

Applying Software Metrics to evaluate Business Process Models

Elvira Rolón

Autonomous University of Tamaulipas-FIANS
Centro Universitario Tampico-Madero
89336 Tampico, Tamps. México
erolon@proyectos.inf-cr.uclm.es

Francisco Ruiz, Félix García, Mario Piattini

ALARCOS Research Group
Department of Information Systems and Technologies
UCLM-Soluziona Research and Development Institute
University of Castilla-La Mancha
Paseo de la Universidad, 4 – 13071 Ciudad Real, Spain
{francisco.ruizg, felix.garcia, mario.piattini}@uclm.es

Abstract

In this paper, we define a set of metrics for the evaluation of conceptual models of business processes. The proposal supposes the adaptation and extension of the FMESP framework (Framework for the Modeling and Evaluation of Software Processes). This adaptation can be carried out thanks to the similarities that exist between both types of processes (software and business). FMESP includes a set of metrics, which provide the quantitative basis necessary to find out the maintainability of the software process models. This proposal has been used as the starting point in proposing a set of metrics for the evaluation of the complexity of business process models defined by BPMN (Business Process Modeling Notation). Moreover, the groups of metrics of FMESP have been extended. This is because the models of business processes represented in BPMN include quite a number of aspects of interest in this domain which are not considered in software processes modelled with SPEM (Software Process Engineering Metamodel).

Keywords: Business Process, Software Process, Metrics, Conceptual Model, BPMN.

1. Introduction

Software processes and business processes present certain similarities. The most common one is that both of them try to capture the main characteristics of a group of partially-ordered activities which are carried out to achieve a specific goal. These objectives are those of obtaining a software product [1] or satisfactory result (generally a product or service) for the customer and other stakeholders of the process in their respective order [2].

As regards the modeling of both types of process, here we also find certain characteristics in common. When talking about the modeling of the software process, it should be pointed out that this refers to the definition of the processes as models, and in [3] it is defined as an abstract description of the activities by which the software is developed, focusing on models that are executable, interpretable or which are able to accede to automated reasoning. In addition to this specification, some of the specific goals and benefits of modeling the software process are defined in [4], such as: 1. Ease of understanding and communication, 2. Process management support and control, 3.- Provision for automated orientations for process performance, 4. Provision for automated execution support, and 5. Process improvement support.

On the other hand, business process models describe how a business works, or more specifically, how it carries out missions, activities, or tasks [5]. Besides all this, in business process modeling the main concept is the business process, which describes the activities involved in the business as well as how they relate to, and interact with, the necessary resources to achieve a target of the process.

Some specific goals of business process modeling are [6, 7]: 1. To facilitate the understanding of the key mechanisms of an existing business, 2. To serve as the basis for the creation of appropriate information systems supporting the business. 3. To improve the current business structure and operation, 4. To show the structure of an innovated business, 5. To identify outsourcing opportunities and, 6. To facilitate the alignment of business specifications with the technical framework that IT development needs.

A particular characteristic between business processes and software processes is the fact that, after more than one decade, we have witnessed the challenge of new technologies, along with more competitive markets that are, constantly-changing business environments and more demanding requirements for customer satisfaction. Consequently, software developers and managers, as well as business people and those involved in organizations in general, have regarded their processes as a focal reference point in the overall task of surviving and prospering [8]. This has raised the need to analyze, evaluate, measure and improve the processes and as a result, modeling of business processes in particular has been becoming increasingly popular in recent years.

In this work we tackle the issue of business processes from the perspective of the models which represent them at a conceptual level. To achieve this, we propose a set of metrics for evaluating business process models represented in BPMN. This proposal is based on the application and adaptation of the measurement framework of FMESP to business process models. That framework includes metrics for the evaluation of software process models defined by SPEM.

The rest of the paper proceeds as follows: In section 2, we describe the standard notations used for the modeling of both software and business processes. In section 3, the proposal of metrics for software process models of the framework FMESP is set out. Then the application and extension of FMESP for supporting the measurement of business process models are presented in sections 4 and 5 respectively. Section 6, provides an overview of the experimental plan to carry out a family of experiments to validate the measures proposed, and finally some conclusions and further work are outlined.

2. Conceptual Models of Software Process and Business Process

In this work, our target is to focus on the conceptual level of business process modeling, since we believe that it is one of the key points in obtaining models of quality that can serve as support for an effective maintainability and management of business processes. Wedemeijer defines the conceptual process model as an abstracted model of the business process, whose purpose is to outline all actions that are indispensable in producing all of the essential results in a customer-triggered business process. This is regardless of how, when, by whom, or by which means these outputs are produced [9].

Conceptual process models show what a system does or must do. They are independent of implementation and the language to perform is usually a graphic language. In the field of software process modeling the SPEM [10] specification is a generic metamodel used to describe a specific software development process or a family of related software development processes. It is structured in five packages: Basic Elements, Dependences, Process Structure, Process Components and Process Lifecycle.

SPEM is based on the UML metamodel and thus benefits from the expressiveness of that metamodel when representing descriptive software process models. The stereotypes of the SPEM profile and its notation are shown in Table 1.

Table 1. Stereotypes of the SPEM profile

Stereotype	Notation	Stereotype	Notation
WorkProduct		WorkDefinition	
ProcessPerformer		ProcessRole	
ProcessPackage		Phase	
Process		Document	
UMLModel		Activity	
Guidance		WorkDefinition	
ResizablePackage		MAS Model Element	

On the other hand, in the business process field there are many different proposals for the modeling of business processes. Nevertheless, all these proposals coincide on one point- the affirmation of the importance of having a graphic notation. Among the methodologies mentioned in the literature, the following deserve special attention in relation to the modeling of business processes: IDEF 0 [11], IDEF 3 [12], UML [7], UML 2.0 [13], and BPMN [14]. The latter is the notation standard on which our work of evaluation of business processes models is based.

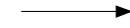
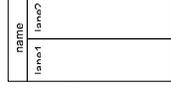
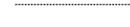
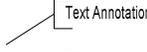
BPMN [14] is the new standard for modeling business processes and Web services processes, proposed by the Business Process Management Initiative (BPMI). BPMI tries to unify the diversity of proposals and terminology related to business process modeling by means of the standard notation BPMN, in the same way as the SPEM [10] specification does in the field of software process modeling.

BPMN provides a graphic notation for expressing business processes in a Business Process Diagram (BPD), based on a flowcharting technique, tailored to create graphic models of business process operations. This allows simple diagrams to be drawn up with ease. At the same time, it is able to handle the complexity inherent to business processes [15]. Another important characteristic of BPMN is that the XML languages designed for the execution of processes of business, such as BPEL and BPML, can be expressed visually with a common notation.

The first goal of BPMN is to provide a notation that can be easily understood by all business users, from business analysts to technical developers and business people [16]. To achieve this, BPMN facilitates the modeling of high-level business process through the BPD, which is composed of two basic categories: the first one composed of core elements with which is possible to develop simple process models. The second one contains a complete list of elements that allows the creation of complex or high-level business process models.

The four basic categories of elements are *Flow Objects*, *Connecting Objects*, *Swimlanes* and *Artifacts*. The symbols of the core elements are shown in Table 2. In addition, within each category of the core elements shown in Table 2 there is a more extensive list of business process constructors in the BPMN notation.

Table 2. Core Elements Set of BPD

BPD Core Element Set			
Flow Objects	Connecting Objects	Swimlanes	Artifacts
 <p>Events</p>	 <p>Sequence Flow</p>	 <p>Pool</p>	 <p>Data Objects</p>
 <p>Activities</p>	 <p>Message Flow</p>	 <p>Lane</p>	 <p>Groups</p>
 <p>Gateways</p>	 <p>Association</p>		 <p>Text Annotation</p> <p>Text Annotation</p>

3. Metrics for Conceptual Models of Software Process

As it has happened with business processes, software process research has taken on huge dimensions over recent years, due to the growing complexity of software systems. This is due to the processes' need to continuously undergo changes and refinements in order to increase their ability to deal with the requirements and expectations of the markets and the stakeholders of the company. Hence, the processes need to be continuously assessed and improved. This has led to a wide range of projects devoted to the creation of quality models and methods for the improvement of the software process [17].

Our work is based on the FMESP proposal [18, 19], which consists of a framework for the modeling and measurement of the software process. FMESP starts from the idea that a good administration of the software processes must be carried out. Its purpose will be to obtain software products which have a high level of quality. Such management considers the aspect of quality in an overall approach that includes two important aspects: process modeling and process evaluation. As a result, it provides the conceptual and technological support for the modeling and measurement of software processes, in the quest for their improvement.

For the evaluation of the software process, FMESP includes a set of metrics. This measures the structural complexity of software process models. The aim is to evaluate the influence of the structural complexity of the software process models on their maintainability. The FMESP metrics have been defined within two different scopes -first of all, model scope, to evaluate the overall structural complexity of the model and- secondly, core element scope, to evaluate the specific complexity of the fundamental elements of the model, namely *activities*, *roles* and *work products*. The model scope metrics are shown in the Table 3.

Table 3. Model Scope Metrics

Metric	Definition	Formula
NA	Number of Activities of the software process model	-
NWP	Number of Work Products of the software process model	-
NPR	Number of Roles which participate in the process	-
NDWPI _{in}	Number of input dependences of the Work Products with the Activities in the process	-
NDWPO _{out}	Number of output dependences of the Work Products with the Activities in the process	-
NDWP	Number of dependences between Work Products and Activities	$NDWP(PM) = NDWPI_{in}(MP) + NDWPO_{out}(MP)$
NDA	Number of precedence dependences between Activities	
NCA	Activity Coupling in the process model	$NCA(PM) = \frac{NA(PM)}{NDA(PM)}$

Metric	Definition	Formula
RDWPIIn	Ratio between input dependences of Work Products with Activities and total number of dependences of Work Products with Activities.	$RDWPIIn(PM) = \frac{NDWPIIn(PM)}{NDWP(PM)}$
RDWPOut	Ratio between output dependences of Work Products with Activities and total number of dependences of Work Products with Activities.	$RDWPOut(PM) = \frac{NDWPOut(PM)}{NDWP(PM)}$
RWPA	Ratio of Work Products and Activities. Average of the work products and the activities of the process model	$RWPA(PM) = \frac{NWP(PM)}{NA(PM)}$
RRPA	Ratio of Process Roles and Activities.	$RRPA(PM) = \frac{NPR(PM)}{NA(PM)}$

The FMESP metrics were defined by analyzing the SPEM metamodel [10] and they are grouped in: base metrics which were obtained by counting the number of significant SPEM metamodel constructors and their relationships and: derived metrics which are obtained as a result of applying measurement functions on another base and/or derived metrics. An example of a software process model represented with SPEM, with the calculation of the model scope metrics can be seen in Figure 1.

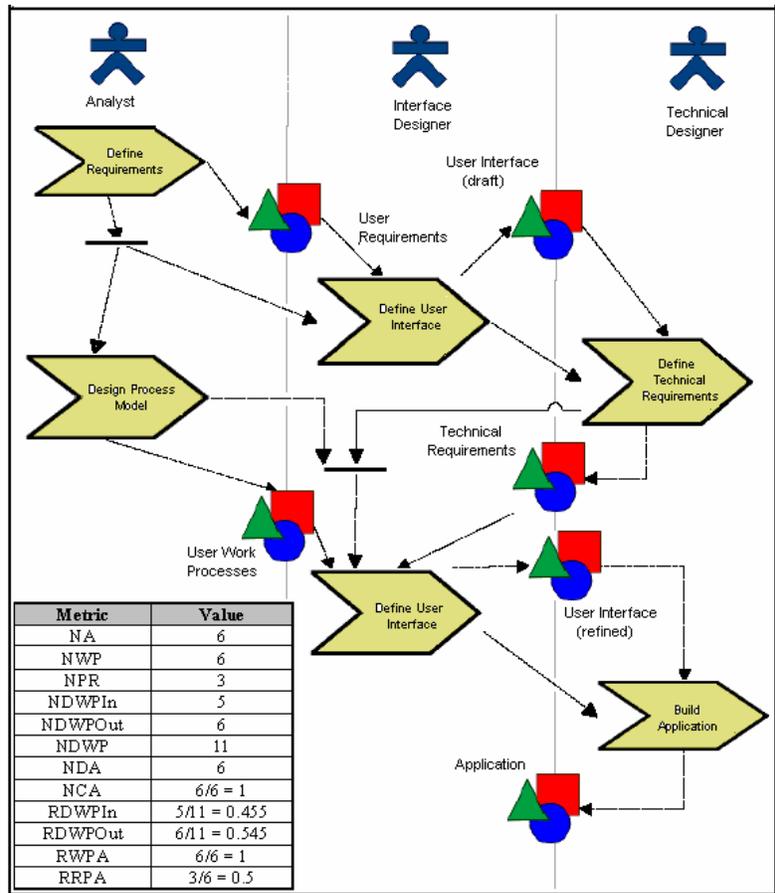


Figure 1. Software Process Model with SPEM and Metric Values

To establish which metrics can be used as SPMs maintainability indicators, a family of experiments was carried out. The metrics NA, NWP, NDWPIIn, NDWPOut, NDWP and NDA were validated as useful maintainability indicators as a result of these experiments [20]. The FMESP metrics defined for the evaluation of the complexity of specific elements in the software process model (activities, work products and process roles) are not described here, as they are outside the scope of this paper.

4. Applying the Proposal FMESP to models BPMN

With the definition and validation of the metrics in FMESP the objective is to determine a group of useful indicators of the maintainability of software process models, by evaluating their structural complexity. The proposal of FMESP started from the fact that the research on software process measurement had focused on the study of the results of the execution and not on the influence that the structural complexity of the processes models could have on its quality.

A similar situation happens in the area of business processes modeling. Looking at research done from the perspective of business people, in the relevant literature we can find a variety of proposals for the evaluation of processes. These are mostly from the point of view of the results obtained in the execution of these processes. So the aspects evaluated in research on business process measurement belong mainly to the level of process execution. Here there are two categories of metrics that are given consideration: operational and structural [21]. There are, nevertheless, also proposals or frameworks whose aim is to evaluate the quality of techniques of business processes modeling [22].

Given our interest in evaluating the business process by starting from the model that represents it at a conceptual level, our work recaptures the FMESP proposal, but adapts and extends it to business process models. To achieve this, we have defined a set of metrics for evaluating the structural complexity of business process models at a conceptual level. The goal is to have empirical evidence about the influence that the structural complexity of business models can have on their maintainability. It can provide companies with the quantitative basis necessary for developing more maintainable business process models.

The first step in achieving this goal is to define a set of suitable metrics for the evaluation of the structural complexity of business models. This definition has been based on the elements that the BPMN metamodel is made up of. These metrics have been grouped into two main categories: Base and Derived Measures. The base measures have been defined by counting the different kind of elements that a business process model is composed of represented with BPMN. In Table 4, the base measures defined for the constructor “Event” in the BPMN metamodel are shown.

Table 4. Base Measures for the element Event in BPD Object Flow

Core Element	Notation	Metric Name	Base Measure
Start Event	 Start	NSNE	Number of Start None Events
	 Timer	NSTE	Number of Start Timer Events
	 Message	NSMsE	Number of Start Message Events
	 Rule	NSRE	Number of Start Rule Events
	 Link	NSLE	Number of Start Link Events
	 Multiple	NSMuE	Number of Start Multiple Events
Intermediate Event	 Intermediate	NINE	Number of Intermediate None Events
	 Timer	NITE	Number of Intermediate Timer Events
	 Message	NIMsE	Number of Intermediate Message Events
	 Error	NIEE	Number of Intermediate Error Events
	 Cancel	NICaE	Number of Intermediate Cancel Events

Core Element	Notation	Metric Name	Base Measure
	 Compensation	NICoE	Number of Intermediate Compensation Events
	 Rule	NIRE	Number of Intermediate Rule Events
	 Link	NILE	Number of Intermediate Link Events
	 Multiple	NIMuE	Number of Intermediate Multiple Events
End Event	 End	NENE	Number of End None Events
	 Message	NEMsE	Number of End Message Events
	 Error	NEEE	Number of End Error Events
	 Cancel	NECaE	Number of End Cancel Events
	 Compensation	NECoE	Number of End Compensation Events
	 Link	NELE	Number of End Link Events
	 Multiple	NEMuE	Number of End Multiple Events
	 Terminate	NETE	Number of End Terminate Events

As we can observe in Table 4, the base measures defined for all the triggers of events are included (Start, Intermediate and End). They belong to the BPD “Flow Objects” category. With these, the cause of the beginning or ending of a flow within the model can be identified, as well as those elements that modify the flow at an intermediate point of the same. The base measures shown in Table 4 were determined according to each of the different types of events that may be contained in a Business Process Diagram. Next, in Table 5 below, the base measures for the BPMN metamodel element “activity” are shown.

Table 5. Base Measures for the element Activity of the BPD Flow Objects.

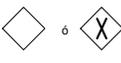
Core Element	Notation	Metric Name	Base Measure	Definition
Task	 Task	NT	Number of Task	Indicates the total number of tasks in the model
	 Looping	NTL	Number of Task Looping	Indicates the total number of task looping in the model
	 Multiple Instances	NTMI	Number of Task Multiple Instances	Indicates the total number of task multiple instances in the model
	 Compensation	NTC	Number of Task Compensation	Indicates the total number of task compensation in the model

Core Element	Notation	Metric Name	Base Measure	Definition
Collapsed Sub-Process	 Collapsed Sub-Process	NCS	Number of Collapsed Sub-Process	Indicates the total number of Collapsed Sub-Processes in the model
	 Loop	NCSL	Number of Collapsed Sub-Process Looping	Indicates the total number of Collapsed Sub-Process Looping in the model
	 Multiple Instance	NCSMI	Number of Collapsed Sub-Process Multiple Instance	Indicates the total number of Collapsed Sub-Process Multiple Instance in the model
	 Compensation	NCSC	Number of Collapsed Sub-Process Compensation	Indicates the total number of Collapsed Sub-Process Compensation in the model
	 Ad-Hoc	NCSA	Number of Collapsed Sub-Process Ad-Hoc	Indicates the total number of Collapsed Sub-Process Ad-Hoc in the model

Within Flow Objects, the activity element of the BPD can be made up of atomic activities (tasks) and of compound activities (collapsed sub-processes) and within each category different classes can be observed, as is shown in the previous table where a metric for each one of the four types of tasks and for the five types of sub-process is defined.

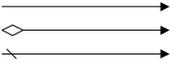
In the same category of “Flow Objects”, the “Gateways” are the elements used to control the divergence and convergence of Sequence Flow. In the BPD, there are five types of Gateways, and we have defined metrics for each type (Table 6).

Table 6. Base Measures for the Gateway Control Types in the BPD Flow Objects.

Core Element	Notation	Metric Name	Base Measure	Definition
Exclusive Decision Data-Based XOR Decision		NEDDB	Number of Exclusive Decision/Merge Data-Based	Indicates the number of points of exclusive decision and merging based on data of the model
Exclusive Decision Data-Event XOR Decision		NEDEB	Number of Exclusive Decision/Merge Event-Based	Indicates the number of points of exclusive decision and merging based on events of the model
Inclusive (OR)		NID	Number of Inclusive Decision/Merge	Indicates the number of points of inclusive decision and merging of the model
Complex		NCD	Number of Complex Decision/Merge	Indicates the number of points of complex decision merging of the model
Paralell (AND)		NPF	Number of Parallel Fork/Join	Indicates the number of points of parallel forking and joining of the process

With these metrics, it is possible to know the number of Gateways that generate forks or joins of sequence flow at a specific point in the process. Other important elements to consider within of the BPD core elements are displayed in Table 7 with their respective base measures.

Table 7. Base Measure for the Connecting Objects, Swimlanes and Artifacts

Core Element	Notation	Metric Name	Base Measure
Sequence Flow		NSFA	Number of Sequence Flows between Activities
		NSFE	Number of Sequence Flows incoming from Events
		NSFG	Number of Sequence Flows incoming from Gateways
		NSFL	Number of Sequence Flows Looping
Message Flow		NMF	Number of Message Flows between Participants in the Process
Pool		NP	Number of Pools in the Process
Lanes		NL	Number of Lanes in the Process
Data Objects (Input)		NDOIn	Number of Data Objects-In of the Process
Data Objects (Output)		NDOOut	Number of Data Objects-Out of the Process

Given the definition of these base measures, it is possible to discover how many significant elements there are in the business process diagram. Based on the base measures defined, the proposal of metrics for business process models includes some significant derived measures, obtained by means of the measurement function, which establishes the existing proportions among different elements of the model. The derived measures for business processes models with BPMN are seen in Table 8.

Table 8. Derived Measure of elements common to the category of Flow Objects

Name	Formula	Metric
TNSE	$TNSE = NSNE + NSTE + NSMsE + NSRE + NSLE + NSMuE$	Total Number of Start Events of the Model
TNIE	$TNIE = NINE + NITE + NIMsE + NIEE + NICaE + NICoE + NIRE + NILE + NIMuE$	Total Number of Intermediate Events of the Model
TNEE	$TNEE = NENE + NEMsE + NEEE + NECaE + NECoE + NELE + NEMuE + NETE$	Total Number of End Events of the Model
TNE	$TNE = TNSE + TNIE + TNEE$	Total Number of Events of the Model
TNT	$TNT = NT + NTL + NTMI + NTC$	Total Number of Task of the Model
TNCS	$TNCS = NCS + NCSL + NCSMI + NCSC + NCSA$	Total Number of Collapsed Sub-Process of the Model
TNA	$TNA = TNT + TNCS$	Total Number of Activities of the Model
TNG	$TNG = NEDDB + NEDEB + NID + NCD + NPF$	Total Number of Gateways of the Model
TNDO	$TNDO = NDOIn + NDOOut$	Total Number of Data Objects in the Process Model
CLA	$CLA = \frac{TNA}{NSFA}$	Connectivity level between Activities

Name	Formula	Metric
CLP	$CLP = \frac{NMF}{NP}$	Connectivity Level Between Pools
PDOPIIn	$PDOPIIn = \frac{NDOIn}{TNDO}$	Proportion of Data Objects, as Incoming Product and the total of Data Objects
PDOPOut	$PDOPOut = \frac{NDOOut}{TNDO}$	Proportion of Data Objects, as Outgoing Product and the total of Data Objects
PDOTOut	$PDOTOut = \frac{NDOOut}{TNT}$	Proportion of Data Objects, as Outgoing Product of Activities of the Model
PLT	$PLT = \frac{NL}{TNT}$	Proportion of Pools and/or Lanes of the Process and Activities in the Model

With the base and derived measures proposed, it is possible to evaluate the structural complexity of business process models expressed in BPMN. When analyzing the model structurally, the quality of the model can also be assessed. In particular, this is done with reference to the three quality criteria for conceptual models given by Lindland: semantic quality, syntactic quality and pragmatic quality [23].

In order to obtain reliable results and to discover whether the measures proposed for the evaluation of business process models are useful in real cases, it is necessary to carry out an experimental validation. To this effect, and basing our proposal on the work presented in [20], we have planned a set of experiments with the aim of evaluating the quality aspects of conceptual business process models expressed with BPMN.

5. Extension of FMESP

In the previous sections, we have described two proposals of metrics for evaluating software process models and business process models respectively. These metrics have been defined on two different metamodels, namely SPEM for software processes and BPMN for business process models. It is important to highlight that SPEM is a generic metamodel, and the measures proposed can be applied to other process modeling languages, even to those not specific to software, such as BPMN. On the other hand, as BPMN focuses specifically on business processes, it presents some aspects that are not contemplated for software processes and this means that new specific metrics are needed.

If we take into account the issues mentioned above, we see that in measuring BPMN business process models the metrics of the framework FMESP for SPEM have been successfully applied. New metrics (not defined in FMESP) have been necessary, however, due to the specific notation of BPMN, so as to be able to model some particular aspects of business processes. Table 9 shows the modeling elements considered in SPEM and BPMN notations.

Table 9. Constructor of SPEM and BPMN for definition of metrics

Element	SPEM (FMESP)	BPMN
Events		✓
Activities	✓	✓
Gateways		✓
Work Products (Data Objects)	✓	✓
Roles (Lanes)	✓	✓
Dependencies (Sequence Flow)	✓	✓
Message Flow		✓
Pools		✓

As we can observe in Table 9, there are some elements in BPMN that are useful for the modeling of business process and which SPEM does not contemplate. These include such components as Events, Gateways, Message Flow and Pools, for which it has been necessary to define new base measures for

these elements in particular. Additionally, a new group of derived measures has been defined, which are not included in FMESP. These come from the new base measures defined for business processes. These new base measures and derived measures are set out in Table 10.

Table 10. New Based Measure and Derived Measure based on BPMN

Element	Category	New Base Measure	New Derived Measure
Events	Start	NSNE, NSTE, NSMsE, NSRE, NSLE, NSMuE	TNSE, TNIE, TNEE, TNE
	Intermediate	NINE, NITE, NIMsE, NIEE, NICaE, NICoE, NIRE, NILE, NIMuE	
	End	NENE, NEMsE, NEEE, NECaE, NECoE, NELE, NEMuE, NETE	
Activities	Tasks	NT, NTL, NTMI, NTC	TNT, TNCS, TNA
	Collapsed sub-process	NCS, CSL, NCSMI, NCSC, NCSA	
Gateways		NEDDB, NEDEB, NID, NCD, NPF	TNG
Sequence Flow		NSFA, NSFE, NSFG, NSFL	
Message Flow		NMF	CLP
Pools		NP	PLT
Data Objects			PDOPIIn, PDOPOut, PDOTOOut

Note that although the activities are seen as possible in both proposals, here they are included as an extension of FMESP. This is because in BPMN, as we have already seen, atomic and compound activities can be observed. These can, in turn, have different characteristics or properties.

With all the metrics defined, both the base and derived ones, we believe that we could obtain information about the structural complexity of the model of business processes. This would allow us to evaluate aspects such as their understandability, coherence, completeness, modifiability and consistency, thus assuring the quality of the model at a conceptual level [23].

In the following section, an example of a business process model using BPMN is presented. In this the metrics, as defined in FMESP for software process models, are applied, as well as the metrics that we have defined for business process models.

5.1 Application Example

To illustrate the calculation of the metrics defined for business process models, an example which has been taken from [24] is provided. The example (Figure 2) represents a concurrent engineering model with message intermediate events, and our objective is to apply the metrics defined in this work so as to find out the structural characteristics of this model.

The values of the metrics defined in FMESP and the set of metrics defined according to BPMN that have been applied in the above model are shown in the Table 11. Due to a lack of space, in the case of the metrics for business processes, only the derived measures will be displayed.

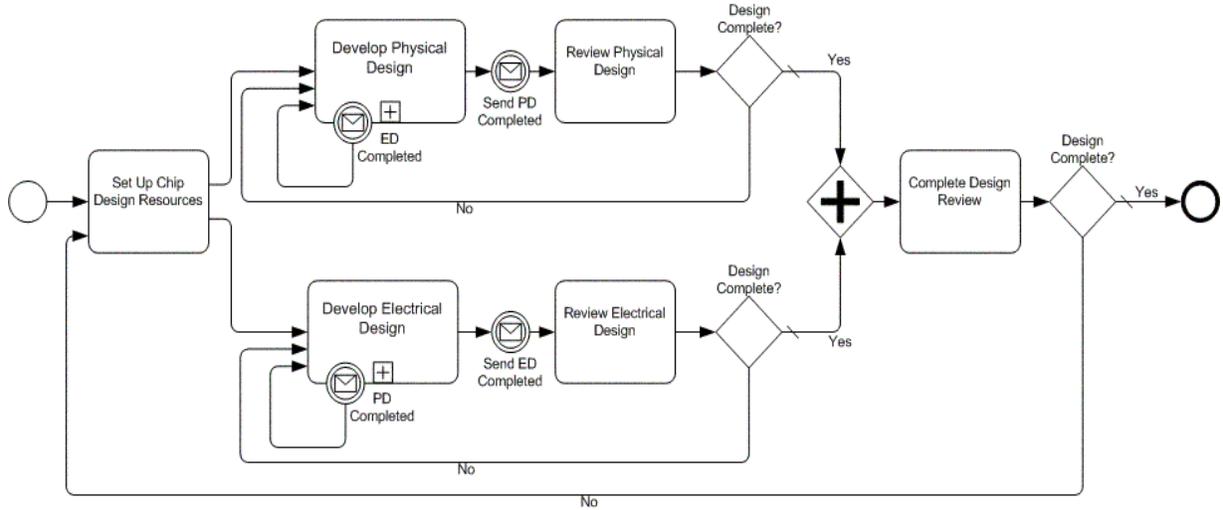


Figure 2. Model concurrent of Engineering with BPMN

Table 11. Value of Metrics defined in FMESP and Derived Measure with BPMN

Metrics of FMESP	
Metric	Value
NA	6
NWP	0
NPR	1
NDWPIIn	0
NDWPOut	0
NDWP	0
NDA	2
NCA	3
RDWPIIn	0
RDWPOut	0
RWPA	0
RRPA	0

Derived Measure with BPMN	
Metric	Value
TNSE	1
TNIE	4
TNEE	1
TNE	6
TNT	2
TNCS	4
TNA	6
TNG	4
TNDO	0
CLA	3
CLP	0
PDOPIIn	0
PDOPOut	0
PDOTOut	0
PLT	0

As can be appreciated by looking at the previous tables, practically there is no difference between the defined values of the metrics for the two types of processes (software and business). The difference that one can observe is in those metrics based on elements that cannot be modeled with SPEM, but which at the same time can be useful in analyzing the business processes models structurally.

Hence we demonstrate that, although in the pertinent literature there are currently no proposals of metrics for the evaluation of business process models at a conceptual level, it is possible to carry out an evaluation of them by applying metrics defined for software process models and by defining new ones specifically for business process models.

With the definition of the proposed metrics, an empiric evidence on the influence that the structural complexity of the models of a business process hold in its maintainability and use is intended to be obtained. All of this based on the idea of the improvement of processes throughout the analysis of such characteristics, which are part of the product's quality features in accordance with the ISO 9126 standard [25].

6. Experimental Plan.

As our objective is centred on the evaluation of business process models (BPMs), the plan for a family of experiments has been developed, using the model which represents them as a starting point, and with the end of validating the proposed measures. In the following section we describe the general plan of the family of experiments.

6.1 Participants

The subjects should have some knowledge of software modelling (UML, databases, etc.) and ideally they should also be familiarized with business process modelling. An introductory lecture about BPM modelling and the BPMN metamodel will be given and a training session will be developed in order to provide the subjects with the necessary knowledge to carry out the tasks required in the experiment. However, the subjects will not be aware of what aspects we intend to study.

6.2 Material

The material prepared consists of ten BPMs represented with BPMN, which have different structural characteristics and dimensions from each other, which is to say that models with different degrees of complexity were selected. Moreover, two questionnaires were formulated for each of the aforementioned models, the first of which consisted of a series of questions related to the model's understandability, and the second of which proposed a series of modifications to be carried out in the model such as evaluating the complexity of the process models presented.

In this way it was possible to prepare material in two groups (X and Y): group X consists of ten BPMs, of which five models have a questionnaire relating to the model's understandability and the other five have a questionnaire relating to the model's modifiability. Group Y is made up of the same ten models as the first set, but with the questionnaires the other way round, which is to say the first five models now correspond to a questionnaire relating to the model's modifiability, and the remaining five to the model's understandability.

6.3 Experimental Task

Each subject will receive material composed of ten BPMs (five with understandability questions and five with modification requests). Depending on the model (group X or Y) the subjects have to carry out one of the following tasks: to answer "yes" or "no" to six questions about the model or to carry out four modifications consisting of adding and/or deleting tasks, data objects, events, roles or dependences among these elements.

Each type of task (understandability or modifiability) to be developed has to be similar in its complexity. For this reason, the only source of variation in effort to perform tasks of the same type is the complexity of each model. Before starting to perform the tasks required in each model the subjects are required to write down the starting time, and at the end they have to write down the finishing time. Finally they have to subjectively rate their opinion of the overall complexity of the model.

The family of experiments is being developed with an integrated population by experts in business analysis and software engineering, and this will allow us to compare the results of both types of profiles and determine the influence of these varying points of view.

7. Conclusions and Further Work.

In this paper, we have put forward the proposal that FMESP can be applied in evaluating business process models at a conceptual level and illustrated how that can be done. The metrics in FMESP were defined following the SPEM terminology and many of them can be applied as useful maintainability indicators. These metrics make it possible to determine the structural complexity of software process models.

Taking into consideration that in the field of process engineering there are no metrics applicable to business process models at a conceptual level, we make use of the philosophy of FMESP in the evaluation of the structural complexity of business process models. We have taken as our starting point a definition of base measures and derived measures following the BPMN terminology. In this work it has been proved that metrics for software process models can be applied to business process models, since they present certain similarities as regards the core elements that both are made up of. However, we have had to extend the metrics defined in FMESP to embrace all the aspects considered within a business process model.

By integrating both proposals, we provide a more refined framework for evaluating business process models. This gives support to Business Process Management, which has as one of its stages the definition and modeling of the process being assessed. It will allow a more appropriate management of the business processes and can provide organizations with important profits. Model metrics can be very useful when selecting the models which are most easily maintained, from among various alternatives in companies who may change their models to improve their business processes. Besides, it can help to facilitate the evolution of business processes in these companies by assessing the process improvement at a conceptual level.

The metrics for business process models provide companies with objective information about the maintainability of these models. More maintainable models can benefit the management of the business processes in two main ways: i) guaranteeing the understanding and the diffusion of the processes as they evolve, without affecting their successful execution; ii) reducing the effort needed to change the models, with a consequent reduction of maintenance.

At present, we are developing a family of experiments whose purpose will be to evaluate quality aspects of conceptual business process models. These experiments will be carried out in a group of people made up of experts in business analysis and in software engineering. We will thereby provide a comparison between results from both kinds of stakeholders, determining the influence from these different viewpoints as a result.

Acknowledgments

This work has been partially financed by the ENIGMAS Project (Junta de Comunidades de Castilla-La Mancha, Consejería de Educación y Ciencia, reference PBI-05-058).

References.

- [1] S.T. Acuña and X. Ferré. "Software Process Modelling". In Proceedings of the 5th. World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001). Orlando Florida, USA. pp. 1-6, 2001.
- [2] A. Sharp and P. McDermott, "Workflow Modeling : Tools for Process Improvement and Application Development". London: Artech House (Pub). 2000
- [3] A. Finkelstein, J. Kramer, and M. Hales, "Process Modelling: a Critical Analysis", in Integrated Software Reuse: Management and Techniques, P. Walton and N. Maiden, Editors. Chapman and Hall and UNICOM. pp. 137-148. 1992.

- [4] B. Curtis, M.I. Kellner, and J. Over, "Process Modeling". Communications of the ACM, Vol. 35 (No. 9): pp. 75-90. 1992
- [5] T. Dufresne and J. Martin, "Process Modeling for E-Business", George Mason University, Spring 2003, INFS 770 - Methods for Informations Systems Engineering: Knowledge Management and E-Business., 2003.
- [6] K. Beck, J. Joseph, and G. Goldszmidt, Learn Business Process Modeling Basics for the Analyst. IBM, July 2005, www-128.ibm.com/developersworks/library/ws-bpm4analyst
- [7] H.-E. Erickson and M. Penker, "Business Modeling with UML- Business Patterns at Work", ed. I. John Wiley & Sons. USA: Robert Ipsen. 2000
- [8] W.A. Florac, R.E. Park, and A.D. Carleton, "Practical Software Measurement: Measuring for Process Management and Improvement", Guidebook. Carnegie Mellon University, April 1997, CMU/SEI-97-HB-003, 1997.
- [9] d.L. Wedemeijer and d.i.E. de Bruin. "Conceptual Process Models: Using Process Architecture in Practice". In Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA '04). Zaragoza, Spain: IEEE Computer Society. pp. 2004.
- [10] OMG, "Software Process Engineering Metamodel Specification", adopted specification, version 1.0. Object Management Group, Inc., November, 2002.
- [11] FIPS, "Integration Definition for Function Modeling (IDEF0)", Standard. National Institute of Standards and Technology, December, 1993.
- [12] R.J. Mayer, C.P. Menzel, M.K. Painter, P.S. de White, T. Blinn, and B. Perakath, "Information Integration for Concurrent Engineering (IICE) IDEF3 Process Description Capture Method Report", Interim Technical Report. September, 1995.
- [13] OMG, "Unified Modeling Language (UML) Specification: Infrastructure, version 2.0", Object Management Group, December, 2003.
- [14] BPMI, "Business Process Modeling Notation", Specification Version 1.0. Business Process Management Initiative, May 3, 2004. available on www.bpmi.org
- [15] M. Owen and J. Raj, "BPMN and Business Process Management. Introduction to the New Business Process Modeling Standard", White Paper. Popkin Software, September, 2003. <http://www.popkin.com>
- [16] S.A. White, Introduction to BPMN. bpmn.org (pub), May 2004 www.bpmn.org
- [17] A. Fuggetta. "Software Process: A Roadmap". In Proceedings of the 22th. International Conference on Software Engineering (ICSE-2000). Limerick, Ireland. pp. 25-34, 2000.
- [18] F. García, F. Ruiz, M. Piattini, G. Canfora, and C.A. Visaggio, "Framework for the Modeling and Evaluation of Software Processes". Journal of Systems Architecture, (accepted to appear). 2006
- [19] F.O. García, "FMESP. Marco de Trabajo Integrado para el Modelado y la Medición de los Procesos Software", in Departamento de Informática, Escuela Superior de Informática. Universidad de Castilla La Mancha: Ciudad Real. 2004.
- [20] G. Canfora, F. García, M. Piattini, F. Ruiz, and C.A. Visaggio, "A Family of Experiments to Validate Metrics for Software Process Models". Journal of Systems and Software, 77 (2): pp. 113-129. 2005
- [21] G.S. Tjaden, "Business Process Structural Analysis", Georgia Tech Center for Enterprise Systems, October, 1999.
- [22] B.-J. Hommes and V. van Reijswoud. "Assessing the Quality of Business Process Modelling Techniques". In Proceedings of the 33rd Hawaii International Conference on Systems Sciences (HICSS 2000). Maui, Hawaii, USA: IEEE. pp. 1007-1016, 2000.
- [23] O.I. Lindland, G. Sindre, and A. Solvner, "Understanding Quality in Conceptual Modeling". Software IEEE, Vol. II (Issue 2): pp. 42-49. 1994
- [24] BPMN, Working Group. Business Process Management Initiative, www.bpmn.org/exampleIndex.htm
- [25] ISO/IEC, "FCD9126-1.2, Information Technology - Software product quality - Part 1: Quality model", 1998.