Benchmarking of Software Production Costs: Results & Recommendations

Leandro Pereira, PhD, Cláudia Teixeira
ISCTE-IUL
Lisboa, Portugal
leandro.pereira@iscte-iul.pt
csbta@iscte-iul.pt

***Abstract* — this paper intends to provide a deeper understanding about the software production cost in the banking sector through a benchmarking analysis. Furthermore, the paper provides a set of recommendations to assist organizations become more efficient, productive and more competitive. This analysis was performed based on a representative sample in the Portuguese market based on a sample of 21 projects. These projects represent a total of 37.800 hours and a 2.984.000€ budget. The data collection was completed during 2 months, January and February in 2016 through different business research methods namely 15 Interviews to project managers regarding 21 projects and historical data collection. The sample went through a segmentation in two groups due to the dispersion verified among the results between some banks. These two segments were classified as "Classic Retail" - national banks of high dimension, with market recognition and influential in the Portuguese economy - and "Niche Banks"- smaller banks. Briefly, in terms of rates and considering the "Classic Retail" segment, the discrepancy of rates between the two niches is visible, the average rate of Classic Retail is 64.7% above the Niche Banks segment. In regards to productivity, The Classic Retail's average is about 11h per unit of software produced while for the Niche segment it is about 2.6h.**

Keywords — Benchmarking; Function Points; Project Portfolio Requirements; Business Analysis; Costs; Software, Processes

I. INTRODUCTION

The subprime crisis1 originated in the United States, allied to the interdependence of the international markets (globalization), new habits and lifestyles and the increase of uncontrolled consumerism, had a strong impact on the European banking industry, with the cost of capital achieving maximum historical levels. In this depressive context, the number of barriers to the credit offer to customers increased and the client demands in the search for solutions that bring value to them grew, accompanied by an increasingly personalized service. The Portuguese banking sector was no exception and, like other sectors of the Portuguese economy, it was also obliged to improve its competitiveness levels.

According to the World Retail Banking Report, a significant proportion of bank customers considers changing to another institution, demonstrating the downward trend of the traditional customer loyalty. Thus, and despite the overall service quality of a bank being the basis of the trust relationship with the customer, the customer's positive experiences are those that demonstrate greater correlation in customer retention and loyalty.

This new reality forces banks to bet on innovation and to present new services and pioneering products in order to ensure their differentiation and thus increase their ability to compete in such a fierce market - retaining current customers and gaining new customers.

For this purpose information systems take a preponderant role, in particular regarding the Portuguese banking sector. The concern is the imperative productivity gain, both at an internal level, through process automation, and at an external level, adding value to the products/services provided to the customer.

However, in the current scenario of global negative profitability, bank entities have fallen into an excessive concern with cost reduction and into the misjudgment of reducing their CAPEX, i.e. the budget for new investments. Nevertheless, it is in times of economic weakness that it is necessary to strengthen competitiveness and to leverage organizations' profitability.

Yet, the need to rationalize resources, to control waste and to reduce costs is equally a priority and it is in this framework that banks have sought to control costs with their software production, namely without "neglecting" the first objective of creating value.

Moreover, it is established that the implementation of a good banking software is crucial in business fluidity and success as so far it consolidates information and the activity's critical internal processes. Banks have been able to take advantage of the system's technological and computing development and thus improve significantly their business model performance, becoming, however, dependent on a good software that ensures the security of electronic transactions, automate tasks and operations and enables information mass treatment (with a strong impact on the bank's administrative back office). On the other hand, the bet on a good software allows to leverage the expansion of e-commerce, giving strong support to wholesale banking teams2 and financial markets activities (using in branches microcomputers and the software that enables support for the tasks of "advice-sales"), with the coverage rate and users of workstations and terminals reaching about 100%. In this sense, the dependency of information technologies and

computer systems is extremely high and determinant in the banking institutions' success.

To cope with the banking market's increasing competitiveness, banks are forced to reduce their costs with software production, without compromising the quality of their service.

Having this said, this paper main research questions are:

- How to benchmark banking companies performance through a project portfolio perspective?
- How are Portuguese Banks (software production) positioned between each other in terms of rates and productivity performance?

II. Literature Review

A. Function Points Role

According to Wagner & Ruhe [1], the lines of code or the function points are typically used for measurement of productivity in software development.

According to Scacchi [2], software team productivity should be measured to reduce the software development costs, to assist improving the quality of deliverables and to increase the rate at which software is to be developed. Also, measuring productivity helps in identifying underutilized resources [3]. The study of software productivity is important because higher productivity leads to lower costs [4]. So, by knowing the software performance and making productivity improvements can result into substantial amount of savings in development costs [5].

B. Function Points Model

According to Wagner & Ruhe [1], the lines of code or the function points are typically used for measurement of productivity in software development.

According to IFPUG Function Point Analysis [6], the functional size is calculated by quantifying the number of the five types of components that exist, namely: External Inputs (e.g., transaction types entering data into the system), External Outputs (e.g., report types producing information output), External Inquiries (e.g., types of inquiries supported), Internal Logical Files (data maintained by the system), and External Interface Files (e.g., data accessed by the system but not maintained by it). For one thing, logical data files contain Internal Logical Files and External Interface Files indicated respectively by 'ILF' and 'EIF'; for another, functions contain External Inputs, External Outputs and External Inquiries indicated respectively by 'EI', 'EO' and 'EQ'. These components are weighed (according to their complexity-low, average or high), and their weights are summed. This value is then adjusted using fourteen general system characteristics to produce a functional size measure.

Regarding the amount and complexity of the data (types) 'handled' by a logical data file transactional function determines the amount of functionality that this piece of software delivers, therefore its functional size. The boundary separates the software system being measured from its environment. Within this environment are the users of the system, which may include other systems. Fig.1 illustrates a typical FP structure

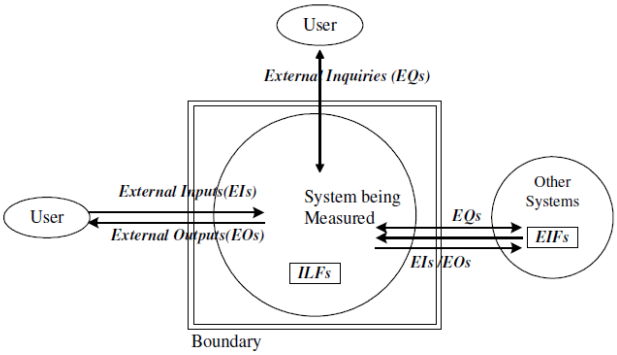

Fig. 1 – Function Points Structure (Zickert, F. & Beck, R. (2013)

C. Function Types and Details

According to Zickert and Beck [6], there are two major function classes: transactional functions and data functions. Transactional functions represent external inputs (EI), external outputs (EO), and external inquiries (EQ). Data functions in FPA are either internal logical files (ILF) or external interface files (EIF).

An EI is an input that originates from the user or another software component outside the counted system boundary. It may also use data from outside the system boundary (EIF) and it updates data inside the system boundary (ILF). For example, when the user enters data into the system is an EI. The entered data becomes an ILF when it is stored in the system. If the data is not entered by a user but sent from another system, the function is still an EI. The data also becomes an ILF when it is received and stored by the system. However, in this case, the origin of the EI is an EIF, which is the file saved at the other system when it is sent [6]

On the other hand, an EO is an output that originates within the boundary of the counted system and uses data (ILF) from inside the system that is transmitted to a user or another software component outside the system. In regards to an EQ, it is an online input that results in an intermediate software response. Its main intent is presenting information to a user through retrieval of data. It does not update an ILF but provides a direct response outside the system boundary. An ILF is a logical group of data that resides within the boundary of the software under construction. On the other hand, an EIF is a logical group of data that resides outside the software boundaries and that provides data for the software [6].

Results from a research, proved that both the interrater and intermethod reliability of FP measurement are sufficiently high so their reliability should be a barrier to their continued adoption and future development. Therefore, efforts in the automation of FP data collection should be pursued [7].

In order to make decisions about software quality, the use of software metrics such as function points reduces subjectivity in the assessment of software quality by providing a quantitative basis. These measures provide a guide to the development of software, particularly in software project management aspects such as goal setting, improving quality and productivity, aiding project planning and management and most important of all, increasing client's confidence. Although software metrics

are used, they do not eliminate the need for human judgement in software evaluations.

III. RESEARCH METHODOLOGY

For the purpose of assessing the costs of software production in the domestic market, an industry benchmarking was performed at the level of the cost of the hourly rate (€/h) in terms of productivity (h/fp), i.e. the number of hours needed to produce a unit of software (a function point is a unit of measurement of software, that express the "amount" of software functionality of an information system delivered to a user). This analysis was performed based on a representative sample of the Portuguese market: 3 projects from 7 different banks, representing a total of 21 projects. These projects have a total of 37.800 hours which represent a total budget of 2.984.000€. The data collection was completed during 2 months, January and February in 2016 through different business research methods namely 15 Interviews to project managers regarding 21 projects and historical data collection.

The method used to treat the data information was the Function Points Method.

IV. DATA RESULTS AND ANALYSIS

In order to reply the proposed research questions, the benchmarking pursued was based on the Function Points Method, which by collecting the needed variables allowed to perform the comparison between the companies selected.

To perform a benchmarking analysis, it is essential the usage of metrics or indicators that allow to perform the comparison between the several companies. The fruit of the conclusions taken by comparison will allow to extract a good reference to leverage changes in processes or management policies in order to improve results.

Having the function points model defined and the required components information collected for each identified project, the authors were able to answer the second question about how each Bank is ranked between each other on terms of rates and productivity performance (benchmarking analysis).

Having pursued the function points technique, it was collected two main metrics: rate per hour (€/h) and the effort spent per function point produced (h/fp) for 21 projects (3 projects from 7 banks). The following matrix in Fig. 2 presents the results:

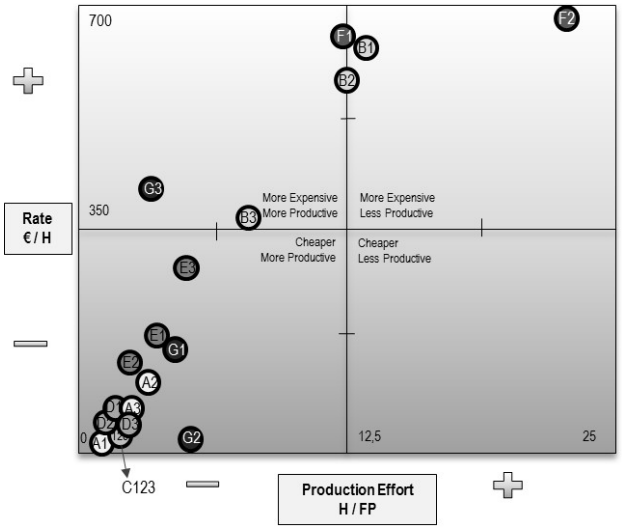

Fig. 2 - Projects Perfomance from 7 banks (€/h) vs (H/FP)

As presented in Figure 2, it is possible to conclude that the majority of the projects present similar performance, mostly located in the 3rd Quadrant, however projects of two banks present a bigger variance compared to the remaining ones.

In order to better understand the software performance in a bank perspective, the following matrix (Fig. 3) was prepared:

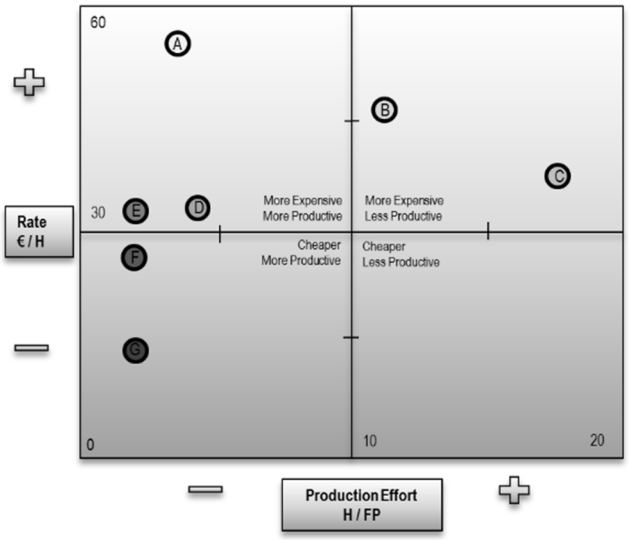

Fig. 3 - Banks Performance (€/h) vs (H/FP)

After analysing the data results, it was performed a market segmentation into two groups due to the dispersion verified among the results between some banks, classified as "Classic Retail" - national banks of high dimension, with market recognition and influential in the Portuguese economy - and "Niche Banks"- smaller banks and therefore with less recognition in the domestic market, and international banks specialized in specific areas (e.g. consumption; investment; among others).

A. *Rates Benchmarking €/H*

On the basis of the banks' costs for each cost centre and based on the bottom-up technique, it was possible to collect the rates charged in each of the banks, with average values shown in Figure 4.

Figure 4 - Charged Rates

In terms of rate and considering the "Classic Retail" segment, the discrepancy of rates between the two niches is visible – the average rate of Classic Retail is 64.7% above the Niche Banks segment. This may be explained by several reasons: the Banks belonging to the Niche Banks are more recent and usually the software development services are sourced through an off shoring or nearshoring format. Therefore, banks belonging to "Niche Banks" are able to get lower production costs hence become more competitive then the banks in the "Classic Retail" segment.

B. *Productivity Benchmarking H/FP*

Because the total cost of production is influenced by the number of hours needed to produce a unit of software, the calculation of productivity is crucial in this analysis.

Figure 5 presents the Ranking of average productivity/Bank (h/fp).

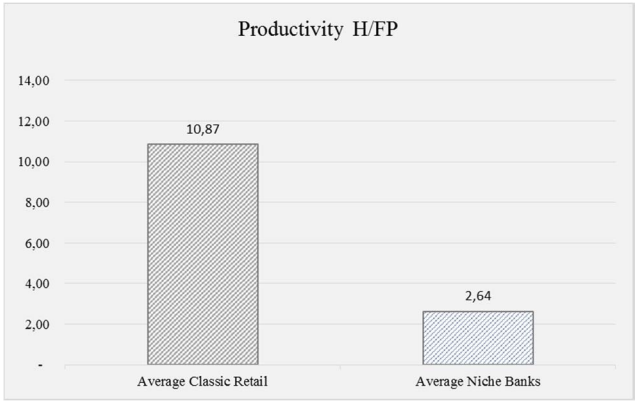

Fig. 5 - Average Productivity per segment type (H/FP)

When analysing the productivity for the Classic Retail segment, the variation between segments is visible. The Classic Retail's average is about 11h per unit of software produced while for the Niche segment it is about 2.6h, i.e. the Classic Retail segment requires more 311% of effort relative to the second segment. This may be explained by the fact that Niche Banks have a smaller dimension have of software services supply then the Classic Banks and are also better prepared with a whole system integration which leverage the productivity performance.

When analysing these metrics among the 21 projects in these 7 banks it was possible to conclude there are different maturity levels in regards to an appropriate scope definition of projects. It was also noticed a lack of a well-defined requirements process and practices which are currently affecting the efficiency of several tasks and human resources allocation.

Some banks also seem having a lack of communication between the Maintenance Department (who act on incidences and corrections) and the Development Department (new projects), which leads to some difficulties in regards to tasks priorisation and resources management.

In addition to our analysis, it is interesting to observe the productivity per industry at the international level, shown in Figure 6. In the Banking market, it was found that on average 8.6h are spent, demonstrating that the national banking sector (Classic Retail) supports 24% more than the average of international banking. Out of curiosity, the Telecommunications industry and Industrial Production are those that require fewer hours per function point production (2h and 3h respectively) as opposed to the world of accounting with 11h and Banking and Insurance with 8.6h.

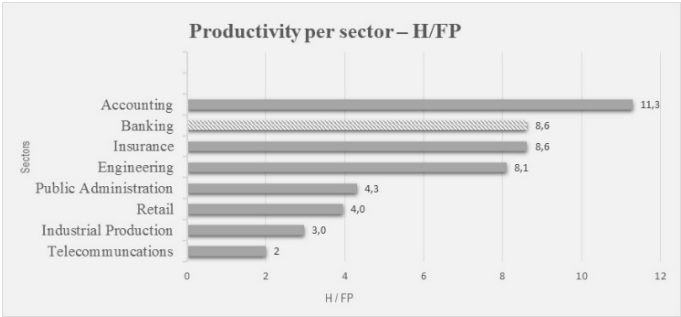

Figure 6 - Effort H/FP per Industry

Source: Data collected by "International Software Benchmarking Standards Group (ISBSG)" and IEEE Software. Website consulted, 2016: http://ieeexplore.ieee.org/

C. *Productivity Benchmarking €/FP*

An interesting point that results from this analysis is that banks that focus on a cheaper rate (€/h) require more hours of effort in producing each unit, becoming the banks with the higher production costs, as shown in Fig. 7.

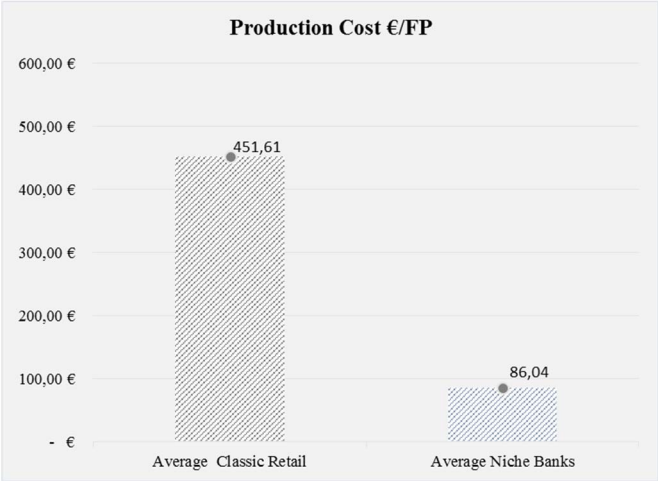

Figure 7- Production Cost €/FP Ranking (elaborated by the authors, 2016)

Classic Retail Banks present, on average, production costs of about 451,61 €/fp, compared to the Niche Banks, which have an average cost of only 86€/fp (438%).

In terms of conclusion, the benchmarking analysis follows the matrix presented below in Figure 8, which reflects the relative position of the respective segments based on rate (€/h) and effort (h/fp).

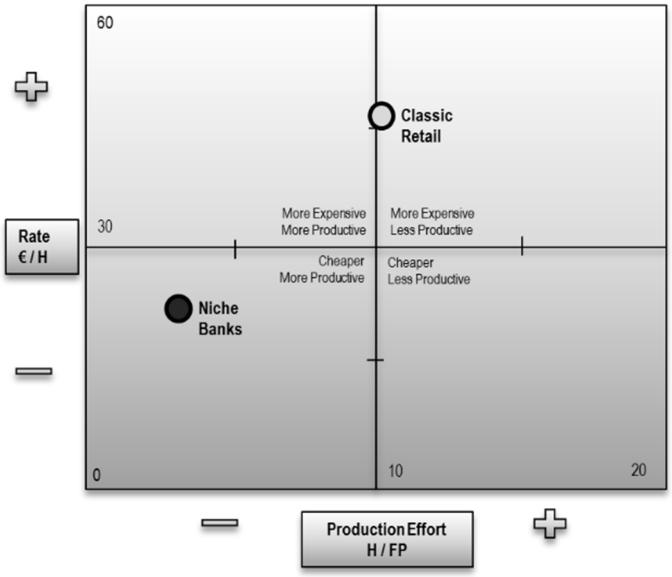

Fig.8 - Productivity Benchmarking Matrix (elaborated by the authors, 2016)

It is clear that the Classic Retail segment will need to undertake the actions necessary to its repositioning into the quadrant "Cheaper More Productive", the quadrant in which it will be maximizing its efficiency, productivity and with lower costs.

Table 1 summarizes the global benchmarking regarding the production of software in banking:

TABLE I. GLOBAL BENCHMARKING

Bank Segment	Rate €/H	Production Effort H/FP	Production Cost €/FP
Classic Retail	46,47 €	10,7 H	463,42 €/fp
Niche Banks	26,63 €	2,6 H	86,04 €/fp

Source: Elaborated by the authors, 2016

After reading Table 1, the difference of the three variables (cost, productivity and production) between the two defined market segments is noticeable. The Classic Retail segment shows much higher values compared to Niche Banks, charging 20€/h more and requiring more 6 hours to produce one unit of software (FP), obtaining an equivalent production cost of 463.42€/fp, representing more 377€ than the Niche Banks segment.

In the process of assessing banking's maturity, it is essential to consider the maturity of the industry in terms of the knowledge area of cost (estimation), in terms of the knowledge area of scope (requirements) and in terms of the knowledge area of quality (performance metrics and continuous improvement plan).

Besides the recommendations mentioned along the data results analysis, there are other ones that are important to consider towards the banking performance in regards to software performance.

The existence of PMO (Project Management Office) formally defined is critical in order to have a functional unit with enough power and responsibility in regards to project management. With a well-established PMO, organizations are better prepared to provide a good planning of all projects, an estimate model that is normalized and applied across the enterprise, the existence of a well consolidated portfolio reporting, the existence of an integrated information system for project management and the standardization and independence of the testing process. While doing this, it is crucial to integrate the estimate model with the Work Breakdown Structure, time report and control, as well as to professionalize the functions of Business Analyst and Project Manager.

Sometimes, some organizations face a disorganization in the company employee's responsibilities definition. In these cases, it is fundamental to analyse the entity's current organic structure and redefine it in order to define the correct company responsibility in the requirement's definition.

Another important aspect is to assure an efficient and effective communication. In order to maximize the communication's efficiency and effectiveness and the involvement of the responsible people effectively needed, it is recommended to create a team of Business Analysis experts by each "internal client" together with a well stablished requirements management methodology with meta-language and standardization of formal synthetic concepts.

On the other hand, to better monitor and control processes, it is essential that banks create productivity metrics in order to measure and evaluate the entire production cycle, from requirements to corrective maintenance, evolution of the estimates 'model to incorporate confidence analysis and Function Points, thus avoiding Parkinson's law (work expands in order to fill the time available (deadline) for its realization).

Furthermore, in a few banks it is spent a high volume of time in the incidents resolution process, so this brings the need to maximize the efficiency of this processes. In terms of maintenance, it is important to optimize the process through artificial intelligence and knowledge base in order to "depersonalize" it. In addition, it is recommended to integrate the corrective maintenance process with the development and finally to implement a continuous improvement process cross-sectional and integrated with the projects and people.

The implementation of these recommendations allows the increase in team's efficiency, resources' maximization and results' monitoring. This will contribute to reduce the level of effort (hours) in the software production process and thus contribute to gradually achieve the international banking average of 8.6 h/fp and the average of Portuguese banking of 6.09h/fp.

V. CONCLUSIONS

Considering the research taken place, in order to pursue a software performance analysis namely within the banking industry, the function point model is a very useful and reliable tool since it considers several metrics which allow to determine the level of complexity of the software development within an organization. Having this information and the amount of effort (time) spent it is possible to calculate the level of productivity per function point produced and the according production costs.

Regarding the banking benchmarking it is clear the efficiency difference between the two types of Banks (the national banks of big dimension versus the niche banks of small dimension or specialized in specific areas) in regards to software production. Smaller banks, due their structure, are able to be more efficient in software production and consequently more competitive then Classic Banks.

The research authors strongly recommend that this research should be further developed in the future where all the recommendations results should be duly validated.

VI. REFERENCES

[1] Wagner, S., & Ruhe, M. "A Systematic Review of Productivity Factors in Software Development. Proceedings of 2nd International Workshop on Software Productivity Analysis and Cost Estimation" (SPACE 2008), State Key Laboratory of Computer Science, Institute of Software, 2008.

[2] Scacchi, W. "Understanding Software Productivity. Article in software Engineering and Knowledge Engineering: Trends for the Next Decade edited by D.Hurley", Vol. 4, World Scientific Press, 1995.

[3] Nwelih, E., & Amadin, I.F. "Modeling Software Reuse in Traditional Productivity Model" Asian Journal of Information Technology, 7(11):484-488, 2008.

[4] Bouchaib, B., & Charboneau, R., "An Evaluation of Productivity Measurements of Outsourced Software Development Projects: An Empirical Study", 2005.

[5] Sudhakara, G.P. & Farooqband, A. & Patnaikc, S. "Measuring productivity of Software Development Teams" Serbian Journal of Management 7 (1), 65-75, 2012.

[6] Zickert, F. & Beck, R. "A mapping model for assessing project effort from requirements". Information Systems and eBusiness Management 11(3), 377-401, 2013.

[7] Kemerer, Chris F. "Reliability of function points measurement." Association for Computing Machinery. Communications of the ACM, 36.2 : 85, 1993.