An Effort Prediction Model Based on BPM Measures for Process Automation

Banu Aysolmaz[1], Deniz İren[2], and Onur Demirörs[1]

[1] Informatics Institute, Middle East Technical University, Ankara, Turkey
{aysolmaz,demirors}@metu.edu.tr
[2] Computer Center, Middle East Technical University, Ankara, Turkey
diren@metu.edu.tr

Abstract. BPM software automation projects require different approaches for effort estimation for they are developed based on business process models rather than traditional requirements analysis outputs. In this empirical research we examine the effect of various measures for BPMN compliant business process models on the effort spent to automate those models. Although different measures are suggested in the literature, only a few studies exist that relate these measures to effort estimation. We propose that different perspectives of business process models need to be considered such as behavioral, organizational, functional and informational to determine the automation effort effectively. The proposed measures include number of activities, number of participating roles, number of outputs from the process and control flow complexity. We examine the effect of these measures on the automation effort and propose a prediction model developed by multiple linear regression analysis. The data were collected from a large IS integration project which cost 300 person-months along a three-year time frame. The results indicate that some of the measures collected have significant effect on the effort spent to develop the BPM automation software. We envision that prediction models developed by using the suggested approach will be useful to make accurate estimates of project effort for BPM intensive software development projects.

Keywords: Business process model measures, business process automation, project management, effort prediction model.

1 Introduction

Various measures have been suggested and utilized for business process models in the literature for different purposes such as; evaluating quality [10, 21], understanding the error probability [16], assessing the understandability and maintainability [8, 9, 20], measuring the similarity between models [7] and measuring functional size for the software to be developed based on the process models [12].

The business process models are commonly used as a tool to analyze requirements in early stages [2, 18]. When available, business process models contain valuable information on the size of the system to be automated in early phases of the development life cycle [12].

S. Nurcan et al. (Eds.): BPMDS 2013 and EMMSAD 2013, LNBIP 147, pp. 154–167, 2013.

Only a few studies exist in the literature analyzing the size of business process models to be used as a basis of system development effort prediction [12, 18]. These studies suggest a methodology to determine the functional size of the software to be developed indirectly, by estimating the size with respect to well-known functional size measurement methods.

Currently, we develop an Integrated Campus Information System in Middle East Technical University (METU) based on the university's business processes. This project aims to automate most of the processes of the university including more than 90 process modules. For each process, business process analyses are being conducted and business processes are being defined as models and structured process definitions are being written. The project is divided into phases in which a predefined set of process modules are automated. For each of the phases, the project management office requires an estimation of the automation effort to use for planning, budgeting and subcontracting the project.

In this study, we explain our empirical research focusing on developing an effort prediction model which is based on direct measures on business process models. We aimed at identifying the effect of various business process model attributes to the automation effort and suggest an effort estimation model by using multiple linear regression analysis technique. The measure set we used covers different perspectives of business process models which are functional, behavioral, organizational and informational. We used the historical data gathered over the last three years covering approximately 300 person-months of effort and 10 business process modules of varying sizes which have been analyzed, modeled and automated into working software.

The remainder of the paper is organized as follows. In the following section, we provide a brief overview of the existing literature. In the following section, we describe the case, introduce the research problem together with the proposed solution, explain the case study plan and identify the selected set of business process model measures together with the rationale for selection. In the fourth section, we describe the empirical approach used including the data collected, the method applied and the results. The last section provides a discussion of conclusions, limitations and future work.

2 Related Work

There is an extensive body of literature on definition and discussion of various business process model measures. Business process measures are usually derived from software metrics [1, 10, 21, 30]. Process size is often determined by using a corresponding software size metric; "Line of Code". Some of the size measures used are; number of activities, joins and splits [1, 16], diameter, density [16], and cross-connectivity [22].

Another basic measure derived from software measurement domain is complexity. Cyclomatic Complexity is usually translated to business process models as Control Flow Complexity (CFC) where different types of splits are handled separately [21].

Other complexity measures have also been suggested such as; Halstead-based Process Complexity, Coefficient of Network Complexity [17], nesting depth, jump-outs from control structure, cognitive complexity [9]. We observed that CFC is the most commonly used complexity measure for business process models [1, 9, 21]. It is important to mention that CFC only covers control-flow complexity, and disregards complexity that might have been introduced because of data elements captured within process models.

Fan-in and fan-out metrics are used in different ways like inputs and outputs [17] and reference numbers to and from a process module [9]. Measures like coefficient of connectivity, separability, sequentiality, depth, structuredness were suggested to measure error proneness of the process models [16, 30].

There are other measures that attempt to involve data complexity. Interface complexity measure [1] includes number of inputs and outputs in the process. Coupling refers to the number of activity couples which contain one or more common data elements and cohesion expresses the coherence within the activities of the process model [17]. Dhammaraksa and Intakosum [6] emphasize that current size measures do not consider all perspectives of process models; namely functional, behavioral, informational and organizational. In their research they provide measures for each perspective.

Many studies include further analysis and empirical work to use measures to evaluate error probability in process models, understandability and complexity. Mendling et al. [16] show that there is a strong connection between formal process model errors and a set of measures on structural and behavioral aspects of process models. In another research they emphasize that there is a negative correlation between size and quality aspects. Mendling et.al. [23] illustrate that higher density of arcs and larger number of paths in a model affects the understandability negatively.

We observe that many measures are suggested to measure the size of business process models. But these measures are collected for other purposes, like analyzing understandability and error-proneness of models. There are a few research published just in recent years focusing on business process size in order to evaluate its effect on development effort of related software systems by using COSMIC model.

COSMIC is a widely accepted method for functional size measurement of software, and accepted as an international standard [3, 11]. An extensive survey on conceptual model based functional size measurement research [15] revealed that all studies calculate size from UML diagrams. We found other studies focusing on using business process models to measure COSMIC size. Lavazza and Bianco [14] used UML activity diagrams to estimate the COSMIC functional size of the system. Their approach is indirect effort estimation method from business process models.

Monsal and Abran [18] defined a set of BPM rules for software to be developed, the users of the software and data movements. By conforming to these rules during modeling, it is assured that COSMIC size measurement can be calculated. The authors developed rules for Qualigram notation (their own modeling notation) and BPMN. Another important study by Kaya and Demirörs [12] describes the size measurement method based on EPC notation. For each function in EPC, further analysis is conducted (namely FAD) to determine data movements, users and related systems.

With this method, COSMIC size is calculated by counting data movements for related systems. The approach is also automated on a modeling tool. These approaches aim to measure the size of the system to be automated based on process models, rather than determining measures of business process size that can be used to identify the automation effort.

3 The Case Study

3.1 Description of the Case

METU has 24.000 students currently enrolled. There are 40 undergraduate departments and 160 graduate programs. METU also has 21 interdisciplinary research centers. More than 5.000 personnel are working for the university. A large number of IT systems have been developed since the establishment of the university Computer Center. However most of these IT systems run independently, not communicating with each other and using various technologies for data storage and communication. As a result of this crowded, complex environment, problems emerged such as out-of-control duplication of data, non-standard communication, lack of control over IT service levels and very high maintenance costs.

The Integrated Campus Information System (ICIS) project was initiated in 2009 by Computer Center in order to solve these problems. ICIS aims at integrating the existing IS applications in accordance with the university strategic plan. Initially Computer Center developed a business process map consisting of all the business processes of METU. These processes were prioritized in line with the master project plan.

Since the beginning of the project, the activities of analyzing and modeling of business processes and eventually developing software running on automated business process models are conducted iteratively for each process module. More than 300 person-months are utilized in the last 3 years.

The method for developing process automation can be summarized as follows. First all stakeholders involved in the business process are contacted. Then modeling experts start analysis sessions with the stakeholders. The analysis team puts in additional effort to produce process definitions documentation. The process definition documents contain textual definitions of processes organized in sets and subsets, stakeholders, business rules, risks, inputs and outputs of the process and data elements. Then the BPMN models of the processes are developed in accordance with the textual definitions. The data elements are fed into the university data dictionary. In compliance with the data element definitions, the web services are implemented and SOA mediation layer are integrated with them. Then the software models associated with the process model are developed. After functional testing the process automation software to the university portal are integrated via the user interface portlets.

The models are developed by using Eclipse BPMN modeler, in compliance with BPMN 2.0 standard. Activiti is utilized as the underlying process engine. Business rules are represented in Drools. All programs are coded with Java, JSP and Javascript.

Considering this software lifecycle, we observe that work outputs produced in BPM software automation projects differ from traditional software artifacts. Usually, in addition to traditional artifacts like software requirements specification and software design description documents, business process models are developed. These models are utilized as the basis for the development activities, as the code is produced focused on each activity presented in the process model.

There are about 90 process modules identified in the business process map. The concept of “process module” is used for a group of process models which are coherent and focus on a specific working area of the organization (like budgeting). All of the process models under a process module are connected to the hierarchical structure of the related module. Until now, the team completed developing 10 modules and working on 15 modules at the moment. We were able to collect both business process model measures and total development effort for the 10 modules completed. The university needs to make effort estimation for the rest of the modules to be developed in phases to use as the basis of planning and budgeting.

3.2 Problem Statement and Proposed Solution

Effort estimation for software development is a critical activity for project planning. Widely accepted and commonly used effort estimation methods exist for software development activities producing traditional requirements analysis outputs, like COSMIC function point measurement. BPM software automation projects require different approaches for effort estimation for they are developed based on business process models. Early outputs of the development lifecycle for this kind of projects are business process models which embed information on the requirements of the system. There are a few studies that utilize business process models to determine COSMIC size measurement of the system to be automated. However, to our knowledge, the effect of different business process model properties on the development effort is not studied.

The project management office in METU Computer Center requires all projects to have an effort estimation method based on something more than educated guess that utilizes their existing experience from the project. ICIS project is the first and the only project carried out in the university’s Computer Center in which business process automation is used. Our experience supports our previous observations that the work products of process automation projects differ from traditional software development projects. So, in order to be able to estimate the project effort for upcoming phases of the ICIS project (and potentially other process automation projects), the need for a viable and empirical estimation method emerged.

We were not able to derive an extensive conclusion from the literature for what possible business process model properties we can use to build up an effort estimation method for our case. Thus, the first step of our proposed solution is identification of a set of business process model measures that may have effect on the automation effort. Then, we propose to perform statistical analysis on the collected data to identify the measures that have significant impact on the effort; determine the effect of each measure and formulate an effort prediction model.

3.3 Case Study Plan

The basic procedure common for effort prediction is applied in this study. Following tasks are envisaged within the case study plan:

- Identify the measurable properties of business process models and select the set of measures to be collected that are anticipated to have effect on automation effort,
- Collect data on the selected case; e.g. collect the data for selected measure set and measure the effort spent to automate the processes for each process module,
- Correlate the data collected on business process model measures to the related effort measurements and develop an effort prediction model by multiple linear regression analysis,
- Analyze the effect of each measure on the automation effort and update the model,
- Evaluate the prediction model considering the statistical results and by comparison of the predicted values to real values.

Apparently, many attributes of business process models will have effect on automation effort. But in this study, we specifically aim at identifying the size of the effect in addition to the measures with strongest effect on the development effort. For this, we identify process model measures that compose a sound set to reflect basic size properties of business process models. The selected measures and rationale for selection are explained in the below section.

3.4 Selection of the Business Process Model Measure Set

There are various measures for business process models that may have influence on the automation effort. Our aim in this empirical study is to reveal a set of business process measures that can largely explain the automation effort and suggest an effort prediction model by means of linear regression analysis. These measures shall not be highly correlated to each other and shall represent the business process models meaningfully in BPM domain.

As Curtis et.al. state [4] that to adequately describe a process, four different perspectives shall be taken into consideration: functional, behavioral, organizational and informational. In this research, we consider that our measure set shall cover information regarding each perspective so that we can find out specific contribution of each perspective on the automation effort.

The data is collected from a set of business process models that are developed by the same group of process modeling experts consistently using the same modeling methods. Considering this, we also assume that all of the models conform to basic modeling principles (like block structure [13]), and there is not much deviation between error-proneness and style of the models. Moreover, our aim is not to evaluate how “good” the models are. The nesting depth, which is found to be an important measure for complexity [9], is also similar among models. There are no models with more than 2 levels of nesting depth. Thus we scope out the quality and model structure measures like structuredness, separability, sequentiality [16, 13, 30], correctness of models and nesting depth.

We neither included an extra measure for data complexity like interface complexity suggested by Cardoso et.al. [1] nor other complexity measures such as Halstead-based Process Complexity, Coefficient of Network Complexity measures. We avoided utilizing derived measures and we identified base measures covering each model perspective so that we can observe their effect individually as a result of our linear regression analysis. If we had used derived measures, the high collinearity between independent variables of the regression would have reduced the reliability of the results.

As the initial set of measures, for functional perspective, we selected “number of activities”. We also considered using “number of human tasks” and “number of automated tasks” embedded in our BPMN notation. But to prevent dependency on modeling notation we decided on only choosing number of activities.

We also considered utilizing fan-out measure as an indicator of number of subprocesses referenced from a process diagram. This measure can also add value for representing the functional perspective. However, due to the nature of processes modeled so far in this project there were only a few references to other processes. Thus we were not able to use fan-out as a potential predictor of effort in this research.

Behavioral perspective of a process model represents sequencing and possible alternative flows of the model. Cyclomatic complexity (CC) which is the basic complexity measure for software engineering domain covers complexity introduced by decision points. Control Flow Complexity (CFC) applies a different calculation for each decision point type (and, or, xor). We calculated both measures for our model, but as a result of our statistical analysis we observed that they are highly correlated to each other and CFC has more power on explaining the effort variable. Considering that, we dropped CC and just used CFC for further analysis.

Informational perspective represents the informational entities “produced or manipulated” by the process. To present this perspective, we selected to analyze three measures: number of outputs, number of inputs and total number of inputs and outputs. Although the inputs are used by functions of the system, whole data covered by inputs may not be representing the manipulated entities. As our initial statistical analysis also supported that the number of outputs are better correlated with the effort, we dropped the other measures and used only the number of outputs.

Organizational perspective represents the performers of the activities in process models. To depict this perspective, we chose “the total number of roles” in each process model.

We considered using other measures that may affect the automation effort; like number of business rules, number of performance indicators. As our sample size is limited and these measures are not about the key aspects of models, we left the analysis of these measures for future work.

For reasons provided we selected the measures that are defined below. It should be noted that each process module is composed of multiple subdiagrams. To calculate the following measures, the values of individual subdiagrams are added up as explained below.

- **Number of Activities**: NOA refers to the total number of activities (human and automated) in all subdiagrams of the process module.
- **Control Flow Complexity**: CFC is the sum of all complexity introduced by each split in the models. Joins are not considered in the measure. For each XOR split, the number of splits are counted, and for each AND split, "1" is added to the count. (there are no OR element in the models). The CFC values of all subdiagrams are added up.
- **Number of outputs**: NOO refers to the total number of output elements produced within the process module.
- **Number of roles**: NOR is the number of all roles performing in the related process module.

To summarize, in measure selection, we aimed at identifying at least one measure for each process perspective; functional, behavioral, informational and organizational. These measures are chosen to represent one of these perspectives directly, with the least possible collinearity between each other. That is the reason we focused on identifying base measures rather than derived measures. Additionally, due to the nature of the processes and the aim of the study, we scoped out quality and model structure metrics.

The process models used in this research are developed using BPMN notation [19]. However, the set of symbols used and the measures are common to many business process modeling languages, therefore the results can be applied to other languages.

4 Case Study Implementation

In this section we explain the data collected in this research and the methodology to develop an effort prediction model by using the defined set of process model measures. We conduct linear multiple regression analysis in which we utilize process model measures as independent variables and automation effort as the dependent variable.

4.1 Data

There are 10 completed process modules in the ICIS project for which automation effort (in person-months) was collected. These modules, focusing on research and financial management processes of the university, are considered large, as the average effort for development is 13 person-months. We cannot decompose the effort data into smaller pieces of process modules, so we need to utilize the data from 10 process modules as our sample data. We collected four process model measures for each of these modules as determined in previous section. We are aware that the sample size should be larger to attain more generalizable results. However, due to the nature of software development projects, it is difficult to collect data for large sample sizes. Given that this data have been collected over a three-year period, even the sample size is low, the collected data is still precious and can be utilized to draw conclusions.

The independent variables (NOA, CFC, NOO and NOR) are measured for each process module by examining each process model diagram and process definition documents. The effort data was collected and monitored for each process module as part of project management activities. The effort data covers all activities in the project, including process analysis, modeling and automation. As we think that the activities of process analysis and modeling are part of requirements analysis and software design; we also included their effort in our measurements.

Before starting the regression analysis, the data is examined for outliers. Only one outlier was observed for NOO measure. The related process module is examined and it is concluded that it is a special case because of the nature of the module; and the value is normalized in order not to affect the results. The data collected for the ten sample, including independent variables NOO, CFC, NOA, NOR and the dependent variable effort can be seen in Table 1.

Table 1. Collected Data

NOO	CFC	NOA	NOR	Effort (person-months)
3	9	15	5	5
1	8	13	2	8
2	13	23	4	3
12	27	61	6	22
13	37	100	13	18
2	0	9	4	3
9	74	167	20	20
5	63	72	16	24
2	2	24	11	4
11	30	35	13	22

4.2 Application of Multiple Linear Regression Analysis Method

In our study, we want to examine the effect of the independent variables (observe the correlation of these variables between each other, if exists) and fit a predictive model to our data to use for prediction of our dependent variable; effort. We assume that there is a linear relation between each of the independent variables and the dependent variable. The rationale behind this assumption can be explained; as work products –in this case process models- grow in size and complexity, the automation effort increases in a linear way. We already know from our experiences and due to the nature of the process modeling and software development activities that all of our process measures are positively correlated with the effort. To further analyze the effect of each measure and to identify their impact on the automation effort with a prediction model, multiple linear regression analysis method was applied.

The process measures are expected to be positively correlated to the effort, and if all of the measures are zero (e.g. there is no process model at all), the effort will be zero, too. This is why we forced the regression coefficients to be calculated through the origin and we have intercept value of zero.

In the first step, all of the four measures are added to the regression analysis. We observed very high collinearity statistics (> 0.9) between CFC and NOA. This high collinearity also caused the regression results to be insignificant. Further analysis of correlation between CFC and NOA supported that NOA is highly dependent on CFC. We dropped NOA from our regression analysis, as we concluded that we already cover information provided by NOA measure by means of CFC.

Conducting further analysis with the three measures; CFC, NOA and NOR, we observed that the regression coefficient for NOR is insignificant and the value of the coefficient is also very low. We concluded that the effect of number of roles (NOR) measure is insignificant for our prediction model; and we cannot identify a meaningful coefficient with our limited sample data. As a result, we dropped this measure from our model.

With the two independent variables; CFC and NOO; we ran our regression model and observed statistically significant results. The results are provided in the following section.

4.3 Results

A linear multiple regression model was used to develop an effort prediction model from the process model measures. The potential predictor measures used in the analysis were; Control Flow Complexity (CFC) and Number of Outputs (NOO). Table-2 provides descriptive statistics, where effort is shown in person-months. Table-3 shows zero order correlations which are statistically significant ($p<.01$) along with regression coefficients which are also statistically significant ($p<.05$).

Our prediction model was able to account for a 79.8% of the variance in automation effort, $F(2,7)=18.806$, $p<.01$, Adjusted $R2=.798\%$. The mean magnitude of relative error (MMRE) of the model was calculated as MMRE=30.20%. The prediction quality of our model was PRED(30)=0.60. This means that by using the prediction equation with inputs of NOO and CFC values, the model can predict the effort for 60% of the sample with less than 30% deviation.

Table 2. Descriptive Statistics

	Mean	Std.Dev.	N
Effort	12,90	8,9	10
NOO	6,00	25,4	10
CFC	26,30	4,7	10

Table 3. Regression coefficients and zero order correlations

Model	Beta	Sig.	Correlations	
			Zero-order	Sig.
NOO	,528	,025	,806	,002
CFC	,513	,022	,813	,002

a. Dependent Variable: Effort

5 Conclusions, Limitations and Future Work

5.1 Conclusions

In this empirical research, we proposed an approach to determine a business process model measure set that can affect the automation effort of the related process models. Then, we conducted a linear multiple regression analysis on a set of data collected from Integrated Campus Integration System (ICIS) Project, which is a process automation project running over 3 years with more than 300 person-months of effort. The independent variables are selected to be four measures representing different perspectives of BPM; Number of Activities (NOA), Control Flow Complexity (CFC), Number of Outputs (NOO) and Number of Roles (NOR); and the dependent variable is the automation effort.

The resulting prediction model was able to explain a large amount of variance on the effort which was caused by the predictor variables (79.8%). The analysis results were statistically significant with two independent variables; CFC and NOO having almost the same effect on the automation effort. Also the evaluation of predictive quality of our model was performed by using mean magnitude of relative error and PRED(q) criteria, which is a widely accepted way of determining estimation quality in software engineering domain [24]. According to Conte et al. the it is desirable to have PRED(25) calculation over 0.75. Our model fails to achieve this level of prediction quality, which we explain with the low level of precision due to the lack of enough samples.

Through the steps of statistical analysis, we have scoped out NOA and NOR measures from the analysis. This does not mean these measures have no effect on the automation effort. We observed high collinearity between NOA and CFC. Our early analysis results showed CFC had larger effect on the effort. The primary cause of this result is that, CFC not only expresses the size but also the complexity of the model; meaning that CFC measure already covers the effect of NOA measure, resulting in no need to use NOA as an additional measure. We also observed that the NOR measure had relatively smaller effect on the effort thus omitted. However we suggest that this effect might be observed with better accuracy given a larger sample.

As a result of our linear regression model, we conclude that CFC and NOO have the most significant effect on the automation effort among our BPM measures; and both have almost the same weight of effect.

5.2 Limitations and Future Work

Small sample size in statistical analysis is a major thread to the validity of this research [5]. However, it is a well-known fact that larger samples are sometimes very difficult and time-consuming to establish in software domain [25]. Usage of statistical methods even with small sample size is common in software engineering domain [26, 27, 28, 29]. The result of such cases should be dealt with caution.

The most important limitation of our research is the small sample size. Still, we pursued our studies considering that the measurements made over the three years of the project covering more than 300 person-months of effort.

During the initial phases of statistical analysis, we dropped the measures; number of inputs and total number of inputs and outputs from the model; as we observed better correlation with number of outputs (NOO) measure. Based on our previous studies [12], we foresee that we could achieve better correlation with informational perspective of BPM if we could identify the total number of data movements; which is planned as the future work.

We plan to continue gathering data the same way and form a larger historical data pool in this project. With a larger sample size, we foresee we can attain more significant results for the existing measures and we can further examine the effects of more specific measures such as number of human tasks, number of automated tasks, fan-in, fan-out and number of business rules on development effort.

The current studies which utilize business process models for effort estimation, focus on some specific properties of models; especially informational perspectives [12, 18]. One of these studies is conducted by our research group [12]. We plan to use the results of this empirical research to extend our previous study by inserting different properties of business process models in BPM size and automation effort estimation method. We also believe that the approach and results of this study will contribute to other researchers to consider how various BPM properties may affect the automation effort.

Usage of statistical research methods in software engineering and business process management domains is rare; especially when the case requires data on effort. This is mostly caused by the difficulties of collecting a large sized sample. In this research we found the opportunity to utilize statistical methods and reach meaningful results. Thus we see this research is important because it exemplifies the usage of statistics as a powerful research method in business process modeling domain. We will be using the prediction model developed in this study for effort estimation of following phases of the project. We plan to replicate the same analysis as the ICIS project progresses in upcoming years to evaluate our prediction model as well as our research method.

We think that this research is important as it utilizes empirical outputs to support the development of estimation models; and we expect enhancements in the model with the collection of more data. We foresee that researchers and practitioners can benefit from this approach by following an analysis similar to the one described in this paper to formulate their prediction model coefficients using their own historical data.

References

1. Cardoso, J., Mendling, J., Neumann, G., Reijers, H.A.: A Discourse on Complexity of Process Models. In: Eder, J., Dustdar, S. (eds.) BPM 2006 Workshops. LNCS, vol. 4103, pp. 117–128. Springer, Heidelberg (2006)
2. Coskuncay, A., Aysolmaz, B., Demirors, O., Bilen, O., Dogan, I.: Bridging The Gap Between Business Process Modeling And Software Requirements Analysis: A Case Study. In: MCIS 2010 Proceedings, paper 20 (2010), `http://aisel.aisnet.org/mcis2010/20`
3. COSMIC: The COSMIC Functional Size Measurement Method Version 3.0.1, Measurement Manual (The COSMIC Implementation Guide for ISO/IEC 19761: 2003). The Common Software Measurement International Consortium (COSMIC) (2009)
4. Curtis, B., Kellner, M., Over, J.: Process Modeling. Communications of the ACM, Special Issue on Analysis and Modeling in Software Development 35(9), 75–90 (1992), doi:10.1145/130994.130998
5. Myrtveit, I., Stensrud, E., Shepperd, M.: Reliability and Validity in Comparative Studies of Software Prediction Models. In. IEEE Transactions on Software Engineering 31(5), 380 (2005)
6. Dhammaraksa, K., Intakosum, S.: Measuring Size of Business Process From Use Case Descriptions. In: Computer Science and Information Technology, ICCSIT 2009, pp. 600–604. IEEE (2009)
7. Dijkman, R., Dumas, M., van Dongen, B., Kaarik, R., Mendling, J.: Similarity of Business Process Models: Metrics and Evaluation. Information Systems Journal 36, 498–516 (2011)
8. Ghani, A.A.A., Wei, K.T., Muketha, G.M., Wen, W.P.: Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models Using Goal-Question-Metric (GQM). IJCSNS International Journal of Computer Science and Network Security 8(5) (2008)
9. Gruhn, V., Laue, R.: Approaches for Business Process Model Complexity Metrics. In: Abramowicz, W., Mayr, H.C. (eds.) Technologies For Business Information Systems, pp. 13–24 (2007), doi:10.1007/1-4020-5634-6_2
10. Guceglioglu, A.S., Demirors, O.: Using Software Quality Characteristics to Measure Business Process Quality. In: van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F. (eds.) BPM 2005. LNCS, vol. 3649, pp. 374–379. Springer, Heidelberg (2005)
11. ISO/IEC: 19761:2003 Software engineering- COSMIC-FFP- A functional size measurement method. International Organization for Standardization, Switzerland (2003)
12. Kaya, M., Demirörs, O.: E-Cosmic: A Business Process Model Based Functional Size Estimation Approach. In: 37th EUROMICRO Conference Software Engineering and Advanced Applications, SEAA, pp. 404–410 (2011)
13. Laue, R., Mendling, J.: Structuredness and its significance for correctness of process models. Information Systems and E-Business Management 8(3), 287–307 (2010), doi:10.1007/s10257-009-0120-x
14. Lavazza, L., Del Bianco, V.: A case study in COSMIC functional size measurement: The rice cooker revisited. In: Abran, A., Braungarten, R., Dumke, R.R., Cuadrado-Gallego, J.J., Brunekreef, J. (eds.) IWSM 2009. LNCS, vol. 5891, pp. 101–121. Springer, Heidelberg (2009)
15. Marín, B., Giachetti, G., Pastor, Ó.: Measurement of Functional Size in Conceptual Models: A Survey of Measurement Procedures Based on COSMIC. In: Dumke, R.R., Braungarten, R., Büren, G., Abran, A., Cuadrado-Gallego, J.J. (eds.) IWSM 2008. LNCS, vol. 5338, pp. 170–183. Springer, Heidelberg (2008)

16. Mendling, J.: Validation of Metrics as Error Predictors. In: Metrics for Process Models. LNBIP, vol. 6, pp. 135–150. Springer, Heidelberg (2009)
17. Mendling, J.: Metrics for Business Process Models. In: Metrics for Process Models. LNBIP, vol. 6, pp. 103–133. Springer, Heidelberg (2009)
18. Monsalve, C., Abran, A., April, A.: Measuring Software Functional Size from Business Process Models. International Journal of Software Engineering and Knowledge Engineering 21(3), 311–338 (2011), doi:10.1142/S0218194011005359
19. OMG, OMG business process model and notation (BPMN), version 1.2 (Object Management Group) (2009)
20. Reynoso, L., Rolón, E., Genero, M., García, F., Ruiz, F., Piattini, M.: Formal Definition of Measures for BPMN Models. In: Abran, A., Braungarten, R., Dumke, R.R., Cuadrado-Gallego, J.J., Brunekreef, J. (eds.) IWSM 2009. LNCS, vol. 5891, pp. 285–306. Springer, Heidelberg (2009)
21. Vanderfeesten, I., Cardoso, J., Mendling, J., Reijers, H., van der Aalst, W.: Quality Metrics for Business Process Models. In: Fischer, L. (ed.) 2007 BPM & Workflow Handbook, Workflow Management Coalition, Lighthouse Point, Florida, USA, pp. 179–190 (2007)
22. Vanderfeesten, I., Reijers, H.A., Mendling, J., van der Aalst, W.M.P., Cardoso, J.: On a Quest for Good Process Models: The Cross-Connectivity Metric. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 480–494. Springer, Heidelberg (2008)
23. Mendling, J., Reijers, H., van der Aalst, W.: Seven Process Modeling Guidelines (7PMG). Information and Software Technology Journal 52, 127–136 (2010)
24. Conte, S., Dunsmore, H., Shen, V.: Software Engineering Metrics and Models. Benjamin-Cummings, Menlo Park (1986)
25. Frakes, W.B., Succi, G.: An industrial study of reuse, quality, and productivity. Journal of Systems and Software 57(2), 99–106 (2001), doi:10.1016/S0164-1212(00)00121-7, ISSN 0164-1212
26. Moseley, C.W.: A timescale estimating model for rule-based systems. Ph.D. diss., North Tex. State Univ. (1987)
27. Rainer, A., Hall, T.: Key success factors for implementing software process improvement: a maturity-based analysis. Journal of Systems and Software 62(2), 71–84 (2002), doi:10.1016/S0164-1212(01)00122-4, ISSN 0164-1212
28. Potok, T.E., Vouk, M., Rindos, A.: Productivity analysis of object-oriented software developed in a commercial environment. Software – Practice and Experience 29(10), 833–847 (1999)
29. McBride, T., Henderson-Sellers, B., Zowghi, D.: Software development as a design or a production project: An empirical study of project monitoring and control. Journal of Enterprise Information Management 20(1), 70–82 (2007)
30. Muketha, G.M., Ghani, A.A.A., Selamat, M.H., Atan, R.: A Survey of Business Process Complexity Metrics. Information Technology Journal 9(7), 1336–1344 (2010)