Extending BPMN for supporting customer-facing service quality requirements

Extending BPMN for Supporting Customer-Facing Service Quality Requirements

Kawther Saeedi, Liping Zhao
School of Computer Science
The University of Manchester
Manchester, UK
saeedik@cs.man.ac.uk,
liping.zhao@manchester.ac.uk

Pedro R. Falcone Sampaio
Manchester Business School
The University of Manchester
Manchester, UK
P.Sampaio@manchester.ac.uk

Abstract—Service-oriented computing promises to create flexible business processes and applications on demand by dynamically assembling loosely coupled services within and across organizations. Quality requirements play a central role in service sourcing and, together with Service Level Agreements, facilitate service selection and measurement of service delivery effectiveness. This empowers customers to make better decisions when faced with multiple service offerings and varying service costs. However, existing business process modeling languages provide little support for quality requirements annotation and specification. This paper argues that quality requirements are a central aspect of business process modeling specification, and thus proposes to incorporate time, cost and reliability quality requirements as extensions to the Business Process Modeling Notation (BPMN). These quality requirements are evaluated based on analytical model using reduction rules. An example of online purchasing business process is illustrated to demonstrate the applicability of the proposed approach.

Keywords- business process, business process model, quality requirements, workflow, services composition, reduction rules

I. INTRODUCTION

Service-oriented computing (SOC) promises to create flexible business processes and applications on demand by dynamically assembling loosely coupled services within and across organizations [1]. The SOC paradigm supports rapid, low-cost, interoperable, and distributed development of software applications [1]. However, developing coarse grained applications composed at the business process abstraction level needs identification, validation and illustration of a business process using high-level modeling languages [2]. A business process model describes the workflow of activities within and across organizations. In a service-oriented context, a business process is exposed as a set of logically connected services [3] where a service performs an activity within a business process.

Customer satisfaction is a strong indicator of corporate success [4] because it helps retain the current customers and attract new ones [5]. Quality requirements serve as a major driver in service markets because they provide assurance towards service performance and also help to ensure customer satisfaction [6]. In a SOC context, service quality can be used to differentiate different services that provide the same functionality [7]. In addition, service quality features expressed in a Service Level Agreement (SLA)

have become a significant factor for customer trust, loyalty and retention [8] [9].

Yet, current business process modeling languages do not focus on capturing non-functional requirements such as quality requirements and service level agreement characteristics (SLAs), and consequently, business process modeling suffers from the following problems:

- It is difficult to capture quality requirements at the modeling stage because the main focus is given to capturing functional requirements [10]. This leads to missing key system requirements, delays in system development, inconsistent system specifications and increased costs due to fixing defects or mismatching requirements.
- It is impossible to differentiate different services that provide the same functionality taking into account service quality features to compare the different service offerings.
- Lack of coordination and cross-cutting concern assessment among the quality requirements for overall end-to-end business process execution which leads to potential conflicts among quality requirements and failure in achieving business process objectives.
- Lack of SLAs for services composition at the business process modeling stage create barriers to manage different SLAs for a business process if SLAs are not explicitly integrated into the business process modeling framework.

This paper aims to address the above problems by proposing to capture and represent quality requirements at the business process modeling stage. Specifically, this paper proposes an extension to BPMN (Business Process Modeling Notation) to support quality requirements. BPMN has been chosen because it has become a "de facto" notation for modeling business processes management solutions and web services [10]. The proposed extension will maintain the same abstraction level consistent with business process models created by BPMN and will focuses on a core set of quality requirements closely related to customers' preferences and commonly addressed in SLAs, which are time, cost, and reliability. This paper also proposes an analytical based quality requirement evaluation model for evaluating the qualities of an end-to-end business process.

II. QUALITY REQUIREMENTS

There exist a wide range of quality models, which vary according to the modeling concerns. Thus, each model addresses different set of quality requirements. For example, ISO 9126 QoS model focuses on software engineering product quality by addressing reliability, usability, efficiency, maintainability and portability quality requirements [11]; and IBM QoS requirements model focuses on network and real-time resource allocation by addressing availability, accessibility, integrity, performance, reliability, regulatory and Security quality requirements [12]. Yet, none of the quality models captures a complete set of quality requirements. In this paper, we focus on a core set of quality requirements, namely time, cost and reliability, and propose an extension to BPMN based on this set of qualities. Our focus is based on the following reasons. First, time, cost and reliability are common customer facing quality requirements that have a direct effect on customer satisfaction; second, these qualities are closely related to business goals and SLAs. Finally, these three qualities are commonly addressed in workflow evaluation literature [18] [33] [34] [35] [36]. In the following sections, we provide a detailed definition for these three quality requirements.

A. Time

The time taken to complete a specific request (service, task or activity) is a key indicator of service performance. Time is mainly measured by response time, throughput, or latency [17]. Throughput is the number of requests served in a specific period of time [17]. Latency represents the actual time between the sending of a request and receiving a response [17]. Response time is the maximum time required to complete a request [17]. In this paper, time is modeled as response time which is more intuitive for business process modeling and consistent with the level abstraction of business process models.

The response time (RT) contains three major components: Processing time, waiting time and data transfer time. The Processing Time (PT) is the actual time for processing. Waiting Time (DT) refers to the non-value-added time needed before processing such as queuing delays and the setup time [18]. Data Transfer Time (DT) is the time required to transfer data. Data transfer time is commonly required for data intensive tasks where large amounts of data are transferred. The transmission time is estimated based on the past execution of a request [19] or application test results.

Response time can be affected by different constraints, such as peak transaction (P) and maximum data transfer rate (MDT). Peak transaction is also known as peak time where the maximum number of concurrent transactions is reached. The maximum data transfer rate is the maximum bandwidth of data transfer rate. Table 1 lists the response time formula, its major components and constraints.

B. Cost

Cost is the amount of money that needs to be paid to setup, execute and monitor a task. It is essential to estimate the cost in order to guarantee the business financial plans [18] and goals fulfillment. Cost consists of enactment cost (EC) and realization cost (RC). The enactment cost is the cost associated with the deployment, management and monitoring of a business process [18] which is estimated by contract agreement. The realization cost is the cost associated with the task's execution such as labor cost, machine cost, direct material cost, and setup cost [18]. The realization cost is estimated based on a transaction, data transfer rate or pre-payment for a specified period of time. Table 2 summarizes the cost formula, components and constraints.

C. Reliability

Reliability is the ability to perform a promised task within a specified time [6] [17]. In other words, this requirement assures a reliable service delivery [12] and a reliable system [20]. Reliability is a technical measure related to hardware and software configurations [19]. Reliability is measured in terms of system failure rate (SF) and process failure rate (PF) [18]. System failures are related to information technology and software failures include operating systems, communication protocols and hardware. Process failures are related to business process exceptions which lead to an exceptional termination [18] such as login failure or expiry of session variable. Table 3 shows the reliability computations, attributes and constrains. Reliability is measured from the historical data about past executions [19] using the following expression [18]:

$$R = 1-(SF+PF)$$
Where
$$SF = \text{No. system failure / total No. of execution}$$
$$PF = \text{No. Process failure / total No. of execution}$$

$$\text{(1)}$$

TABLE I.   RESPONSE TIME FORMULA, COMPONENTS AND CONSTRAINTS

Response Time	Components	Constraints	Variables
RT = PT+WT+DT	Processing time (PT)	No. concurrent transactions (CT) <= P	PT: time CT:int P: int
	Waiting time (WT)	CT <= P	WT: Time CT:int P: int
	Data transfer time (DT)	TransferredData (TD) <= MDT	DT: time TD: byte MDT:byte

TABLE II. COST FORMULA, COMPONENTS AND CONSTRAINTS

Cost	Components	Constraints	Variables
C = EC + RC	Enactment Cost (EC)	A cost for each Execution (TC)	TC:int Currency:str
		*A contract cost (CC) for a specified period and number of tasks. *Executed Tasks (ETask) <= specified No. tasks(NoTask) *Execution time within the specified period	CC:int Currency:str Period: time ETask:int NoTask:int Execution: time
	Realization cost (RC)	Contract related cost such as maintenance term, installation charges, renewal charges and monitoring charges	RC:int Currency:str RenewT:time

TABLE III. RELIABILITY COMPUTATIONS, ATTRIBUTES AND CONSTRAINS

Reliability	Components	Constraints	Variables
R=1-(TF+SF)	ProcessFailure(TF)	CT <= P	TF: double TFNo: int Total ExNo: int P: int CT:int
	SystemFailure (SF)	CT <= P	SF: double SFNo: int Total ExNo: int P: int CT:int

III. BUSINESS PROCESS MODELLING NOTATION

Business process models are used for presentation, identification, validation, improvement and implementation of business process systems [2]. A business process model can be represented using a visual notation or formalism. Visual modeling, such as using UML modeling notations is generally regarded as easy to understand by humans. Formalism, such as WSBPEL[21], while providing software execution details, are difficult to understand by humans.

OMG released the BPMN standard in 2006, to bridge the technical gap between a visual model and a formal model of a business process. BPMN provides notations to business process designers and managers. In addition, provides a formal execution semantics for BPMN, and thus facilitates the mapping of business process models to an execution language [21]. At the human level, BPMN represents business process models in a flow-chart format, similar to UML's Activity Diagram, for flexibility and readability [21]. This simple and commonly used format enables BPMN to handle the complexity of business processes and also B2B collaboration concepts using different types of model such as processes, choreographies and collaboration models. In addition, BPMN supports the expression of

advanced procedural concepts such as exception handling, transactions, and compensation [21].

BPMN notational elements are divided into five basic categories: Flow Objects, Data, Connection Objects, Swimlanes, and Artefacts [21] as shown in Fig 1. Flow objects are the main graphical elements which define the behaviour of a business process; it includes events, activities and gateways. Swimlanes group modeling elements by representing organizational aspects which are denoted in business process diagrams by swimlanes via a two level hierarchies. Pools represent organizations or participants in B2B situations. Lanes are sub-partitions of the pool that represent organizational entities. Connection objects are used to connect BPMN elements. It consists of a sequence flow to specify the order for flow objects; a message flow to describe the flow between business partners in different pools and an association to link an artifact with an element of the diagram. Data objects represent the data element created, manipulated and used during the execution of a process. Data objects do not represent the structure of the data or query manipulation structure. A message is used to depict the contents of a communication between two Participants. Artefacts are used to show additional information about a business process; it includes Group and Annotation. Within the basic categories, there are additional modeling elements support for complex requirements without significantly changing the general appearance of the diagram [21]. Therefore, the reader of a BPMN diagram can easily recognize the basic types of element and understand the diagram [21].
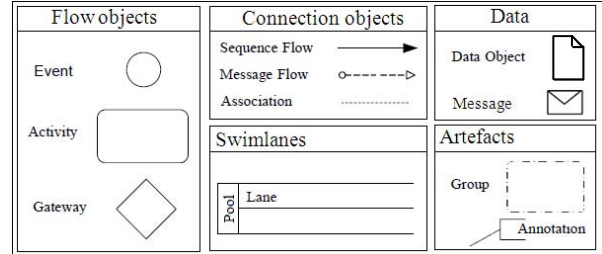

Figure 1. Elements for BPMN basic categories.

The current incarnation of BPMN models is notably lacking important business process concepts such as business rules, data, resources, quality requirements, business goals and organizational models. BPMN, however, provides guidelines to support notation extensions [22]. The extension guidelines in BPMN v1.0 [22] lack formal specification details, and as a result, different researchers have resorted to the use of different approaches and metmodels for extending BPMN. A standard metamodel is important to define the semantics of the graphical elements precisely, and to enable interchange of models [32]. BPMN v2.0 has introduced an extensibility mechanism which allows BPMN adopters to attach additional attributes and constructs to standard BPMN modeling elements [21]. The new extension constructs can produce more interchangeable business process models, because the standard elements can

interact and can be understood by other BPMN adopters [21]. This paper adopts BPMN v2.0 to support the aforementioned three quality requirements.

IV. SUPPORTING TIME, RELIABILITY AND COST IN BPMN

This section describes our proposal for an extension to BPMN to incorporate time, cost and reliability requirements into business process modeling. The proposal uses the extension specifications of BPMN v2.0 [21] because it provides formal specifications of extension constructs using class diagram for a metamodel representation. BPMN Extension consists of four different elements: Extension, ExtensionDefinitions, ExtensionAttributesDefinisition and ExtensionAttributesValue [21]. The ExtensionDefinition is independent of BPMN elements that defines and groups the extension attributes. The ExtensionAttributeDefinition defines new attributes which can be attached to BPMN element. The ExtensionAttributeValue represents actual extension attribute value. Finally the Extension element links an ExtensionDefinition and its attributes to a specific BPMN model definition [21].

Fig. 2 a) illustrates quality requirements extension metamodel class diagram for our proposal, adopted from BPMN v 2.0 [22]. The QualityRequirements class defines the quality requirements extension which refers to ExtensionDefinition class and groups the extension attribute classes of: Time, Cost and Reliability classes into the QualityRequirements class using composite relationship which illustrated further in Fig. 2 b). Each of these classes refers to ExtensionAttributesDefinition class that defines the extension related attributes. ExtensionAttributesValue classes illustrate the corresponding values and types linked to ExtensionAttributesDefinition classes using composite relationship. The Extension class is linked the QualityRequirements with the Definitions class that

contains object for all BPMN elements and inherit the attributes and model associations of BasicElement [21].

Activity elements represent units of work performed in a business process. The proposed extension targets the activity elements which includes Task, SubProcess and CallActivity [21]. To illustrate this relation in high level the QualityRequirements class is linked with Activity class using composite relationship as shown in Fig. 3 that implies each activity has quality requirements attributes. The extension is applied to all child classes of the activity class using inheritance relationship; this includes SendTask, ReceiveTask, ServiceTask, UserTask, ManualTask, ScriptTask and BusinessRuleTask.

The graphical representation of the extended notations should be simple, economic, consistent, non-ambiguous scalable, and visually intuitive. A table format is chosen to represent quality requirements and related values as shown in Fig.4. Fig. 4 a) represents the collapsed view of the activity with quality requirements. This gives a simple and aesthetic view of quality representation in BPMN. Fig.4 b) represents the detailed values of the quality requirements in a table to simplify reading the notations.

The quality requirements are identified based on measurements and data relating to the actual work performed by a process. To obtain this data, modellers need to know what requires quality measurement (i.e. need to know what the functional concerns are before work out the non-functional concerns). However, non-functional requirements are overlooked at the modelling stage because the modelling focus is concentrated on the functional aspects of the business process [10]. Therefore, we propose modelling the quality requirements while and after modelling the functional aspects. The implantation of this proposal will be undertaking using Eclipse process modeller software.
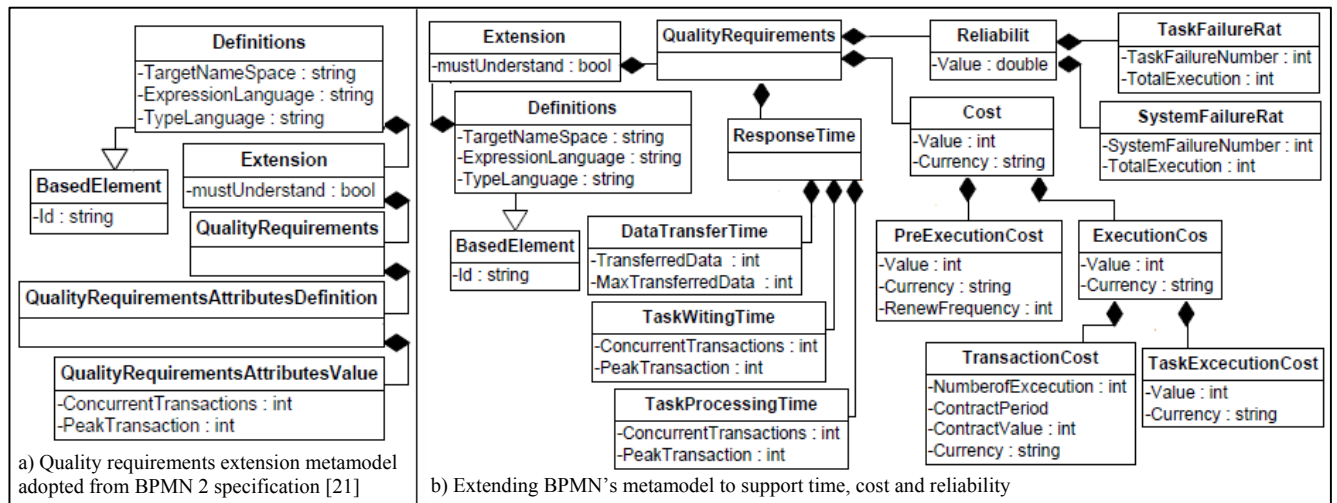

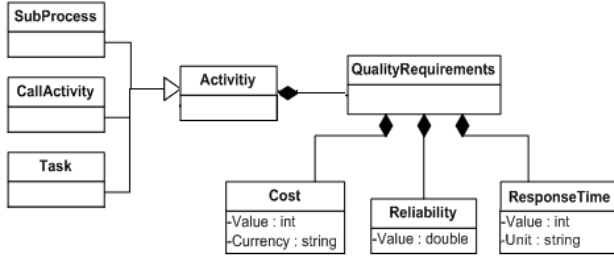Figure 2. Quality requirements extension metamodle class diagram.

Figure 3. Linking the activity element with quality metamodel.
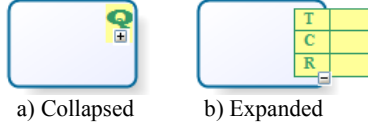

a) Collapsed   b) Expanded

Figure 4. Quality requirements representation.

V. EVALUATING BUSINESS PROCESS QUALITY REQUIREMENTS

Thus far, we have discussed the representation and measurement of quality requirements for individual activities. However, it is also useful to understand the quality requirements for the business process as a whole. Therefore, an evaluation model is required for aggregating the quality requirements of business process activities in order to quantify the quality requirements for the overall business process. The evaluation model and related formulae need to be matched with business process structural models as different workflow structures exist in a business process model.

This section proposes a method for determining the quality requirements for the overall business process. The evaluation models of workflow quality requirements fall into two main categories, simulation models [33] [34], and analytical models [33] [18] [34] [35]. This research uses analytical models to compute the overall business process quality requirement because it is a basic model and does not requires a tool for demonstration. Reduction rules is the common technique of the analytical models [35] [33] [34] [18]. There are six distinct reduction rules which are sequence, parallel, conditional, fault tolerant, loop and network [18]. The reduction rules are repeatedly applied to a business process until only one atomic task remains. The structure of business process changes each time a reduction rule applied. After a number of iterations only one task remains. This task contains the quality requirement metrics correspond to the overall business process [18].

The study focuses on four basic structural workflow models: sequence, parallel, conditional and loop [37], as they are the building blocks for constructing business processes.

Sequence: sequential workflow structure of a business process implying that the activities are executed in sequential order. The sequential reduction rule transfers the sequential activities A_i and A_j to a one activity Aij as shown in Fig. 5 A). The time and cost are computed by summing the attribute values of the sequential activities. The reliability rate is computed by multiplying the reliability rates of the sequential activities where the produced reliability rate is between 0 and 1. The quality attribute values are computed using (2):

$$\textbf{Time: } T(A_{ij}) = \sum_{i=1}^{j} T(A_i)$$

$$\textbf{Cost: } C(A_{ij}) = \sum_{i=1}^{j} C(A_i)$$

$$\textbf{Reliability: } R(A_{ij}) = \prod_{i=1}^{j} R(A_i) \tag{2}$$

Parallel: parallel workflow structure of business process implying that activities are executed in parallel. The parallel reduction rule reduced the n sequential tasks into one task A1n as shown in Fig. 5 B) and incoming activity Aa and the outgoing activity Ab remain the same. Parallel activities takes the time of the longest activity to reach the joining point, therefore the maximum time is used to compute the time attribute value. The cost and reliability are computed using the same formulae of sequential structure. The computation formulae of quality attribute values are shown in (3):

$$\textbf{Time: } T(A_{1n}) = \text{Max } \{T(A_{1n})\}$$

$$\textbf{Cost: } C(A_{1n}) = \sum_{i=1}^{n} C(A_i)$$

$$\textbf{Reliability: } R(A_{1n}) = \prod_{i=1}^{n} R(A_i) \tag{3}$$

Condition: conditional workflow structure of business process means one activity is executed based on a specified condition. The conditional reduction rule shown in Fig. 5 C) can be reduced to three sequential tasks where the incoming activity A_a and outgoing activity A_b remain the same. The computation formulae of quality attribute values are based on the occurrence probability of each activity. Time, cost and reliability are computed by summing the occurrence probability multiplied by the quality values as shown in (4):

$$\textbf{Time: } T(A_{1n}) = \sum_{1=i}^{n} P_i * T(A_i)$$

$$\textbf{Cost: } C(A_{1n}) = \sum_{1=i}^{n} P_i * C(A_i)$$

$$\textbf{Reliability: } R(A_{1n}) = \sum_{i=1}^{n} P_i * R(A_i) \tag{4}$$

Loop: loop workflow structure of business process means one or more activities are executed at specified number of times base on a specific condition. The loop

reduction rule in Fig. 5 D) shows that an activity Ai executed n number of times. The computation formulae shown in (5) of quality attribute values are based on the number of execution n. Time and cost are calculated by multiplying n by the attribute values. In the conditional loop the exact iteration count is unknown therefore the probability P_n can be used. P_n is the probability of executing the loop n times and P_n is equal 1 when n is predefined [37]. Reliability value does not change by the number of execution of the same activity:

$$\textbf{Time: } T(A_{in}) = (n*P_n) * T(A_i)$$

$$\textbf{Cost: } C(A_{in}) = (n*P_n) * C(A_i)$$

$$\textbf{Reliability: } R(A_{in}) = R(A_i)$$

(5)

VI. EXAMPLE

This section demonstrates the proposed concepts using an online purchasing business process scenario. The business process model described in Fig. 6 A) illustrates a high level model of the process activities. The figure also depicts service collaboration between online store and payment service to perform the payment activity. The quality attributes of time, cost and reliability are assigned to each activity of the business process. These values are estimated through observing several online purchase applications.

A) Sequential Process Reduction

B) Parallel Process Reduction

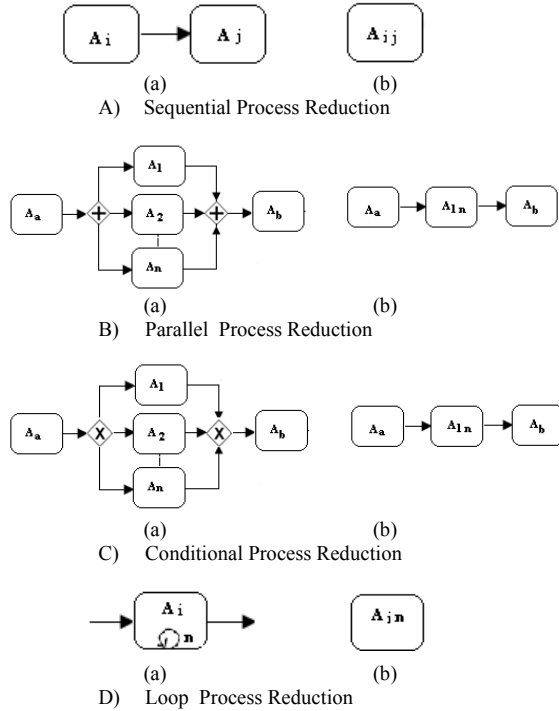C) Conditional Process Reduction

D) Loop Process Reduction

Figure 5. Reduction Rules for basic workflow structure

Three reduction rules are applied to evaluate the overall business process as shown in Fig. 6 B). Fig. 6 B) {1} illustrates the business process before applying a reduction rule. A sequential reduction rule is applied to the first two activities of Fig. 6 B) {1} Receive Order and Payment Service. The quality attributes values of the produced task after applying (2) is Time =5 sec, Cost =0.13$ and Reliability = 79% and the produced model is Fig. 6 B) {2}. A parallel reduction rule applied to Invoice and Order Delivery in Fig. 6 B) {2}. The quality attribute values of the produced task after applying (3) is Time =3 day, Cost =3.03$ and Reliability = 90% and the produced model is Fig. 6 B) {3}. Finally a sequential reduction rule is applied to Fig. 6 B) {3} the produced model and the quality attribute values of the produced task after applying (2) on the remaining activities is Time =3 days and 5 sec, Cost =3.16$ and Reliability = 71%. Fig. 6 B) {4} illustrates the overall business process quality attribute values.
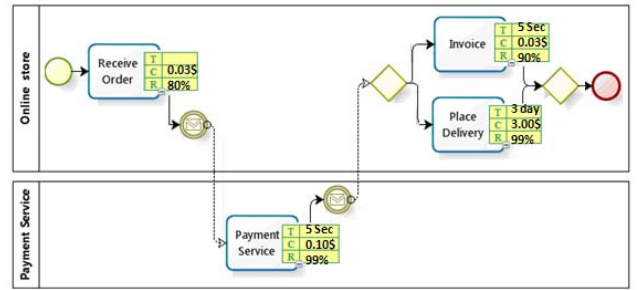
A) Quality requirements model.
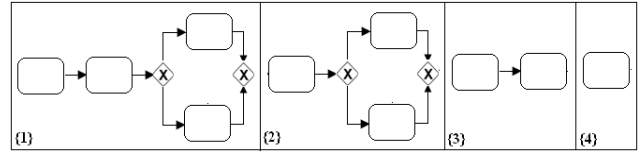
B) Reduction rules applied.

Figure 6. Qualiy requirments and Reduction rules for the online purchase business process example.

VII. RELATED WORK ON BPMN EXTENSIONS

BPMN provides guidelines to support notation extensions [22]. These guidelines enable researchers to extend BPMN with additional modeling features such as business goals and performance measures [23], cost and reliability requirements [25] [26], security requirements [24], data modeling [27], business process visualization [28], temporal constraints [29], human resource allocation and authorization constraints [30]; and choreography technical and implementation details [31]. This section discusses some of the closely related work to ours.

Korherr and List (2007) extended BPMN to model goals and performance measures by annotating Cost, Time and Quality values [23]. Their quality measures are expressed by customer satisfaction rate or customer complain rate. These

three elements complement the quality requirements we identified of time, cost and reliability for similar reasons which are affect customer satisfaction, support business goal and commonly addressed in SLAs. Korherr and List (2007) also proposed a metamodel for the additional elements and use labels to represent element values at process level [23].

Magnani and Montesi (2007) extended BPMN notation with a cost annotation which represents cost value, cost interval, or average cost [25]. They also proposed an overall business process cost evaluation that considers heterogeneous behavior of a business process [25].

In another proposal [26], cost and the reliability values were represented by textual annotation. Similar to our evaluation model, Path and Wiring (2009) also proposed four patterns for business process model evaluations that are sequential, parallel, conditional and successive possibilities [26].

Gagné and Trudel (2009) proposed Time-BPMN which extends BPMN with attributes and properties to capture the temporal constraints and dependencies of a business process. Time-BPMN concentrates on temporal constraints and dependencies which have visual depictions and control activity workflow in a business process [29]. Time-BPMN enhances the workflow description from a business process function perspective. Our proposal, on the other hand, addresses time from a non-functional perspective.

In addition, our proposal introduces a class metamodel in the same fashion as BPMN v0.2 extensibility feature to ensure that models can be interchangeable. Furthermore, our extension notation has a number of advantages. Its tabular representation can concisely and precisely show the quality requirements and it is flexible to configure other quantitative quality requirements. In addition, the use of table to define the extension allow BPMN adopters or modeling tools to recognize the extension values which then can be used for evaluations, validations and other operations. It is light-weighted and can be hidden or expand as appropriate which maintain the simplicity and general look of BPMN model.

VIII. CONCLUSIONS

This paper proposes and illustrates an approach for incorporating a set of quality requirements into BPMN. The aim of this approach is to support quality requirements at business process level to help the modeler choose appropriate services for business processes and aid the specification of SLAs. To maintain the same abstraction level as business process models, the proposed approach focuses on service quality requirements closely related to customers' preferences commonly addressed in SLAs, which are time, cost, and reliability. The paper also proposes an analytical based evaluation model, using reduction rules, for evaluating overall business process quality requirements.

This proposal, however, has several limitations. First, it only permits the quality requirements to be attached to the activity elements, not to other business objects. Second, the proposed evaluation method is only applicable to the basic four workflow structural models which are sequential, parallel, conditional, and loop. The method does not support other complex workflow models such as exception handling, transactions, and compensation [21]. Third, our proposal focuses only on the BPMN models and has not considered how to map quality requirements onto lower-level, executable models such as BPEL. These limitations will be addressed in our future work. In particular, we aim to develop a generic meta-model for representing and structuring quality requirements. Such a model will be able to support a large set of common quality requirements, beyond those illustrated in this paper.

REFERENCES

[1] M. P. Papazoglou, P. Traverso, S. Dustdar and F. Leymann, "Service-oriented computing: state of the art and research challenges," Computer, vol. 40, pp. 38-45, 2007.

[2] M. Weske, Business Process Management: Concepts, Languages, Architectures 1ed. New York: Springer-Verlag Berlin Heidelberg, 2007.

[3] J. Yan, R. Kowalczyk, J. Lin, M. B. Chhetri, S. K. Goh and J. Zhang., "Autonomous service level agreement negotiation for service composition provision," Future Generation Computer Systems vol. 23, pp. 748–759, 2007.

[4] A. Parasuraman, V. A. Zeithaml and L. L. Berry, "A conceptual model of service quality and its implications for future research," The Journal of Marketing vol. 49, pp. 41-50 1985.

[5] J. R. Evans and W. M. Lindsay, The Management and Control of Quality, 3ed, West Publishing Co, 1996.

[6] H. Li and J. Meissner, "Improving Quality in Business Process Outsourcing through Technology," unpublished|.

[7] D. Mukherjee, P. Jalote and M.G. Nanda, "Determining QoS of WS-BPEL compositions," Service Oriented Computing - ISOC 2008, vol. 5364, Springer-Verlag,Heidelberg, November 2008.

[8] Q. Du, C. Chi, S. Chen and J. Deng, "Modeling service quality for dynamic QoS publishing," in IEEE International Conference on Services Computing, 2008, pp. 307-314

[9] J. Collier and C. Bienstock, "Measuring service quality in e-retailing," Journal of Service Research, vol. 7, pp. 213-233, 2006.

[10] K. Pfitzner, G. Decker, O. Kopp and F. Leymann, "Web service choreography configurations for BPMN," Service-Oriented Computing - ICSOC 2007, vol. 4907, pp. 401-412, 2009 [Service-Oriented Computing - ICSOC 2007 Workshops, international Workshops, Vienna, Austria, September 17, 2007].

[11] B. Zeiss, D. Vega, I. Schieferdecker, H. Neukirchen and J. Grabowski, "Applying the ISO 9126 quality model to test specifications," Software Engineering, 2007, pp. 231-242.

[12] A. Mani and A. Nagarajan. (2002, 08/10/2009). "Understanding quality of service for web services," Available: IBM DeveloperWorks, http://www.ibm.com/developerworks/library/ws-quality.html

[13] A. Eleyan, L. Mikhailov, L. Zhao, "Quality-of-service support in web services architecture," Ingénierie des Systèmes d'Information (ISI), vol. 9, pp. 185-203, 2004.

[14] E. Wustenhoff, (2002, 18/11/2009), "Service level agreement in the data center". Available: Sun BluePrints™ OnLine http://www.sun.com/blueprints/0402/sla.pdf

[15] J. Myerson, (2004, 18/11/2009), "Use SLAs in a web services context, part 1: guarantee your web service with a SLA. Available: IBM developerworks http://www.ibm.com/developerworks/library/ws-sla/

[16] P. Bianco, G. A. Lewis and P. Merson, (2008, 20/11/2009), "Service level agreements in service-oriented architecture environments. Available: Carnegie Mellon University http://www.sei.cmu.edu/reports/08tn021.pdf

[17] S. Ran, "A model for web service discovery with QoS," ACM SIGecom Exchanges, vol. 4, pp. 1-10, 2003.

[18] J. Cardoso, A. Sheth, J. Miller, J. Arnold and K. Kochut, "Quality of Service for workflows and web Service Processes," Journal of Web Semantics, vol. 1, pp. 281-308, 2004.

[19] J. Zhang, W. Nie, M. Panahi, Y. Chang and K. Lin, "Quality driven web services composition," in 12th international conference on World Wide Web, Budapest, Hungary, 2003, pp. 411 - 421

[20] G. Weikum, "Towards guaranteed quality and dependability of information services," in 8th German Conference on Databases in Office, Engineering, and Scientific Applications, 1999.

[21] OMG. (2009), "Business Process Model and Notation (BPMN),FTF Beta 1 for Version 2.0." Available: http://www.omg.org

[22] OMG. (2006), "Business Process Modeling Notation Specification Virsion 1.0." Available: http://www.bpmn.org/

[23] B. Korherr and B. List, "Extending the EPC and the BPMN with business process goals and performance measures," in Ninth International Conference on Enterprise Information Systems, Madeira, Portugal, 2007, pp. 287-294

[24] A. Rodriguez, E. Fernandez-medina and M. Piattini, "A BPMN extension for the modeling of security requirements in business processes," Institute of Electronics, Information and Communication Engineers (IEICE) Transactions on Information and Systems, vol. E90-D, pp. 745-752, 2007.

[25] M. Magnani and D. Montesi. (2007), "Computing the cost of BPMN diagrams". Available: University of Bologna http://www.cs.unibo.it/pub/TR/UBLCS/2007/2007-17.pdf [Technical Report Unpublished]

[26] P. Sampath and M. Wirsing, "Computing the cost of business processes," in Third International United Information Systems Conference, Sydney, Australia, 2009.

[27] M. Magnani and D. Montesi, "BPDMN: a conservative extension of BPMN with enhanced data representation capabilities", CoRR, vol. 0907-19782009, 2009 [informal publication]

[28] M. Momotko and B. Nowicki, "Visualisation of (distributed) process execution based on extended BPMN," presented at the Proceedings of the 14th International Workshop on Database and Expert Systems Applications, 2003.

[39] D. Gagné and A. Trudel, "Time-BPMN," in IEEE Conference on Commerce and Enterprise Computing, 2009.

[30] C. Wolter and A. Schaad, "Modeling of task-based authorization constraints in BPMN," Business Process Management, vol. 4714, pp. 64-79, 2007.

[31] G. Decker, O. Kopp, F. Leymann and M. Weske, "BPEL4chor: extending BPEL for modeling choreographies. ," in International Conference on Web Services (ICWS), 2007.

[32] OMG. (2007), "Business Process Model and Notation (BPMN) 2.0 Request For Proposal. Available: http://www.omg.org/bpmn

[33] Y. Xia, H.P. Wang, Y. Huang and L. Yuan, "A stochastic model for workflow QoS evaluation," Scientific Programming, vol. 14 pp. 251-265, 2006.

[34] T. Lv, "Research on workflow QoS," presented at the International Joint Conference on Artificial Intelligence, Hainan Island, China, 2009.

[35] S. Hwang, H. Wang, J. Srivastava and R. A. Paul, "A probabilistic QoS model and computation framework for web services-based workflows," Conceptual Modeling – ER 2004, vol. 3288, pp. 596-609, 2004.

[36] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, Q. Z. Sheng, "Quality driven web services composition", in the 12th international conference on World Wide Web, Budapest, Hungary, ACM, 2003, pp. 411 - 421.

[37] Z. Zheng, M. R. Lyu, " Collaborative Reliability Prediction of Service-Oriented Systems", in the 10th ICSE, Cape Town, South Africa, ACM, 2010, pp. 35- 44.