Information and Software Technology 56 (2014) 134–162

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

An integrated approach based on execution measures for the continuous improvement of business processes realized by services

Andrea Delgado [a,*], Barbara Weber [b], Francisco Ruiz [c], Ignacio Garcia-Rodríguez de Guzmán [c], Mario Piattini [c]

[a] Computer Science Institute, Faculty of Engineering, University of the Republica, Julio Herrera y Reissig 565, 11300 Montevideo, Uruguay
[b] Quality Engineering Research Group, Computer Science Institute, University of Innsbruck, Technikerstraße 21a, 6020 Innsbruck, Austria
[c] Institute of Information Technologies and Systems, University of Castilla – La Mancha, Camino de Moledores s/n, 13051 Ciudad Real, Spain

ARTICLE INFO

Article history:
Received 26 June 2012
Received in revised form 12 August 2013
Accepted 27 August 2013
Available online 4 September 2013

Keywords:
Business Process Management (BPM)
Service Oriented Computing (SOC)
Continuous Process Improvement (CPI)
Business Process Intelligence (BPI)
Business Process execution measurement
ProM framework

ABSTRACT

Context: Organizations are rapidly adopting Business Process Management (BPM) as they focus on their business processes (BPs), seeing them to be key elements in controlling and improving the way they perform their business. Business Process Intelligence (BPI) takes as its focus the collection and analysis of information from the execution of BPs for the support of decision making, based on the discovery of improvement opportunities. Realizing BPs by services introduces an intermediate service layer that enables us to separate the specification of BPs in terms of models from the technologies implementing them, thus improving their modifiability by decoupling the model from its implementation.
Objective: To provide an approach for the continuous improvement of BPs, based on their realization with services and execution measurement. It comprises an improvement process to integrate the improvements into the BPs and services, an execution measurement model defining and categorizing several measures for BPs and service execution, and tool support for both.
Method: We carried out a systematic literature review, to collect existing proposals related to our research work. Then, in close collaboration with business experts from the Hospital General de Ciudad Real (HGCR), Spain, and following design science principles, we developed the methods and artifacts described in this paper, which were validated by means of a case study.
Results: We defined an improvement process extending the BP lifecycle with measurement and improvement activities, integrating an execution measurement model comprising a set of execution measures. Moreover, we developed a plug-in for the ProM framework to visualize the measurement results as a proof-of-concept prototype. The case study with the HGCR has shown its feasibility.
Conclusions: Our improvement vision, based on BPs realized by services and on measurement of their execution, in conjunction with a systematic approach to integrate the detected improvements, provides useful guidance to organizations.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The complexity and size of the current systems to support the business of organizations has grown considerably in recent years, thus increasing the difficulty in managing them properly. Business Process Management (BPM) [1–3] is becoming increasingly important for those organizations which need to gain a better insight into the way their business processes (BPs) are executed. BPM helps organizations to manage their BPs, assisting them in checking that their outputs are maintained in the range defined as successful with respect to the business goals of the organization. BPM provides the means for guiding and supporting the modeling, implementation, deployment, execution, and evaluation of BPs in an organization, based on the BP lifecycle [1], which establishes the main phases and activities that organizations have to carry out in their efforts to manage BPs: Design & Analysis, Configuration, Enactment and Evaluation of BPs. In the Design & Analysis phase, BPs are first identified and modeled and then validated and verified; in the Configuration phase BPs are implemented in the chosen technology, and then tested and deployed in the selected platform. The Enactment phase involves the execution and monitoring of BP instances and the registration of execution data in execution logs. Finally, in the Evaluation phase execution log data is evaluated to provide insight into the real execution of the BPs using, for example, process mining techniques. The boom in

* Corresponding author. Tel.: +598 27114244; fax: +598 2711 0469.
E-mail addresses: adelgado@fing.edu.uy (A. Delgado), barbara.weber@uibk.ac.at (B. Weber), francisco.ruizg@uclm.es (F. Ruiz), ignacio.grodriguez@uclm.es (I. Garcia-Rodríguez de Guzmán), mario.piattini@uclm.es (M. Piattini).

0950-5849/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.infsof.2013.08.003

the use of BPM to support the BP lifecycle introduces both a fresh approach to business management and new challenges for the undertaking of management efforts. One of these challenges is that without proper guidance for applying BPM, the results are barely predictable, with the drawbacks that this entails.

According to the Gartner Group [4] "organizations carry out BPM projects in order to improve one or more business processes". This places BP improvement as the number one motivation for BPM. In the same survey, the top five business goals identified include: improving customer satisfaction, improving the quality of BPs, reducing costs, improving BP agility, and supporting continuous process improvement. Measurement activities are implicit in the lifecycle, but to guide the integration of execution measurement through the whole lifecycle we must define measurement activities explicitly. This will help obtain insight into the real execution of BPs, knowledge that is needed for BP improvement.

The implementation of BPs by means of different systems in the organization also affects the way business people can perform the defined activities and how they perceive the software support provided. The traditional vertical vision of Information Technologies (IT) for implementing BPs, based on sections or areas of the organization, made the modification of BPs a challenging activity that required the integration of various heterogeneous information systems. These changes are not easy to introduce, in general, due to the implicit implementation of the BPs in the systems supporting them. Among other disadvantages, this often leads to more time being spent than initially planned. In addition, it may result into high costs and unfulfilled expectations of the business area regarding the functionality offered by the implemented BPs [5,6]. Service Oriented Computing (SOC) [7] provides the basis for defining services that can implement parts of BPs (activities, sub-processes) or even a BP as a whole, by introducing an intermediate layer of services between BP definition and their implementation by means of different technologies. This approach helps bridge the so-called business-systems gap caused by different views and expectations between the business and the IT side when introducing changes into BPs. The service layer enables us to separate the specification of BPs in terms of models from the technologies implementing them, thus improving their modifiability by decoupling the model from the implementation in the technology selected [5,6]. The approach provides the basis for integrating changes with minimum impact, both at the BPs level and at the system level, allowing for the organizational agility needed to respond to new demands or corrective measures [5,6].

In this article we present an approach for the improvement of BPs which extends the traditional BP lifecycle [1] with measurement and improvement activities. We have called this BP Continuous Improvement Process (BPCIP). Our proposal includes a model that integrates execution measures for BPs with services implementing them in a comprehensive way; we have called this model BP Execution Measurement Model (BPEMM). In previous work we presented an initial definition of the BPCIP [8], which we have extended significantly by redefining the complete BPCIP lifecycle based on the feedback from the conference and by completing the set of execution measures to be integrated into the BPEMM.

Both the BPCIP and the BPEMM are part of a larger framework called MINERVA [9,10], which we defined with the aim of providing an integrated approach to support the continuous improvement effort in an organization based on the realization of BPs by services with a model driven approach. Our purpose was also to enable BP execution measurement, as far as both BPs and services are concerned. The framework is organized in three dimensions: conceptual, methodological and tool support. Previous contributions described the MINERVA framework as a whole [9,10], the conceptual [11] and the tool [12] dimensions, as well as part of the methodological dimension: the BPSOM methodology and the model-driven approach [13,14]. We will not deal with these aspects in detail here. Contributions of this paper concern the support for measurement and improvement of BPs that are part of the methodological dimension of MINERVA. What this article contributes therefore, is as follows:

- The redefined BPCIP lifecycle, based on the BP lifecycle [1] and extended with measurement and improvement activities, as well as the elements to carry out the defined activities, such as roles and input and output artifacts.
- The complete set of integrated execution measures defined in the execution measurement model BPEMM, along with the addition of a "cube" presentation for the tridimensional taxonomy we have defined to organize the measures.
- A proof-of-concept prototype of the ProM BPEMM plug-in, to support the visualization of the results of the execution measurement of measures defined within the BPEMM, as well as to demonstrate the feasibility of our definitions.
- The validation of our approach by means of a case study undertaken in the Hospital General de Ciudad Real (HGCR), with the help of business experts.

1.1. Background

> *"Measurement is the first step that leads to control and eventually to improvement. If you can't measure something, you can't understand it. If you can't understand it, you can't control it. If you can't control it, you can't improve it." H. James Harrington (Harrington, 1991) [15].*

Continuous process improvement refers to a status in which the organization is continuously analyzing the way it carries out its business. Its goal is to find improvement opportunities for the performance of the organization's BPs [16,17]. Measurement of BP execution provides the basis for analyzing the real behaviour in the organization. It helps to detect deviations from planned behaviour, as well as to discover improvement opportunities for the BPs. Once improvement opportunities are detected, organizations need to define changes in the BPs that will lead to a new improved version of these for better achieving the business goal set by the organization. Execution measurement then becomes the enabler towards understanding and controlling the real occurrences of BPs in the organization, establishing an ongoing BP improvement culture [16]. To improve processes continuously, several aspects have to be taken into account. A key one refers to the creation of an organizational improvement context, which comprises a number of elements including business and software teams committed to the improvement initiative and a systematic approach for conducting improvement efforts, the explicit specification of BP models and the software implementing them, a definition of measures collected during BP execution, or techniques and tools to enable the evaluation of the collected execution logs. One key statement of process management is that quality of products and services is largely determined by the quality of the processes used to develop, deliver and support them. An effective process is capable of bringing people, tools and methods together into an integrated whole which produces the expected outcomes [16].

An improvement approach has to support the identification of process deficiencies and provide guidance for introducing improvements in a systematic way. To carry this out, measures of the BP, activities, performance, resources, cost and results have to be defined, implemented, collected and analyzed on a regular basis [16]. Organizations in different domains such as software, manufacturing, marketing, banking, and finance share similar problems. These could include, for instance, overworked staff, thanks to poor estimating and planning, or excessive rework. There may also be a lack of consistent and stable processes, often with multiple ways to do similar things, or no sound basis for measurement and management. A foundation for organization-wide approaches and solutions may be lacking, and disappointing results from automation might exist. Other problems may arise, such as mixed results when applying approaches like Six Sigma or Business Process Reengineering (BPR), or improvements that are too localized and sub-optimal from a global business perspective [16].To meet these challenges, an improvement approach has to provide specific artifacts, along with a systematic way to support and guide the improvement effort in the organization [16]. It is not enough to provide measures and the means to analyze them, including tool support; it is essential to align the measures that are related to business strategy and goals for the entire organization with the ones that are specific for each BP. In a mature organization there is an objective quantitative basis for analyzing problems occurring in BP operation. Therefore, when changes are needed to address these problems the different options are understood as are the overall effects and consequences of choosing a particular option. Historical documented data is available and estimates and plans are based on this data, helping to achieve the expected results for aspects that include cost, schedule, performance and quality. A systematic improvement effort helps organizations move along the path from an immature status to a mature one [16,17].

Business Process Intelligence (BPI) [18,19] focuses on the collection and analysis of information from BPs to support decision making, and includes terms such as Business Activity Monitoring (BAM) and (Business) Process Mining (PM). BAM refers to the real-time monitoring of BP execution, focusing mainly on showing performance indicators. Several BI products exist, but they typically have a "data-centric focus", and are "unaware of the processes the data refers to" [19]. This approach is based on traditional Online Analytical Processing (OLAP) engines that enable an analysis of the OLAP cubes. That analysis makes it possible to drill down and up among the defined dimensions (typical dimensions are regions, periods, and products) to see the consolidated information we wanted to obtain, and to "dice" the cube to view the multidimensional data from different angles [19]. Process mining, on the other hand, takes a BP execution event log as input and extracts information about its real execution. This makes three different types of process mining possible: discovery, to find the corresponding BP model, conformance, to check if the execution corresponds to the existing BP model, and enhancement, to extend or improve an existing BP model with its execution information [19]. This means that not only the control-flow of the BP can be mined from the event log; so also may the organizational perspective (to analyze the relation between resources and activities), the time (to show performance information, bottlenecks, resource utilization), or the decisions (to discover rules based on decision points in the BP) [19]. As stated in [19], "although in recent years the tool support for the approaches mentioned has matured considerably, most tools are not really "intelligent" and do not provide any process-mining capabilities. The focus is on querying and reporting, combined with simple visualization techniques showing dashboards and scorecards". In contrast, the ProM[1] framework is one tool, among others, that does support process mining and which at the same time is available for free use [19]. ProM is based on several plug-ins that use data from the execution of BPs registered in event logs to discover BP models, to check conformance with existing BP models, and to extend existing BP models with execution information. However, few plug-ins provide information for operational support [20,21]. There is, moreover, a general lack of an integrated view that provides the business area with the complete picture needed to analyze the execution of BPs in the organization, while simultaneously including information on the system execution.

According to [22], several improvement initiatives have become popular over recent decades. These include Just in Time (JIT), TQM (Total Quality Management), Lean manufacturing, BPR (Business Process Reengineering), and Six Sigma. Approaches such as TQM and Six Sigma are based on an analysis of business processes using different techniques, as well as on integrating improvement opportunities, based on an improvement cycle (PDCA, DMAIC); all these approaches are used by many organizations. Other life-cycle models such as IDEAL were developed specifically for software process improvement, based upon the Capability Maturity Model (CMMI), in widespread use by software organizations. In [23–26], several aspects of the use of Six sigma, TQM, CMMI and IDEAL are presented and discussed, along with results from the application of these initiatives. One finding of such studies is that "like TQM, Six Sigma requires a strong incorporation of the corporate control system to enable companies to objectively measure and monitor their long-term development and monetary outcome of TQM using statistical techniques" [27]. The reference to the use of CMMI and ISO models is also interesting; these are difficult for small and medium organizations (SME) to apply because of the complexity of their recommendations and the large investment needed in terms of time and resources.

1.2. Research question and methods

The outline of the problem we have presented above, along with the background we have sketched out, led us to identify some research challenges for our work. These refer to the lack of an integrated approach that would help organizations to improve their BPs based on their implementation with services. There is also a need for an execution measurement approach that is guided by a systematic way of integrating the improvement opportunities found in the BPs. Many existing approaches are too general, or address only one of the problems; for example, they may tackle only the definition of execution measures for collecting data about BP execution, or techniques for analyzing execution measures, or improvement processes to guide the improvement effort (c.f. Section 6). When trying to put all the views and elements together, improper integration might lead to unwanted results. We have been working with the Hospital General de Ciudad Real (HGCR) since 2007, in a project to introduce BPM in the organization. In this project, a selected set of BPs were modeled with BPMN, and we then assessed the quality of the BP models obtained, as well as the process followed to generate them [28]. This led to several interesting findings, for both the HGCR and the BPM community. Many of the problems detected in the HGCR inspired us to work on a different approach for the management and improvement of BPs. Moreover, the hospital wanted to select a BPMS that would enable it to initiate a program for implementing and executing the selected subset of BPs. It also wished to introduce and institutionalize an approach that would further guide the modeling, implementation, execution, evaluation and improvement of BPs in the organization.

Taking all this into consideration, we concluded that having an integrated approach to carry out improvement efforts in organizations in a systematic way could be very useful. Such an approach would need to bring together (1) the realization of BPs by means of services, as well as (2) the definition, implementation, collection and calculation of execution measures for both BPs and service execution. This would have to be presented in an integrated view, showing key execution measurement results. Moreover, the measurement approach would be embedded within a continuous improvement approach, defined as simply as possible, so as to be

[1] http://www.promtools.org/prom6/

easily integrated into the organization's culture and way of doing business. That would allow organizations to find improvement opportunities for both the BPs and the services implementing them, as well as to perform the integration of improvement of both BPs and services in a systematic way. They would thereby achieve the overall business goals of the organization, as well as the ones specific to each BP analyzed. With all the above needs in mind, we formulated the following research question:

> ''How to set up a continuous improvement cycle for business processes implemented by services in organizations, based on BP execution measurement?''

Given the breadth of the question, we looked for more in-depth answers, entering into more specific aspects derived from the initial question:

(i) what measurement and improvement activities are needed in the BP lifecycle when BPs are realized by services ? (i.e., research question RQ1)
(ii) what measures are appropriate for obtaining information about BP and service execution within the organization? (i.e., research question RQ2)
(iii) what tools should be available to support the definitions in RQ1 and RQ2? (i.e., research question RQ3)
(iv) are the proposals made in RQ1, RQ2 and RQ3 appropriate and useful for use in organizations ? (i.e., research question RQ4)

The measurement and improvement activities that we have included in BPCIP constitute the answers to RQ1 and are set out in detail in Section 2. The answer to the second question is the execution measurement model BPEMM, where we organize all the execution measures in a taxonomy, which is detailed in Section 3. The third question relates to the tool support we have defined; the answer to this is set out in Section 4. Finally, the answer to the fourth question corresponds to the validation of the proposal and we present this in Section 5.

To address the research objectives introduced previously, we applied a combination of different research methods. In the first place, we carried out a systematic literature review at the beginning of our research work, to collect existing proposals related to our research question (cf. [29]). Moreover, execution measures presented as part of RQ2 were distilled by means of an extensive literature review that was guided by experts in the subject, as well as by a systematic review about BP measures that had been carried out [30] in the context of a related research work. After that, we followed design science principles for the development of our proposal, as suggested by [31]. We worked closely together with business experts from the quality group of the HGCR, conducting several workshops for requirements elicitation to ensure that the proposals addressed real business needs. The proposal was then developed by means of several iterative cycles. In an initial cycle we defined the problem, reviewed existing literature and defined the main elements of the general MINERVA framework. For each of the three types of elements defined in the framework (conceptual, methodological, tool support) general cycles were carried out. This enabled us to answer the sub-questions and to define the different components of the framework, based on those answers. We created a set of artifacts to address the business requirements related to the research question of the proposal we are presenting here. These artifacts are: (i) a method description (i.e., BPCIP), (ii) an execution measurement model (i.e., BPEMM), and a (iii) tool chain. To validate the appropriateness of the artifacts we designed, a case study that was conducted at the end.

1.3. Comparison

The added value of our integrated approach based on execution measures for the continuous improvement of BPs realized by services when compared with existing proposals can be stated as follows: firstly, the BPCIP improvement approach includes only a minimal set of key activities, roles and artifacts for guiding the BP lifecycle with explicit measurement and improvement activities. Our case study showed that this makes it easily understandable and easy to integrate into the organization's culture and way of work. Other approaches, such as Six Sigma, define several roles and activities on the basis of a statistical analysis of BP behaviour; that can be difficult to integrate in organizations. Similar considerations apply for CMMI [26]. Secondly, the guidance provided by the BPCIP is supported by the execution measures integrated in the BPEMM model. Although several execution measures exist in literature, we selected the most important ones for measuring the execution of BPs and services and integrated these into our proposal. This created a centralized model which allows both business and IT to refer to the same definitions. This in turn serves as the basis for guiding the defined measurement and improvement activities. Although the BPMM, CMM, and CMMI models also provide measurement guidance to reach the fourth and fifth levels, with a statistical analysis focus, they do not give an explicit definition of measures and or measurement, or a roadmap for the execution of improvement activities [17]. The integration of BP execution with the execution of the services implementing those BPs, as defined in our proposal, provides a complete vision of the real operation in the organization. This can be presented both to the business and IT area enabling the discovery of improvement opportunities. Another added value of our proposal refers to the BPEMM ProM plug-in we have developed to support the processing of the BPEMM execution measures. It shows the measurement results in a graphical and easy-to-use way, which will be accessible to the ProM community and organizations wishing to evaluate it. In relation to this, twelve open-source process engines were assessed in how far the data needed for the calculation of the BPEMM measures is registered by the tools, allowing integration with ProM for the analysis.

The rest of the article is organized as follows: in Section 2 we present the BP Continuous Improvement Process (BPCIP) we defined, describing its disciplines, activities, roles and phases. In Section 3 we set forth the definition of the BP Execution Measurement Model (BPEMM) along with the specification of the execution measures we integrated, and in Section 4 there is an explanation of the tools provided to support the proposal. Section 5 presents a case study to demonstrate the applicability of the approach, and in Section 6 of related work other proposals are discussed. Finally, Section 7 discusses the conclusions we have drawn and outlines future work.

2. BP continuous improvement process (BPCIP)

In this section we present the BPCIP improvement process that we defined, which is our proposal for answering research question RQ1 as stated in Section 1. BPCIP's main objective is to guide the execution measurement and improvement efforts in the organization, providing a systematic way of integrating improvement opportunities found in BPs and service implementation. We believe that the explicit definition of measurement and improvement activities will help to guide the execution measurement and improvement effort throughout the whole lifecycle. That in turn

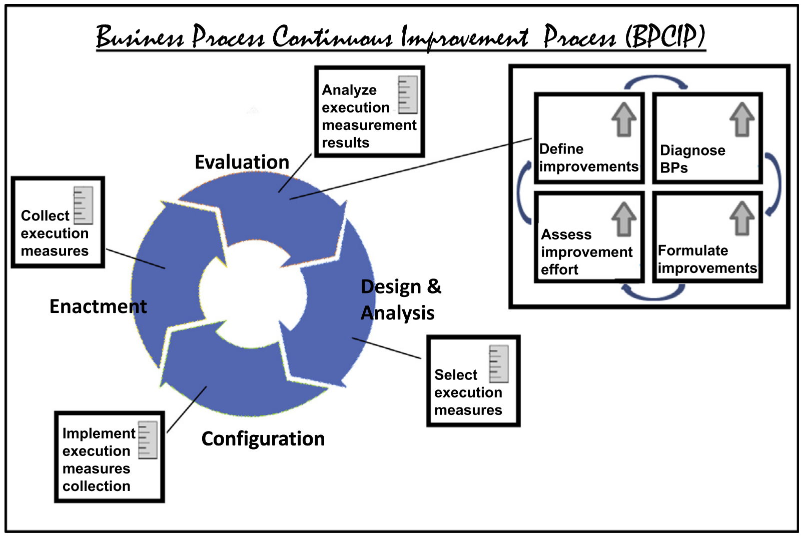

Fig. 1. BPCIP lifecycle extending the BP lifecycle (Weske, 2007) with execution measurement and improvement activities.

will help to obtain the insights needed about the real execution of BPs, and will assist adopters of the proposal in taking the corresponding actions to improve them. This is why we defined BPCIP by extending the BP lifecycle [1] with explicit measurement activities and the execution measures in the BPEMM model, along with explicit improvement activities we defined based on the improvement process PmCompetisoft [26]. We updated an initial definition of BPCIP [8], which we have extended significantly by redefining the complete BPCIP lifecycle based on the feedback from the conference, to make the BPCIP lifecycle less complex than proposed at first; this is the version presented here. Following design science principles, we based the definitions in BPCIP on the existing body of knowledge, as well as on input from the business experts in the Hospital General de Ciudad Real (HGCR). In Fig. 1 the complete lifecycle defined by BPCIP is presented.

BPCIP defines *Disciplines* and *Phases* in a similar way to the Unified Process [32]. We have implemented the BPCIP as an EPF Composer[2] Method plug-in, in order to provide interoperability with other processes defined in the same way. Users can access it easily on MINERVA's Web site.[3] We present the BPCIP including its Disciplines and Phases, in the next two sections: BPCIP Disciplines (cf. Section 2.1) and BPCIP Phases (cf. Section 2.2).

2.1. BPCIP Disciplines

To establish the Disciplines of BPCIP we use a "primary categorization mechanism for organizing tasks that defines a major area of concern and/or cooperation of work effort" [33], identifying activities, artifacts and roles needed to guide the management and continuous improvement of BPs in the organization. The Disciplines and their activities including their input and output artefacts and roles are presented below, organized by Discipline.

The *Business Modeling* Discipline aims to obtain a map of the organization and its BPs, to gain a better understanding of the business by representing their BPs explicitly as models. An explicit representation of an organization's BPs has the advantage of showing how the BPs are performed in the organization supporting the discovery of improvement opportunities. Organizations may also have to re-design BPs to integrate improvements after analyzing their execution, or when validation and/or verification results are not as expected. There are three activities in this Discipline:

- *BM1 – Assess the Organization:* This activity provides insights into the business area of the organization, its business goals and the business process defined, together with how they operate. The objective is also to look at the employees and other participants involved (i.e., customers, competitors) and the technologies used in the organization, among other elements, as well to consider the problems and areas for improvement. Organizations can use in this activity the OMG Business Motivation Model (BMM) [34] to show the goals and strategy of the organization.
- *BM2 – Identify and model BPs:* This activity aims to identify and model BPs in the organization. It includes the activities carried out to perform the BP, and how these are performed (i.e., manually or automatically). It also covers the control flow of the BP, defined by the sequence of activities and the diverging and converging of flows based on decision nodes (i.e., gateways). It takes into account the internal participants (i.e., roles, organizational units) and the external participants (i.e., suppliers, clients, and business partners), inputs and outputs managed through the BP execution and the resources needed. There are a great variety of notations for BP modeling [29], although in recent years BPMN - which was initially promoted by [35] - has emerged as the most widely-used standard. Thus, we will use Business Process Model and Notation (BPMN2) [36] for the specification of BP models.
- *BM3 – Redesign BPs:* This activity takes place mainly in the context of an improvement effort after improvement opportunities have been detected for the BP. Redesign is about including the improvement opportunities that have been identified as part of activity IM3 in the BP model. These are opportunities that come to light thanks to an evaluation of the real BP execution when it has been analysed. Organizations may also perform this evaluation after carrying out a validation and/or verification of the BP model, to include improvements found prior to its implementation. To guide the redesign of BP models several proposals and approaches exist (e.g. [27,37–39]).

The *Business Process Validation & Verification* Discipline aims to validate and verify the BP model prior to its implementation to detect improvement opportunities early in the BP lifecycle. Validation of a BP refers to testing whether the BP behaves as expected. Verification, in turn, refers to checking whether the BP does not exhibit any undesired properties, such as the existence of deadlocks. There are two activities in this Discipline:

[2] http://www.eclipse.org/epf/
[3] http://alarcos.esi.uclm.es/MINERVA/BPCIP/

- VV1 – *Validate Business Processes:* The particular validation approach to be used to gain insight into the characteristics of the BP model prior to its implementation has to be selected. There are several validation techniques to validate the model, which are mainly classified into two distinct groups: analytical techniques and simulation (e.g. [22,40]). BP simulation is based on tools which provide the environment needed to simulate BPs execution. Analytical methods, in turn, include the use of mathematical formalisms such as queuing theory. Apart from those, other quality characteristics can be assessed such as complexity, coupling of cohesion, which provide extra information about the BP model, using existing design measures (e.g. [41–44]).
- *VV2 – Verify Business Processes:* Verification of BP model is performed to gain insight into several characteristics of the model prior to its implementation, for example, detecting deadlocks caused by violations of the soundness property [1]. To carry out this verification it may be necessary to transform the BPMN2 model into a suitable notation, such as Petri Nets.

The *Business Process Implementation* Discipline aims to implement BPs with services driven by models, using, for example, the BPSOM methodology. The BPSOM methodology generates service models, specified in Service Oriented Architecture Modeling Language (SoaML) [45] from BPMN2 models [13,14] and integrates tool support for performing the defined activities. Depending on the particular language used by the chosen process engine, like, for example, BPMN2, XML Process Definition Language (XPDL) [46], or Web Services Business Process Execution Language [47] further work needs to be done on the BP model. Several existing tools generate XPLD/WS-BPEL models from BPMN2 models, which can be used to perform this transformation.[4] We defined two activities in this Discipline:

- *I1 – Implement BP with services:* This activity involves the carrying out of two main tasks: first of all, there is the generation of the executable BPMN2/XPDL/WS-BPEL from the BPMN2 model specified in the Business Modeling Discipline. Secondly, there is the development of service models from the same BPMN2 model, making it possible to implement the services to realize the BP (using BPSOM).
- *I2 – Re-implement services:* This activity is carried out in the context of an improvement effort, when an improvement opportunity for the implementation of the services exists. The corresponding specification of services has to be updated accordingly, taking into account the definitions of the BP, as specified in the BP document.

The *Business Process Analysis* Discipline aims to analyze the execution of BPs; both in real time, by means of monitoring BP execution, as well as after the execution of a considerable amount of BP instances, registering the corresponding data. Based on the data provided in real time, business people can take immediate decisions to improve the execution of BPs. These decisions may include the assigning of more resources if a bottleneck is detected. The second kind of analysis sets out to process data from the execution of the BPs. This provides the business area with information that enables it to find improvement opportunities for the BP. In this discipline, there are two activities:

- *A1 – Monitor BPs execution:* The BP execution is monitored on the basis of the software products. This provides real time access to BP execution data, such as the current state of the BP case and its activities. Monitoring BP execution helps the organization to carry out, for example, re-assigning of resources to BPs.
- *A2 – Analyze BP execution:* From the data registered in the event logs and services/systems logs for each BP case executed, we may use several Process Mining (PM) techniques provided by the ProM framework to analyze the execution of the BP. We carry out this analysis by means of the ProM plug-ins for Process Mining (PM) and the ProM BPEMM plug-in for the calculation and visualization of the execution measures.

The *Business Process Execution Measurement* Discipline sets out to show explicitly the execution measurement activities to perform in the extended BP lifecycle of BPCIP. We have defined this discipline in consonance with the vision proposed in the CMMI, which establishes an explicit Measurement Process Area with specific activities to guide the measurement effort. To extend the BP lifecycle with explicit execution measurement we defined four activities:

- *EM1 – Select execution measures:* This activity sets out to determine which execution measures will be calculated from the execution of the BP, selecting them from the ones integrated in the BPEMM. The base measures established for each goal in the BPEMM provide the data to be collected from the BP execution to enable the calculation of the remaining measures.
- *EM2 – Implement execution measures collection:* In this activity the execution measures from the BPEMM have to be implemented in the infrastructure for the execution of the BP and the services, i.e., integrating the base measures that define the data to be collected into the process engine and the infrastructure used for service execution.
- *EM3 – Collect execution measures:* The process engine and the infrastructure executing services collect the data for the execution measures selected from BPEMM, as the BP cases execute. In particular, the engine registers data such as the start and completion time of activities, as well as the corresponding performers and the data involved, making it possible to calculate the execution measures in the BPEMM.
- *EM4 – Analyze execution measurement results:* To analyze the execution measurement results as defined in the BPEMM, the ProM plug-in we have developed (cf. Section 4) calculates and visualizes the execution measures organized in different views (cf. Section 3.2). Based on the measurement results for the BP, the business area can find improvement opportunities that may be incorporated in the BP to better achieve the business and BP goals defined, by performing activity BM3.

Finally, the *Business Process Improvement* Discipline shows explicitly the improvement activities in the context of the BPCIP. The activities provide the basis for systematically guiding the improvement effort that seeks to integrate improvement opportunities into the BPs. Four activities should be performed after improvement opportunities for the BP have been found. The focus is on an agile integration which minimizes the effort of carrying out the improvement process, thus, making it easier to use. The definition of the improvement activities uses the improvement process PmCompetisoft [26] as its basis; this process takes those objectives into account. There are four activities in the Improvement Discipline:

- *IM1 – Define improvements:* This activity implies stating what improvements will be carried out to integrate the improvement opportunities found by analysing the execution measurement results (i.e., EM4). The improvement proposal is aligned with

[4] http://www.businessprocessincubator.com/do/convert.html

the strategic business goals of the organization and the specific goals for the BP; the management area must approve this proposal to ensure the sponsorship for the improvement.

- *IM2 – Diagnose Processes:* The diagnosis of the BP using the BPMM standard implies a review of the definition of the BP and the way it is being carried out in the organization, to assess its maturity level. The BPMM follows the format that is set out by the software maturity models (CMM, CMMI) and includes several Process Areas and Key Activities, which, when performed, allow the BP to gain maturity by moving to a higher maturity level.
- *IM3 – Formulate improvements:* This activity aims to establish explicitly which parts of the BP model and which services will be modified to achieve the improvements defined. After performing this activity, the complete BP lifecycle (from the Design & Analysis phase to the Evaluation phase) is executed again to create the new BP version including the improvements, to implement and execute it and to collect data about its execution.
- *IM4 – Assess improvement effort:* This activity implies the evaluation of the achievement of the goals specified for the improvement effort, both in terms of the improvement of the BP as defined and as regards the schedule, resources and cost established for the cycle. The old and the new BP version outcomes are compared in order to see whether a real improvement has been achieved. It also includes the carrying-out of a post-mortem analysis to assess the development of the improvement cycle.

2.2. BPCIP Phases

The BPCIP consists of four phases, starting with the modeling of a new BP or with redesigning an existing one in BPMN2 following best modeling practices such as workflow patterns [48], and then implementing it by means of services based on models. The execution of the BP is measured and evaluated using BPEMM execution measures, to identify improvement opportunities. These improvements can then be fed back into the BP following a systematic approach based on the improvement activities we have established. Fig. 2 presents the activities we added to be performed in each BPCIP lifecycle phase; Table 1 shows a summary of the activities relating the disciplines and phases, following the Unified Process style.

3. BP Execution Measurement Model (BPEMM)

In this section we present the BPEMM measurement model, which is what we propose as an answer to research question RQ2, stated in Section 1. The BPEMM model provides a set of execution measures for BPs and service execution, for use in the context of the continuous improvement effort of the organization. We believe that having a predefined set of execution measures will help organizations to focus on the evaluation of selected aspects of BP execution, preventing them to spend valuable time in defining the execution measures by themselves. For defining the BPEMM we followed design science principles, by choosing the execution measures of BPEMM according to findings from the expert-guided review of existing literature [22,37] that we carried out, and on the basis of a systematic review on general BP measures [30] from a related work, as well as by using input from the business experts belonging to the hospital (HGCR).

Then, to define the structure of the BPEMM model we applied the same reasoning that underpins the structure of existing software architectural approaches which handle complexity by defining different views for different purposes [49,50]. This allows overcoming the difficulty of showing different aspects of the solution to different stakeholders with different needs, thus organizing our measures in three different views: Generic, Lean and Service views. In this sense, inside each view, we also organized measures by means of four dimensions and into a three-level hierarchy. We describe the BPEMM execution measurement model structure in Section 3.1 and the definition of the selected execution measures in Section 3.2.

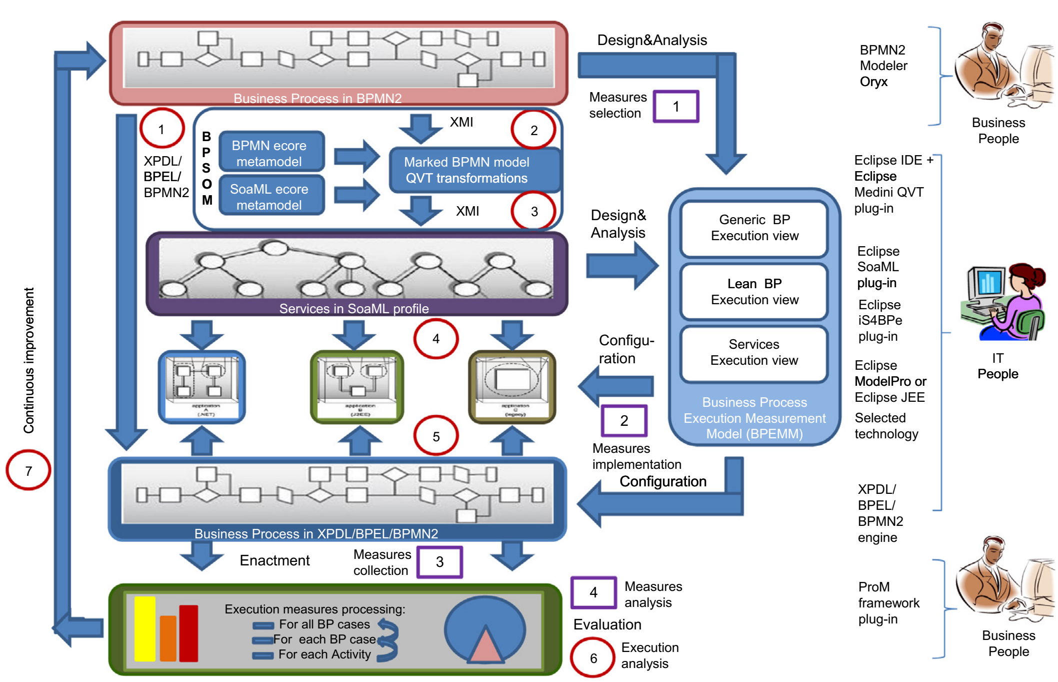

Fig. 2. BPCIP work method in the MINERVA framework.

Table 1
Summary of activities in Disciplines and Phases.

Discipline/Phase	Design & Analysis	Configuration	Enactment	Evaluation
Business Modeling	BM1-Assess the organization BM2-Identify BPS BM3-Redesign BPS			
BP Validation & Verification	VV1-Validate BPS VV2-Verify BPs			
BP Implementation		I1-Implement BPs with services I2-Re-implement services		
BP Analysis			A1-Monitor BP execution	A2-Analyze BP execution
BP Execution measurement	EM1-Select execution measures	EM2-Implement execution measures	EM3-Collect execution measures	EM4-Analyze execution measurement results
BP improvement				IM1-Define improvements IM2-Diagnose processes IM3-Formulate improvements IM4-Assess improvement efforts

3.1. BPEMM overview

BPEMM is based on the Goal, Question, Metrics (GQM) [51] paradigm, the foundation of which is the idea that an organization must first specify its goals if it is to measure what the organization does in a meaningful way. Initially defined to evaluate defects in software projects, its use has extended to several other domains such as improvement efforts in software organizations, and design of Software Engineering experiments. Our proposal includes several elements that also come from the software area, such as the improvement activities added to the BP lifecycle, and the BPMM model which has its origins in CMMI and CMM. The use of GQM to define the BPEMM is set in the same direction. The aim of BPEMM is to help business people to choose predefined execution measures for BPs implemented by services; it was designed specifically for these kinds of systems. Therefore, BPEMM measures have three main elements:

- Goal: defined for the organization, section, project or process, from various points of view and models.
- Question: describes how each goal will be evaluated from the point of view of a quality characteristic.
- Metric: a set of data, which can be objective or subjective, put together to answer each question quantitatively, also by means of elements defined in the Software Measurement Ontology (SMO) [52], such as:
- Base measure: a measure of an attribute with no dependence upon any other measure, and whose measurement approach is a measurement method.
- Derived measure: a measure derived from other base and derived measures, using a measurement function as measurement approach.
- Indicator: a measure derived from other measures; its measurement approach is an analysis model which has an associated decision criterion (defining ranks to which the measurement results can belong).

To make the structure we have defined for the specification of execution measures in the BPEMM clear, Table 2 shows an example for the Throughput Time (TT) of the BP. Note that this is not the complete definition of measures for TT which is presented in Section 3.2 and Appendix A, but just an example illustrating the way in which measures are defined in the BPEMM. As it can be seen in Table 2, the specification starts with the definition of a goal and questions to provide a way of reaching the goal; in the example there is only one goal specified, but there can be as many as necessary.

For each question, we specify several execution measures by means of the SMO. They include base measures which will be implemented in the BP to gather the corresponding data. They also comprise several derived measures using this data and maybe other derived measures, as well as indicators which can also be determined using the previous ones as a basis. Indicators also have ranks to which they can belong, attached to their measurement results; these provide information about the results obtained. We use the ProM metaphor of a traffic light to assign colours to ranks, with the following meaning: "Green" for OK, "Yellow" for Warning and "Red" for Stop (indicating problems).

For each measurable concept identified as being of interest for the business area we have specified several execution measures for BPs and services realizing them in the BPEMM. Table 3 displays the measurable concepts along with their definition and the goals they are related to.

3.1.1. BPEMM views, dimensions and hierarchy

BPEMM execution measures are grouped according to three specific views: Generic BP, Lean BP and Service execution, to manage complexity and to clearly separate different concerns. We organize these views taking into account the dimensions time, cost, flexibility and quality in the "Devil's quadrant" [37,53], as shown in Fig. 3. The use of the "Devil's Quadrant" dimensions helps analyzing the trade-offs that have to be considered when designing

Table 2
Example of execution measures specification in BPEMM.

Goal	**G1**	**Minimize the Throughput Time (TT) of the BP**
Question	**Q1**	**What is the actual TT of the BP?**
Measures	M1 (base)	Start time of an Activity (ST)
	M2(base)	Completion time of an Activity (CT)
	M3(derived)	Working time of an Activity (AWoT = CT − ST)
	M4(derived)	Throughput Time of a BP case (BPTT = TWoT + TWaT)
	M5 (indicator)	Average BP Throughput Time for all BP cases (ABPTT = L BPTT/Total BP cases) Decision criteria = Inverse Percentage DC R1:
D. Criteria	Percentage DC	R1: 0 <= TTI <= L1 = "LOW" = RED;R2: L1<=TTI < L2 = "MEDIUM" = YELLOW; R3: L2 <= TTI = "HIGH" = GREEN

Table 3
Measurable concepts and Goals defined by execution view.

Execution view	Measurable Concept	Definition	Goals
Generic	Throughput Time (TT)	Total time from the moment in which a BP case is initiated to its completion [22]	Min TT, Max efficiency
	Capacity	Number of BP cases per unit of time that the BP can handle, for resources (bottlenecks) [22] (adapted)	Max capacity, Min bottlenecks
	Resources	Tangible assets necessary to perform activities within a BP [22]	Min quantity, Max utilization
	Cost	Of human resources to produce a good or deliver a service [37]	Min cost
	Path execution	Successful path execution (i.e. charging credit card) vs. unsuccessful path (i.e. charged rejected)	Max successful, Min unsuccessful
	Final state	State of BP ending, apart from successful or unsuccessful (i.e. normally, aborted, cancelled)	Max normal, Min abnormal
	Quality	External: user satisfaction with the product or process, internal: condition of working in BP [37]	Max external and internal quality
	Flexibility	Ability to react to changes, i.e. resources executing tasks, process handling cases, change workloads, change structure [37]	Max flexibility Min time for introducing changes
Lean	Rework	Loop in the BP with control specifying criteria to allow a job to continue processing [22]	Min execution of rework loops
	Value-adding activities	Activities essential for the BP to meet customer's expectations [22]	Max value-adding activities
	Non value-adding activities	Activities that does not add value to the customer [22]	Min non value-adding activities
	Defects/errors	Defects/errors in process/products causes repair, rework and waste [22]	Min defects/errors
Services	Response Time	Guaranteed time interval for the execution of the response of an event [58]	Guarantee a defined response time
	Throughput	Number of event responses completed over a given observation interval [58]	Guarantee a defined throughput
	Capacity	Maximum achievable throughput without violating specified responses time [58]	Guarantee a defined capacity
	Availability	Service readiness for usage [58]	Guarantee a defined availability
	Reliability	Service ability to keep operating over time [58]	Guarantee defined reliability
	Confidentiality	Property that data be inaccessible to unauthorized users [58]	Guarantee defined confidentiality
	Integrity	Property that the data be resistant to unauthorized modification [58]	Guarantee defined integrity

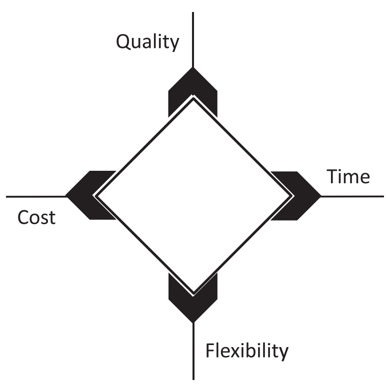

Fig. 3. Dimensions of the devil's quadrant [53,37].

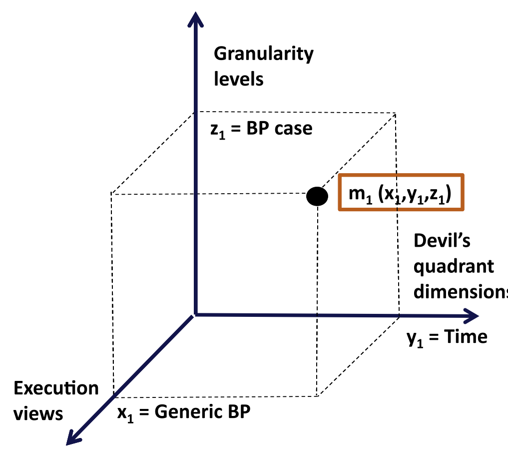

Fig. 5. BPEMM "Magic cube" defined by the three dimensions for execution measure specification.

or redesigning a BP; changes in one dimension can impact another negatively, i.e. removing an activity can improve the duration of the BP, but at the same time may have a negative effect on its quality.

Measures are also arranged in a three-level hierarchy as shown in Fig. 4, which defines the Granularity level of the execution measures. At the third level, measures for each activity instance are registered; at the second level these measures are combined to calculate them for each corresponding BP case, and, finally, at the first level the BP case measures are combined to calculate the ones for the BP, such as averages or percentages.

For the tridimensional organization of BPEMM, we have created a "cube" view diagram (cf. Fig. 5), to show the elements: Execution views (Generic BP, Lean BP and Services), "Devil's quadrant" dimensions (time, cost, flexibility and quality) and Granularity

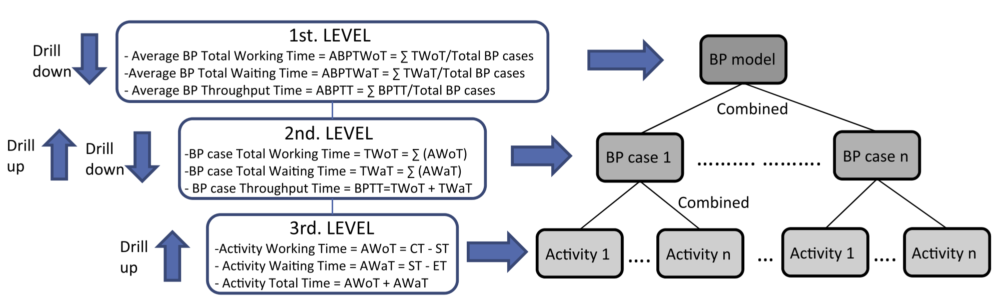

Fig. 4. Defined hierarchy of levels for execution measures: BP, BP instances and BP activity instances.

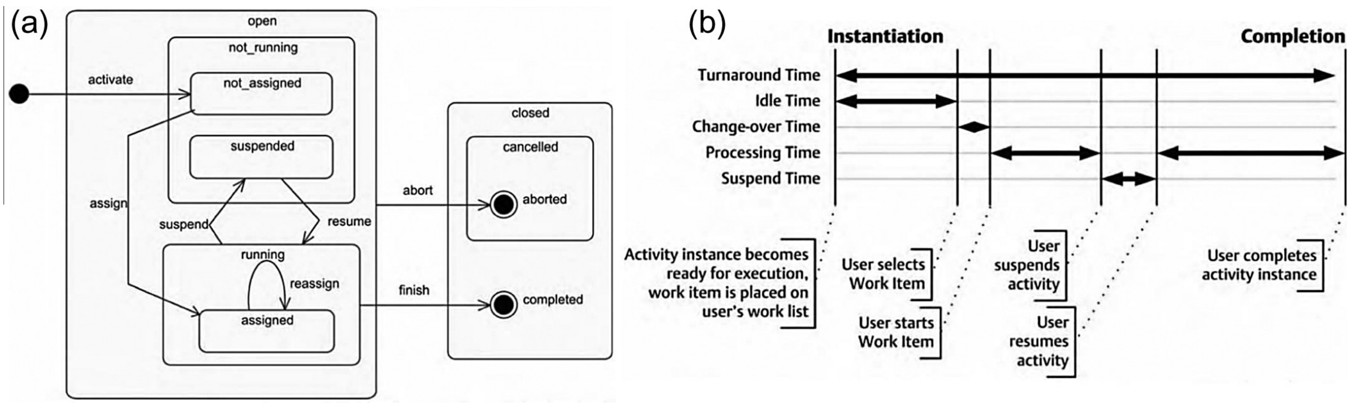

Fig. 6. (a) Activity lifecycle and (b) execution times from (zur Muehlen, 2004).

levels (BP, BP cases and Activity instances). These three dimensions are used to present the measurement results for several combinations of elements from the cube dimensions.

In the 3D-space presented in Fig. 5 the example point $m(x_1, y_1, z_1)$ represents the selection of: the Generic BP execution view from the Execution view dimension, the Time dimension from the dimensions of the Devil's Quadrant dimension and the BP case from the Granularity level dimension. This will result in the set of measures that can be calculated to show, for example, execution measures in the generic view, related to the Time dimension at the granularity of BP cases such as the Total working time of a BP case, or the Throughput Time (TT) of a BP case.

3.1.2. Assumptions for calculations

In this section we present the assumptions we make for the calculation of BPEMM execution measures, regarding the Time dimension, the need for the BP model and for context data, and the service measures. We need to define how to tackle various aspects of the BP execution in order to be able to calculate the measures.

The execution times for the activities of a BP include processing (or service) times, as well as waiting (and queuing) times. The times relate to the lifecycle of activities which defines the states in which an activity may be during BP execution. Fig. 6(a) displays an example of an activity lifecycle; in (b), there is a display of related execution times. As regards the activity lifecycle and the different states and transitions that can occur during BP execution, we have simplified the approach. We establish a minimum core set of data that we must collect to be able to calculate them. This corresponds to the three key times of: enabled, start and complete time. We have decided to use these key times because they are the ones most commonly provided by BP engines (c.f. Section 4.2). However, if the enabled time of the activities is not registered, we consider the completion times of all activities required for their enablement and take here the latest time as the enable time.

Nevertheless, it is easily possible to extend the BPEMM execution measures such that we can take other times that can be registered into account: these might include, for example, *suspended* and *resumed*, which we will leave for future work.

To calculate the cost of the BP associated with human resources, data on the resource executing each activity is required. With respect to states for the BP case, we usually register the start and completion times of BP cases. In case this information is not registered we can derive them directly from the ones defined for the activities, considering the structure of the BP model. For example, the start of the BP case can be approximated as the enable time of the first activity and the completion time of the BP case as the completion time of the last activity.

For the calculation of service execution times, we also use several times, starting from the enabled, start and completion times for BP activities. We also defined specific times to register the execution of services in the associated infrastructure, such as when the invocation is received, the execution starts and the execution completes and responds to the BP. We explain all this in the Service execution view (cf. Section 3.2.3), as there are several concepts and definitions that we need to clarify to help the reader understand those times.

The control flow view in BP models defines the order in which the activities can be executed, and bifurcations that may occur in the flow; several workflow patterns [48] have been defined to express these modeling options. For the calculation of execution measures three key patterns are particularly important, as they imply different ways of calculation. Fig. 7 presents these three patterns, along with the corresponding formulas to calculate their associated times (adapted from [22]). To perform these calculations we will require the BP model.

The patterns presented are: (a) loop: an activity or several activities are repeatedly executed; (b) exclusive branching: only one of several branches can be executed, and (c) parallel branching: all the branches defined by the decision point will be executed.

In (a), we have to add up all the execution times for the activities executed within the loop to obtain the execution time for that loop. In (b), we add up only the activities of the executed branch. In (c), only the maximum of the execution times for all branches is taken into account. For other patterns, the calculation is similar; for example, to calculate the times for an inclusive branching, the formula is the same as for the parallel path, but we take only the executed branches into account.

As far as the BP model is concerned, we assume that each activity will have a unique label. This allows us to define, for example, domain-specific measures like the definition of a "successful branch" (cf. Section 3.2.1) or to specify which activities take part in a rework loop (cf. Section 3.2.3), using the label of the activity as identifier.

All the information that we need and which cannot be obtained from the collection of data from the BP execution will come in the form of a document or context data provided by the business area. It may include information such as the definition of rework loops by indicating which activities correspond to it, or the salary of each participant in the BP which is necessary to calculated measures related to the Cost dimension.

3.2. BPEMM execution measures

In this section we present the set of execution measures included in the BPEMM, classified according to the three execution

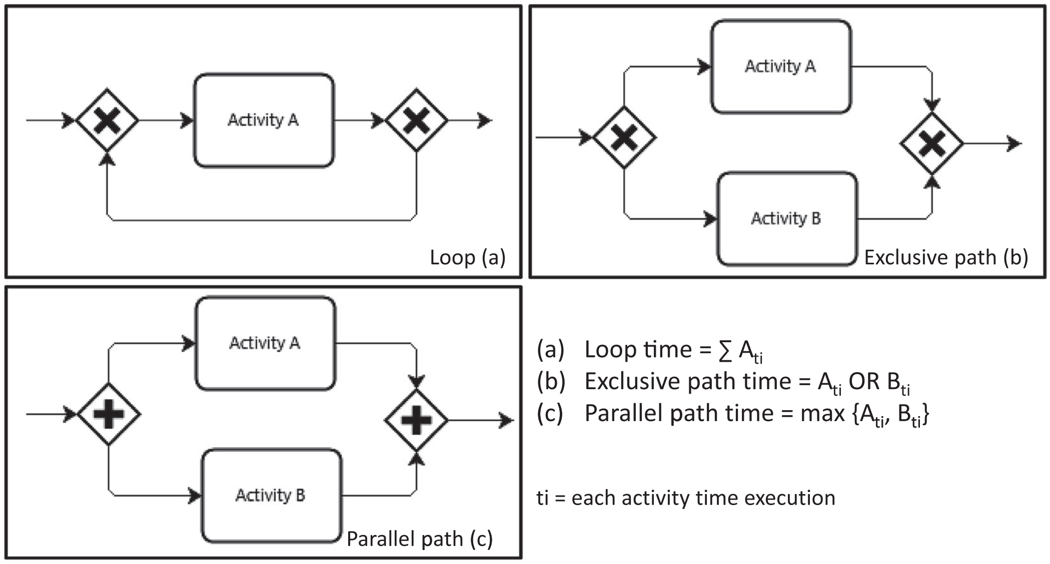

Fig. 7. Calculation of times for different patterns (adapted from [22]).

views we defined: Generic BP, Lean, and Services. For each one we provide the dimensions of time, cost, quality and flexibility-if definitions for a particular dimension exist-and for the hierarchy defined.

3.2.1. Generic BP execution view

In this view we integrate generic and domain specific execution measures for domains such as healthcare, software or production. These measures relate to generic BP characteristics that are not themselves related to the type of BP. Generic measures include the duration of activities, the duration of the complete BP, costs, roles involved, along with quality as perceived by the user. Domain-specific measures, in turn, refer to measures that have to be instantiated for each domain. The business area defines this kind of measures as Key Performance Indicators (KPI taking into account specific expected results, such as the quantity of received or delivered orders, products, or successful payments.

In the *Time dimension*, one key measure refers to the Throughput Time (TT) [22,37–39], which, as presented in Table 3, is defined as the total time from the moment in which a BP case is initiated, to its completion [22]. Considering the enabled time, the start time, and the completion time of an activity (cf. Section 3.1.2) we can calculate, among others, the working and waiting time of an activity, the Throughput Time (TT) of a BP case, and the average TT for all BP cases. Table 4, included in Appendix A, shows all the measures defined for the Time dimension – BP Throughput Time (TT).

We integrate several existing execution measures for the Time dimension. These are based on those defined in [22,37,54], and new ones are established, such as the Index between Working and Waiting time for activities and BP cases (M10, M11). In Fig. 8 we set out, in a very simplified view, the relationship between the times we use.

The execution measures presented in Table 4 in Appendix A allow us to show information for each level of the hierarchy, as we calculate times for each activity, each BP case and averages for the BP model level. When analyzing the measurement results for the Time dimension, a key aspect is to search for high waiting times and high percentages of waiting times at each of the levels and to look for high average times for the TT of the BP. We should also keep a lookout for high indexes between working and waiting times, which will indicate that for the level analysed (for example an activity) the waiting time is much higher than the working time.

Another key measure for the *Time dimension* is the Capacity of the BP, which represents the number of BP cases that can be processed by a unit of time. This is restricted by the resources associated with the roles executing the activities in the BP [22]. We should remember that the capacity of the BP refers to the resource type and resources available for each one, so a key measure for capacity is to find the bottleneck of the BP, i.e., the resource type with the smallest capacity. The base measure defined for the BP capacity is the assignation of resources to roles executing each activity. The execution event log does not contain this; it comes rather in the information we gather from the business area, in the form of a document or context data to specify it; the BPMN2 model gives us the information about the role assigned to each activity. In Table 5, included in Appendix A, we present the BPEMM execution measures for the Time dimension – BP Capacity, which we have adapted from [22].

Since we know from the registered data how many times each activity has been executed, and we also know the average working times for the activities of the BP, we can calculate the unit capacity of each resource type. It indicates the number of BP cases each one can complete per unit of time (with the unit of time in the average time of the activities: seconds, minutes, hours, etc.). Hence, the resource type with the smallest capacity indicates the bottleneck in the process. It would be interesting for managers to see the average times of execution for each activity assigned to each person, so we plan to measure this.

For the *Cost dimension*, we have integrated execution measures to calculate the operational cost (i.e., the cost associated with resources assigned to activities). These costs can be human resources, or material resources, such as an operating room. To be able to calculate the cost, information about the base cost of each resource must be provided in the desired unit of time (hour, week, or month) so that we can use this to calculate their participation in the BP. Table 6, included in Appendix A, displays the execution measures determined for the Cost dimension. As can be seen, we calculate costs for each activity in the BP, for each BP case and for all BP cases, to provide a complete view that covers the hierarchy defined. Although this is an initial set of measures, it can provide valuable information regarding the costs associated with the BP. We will deal with this particular issue in future work.

Concerning the *Quality dimension,* we provide information about the way in which the BP cases "end" their execution, with two different approaches: the first one corresponds to the type of ending as defined in the BP lifecycle, such as completed, terminated and aborted. These measures provide information about the number of BP cases ending successfully or unsuccessfully, as well as the percentage they represent in the total execution of BP cases. These execution measures for the Quality dimension by type of ending are shown in Table 7, which we include in Appendix A. The second one corresponds to the definition of the domain-specific measure "successful path" that has to be instantiated for each BP and domain, by defining the activities included in the successful execution branches of the BP. The business people must do this when they are selecting the execution measures from the BPEMM. A document or context data registers all this, which will then be used for the calculation of the measures. Table 8, in Appendix A, sets out the execution measures with respect to successful branch execution. This approach is complementary to the first one presented, as it could be interesting to know, for all the BP cases, which ended in the complete state (previous measures), and which of them were successful or unsuccessful. An example of this might be the charging of a credit card for a sale. That information can be useful to detect, for example, a malfunction in the implementation of the BP, where many credit card charges are rejected due to poor interaction between services implementing the BP. Other measures for quality that we have not presented here include measuring client satisfaction and employee satisfaction by means of questionnaires about their perception of the BP execution. These can also lead to an improvement of the BPs.

The execution measures in the *Flexibility dimension* have to do with, for example, the times regarding the "change process" for the BP, which may originate from different sources: a request to change something in the BP by any person involved in its execution, an unforeseen event that prevents them from following the BP model as defined, or an improvement opportunity detected by business people in the evaluation phase when analyzing the BPEMM execution measure values. In this dimension what we want to measure is, for example, how long it will take to incorporate the needed change into the BP. To that end we will define, model and execute a change process, as with any other BP in the organization. Business people will use this process to track the changes made to the BP, as well as the duration of the change process. This means that the flexibility execution measures build upon the execution measures of the time and Quality dimension, but they focus on the particular BP that introduces changes into any other BP.

3.2.2. Lean execution view

This view defines execution measures used to collect information for detecting waste in BP execution. It aims to find activities, paths or parts in the BP that, if used to best advantage, can lead to an optimization and improvement of the complete BP [22]. Lean thinking was first introduced in the Toyota Production System (TPS); the basic tenet is the identification and elimination of waste, which is categorized in seven types: overproduction, waiting, transport, extra processing, inventory, motion and defects). Lean principles and waste types have been adapted to areas other than the manufacturing one, such as software development [55], information management [56] and healthcare [57]. Non value-adding activities usually have to do with handoffs, delays, rework, and control activities in loops.

In the context of this view the most important dimensions of the Devil's quadrangle are the Time dimension (focused on rework loops) and the Quality dimension; the cost and Flexibility dimensions, in turn, are the same as those defined for the Generic BP view. Rework loops play an important role in the Lean execution view, since they allow for the identification of non-value adding activities that generate delays. Rework loops are loops where the controlling business rule refers to quality aspects in the execution of the previous activities (for example, examining a product to see if it has been repaired) and determines if the flow can continue, or if it has to be re-executed to meet the quality levels established for the execution. Normal loops representing repetitive action (which are not rework loops) will not be considered by our measures, since these loops do not represent waste in form of non-value adding activities (e.g., a loop representing the buying of goods on a web page where the customer adds repeatedly items to the shopping cart). To be able to differentiate rework loops from normal loops, rework loops must be defined by business people in the context data document, by explicitly identifying the activities involved in each rework loop execution. Thus, we focus the definition of lean execution measures with respect to the Time dimension on the discovery of rework in a BP. From the Time dimension of the Generic BP view, we use defined base and derived measures to calculate the rework measures. Table 9, which we have included in Appendix A, displays the measures defined for the rework in the Lean execution view.

In the *Quality dimension*, execution measures for Lean involve the discovery of defects during BP execution. Any worker detecting a defect during BP execution may report the defect and even cancel the BP case if the error is not recoverable.

3.2.3. Service execution view

This view contains measures regarding the execution of services realizing the BPs, taking the Quality of Service (QoS) requirements for this type of software into account. Service measures take quality attributes for services as their starting point-attributes such as: performance (i.e. response time: processing time and latency, throughput, capacity), security (i.e. confidentiality, integrity), dependability (i.e. availability, reliability), as defined in [58–63].

To calculate the measures corresponding to the service execution view in the Time dimension, we have defined six times of interest for the activities and service execution to be registered in the BP execution. In the first place, in the BP activity we log the enabled time, the start time corresponding to when a service is invoked, and the complete time, corresponding to when the service returns an answer after processing the request. In the second place, in the service itself we log the times in which the service receives the invocation (enabled), starts its execution (start), completes its execution (complete) and sends the result to the BP engine, as shown in Fig. 9.

To calculate the measures for each service invoked, the data about the invocation has to be logged in the service. The application server running the service, for example can include this, by such means as logging the time of the invocation, origin and credentials in each invocation to the service. When the service is invoked outside the organization and we do not have control over

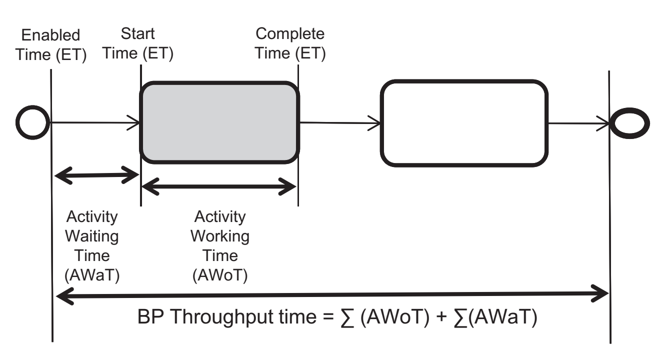

Fig. 8. Defined times for activities and BP instances execution.

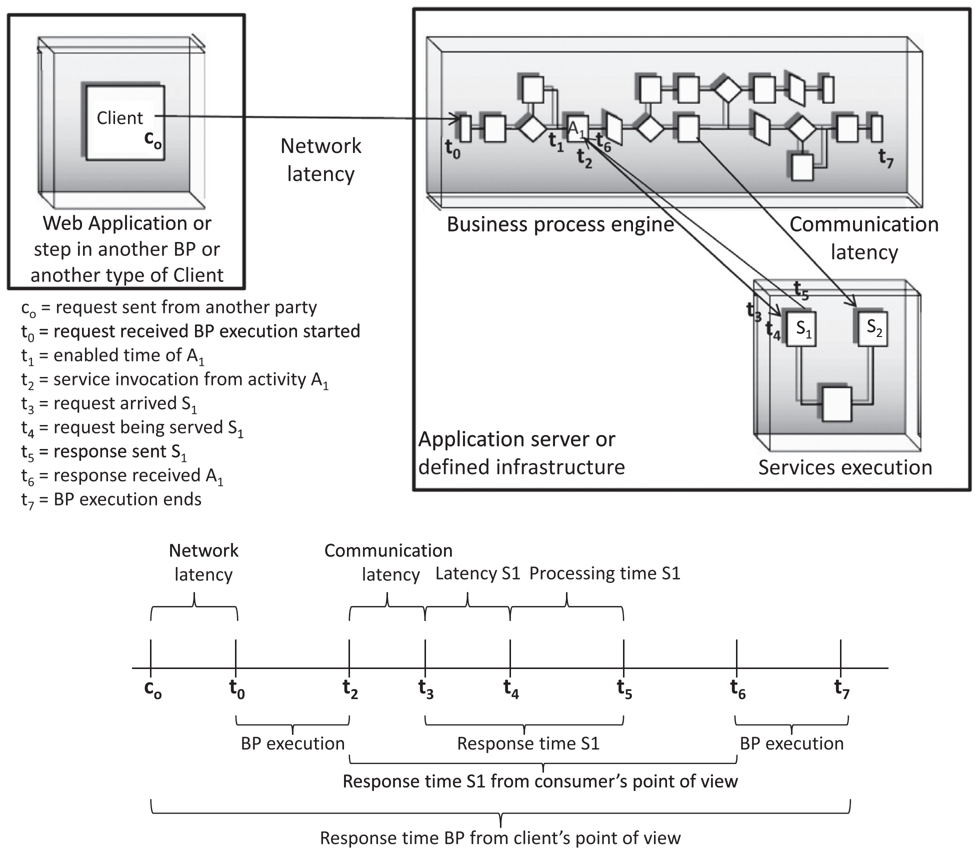

Fig. 9. Defined times for service execution and BP activities.

the environment of execution, we can use information from the communication, but the internal times for the service will be difficult to obtain.

As shown in Fig. 9, we use the time t_1 to log the enabled time of the activity invoking the service, t_2 and t_6 to log in the activity the invocation and the answer received from the service, respectively. We do this in the same way as if they were the start and completion times as before, the only difference being that the resource executing it is the system; t_3, t_4 and t_5 are used to register times in the service: the invocation received (enabled = t_3), the start = t_4 and the complete = t_5 times of the service processing.

When we do not have the complete data on service execution, we can nevertheless use the times registered from the point of view of the BP, to calculate the response time of the service. This will include communication latency, as well as latency and processing times for the service. In this way, we are able to log the start and complete times for the activity in the BP engine in the same way as we do with the rest of the activities in the BP; we calculate the waiting and working times of the activity, which include the times for the service execution. With this approach, we also include service execution times in the calculation of the Throughput Time (TT) of the BP; these can be shown only for those activities that involve the invocation of a service, providing the service measures for the BP implementation within the organization, or evaluating the interactions with partners.

As discussed before, we need to include several base measures in the implementation of services or in the infrastructure executing them. What we provide in this view, therefore, is the data that has to be logged, so that after implementation, we can add this to the services. In Table 10 in Appendix A, we show the service execution measures defined for the Time dimension – Service Response Time.

Other measures for service execution times correspond to the service throughput, which refers to the number of event responses completed over a given observation interval. We can calculate these only from the implementation of the service or the service infrastructure, so they make sense for internal services in the organization only. We present the service execution measures defined for the Time dimension – Service Throughput in Table 11 of Appendix A.

The measures for service execution times set out in Table 12 of Appendix A for the Time dimension – Service Capacity, are related to the previous ones, throughput and response times, and correspond to the service capacity. This refers to the maximum throughput that is achievable without violating specified response times. Once more, we can calculate this only from the implementation of the service or the service infrastructure, so these make sense for internal services in the organization only.

Most of the service execution measures data can only be gathered by implementing the log of the base measures in the services and/or the infrastructure. For example, to know how many unauthorized attempts to invoke a service have taken place over a certain period of time, we need to log all the invocations rejected because of the presentation of invalid credentials to the service. For this and related goals, the service execution measures defined for the Quality dimension, regarding availability, reliability and confidentiality, are shown in Table 13 of Appendix A.

4. Tool support

As part of our proposal we integrated several existing tools and developed new ones to answer Research Question RQ3, set forth in Section 1. One of the most common shortcomings of several meth-

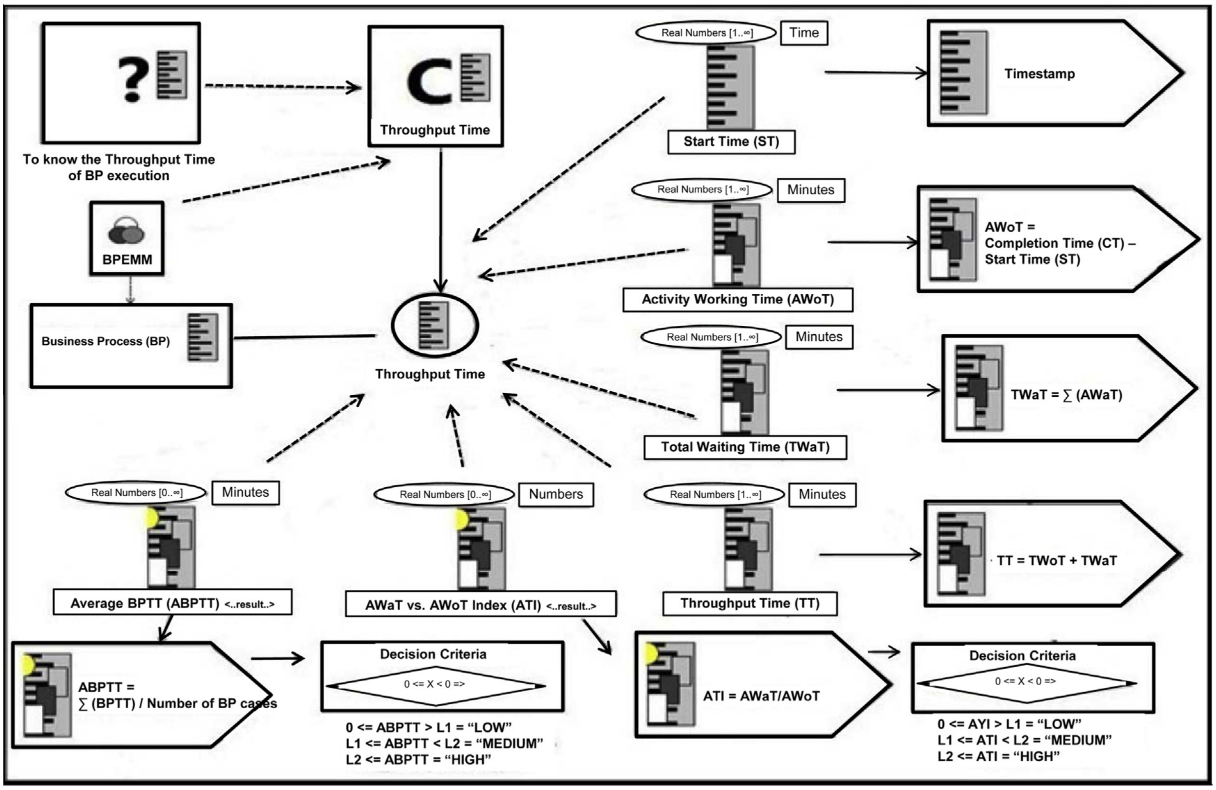

Fig. 10. SMTool example of use for the specification of the execution measures in graphical form.

odological proposals is their missing tool support to aid in the realization of the activities they define. However, the lack of respective support makes the proposals difficult to apply hindering real integration in organizations. That is the reason why we include tool support for use in the phases and activities we defined in the BPCIP, and to support the implementation, calculation and evaluation of the execution measures in the BPEMM. On the one hand, we evaluated several existing proposals and incorporated into our approach selected tools which meet our requirements for graphically specifying the execution measures (cf. Section 4.1) as well as for executing BP models and registering the data needed to calculate the execution measures we proposed, by using standard technology (cf. Section 4.2). On the other hand, we developed a proof-of-concept prototype, to integrate the evaluation of our execution measures into the ProM framework, following design science principles and refining the tool iteratively until the requirements were met (cf. Section 4.3).

4.1. Graphical specification of execution measures

In Section 3.1, we describe how the specification of the execution measures is done by means of the GQM in textual form, following the definitions of the SMO. This way of specification could be clear enough for business people to understand the BPEMM execution measures defined for the execution of each BP in the organization. But, following the adage "a picture is worth a thousand words", we also believe it is important to provide graphical support, in order to give a quick overall view of the elements involved in the measurement. For this reason, we have integrated the SMTool [64]. SMTool implements the concepts and relationships defined in the SMO, providing a graphical view of any measurement model, using the Software Measurement Modeling Language (SMML) [65]. The SMML diagrams are used in the Design & Analysis phase to make it easier for business people to understand BPEMM measures. Fig. 10 shows an example of the use of SMTool for the defined execution measures in the Generic BP view for the Time dimension.

There are several concepts and relationships shown for the definition of the execution measures: in the first place, in the top left-hand corner of the diagram, the information needed "To know the Throughput Time (TT) of BP execution" appears; this has to do with the measurable concept "Performance" to its right, and below this, the attribute to be measured – the "Throughput Time (TT)" of the entity "Business Process (BP)"– to its left. Then the three types of execution measures are defined, to measure the TT, that is: base measures (identified by the rule), for example "Start Time", in the upper right-hand corner of the diagram, derived measures below that (identified by the rule, inside the rectangles), for example "Activity Working Time (AWoT)", "Total Waiting Time (TWaT)" and "Throughput Time (TT)", and the indicators below to the left (identified by the rule inside the rectangles, with the yellow light); for example, "AWaT vs. AWoT Index (ATI)" and "Average BPTT (ABPTT)", each with its associated measurement approach.

4.2. Registration of event log data

The selected BPEMM execution measures have to be implemented in the infrastructure supporting the execution of the BPs in the organization. A key aspect in this sense is to know how to include the registration of the data that we need to collect in the event logs. To find out what type of execution event log information is registered, we have evaluated several BP engines that execute BPs in the languages of BPMN2, XPDL and BPEL, as these are widely used among enterprises. This helps us to differentiate between the data that we can obtain easily from the native log system of the engine, and the data we need to implement and add to the engine to be able to register the required information. We focused our evaluation on open source software based on their

Table 14
Business Process engines facilities for registering the defined data in the execution event logs.

Standard	XPDL					WS-BPEL					BPMN2	
Engine/Data	Bonita	Enhydra Shark	Joget	OBE	Wfm Open	Orchestra	Intalio	Apache ODE	Riftsaw	jBPM	Activiti	jBPM5
Activity ET	✓	✓	✓	–	–	–	✓	✓	✓	–	–	–
Activity ST	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
Activity CT	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
Service IT	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
Service AT	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
BP id	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
BP case id	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
BP case end state	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Activity id	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
Activity instance id	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
Activity user	✓	✓	✓	✓	✓	–	–	–	–	✓	✓	✓

availability, and the fact that any organization is able to evaluate, use or even extend it. In Table 14 we present the facilities that each evaluated engine provides to support the logging of execution data.

As it can be seen, most of the engines innately support the logging of the start and completion times for an activity (including the ones invoking services), but the enabled times are not widely supported. For engines where the enabled time is absent we estimate these times on the basis of the complete times of the immediately-preceding activities in the BP model as described in Section 3.1.2. All BP engines support the identification of BPs and BP cases, along with the identification of activities and activity instances, each one of which is necessary for the calculation of the rest of the execution measures. Most of the BP engines also register the end state of the BP case and the user carrying out the activities. We have selected three BP engines for the execution of BPs in the languages of BPMN2, XPDL, and WS-BPEL: Activiti,[5] Bonita[6] and Intalio[7] community edition respectively. One objective of our proposal is to integrate as much as possible existing technology and to only develop tools when existing technologies not sufficiently to meet our requirements. We have done several interoperability tests, transforming event logs containing the data they register, into the MXML event log format of the ProM framework. To do so, we use the ProMimport[8] framework and Fluxicon[9] tools. These make it possible to transform the exported event logs, in .cvs format, into the MXML format; the logs are thus successfully loaded and visualized in the ProM framework. For service execution measures we need another form for the registration of event logs, as services will be executed generally on an application server, or similar technology. We can thus filter the logs of the server, to obtain the data logs from the service execution, or we may implement a specific measurement module to register the base measures that we need to calculate the rest of the measures, as proposed in [62]. Nevertheless, we can register in the BP engines the invocation and answering time of the service, as start and complete times of the activity invoking the service.

4.3. Calculation and visualization of execution measures

The ProM framework provides several plug-ins to extract information from the real execution of the BP, focusing mainly on three aspects: the discovery of non-existing BP models, the comparison of existing BP models with their real execution, and the enhancement of existing models with information about their real execution. For the analysis of BP execution measures from an operational point of view, few plug-ins for the making of organizational decisions exist. For example, as regards performance analysis: there is a plug-in that provides a basic performance analysis with selected performance measures, and another one which is based on a Petri Net model of the BP and which also provides selected performance measure calculations. We believe that a plug-in providing a set of selected execution measures as the ones we defined in BPEMM, could be an interesting addition to the ProM framework. We defined therefore two main objectives for the plug-in: (i) to contribute to the ProM community with the provision of a novel plug-in for the analysis of BP execution; this is a plug-in that has been extended with service execution measures and provides a business view, and ii) to bring the ProM framework to the business area as a key tool for analyzing several aspects of the BP execution, with particular focus on decision making. Based on the BPEMM execution measures, we may detect improvement opportunities, both for the BP model and its implementation by services, thus providing insight into the real execution of the BPs with respect to the goals of the organization and the BPs analyzed.

So far, we have implemented a prototype of the ProM BPEMM plug-in, providing the execution measures defined for the Time dimension of the Generic BP view that only covers performance measures. As input, it requires a BPMN2 model of the BP, and a configuration file with information of the context data to calculate the rest of the execution measures defined. In Fig. 11 a screenshot of our ProM BPEMM plug-in is shown calculating the Generic View measure throughput time for all BP cases for the execution of the BP that is described in Section 5. We present the Throughput Time (TT) for each BP case in graphical form on the right, also showing the Total Working Time and Total Waiting Time. On the left, the results of the indicators are shown, along with a colored traffic light showing the associated meaning (Green = OK, Yellow = Warning, Red = Problems). We obtain a view on each BP case execution by changing the Measure level to each BP case, where for each activity executed in the BP case, we see the Total execution time, along with the Activity Working Time and Activity Waiting Time. We provide another view for each activity by taking into account all the BP cases in which it has been executed; this allows us to analyze the summarized behaviour of the activity, showing the defined indicators on the left, and the specific information about each BP case where it has been executed. We will show these in Section 5.

5. Case study

In this section we present the validation of the proposals presented in Sections 2, 3 and 4, which correspond to the results of the research we have carried out. We answer specifically the research question RQ4, presented in Section 1. To assess the applicability of the execution measurement and improvement activities defined in BPCIP, the execution measures integrated in BPEMM, as well as the tool support provided, we carried out a case study

[5] http://activiti.org/
[6] http://www.bonitasoft.com/
[7] http://bpms.intalio.com/community
[8] http://www.promtools.org/promimport/
[9] http://fluxicon.com/

Fig. 11. ProM BPEMM plug-in example for BP cases option.

using a real BP from the Hospital General de Ciudad Real (HGCR), with whom we were collaborating. As mentioned in Section 1, the HGCR started a project with the UCLM in 2007, which included the modeling of several BPs with BPMN, the definition and evaluation of characteristics to select a BPMS, and the implementation and deployment of the BPs modeled, as well as guidance for the management and improvement of BPs. We carried out the case study following the definitions in [66], in the context of this project with the quality group of the HGCR, selecting a representative BP process from the ones already modeled in the past.

As the infrastructure and the IT area of the HGCR were not available for participation in the research as we had established initially, we decided to overcome this problem by splitting the case study into two sub-parts, based on the possibilities we had at that particular moment in time:

- Part 1: on the one hand, we decided to set up a pre-production environment, not in the organization but in our own laboratory, by evaluating the process engines for BPMN2, XPDL and WS-BPEL, presented in Table 14. To do that, we implemented and executed the BP model from the HGCR in the process engines selected (Activiti, Bonita and Intalio respectively), registering data from the execution of BP cases and exporting it for loading in the ProM framework.
- Part 2: on the other hand, the IT people did not execute the prototype in the context of the HGCR itself, but rather in our laboratory. We were therefore not able to execute a considerable number of the BP cases to gather the necessary data. This led us to decide to simulate the BP model, to make it possible to gather enough data by using CPNTools[10] in conjunction with information given to us from the HGCR Quality group and experts on the BP. CPNTools allowed us to register data from the simulation in MXML format for loading into the ProM plug-in which we have developed to analyze the measurement results.

The definition, execution, data collection and analysis of the case study is presented in the following sections, based on the guides in [66–68]: case study design (cf. Section 5.1), case selection (cf. Section 5.2), procedures and roles (cf. Section 5.3), application of the proposal (cf. Section 5.4), data collection and analysis (cf. Section 5.5), validity threats (cf. Section 5.6) and reflections and lessons learned (cf. Section 5.7).

5.1. Case study design

The type of case study we carried out was a single-case, in a single organization and in a single project of the organization; this corresponds to a holistic case [66]. The object of the study was the execution measurement, analysis and improvement of a BP in the context of the hospital and the pre-production environment (laboratory) defined; the unit of analysis was the quality group of the HGCR.

The main research question for this case study was stated as "Does MINERVA provide, by means of BPCIP and BPEMM, a useful proposal for carrying out BP continuous improvement based on execution measurement of BPs in organizations?" with the aim of validating the complete proposal corresponding to a rephrase of RQ4. Additional research questions derived from this to validate each specific part of the proposal corresponding to RQ1, RQ2 and RQ3 were defined respectively as: (1) are the execution measurement and improvement activities proposed in BPCIP appropriate and useful for business people in the management and improvement of BPs in the organization?, (2) are the execution measures integrated in BPEMM appropriate and useful for business people to be able to obtain information on the execution of the BP?, and (3) is the tool support provided to implement, execute and analyze the execution measurement results appropriate and useful for the software team and for business people, respectively?

5.2. Case selection

As case we selected the Patient Major Ambulatory Surgery (MAS) process from the HGCR for several reasons: first of all, it involves several organizational units in the organization and had been modeled in BPMN in a previous stage of the project. That meant that we could use the existing model. Secondly, there were several indicators defined for the BP, so they could be compared with the execution measures integrated in BPEMM; and thirdly, the person responsible for the quality group who was also the responsible for the BP, was interested in implementing this partic-

[10] CPN Group, http://cpntools.org/start.

ular BP in a BP engine. The BP involves the following participants: Patient, MAS Unit – which in turn is made up of these participants: Secretary, Nurse and Auxiliary Nurse – and the Surgical Block, which in turn is composed of the following participants: Surgeon, Anaesthetist, Instrument Nurse, Operation Room Attendant and Operation Room Auxiliary. The first sub-process in the HGCR is "Admission and Registration", in which the patient goes to the hospital to have the surgery. Secondly, we have the "Preparation for MAS", in which the patient is given the clothes and the particular place in the list for surgery. After this the "Pre-intervention" and the "Intervention" (the surgery) are performed, followed by the "Post-intervention", the "Observation and recovery" and finally the "Discharge Patient". In Fig. 12, we present a global view of the Patient MAS, as modeled by the HGCR staff. A more detailed version including all the defined sub-processes (in Spanish only), is available online (http://161.67.140.34:82/cma/).

5.3. Procedures and roles

The main roles participating in the case study were the responsible for the BP played by the person responsible for the quality group of the HGCR, and the responsible for the improvement played by this author. The Quality group provided the Patient MAS BP model and the "Admission and Registration" and "Preparation for MAS" sub-processes were implemented as in the pre-production environment for the BP engines Activiti and Bonita, and the "Pre-intervention" sub-process in Intalio. We attempted to cover different ways to model BPs with the selected languages and engines. For the simulation, we modeled an adapted version of the first two sub-processes mentioned above. This is because most of the activities in the original sub-processes are manual, so we defined several service activities between pools. We show the adapted Patient MAS BP in Fig. 13. The indicators already defined for the Patient MAS BP were also given, so they could be compared with the ones integrated in BPEMM, for evaluation by the quality group.

Several executions of the BP were performed in each process engine in the laboratory, registering the corresponding execution data, which we extracted in .csv format and transformed into MXML format, as mentioned before. We did this to assess how feasible it was to load the BP execution data into the ProM framework for further analysis. As mentioned at the beginning of this section, BP executions were not real and there were too few of them, so we conducted a simulation of the BP in CPNTools. To do that, we obtained data regarding the necessary parameters, such as duration of activities and performers, number of surgeries performed each day, and arrival of patients at the Hospital, from the quality group of the HGCR, specifically from the person for the quality group which acts as the owner of the process. For each activity we set the estimated time of execution in the activity definition, we defined the function for the generation of BP cases at the beginning of the BP as an expression of the "number of patients in a given hour", and we simulated the execution of a thousand BP cases. CPNTools allowed the execution logs in MXML format to be registered, one for each BP case. By means of the ProMImport framework, we merged these execution logs into one that would be loaded into the ProM framework for further analysis. To do this, we used the prototype of the ProM BPEMM plug-in (among other functionalities of ProM), enabling us to find improvement opportunities. We integrated these into the BP, generating a new version of the BP. We simulated this new one once more, to compare and to assess whether the improvements integrated allowed us to achieve the desired goals. The research group acted as a consultant on MINERVA, leading the implementation of the BP in the pre-production environment (laboratory) and running the simulation of the BP in CPNTools for both versions of the BP. When we completed the evaluation and the simulation, we carried out an interview with the person responsible for the quality group, who was also the responsible for the BP, presenting the results for him to give an expert's opinion on how he perceived the usefulness of the proposal.

5.4. Application of the proposal

The execution of BPCIP activities is described below, as presented in Section 2. Since the model was already specified, the activities from the Business Modeling Discipline in the Design & Analysis phase were not executed as defined. We executed them only to specify the chosen sub-processes from the BP model in each of the process engines; and we adapted one as Petri Net for the simulation. We evaluated the activity EM1 – *Select execution measures*, in the meeting with the person responsible for the quality group of the HGCR, based on the discussion of the indicators they already had, together with the execution measures proposed. As the ProM plug-in implements only the Throughput Time (TT) measures for now, these were the only ones we could assess. The BP execution Average TT goal was defined to be under 90 min, the Warning rank between 90 and 120 min and the Problems rank to be over 120 min, for the sub-processes "Admission and Registration" and "Preparation for MAS", which were the ones modeled for simulation.

The Configuration phase corresponds to the implementation we carried out in the pre-production environment, for which the first two sub-processes of "Admission and Registration" and "Preparation for MAS" were modeled for Activiti and Bonita. We modeled the "Pre-intervention" sub-process in Intalio. This was because the main objective for the implementation was to assess the feasibility of executing the Patient MAS BP in different process engines and languages, registering the execution and extracting the execution data to be loaded into the ProM framework. We do not present the implementation in the BP engines here, as the most important result obtained is the assessment of the execution and registration of execution data to be loaded in the ProM framework. Activity I1 – *Implement BP with services* – was performed to make the BP model executable.

We divided the Execution phase in two parts. In the first one, we executed the BP model implemented previously on each BP engine; the execution data for each BP instance was then registered, as defined by activities EM2 – *Implement collection of execution measures* and EM3 – *Collect execution measures*. Executing the BP in three different engines we could demonstrate that it is possible to obtain the execution data from the engines and to load it into ProM. In the second one, the BP model was specified as a Petri Net in the CPNTools, and we added to this the information for the parameters, such as duration of activities and resources, as defined by business people.

We performed the simulation of the Patient MAS in CPNTools so that we could obtain a considerable amount of execution data for the BP enabling the further analysis by means of the ProM BPEMM plug-in prototype This enabled us to calculate and visualize several of the chosen measures. The BP was modeled as Petri Net and enhanced with data that we had collected previously to carry out the simulation from the real execution of the BP in the hospital.

To model the Petri Net, we followed the proposals in [69,70], by defining a hierarchical Petri Net. This provides a global view of the collaboration between the participants in the BP, but logs only the execution corresponding to the Hospital and the realization of the BP by services. In Fig. 14, the global view of the Petri Net corresponding to the Patient MAS is shown, with the participants in the collaborative BP and their interaction.

We modeled each pool in the BP as a substitution transition, which in turn we modeled in a subpage of the hierarchy defined. The hospital subpage models the Petri Net corresponding to the

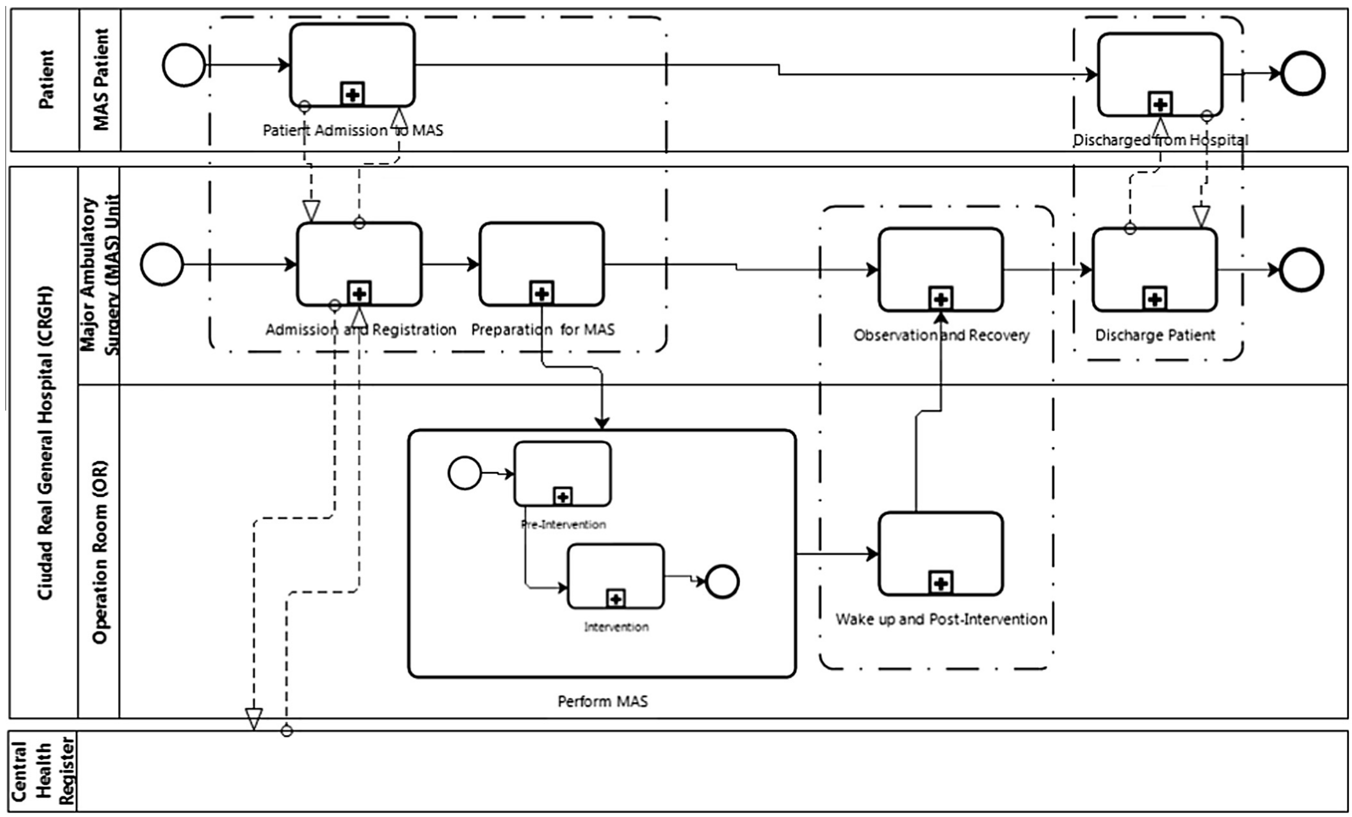

Fig. 12. Complete Patient MAS BP from HGCR.

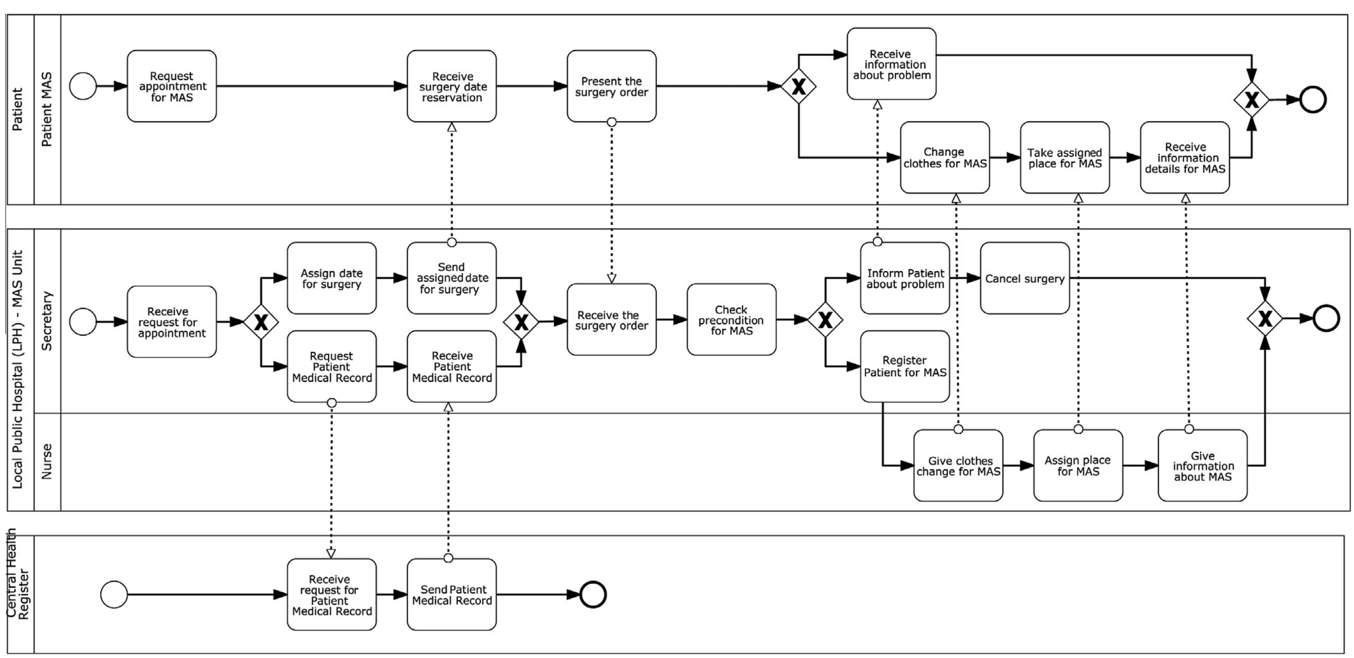

Fig. 13. Patient MAS "Patient Admission and Registration" and "Preparation for MAS" sub-processes adapted for simulation.

hospital and on this sub-page each activity is also modeled as a substitution transition. This allows us to log the set of three times for the execution of the activities in the BP: enabled, start and complete times in each subpage.

Fig. 15 displays the subpage for the BP model, showing the transition substitution approach for modeling each activity as a substitution transition, detailed on its own page, on which we can log the time points defined for each activity. The resource modeling approach we used is based on the one proposed in [69]. The aim was to simulate the realization of each activity by a person assigned to the roles of the BP. This became a centralized place that was shared between all the pages, to simulate the availability of resources for carrying out the different activities. We have added the "system" resource for executing the automated activities realized

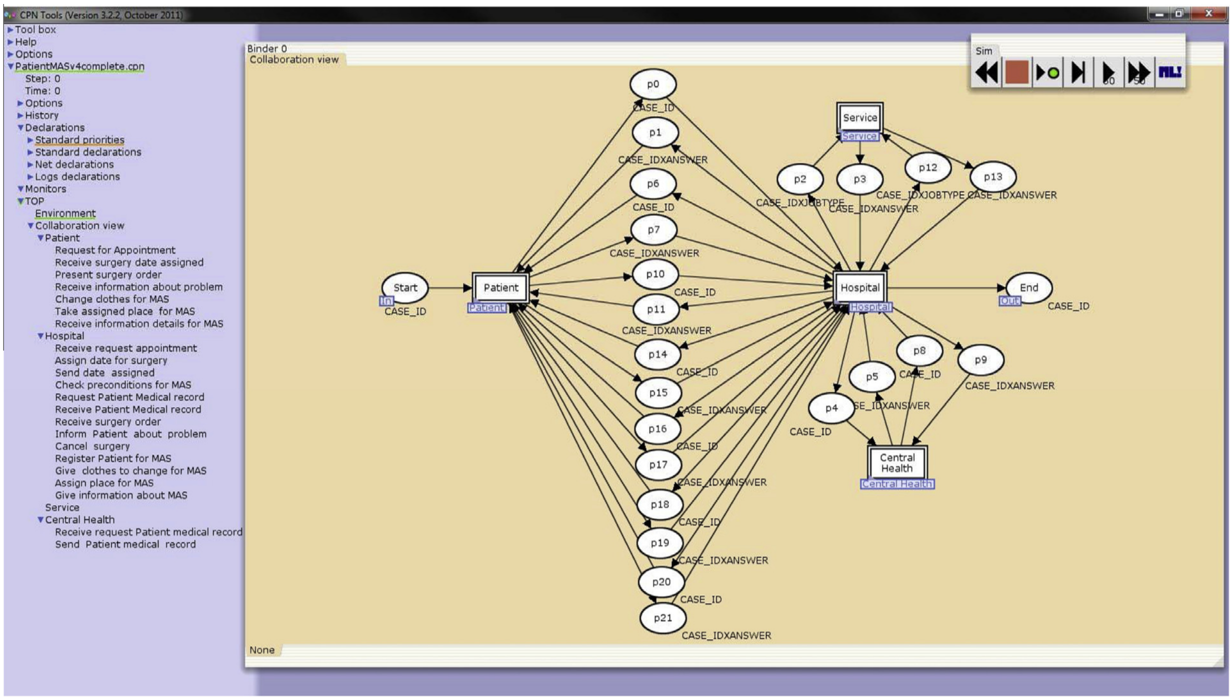

Fig. 14. Global view of the Patient MAS Petri Net defined in CPNTools.

by services, and the "service" resource for executing services. As the resources were modeled in a centralized (fusion) place, shared by all activities in the BP, the availability of resources was simulated in a more realistic way, since we divided it between BP cases and activity instances.

In Fig. 16 the resource modeling approach is shown for the "Check preconditions for MAS" activity, as an example of what was modeled for each activity defined in the BP.

The centralized (fusion) place allows resources to execute each activity when they become available, and to be returned to the central place once the execution of the activity has finished. We have shown the definition of the three execution times to be logged for each activity in the BP model, along with the inscriptions needed for the ProM log generation in the MXML format. A more complete approach for resource modeling is proposed in [70] but we decided not to use it for this simulation, as it introduces more complexity into the modeling of the BP. For service execution modeling we used a queuing approach for each service executing in the Server, to emulate the real execution of several services in the same infrastructure. We define the service resources

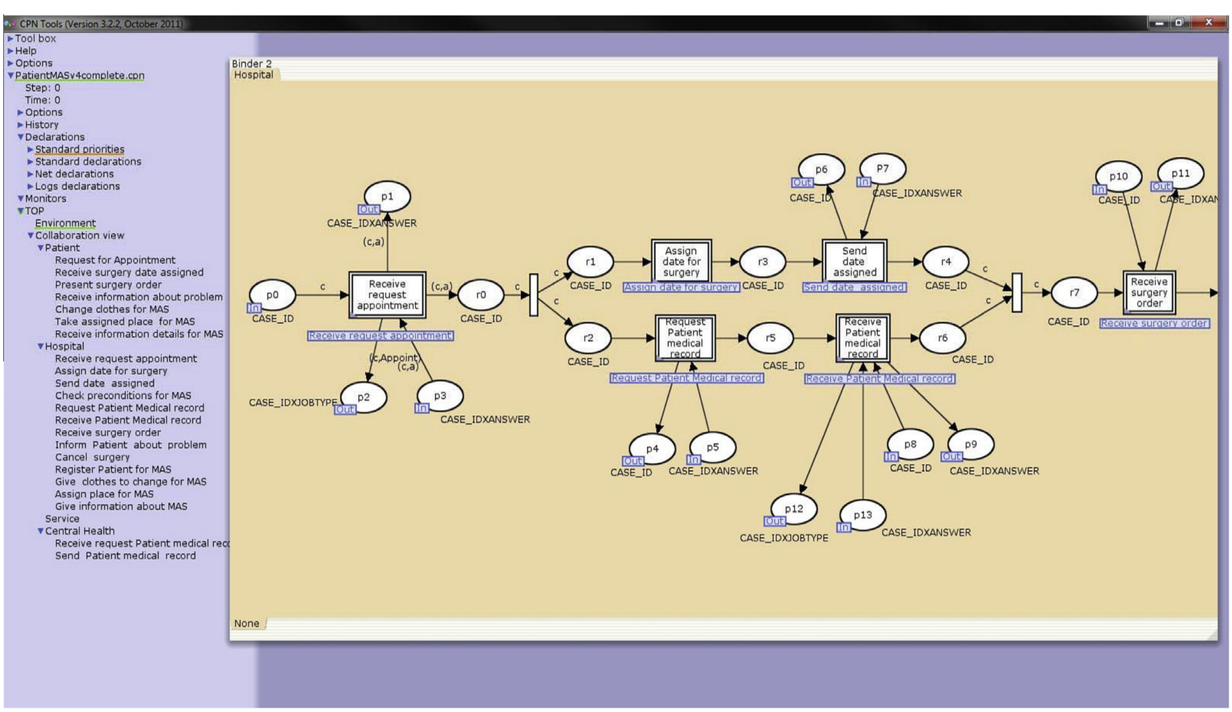

Fig. 15. Hospital sub-page showing the modeling of the transition substitution.

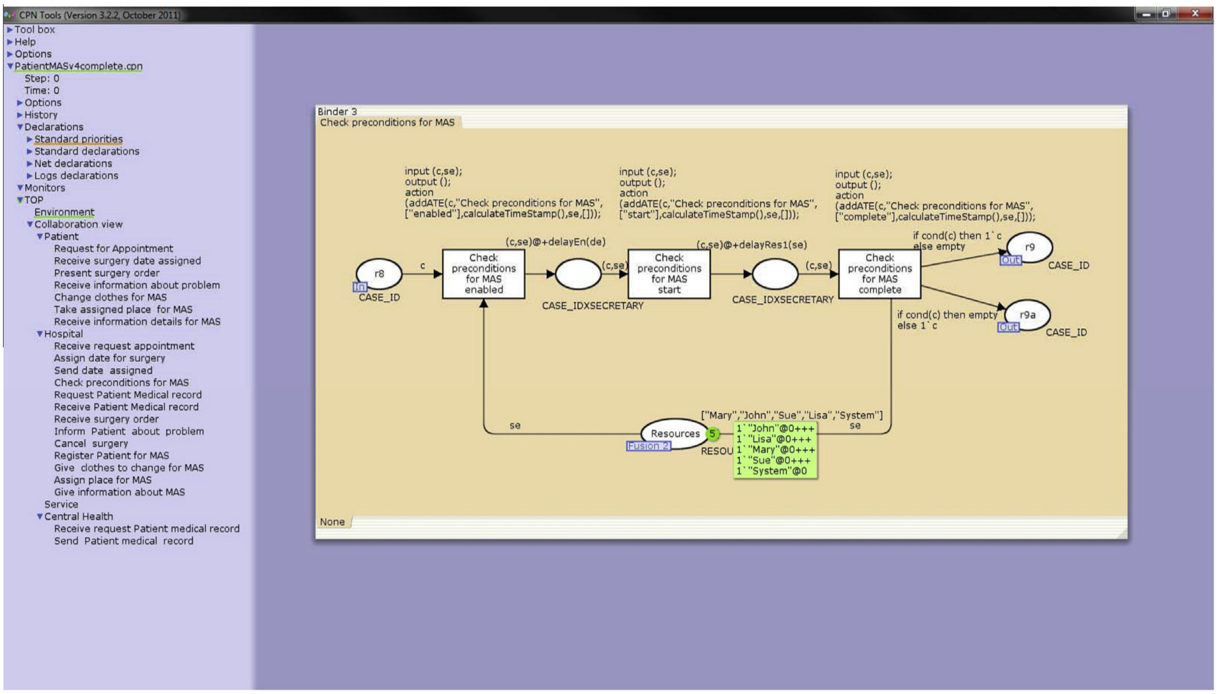

Fig. 16. Resource modeling example for activity ''Check preconditions for MAS''.

to deal with the execution of the services invoked, running simulations with several different configurations for the delay in each queue and for the maximum of jobs to be enqueued at each time, as well as for the time in processing each service invoked, and the result of the execution. Fig. 17 presents the service subpage, showing the modeling of the two services offered by the HGCR. The steps carried out for the simulation preparation correspond to the implementation of the measures in the process engine in the Configuration phase. They include the ProM log inscriptions to generate the event logs for calculating the execution measures. Specific aspects of Petri Nets and CPN Tools modeling, simulation and ProM event logs can be seen in [71–73].

In the Evaluation phase, once we obtained the event logs corresponding to the simulated execution, we could load them into the BPEMM ProM plug-in, to calculate and visualize the execution measures defined. As the CPN Tool created a log for each case of the execution of the simulation, we used the ProMimport framework to merge the logs. In doing so we generate one MXML file

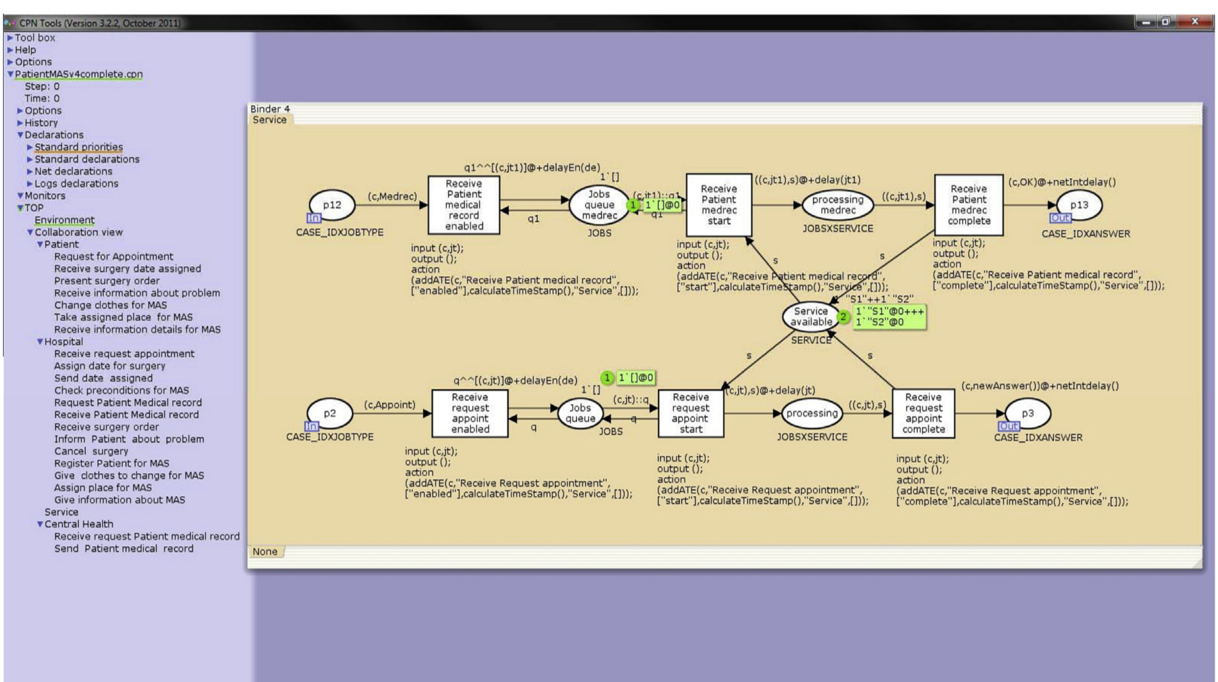

Fig. 17. Service sub-page with a queuing approach for the HGCR service simulation.

Table 15
Example of event log generated from the CPN Tools simulation.

Case id	Activity id	Event type	Timestamp	Originator
1	Receive request for appointment	enabled	01-08-2011:21:06	System
1	Receive request for appointment	start	01-08-2011:21:10	System
1	Receive request for appointment	enabled	01-08-2011:21:12	Service
1	Receive request for appointment	start	01-08-2011:21:15	Service
1	Receive request for appointment	complete	01-08-2011:21:18	Service
1	Receive request for appointment	complete	01-08-2011:21:20	System
2	Receive request for appointment	enabled	01-08-2011:21:20	System
2	Receive request for appointment	start	01-08-2011:21:24	System
2	Receive request for appointment	enabled	01-08-2011:21:26	Service
2	Receive request for appointment	start	01-08-2011:21:30	Service
2	Receive request for appointment	complete	01-08-2011:21:36	Service
2	Receive request for appointment	complete	01-08-2011:21:38	System
2	Request patient medical record	enabled	01-08-2011:21:40	System
2	Request patient medical record	start	01-08-2011:21:45	System
2	Request patient medical record	complete	01-08-2011:21:49	System
1	Assign date for surgery	enabled	01-08-2011:21:50	Juan
1	Assign date for surgery	start	01-08-2011:21:51	Juan
1	Assign date for surgery	complete	01-08-2011:21:59	Juan

containing the execution of the total BP cases run in the simulation, of which there were a thousand. Table 15 shows an example of an event log from the CPN Tools simulation.

As the BP was not deployed in the organization business people could not perform many of the proposed activities themselves. We therefore provide an example here of how the activities should be carried out to find improvements for the BP, integrating these to generate a new version of the BP that will be executed again. We can thus compare the new execution results with the ones from the previous version. Recalling that the BP execution Average Throughput Time (TT) goal was set by business people to be under 90 min, the Warning rank between 90 and 120 min and the Problems rank to be over 120 min, these times should be analyzed to see whether they are in accordance with the definitions or not. This analysis corresponds to the execution of the EM4 – *Analyze execution measurement results* activity. The event log loaded in the BPEMM ProM plug-into calculate and visualize the selected time measures is the one shown in Fig. 11 in Section 4.3, for all BP cases executed with average times, so we do not show it here again.

It can be seen that the measure for the Average TT for all the BP cases is above the goal set; the time was found to be 126,27 min. Fig. 18 displays the times for each BP case (selected BP case 999), as well as each activity executed in the case, with total times and percentage times as defined by the indicators. It can be seen that the TT for this particular case is above the goal that had been established, and the waiting time for the Activity "Assign date for Surgery" is almost twice its working time. It would be interesting then to drill down through all BP cases to the times for the Activity, to analyze the execution of the activity in all of them. We would find that in most cases the waiting time in each one is greater than, or equal to, its working time, as shown in Fig. 19.

We thus found an improvement opportunity referring to the definition of the activity "Assign date for surgery". When analyzing the BP, it was found that this activity was performed manually by the individuals assigned to the Secretary role. They have to take the activity from their work list and assign a suitable day and hour to carry out the surgery, by looking in the calendar for available surgery slots. This is not the only activity in which

Fig. 18. ProM BPEMM plug-in time measures for each BP case and its activities.

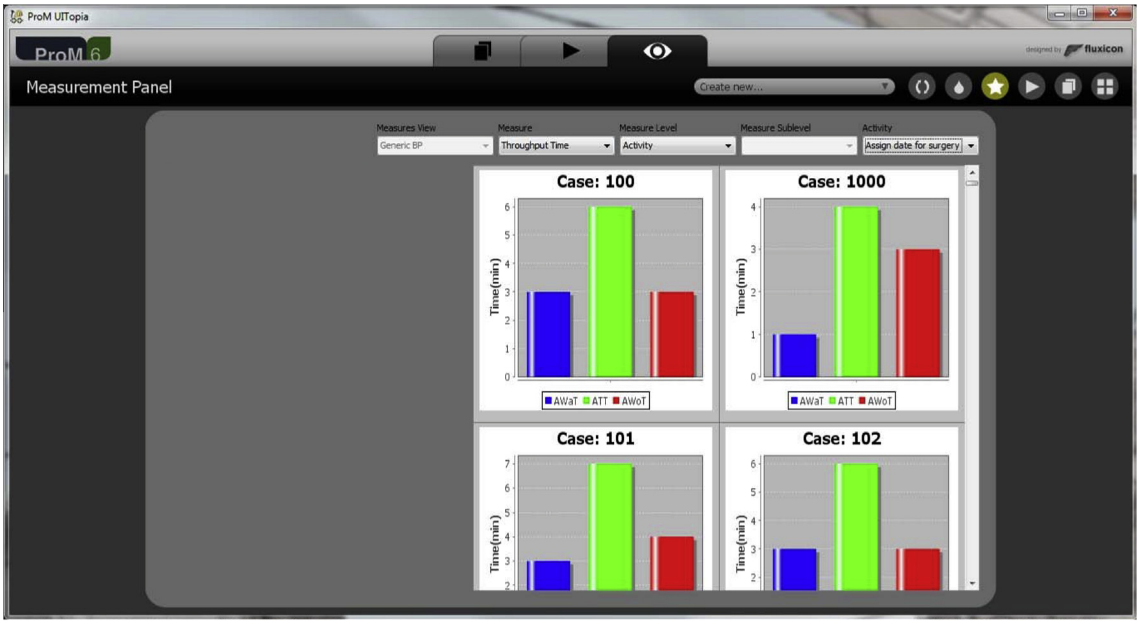

Fig. 19. ProM BPEMM plug-in time measures for the activity "Assign date for Surgery".

these people participate, so a long time may pass between one action and another.

Once we found the improvement opportunities, the improvement activities were executed in the Evaluation phase. This was to define and plan the modification of the BP in the next BP lifecycle execution. It started with the BP model, to see if there was a suitable redesign that allowed being a change in the activity and/or related activities. In the Define improvements activity, the improvement we wished to incorporate into the activity "Assign date for surgery" was specified, its goal being to lower the activity waiting time to fifty percent of its current length. The Diagnose BP activity was not executed, as the BP we were dealing with was not the original one from the organization, and the real value of this activity was to detect organizational improvements for the definition of the BP. In the Formulate improvements activity we specified the need to evaluate several redesign alternatives for the activity and related ones.

After the activities and associated documentation to support the improvement effort were executed in the Evaluation phase, we conducted the Design & Analysis phase again, to evaluate the redesigns for the BP model in the activity BM3 – Redesign BPs, as shown in Fig. 20. From the possible redesign heuristics in [37], the Task Composition (COMPOS) and Task Automation (AUTO) ones were combined, obtaining one automated activity to assign and send the surgery date.

In the Configuration phase we modified the new version of the BP to support the changes made. In this case study we modeled another service in the CPNTools Petri Net, to perform the assignation of the date for the surgery, as well as to send the information to the patient. In addition, we adjusted the measure implementation to register the data needed from the new BP version, adding the corresponding data for the changed activity and the new service.

In the Execution phase, we simulated the new version again (in a real environment it would be executed once more in the process engine), generating the event logs. In the Evaluation phase the event log with the information of the simulation of the new BP version was loaded in the BPEMM ProM plug-in, to calculate and visualize the chosen measures. In a real setting the two BP versions would be compared to see whether the introduction of the improvement had led to an actual improvement in the execution

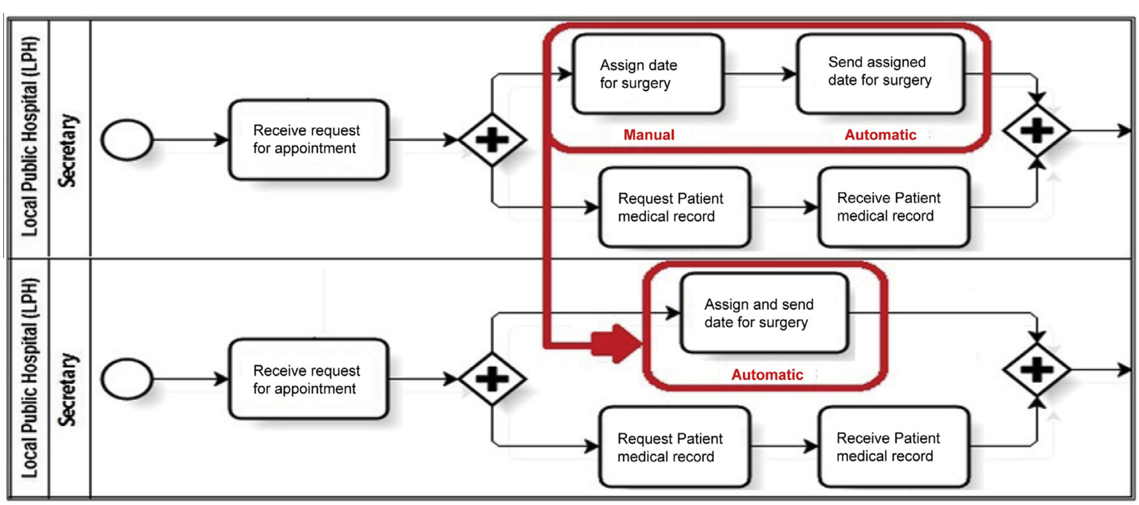

Fig. 20. Redesign options for the activity "Assign surgery date".

of the BP. Since our executions are based on simulation, however, such a comparison is little meaningful. We therefore focussed our evaluation on the assessment of feasibility of the BPCIP proposal (see Section 5.5).

5.5. Data collection and analysis

We performed the collection of data from the execution of the Patient MAS BP in the BP engines which was carried out in the pre-operation environment (laboratory). We also used the answers in the interview with the person responsible for the quality group, who was at the same time the Process Owner. The data of the case study carried out in the BP engines was registered during the execution and is available in the corresponding internal reports (in Spanish only). We recorded the answers from the interview as it took place. In the interview the results of the simulation of the BP were shown to the person in charge, along with the execution measures applied. We also showed him our suggestions for the improvement of the BP, along with the new results obtained.

The interview with the person responsible for the quality group revealed that they evaluated these measures positively, as they represented and extended the indicators they already had for the BP. By way of example, one of the main indicators they are interested in is the occupation of the Operation Room (OR). This is covered by the execution measure M9 (derived) for the Capacity utilization for a resource in the Time dimension of the BP Generic view, as presented in the Section 3.2. The complete lifecycle of BPCIP was discussed and the simulation results for both versions were presented in the ProM plug-in. We did this to show what the complete cycle would be like (although the ProM plug-in is a prototype, and at present only provides the measures for Throughput Time (TT) of the BP from the Time dimension of the Generic view). It was positively perceived and seen as feasible for integration in the organization. We will assess this integration in a future case study, when the IT area of the HGCR is able to participate.

As for the evaluation of the ProM plug-in carried out in the laboratory using the data gathered in the simulation, what we wanted to assess was the feasibility of implementing the BPEMM measures as a ProM plug-in analyzing the results in the three dimensions defined by the BPEMM cube, as presented in the Section 3.2 All of this was achieved, and with positive results.

5.6. Validity threats

The validity threats that could affect the case study were analyzed as part of the planning and we describe these below:

Construct validity:

- The questions asked in the interview to assess the case study were based on the research questions defined; we determined the topics to which the questions referred on the basis of desirable characteristics that a framework such as MINERVA should provide; they include guidelines, execution measurement and improvement activities, as well as tool support.
- We established the characteristics assessed for the execution of the BP selected in the process engines chosen on the basis of desirable characteristics which these tools would provide, as defined in many existing evaluations and guides.

External validity

- The organization in which we carried out the case study presents several characteristics of organizations that would be interested in applying MINERVA. For example, there are several BPs to be executed in a BPMS, from which we were able to gather execution data and analyze it to find improvement opportunities. The hospital already has a quality group, which is a key factor for improvement efforts.
- Although the interview was with only one person, who is the person responsible for the quality group who was at the same time responsible for the BP, we can rightfully consider his answers as expert opinion on the subject, as he leads quality and improvement efforts in the organization. Nevertheless, we are aware that the opinion of only one person does not allow us to affirm anything more than that the BPCIP proposal could be useful in an organization such as that particular one. This prevents us from generalizing the results presented until it is possible to carry out another case study to confirm the trends perceived.
- The fact that the IT area could not participate in the implementation of the prototype of the selected BP, which meant that we had to perform a simulation instead, is also a threat. When the IT area is able to participate the BP should be actually implemented and executed within the organization.

Reliability

- This threat is concerned with to what extent the data and the analysis are dependent on the specific researcher. Threats of this type of validity could occur if it is not clear how to code collected data, or if questionnaires or interview questions are unclear. We took into all these considerations into account when defining the questions and the scale for the answers.

5.7. Reflection and lessons learned

The implementation and execution performed in the three process engines chosen in the pre-production environment defined, together with the simulation of CPNTools, allow us to draw some conclusions. The first one is about the Configuration and Execution phase activities and artifacts defined. We found that it is feasible to put them into operation in organizations for the implementation and execution of their BPs. We can also affirm that the tool support is validated, as the three process engines allow the implementation and execution of the Patient MAS BP, as defined by the HGCR. This includes the registration of execution data and the extraction of this data for loading into the ProM framework, and the ProM BPEMM plug-into perform the analysis of the BP execution. On the basis of the analysis of the answers from the interview with the person responsible for the quality group and for the BP, the proposals in BPCIP can feasibly be incorporated in organizations wanting to manage and improve their BPs. Moreover, the execution measures provided by BPEMM in the ProM plug-in prototype will allow us to calculate and visualize several existing indicators for the BP and also provide more information for the analysis of BP execution.

As regards lessons learned, this case study showed us that it takes time and effort to set up the environment for BP execution, at least with the process engines selected. Once everything is in place, however, the implementation and execution of BPs can be done with successful results. Although our proposal can be realized entirely with open source software, we are aware that several commercial tools exist which might also provide many of the functionalities required to support the BPs lifecycle. We assessed and successfully carried out the interoperability from BP engines with the ProM framework, by means of the chain of outputs/inputs from the .csv file extracted from the process engine. We transformed this with ProMImport and/or Fluxicon into MXML format and loaded it in the ProM framework. One issue detected, which needs to be improved, is the interoperability from BP modeling tools to BP engines. The ProM BPEMM plug-in, although a prototype for now, allows us to analyze the execution measurement results from the BP execution, presenting information for all the BP cases executed, each BP case executed and the corresponding activities, as well as for each activity in all BP cases. This ensures that we have the different levels of granularity that have been defined to provide the insight into BP execution that we are seeking.

6. Related work

In this section we present the related work we found in the literature review carried out at the beginning of our research work. We took the existing improvement and measurement approaches into account, as well as the execution measures we found. These were incorporated, and they formed the basis for the definitions we produced for our proposals, as presented in Section 2 and 3.

Several improvement initiatives have been defined for organizations wanting to improve their BPs in general, and also software processes in software enterprises [22]. They include such initiatives as TQM (Total Quality Management), BPR (Business Process Reengineering), Six Sigma, CMMI (Capability Maturity Model Integration, CMM, and BPMM (Business Process Maturity Model). TQM and Six Sigma focus on an analysis of business processes with different techniques, in an attempt to find and incorporate improvement opportunities. They are based on an improvement cycle (PDCA, DMAIC), and are used by many organizations. Other life-cycle models such as IDEAL were also developed specifically for software process improvement that is based upon the CMMI; they are also widely-used by software organizations. In [23–26], several aspects of the use of Six sigma, TQM, CMMI, IDEAL and other improvement models and standards, as well as results obtained from their application, are presented and discussed. The BPMM [5] is another improvement model that is based on CMM and CMMI. We have integrated it in our proposal as part of the BPCIP improvement process, using it as a guide for the evaluation of the definition of BPs in the organization, as mentioned in Section 2. BP execution measurement is an integral part of any of these improvement models, as it is the basis used for gathering data to assess the real operation of the organization.

Existing proposals have treated several aspects of BP execution measurement, such as those in van der Aalst et al. [19], Laguna et al. [22], Reijers [37], Netjes [38], Maruster et al. [39], zur Muehlen [54] and Wynn et al. [74], where concepts and measures for BP execution are presented and analyzed. We based our definitions on these pieces of work and have integrated several execution measures defined in them, as they provide different and complementary views of the key aspects to be taken into account for BP execution. Other proposals to align measures with organization goals also exist, such as Balance Scorecard [75], which is similar to the GQM technique we used because we found it the most suitable one for our approach. A comparison between BSC and GQM is provided in [76].

In Laguna et al. [22] and Reijers [37], analytical techniques are used to analyze and predict BP performance and other characteristics that have to be evaluated, such as cost of the BP or use of resources and detection of bottlenecks presenting definitions and formulae. Reijers [37], for his part, proposes a set of heuristics for the redesign of BPs and these are also used in Netjes [38]. In Laguna et al. [22], Netjes [38], Maruster et al. [39] and Wynn et al. [74], simulation techniques are used to evaluate different aspects of BP models and to compare redesign options. To do this they use predictive calculus or existing execution event logs. Netjes [38] presents a specific plug-in for the ProM framework, designed to evaluate the redesign options for BP. In [74], the use of existing data in event logs to simulate BP execution and to provide predictions for future execution is proposed. Those predictions are based on the previously-known data. Zur Muehlen [54] and Casati et al. [77] suggest the use of data warehouses to store, analyze and evaluate BP execution and associated metrics. In zur Muehlen [54], a lifecycle for activities and BP cases is also put forward as a proposal. They include a model for logging relevant information of the BP execution, as well as a tool that supports the approach. Van der Aalst [19] describes the analysis of event logs by applying Process Mining techniques, using the ProM framework for process discovery, conformance checking and extension of models with execution information. The existing plug-ins enable us to mine the event logs to discover information about the BP execution, such as the Petri Net model of the BP, or the organizational view.

Several tools to support these approaches exist, such as ARIS [78], which provides a Process Performance Manager to gain insight into performance and other measures, or the ProM framework we have already mentioned. There are two specific ProM plug-ins for analyzing BP performance that we have evaluated, the "Petri Net performance analysis" plug-in and the "Basic performance analysis" plug-in. Both of these plug-ins calculate performance measures similar to the ones we have defined for the Time dimension of the Generic BP view, mostly regarding duration and other times for the BP execution. To identify paths and to calculate the enabled time of activities when this is not present in the event log, the former plug-in requires the model of the BP, but in Petri Net notation instead of BPMN2 as ours does.

Our approach goes beyond the existing state-of-the art by extending the BP lifecycle with explicit measurement and improvement activities. We also extended the analysis of BP execution by means of execution measures for the dimensions presented, adding views such as the service execution view. This latter view also analyzes the technical aspect of BP execution. These features will help to provide more insight into the detection of improvement opportunities from BP execution. There are several other plug-ins that exist in ProM which permit other aspects of BP execution to be analyzed. None of them, however, provides in a single plug-in all the measures to perform a global analysis of the BP execution against business goals, as ours does.

On comparing our proposals with the previously mentioned ones, we see that the activities we proposed in the BPCIP improvement process, which are based on [18], constitute a core of key improvement activities. Compared to other proposals, such as IDEAL, PDCA, and DMAIC it entails fewer, but key activities. This makes our proposal simple enough to be easily understood and used by both business and IT people. To the best of our knowledge, the BPEMM execution measurement model is the first model that provides a structured way of classifying and organizing BP execution measures in a consistent way, around the three views defined. It enables a more comprehensive and integrated vision of the real execution of BPs. This, in turn, will be more directly related to the business goals of the organization, as well as to the specific goals for the BPs under analysis, by means of the GQM approach used. As we have shown in the assessment in our case study within the HGCR, the model provides the basis needed to manage the improvement effort in BPM implementations within organizations, providing tool support for each activity to be performed. In summary, we believe that our proposal provides an integrated and holistic view of the problem, which makes it an applicable and usable guide for the continuous improvement of BPs in a real organizational context.

7. Conclusions and future work

In this paper, we have presented our proposal for the continuous improvement of BPs realized by services based on execution measurement. We identified the need for research on this subject, given the increasing importance organizations have been putting on BP management and improvement over the last few years [4]. We also detected the practical needs of the Hospital General de Ciudad Real (HGCR), with whom we were working in a collaborative project, and took them on as motivation for our research. To carry out this research we have adopted a pragmatic, philosophical stance and used different research methods to help us gather existing knowledge, understand the problem, develop solutions and validate them within the HGCR. The MINERVA framework we defined in previous work, provides an integrated approach to support the continuous improvement effort in an organization based on the realization of BPs by services with a model driven approach. It can be viewed as an umbrella under which we provide answers to the research questions that guide our research work, including the ones presented here, as both the BPCIP and the BPEMM are part of it, as well as the tool support we provide.

Regarding the need of a systematic approach for guiding the integration of execution and improvement efforts in organizations, we provide with the BPCIP the methodological approach (addressing research question 1). BPCIP comprises explicit activities extending the BP lifecycle [1], with roles and artefacts integrated in several disciplines, which help to guide these efforts. With respect to the issue of selecting the appropriate set of execution measures to provide useful information for the evaluation of BPs real execution, we believe that the ones we selected and provide in the BPEMM execution measurement model constitute a useful guide (addressing research question 2). Using these can save organizations valuable time and efforts regarding the revision of existing literature, relating goals with specific data to be registered, and defining the formulae for their calculation. Finally, we believe that tool support is an essential element of a methodological approach. Based on this, we integrated several existing tools and newly developed ones, to support the definitions we made both in BPCIP and BPEMM (addressing research question 3). To validate the complete proposal (i.e. BPCIP, BPEMM and tools support), we carried out a case study in the context of the HGCR, Spain, including a simulation of BP execution based on knowledge from business experts. The results of the case study showed that our proposal covers the most important needs regarding execution measurement and improvement efforts, when organizations are attempting to carry out a continuous improvement cycle (addressing research question 4).

As with every type of research, our work is subject to limitations. First of all, the case study we carried out involves a single organization, albeit a large and complex one. We therefore believe that further case studies need to be undertaken to improve external validity of our results, which we leave for future work. Another limitation within the case study has to do with the simulation of BP executions we carried out. We tried to mitigate this threat by gathering the input for our simulation from business experts. Other aspects can be taken into account to steer process improvement. These might include structural properties of BP models and simulation of different scenarios prior to the implementation and execution of the modeled BPs. Several proposals already exist that cover these, which are compatible with our definitions and therefore can be integrated into our proposal directly. Furthermore, our proposals focused on a technological context in which BPs are executed in a process engine within a BPMS invoking services implemented from existing, new, or partner systems. However, we believe that it is possible to use the BPCIP activities and the execution measures of the BPEMM model in several other organizational and technological contexts, i.e. BPs executed without BPMS support, or other organizational ecosystems.

Regarding current and future work we aim to improve the technological support for assessing BPEMM execution measurement results, by extending the BPEMM ProM plug-in so that it shows the calculations of all the execution measures of the BPEMM model. In addition, we plan to improve the definition of additional aspects like quality, as perceived by the client and the participants, and to formalize to what extent organizations and their BPs are flexible enough to allow the improvement of BPs in a direct manner.

Acknowledgments

This work has been partially funded by the Agencia Nacional de Investigación e Innovación (ANII, Uruguay), PEGASO/MAGO project (Ministerio de Ciencia e Innovación MICINN, Spain, FEDS FEDER, TIN2009-13718-C02-01) and GEODAS-BC (Ministerio de Economía y Competitividad, MINECO, Spain, TIN2012-37493-C03-01). We also wish to thank the Hospital General de Ciudad Real (HGCR), especially the person responsible for the quality group. We are grateful for his participation and enrolment in the research. Thanks also goes to the student of the University of Innsbruck who developed the ProM BPEMM plug-in prototype. In addition, we would like to thank the editor and the anonymous reviewers for their constructively critical comments on earlier versions of the article, as well as for their very useful feedback on those versions.

Appendix A

See Tables 4–13.

Table 4
Measures for Generic BP execution view & Time dimension for Throughput Time (TT).

Goal	**G1**	**Minimize the Throughput Time (TT) of the BP**
Question	**Q1**	**What is the actual TT of the BP**
Measures	M1 (base)	Enabled time of an Activity (ET)
	M2 (base)	Start time of an Activity (ST)
	M3 (base)	Completion time of an Activity (CT)
	M4 (derived)	Working time of an Activity (AWoT = CT − ST)
	M5 (derived)	Waiting time of an Activity (AWaT = ST − ET)
	M6 (derived)	Total time of an Activity (ATT = AWoT + AWaT)
	M7 (derived)	Total Working time of a BP case (TWoT = ∑ (AWoT))
	M8 (derived)	Total Waiting time of a BP case (TWaT = ∑ (AWaT)
	M9 (derived)	Throughput Time of a BP case (BPTT = ∑ (ATT)) or (TWoT + TWaT) for the corresponding paths
	M10 (indicator)	Activity Working time vs. Activity Waiting time index (ATI = AWaT/AWoT) Decision criteria = Index DC.
	M11 (indicator)	Total BP Working time vs. Total BP Waiting time index (TTI = TWaT/TWoT) Decision criteria = Index DC.
	M12 (indicator)	Percentage of total BP Working time in total BP TT (PWoT = TWoT * 100/BPTT) Decision criteria = Percentage DC.
	M13 (indicator)	Percentage of Total BP Waiting time in Total BP TT (PWaT = TWaT * 100/BPTT) DC = Inverse Percentage DC
	M14 (indicator)	Average BP Throughput Time for all BP cases (ABPTT = ∑BPTT/Total BP cases) DC = Inverse Percentage DC
	M15 (indicator)	Average BP total Working time for all BP cases (ABPTWoT = ∑TWoT/Total BP cases) Decision criteria = Percentage DC
	M16 (indicator)	Average BP total Waiting time for all BP cases (ABPTWaT = ∑TWaT/Total BP cases) DC = Inverse Percentage DC
	M17 (indicator)	Average Activity total Working time for all BP cases (AATWoT = ∑AWoT/Number of BP cases in which the activity was executed) Decision criteria = Inverse Percentage
	M18 (indicator)	Average Activity total Waiting time for all BP cases (AATWaT = ∑AWaT/Number of BP cases in which the activity was executed) Decision criteria = Inverse Percentage DC
	M19 (indicator)	Average Activity total time for all BP cases (AATT = ATT/Number of BP cases in which the activity was executed) Decision criteria = Inverse Percentage DC
Decision Criteria	Index DC	R1: 0 <= TTI <= L1 = ''LOW'' = GREEN; R2: L1 <= TTI < L2 = ''MEDIUM'' = YELLOW; R3: L2 <= TTI = ''HIGH'' = RED
	Percentage DC	R1: 0 <= TTI <= L1 = ''LOW'' = RED; R2: L1 <= TTI < L2 = ''MEDIUM'' = YELLOW; R3: L2 <= TTI = ''HIGH'' = GREEN

Table 5
Measures for Generic BP execution view & Time dimension – Capacity.

Goal	**G2**	**Maximize the capacity of the BP**
Question	**Q1**	**What is the actual capacity of the BP**
Measures	M1 (base)	Number of resources per role defined in the BP = NRRBP (from context data)
	M2 (base)	Number of execution of each activity in all BP cases (NEA = count the times the activity is executed in all BP cases)
	M3 (derived)	Number of jobs processed by each activity in the BP (NJA = NEA/Total BP cases)
	M4 (derived)	Unit load for a resource in the BP (ULR = ∑(AATWoT * NJA)
	M5 (derived)	Unit capacity for each resource (UCR = 1/ULR indicates the number of jobs each resource can complete per unit of time)
	M6 (derived)	Pool capacity for each role in the BP (PCR = UCR * NRRBP)
	M7 (derived)	Process capacity of the BP (PCBP = Bottleneck of the BP = smallest of measure 6)
	M8 (derived)	Throughput rate of the BP = arrival rate to the system corresponding to the average number of jobs eventually served per unit of time (TRBP = total of BP cases/number of time periods)
	M9 (derived)	Capacity utilization for a resource (CUR = TRBP/PCR)

Table 6
Measures for Generic BP execution view & Cost dimension.

Goal	**G3**	**Minimize the cost of the BP**
Question	**Q1**	**What is the actual cost of the BP**
Measures	M1 (base)	Resource cost per unit of time = RCT (from context data)
	M2 (derived)	Cost per activity in a BP case (ACo = AWoT * RCT)
	M3 (derived)	Total cost per activity in all BP cases (TACo = ∑ACoBP(i))
	M4 (derived)	Total cost of BP case (TCo = ∑ACo(i))
	M5 (derived)	Total cost of BP for all BP cases (TBPCo = ∑TCo)
	M6 (indicator)	Percentage of activity cost in BP case (PACo = ACo * 100/TCo) Decision criteria = Cost DC
	M7 (indicator)	Percentage of activity cost in all BP cases (PTACo = TACo * 100/TBPCo) Decision criteria = Cost DC
	M8 (indicator)	Percentage of BP case cost in all BP cases (PTCo = TCo * 100/TBPCo) Decision criteria = Cost DC
	M9 (indicator)	Average cost of BP for all BP cases (ABPCo = TBPCo/Total BP cases) Decision criteria = Cost DC
	M10(indicator)	Average cost of activity for all BP cases (AACo = TACo/Total BP cases) Decision criteria = Cost DC
Decision criteria	Cost DC:	R1: 0 <= TTI <= L1 = "LOW" = GREEN; R2: L1 <= TTI < L2 = "MEDIUM" = YELLOW; R3: L2 <= TTI = "HIGH" = RED

Table 7
Measures for Generic BP execution view & Quality dimension – Type of ending.

Goal	**G4**	**Maximize the number of BP cases ending normally**
Question	**Q1**	**What is the actual number of cases ending normally**
Measures	M1 (base)	Number of BP cases ending in the selected state = NBPE (count BP cases ending in states: COMPLETED, TERMINATED, ABORTED)
	M2 (indicator)	Percentage of BP ending in completed state in total BP cases (PBPCo = NBPE * 100/Total BP cases) for state = COMPLETED. Decision criteria = Percentage Complete DC
	M3 (indicator)	Percentage of BP ending in terminated state in total BP cases (PBPTe = NBPE * 100/Total BP cases) for state = TERMINATED. Decision criteria = Inverse Percentage Complete DC
	M4 (indicator)	Percentage of BP ending in aborted state in total BP cases (PBPCo = NBPE * 100/Total BP cases) for state = ABORTED. Decision criteria = Inverse Percentage Complete DC
Decision criteria	Percentage Complete	R1: 0 <= TTI <= L1 = "LOW" = RED; R2: L1 <= TTI < L2 = "MEDIUM" = YELLOW; R3: L2 <= TTI = "HIGH" = GREEN

Table 8
Measures for Generic BP execution view & Quality dimension – Successful branch.

Goal	**G2**	**Maximize the number of BP cases ending successfully (executes the successful branch of the BP)**
Question	**Q1**	**What is the actual number of BP cases ending successfully**
Measures	M1 (base)	Number of BP cases ending successfully or unsuccessfully = NBPBE (count BP cases with activities in the successful or unsuccessful branch as defined in the context data)
	M2 (indicator)	Percentage of BP ending successfully in total BP cases (PBPSB = NBPBE * 100/Total BP cases) for successful branch. Decision criteria = Percentage Successful DC
	M3 (indicator)	Percentage of BP ending unsuccessfully in total BP cases (PBPUSB = NBPBE * 100/Total BP cases) for unsuccessful branch. Decision criteria = Inverse Percentage Successful DC
Decision criteria	Percentage Successful	R1: 0 <= TTI <= L1 = "LOW" = RED; R2: L1 <= TTI < L2 = "MEDIUM" = YELLOW; R3: L2 <= TTI = "HIGH" = GREEN

Table 9
Measures for Lean execution view & Quality dimension.

Goal	**G1**	**Minimize the rework in loops of the BP**
Question	**Q1**	**What is the actual quantity of rework due to BP loops**
Measures	M1 (base)	Number of executions of an activity in a rework loop = NARL (counts the times each activity is executed in a rework loop as defined in the context data)
	M2 (derived)	Activity Working time for the rework in a loop (AWoTRL = ∑AWoT(ei) being ei each execution of the activity in the loop)
	M3 (derived)	Total Working time for the rework in a loop (TWoTRL = ∑AWoTRL (ai) where ai represents an activity in the loop)
	M4 (derived)	Total Working time for rework in all loops of BP case (BPTWoTRL = ∑TWoTRL (li) where li represents a loop in the BP)
	M5 (derived)	Total Working time for rework of an activity in all BP cases (TAWoTRL = ∑AWoTRL)
	M6 (derived)	Total of BP cases with execution of rework loops (TBPERL = ∑BP cases with execution of rework loops)
	M7 (indicator)	Percentage of rework time in BP case due to loops in the total BP TT (PBPTWoTRL = BPTWoTRL * 100/BPTT) Decision criteria = Percentage DC
	M8 (indicator)	Percentage of BP cases with execution of rework loops (PTBPERL = TBPERL * 100/Total BP cases)
	M9 (indicator)	Percentage of rework time for an activity due to execution of rework loops in all BP cases (AAWoTRL = TAWoTRL * 100/Number of BP cases in which the activity was executed)
Decision criteria	Percentage DC:	R1: 0 <= TTI <= L1 = "LOW" = GREEN; R2: L1 <= TTI < L2 = "MEDIUM" = YELLOW; R3: L2 <= 100 = "HIGH" = RED

Table 10
Measures for Service execution view & Time dimension – Service Response Time.

Goal	**G1**	**Guarantee (average) service response time to (L1) seconds (L1 label to be changed)**
Question	**Q1**	**What is the actual (average) response time of the service**
Measures	M1 (base)	Invoke time of a service from the activity in the BP (IT = timestamp)
	M2 (base)	Enabled time of a service (ET = timestamp)
	M3 (base)	Start time of a service (ST = timestamp)
	M4 (base)	Completion time of a service (CT = timestamp)
	M5 (base)	Failed time of a service (FT = timestamp)
	M6 (base)	Answer time from the service to the activity in the BP (AT = timestamp)
	M7 (derived)	Service processing time (SPoT = CT - ST)
	M8 (derived)	Service latency time (SLaT = ST - ET)
	M9 (derived)	Service response time (SRpT = SPoT + SLaT)
	M10 (derived)	Service answer time from the BP (SAnT = AT - IT)
	M11 (indicator)	Service Processing time vs. Service Latency time index (STI = SLaT/SPoT) Decision criteria = Index DC
	M12 (indicator)	Average service response time in all BP cases (ASRpT = ∑SRpT/Total service executions in all BP cases) Decision criteria = Index DC
	M13 (indicator)	Average service answer time in all BP cases (ASAnT = SAnT/Total service executions in all BP cases) Decision criteria = Index DC
Decision criteria	Index DC:	R1: 0<=TTI<= L1 = "LOW" = GREEN; R2: L1<= TTI < L2 = "MEDIUM" = YELLOW; R3: L2<= TTI = "HIGH" = RED

Table 11
Measures for Service execution view & Time dimension – Service Throughput.

Goal	**G2**	**Guarantee service throughput to (S) service execution completed per period (P1) (S and P1 labels to be changed)**
Question	**Q1**	**What is the actual service throughput S1 service execution completed over the period P1**
Measures	M1 (base)	Number of S1 service execution over the period P1 = NSEOP (count service execution COMPLETED, FAILED or IN PROGRESS in the period P1)
	M2 (indicator)	Percentage of S1 service execution completed over the period P1 (PSECP = NSEOP * 100/Total service execution including in progress) Decision criteria = SE completed DC
	M3 (indicator)	Percentage of S1 service execution failed over the period P1 (PSEFP = NSEOP * 100/Total service execution including in progress) Decision criteria = Inverse SE completed DC
Decision criteria	SE completed	R1: 0<=TTI<= L1 = "LOW" = RED; R2: L1<= TTI < L2 = "MEDIUM" = YELLOW; R3: L2<= TTI = "HIGH" = GREEN

Table 12
Measures for Service execution view & Time dimension - Service Capacity.

Goal	**G3**	**Guarantee service capacity to (S) service execution maintaining the (L1) seconds defined for service response time (S and L1 labels to be changed)**
Question	**Q1**	**What is the actual service capacity**
Measures	M1 (base)	Number of S1 service execution completed in <= L1 s over the period P1 (NSECLP = ∑SECLP in the period P1)
	M2 (base)	Number of S2 service execution completed in L2 > L1 s violating agreements over the period P1 (NSECVLP = ∑SECVLP in the period P1)
	M3 (base)	Number of S3 service execution in progress in L2 > L1 s violating agreements over the period P1 (NSEIPVLP = ∑SEIPVLP in the period P1)
	M4 (indicator)	Service capacity (SCA = NSECLP * 100/NSECLP + NSEIPVLP + NSECVLP + NSEFP) Decision criteria = Percentage SCA
	M5 (indicator)	Service capacity violation rate (SCVR = NSECVLP + NSEIPVLP * 100/NSECLP + NSEIPVLP + NSECVLP + NSEFP) Decision criteria = Inverse Percentage SCA
Decision criteria	Percentage SCA	R1: 0<=TTI<= L1 = "LOW" = RED; R2: L1<= TTI < L2 = "MEDIUM" = YELLOW; R3:L2<= TTI = "HIGH" = GREEN

Table 13
Measures for Service execution view & Quality dimension.

Goal	**G2**	**Guarantee (A1) availability for the service (A1 label to be changed) – Dependability**
Question	**Q1**	**What is the actual availability of the service**
Measures	M1 (derived)	Service down time (SDT = ET - FT being ET the time when the service is back up)
	M2 (derived)	Total service down time over the period P1 (TSDT = ∑SDT in the period P1)
	M3 (indicator)	Service Availability over the period P1 (SA = P1 - TSDT/P1 * 100) Decision criteria = Percentage SR
Goal	**G2**	**Guarantee (R1) reliability for the service (R1 label to be changed) – Dependability**
Question	**Q1**	**which is the actual reliability of the service**
Measures	M1 (base)	Number of service execution initiated over the period P1 = NSEIP (counts the services ST initiated in the period P1)
	M2 (indicator)	Service Reliability (SR = NSRECP/NSEIP * 100) Decision criteria = Percentage SR
Goal	**G3**	**Guarantee (C1) confidentiality level for the service (C1 label to be changed) – Security**
Question	**Q1**	**What is the actual confidentiality level of the service**
Measures	M1 (base)	Number of service invocations rejected due to invalid credentials over the period P1 = NSIR (counts the service invocations rejected in the period P1)
	M2 (indicator)	Percentage of service invocations rejected in all services invocations over the period P1 (PSIRSI = NSIR * 100/NSIR + NSEIP) Decision criteria = Inverse Percentage SR
Decision criteria	Percentage SR	R1:0<=TTI<=L1 = "LOW" = RED; R2: L1<= TTI < L2 = "MEDIUM" = YELLOW; R3:L2<= TTI = "HIGH" = GREEN

References

[1] M. Weske, BPM Concepts, Languages, Architectures, Springer, 2007.
[2] W. van der Aalst, A. ter Hofstede, M. Weske, Business process management: A survey, in: International Conference on Business Process Management (BPM0'3), 2003.
[3] H. Smith, P. Fingar, Business Process Management: the Third Wave, Meghan-Kiefer, 2003.
[4] G. Gartner, Survey Analysis: BPM Spending to Grow Significantly in 2011, 2011, February. <http://www.gartner.com>.
[5] T. Erl, Soa: Concepts, Technology, and Design, Prentice Hall, 2005.
[6] D. Krafzig, K. Banke, D. Slama, Enterprise Soa Best Practices, Prentice Hall, 2005.
[7] M. Papazoglou, P. Traverso, S. Dustdar, F. Leymann, Service-oriented computing: state of the art and research challenge, IEEE Computer Society (2007).
[8] A. Delgado, B. Weber, F. Ruiz, I. García-Rodríguez de Guzmán, Execution measurement-driven continuous improvement of business processes implemented by services. In: 6th International Conference on Evaluation of Novel Approaches to, Software Engineering (ENASE'11), 2011.
[9] A. Delgado, F. Ruiz, I. García-Rodríguez de Guzmán, M. Piattini, MINERVA: Model driven and service oriented framework for the continuous business process improvement & related tools, in: 5th International Workshop on Engineering SO Applications (WESOA'09), in (ICSOC'09), 2009.
[10] A. Delgado, I. García-Rodríguez de Guzmán, F. Ruiz, M. d Piattini, Model-driven and service-oriented framework for the business process improvement, Journal of Systems Integration 1 (2010).
[11] A. Delgado, I. García-Rodríguez de Guzmán, F. Ruiz, M. Piattini, Towards an ontology for service oriented modeling supporting business processes, in: 4th International Conference on Research Challenges in Information Science (RCIS'10), 2010.
[12] A. Delgado, I. García-Rodríguez de Guzmán, F. Ruiz, M. Piattini, Tool support for service oriented development from business processes, in: 2nd International Workshop on Model-Driven Service, Engineering (MoSE'10), 2010.
[13] A. Delgado, F. Ruiz, I. García-Rodríguez de Guzmán, M. Piattini, Business process service oriented methodology (BPSOM) with service generation in soaml, in: 23rd International Conference On Advanced Information, Systems Engineering (CAISE'11), 2011.
[14] A. Delgado, F. Ruiz, I. García-Rodríguez de Guzmán, M. Piattini, Model transformations for business-it alignment: from collaborative business process to soaml service model, in: 27th Symposium on Applied, Computing (SAC'12), 2012.
[15] H.J. Harrington, Business Process Improvement: The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness, first ed., McGraw-Hill, 1991. ISBN 978-0070267688.
[16] OMG, Business Process Maturity Model (BPMM), 2008. <http://www.omg.org/spec/BPMM/>.
[17] L. Sánchez González, A. Delgado, F. Ruiz, F. García, M. Piattini, Measurement and maturity of business processes, in: J. v.d.A.W. Cardoso (Ed.), Handbook of Research on Business Process Modeling, Information Science Reference (IGI Global), 2009, pp. 532–556.
[18] M. Castellanos, A. Alves de Medeiros, J. Mendling, B. Weber, and A. Weijters. Business Process Intelligence, Handbook of research on business process modeling. In v. d. A. W. Cardoso J. (Ed.). Information Science Reference (IGI Global), (2009) 456–480.
[19] W.M.P. van der Aalst, Process Mining, Discovery, Conformance and Enhancement of Business Processes, first ed., Springer, 2011. No. ISBN 978-3-642-19344-6.
[20] M. Westergaard, F. Maggi, Modeling and Verification of a Protocol for Operational Support Using Coloured Petri Nets, Applications and Theory of Petri Nets, Lecture Notes in Computer Science, vol. 6709, Springer, 2011, pp. 169–188.
[21] H. Schonenberg, B. Weber, B. van Dongen, Wil M. P. van der Aalst, Supporting Flexible Processes through Recommendations Based on History, 6th International Conference on Business Process Management (BPM'08), Lecture Notes in Computer Science, Springer, 2008, pp. 51–66.
[22] M. Laguna, J. Marklund, Business Process Modeling, Simulation and Design, Prentice Hall, 2005.
[23] Young Hoon Kwaka, Frank T. Anbarib, Benefits, obstacles, and future of six sigma approach, International Journal of Technological Innovation Entrepreneurship and Technology Management Technovation 26 (5) (2006) 708–715.
[24] R. Galloway, Six Sigma Implementation: An Empirical Analysis of Critical Success Factors and Performance Outcomes, Master Thesis, Department of Economics and Business Administration, University of Maastricht, The Netherlands, 2009.
[25] M. Draghici, A. Petcu, TQM and Six Sigma – the Role and Impact on Service Organization, The Romanian Economic Journal, Year XIII, no. 36 (2010) 123–135.
[26] F. Pino, J. Hurtado, J. Vidal, F. García, M. Piattini, A process for driving process improvement in vses, in: International Conference on Software Process (ICSP'09), 2009.
[27] G. Wessel, P. Burcher, Six sigma for small and medium-sized enterprises, The TQM Magazine 16 (4) (2004) 264–272.
[28] E. Rolón, F. García, F. Ruiz, M. Piattini, L. Calahorra, R.P. Sánchez, T. Rodríguez, Healthcare Process Development with BPMN, in: Handbook of Research on Developments in e-Health and Telemedicine: Technological and Social Perspectives, IGI Global, USA, 2010, pp. 1024–1047. ISBN 978-1-61520-670-4.
[29] A. Delgado, F. Ruiz, I. García-Rodríguez de Guzmán, M. Piattini, Application of service-oriented computing and model-driven development paradigms to business processes: a systematic review, in: 5th International Conference on Software and Data Technologies (ICSOFT'10), 2010.
[30] L. Sánchez González, F. García, F. Ruiz, M. Piattini, Measurement in business processes: a systematic review, Business Process Management Journal 16 (1) (2010) 114–134.
[31] A. Hevner, S. March, J. Park, S. Ram, Design science in information systems research, Journal MIS Quarterly 28 (1) (2004) 75–105.
[32] I. Jacobson, G. Booch, J. Rumbaugh, The Unified Software Development Process, Addison-Wesley, 1999.
[33] RUP-IBM, Rational Unified Process (RUP), 1999-2011. <http://www.-01.ibm.com/software/awdtools/rup/>.
[34] OMG, Business Motivation Model (BMM), 2010. <http://www.omg.org/spec/BMM/>.
[35] BPMI, Business Process Management Initiative, 2000-2005. <http://www.bpmi.org/>.
[36] OMG, Business Process Model and Notation (BPMN2), 2011. <http://www.omg.org/spec/BPMN/2.0/>.
[37] H. Reijers, Design and Control of Workflow Processes: BPM for Service Industry, Springer, 2003.
[38] L. Maruster, N.van. Beest, Redesigning business processes: a methodology based on simulation and process mining techniques, Knowledge and Information Systems 21 (3) (2009).
[39] M. Netjes, Process improvement: The creation and evaluation of process alternatives, Beta, 2010.
[40] W.M.P. van der Aalst, M. Voorhoeve, Business process simulation, lecture notes 2ii75, 2010–2011.
[41] J. Cardoso, Control-flow complexity measurement of processes and Weyuker's properties, in: 6th International Conference on Enformatika, International Academy of Sciences, 2005.
[42] E. Rolón, F. Ruiz, F. García, M. Piattini, Evaluation measures for business process models, in: 21st Symposium on Applied, Computing (SAC'06), 2006.

[43] J. Mendling, Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness, Springer, 2008.
[44] L. Sánchez González, F. García, I. Mendling, F. Ruiz, Assessment and prediction of business process model quality, in: 18th International Conference on Cooperative, Information Systems (COOPIS'10), 2010.
[45] OMG, Service Oriented Architecture Modeling Language (SoaML), Beta 2, 2009. <http://www.omg.org/spec/SoaML/1.0/Beta2/>.
[46] WfMC, Xml Process Definition Language (XPDL), 2008. <http://www.wfmc.org/xpdl.html>.
[47] OASIS, Web Services Business Process Execution Language (ws-bpel), 2007. <http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html>.
[48] W. van der Aalst, A. ter Hofstede, B. Kiepuszewski, A. Barros, Workflow patterns, Distributed and Parallel Databases 14 (3) (2003) 5–51.
[49] P. Kruchten, The "4+1" View, Model of Software Architecture, IEEE Software 12 (6) (1995) 42–50.
[50] P. Clements, Documenting Software Architecture: Views and Beyond, first ed., Addison-Wesley Professional, 2003.
[51] V. Basili, Software Modeling and Measurement: The GQM Paradigm (Tech. Rep.), University of Maryland, CS-TR-2956, 1992.
[52] F. García, M. Bertoa, C. Calero, A. Vallecillo, F. Ruiz, M. Piattini, et al., Towards a consistent terminology for software measurement, Information and Software Technology 48 (2005).
[53] N. Brand, H. van der Kolk, Workflow Analysis and Design, 1995.
[54] M. zur Muehlen, Workflow-based Process Controlling, Foundation, Design, and Application of Workflow-driven Process Is, LogosVerlag, 2004.
[55] M. Poppendieck, Principles of lean thinking, in: Oopsla (Object-Oriented Programming, Systems, Languages & Applications) Onward!, 2002.
[56] B. Hicks, Lean information management: understanding and eliminating waste, Information Management (2007).
[57] C. Jimmerson, D. Weber, D. Sobek, Reducing waste and errors: piloting lean principles at intermountain healthcare, Quality & Patient Safety (2005).
[58] M. Barbacci, M. Klein, T. Longsta, C. Weinstock, Quality Atributtes (Tech. Rep.), Software Engineering Institute (SEI), CMU/SEI-95-TR-021, 1995.
[59] L. O'Brien, L. Bass, P. Merson, Quality Attributes and Soa (Tech. Rep.), Software Engineering Instittue (SEI), CMU/SEI-20055-TN-014, 2005.
[60] P. Clements, R. Kazman, M. Klein, Evaluating Software Architectures: Methods and Case Studies, Addison Wesley, 2001.
[61] L. Bass, P. Clements, R. Kazman, Software Architecture in Practice, Addison Wesley, 2003.
[62] A. Sahai, J. Ouyang, V. Machiraju, K. Wurster, Specifying and guaranteeing quality of service for web services through real time measurement and adaptive control (Tech. Rep.), Management Project, E-Services Software Research Department, HP Laborataries, 1501 Page Mill Road, Palo Alto, CA 94034, 2001.
[63] J. Cardoso, A. Sheth, J. Miller, Workflow quality of service, in: International Conference on Enterprise Integration Modeling Technology (ICEIMT/IEM'02), 2002.
[64] B. Mora, F. García, F. Ruiz, M. Piattini, Smml: Software measurement modeling language, in: 8th Oopsla Workshop on Domain-Specific Modeling, (oopsla'08), 2008.
[65] B. Mora, F. García, F. Ruiz, M. Piattini, Graphic versus textual software measurement modelling: an empirical study, Software Quality Journal 19 (1) (2011) 201–233.
[66] R. Yin, Case Study Research: Design and Methods, Sage Publications, Inc., 2002.
[67] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering 14 (2) (2008) 131–164.
[68] P. Brereton, B. Kitchenham, D. Budgen, Z. Li, Using a protocol template for case study planning, in: Evaluation and Assessment in, Software Engineering (EASE'08), 2008.
[69] A. Rozinat, R.S. Mans, M. Song, W. van der Aalst, Discovering colored Petri Nets from event logs, Software Tools for Technology Transfer 10 (1) (2008) 57–74.
[70] W. van der Aalst, J. Nakatumba, J. Rozinat, N. Russell, Business Process Simulation: How to get it right?, Handbook on Business Process Management, International Handbooks on Information Systems, Springer-Verlag, 2010.
[71] A. Alves de Medeiros, C. Guenther, Process mining: Using cpn tools to create test logs for mining algorithms, in: 6th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the Cpn Tools, 2005.
[72] W. van der Aalst, C. Stahl, Modeling Business Processes-A Petri Net-Oriented Approach, The MIT Press, 2011.
[73] K. Jensen, Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use, Springer, 1997.
[74] M. Wynn, M. Dumas, C. Fidge, A. ter Hofstede, W. van der Aalst, Business process simulation for operational decision support, in: International Workshop on Business Process Intelligence (BPI'07), in (BPM'07), 2007.
[75] R. Kaplan, D. Norton, The balanced scorecard measures that drive performance, Harvard Business Review 10 (1) (1992).
[76] L. Buglione, A. Abran, Balance scorecards and GQM: what are the differences? in: Fesma-aemes Software Measurement Conference, 2000.
[77] F. Casati, M. Castellanos, U. Dayal, N. Salazar, A generic solution for warehousing business process data, in: 33rd International Conference on Very Large Data Bases (VLDB'07), 2007.
[78] IDS Scheer, Software AG, ARIS, Germany.