What Is the Cost of One IFPUG Method Function Point? – Case Study

Beata Czarnacka-Chrobot
Department of Business Informatics, Warsaw School of Economics, Warsaw, Poland, bczarn@sgh.waw.pl

Abstract - *Software Functional Size Measurement (FSM) methods more and more often are used worldwide as a basis for estimating/measuring the Dedicated Software System (DSS) Development and Enhancement Projects (D&EP) costs. It involves adopting specified per-unit cost measured with regard to the product's functional size unit. In this paper we present a case study on tender competition concerning enhancement of DSS of specific public administration institution in Poland where one of the two potential developers offered possibility to modify such system at the cost of 1 cent per 1 Function Point (FP) of the International Function Point Users Group (IFPUG) method, whereas another one attempted to prove that enhancement at such unit cost was not possible to carry out. The goal of this paper is to analyse likely per-unit costs of the DSS enhancement with regard to 1 IFPUG FP. These issues classify into economics problems of Software Engineering Research and Practice.*

Keywords: dedicated software systems development and enhancement projects, per-unit cost, software size measurement, functional size measurement, IFPUG method, function point

1 Introduction

Like any other product, especially of engineering character, software systems too are characterised by some attributes that should be subject to measurement. The main attribute of every product is its size. However, software engineering cannot boast about such a degree of maturity with regard to the units intended for product size measurement (in this case product being software systems) as other engineering disciplines can (e.g., construction engineering where distinct, precise measure, that is square meter, is being used for the measurement of the apartment size). This constitutes the main cause of the problems with reliable and objective estimation and measurement of such basic attributes of projects aimed at developing, modifying/enhancing and maintaining software systems as work effort, total costs, per-unit costs, execution time or productivity. "Measurement of software size (...) is as important to a software professional as measurement of a building (…) is to a building contractor. All other derived data, including effort to deliver a software project, delivery schedule, and cost of the project, are based on one of its major input elements: software size." [1, p. 149].

However, it is not possible to give answer to the question about above mentioned project's attributes, in particular of per-unit cost, without prior adoption of adequate, i.e., sufficiently reliable and objective, software system size unit. Among the three measures of software system sizes being used in practice, that is: (1) programming units (e.g., source lines of code), (2) construction complexity units (e.g., object points), and (3) functionality units, this is just functionality units that now are the most widely recognised worldwide [2]. This has been confirmed by the fact they were accepted by the international standardization organizations: ISO (International Organization for Standardization) and IEC (International Electrotechnical Commission) as the only appropriate units of software system size – in the ISO/IEC 14143 norm, which standardizes the concept of the so-called software Functional Size Measurement (FSM) [3].

As a result of many years' verification of particular FSM methods reliability and objectivity, five of them (out of over 25) were recognised by the ISO and IEC as complying with the rules contained in the ISO/IEC 4143 norm and accepted as international standards as well. Those methods include the following:

1. International Function Point Users Group (IFPUG) Function Point (FP) method (ISO/IEC 20926 standard [4]).
2. Mark II (MkII) function point method, developed by the United Kingdom Software Metrics Association, i.e., UKSMA (ISO/IEC 20968 standard [5]).
3. Netherlands Software Metrics Association (NESMA) function point method (ISO/IEC 24570 standard [6]).
4. Common Software Measurement International Consortium (COSMIC) function point method (ISO/IEC 19761 standard [7]).
5. Finnish Software Measurement Association (FiSMA) FSM method (ISO/IEC 29881 standard [8]).

The most popular FSM method, at least in Poland, has been so far the IFPUG function point method ([9]) – and this is the method that in the discussed tender competition was chosen by the client as a point of reference for the offered per-unit costs, that is the costs measured with regard to 1 FP.

It should be mentioned that the IFPUG FP method offers calculation of function points at two levels [10]: (1) the so-called unadjusted FP; (2) the so-called adjusted FP. This is only the level of unadjusted FP that has been recognised as a standard of the software system FSM by the ISO/IEC [4]. Calculating the number of adjusted function points consists in correcting functional size (number of unadjusted FP) using the so-called VAF (Value Adjustment Factor), calculated with the use of 14 pre-defined so-called general system characteristics in order to evaluate the overall complexity of software system. Its purpose is to adjust the previously determined functional size to the environment of the specific project by taking into account the influence of technical and quality-related requirements on the project. The VAF's range is <0.65, 1.35>, which means that it can adjust functional size by maximum ±35% therefore it does have influence on the system's total cost. Since the publishing of the definition part of the ISO/IEC 14143 norm for the first time (in 1998), per-unit costs have been measured with regard to the functional size (as being recognized by those standardization organizations), i.e., with regard to unadjusted FP – hence further in this paper the IFPUG function points shall be understood as unadjusted FP.

What's more, it should be stressed that what is being considered here are per-unit costs of activities concerning software system *dedicated* to the needs of a specific client which is of significance since in case of commercial software packages designed for a mass consumer, where specified number of licences is being sold (e.g., MS Office), per-unit costs are calculated in a completely different way. Moreover, these activities make up a Dedicated Software System (DSS) Development and Enhancement Project (D&EP), in particular modification/enhancement of the existing system, and they do not contribute to maintenance project, in case of which per-unit costs require analysis of other benchmarking data resources.

Thus in this paper we present a case study concerning tender competition for enhancement of the software system dedicated to specific institution of public administration in Poland where one of the two potential developers offered possibility to modify such system at the cost of 1 cent per 1 IFPUG FP whereas another one attempted to prove that enhancement of the system at such unit cost was not possible to carry out. Hence the goal of this paper is to analyse likely per-unit costs of the dedicated software system enhancement with regard to 1 FP of the IFPUG method, and in particular to compare the offered per-unit cost against the selected resources of benchmarking data.

2 Analysis of data for per-unit costs of DSS enhancement with regard to 1 IFPUG FP

Per-unit costs of the D&EP (i.e., developing from scratch or enhancing the existing software systems) are difficult to estimate if a provider of the dedicated system does not have at their disposal their own resources of appropriate benchmarking data, on the basis of which they would be able to determine their own (organizational) per-unit costs with regard to 1 IFPUG FP. This results from the fact that such data depend on a number of specific factors – on a general level including first of all work costs that vary from country to country as well as type of project, type of software system, field of system application and technological environment of project execution (hardware platform, programming languages used, etc.) as well as many other factors having an effect on a large differentiation of development teams productivity.

However it should be pointed out that relatively few development organizations possess appropriate resources of own benchmarking data as the condition to have them is not only effective implementation of measurement programmes, what *per se* is not a frequently found phenomenon, but having collected such data for relatively large number of similar projects having been executed in the past and, additionally, referring them to the right unit of software system size (see e.g., [11]). Even more such situation may be found in Poland where FSM methods, including the IFPUG function point method, have been employed for relatively short time [9]. This is when the usefulness of repositories with general data, offered by organizations such as for example International Software Benchmarking Standards Group (ISBSG), comes out. It is worth mentioning that according to C. Jones's estimations there are dozen or so resources of benchmarking data for the discussed types of projects now yet definite majority of them are not widely available. What is more, part of them feature data concerning relatively little number of projects, and also – they not always relate to the IFPUG FP method [12].

2.1 The ISBSG data

2.1.1. The ISBSG data repository

At the moment the ISBSG is an organization that provides the largest, commonly recognised and accessible repository containing general benchmarking data for DSS D&EP whose products are measured with the use of the IFPUG function point method [13]. The ISBSG is a *non-profit* organization that was established in the second half of the 1990s with the mission to enhance processes of software projects execution in business entities as well as in public administration institutions. This mission is being fulfilled by developing, maintaining and exploiting three kinds of repositories with benchmarking data. One of them, the largest one (current version of repository contains data concerning over 5600 projects from 29 countries), comprises data for development and enhancement projects. It is normalised in accordance with the ISO/IEC 15939 standard [14], verified and representative of current technology.

Data collected in the discussed repository are being classified by the ISBSG with regard to the following criteria – they are of importance as they have an effect on how high are per-unit costs with regard to 1 IFPUG FP ([15][16]):

- country where project was undertaken
- context of the project, including: type of organization and business area
- type of project, including: type of activities (enhancement of the system or development of the system from scratch), purpose of the project and size of development team
- type of product, including: type of application and product size (in definite majority of cases expressed in the IFPUG FP)
- project execution environment, including: programming language and hardware platform
- project development methods and tools being used.

However, when using data gathered by this organization one should keep in mind that these data are rather representative of the above-average projects, which results from the following facts:

- Criteria of data collection for ISBSG repository take into account only those organizations that use FSM methods, including the IFPUG FP method above all, and these organizations are considered more mature than the others as they accomplish programmes concerning implementation of software measures.

- Data to be included to the ISBSG repository are chosen by the providers themselves – they may choose projects that are typical of them as well as projects characterised by the best attributes.
- The ISBSG repository does not include a good deal of data about really large projects.

However, one has to point up that those data are subject to rigorous process of verification with regard to quality. Thus the ISBSG data are valued in the IT industry while general conclusions coming from their analysis are consistent with the conclusions resulting from the analysis of other organizations benchmarking data repositories.

2.1.2. Per-unit costs according to ISBSG data

The ISBSG produces cyclical analytical reports based on the data concerning DSS D&EP. What appears of significance from the perspective of the subject matter being discussed in this paper is the ISBSG report titled "Software Project Costs" [17], which analyses the size of per-unit costs with regard to 1 IPFUG FP. Data analysed therein indicate that:

1. For definite majority of cases, per-unit costs measured with regard to the product functional size unit (1 IFPUG FP) range from USD 300 to USD 1000 per 1 FP, with an average of about USD 750 per 1 FP. Taking into account all analysed projects, the spread is from USD 17 to USD 2727 per 1 FP (extreme values for the so-called outlier projects) while the cost median is USD 716 per 1 FP. These costs are measured by taking into account development team and support personnel (e.g., data base administrators) – they are approx. 15% higher than costs estimated for development teams only.
2. For definite majority of projects, per-unit costs measured with regard to the work time unit (1 hour) range from USD 60 to USD 105 per hour, with an average of about USD 80 per hour. Taking into account all analysed projects, on the other hand, the spread is from USD 7 to USD 570 per hour (extreme values for the outlier projects) while the cost median is USD 69 per hour and the mean is USD 84 per hour. As in the previous case, these are costs measured with development team and support personnel being taken into account.

On the basis of the above, the ISBSG recommends employing the following rules of thumb for the discussed projects:

1) cost per 1 IFPUG FP ranges from USD 300 to USD 1000, with an average of about USD 750 per 1 FP
2) cost per 1 hour ranges from USD 60 USD to USD 105, with an average of about USD 80 per 1 hour.

What is more, the ISBSG data indicate that PDR (*Project Delivery Rate*)[1] median, that is middle value of the number of person-hours necessary to deliver 1 IFPUG FP, ranges from about 8 to 11 person-hours per 1 FP – mainly depending on the project type, software system (product) type, application area and technology[2]. Besides, productivity is significantly lower (that is PDR is higher) in case of projects consisting in enhancement of software systems rather than in case of projects consisting in developing such systems from scratch [18, pp. 8, 13, 15, 22]. Taking into account those values together with the cost per hour gives us the spread of costs from USD 480 per 1 FP to USD 1155 per 1 FP, that is on average from USD 640 to USD 880 per 1 FP, which roughly confirms the conclusions coming from the above analysis of the unit cost per 1 IFPUG FP.

Moreover, if project is executed by an outside provider, one should differentiate internal per-unit costs (provider's per-unit work costs) from external ones (per-unit costs offered by provider to a client, including profit as well). According to the ISBSG, the latter usually exceed internal per-unit costs by 2.5 to 3 times, and in big corporations even by 6 times [19, p. 128].

Per-unit cost measured with regard to 1 IFPUG FP for given types of applications reads for example as follows: web and content management applications – USD 800 per 1 FP, CRM and administration applications – USD 400 per 1 FP, report generators – USD 200 per 1 FP.

2.2 Other sources of benchmarking data

As mentioned above, per-unit costs of DSS D&EP with regard to 1 IFPUG FP depend on numerous factors, which has been the subject of studies carried out by Capers Jones, among others (see e.g., [20, pp. 24-26]). In Table 1 and in Table 2 we present how those costs depend on work costs that vary from country to country. On the other hand, Table 3 shows the so-called effectiveness of exemplary programming languages and several tools, by which we understand the average number of source lines of code required to deliver 1 IFPUG FP depending on the programming language/tool being used.

Table 1. Countries with the highest average per-unit costs (per 1 IFPUG FP) in USD

No.	Country	Per-unit costs (per 1 IFPUG FP)
1.	Japan	1600
2.	Sweden	1500
3.	Switzerland	1450
4.	France	1425
5.	United Kingdom	1400
6.	Denmark	1350
7.	Germany	1300
8.	Spain	1200
9	Italy	1150
10.	USA	1000

Source: [21, p. 29].

[1] PDR is the inverse of productivity, being the ratio of the number of function points to the effort (work effort). Naturally PDR depends on a number of factors – there are nearly 50 such factors mentioned in the ISBSG repository.

[2] In this case median is a value more reliable than arithmetic mean as the impact of several atypical (the so-called outlier) projects is thus avoided.

Table 2. Countries with the lowest average per-unit costs (per 1 IFPUG FP) in USD

No.	Country	Per-unit costs (per 1 IFPUG FP)
1.	India	125
2.	Pakistan	145
3.	*Poland*	*155*
4.	Hungary	175
5.	Thailand	180
6.	Indonesia	185
7.	Venezuela	190
8.	Columbia	195
9	Mexico	200
10.	Argentina	250

Source: [21, p. 30].

Table 3. Programming languages table – fragment for the selected languages and tools*

Programming language/tool	Average number of lines of code per 1 IFPUG FP
Assembly languages	320
C	128
Basic (interpreted)	128
COBOL	107
FORTRAN	107
Basic (compiled)	91
Pascal	91
PL/I	80
Ada83	71
Lisp	64
Prolog	64
C++	53
Java	53
Ada95	49
AI Stell	49
Visual Basic	32
Delphi	29
Smalltalk	21
HTML	15
SQL	12
First generation languages (1GL)	320
Second generation languages (2GL)	107
Third generation languages (3GL)	80
Fourth generation languages (4GL)	20
Object languages	30
Report generators	80
Code generators	15
Spreadsheets	6

* This table comprises about 600 programming languages and is continually updated. Its current full version may be found on the Software Productivity Research website: http://www.spr.com/products/programming.shtm.

Source: [22, p. 117] and [23, p. 78].

What is more, in the subject literature one may also find a common view about occurrence of the phenomenon of diseconomies of scale in case of DSS D&EP [24]. This means that as the system size (measured e.g., in IFPUG FP) increases, per-unit costs grow too, and they do not decrease instead - which is contrary to the situation taking place in vast majority of other projects, including engineering ones. Data displayed in table 4 confirm this phenomenon, at the same time showing how per-unit costs for development and implementation are being determined.

Table 4. Average per-unit costs per 1 IFPUG FP with regard to the software system size in IFPUG FP

Number of IFPUG FP	Per-unit costs (per 1 IFPUG FP) for development	Per-unit costs (per 1 IFPUG FP) for implementation	Per-unit costs (per 1 IFPUG FP) - total
1501 – 2000	242	725	967
2001 – 2500	255	764	1019
2501 – 3000	265	773	1058
3001 – 3500	274	820	1094
3501 – 4000	284	850	1134

Source: [25].

It should be noted, however, that some studies have appeared recently, indicating quite an opposite phenomenon, that is occurrence of economies of scale in the execution of the discussed projects, which means decrease in costs per unit with the increase in software system size at the same time ([18][24]). This, however, applies only to specific types of systems and those D&EP projects with relatively little increase in product size.

What also is of significance to the subject matter considered here is the fact that according to the studies by C. Jones, consultants carrying out analysis with the use of the IFPUG FP method charge on average USD 5 per 1 function point calculated [26, p. 3.].

3 Concluding Remarks

The above presented data vary greatly - as there is no possibility to derive accurate values for the per-unit cost calculated with regard to 1 IFPUG FP without taking account the specificity of given development organization. Since this cost has influence on a number of factors – major ones were mentioned in the paper. However, lack of own (organizational) resources of adequate benchmarking data continues to be common situation - not only in Poland but worldwide as well. Hence there is the necessity to employ general data.

On the basis of the above presented general benchmarking data for DSS D&EP it should be stated that adopting per-unit cost for enhancement project on the level of 1 cent per 1 IFPUG FP entails the following paradoxes:

- Such cost is 1 700 times lower than the lowest per-unit cost noted in the ISBSG repository – considering per-unit cost for development team alone will not change this fact considerably (then it will be nearly 1 500 times lower).
- Such cost is 30 000 times lower than the lowest per-unit cost recommended by the ISBSG for dedicated software systems.
- Such cost is 75 000 times lower than the average per-unit cost recommended by the ISBSG for dedicated software systems.
- Given that this is an internal cost, the costs of 8 to 11 hours of work are estimated to be 1 cent – yet enhancement is characterised by significantly lower productivity than development of the system from scratch. In case of external cost, those costs are estimated to be even lower as the internal per-unit cost, with the lowest difference resulting from the ISBSG data being taken into account, is 0.4 cent per 1 FP.

- A question then arises whether this very low per-unit cost already takes into account the phenomenon of diseconomies of scale, that is increase of such cost together with the increment of system size.
- Comparing with the per-unit cost of the cheapest per system size unit types of application, such cost is 20 000 times lower.
- Comparing with the average per-unit cost for Poland such cost is 15 500 times lower (additionally it should be assumed that per-unit costs for Poland have grown since 2000 due to the increase in work costs).
- Assuming that even most efficient programming languages (of fourth generation) will be used for the software system enhancement, such per-unit cost means that writing 2 000 lines of code costs USD 1 on average.
- This cost is 500 times lower than the average consultant's pay for 1 calculated function point – even if this pay is significantly lower in Poland, it still without doubt is repeatedly higher than 1 cent.

In view of the above paradoxes, mostly diametrical differences resulting from the comparison of general benchmarking data with the adopted per-unit cost on the level of 1 cent per 1 IFPUG FP, place and time factors (as well as inflation related to them) do not really matter, similarly as the fact whether the per-unit cost is an external or internal cost.

Thus it should be stated that both general data, those collected in ISBSG repository and those coming from other sources having been recognised in the IT industry, as well as common sense rules of rational economic approach unequivocally indicate that *it is not possible to develop, and in particular to enhance software system dedicated to the client's needs at the cost per unit amounting to 1 cent per 1 IFPUG FP*, at the same time assuming the lack of subsidization for those works with maintenance costs or other project-related costs, which naturally should not have happened.

It is worth mentioning that the analysis of likely per-unit costs of the DSS enhancement with regard to 1 FP of the IFPUG method carried out following the above described manner resulted in client rejecting the provider offering such costs in the tender competition being considered.

4 References

[1] M. A. Parthasarathy. "Practical software estimation: function point methods for insourced and outsourced projects", Addison Wesley Professional, 2007.

[2] B. Czarnacka-Chrobot. "The Economic Importance of Business Software Systems Functional Size Measurement", Software Engineering, vol. 1, no 1, Scientific & Academic Publishing, Rosemead, California, USA, 2011, pp. 9-23.

[3] ISO/IEC 14143 Information Technology – Software measurement – Functional size measurement – Part 1-6, ISO, Geneva, 1998-2011.

[4] ISO/IEC 20926 Software and systems engineering - Software measurement - IFPUG functional size measurement method 2009, edition 2, ISO, Geneva, 2003-2009.

[5] ISO/IEC 20968 Software engineering – Mk II Function Point Analysis - Counting practices manual, ISO, Geneva, 2002.

[6] ISO/IEC 24570 Software engineering – NESMA functional size measurement method version 2.1 - Definitions and counting guidelines for the application of Function Point Analysis, ISO, Geneva, 2005.

[7] ISO/IEC 19761 Software engineering – COSMIC: a functional size measurement method, edition 2, ISO, Geneva, 2011.

[8] ISO/IEC 29881 Information Technology – Software and systems engineering – FiSMA 1.1 functional size measurement method, ISO, Geneva 2010.

[9] B. Czarnacka-Chrobot. "Analysis of the Functional Size Measurement Methods Usage by Polish Business Software Systems Providers"; in: Software Process and Product Measurement, A. Abran, R. Braungarten, R. Dumke, J. Cuadrado-Gallego, J. Brunekreef, Eds., Lecture Notes in Computer Science, vol. 5891, pp. 17–34, Springer-Verlag, Berlin-Heidelberg, 2009.

[10] IFPUG, "Function point counting practices manual, release 4.3", Part 0-5, International Function Point Users Group, NJ, January 2010.

[11] B. Czarnacka-Chrobot. "The Role of Benchmarking Data in the Software Development and Enhancement Projects Effort Planning"; in: New Trends in Software Methodologies, Tools and Techniques, H. Fujita, V. Marik, Eds., Frontiers in Artificial Intelligence and Applications, vol. 199, pp. 106-127, IOS Press, Amsterdam-Berlin-Tokyo-Washington, 2009.

[12] C. Jones. "Sources of Software Benchmarks", Version 13, Capers Jones & Associates LLC, November 2011.

[13] http://www.isbsg.org.

[14] ISO/IEC 15939 Systems and software engineering – Measurement process, ISO, Geneva 2002-2007.

[15] International Software Benchmarking Standards Group. "Release 10 Repository Demographics", ISBSG, Hawthorn, VIC, January 2007.

[16] International Software Benchmarking Standards Group. "Data demographics release 11", ISBSG, Hawthorn, Australia, June 2009.

[17] International Software Benchmarking Standards Group. "The ISBSG Special Analysis Report: Software Project Costs", ISBSG, Hawthorn, VIC, June 2005.

[18] Ch. Symons. "The Performance of real-time, business application and component software projects", COSMIC and ISBSG, September 2009.

[19] "Practical Software Project Estimation", P.R. Hill, Ed., ISBSG, McGraw-Hill, 2010.

[20] L. Buglione. "Some thoughts on productivity in ICT projects, version 1.3", WP-2010-01, White Paper, August 23, 2010.

[21] C. Jones. "Software Benchmarking: What Works and What Doesn't?", Boston SPIN, November 2000.

[22] M. Flasiński. "Zarządzanie projektami informatycznymi" [„IT Projects Management"], PWN, Warsaw, 2006.

[23] C. Jones. "Software Assessments, Benchmarks, and Best Practices", Information Technology Series, Addison-Wesley, 2000.

[24] B. Czarnacka-Chrobot. "(Dis)economies of Scale in Business Software Systems Development and Enhancement Projects"; in: Proceedings of the 10th International Conference on Software Engineering Research and Practice (SERP'11), The 2011 World Congress in Computer Science, Computer Engineering & Applied Computing (WORLDCOMP'11), H.R. Arabnia, H. Reza, L. Deligiannidis, Eds., Vol. I, pp. 80-86, CSREA Press, Las Vegas, Nevada, USA, 2011.

[25] P. Ratford, R. Lawrie. "The Role of Function Points in Software Development Contracts", White Paper, Charismatek, 2000.

[26] C. Jones. "A New Business Model for Function Point Metrics", Version 10.0, Capers Jones & Associates LLC, August 2009.