BPEL Aided Framework for Constructing Monitoring Rules in Service Oriented Architectures

Plamen Petkov and Markus Helfert
School of Computing, Dublin City University, Glasnevin, Dublin 9, Ireland

Keywords: Service Oriented Architectures, BPMN, Monitoring Rules, Data Quality, Data Monitor, Business Rules, Rule Repository.

Abstract: Since the ascension of Service orientation architectures, data quality topic is becoming more and more extensive. Based on a data quality management approach, we propose a framework for analysing data produced by the composite service execution. In this paper we focus on the preparation stage, the stage where the quality is defined. Apart from other methods available which involve extracting or importing business rules into business processes, we propose technique for constructing monitoring rules which serves as a data quality arbiter. The process of composing is aided by the BPEL language. More particularly, service variables are obtained from the BPEL and WSDL repository and sets of operations are suggested to the business body. Eventually, monitoring rules are stored into a practical oriented repository.

1 INTRODUCTION

Service Oriented Architecture (SOA) has gained much acceptance in the past few years as the model for application development that uses groups of services that are orchestrated around business processes. SOA strives to provide existing functions of an information system (IS) as "services" that can be accessed in a loosely coupled way (Papazoglou, 2007), independently from the technical platform. Generally, in SOA, workflow or orchestration processes are fundamental. However, SOA comprises more than just the manner in which applications are deployed and cooperate with each other – it also involves how they interact with data.

Nevertheless, in more complex architectures, loosely coupled data can cause serious problems to the whole service composition and thus make it difficult to handle. In (Petkov, 2012) it was presented some data issues that mostly occur within SOA. There is no efficient way to managing such architectures without having the awareness of the data, processes and events running within the enterprise environment. This is why, vital assistant for proficient and effective deployment and operation of a SOA-based net-centric system is a comprehensive monitoring capability. Still, present monitoring solutions fall short with respect to such systems because they do not hold the capabilities to implicitly aggregate metrics, effectively detect inconsistent or inaccurate data, and so to provide comprehensive shared situational perception.

In this paper we propose an approach on how Business Process Execution Language (BPEL) in SOA can aid the business architects with building business rules that will eventually facilitate detection of inaccurate and inconsistent data. The approach follows the Data Quality Management (DQM) (Oracle, 2008) model. It focuses on constructing the business rules which is part of the preparation stage/defining the quality in DQM.

2 BUSINESS RULES IN IS

The related work in this area is very diverse and scopes huge area of business rules knowledge. Business rules deliver knowledge and this knowledge can be used to achieve different goals (Hay et al., 2003). For instance, a rule can be used to provide additional information to a business process and in this way to alter a data object or change values in a database. In the other hand, same rule can be used to verify the correctness of an informational object. As a result, two groups of rules can be distinguished: (1) group used to manipulate

Petkov P. and Helfert M..
BPEL Aided Framework for Constructing Monitoring Rules in Service Oriented Architectures.
DOI: 10.5220/0004368201410144
In *Proceedings of the 9th International Conference on Web Information Systems and Technologies* (WEBIST-2013), pages 141-144
ISBN: 978-989-8565-54-9

information, usually by bringing extra value to it; and (2) group of rules that access and monitor existing information without changing it.

When taking into account the relationship between business rules and business processes, literature identifies two research streams – (1) one which targets deriving and discovering business rules from business processes (Vasilecas & Normantas, 2011), (Ramsey & Alpigin, 2002), (Earls et al., 2002), and (2) another that aims integrating the BR into the BP. (Ly et al., 2008)

In contrast with the approaches above, our goal is not concerned to discover or derive business rules, neither to integrate business rules into business processes. It aims to assist the business bodies and architects with building rules, leaving the flexibility business ultimately to decide quality of data and in this way achieving the main goal which is to detect inaccurate data within SOA.

3 CONSTRUCTING QUALITY RULES

As previously mentioned, business rule can be used to verify a value of data and in this way to be used as part of monitoring process. As it was outlined in (Wand Y, 1996) data must be persistent in order to be evaluated. An example of monitoring rule in SOA context is presented by statement (1):

```
IF NOT (
SrvA.Oper.GetCustStatus = 'Gold'AND
SrvB.Oper.GetCartQuty > 10 AND
SrvC.Oper.GetShippingCost = 0 )              (1)
THEN
GENERATE_RULE_EXEPTION ERROR
```

At this stage, the left side (before THEN) of the aforementioned statement can be brought down to evaluating sets of simple predicates. The final result would be evaluated by using logical conjunction (Hazewinkel, 2001). In logical conjunction, the result will be true if all of its operands are true, otherwise the value is false.

In the general case the complete rule expression for monitoring purposes could be exemplified in the following way:

```
IF NOT (
Predicate_1 BITWISE_OPERATION
Predicate_2 BITWISE_OPERATION
. . .                                        (3)
Predicate_N )
THEN
GENERATE_RULE_EXEPTION ERROR
```

where `Predicate_1 . . . Predicate_N` are predicates linked with Bitwise Operations. Each predicate is composed of three objects - [returned_value] *comparison_operation* [expected_value] where [returned_value] is the value delivered by the target service operation (usually fetch function e.g. `getOperation()`); *comparison_operation* is any of the following logical operations: '<'(less), '>'(greater), '='equal, '!='(not equal), '<='(less than or equal) '>='(greater than or equal); [expected_value] is expected value, the one to be conditioned, usually defined by the user; `BITWISE_OPERATION` is any of the Boolean algebra operations (Koppelberg, 1989): '∧' (AND, conjunction), '∨' (OR, disjunction), '¬' (NOT, negation) . The whole predicate set must be inverted using negating Boolean logical operation (NOT), if the logic is positive. If negative logic is used, negate operation must be omitted after predicates are been evaluated.

4 FRAMEWORK FOR COMPOSING BUSINESS RULES AID BY BPEL LANGUAGE

Framework for composing quality rules is part of overall Data Quality Management (DQM) process which identifies four main stages in data quality cycle - Quality Assessment, Quality Design, Quality Transformation and Quality monitoring. Since DQM provides us only with general guidance how to manage quality of information, the framework we propose aims to show in detail how to define quality of information in SOA. The framework is focused particularly on the 'Quality Design' of DQM. We name this stage 'Preparation stage'. This is the stage were the quality is defined. It is comprised of four steps: (i) selecting BP; (ii) extracting the services and variables involved into the chosen business process or sub-process; (iii) constructing business rules based on the services and variables involved in the chosen BP; (iv) mapping and saving created rules into repository; The preparation stage is depicted on Figure 1.

The approach we propose aim to construct quality rules which serve as data quality arbiter, and that construction is aided by BPEL language. In order to achieve the goal, we incorporate into our framework some of the vital building blocks in SoA such as Business Process Engine (BPE), Enterprise

Service Bus (ESB), WSDL Repository and Business Processes' Repository (part of BPE).

Selecting a Business Process. In this step (Figure 1 – (i)) a business process or sub-process is selected by an authorised person. Based on the selection made, later in the preparation process, objects and function names are extracted in order to aid constructing of the business rules. Hence, it is necessity business person to be familiar with the selected process its activities. It is important that the scope of the process is well defined as this will affect construction of the rules later in preparation stage.

Extracting Services' Names, Operation and Variables. In extraction phase, the business process chosen from previous step undergoes series of analysis performed by business process reader. Ultimately specific meta-data (data about the process) is obtained and temporary stored by the rule builder (Figure 1(ii)). This meta-data includes 'service name', 'operation', 'input' and 'output' variables. In Service-oriented Architectures business process usually is described with Business Process Execution Language (BPEL). It is a powerful tool build on XML technology stack that provides a rich vocabulary for description of business processes. The vocabulary includes tags for describing primitive activities – <invoke>, <receive> as well as structural activities <sequence>, <flow>.

For the purpose of the extraction, BPEL file containing the prescription of the selected process is opened for examination. The process of analysing consists of scanning the file and searching for BPEL attributes that correspond to the metadata that need to be collected. The data needed for next step are 'service name', 'operation', 'input' and 'output'. The representative BPEL attributes are as follow: 'partnerLink', 'operation', 'inputVariable' and 'outputvariable'. The implementation of the BP Reader is intentionally omitted due to the scope of this paper.

Building up the Rules. This stage focuses on actual composing of the rules. The actual construction of the rules is done by business body – architect or administrator. The metadata stored temporary form BP reader in extraction stage then is referred to WSDL repository and files containing the relevant variables are opened (iii). Then a list of services along with their data/read functions is presented to the user. Next the business administrator/architect composes the rules using the methodology we described in detail in Section 3. In this way business body is allowed to compose simple rules – composed only of one statement or complex rules – combination of statements linked with logical operators in specific order.

Mapping Rules with Processes and Storing them into Repository. The composed quality rules are stored (iv) in an external repository. In order to maintain their execution in loosely coupled way later in the *execution stage*, we introduce a holistic way to store the rules into XML files. Hence we propose a suitable template. The template is presented below along with an example:

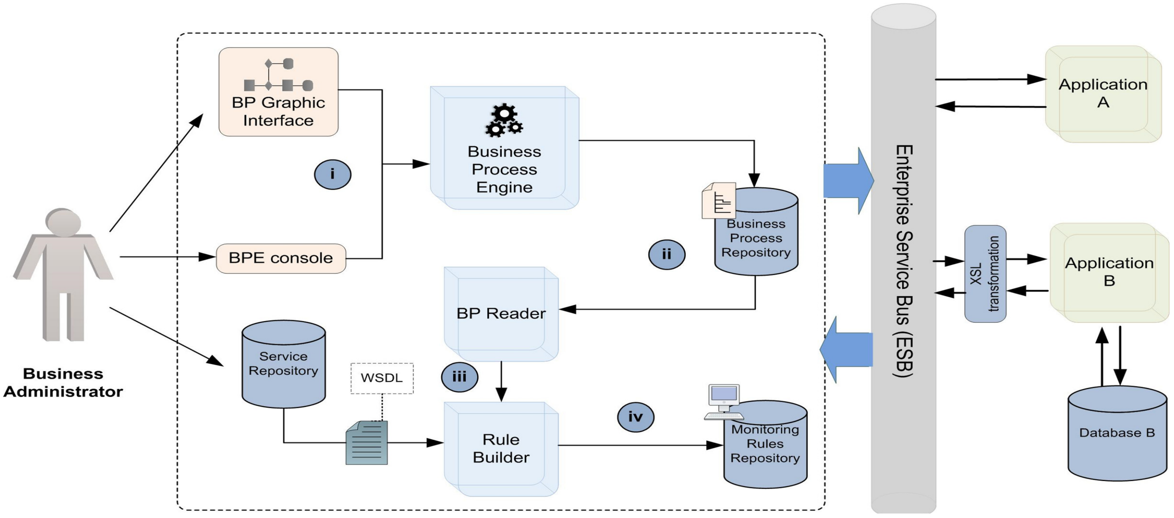

Figure 1: BPEL aided framework.

```
<ruleList>
 <ruleSet id = “23”
  name = “free_delivery”
  processId = “12” >
  <rule id = “1” name= “gold_cust”>
  <description>Check if customer status
  is gold</description>
   <serviceName>custSRV</serviceName>
   <portType>customerDb</portType>
   <serviceOperation>getCustomerStatus
   </serviceOperation>
   <inputVariable>332</inputVariable>
   <comparator>=</comparator>
   <expectedVariable>'GOLD'
   </expectedVariable>
   <ruleLinks>
    <ruleLink>
    <linkedRuleId>10<linkedRuleId>
    <booleanOperation>
    AND
    <booleanOperation>
    </ruleLink>
   </ruleLinks>
    ...
 </ruleSet>
</ruleList>
```

It is necessary to notice few important features that this template offers: First by storing identification number of the process into the rule record will avoid data ‘dead locking’ later in the execution step. Second, introducing ‘`<ruleLinks>`’ tag will allow building complex set of rules by storing information about their relations e.g. logical AND, OR , XOR operations.

5 CONCLUSIONS

Inspired from research in the area of data quality and service oriented architectures, in this paper we have presented a BPEL aided framework for constructing monitoring rules in web service composition and execution. More specifically, we have divided the framework into four phases, namely ‘selecting business process’ and ‘extracting services’ names operation and variables’, ‘constructing the rules’ and ‘mapping and saving the rules into external repository’.

Apart from other well-known methods for directly extracting or integrating business rules form and into business process, our approach differs in the way by providing the business bodies with a holistic solution that will aid them with building rules which will serve as data quality arbiter. The approach uses BPEL language to extract services names and operation and then with backend by WSDL library offers a list of operations that aid the business architect witch composition of the rules.

ACKNOWLEDGEMENTS

This work was supported by the Irish Research Council (IRC) under Postgraduate Scholarship scheme.

REFERENCES

Earls, A., Embury, S. & Turner, N., 2002. A method for the manual extraction of business. *BT technology journal*, 20(4), pp.127-45.

Hay, J., Kolber, A. & Anderson, K., 2003. *Defining Business Rules - what are they really*. http:// www.businessrulesgroup.org/first_paper/BRG-whatisBR_3ed.pdf [Accessed November 2012].

Hazewinkel, M.., 2001. Conjunction. In *Encyclopedia of Mathematics*. Springer.

Koppelberg, S., 1989. General Theory of Boolean Algebras. In J.D. Monk & J.D. Bonnet, eds. *Handbook of Boolean Algebras*. Amsterdam.

Ly, L., Rinderle, S. & Dadam, P., 2008. Integration and verification of semantic. *Journal of Data & Knowledge* , 64, pp.2-23.

Oracle, I., 2008. Understanding Data Quality Management. In *Oracle® Warehouse Builder User's Guide*. Oracle Inc.

Papazoglou, M. P., Traverso, P., Dustdar, S. & Leymann, F., November 2007. *Service-Oriented Computing:State of the Art and Research Challenges*. IEEE Computer.

Petkov, P. H. M., 2012. Data oriented challenges of service architectures a data quality perspective. In *CompSysTech*. New York, NY, 2012.

Ramsey, F. & Alpigin, J.i., 2002. A simple mathematically based framework for rule. In *26th Annual International Conference In Computer Software and Applications*., 2002. IEEE.

Vasilecas, O. & Normantas, K., 2011. Deriving business rules from the models of existing information systems. In *12th International Conference on Computer Systems and Technologies*. New York, NY, 2011. ACM.

Wand Y, W. R., 1996. Anchoring Data Quality Dimensions in Ontological Foundations. *Communications of the ACM*, pp.86-95.