USING MDA FOR INTEGRATING WEB SERVICES IN THE WORKFLOW SYSTEMS

Mounira Zerari *, Mahmoud Boufaida *

* LIRE Laboratory, Mentouri University of Constantine, Algeria.
{Zerari.mounirabis, boufaida_mahmoud}@yahoo.fr

ABSTRACT

Workflow Management Systems (WfMS) support the definition, management and execution of business processes inside organizations. However, they suffer from lack of flexibility. They are inappropriate to support easily some mechanisms such as cooperative interactions and exchanging data. Indeed, one of the most important requirements emerging from new workflow applications that must be permanent for the process definition is the ability of communicating with outside organizations (partners, clients, and suppliers) and incorporating them in a process. In the last years, Web Services are emerged as a major evolution in technology to provide relationships among inter-enterprises and enterprises-costumers. Hence, we try to exploit this to integrate a web service in WfMS. However, those enterprises rely on the wide variety of languages and technologies which are heterogeneous in permanent evolution. Consequently our proposed approach is based on principles of MDA (Model Driven Architecture).

Keys words: *Business process, Models, mapping, B2B.*

1. INTRODUCTION

In recent years, Workflow Management Systems (WfMSs) have proved to be very useful for the management of business processes. Applications based on Workflow bring some fundamental benefits such as flexibility in changing the model of the underlying business process, integration capabilities for even disparate applications, reusability of activity and process models and scalability of application development and execution [5]. Workflow systems have been widely adopted by enterprises to streamline the business processes and make the business process execution more efficient through automating the process execution, speed up business processes and as a result deliver a faster service to the customer.

Nowadays, enterprises aim at realizing their dynamic e-business initiatives. Hence, workflow based enterprise information systems have to be extended to support inter-enterprise collaboration. Indeed business environments are dynamic and they require the workflow system to be flexible enough to be able to cope with rapidly changing situations. So, the flexibility and the ability to support inter-enterprise cooperation are becoming major challenges for workflow management. However, current workflow systems provide little support for these two aspects due to the lack of dynamic binding activity instances with optional applications at run time. But, how can WfMSs be used if a part of the process is carried out in another organization?

In this paper, we propose a solution based on a technology that facilitates the relationship among inter-enterprises and enterprises-consumers, called Web service. Indeed, Web Services are emerging as systematic and extensible frameworks for application-to-application interactions. Built upon existing web protocols and open XML, standard information can be seamlessly passed between remote applications running in different environments. These standards herald the next wave of Web services, providing real business delivery mechanisms and strategic opportunities.

Currently, the integration of the web services in the Workflows system is achieved without methodological setting and using ad hoc ways. It is based on several standards that are immature. Indeed, these works provide generally a canonical model, which is insufficient such as XPDL (XML - Process Language Definition) [10], WSFL (Web Services Flow Language) of IBM, XLANG of Microsoft [1], BPML (Business Process Modeling Language) of BPMI. However, there is a need of independence from languages and technologies. Therefore our proposed approach is based on the MDA (Model Driven Architecture) principles [9]. It is a new way of writing specifications, based on a platform-independent model.

This paper is organized as follows. In the next section, we describe some characteristics of WfMS. Section 3 presents technologies of web services and the principles of MDA. In section 4 we present an overview of general approach based MDA. Section 5 describes the mapping between the different models, the infrastructure that supports our approach and shows the feasibility of our approach by using an example. Finally, section 6 exposes the conclusion and discusses future work.

2. WORKFLOW SYSTEM FLEXIBILITY

Workflow is concerned with the automation of procedures where documents, information or tasks are passed between participants according to a defined set of rules to achieve, or contribute to, an overall business goal.

We define Workflow as «the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules» [10]. The process definition consists of a network of activities and their relationships. It indicates the start and termination of the process, and information about the individual activities, such as participants, associated IT applications and data. A system that completely defines, manages and executes these workflows (one or more workflow engine) through the use of software is defined with Workflow Management System (WfMS).

WfMS may be characterized as: the Build-Time functions, Run-Time control functions and Run-Time Interactions.

- Build-Time functions include business defining, possibly modeling, the workflow process and its activities.
- Run-Time control functions have the responsibility of managing the workflow processes and sequencing the various activities deriving from each process.
- Run-Time interactions with a human user tools for processing the various activity instructions (steps).

Flexibility has been addressed in the workflow community in a number of research efforts. A typical example of a software development effort is the WASA project [5], in which a workflow management system was designed that supports flexibility through mechanisms supporting on-the fly changes to process specifications of workflow instances. A different way of achieving flexibility in workflow process specifications is by distinguishing core process specifications and exception specifications. Cross-organizational workflow management has been the topic of a number of research projects on the integration of process support and electronic business. Well-known examples in this field are WISE and Cross Flow [5]. In the WISE approach, a global process specification is used to link activities or sub processes of organizations participating in a virtual enterprise. In the Cross Flow approach [5], dynamic virtual enterprises are supported in a service outsourcing environment. Cross-organizational workflow processes are specified in electronic contracts between consumer and provider in a service outsourcing relationship [5].

Our work combines aspects of Web services and of workflow management. Hence, we present, in the next section, the domain of Web services in the context of technologies and standards developed.

3. WEB SERVICES & MDA APPROACH

Web services are one of the promising technologies to dominate the next generation Internet applications. Web services have been often characterized as “self-contained, modular applications that can be described, published, located, and invoked over a network, generally the Web” [8].

3.1. WEB SERVICES CHARACTERISTICS

Web services provide a service oriented and component based approach to implement application-to application interaction. Web services are relatively simple and easy to implement and deploy since most enterprises have already setup Web servers. Web Service Description Language (WSDL) based on XML is used to describe Web service interface. SOAP (Simple Object Access Protocol) based on XML as well provides a lightweight message transport protocol to enable the involved two parties to exchange message in the Web service transaction. UDDI (Universal Description, Discovery and Integration), a registering standard, is used in the Web service architecture to support service publishing and discovery.

Web Services allow software developers to easily integrate different kinds of applications and services with each other, without having to worry about the underlying protocols, interfaces, environmental conditions, etc [8]. There needs to be a way to relieve software developers of protocol that is capable of capturing the workflow of a business process. The protocol needs to create an XML representation of the underlying business process, so middleware applications can understand this protocol. XLANG [3], proposed by Microsoft, is the XML business process language used in BizTalk Server. This language provides a way to orchestrate applications and XML Web services into larger-scale, federated applications by enabling developers to aggregate even the largest applications as components in a long-lived business process. An XLANG service description extends a WSDL service description with an extension element describing the behavioral aspects of the service. IBM also creates WSFL (Workflow Service Flow Language) [3] beyond WSDL to support service composition. WSFL uses ‘Flow Model’ to describe business processes and implementations and uses ‘Global Model’ to describe the composition of different Web services. It also extends WSDL to support describing internal application [3].

Constantly, growing number of those standards increases the difficulty of interoperability problems. Fortunately, the fundamental principles that are used by different system modelling frameworks are not so numerous. The point is that these technologies are bound to change.

In fact, to resolve this problem, we raise the level of abstraction by using MDA (Model Driven Architecture) approach. Thus, we will present principles and standards of MDA.

3.2 PRINCIPLES OF MDA

MDA proposed by the Object Management Group (OMG) is a recently emerging vision on system modelling that targets integration of different successful industrial solutions for the system architecture.

MDA unifies and simplifies modelling, design, implementation, and integration of applications. In MDA terminology there is the application’s Platform-Independent Model (PIM). Working from the PIM, MDA tools follow rules mapping to generate an intermediate model PIM tailored to the target middleware implementation platform termed a Platform Specific Model (PSM). This intermediate product will reflect non-business, computing-related details (typically affecting performance and resource usage) added to the PIM.

The exploitation of those principles to integrate Web service to the WfMS is described in the next section.

4. OVERVIEW OF THE PROPOSED APPROACH

In our approach, Web services, as we have seen above, are considered as partner that a process invokes at the time of his deployment (see figure 1).

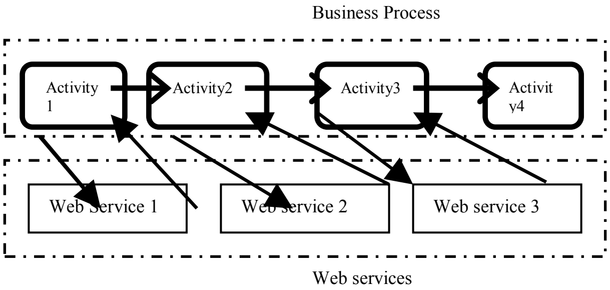

Fig.1: The process requiring Web services

Thus, a workflow service that adheres to the WfMC (Workflow Management Coalition) standard will consist of two parts: First, the user will define the workflow processes, and the activities in each process. Furthermore the conditions should be met for the activities to be started.

When the workflow definition is available, an Enactment Service will make sure that the activities are executed in the right order under right conditions [2]. Therefore the development of a WfMS requires two distinct steps:

1. The design and the definition of the Business Process.
2. An Enactment Service (workflow engine), for the deployment and the execution of the business process.

The design and the definition of the Business Process can be expressed in several languages (XPDL, BPEL, BPEL4WS…), and enactment Service in different programming languages (Java, c#...).

MDA unifies and simplifies the modeling, the design, the implementation, and integration of applications. It defines PIMs and PSMs that streamline platform integration issues and protect investments against the uncertainty of changing platform technology. We use this model according to the following architecture:

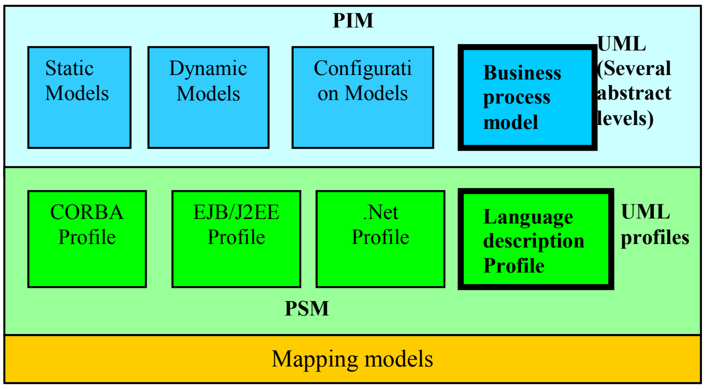

Fig.2: Adopted Architecture for MDA

The developing of WfMS, according to WfMC conditions, using MDA involves the use of MDA's models (PIM and PSM models). Consequently, our approach, that exploits these principles and these models, is based on two levels:

- ○ The process definition and design step.
- ○ The deployment and the execution of the process by a workflow engine.

At each level, we have to use PIM and PSM models in different phases of the development (see fig3). These two models represent details of the manner that a process will integrate and invoke Web services in different level of abstraction.

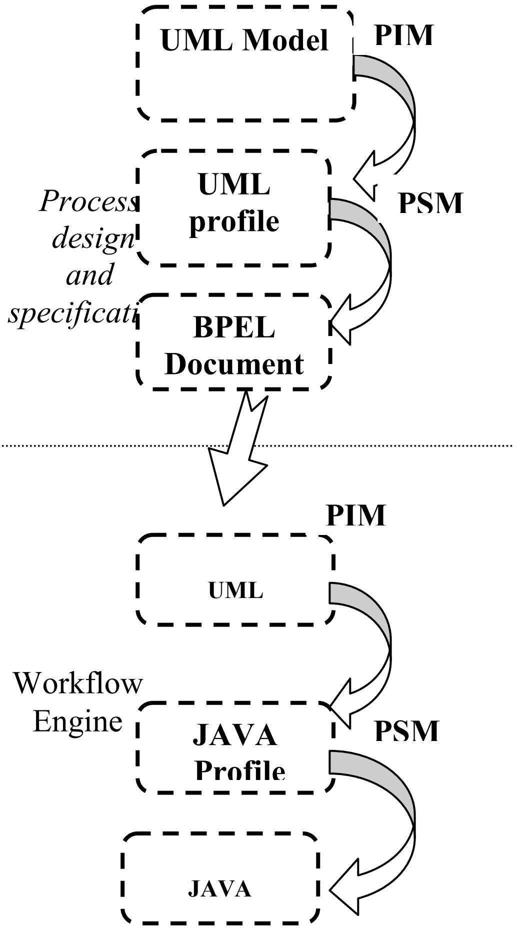

Fig 3: Organization of *PIM* and *PSM* models to design WfMS

Let's see with more details the two phases of our approach in the next section.

5. DIFFERENT PHASES OF THE PROCESS

In order to create different models PIM and PSM of figure 3 and modeling the two levels we have to use UML diagrams. Those are exploiting as follow:

5.1. DESIGNING OF BUSINESS PROCESSES

During this phase, a business process is translated from the real world into a formal or semi-formal definition. The resulting definition is sometimes called a process model, a process template, process meta data, or a process definition.

The PIM model: The PIM in Figure 4 describes at a high-level a business process. Here, we have to represent the main steps, and their arrangement. We specify neither the actions that are executed by the activities nor the applications that are invoked. This model is realized using UML. In this case it will be the activity diagram that describes the steps of the process, as we can note through a business man travel example in figure 4.

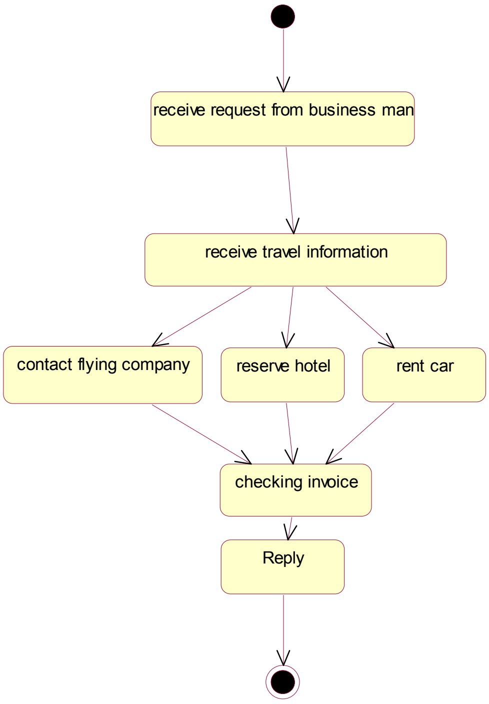

Fig.4 : PIM model concerning business man travel

In a following step we can incorporate rules that manage the business Process, always in PIM Model and with UML's activity diagram. After this, we will add more technical details concerning business process in the next step. We bind the activities already defined with applications and Web services.

Hence at the end of this step, we obtain a complete description of the business process (activities, their arrangement, Web services and partners that will participate to the business process), as PIM model, which is independent from any technology. The business process is designed and structured without concerning for the process definition language.

- The PSM model: The PIM model is mapped into PSM. The PSMs are so-called constrained formal models since they express more details [11]. The PSM Model is the intermediate representation of the business Process; it will be between the abstraction representation of the process (PIM) and the implementation level. PSM is realized by The UML Profile. A UML profile is a UML dialect defined via UML's built-in extension mechanisms. It involves selecting a subset of UML and then using the UML extension facilities to extend that subset.

The fundamental use of UML profiles is to model a particular domain, such as business information and business services. The other purpose of UML profiles is to parameterize mappings to technologies, as we do in this situation; we use a profile for parameterizing the mappings to specific implementation technologies.

The ability to extend UML is essential to MDA. This extension is possible with the notions of stereotypes and profiles. A UML Profile is used to define a specific set of extensions to the base UML in order to represent a particular domain of interest. In the UML profile, a process is represented as a class with the stereotype <<Process>> (see fig.5). The attributes of the class correspond to the state of the process. This process is defined in terms of its interactions with partners. They may provide services to the process, require services from the process, or participate in a two-way interaction with the process [6].

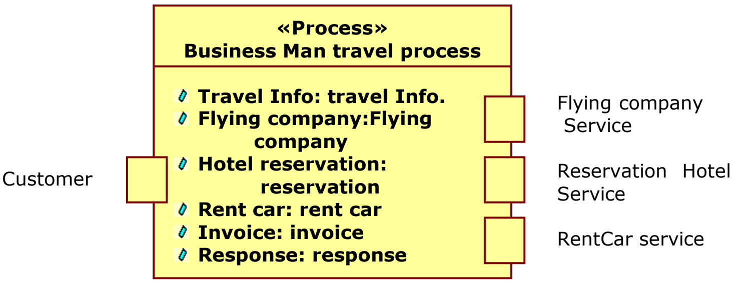
«Process»
Business Man travel process
Travel Info: travel Info.
Flying company:Flying company
Hotel reservation: reservation
Rent car: rent car
Invoice: invoice
Response: response
Customer
Flying company Service
Reservation Hotel Service
RentCar service

Fig.5: A UML class modeling a BPEL process

The UML profile uses stereotypes that concerns details implementation; for example «Invoke » and «Assign» are stereotypes in UML and activities in BPEL4WS [6].

The behavior of the class is described using an activity graph. The partners with which the process communicates are represented by the UML partitions. Activities that involve a message send or receive operation to a partner appear in the corresponding partition. Figure 6 represents the UML diagram BPEL profile that concerns the same Business man travel process.

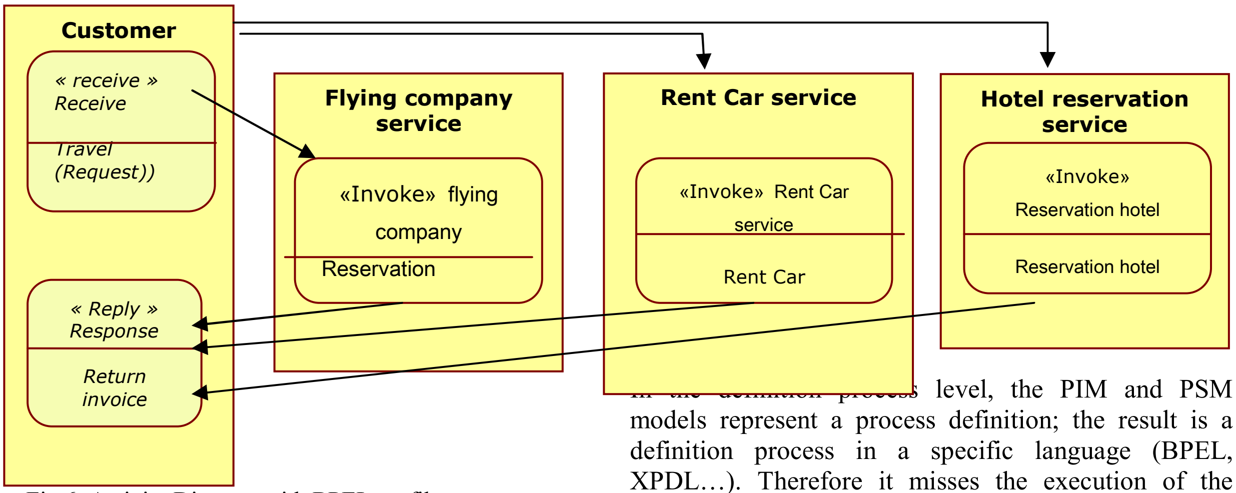

Fig.6: Activity Diagram with BPEL profile

Note that no detail of language was incorporate here. This UML profile represents PSM because it gives a model of the process in the way that facilitates the mapping from UML to a particular language definition (BPEL). However PIM should represent the business process in an independent way.

Using the UML profile permits describing a mapping to BPEL4WS, this can be automated to generate web services artifacts.

- Mapping to BPEL4WS: Table 1 shows an overview of the mapping from the profile to BPEL4WS (version 1.0) covering the subset of the profile introduced [4].

« Process »	BPEL process definition
Activity graph on a « process »	BPEL activity hierarchy
« Process » class attributes	BPEL Containers
Hierarchical structure and control flow	BPEL sequence and flow activities
« receive », « Reply » « Invoke» Activities	BPEL receive, reply, involve activities

A cut down version of the BPEL document that would be generated from the business man travel example in this paper is shown in Figure 7 (much of the detail is omitted here due to space constraints).

```
<process
      abstractProcess="no"
      containerAccessSerializable="no"
      enableInstanceCompensation="no"
      name="Business Man Travel"
      suppressJoinFailure="yes"
      targetNamespace="http://www.bpel-
      examples.ibm.com/
      Businessman travel/Business man
      travelProcess/BMT.bpel">
<containers>
   <containermessageType="TraveliInfo"
                                name="Travel info"/>
   <container messageType="flying company:  "
name="flying company"/>
   <container messageType="reseravation hotel:
   reservation hotel" name="Reservqtion hotel "/>
</containers> </process>
```

Fig7.: Cut down BPEL program.

In the definition process level, the PIM and PSM models represent a process definition; the result is a definition process in a specific language (BPEL, XPDL…). Therefore it misses the execution of the process in order to invoke the Web service and controlling the execution under the right conditions, it will be the role of the second part according to WfMC, the workflow enactment service or the workflow engine.

Interaction with the process control software (workflow engine) is necessary to transfer control between activities, to ascertain the operational status of processes, to invoke application tools and pass the appropriate data, etc.

In the definition process level, the PIM and PSM models represent a process definition; the result is a definition process in a specific language (BPEL, XPDL…). Therefore it misses the execution of the process in order to invoke the Web service and controlling the execution under the right conditions, it will be the role of the second part according to WfMC, the workflow enactment service or the workflow engine.

Interaction with the process control software (workflow engine) is necessary to transfer control between activities, to ascertain the operational status of processes, to invoke application tools and pass the appropriate data, etc.

MODELING OF WORKFLOW ENGINE

At run-time the process definition is interpreted by software which is responsible for creating and controlling operational instances of the process, scheduling the various activities steps within the process and invoking the appropriate Web services and application resources. This software is called workflow enactment service, or workflow engine, which has as responsibility to interpret the process definition in a specific language and to orchestrate the starting and stopping of workflow activities accordingly. It should be able to read the definition process.

Exploiting the MDA's principles in this context, i.e. developing a Workflow engine, suggest us to use the two MDA Models too, which interfere also in two level of abstraction. (see Fig 3).

- The PIM model: here, the PIM model represents the workflow engine components in the run-time as UML class Diagram, where a workflow process instance class represents an instance of a particular workflow process definition. A workflow process instance is associated with a set of workflow activity instances; these ones are contained in the process instance, which may realize a work item. A workflow process instance is associated with one workflow process definition – its implements that process definition. . The elements composing a workflow engine according to WfMC are designed with UML class diagram (see figure 8).

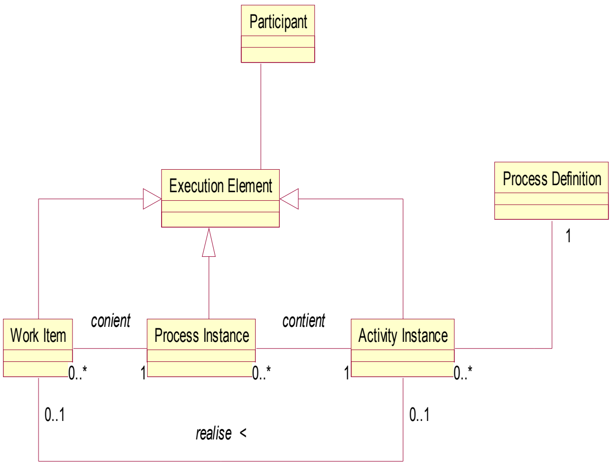

Fig8: The PIM model of the workflow engine.

- The PSM model: The PSM is a UML profile model of a programming language that is Java or C# for example.

Both PIM and PSM descriptions of applications are independent from technology constraints, PSM introduce technical aspect, but by saving this independence.

The code: Finally, a Business process definition is deployed in a workflow engine developed in a programming language. This program is obtained according to mapping rules that are adopted to obtain C# program. The mapping rules transform an UML class to C# class, Operations are considered as Method

6. CONCLUSION

In this paper, we have presented an approach that adopts a Model Driven development to integrate Web services to the WfMS. This approach is based on the principles of MDA. It permits to automate business processes in a high level of abstraction. This approach enables Web services, participating to a Business process, to be incorporated into an overall workflow system.

We have proceeded according to WfMC principles. In fact, the approach is composed of two phases modeling Process and the execution phase. Thus the MDA models (PIM, PSM) interfere in the tow level.

Developing WfMS that integrate a web service during the execution of a business process using MDA presents advantages in terms of flexibility, raising abstraction, independency of technologies.

However the mapping between different models of MDA is a way that requires effort and researches [4]. Indeed we must have models and rules of transformations in order to realize mapping between UML and several modeling process language. Other UML profiles should be developed, in order to make the mapping from UML and other languages possible. In the near future, different tools will be exist to automate transformations, and achieve the absolute independence from technologic constraints.

REFERENCES

[1] Jean-Marie Chauvet, : « Web Services avec SOAP,WSDL, UDDI, ebXML…», Editions Eyrolles 2002.

[2] Emde Boas G.: From the Workfloor: Developing Workflow for the Generative Model transformer.2002. Available from www.softmetaware.com

[3] Gardner Tracy: UML Modeling of Automated Business Processes with a Mapping to BPEL4WS. IBM 2003. pp. 1-4

[4] Gardner Tracy: Mapping from UML to the Business Process Execution Language for Process Execution Language for Web Services (BPEL4WS) Web Services (BPEL4WS). OMG MDA Implementers' Workshop.2003. pp. 1-18

[5] Grefen.P, Hu.J, "Conceptual Framework and architecture for service mediating workflow management",2002.

[6] Mantell K: from uml to bpel. ibm. 2003. available from www-106.ibm.com/ developerworks/ webservices/ library/ws-ml2bpel. pp. 1-10

[7] Matou sek P: Verification of Workflow specification standards. ICEIS 2003. pp. 8

[8] Modeling and Composing Web Services with MDA and Workflow. 2004. available from www.eseo.fr.

[9] Sigel, J: Using OMG's Model Driven Architecture (MDA) to Integrate Web Services. Object Management Group, 2002.

[10] WfMC. The Workflow Reference Model. Document Number TC00-1003. 95 pp 6-18.

[11] Wang N; Shmidt D; Gokhale A, "Using Model-Integrated Computing to Compose Web Services for Distributed Real-time and Embedded Applications", 2003.