In proceedings of FOIS'98, IOS Press, Amsterdam, 1998.

[6] Mizoguchi R.,Vanwelkenhuysen J. and Ikeda M., *Task Ontology for Reuse of Problem Solving Knowledge*. In proceedings of the 2nd International Conference on Building and Sharing of Very Large-Scale Knowledge Bases. (KB & KS'95), (1995).

[7] Noy N. F. and McGuinness D. L. *Ontology development 101: A guide to creating your first ontology;* Stanford Knowledge System Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, (2001).

[8] OWL Web Ontology Language Overview, Retrieved from http://www.w3.org/TR/owl-features

[9] TopBraid Composer , retrieved from http://www.topbraidcomposer.com

[10] Protégé, Retrieved from http://protege.stanford.edu

[11] Zhao J., Wroe C., Goble., Stevens R., Quan D., and Greenwood M., *Using Semantic Web Technologies for Representing e-Science Provenance* in Proc 3rd International Semantic Web Conference ISWC2004, Hiroshima, Japan, Springer LNCS 3298, (2004).