Considering Quality Factors for Business Processes During Requirement Engineering

Abstract - Business process (BP) modelling is considered as an important part of Requirement Engineering in describing the context within which requirements are articulated, specified and analysed. The intertwining of business and information systems imposes considerable stresses on requirements engineers to consider quality factors that need to be present in both business processes and the software that supports these processes. We argue that the quality of BPs has to be taken into account at the requirement engineering stage, where quality requirements are to be defined, along with the corresponding quality factors and the metrics that will be used to evaluated them. We consider that a BP is represented by a model and that quality requirements are to be examined within the BP model. In this work, our goal is to provide a support for evaluating the quality of a BP through the corresponding model. We present a quality framework for BPs and a quality-oriented BP metamodel. This metamodel is enriched with quality-related attributes that will be used to evaluate the quality factors. To demonstrate the utility of the proposed quality framework we will use a BP model relating to the management of spectators at an Olympics venue, we will show the instantiation of our metamodel in this context along with evaluation results for some quality factors.

Keywords: Requirement Engineering, Business Process Quality

I. INTRODUCTION & MOTIVATION

The intertwining of business to software highlights the importance of a continuous ‘dialogue’ between the domain expert and the software expert and this can be achieved only during the requirements elicitation and analysis phase. The genesis of Requirements Engineering (RE) research around 30 years ago was motivated by practitioners who noticed the urgent need for disciplined RE in large software projects [24][8]. Much of RE research since then has focused on artifacts that help capture, share, represent, analyze, negotiate, and prioritize requirements as a basis for design decisions and interventions (for recent reviews see [30][29][5]).

Business process (BP) modelling is considered as an important part of Requirement Engineering. Considering the quality of business processes is essential, as this will drive the quality of the future software. Besides, socio-technical aspects are increasingly important in BP modelling. Assessing the quality of a business process is a complex task: on the one hand, there are different perceptions of quality corresponding to different stakeholders (business, modelers, software engineers, etc.); on the other hand, there are many approaches, quality criteria and evaluation tools.

In this paper, we are focusing on examining the level of quality of a business process and we will address this need by investigating how best one can analyse the quality of a business process against the requirements.

Business processes can be described by the mean of business process models (BPMs). A BPM is a representation describing the order of activities within a process. There are several proposals for BPM languages, such as the UML Activity Diagram [31] or the Business Process Modelling Notation (BPMN) [32]. In this paper, the key issue is to determine to what extent the quality requirements are satisfied by a BP through the corresponding model. We will propose a framework for business process quality evaluation and a quality-oriented meta-model for BPs. We believe that this quality framework can be useful to several research areas: in requirement engineering and software engineering, to allow handling non functional requirements at the earliest design stage, in process modelling, to enable quality driven modelling and re-design, or in service-oriented architectures, where the notion of quality of the web services and their composition can be a critical issue.

This paper is organized as follows. In section II, we will present some works related to quality of BPs. Section III will present our general framework for evaluating the quality of BPs. In section IV, we will first present some quality factors for BPs, then we will present the enriched BP metamodel, containing all the attributes and concepts that are necessary to evaluate these factors. In section V, we will illustrate using a real application how our metamodel is instantiated, and finally, section VI will conclude the paper and give some perspectives.
II. RELATED WORKS

In software engineering and information systems, the evaluation of the quality of business process models has been addressed by several research communities. In business process modeling, some works [7][8][9][13][17][19][23] have highlighted the importance of evaluating the quality of BPMs and proposed to adapt classical quality metrics used in software engineering. In [27], an overview of current business process metrics is presented, and the implementation of these metrics is presented using the ProM analysis tool. In [28], the authors propose a weighted coupling metric for BPM defined as a measure of the number of interconnections between the activities in a BPM, similarly to the definition of coupling in software engineering.

In [25], a model for measuring BP quality is proposed, inspired by the ISO/IEC 9126 Software Product Quality Model and its four-level structure: categories, characteristics, sub-characteristics and finally metrics. The characteristics are Functionality, Reliability, Usability and Maintainability. In [13], four generic categories of BP quality are identified and a quality framework for BPs is proposed. Four categories of BP quality are identified: function quality, input/output quality, quality of non-human resources and quality of human resources. For each category, a set of quality dimensions are identified. The two approaches presented in [25] and [13] present some quality criteria for BPMs, they however don’t delve into the details of the appropriate metrics to measure them. In [23], the authors discuss the critical factors of the quality of BPMs. They propose a framework aiming at developing design recommendations in order to increase the quality of models. The framework consists in six quality factors: correctness, relevance, economic efficiency, clarity, comparability and systematic design.

In the field of web services, some works have addressed the quality aspects [11][9]. The approach presented in [9] focuses on web services composition, and its goal is to analyze the quality of composed services using fault injection techniques. In [22], monitors for quality of service are implemented using aspect-oriented programming, QoS constraints being implemented as aspects. The considered QoS criteria are classified into 3 categories: (i) execution criteria, (ii) service provider criteria and (iii) exception related criteria.

In business and management, several methods (Six Sigma, TQM, EFQM Excellence Model) and standards (ISO 9000) have been proposed to assess the quality of a business process, but the notion of quality considered in these proposals is at the organization level, while our focus is the quality of business process through the models representing them.

The approaches proposed in the field of requirement engineering [1][4][6][10][17] are concerned with the quality of the software, not the quality of the business process. They consider software requirements in the context of some business model. We argue that the quality of the business process should be considered before the quality of the software. For example, it is not worth focusing on the level of modularity of a software, if the overall cost of the corresponding process does not meet the expectations. In our view, assessing the quality of processes is a prerequisite for assessing the quality of the software.

The approaches proposed in BP modelling [26][27][28][13][19] mainly address the quality of the model representing the BP, which is only a facet of the quality of the BP itself. Some of these approaches propose some criteria allowing to evaluate how well the process is represented by the model at hand using criteria such as completeness or correctness [23][13]. Other approaches propose to adapt metrics from software engineering to BP in
order to assess the model [27]. Other approaches empirically validate assumptions establishing a link between the model’s structure and the structure of the corresponding business process [19]. In all these cases, the problem targeted by these approaches is the quality of the model, and their goal is either to determine if the model is a good representation of the actual process, or to evaluate the intrinsic quality of the model, such as its simplicity and its readability, or try empirically to relate the structure of the model to some quality property, e.g. empirically show that error-probability is related to the size of the model.

If we consider the quality criteria proposed in the field of web services [9][22], we can see that these criteria are also relevant for BPs (cost, response time etc.). But in these approaches, the criteria are mainly related to either the performance of web services, or to the quality of a composition of several web services for a specific purpose. We believe that other factors should be considered for BPs, such as their ability to enforce security or privacy.

III. A FRAMEWORK FOR BUSINESS PROCESS
QUALITY EVALUATION

All the approaches presented in the previous section address a specific facet of BP quality. Existing approaches in business and management consist mostly in general principles or guidelines for quality, and the one existing in information systems are often concerned with the quality of the software, not with quality of the business process itself. This work is a first step towards bridging the gap between these two views of business process quality (Figure 1).

Figure 1. Bridging the gap between the business view and the technical view of quality of BPs

We consider that a BP is represented by a model and that some quality requirements are expressed for the BP. We also consider that these requirements are evaluated against the model. Unlike some of the existing approaches, we are neither interested in assessing how well does the model represents the process, nor in checking that the model conforms to the rules of the modelling language. Considering that the model is complete, and that it is semantically and syntactically correct, our goal is to provide a support for evaluating the quality of the business process through the corresponding model. This raises several issues, such as the definition of quality factors and associated metrics for BPs independently of any BP modelling language, or the design of quality evaluation algorithms.

In order to enable efficient management of BP quality, the following issues are raised:

- What are the quality factors and associated metrics relevant for BPs?
- What are the quality services that allow the effective evaluation of these factors and metrics?
- Independently from a specific application, how to capture quality information at the metamodel level?
- Considering a specific application, what are the appropriate quality services allowing to achieve the quality requirements?

These issues correspond to the main components of our quality framework represented in figure 2, and described below.

Definition of Quality Factors and Metrics.

This component of the framework deals with the definition of the different quality properties or quality factors relevant for business processes. Each quality factor is associated with a definition and a set of quality metrics, each one representing one possible way of calculating a value for the considered factor. Several metrics can be associated for the same factor, as there might exist several ways of evaluating it. The output of this process is a library of quality factors, along with the associated metrics.

Definition of Quality Evaluation Tools.

This component deals with the identification of the development of the tools to effectively evaluate the quality of the processes. These tools can either be specifically developed for a given metric, or they may consist in existing remote quality services that can be invoked. The goal of this component is to build a library of quality tools corresponding to the factors and metrics identified by the previous component.

Enrichment of the BP Metamodel.

Given a metamodel of business processes, this component enriches the metamodel with all the properties required to describe various aspects of quality related to a business process. The resulting metamodel will incorporate not only the elements describing the process, but also the ones describing the quality of the process.

Selection of Quality Evaluation Tools.
The three previous components are independent from a specific context, process or quality requirement. The library of quality factors, the library of quality tools and the enriched metamodel are not specific to a particular application. Unlike these components, the selection of quality evaluation tools is done according to some specific quality requirements for a given process and consists in choosing the quality factors, metrics and tools that are appropriate to meet the quality requirements expressed on the considered business process.

**IV. A QUALITY-ORIENTED BUSINESS PROCESS METAMODEL**

In this work, we consider that a business process is represented by a model and that this model is complete, semantically and syntactically correct. We will therefore neither address the problem of evaluating to what extent the model represents correctly and completely the real world, nor the problem of evaluating the validity of the model with respect to the rules of the modelling language. We are interested in evaluating some quality requirements against this model. These requirements might be related to several facets of the BP, such as the execution cost of a BP, its processing time, reliability, availability, security, scalability etc. We also consider that a business process metamodel is available. Our approach is based on the enrichment of this metamodel with all the concepts and properties required to effectively evaluate the quality requirements. In order to capture the quality requirements, we use some quality factors, which will be evaluated using some quality metrics.

We argue that the quality of business processes has to be taken into account at the requirement engineering stage, where quality requirements are to be defined, along with the corresponding quality factors and the metrics that will be used to evaluated them. These metrics will ensure that the quality requirements will be testable, and that the results of these tests will be used to refine the design of the BP model during requirement engineering. This iterative process for dealing with quality during RE is illustrated by Figure 3.

In this section, we will first present some of the quality factors that can be used to capture quality requirements. We will then present a partial business process metamodel and show how it can be extended to take quality aspects into account.

**A. Quality Factors for BPs**

Quality factors for BPs have been the focus of numerous works [26][27][28][13][19]. Rather than defining new quality factors, our goal is to use existing ones and show how a business process metamodel can be be extended so as to allow the evaluation of these factors. For this purpose, we have selected four quality factors: the cost of a BP, its response time and its ability to preserve data security and privacy.

**Response Time of a BP**

![Diagram for Evaluating Quality Requirements for BPs](image)
The response time of a BP is the time required for the execution of the BP. It can be evaluated using different parameters, such as the time required to compute each activity of the BP, the time required to transfer messages from one activity to another or the size of the load.

Cost of a BP

The cost of a business process can be defined based on the cost of the activities that compose this process. If we consider a business process composed of \( n \) activities \( a_i \), and if \( c(a_i) \) represents the average cost of this activity, the cost of the BP can be evaluated as: \( \sum c(a_i), i=1,n \).

Privacy Preserving BP

This requirement expresses the extent to which a BP enforces the privacy of the data. A BP is said to be privacy preserving if it guarantees that data access is always done by actors with appropriate rights. If we consider a business process composed of \( n \) activities \( a_i \), if \( Act_k \) represents an actor involved in the BP and if \( R_{kj} \) represents the access right granted to actor \( Act_k \) on data \( D_j \), an activity \( a_i \) is a privacy violating activity (PVA) if this activity is executed by an actor \( Act_k \), accessing data \( D_j \) and if \( R_{kj} \) is not defined. A possible metric for evaluating the extent to which a BP is privacy preserving is the proportion of PVA in the process:
\[
M_1 = \frac{\text{nb PVA}}{n}.
\]

Security Preserving BP

This requirement expresses the extent to which a BP enforces the security constraints. A BP is security preserving if all its activities are executed by authorized actors.

If we consider a business process composed of \( n \) activities \( a_i \), if we denote by \( S_j \) a security constraint representing the fact that activity \( a_i \) is allowed for actor \( Act_k \), a security violating activity (SVA) as an activity \( a_i \) executed by an actor \( Act_k \) if \( S_j \) is not defined. One possible metric for evaluating the security preserving level of a BP is the proportion of SVA in the BP:
\[
M_2 = \frac{\text{nb SVA}}{n}.
\]

B. Enriching the Process Metamodel

In the present work, one of our contributions is to present a business process metamodel enriched with quality related attributes. These attributes will be used to evaluate the quality factors such as the ones presented in the previous sections. Consider the (partial) BP metamodel presented in figure 4. It represents the main component of a BP: its activities, each one accepting a set of inputs and producing some outputs, the connectors linking activities. These connectors can either be messages or control flows.

![Figure 4. Partial BP metamodel](image)

In order to enable the effective evaluation of some quality factors, we propose to enrich this metamodel with some quality attributes. Consider the cost quality factor. The cost of a business process may be computed using the following metric: \( \sum c(a_i), i=1,n \), where \( a_i \) represents an activity of the business process and \( c(a_i) \) the average cost of this activity. The metamodel of figure 4 will be enriched with the following attributes:

- \( \text{avg\_cost} \) in the Activity class, representing the average cost of this activity,
- \( \text{cost} \) in the Process class, representing the cost of the entire process and
- \( \text{CThres} \) in the Process class, representing the cost threshold above which the cost of a BP is not considered as satisfactory.

The enriched metamodel is shown in figure 5. In the same way, the attributes \( \text{avg\_time} \) and \( \text{resp\_time} \) have been added to the Activity and the Process classes respectively in order to evaluate the Response Time factor.

![Figure 5. Extract of the metamodel with quality information](image)
In order to evaluate the Security Preserving quality factor, the class **Actor** has been added, as well as the association **executes**, linking the activities to the actors executing them. The association **Authorizes** has been added, representing the fact that the execution of an activity by an actor is allowed (this actor has been granted the execution right for this activity). An alternative design would have been to add a restriction association, relating pairs of actors and activities each time the execution of the activity by the actor is not allowed.

In order to evaluate the Privacy Preserving quality factor, the association **Grants** has been added, relating actors to inputs when these actors have been granted an access right on this input. The metamodel can be gradually enriched by all the attributes, classes or associations required to evaluate the quality factors using one or several metrics.

V. USING THE METAMODEL FOR BP QUALITY EVALUATION

To demonstrate the utility of the Business Process Quality framework and the enriched metamodel, we use a small part of a business process model relating to the management of spectators at an Olympics venue [2].

This partial model focuses on the management of spectators as they enter a venue and in particular is concerned with **queue management** as spectators approach the venue from different public transportation terminals and the **security checking** that all spectators are subjected to prior to entering the common area of a venue. This is only but a small fraction of the overall process model as the complete one is beyond the scope of this paper.

Figure 6. A partial process model for managing spectators in an Olympics venue

The problem is one that is generalised across all venues and it is essentially one of identifying the appropriate level of service such that it satisfies the expectations of spectators within the constraints placed upon the organising committee, given that all resources are to be managed by the organising committee itself. There are many different customer groups (e.g. spectators, athletes, Olympic Family, Volunteers etc) but for the purposes of this example we will focus on spectators. Spectators participate in a session within a venue and such participation can only be formally defined if one were to model the typical activities in which spectators are involved (shown in the diagram as the ‘why’ dimension). We are interested in those activities for which a service needs to be provided, either because it is expected by a spectator (e.g. access to ATMs) or it is imposed by some external agent (e.g. security). A service will consume some and the interplay between resource and anticipated level of service constitutes a major issue of negotiation between multiple stakeholders of the Organising Committee during the requirements definition phase. Requirements in this sense are not necessarily exclusively requirements
for some software or hardware system but in a wider sense any kind of resource, human, machine, procedural etc. These resources are managed by different functional areas of the organising committee for example security is responsible for the ‘mag and bag’ machines, catering for the food and drink outlets and so on, and their allocation and management will require some process.

Focusing more on the individual activities of the partial process model of figure 6 we can distinguish the following set of interrelated activities. Firstly, there is the process of queue management as shown in figure 7.

There are a number of quality-related parameters for this process. Specifically, two parameters, (a) the number of ‘entrance queue management staff’ and (b) the level of expertise of such staff shown as the number of ‘spectators per queue management staff per minute’. These two important parameters determine the average waiting time for spectators queued to enter a venue.

Each one of the subsequent activities related to security has quality factors associated with the level of service provided to spectators. In the interest of space we focus on the queue management process and on the way that quality factors for this activity will be handled by the proposed framework. The two quality factors involved are the cost and the response time of the process, and our goal is to find a trade-off between them. Table 2 shows the instantiation of the meta-model for this example.

Now consider the Spectators Checking process in figure 6. The main activities of this process are wand sweeping and item deduction if required. The problem here is to check the spectators as fast as possible keeping the cost of the process acceptable and enforcing the security rule stating that only the authorized staff can be assigned to wand sweeping activities. The involved quality factors in this case are the response time, the cost and preserving security. The cost of wand sweeping is influenced by the number of staff assigned to this task.

---

**Table 2. Queue Management Process**

<table>
<thead>
<tr>
<th>Activities</th>
<th>Quality Factors</th>
<th>Metrics</th>
<th>Metamodel elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectators Entry</td>
<td>Cost</td>
<td>Number of queue management staff</td>
<td>Attribute of the Spectators Entry instance in activity class</td>
</tr>
<tr>
<td>Response Time</td>
<td>(Average time for one spectator entry * nb spectators)/nb staff</td>
<td>Average waiting time for one spectator</td>
<td></td>
</tr>
<tr>
<td>Spectators Inspection</td>
<td>Cost</td>
<td>Number of inspection staff Number of Mag&amp;Bag machines</td>
<td>Attribute of the Spectators Inspection instance in activity class</td>
</tr>
<tr>
<td>Response Time</td>
<td>Average time for one spectator inspection/nb machines</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 3. Spectators Checking Process**

<table>
<thead>
<tr>
<th>Activities</th>
<th>Quality Factors</th>
<th>Metrics</th>
<th>Metamodel elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wand Sweeping</td>
<td>Cost</td>
<td>Number of wand sweeping staff</td>
<td>Attribute of the activity class</td>
</tr>
<tr>
<td>Response Time</td>
<td>(Average time for one spectator * nb spectators)/nb staff</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>Number of staff members that are not wand sweeping staff</td>
<td>Class Actor, association Security constraint,</td>
<td></td>
</tr>
<tr>
<td>Item Deduction</td>
<td>Cost</td>
<td>Number of Item deduction staff</td>
<td>Attribute of the activity class</td>
</tr>
<tr>
<td>Response Time</td>
<td>Average time for one spectator/ nb deduction staff</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The experimentation phase of our approach involves the use of different scenarios regarding the values of the two factors (a) cost and (b) response time. We will show hereafter the evaluation results for the queue management process. The cost is related to the number of volunteers (staff) deployed, each one of which costs approximately 100€ per day.

The first experiment shows the results of simulating the process with 8 staff (cost 800€ per day per session) with a mean waiting time of 26.7 minutes (see figure 8). The second experiment shows the results of simulating the process with 12 staff (cost 1200€ per day per session) with a mean waiting time of 6.7 minutes. The third experiment shows the results of simulating the process with 18 staff (cost 1800€ per day per session) with a mean waiting time of 1.1 minutes. The results of the experiments are summarized in figure 9.

VI. CONCLUSION AND PERSPECTIVES

In this paper, we have presented an approach to handle quality aspects of BPs during requirement engineering. This approach relies on the idea that quality requirements have to be dealt with during requirement engineering, where all the elements allowing quality evaluation should be defined. We assume that the BP is represented by a model which is complete, semantically and syntactically correct. Quality requirements are first defined, then the appropriate quality factors are identified, along with the metrics allowing to effectively evaluate these factors. In our work, quality requirements are captured by extending a BP metamodel with quality related information leading to a quality oriented metamodel. We have illustrated the instantiation of the metamodel using a real application and provided some evaluation results for two quality factors. This work is a first step towards building an open BP quality evaluation platform. The design and implementation of this platform will be addressed in future works, as well as the implementation of a library of quality evaluation services implementing the metrics corresponding to BP quality factors. Another perspective to this work is to deal with the quality analysis problem, consisting in analysing the quality measures to mine some relationships or dependencies between quality factors.

Figure 8. Response time for the queue management process with 8 staff members
REFERENCES


