

*Miladin Stefanovic,
Slavko Arsovski,
Snezana Nestic,
Aleksandar Aleksic*

*Faculty of Mechanical
Engineering, University of
Kragujevac, Serbia*

Integration of Virtual Enterprises Using Service Oriented Architecture

Abstract: *Virtual enterprise integration (VEI) is virtually the most critical success factor for making virtual enterprise (VE) a real, competitive, and widely implemented organizational and management concept. One of possible approaches in virtual enterprise integration is employment of web services. In this paper we will present process-oriented approach for developing an SOA for VEI. The general assumption of suggested approach is that service oriented architecture is based on business service and that business services mostly correspond to exchanged documentation in a real business system. CASE tool for web service specification is also presented.*

Keywords: *Virtual enterprise integration, web services*

1. INTRODUCTION

Concept of virtual enterprises emerged in 1990s. There are number of definition of virtual enterprises. But all concepts summarizes some key points: cooperation and complementarities, networked or distributed organization, temporary organization and infrastructure supporting interoperation. VE is considered as a temporary organization in which various distributed business partners form a cooperative network [1]. Virtual enterprise integration (VEI) is virtually the most critical success factor for making virtual enterprise (VE) a real, competitive, and widely implemented organizational and management concept. However, according to many authors, the present solutions for VEI are either insufficient or inexistent [2]. The main mechanism, or tool, for the processes interaction improvement is the computer and information technology in a variety of specific technologies and aspects. On the other hand there is difference between VEI and EI. Understanding and considering integration as a design process is of the greatest importance for building a new generation of integration models for VE.

Process integration spans all dimensions of

virtuality. In many cases the integration goes to back-end or legacy applications. In other cases it also involves the integration of information or databases from multiple sources. In most cases it is a combination of the two. In fact, it is often the case that applications have their own repositories or databases. For instance there could be many applications or repositories that manage information about customers. In many organizations these are not integrated and there is no global and consistent view of the same customer across departments or organizational units [3].

The corresponding concept in virtual enterprises is the process. An organization is defined by its sets of enacted policies and procedures; so are virtual organizations [4, 5]. Processes are used to implement these policies and procedures. Processes define the control logic of the interactions and exchanges for particular goals. The process models include the various roles and relationships between participants. Processes also define the messages or documents that get processed at each step. Thus connectivity and integration needs not only to address process integration that uses the common interface technologies, but also to provide a direct representation and enactment of the exchange, control, and information flows

between applications, human participants, or trading partners.

The scope of virtual enterprises can be summarized as follows:

1. *Process Integration of applications within the same organization:*
2. *Process Integration of organizations, applications, and content between different organizations in the same enterprise:*
3. *Process Integration of Trading Partners.*

Processes involve choreography of activities. These activities can be carried out by specific individuals; by specific organizational roles; by specific e-business roles; or specific applications. One of the main advantages of business process management is the separation of the operational or process logic from the application. In almost all applications there are processes and workflows. The application programming interfaces and the application packages themselves often embed or capture these processes.

In this paper we will deal process oriented development of web services and service oriented architecture as mean for integration of virtual enterprises.

Service-oriented architectures (SOA) will form the basis of future information systems. Basic web services are being assembled to composite web services in order to directly support business processes. As some basic web services can be used in several composite web services, different business processes are influenced if for example a web service is unavailable or if its signature changes.

The general assumption in this paper is that service oriented architecture is based on business services, and that business services are mostly adequate to exchanged documentation in real business system. The documentation of quality system (DQS) is identified as appropriate approach for description of business processes in business system and identification of exchanged documentation with business environment. In this paper we will present The approach for planning and specification of the Web Service for VEI, based on documentation of quality system is focus of this article. CASE tool for development and specification of the Web Service is also presented.

2. PROCESS-ORIENTED METHODOLOGY FOR DEVELOPING AN SOA

However, many enterprises experience difficulties in the formation and operation of virtual enterprises, especially concerning integration issues. There are many different approaches in definition of an architectural framework, , which aims to support the set-up and operation of virtual enterprises [6, 7, 8, 9]. Many of them use web services and SOA [10, 11, 12].

Based on open XML standards, Web Services are service- oriented architecture that allows for creating an abstract definition of a service, providing a concrete implementation of a service, publishing and finding a service, service instance selection, and interoperable service use.

Process-Oriented Methodology for Developing an SOA could be stated as:

- Step 1: Identification and Modeling of Executable Business Processes
- Step 2: Identification and Modeling of the Atomic Services
- Step 3: Specification of Composite Services
- Step 4: Specification of Deployable Services
- Step 5: Mapping to BPEL

As the figure 1 shows, the high-level work-flow model references objects in the low-level structural model. These references are used for establishing transaction boundaries without adding additional complexity to the structural model diagram.

Process interconnection mechanisms are necessary to coordinate geographically distributed business processes in order to strength awareness inside virtual enterprises, to facilitate multinational e-transactions, etc. Actually, existing business process modelling and enactment systems (workflow systems, project management tools, shared agendas, to do lists, etc.) have been mainly developed to suit enterprise internal needs. Thus, most of these systems are not adapted to inter-enterprise cooperation. As we are interested in workflow processes, we aim, through this paper, to present a model supporting dynamic heterogeneous workflow process interconnection.

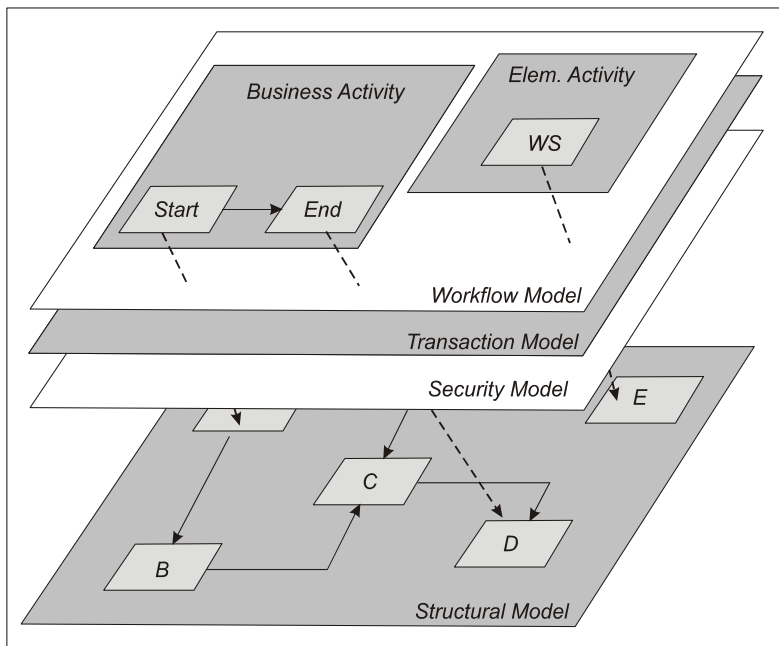
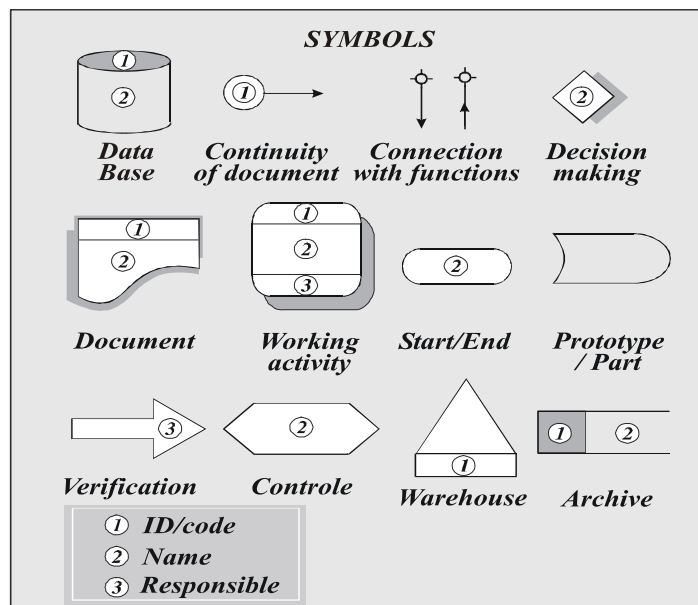


Figure 1 From Business service to structural model



- ① - Code, identification of activity (it also could be name of data base, document name, or archive name)
- ② - Document name (it also could be name of data base, decision making, name of document, Start/End of process, name of process control or name of archive)
- ③ - Identification of responsible person (it also could be identification of responsible person for verification of specific document or activity)

Figure 2 Notation, symbols of AS2

We consider the interconnection of enterprise workflow processes as the management of a “workflow of workflows” in which several heterogeneous workflow systems coexist.

The general assumption is that service oriented architecture is based on business services, and that business services are mostly adequate to exchanged documentation in real business system. The documentation of quality system (DQS) is identified as appropriate approach for description of business processes in business system and identification of exchanged documentation with business environment.

Based on AS2 approach [15], the documentation of a quality system could divide in the following categories: procedure, instruction, or direction (depending on complexity of activities and organizational level on which activities are defined or performed). These entities can be called procedures. From one procedure, we have one or more templates as an output. The procedures and templates are integral part of one scenario, for example: product development or marketing and sale. More functions and organization units from one company are employed in realization of one scenario. The next step is definition of phases and functions, and definition of elements of each scenario and definition of information about elements of scenario. During the distribution of a function in specific phases, we have to define activities, documents, additional elements and connection between elements as well as input of data. This method integrates BSP (Business Process Planning), HIPO (Hierarchy Input, Process Output) and SSA (Structured System Analysis), using documents of quality systems. Symbols, notation of AS2 approach are used in this CASE tool (Figure 2).

3 DEVELOPMENT AND SPECIFICATION OF WEB SERVICES

Companies should start using the Web Services in internal application integration projects or legacy system integration. It is much easier to control, manage, find, execute, and maintain the Web Services within an intranet as compared to using them over the Internet. The

main idea is providing an average company with methodological approach to identify possible business services and the Web Service developing integrated infrastructure specially in virtual enterprises [13, 16].

This method suggests development of the Web Services in following steps [14]:

- Development of a matrix of documentation of quality system. The documentation of quality system could be divided in the following groups: procedures, directions, instructions, templates, and records. The matrix of documentation of the quality system connects procedures and records (e.g. processes and information that present input or output from those procedures). In the first step, the procedures are generalized as the Web Services, and input-output records as messages that the Web Service exchange with environment or as interfaces.
- Development of a communication matrix. Defines entities that take part in communication e.g. entities between documentation is exchanged. In this and previous step, the best beginning is using activity diagrams. From the previous matrix, we select a procedure and define the flow of templates and records to and from that procedure according to aspect of participants in communication. From the communication matrix, we extract the sequential diagrams for the Web Services, with defined scenarios for Web Services (*Request/Response* for example), operations and data types.
- Development of a data matrix, for a selected template/record that is exchanged between two entities in communication, and based on previous matrix, exchanged data, and entities involved in creation and modification of that data are defined.
- The last step is implementation and development of summary description of the Web Service.

4 CASE TOOL Q-WS FOR THE WEB SERVICE DEVELOPMENT

There is now widespread acceptance of the Web Services and service-oriented architectures. But despite the agreement on the

key Web Services standards. there remain many challenges. Programming environments based on WSDL support go some way to facilitating the Web service development.

The general idea for this CASE tool is to use documentation of quality system and activity diagrams as a basis for specification of Web Services and development of WSDL. The tool is a base on the previously presented model for specification of the Web Services based on DQS.

This CASE tool is developed with three layer architecture. These selections ensure: openness to other data base management systems; portability, maintainability, lower costs, friendly user interfaces, software and hardware compatibility and easy integration with different products. The connection to a database is flexible ant it is possible to use different database management systems using just different connection strings in a data connection block. So this CASE tool could be

used with different DBMS. The security and authorization of use of this software was achieved by a set of access rights to DBMS and authorization of access. There are two sets of rights: one full, which enables creates, reads, updates and deletes data, and other that enables just reading.

Case tool enables development of web services through following steps\;

Definition of scenarios - Definition of data flows using AS2 notation [15]. Using development environment depicted in Figure 5, and symbols presented on left side user could form diagrams. Using drag and drop technique, user arranges symbols on the working area and make connections between them (Figure 4). The next step is definition of all parameters for each specific procedure or record. First user defines name, code and other general attributes of procedure (connection with other procedures and records).

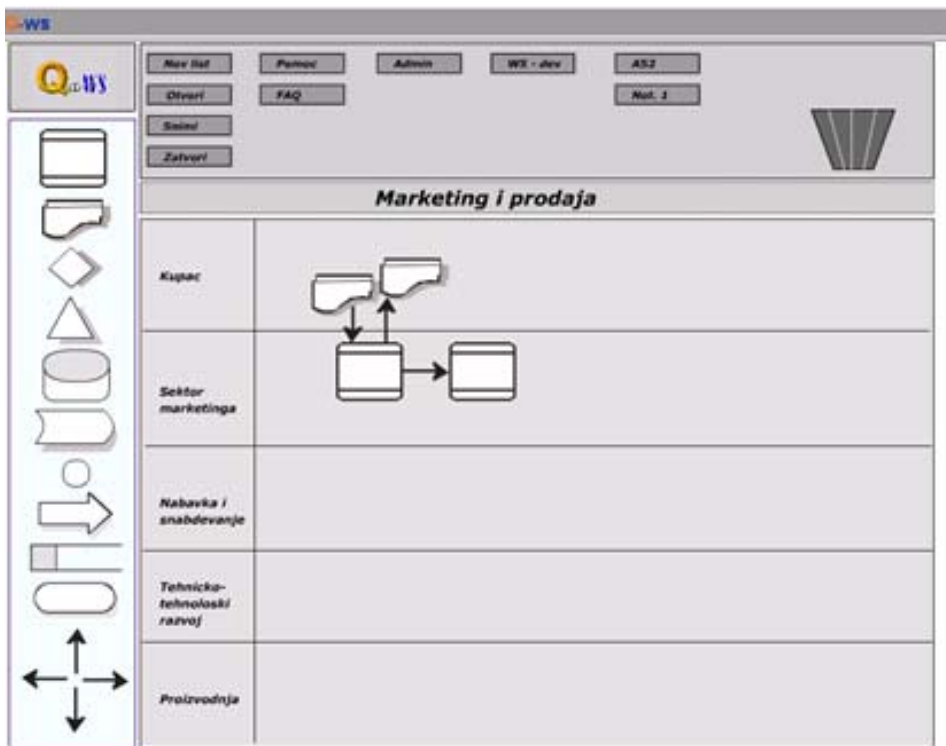


Figure 4 Symbols and data flows (procedures and flows) for Marketing and sale scenario

- **Web Service development** - Analyzing exchange of documentation between two

entities, in presented case customer and marketing function, the software suggests the Web Services with messages that

generally present records or information exchanged between customer and marketing. Users of this software tool could manually correct model or connect or divide specific Web Services and/or operation (and exchanged documentation between two procedures is used as a basis for definition of interfaces between services or legacy system). The beginning is definition of couples or messages exchanged between entities. The records become request/response messages and connection between neighboring procedures (with no interchange of structured documents) produce Web Service.

- **Generation of WSDL code** - In this step, based on input and existing parameters, CASE tool develops part of WSDL code. These properties of Q-WS automate development of Web Service and domain of WSDL code development, and reduce or eliminate causes of errors in code.

5. CONCLUSION

One of the most important issues concerning virtual enterprises is integration.

There are many different approaches in integration of virtual enterprise. It is recognized that SOA and web services could play very important role in integration of virtual enterprises. In this paper we advocate process oriented development of web services.

We suggest that high quality integration of virtual enterprises could be achieved using model of specification and development of the Web Service based on the documentation of the quality systems. Three steps, starting with the matrix of the quality system documentation following with the matrix of communication and the matrix of data enable, specification and development of the Web Services. The general assumption is that service oriented architecture is based on the business service, and business services mostly correspond to exchanged documentation in the real business system. The documentation of the quality system is recognized as documents that describe business processes in the real business system and identify exchanged documentation with the environment.

Presented CASE tool has potential to serve in specification and development of SOA for integration of virtual enterprises.

REFERENCES:

- [1] Kim, T. Y. Lee, S., Kim, K., Kim, C. H. Modeling framework for agile and interoperable virtual enterprises. *Computers in Industry*, April 2006, vol. 57, no. 3, p. 204–217.
- [2] Putnik, G., Cunha, M. M. *Virtual enterprise integration: technological and organizational perspectives*. Published by Idea Group Inc (IGI), 2005
- [3] Yang, J., Li, G., Ai, D. Integration and Innovation Orient to E-Society. *IFIP International Federation for Information Processing*, 2007, volume 51, no. 1, Wang, W. (Eds), (Boston: Springer), p. 282-239.
- [4] Dorn, J., Hrastnik, P., Rainer, A. Web Service Discovery and Composition for Virtual Enterprises. *International Journal of Web Services Research*, 2007, vol. 4 no. 1, p. 23 - 39.
- [5] Schmit, A. B., Dustdar, S. Model-driven Development of Web Service Transactions. *Technologies for E-Services, 6th International Workshop, TES 2005*, Trondheim, Norway, September 2-3, 2005
- [6] Protogeros, N. (Ed.). *Agents and Web Service Technologies in Virtual Enterprises*. IGI Global. 2008, ISBN: 978-59904-648-8.
- [7] Jamroendararasame, K., Susuki, T., and Tokuda, T. A Diagram Approach to Automatic Generation of JSP/Servlet Web Applications. *Proc. of the 6th IASTED International Conference on Software Engineering and Applications*, Boston, USA, 2002. 292-297.

- [8] Jhingran, A. D., Mattos, N., and Pirahesh, H. Information integration: A research agenda. *IBM Systems Journal*, 2002. vol. 41, no. 4., p. 555-562
- [9] Baïnaa, K., , Benalib, K., and Godartb, C., DISCOBOLE: A service architecture for interconnecting workflow processes. *Computers in Industry*, December 2006, vol. 57, no. 8-9, p 768-777
- [10] Camarinha-Matos, L. M. and H. Afsarmanesh, Infrastructures for Virtual Enterprises: a summary of achievements, *Proceedings of PRO-VE'99 - IFIP Int. Conf. On Infrastructures for Virtual Enterprises*, Kluwer Academic Publishers, ISN 0-7923-8639-6, Porto, Portugal, 483-490, 1999
- [11] Fuquan, S. Wang, L., Tingbin, C., Yunlong Q. Dynamic Information Integration of Virtual Enterprises Based on Web Services and J2EE. *International Conference on Wireless Communications, Networking and Mobile Computing*, 2007. New York, USA, Sept. 2007 p. 6146 – 6149
- [12] Petersen S. A, Rao, J., Matskin, M., *Virtual Enterprise Formation supported by Agents and Web Services*. IGI Global;
- [13] Grundy, J., Hosking, J. A Visual Language and Environment for Composing Web Services. *Proceedings of the 2005 ACM/IEEE International Conference on Automated Software Engineering*, Long Beach, CA, Nov 2005.
- [14] Stefanovic, M., Matijevic, M., Eric, M., Simic, V. Method of design and specification of web services based on quality system documentation. *Information Systems Frontiers*, March 2009, vol. 11, no. 1., 75-86.
- [15] Arsovski, S., Arsovski, Z., Stefanovic, M. An Approach of Information System Development in QMS Environment'. *Communication in Dependability and Quality Management, An International Journal*, 2001, vol. 4, no. 2, p. 144-152
- [16] Kassoff, M., Kato, D., Mohsin, W, Creating GUIs for Web Services. *IEEE Internet Computing*, September/October 2003, vol. 7, no. 5, 66-73