Prototype of a Platform for Business Collaboration

Marian Mach, Jan Hreno
Technical University of Kosice
Letná 9, 042 00 Košice, Slovakia
{Marian.Mach, Jan.Hreno}@tuke.sk

Karol Furdik
InterSoft a.s.
Florianska 19, 040 01 Košice, Slovakia
Karol.Furdik@intersoft.sk

Abstract. *The paper presents a platform for maintenance of short-time, project-oriented business alliances. The focus is given on semantic interoperability of services provided and consumed by alliance partners during the process of business collaboration. The structure and content of supporting ontologies is presented as a knowledge representation basis for semantic description of abstract business process models. The implemented platform components, such as messaging and communication infrastructure, semantic process and service management, identity management, security infrastructure and web portal interface are described both on a conceptual and technological level. The approach is demonstrated by a set of pilot applications.*

Keywords. Business alliances, business process modelling, ontology, semantic interoperability, business collaboration platform

1 Introduction

An ability to collaborate flexibly, efficiently and effectively on the globalised market is a crucial success factor and even an imperative for most of enterprises in the emerging digital and knowledge-oriented economy environment [9]. The flexibility in the business collaboration is determined by the ability to create and manage an inter-organisational alliance of enterprises and organisations co-operating on a project base. Such a collaboration, which is usually of temporary nature and is characterised by quickly varying business environments, requires proper technological background enabling interoperable provision and consumption of services between all members of the business alliance. Moreover, business processes in this type of networked enterprises need to be defined, organised and maintained as an inter-company composition of particular processes and services provided by the alliance participants. A platform for interoperable service composition in business alliances, which is based on semantic structures such as ontologies and abstract business models, is presented in this paper.

1.1 Motivation and objectives

The semantic interoperability refers to an approach enabling for services, which are described (i.e. annotated) by concepts of a standardised and shared knowledge base, to be provided, accessed, retrieved, orchestrated and used in a flexible manner. Inputs, outputs, and characteristics of possibly heterogeneous services can be semantically matched and integrated, which allows composing the services into a customisable workflow.

An advanced technology supporting the semantic interoperability in IT solutions built on SOA principles is the *Enterprise Service Bus* (ESB) [2]. This messaging and communication infrastructure serves as a middleware that can facilitate interoperability of heterogeneous services and applications in an enterprise. By defining standardised service interfaces and message routing, the ESB mediates incompatibilities of communicating applications, orchestrates their interactions, and makes the integrated services available for broad access and re-use [10]. Technically, the ESB enables loose coupling of interacting systems and allows to distribute the business logic of a solution into incremental, digestible modules, maintaining their own local control and autonomy.

Our project builds on these principles and encompasses a service-oriented infrastructure including a semantically enhanced ESB for workflow control, handling and transformation of messages. The resulting platform is expected to be tested on pilot applications in Finland (application case: Documentation Management) and Austria (two application cases: Business Alliances and Identity Federations). The *Business Alliances* case is focused on the creation and management of business alliances, including maintenance of service providers, location and configuration of services, integration into a workflow, as well as the tracking, contracting, and ordering of services. The *Identity Federations* case demonstrates the management of user identities in a networked enterprise, namely the maintenance of access rights, roles, and resources within a collaborative environment. The *Documentation Management* application case, focusing on intra-enterprise product development, provides a support for documentation services by means of secure knowledge and content management. In addition, the *SPIKE Furniture Store* application was defined as a representative demonstration of the whole developed platform. It combines the core functionality of the three "basic" cases into an example of the international furniture store. The focus is given on the creation of collaborative value chain that covers the development of a newly designed product.

1.2 Related research

A prerequisite for employing the ESB in an organisation is a specification of a workflow structure, which is capable to model particular processes, tasks, actions, flow of information artefacts and messages between the communicating applications. However, the ESB infrastructure itself and the notations for process modelling are both focusing on syntactic specification of interfaces, message exchange, and workflow structures. Integration based on the meaning, expressed in a machine-readable way, is the current challenge in the ESB-related research [7].

The vision of semantic business process modelling, formulated in [8] and further elaborated, e.g., in the FP6 project SUPER (`http://www.ip-super.org`), aims at achieving a higher degree of automation in discovery and mediation of cooperating services. The usage of semantic technologies and Semantic Web services in the process modelling, service configuration, execution, monitoring, and analysis is envisioned as a method that can overcome the heterogeneity and incompatibility problems towards the semantically interoperable services. It may also help to reduce the human intervention throughout the lifecycle of business process modelling. The *Semantic Service Bus*, as an enhancement of the general ESB, uses semantic description of service capabilities, properties, and exchanged information artefacts, which enables automated service discovery, routing, composition and data mediation [10].

Semantically enhanced business process modelling and design of semantic ESB is in focus of research initiatives such as, for example, the Object Management Group (`http://www.omg.org`), FP6 R&D projects STASIS (`http://www.stasis-project.net`), OPUCE (`http://www.opuce.tid.es`), the above-mentioned project SUPER [1], etc.

2 Semantic knowledge base

Semantic structures designed and implemented to support the service interoperability and workflow management include a structure of resource ontologies representing general e-Business concepts of networked enterprises as well as domain-specific information of particular pilot applications. In addition, abstract models of business processes can be considered as specific conceptual representations of application use cases as well.

2.1 Ontology structure and content

The process of ontology development was accomplished in line with the methodology based on the requirement-driven approach [11]. The requirements of stakeholders involved in a business alliance, i.e. service providers and requestors, were systematically collected, evaluated and formalised into the resulting ontology implementation. All our application cases (ACs) were textually described and analysed from the perspective of required information needs and information quality [6]. Based on this analysis, key terms were identified in structured textual descriptions and corresponding glossaries were constructed. The terms were grouped into a hierarchical structure, which was formalised by controlled vocabularies. Semantic relations between the terms (e.g. is_a, part_of, has, affects, etc.) were specified and represented in a form of ontology-like structures. To ensure wider compatibility and applicability of the resource ontologies, available external ontologies and standards were extensively reused. The implementation of ontologies was determined by the WSMO / WSMO-Lite conceptual framework [13], proposed as a background semantic platform for modelling both ontologies and business process models. The ontologies were implemented in WSML format (`http://www.wsmo.org/wsml/`), using the WSMO Studio toolkit [3].

The structure of the resource ontologies was designed with respect to the proposal of data elements identified in the conceptual perspective of the architecture design [12]. Particular ontology implementations were specified in three logical groups such as Process-related, System-related, and Domain ontologies.

Process-related ontologies, which provide conceptual models for semantic description of business processes and their elements such as *Process, Task, Service, etc.*, consist of:

- *Business process ontology*, a conceptual model of abstract business processes [3] and their mapping into the BPEL elements of the corresponding executable process. It is based on BPMO ontology [1] providing representations of graphical process elements, together with sBPMN and sBPEL ontologies containing WSML models of semantic extensions to BPMN and BPEL 2.0, respectively.

- *Service ontology*, a structure of concepts enabling semantic description of the services included into a collaboration process and referenced by the Task data elements. It consists of WSMO-Lite lightweight ontology for semantic description of Web services [13] and of an ontology that interconnects the service-related concepts with collaboration processes by means of human tasks and available types of online services.

- *Resource ontology*, a set of conceptual types that semantically describe physical information resources - artefacts referenced as inputs and outputs within the activities (i.e. services) of a business process. It includes X12 EDI ontology of standardised attributes for document interchange, the Dublin Core metadata element set (`http://dublincore.org`), the SIOC Core ontology (`http://rdfs.org/sioc/ns`) modelling online communities, and an ontology of generic resource types such as *Document, Report, Message, Template*, etc.

System-related ontologies that semantically describe the platform environment:

- *Core ontology*, a specification of general objects and elements that define a conceptual framework for collaboration environment. It combines the Business Roles and Business Functions ontologies of the project SUPER, non-functional properties of WSMO services, SKOS classification schemes, and general concepts such as *CollaborationObject, Alliance, Contract, Organisation, Address, Person, Actor*, etc.

- *System ontology*, a set of concepts for semantic description of the system data, configuration settings and environment properties of the whole platform, its installations and particular client-side tools.

- *User ontology*, a representation of all the data related to users. It is based on vCard RDF ontology of basic contact data for people and organisations, the WSMO non-functional properties related to the security and access rights issues, and the concepts describing the roles identified for external entities that are expected to communicate with the system.

Domain ontologies extend the general semantic structure of process-related and system-related ontologies towards applications. They provide a domain-specific conceptualisation for these application cases: AC1 Information Hotel, AC2 Legacy Applications, AC3 Identity Federations and AC4 SPIKE Furniture Store. These conceptual models were created from background materials and textual descriptions, which were analysed and processed according to the above-mentioned methodology of requirement-driven approach [11].

Altogether there were implemented ten process-related ontologies, nine system-related ontologies and four domain ontologies. These resource ontologies were published under a single root namespace of `http://www.spike-project.eu/ontologies/` and are freely available for further use.

2.2 Abstract process models

The design of abstract business process models was accomplished in parallel with the development of resource ontologies. User partners, as providers of the specified pilot applications, described the respective application cases in a textual format accompanied by flowchart diagrams of process flow. The provided descriptions were analysed and subsequently formalised into the standard BPMN notation. They were then implemented into the ontology-based visual BPMO representation of business process models by means of the BPMO Modeller tool of WSMO Studio.

The structure of implemented business process models is varying across the application cases. The AC1 Information Hotel has a straightforward value chain defined and this is the reason why this AC is represented by a single and complex process model that integrates all its use cases. The AC4 SPIKE Furniture Store is of similar matter; however, the value chain here is even more complex so that the resulting business process model structure consists of four separate models for each of the AC's use cases. The AC2 and AC3 are more fragmented, they cover particular aspects of the federation of identities and the inclusion of legacy applications into a collaborative process. That is why the respective process models were created separately for individual use cases in these ACs.

An example of BPMO model of the Service ordering use case is presented in Fig. 1. This use case, which is a core of the AC2 Legacy Applications, is modelled as an interaction between three actors - Service requestor, Service provider, and the platform. The Service requestor searches for a proper service (by the Service Search sub-process, which is modelled in a separate use case) and prepares an order for the retrieved service that matches with his/her business needs. Afterwards, the order is processed by the platform. The Service provider is notified; the contract of usage is generated and forwarded to the requestor. After accepting the contract the service is ready for usage. Optionally, the requestor may negotiate the contract details with the Service provider.

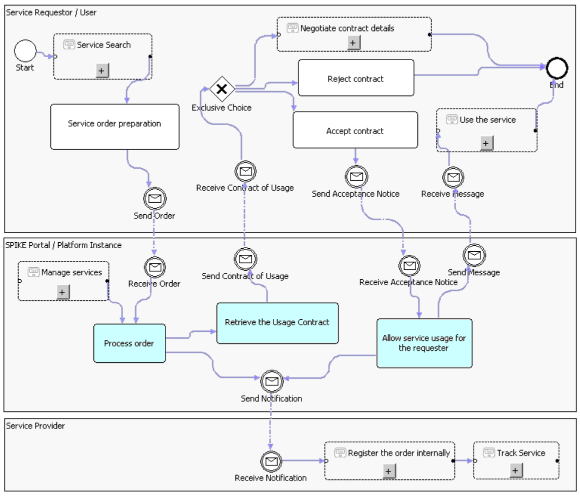

Figure 1: Abstract process model for Service ordering use case, AC2 Legacy Applications

The WSMO / BPMO framework, which is employed for creation of abstract business process models, determines the format and partially also the location of their implementations. The implemented models are stored in files of WSML format, accompanied with the files containing layout settings (i.e. positions of graphical elements displayed in the BPMO Modeller tool). Internally, the business process models can be seen as separate ontologies, which are composed of instances of the BPMO ontology concepts [1]. The namespace for these ontologies is implicitly, by the BPMO Modeller tool, set on the `http://www.ipsuper.org/ontologies/BPMO/bpmo-1-4-20080109#`. Physically, implementations of all abstract business process models of ACs are accessible under a single root location of `http://www.spike-project.eu/BPmodels/`.

3 The semantic service bus

Developed semantic structures of ontologies and abstract process models are initial steps towards the orchestrated workflow of interoperable services. To anchor the process model in real services and artefacts, its activity elements such as goal tasks, web service tasks, and manual tasks need to be further specified by grounding them into a concrete WSDL representation of services. The online services obtain the WSDL descriptions inherently, while for offline services (i.e. activities requiring a human interaction, referenced as manual tasks) the description of service properties can be modelled by the standardised XForms format (`http://www.w3.org/MarkUp/Forms/`). The inputs and outputs of the services that represent the tasks need to be specified by referencing them to semantic representations of particular artefacts exchanged in the workflow.

The Semantic Annotations for WSDL and XML Schema (SA-WSDL) specification [4] was adopted for the semantic description of the web services. SA-WSDL defines XML attributes, which link various WSDL elements to the ontology concepts. It enables specification of the service type or provides a semantic description of the service inputs and outputs. Additionally, SA-WSDL attributes can specify the transformations between the XML messages and the instances used for the semantic data mediation. An automated Semantic Mediator, implemented as an internal module of Semantic Manager [12], can "lift" the data in one XML format to instances in the shared ontology. Similarly, the data can be "lowered" into another XML format. The lifting annotation is used for the first format's schema and the lowering one for the second schema.

SA-WSDL is a simple layer over the WSDL so the SA-WSDL attributes can be directly embedded into the WSDL file, which is published by the service provider. WSMO Studio toolkit provides the user interface of SAWSDL Perspective [3], which allows loading an existing WSDL file and annotating it with the SA-WSDL attributes. The annotated WSDL file is published online and is, together with the corresponding service, automatically registered within the system.

The abstract business process with semantically annotated and grounded tasks is transformed into an executable form of BPEL notation, using the Intalio Designer environment (`http://www.intalio.com`). The executable process workflow can be instantiated and activated in the environment of ESB, which is built on the Java Business Integration (JBI) specification JSR 208.

Fig. 2 presents an overview of general JBI-based ESB, as it was adapted to the platform. The designed JBI-compliant service bus breaks down into the following components:

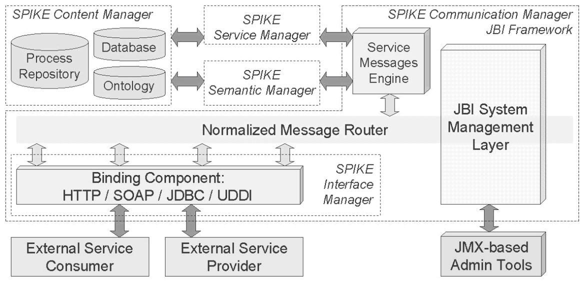

Figure 2: Adapted JBI architecture

- *Normalised Message Router* (NMR), a central messaging backbone of the ESB.
- *Binding Component* (BC) acting as a proxy to a remote service, i.e. it makes remote services available to the service bus in a standardised way - in a form of normalised messages, independently of the service's actual transport protocol and data format.
- *Service Messages Engine* (SME) that provides the business logic during the processing of external services. By wrapping the semantic mediation functionality of the Semantic Manager, the SME transforms normalised XML messages to ontology instances (i.e. semantic lifting of input data) and transforms the output instances back to the normalised messages (i.e. semantic lowering of the mediated data). These transformations are implemented using the standard XSLT technology. Definitions of the transformation, i.e. the XSL style sheets, are specified by means of the SA-WSDL annotations of the message elements [4].

The JBI runtime, integrated into the Communication Manager, provides a standardised environment offering the NMR instance in which the SME and BC can be deployed. A specific runtime environment is created for each of collaboration processes. The configuration artefacts of the service units in such a process will then contain:

- BPEL definitions designed for the process and deployed to the BPEL engine;
- XForms files providing a user interface for human tasks (i.e. the services that are not automatic but require an interaction with human users);

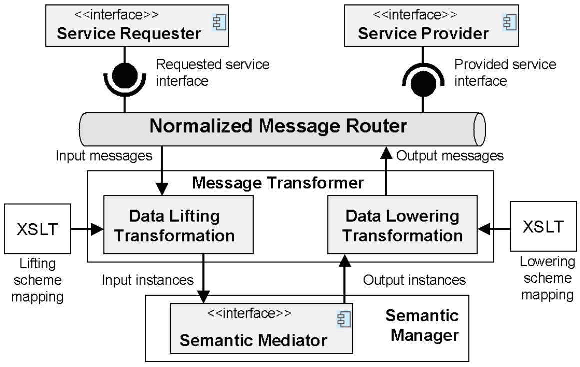

Figure 3: JBI interactions

- WSDL descriptions of external services orchestrated in the collaboration process, semantically annotated using the SA-WSDL annotations;
- resource ontologies referenced from the semantic annotations of WSDL files;
- XSLT transformations used for semantic lifting and lowering of the data to/from the ontology instances in order to mediate the heterogeneous data structure.

To handle XSLT transformations, the Message Transformer was designed as a functional component of the Communication Manager. It implements the SME of the JBI framework. Interactions between the JBI environment, the Message Transformer, and the ontology-based mediation provided by the Semantic Manager are presented in Fig. 3.

A service requester creates a new message exchange request, sends it to the NRM, which then delivers the request to the Message Transformer. In this exchange, the Message Transformer acts as a service provider and sends the requested interface described by means of the SA-WSDL annotation. The SA-WSDL of the requested interface contains definitions of the input messages annotated with the *sawsdl:liftingSchemaMapping* attributes pointing to XSL style sheet for data lifting transformation. The output of the lifting transformation is delivered as a set of instances that semantically represent the input data. The input instances are then sent to the Semantic Manager, which evaluates semantic mappings between the mediated types and infers a set of output instances.

The output instances correspond to the output messages described in the SA-WSDL of the provided interface. Instances are transformed to normalised messages using the XSL style sheet referenced by the *sawsdl:loweringSchemaMapping* attributes. The Message Transformer then creates a new message exchange(s) with the output messages and sends it to the service provider(s) through the NRM - in this case, the Message Transformer acts as a service requester. The return values of the requests (or fault messages) are handled similarly as the input messages; however, the way of transformation is opposite, i.e. the *sawsdl:liftingSchemaMapping* of the provided interface is used for data lifting and the *sawsdl:loweringSchemaMapping* of the requested interface is used to create the exchange results.

4 Platform testing

Semantic structures and technologies, presented in previous sections, were employed in the implementation of the platform prototype. Functional components of the system were created with respect to the overall solution of business collaboration platform, semantically anchored in the structure of developed resource ontologies. The designed abstract process models were semantically annotated, connected with concrete service instances, and transformed into an executable form. The semantic service bus was implemented as a component enabling communication and information exchange between semantically enhanced services operating in the specified workflow.

The platform prototype was tested in the first trial of pilot applications from January till March 2010. The testing was focused on particular system components, aiming to evaluate the functionality of developed modules. Namely, the following features were addressed by the testing procedures:

- Suitability of ontologies and business process models, both abstract and executable;
- Interoperability of services (human tasks, electronic or web services) orchestrated in a workflow and communicated via ESB;
- Identity federation, distribution of access credentials among external applications.

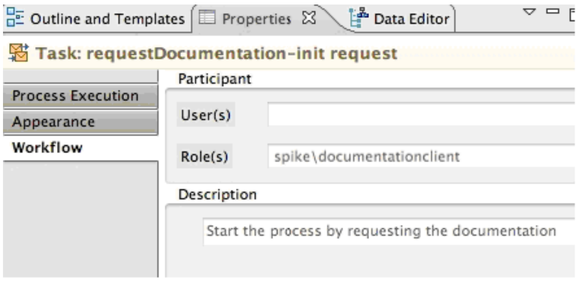

Figure 4: Task specification

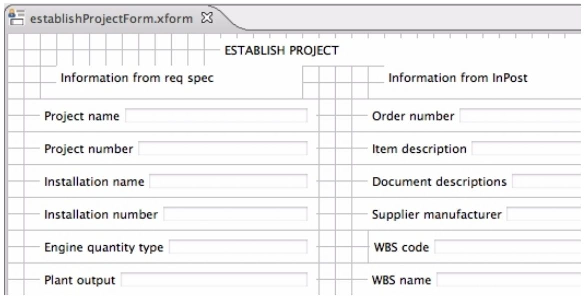

Figure 5: Input form for a human task

Abstract BPMO processes, which were created for each of pilot applications, were transformed into the corresponding executable BPEL representation by the Intalio Designer and Intalio Server tools. Both the selected tools belong to the Intalio BPMS Community Edition solution (`http://community.intalio.com`), which encapsulates a toolkit for visual modelling of business processes in abstract and executable notation. The Intalio solution provides an execution environment for running these processes and supports the BPEL4people extensions for modelling the services performed by human actors in an off-line mode. The latter feature was especially important, since most of services were modelled as human tasks in the first trial. For example, Fig. 4 depicts an Intalio interface for specifying an initial human task *request-Documentation* for starting the process of documentation creation in the Information Hotel AC.

During the run time, the *DocumentationClient* process actor, referenced in corresponding workflow pool, will be asked to fill in an introductory form that specifies an initial request for business collaboration. The input form for this human task needs to be specified in the Intalio Designer, using the XForms format. Since the task is already semantically annotated in the abstract business process model, the form entries are generated according to the ontology concepts presented in the semantic description of the task. The *requestDocumentation* task is annotated by the *CollaborationProcess* concept from the *Business process ontology* and by the *Collaborative Document Production* concept from the domain ontology. Union of attributes of these concepts provides a list of input entry fields that are populated in the XForms representation of the task, presented in Fig. 5.

Online services such as electronic web forms and web services are modelled in a similar manner. In this case, the semantic annotation is already included in WSDL files as the SA-WSDL extension, which is created during the construction of abstract process models and grounded in executable BPEL models. Contrary to the human tasks, there is no need to specify the XForms inputs for online services. Instead, the SA-WSDL representation is employed for seamless exchange of input / output attributes, service parameters and messages. This way, the semantics supports the service interoperability and data mediation for all service types.

Besides the basic orchestration of services, testing included the portlet-based integration of external legacy applications and the related identity federation. The Siemens DAMEX, a toolkit for maintenance of business contracts, was employed as a sample external application within the scope of the Legacy Applications AC. It was included into the workflow and was accessed via ESB as an online service. The security environment, developed within the SPIKE system upon the Shibboleth framework (`http://shibboleth.internet2.edu`), was employed for single sign-on and federation of identities between the external online services [5].

5 Conclusions and future work

Evaluation indicates that the implemented prototype is capable of providing the required functionality. Namely, the underlying semantic structures, ontologies and abstract process models cover well the scope of pilot applications and can serve as workflow specifications for the respective business alliances. Tests performed on the level of components provide a proof of concept that the designed solution, which is based on semantically enhanced ESB, executable workflow engine, security infrastructure, and portal-based visualisation, is suitable for supporting the service interoperability in a business collaboration environment.

However, testing results show also a necessity of some modifications, which will be reflected in further development. The content of domain ontologies should be slightly adjusted towards the actual data, services, and processes of pilot applications. Transformation of abstract process models into the respective executable form needs to be smoothened and automated. The monitoring, reporting, and notification facilities should be added into the business collaboration. And finally, the implemented components need to be integrated into a unified platform.

6 Acknowledgments

The work presented in the paper was supported by the SPIKE project (http://www.spike-project.eu) co-funded by the European Commission within the contract No. 217098 and partially by the Slovak Grant Agency of the Ministry of Education and Academy of Science of the Slovak Republic within the 1/0042/10 Project "Methods for identification, annotation, search, access and composition of services using semantic metadata in support of selected process types".

References

[1] Belecheanu R. et al: Deliverable 1.1. Business Process Ontology Framework. Project IST 026850 SUPER, Public Deliverable, 2007.

[2] Chappell D.A.: Enterprise Service Bus. Sebastopol, CA: O'Reilly Media, Inc., 2004.

[3] Dimitrov, M. et al: WSMO Studio Users Guide. Ontotext Lab., 2007.

[4] Farrell J., Lausen H.: Semantic Annotations for WSDL and XML Schema [on line]. W3C recommendation, August 2007, available at http://www.w3.org/TR/sawsdl/, Accessed: 16th March 2010.

[5] Fernández C. et al: D7.3 Implementation of Security Components for Service Bus Sub-System. Project FP7-ICT-217098 SPIKE, Public Deliverable, 2009.

[6] Furdík K., Mach M., Sabol T.: Towards semantic modelling of business processes for networked enterprises. Proc. of the EC-Web 2009, Linz, Austria, September 2009, LNCS 5692, Springer, 2009, pp. 96-107.

[7] Giangarra P., DeMeester J.: Enabling Network-Centric Operations with Semantic Web Technologies [on line]. W3C submission, April 2005, available at http://www.w3.org/2005/04/FSWS/Submissions/14/Paper.pdf, Accessed: 16th March 2010.

[8] Hepp M. et al: Semantic Business Process Management: A Vision Towards Using Semantic Web Services for Business Process Management. Proc. of the IEEE ICEBE 2005, Beijing, China, 2005, pp. 535-540.

[9] i2010 - A European Information Society for growth and employment. Brussels, EC, COM(2005) 229 final of 1 June 2005.

[10] Karastoyanova D. et al: Semantic Service Bus: Architecture and Implementation of a Next Generation Middleware. Proc. of the 2nd International Workshop on Services Engineering (SEIW) 2007, IEEE, 2007, pp. 347-354.

[11] Klischewski R., Ukena S.: Designing Semantic e-Government Services Driven by user Requirements. Electronic Government, EGOV 07 conference. Proc. of ongoing research, project contributions and workshops. Linz, Austria, Trauner Verlag, 2007, pp. 133-140.

[12] Mach M., Bednár P., Furdík, K.: Support for Forming Temporal Business Alliances as Networked Enterprises. Proc. of CECIIS 2009. Varaždin, Croatia, University of Zagreb, FOI, 2009, pp. 179-186.

[13] Vitvar, T., Kopecky, J., Fensel, D.: WSMO-Lite: Lightweight Semantic Descriptions for Services on the Web. WSMO Deliverable D11, Ver.0.2. DERI, 2008.