From BPMN 2.0 to the Setting-Up on an ESB – Application to an Interoperability Problem

Y. Lemrabet, D. Clin, M. Bigand, and J.-P. Bourey

Univ Lille Nord de France, F-59000 Lille, France
Laboratoire de Modélisation et de Management des Organisations,
Ecole Centrale de Lille, BP48
59651 Villeneuve d'Ascq cedex, France
Youness.Lemrabet@centraliens-lille.org,
{David.Clin,Michel.Bigand,Jean-Pierre.Bourey}@ec-lille.fr

Abstract. To solve interoperability problem from semantic level, we propose to contribute to orchestration of the business processes to implement a mediation based on Enterprise Service Bus (ESB). We show how to take advantage of the forthcoming version of *Business Process Modeling Notation 2.0* (BPMN 2.0) of the *Object Management Group (*OMG) within the framework of a *Services Oriented Architecture (*SOA) development. This new version of BPMN is characterized by the addition of the notion of private/public process, choreography and conversation which are essential components of SOA architecture. We propose a model driven approach using an intermediate BPMN 2.0 model stemming from the high-level Modeling of the business processes defined from the industrial partner's requirements, to implement the solution on an ESB. The results, within a project whose objective is to allow the *Small and Medium Enterprises* (SME) to benefit from a collaborative interoperable architecture, allows the definition of domain process interfaces.

Keywords: Model Driven Interoperability, ESB, BPMN2.0, SOA.

1 Introduction

Each firm needs to bring its *Information System* (IS) into line with its business processes in order to improve its efficiency, its reliability and to rationalize its development's costs.

Moreover, the rapid evolution of technology means that enterprises need to advance ever faster, and they must establish collaborative networks, such as Extended and Virtual Enterprises, so as to be able to take advantage of their core competences and to look for others that they do not have [1]. So, enterprises have to fit their functions and processes with each other's in order to improve their competitiveness and to take advantage of new market opportunities; it is an interoperability problem whose issue is to save time and to make the exchanges reliable.

L.M. Camarinha-Matos et al. (Eds.): PRO-VE 2010, IFIP AICT 336, pp. 722–729, 2010.

Both Business Process and Enterprise Modeling techniques and methods have been successfully used by enterprises to integrate their information and manufacturing systems throughout the last few decades. More recently, European research programs, like Athena and InterOp[1], have been developed to improve the companies' interoperability.

Traditionally, researches on Interoperability tend to focus on one aspect of interoperability: (i) businesses; (ii) processes; (iii) services; and (iv) data interoperability. But in order to develop a practical solution for enterprise, interoperability problem should be addressed from both organizational and technical viewpoints. Businesses and processes interoperability are considered mainly at the organizational level, whereas services and data interoperability require focus on (information) technology issues [2]. In this paper, we are interested in solving the interoperability problem within the framework of the ASICOM[2] project.

ASICOM project will provide SMEs in both trade and logistics sectors with a pragmatic and generic approach to set up more rapidly simplified interoperable and adaptable solutions for improving communication with their partners (dematerialization).

The ASICOM project aims to support the development and realization of systems based on four SMEs requirements. The first requirement was the use of open-source solutions and frameworks to reduce the costs for SMEs. The second one was the respect of standards to facilitate the interoperability and communication at both the modeling and the technical levels, including the semantics aspects. The third one was the capacity of solutions to be deployed in a heterogeneous environment to improve communicability between exiting information systems already implemented in the SMEs. The last requirement was the agility of the proposed approach: indeed, due to the constant evolution of the technology, of the Customs regulations, of the trade partnerships it is needed to keep traces between the business level and the *Information Technology* (IT) level.

In order to meet these requirements, the following decisions have been made:

- A model driven approach was chosen. This kind of approach based on separation of concerns, modeling techniques and model transformations makes it possible to build models at different abstraction levels (from Business to IT) and to establish links between these representations.
- To meet the second requirement, standard notations and languages such as BPMN[3], *Business Process Execution Language* (BPEL)[4], *Unified Modeling Language* (UML)[5], *eXtended Markup Language* (XML)[6], and *Web Service Description Language* (WSDL)[7] … were selected.
- A SOA based solution was preferred in order to make easier the integration of heterogeneous existing systems. It also encourages reusability thanks to the service approach.

[1] http://interop-vlab.eu
[2] French acronym for Architecture of Interoperable Information System for Trade Industry.
[3] http://www.omg.org/spec/BPMN/2.0
[4] http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf
[5] http://www.omg.org/spec/UML/2.2/
[6] http://www.w3.org/TR/2008/REC-xml-20081126/
[7] http://www.w3.org/TR/2008/REC-xml-20081126/

- The open-source Petals ESB from the Petals SOA Suite[8] was chosen to implement the proposed service oriented approach. Some elements of comparison with other historical integration platforms are given in section 3.4.

The paper is structured as follows: the first section describes the context of the study and identifies the challenges addressed by the ASICOM project. The second section outlines the methodological and technical solution developed in the project. Finally the third section presents the industrial use cases wherein the developed technologies are tested and demonstrated.

2 Methodological and Technical Aspects

2.1 Model Driven Interoperability

Solving interoperability problems not only at the code level but also starting from a higher level of abstraction is a challenge that was one of the objectives proposed by InterOP NoE, and more precisely by Task Group 2 (TG2), in order to search for solutions to achieve interoperability following a model-driven approach.

The aim of TG2 has been to analyze and propose guidelines and methods that can help to solve the interoperability problems of *Enterprise Software Applications* (ESA) starting out from the enterprise models level and using an approach based on *Model Driven Architecture* (MDA) [3]. This method is called *Model Driven Interoperability* (MDI) [4]. The work of TG2 first focused on the models and transformations to be performed at the *Computation Independent Model* (CIM) level from the theoretical point of view. At this level the GRAI method was chosen as means to capture the enterprise models at the highest level of abstraction and BPMN (for business process view) and UML (business data view) were selected to play an interface role between enterprise models and IT models. To address the feasibility of its proposal, TG2 has tested it with transformation tools [4].

2.2 BPMN 2.0

BPMN is an OMG specification providing a notation for modeling business processes. The initial objective of BPMN was to give a picture of the business processes that can be shared between the different stakeholders, and that can be transformed in an execution language in order to be executed (on a workflow tool or using an ESB orchestration tool for example). It can be used by the business analysts who create the initial drafts of the processes, by the technical developers in charge of the implementation of the technology that will perform those processes, and finally, by the business people who will manage and monitor those processes.

The new version, BPMN2.0 which adoption is planned by end of June 2010, proposes numerous improvements of the previous version among them:

- Of course, the resolution of previous version's inconsistencies and ambiguities;
- A better formalization of the execution semantics for all BPMN elements;

[8] http://www.petalslink.com/societe/societe-metiers

- A mapping of a subset of BPMN model to an execution language of BPM Systems (WS-BPEL 2.0);
- A definition of new diagrams: Conversation Diagram, Choreography Diagram...

These two last points are very important evolution in the framework of a SOA-based development. Indeed private and public processes are useful to give the specification of what is visible or not from other participants. Moreover a Conversation Diagram which provides a "bird's eye" perspective is composed of a set of pools, represented as "black" boxes, exchanging messages grouped into a Conversation. A Conversation will ultimately be executed through an orchestration Process. In contrast to orchestration Choreography Diagram provides a multi-party perspective of a Conversation and therefore it focuses rather on information exchange (Messages) between participants. It makes it possible to derive the Process interfaces of each participant's process. As it will be presented in the following sections, a choreography Diagram gives a well adapted specification of what will be executed within an ESB orchestration engine.

2.3 Service Oriented Architecture

Service oriented architecture (SOA) is a way of organizing and understanding organizations, communities and systems to maximize agility and scale, it is also seen as an enabling factor for enterprise interoperability. SOA is not a technology nor a software solution but an architectural and organizational approach. The choice of technologies and tools is secondary [5].

Service oriented architecture is a paradigm that utilizes services as fundamental elements to develop applications [6], it place the service concept as the primary mean to achieve enterprise strategic objectives [7]. Not only Service is considered as the unit construction of system, but also each service is reusable, shared and loosely-coupled with the other services.

From the technical perspective SOA services can be considered as "functions" that are accessible across a network via well-defined interface and to implement technical services we need an SOA integration platform.

2.4 SOA Integration Platform: ESB

Integration platform must allow an independent mechanism to integrate disparate systems to ASICOM. SOA integration platform is the technical part of SOA that enable interoperability. Thus it must provide the capability to mediate, transform, route and transport service request from the service consumer to the service provider.

The application integration concept is nothing new. We have been dealing with this mechanism since we have had to connect more than two business systems. Several technical solutions have been proposed to address the problem of integration: *Extract Transform Load* (ETL), Middleware, *Enterprise Application Integration* (EAI), *Enterprise Service Bus* (ESB. The ASICOM platform must allow several participants with different ITs to communicate with each others. But traditional middleware provides only a point-to-point solution between systems.

ETLs are used to integrate data from multiple sources and integrating this data into a central warehouse. These tools are not a real time process; They operate in a Batch

mode, so they are not adapted to our needs. Moreover they don't support the mechanisms used to manipulate services (example: orchestration, etc.).

EAI reached its peak in the 1990s [8]. This technology was a response to the demand of enterprises that need to share their data and process without having sweeping changes to their applications and data. With EAI all integration is done in a hub, for each system it is enough to provide and maintain only one adapter to the hub instead of a specific adapter for each of the systems with which it communicates [9]. The downside of EAI is twofold. First EAIs products are proprietary and expensive. Second they are not based around standards.

An ESB is an architectural pattern of middleware that enable standards-based integration between applications and services. It's provides a broad set of capabilities. Firstly it's decoupling the interactions between the customers and the providers: services are accessible independently of implementation and location, they may be accessed by the service consumer in the same way. Secondly it's allows services to be exposed consistently across multiple communication protocols like *Simple Object Access Protocol* (SOAP)[9], *Simple Mail Transfer Protocol* (SMTP) [10], etc. Other functionalities are supported through the ESB such as transformation with XSLT[10], routing and security.

3 Application to ASICOM

Several service composition approaches have been proposed to address the problem of interoperability but none of them cover all the aspects of the interoperability. The majority of the approaches mainly focuses on the services and processes design. A major business opportunity and research challenge is to delivering executable services from models to the end-users [2].

In this section we combine the benefits of different approaches (Praxem Institute[11], SOMA[12]) to provide a method with a collection of comprehensive techniques for the model driven engineering of service-oriented landscapes. The ASICOM project focuses more precisely on relations with Customs to facilitate procedures such as on-line clearance of goods, payment of customs duty, and management of bonded warehouses…

3.1 ASICOM Model from BPMN 1.2 and BPMN2.0

The example we introduce describes the process of discharging the end of the Community transit procedure used for external Community transit document named *T1*[13]. The process is triggered by the goods arrival at the stockist (See figure 1).

In this orchestration the central process (ASICOM process) takes control of the involved partners and coordinates the execution of different operations. The involved partners do not need to know that they are taking part in a higher level business process. Only the central coordinator of the orchestration is aware of this goal.

[9] http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
[10] http://www.w3.org/TR/xslt
[11] http://www.praxeme.org/DocumentsGeneraux/Praxime-LivreBlanc_v2.SLB02.pdf
[12] http://www.ibm.com/developerworks/library/ws-soa-design1
[13] http://ec.europa.eu/taxation_customs/resources/documents/customs/procedural_aspects/transit/common_community/transit_manual_en.pdf

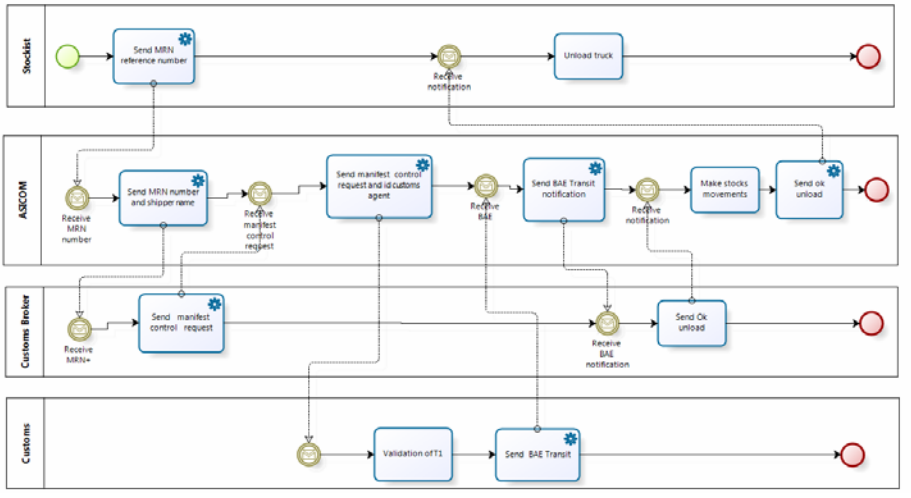

Fig. 1. Discharge T1 when goods arrive to the stockist orchestration diagram (BPMN 1.2)

Among its new features, BPMN 2.0 introduces choreography diagrams. Choreography process differs in purpose and behavior from an orchestration process. It focuses only on the exchange of information between the Participants (figure 2).

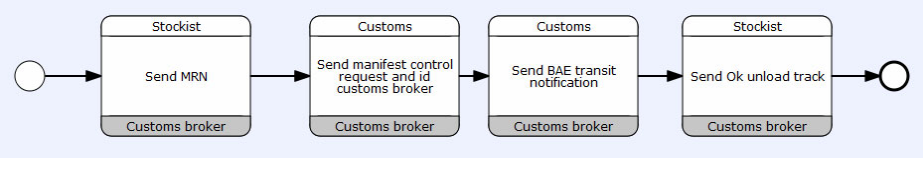

Fig. 2. Discharge T1 when goods arrive to the stockist choreography diagram (BPMN 2.0)

3.2 Deployment onto Petals ESB

BPMN specifications explicitly suggest BPEL to be used for the execution of business process. So after the description of the BPMN we refine it with the implementation details using BPEL. Then we use the orchestrations diagram to create the BPEL process (The BPMN 2.0 specifications precise that implementation is not expected to support directly Choreography Modeling elements).

In this scenario partners are represented by Web services, which have been developed without any transformation. The BPEL Process receives a request from the stockist. To fulfill it, the process invokes the involved Web services of the customs broker and customs sequentially and then responds to the original caller (figure 3).

We have chosen an Enterprise Service Bus to support the ASICOM SOA infrastructure. ESBs are the latest Middleware technology so they well integrate the principles of SOA. They provide an infrastructure that removes the direct connection dependency between service consumers and providers. We have opted for Petals ESB, owned by the Petals Link open source SOA solutions provider.

Fig. 3. Discharge T1 when goods arrive to the stockist BPEL diagram

The BPEL process is packaged in a Service Assembly component and deployed into Petals ESB, which uses the BPEL Service Engine component to run the BPEL process. The figure 4 shows the target architecture of the ASICOM platform that supports the execution of business process.

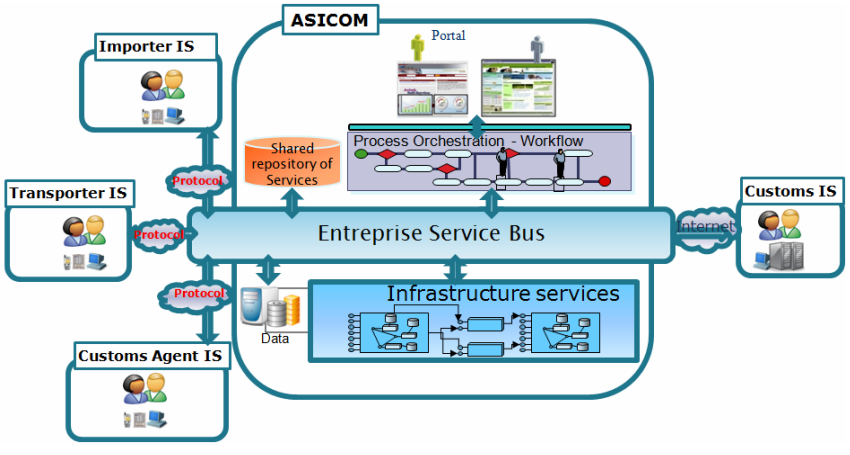

Fig. 4. ASICOM platform Architecture

4 Conclusion

The proposed method is a top-down realization of SOA based on a model driven approach. It uses BPMN at business level to describe the semantics of processes; then a mediation pool is added that can be transformed into a choreography diagram and that allows the highlighting of the partners services. Besides the BPMN process is refined with the implementation details using BPEL. And finally the BPEL is packaged and deployed as a BPEL Service Assembly on the Petals ESB. BPMN2.0 interest is to give more synthetic and so readable diagrams.

Our current works concern models transformations and the emerging standard SoaML [11] to support the activities of service modeling and the central aspects of SOA. SOA will be used to expose reusable services from the ASICOM infrastructure applications, and BPM technology will be used to consume those services by combining

them together as steps in a business process. The proposed method is being currently implemented within the SMEs context of ASICOM and is not limited to SMEs nor to trade industry. We are also applying it within aeronautics domain to solve interoperability problems between a manufacturer and its subcontractors for the design of aircraft parts.

Acknowledgements. This work was partially funded by the ASICOM project. This project started in April 2008 was approved by two French poles of competitiveness: PICOM[14] in Trade Industries domain and Nov@log[15] in Logistics domain.

References

1. Tae-Young, K., Sunjae, L., Kwangsoo, K., Cheol-Han, K.: A modeling framework for agile and interoperable virtual enterprises. Comput. Ind. 57(3) (2006)
2. Rodrigo, M.P., Eduardo, S., Marten, V.S., Dick, A.C.Q., Luís, F.P.: Enterprise Interoperability with SOA: a Survey of Service Composition Approaches (2008)
3. OMG. : MDA Guide Version 1.0.1 (2003), http://www.omg.org/docs/omg/03-06-01.pdf
4. Bourey, J.-P., Grangel, R., Doumeingts, G., Berre, A.: INTEROP NoE: DTG2.2: Report on Model Interoperability (2006), http://interop-vlab.eu/ei_public_deliverables/interop-noe-deliverables/tg2-model-driven-interoperability
5. Marks, E., Bell, M.: Service-Oriented Architecture: A planning and Implementation Guide for Business and Technology, 375 p. Wiley Editions, Chichester (2006)
6. Kim, H.-k.: Modeling of Distributed Systems with SOA & MDA. IAENG International Journal of Computer Science, IJCS 35(4), 35-4-10 (2008), http://www.iaeng.org/IJCS/issues_v35/issue_4/IJCS_35_4_10.pdf
7. Eloudrhiri, S., Halin, P., Monti, V.: Urbanisation et SOA Vers une Entreprise Agile version 1, Centre de compétences TIC, Walloon Region, Belgium (2008)
8. Davies, J., Schorow, D., Ray, S., Rieber, D.: The Definitive Guide to SOA: Oracle Service Bus, 2nd edn. Apress (2008)
9. Josuttis, N.M.: Josuttis.: SOA in Practice the Art of Distributed System Design. O'Reilly, Sebastopol (2007)
10. Postel, J.B.: RFC821 - Simple Mail Transfer Protoco Information Sciences Institute University of Southern California 4676 Admiralty Way Marina del Rey, California 90291 (213), pp. 822–1511 (1982)
11. SoaML, Service oriented architecture Modeling Language (SoaML) – Specification for the UML Profile and Metamodel for Services (UPMS). Revised Submission, OMG document: ad/2008-11-01 (2008), http://www.omg.org/docs/ad/08-11-01.pdf

[14] http://www.picom.fr/
[15] http://www.novalog.eu/index.asp