Correspondence between Multiple Views in a SOA Trans-National Business System

Anca Daniela Ionita, Monica Florea and Lucian Jelea

Abstract —**The difficulties of developing an e-Business system are generated by the need to simultaneously attain objectives like: user friendliness for the multiple actors, maintainability and extensibility under business evolution conditions, flexibility for dynamic retrieval of service providers, traceability of the system and user performance. The paper presents the correspondences established between use-case, process, logical, implementation and deployment views in LD-CAST system, such as to attain these objectives, while hiding the back-stage complexity for the users pertaining to the business domain. The solution is based on Service Oriented Architecture (SOA), Business Process Modeling (BPM) and semantic web.**

Index Terms— **e-Business, Service Oriented Architecture, UML Modeling, Web Service Orchestration**

I. Introduction

Electronic services are currently developed for a large variety of applications, dedicated to citizens, businessmen, civil servants, medical staff and teachers. Interoperability frameworks are being created for the cooperation and integration of public institutions, as well as for supporting business development, with the aim to address organizational, semantic and technical issues [1]. The scope of this paper is related to applications for business development in the context of the enlarged Europe, in connection with the e-services frameworks existent at national levels.

Trans-border processes involve difficulties related to: multi-actor environments that require multilateral agreements; non-homogeneity of security policies of the involved institutions; non-interference with the internal work of administrations, enterprises or institutions that deliver services; the necessity to create customer-oriented services, processes and tools with strong multilingual facilities.

The technical problems arise from the necessity to create large-scale distributed systems that may be accessed through a unique, one-stop portal, hiding the existent non-homogeneity of procedures, document formats and legislations among European countries. Service Oriented Architecture [2] offers possible solutions for these problems, by supporting atomic and composite services, eventually delivered by multiple cross-border Service Providers (SPs). Semantic non-homogeneity can be treated by defining ontologies [3] that characterize generic processes, abstract web services and service providers from various countries, as well as common quality of service (QoS) notions, necessary for establishing service level agreements. Sometimes, correspondences between local ontologies (either regional or national) and central, European level ones, have to be defined, implemented and maintained [4]. The present e-Business systems are confronted to the following challenges:

1) integrate heterogeneous types of business services into seamless processes - at local, national, and Pan-European levels - eliminating the loose electronic integration of activities for SPs from different European countries and regions;
2) avoid the ambiguity in defining documents, processes, and services;
3) improve the automation of the request resolution inside SPs;
4) improve validation and monitoring of services and processes provided by multiple SPs;
5) attain flexibility in service and process reconfiguration and maintenance;
6) dynamically propagate changes in service and process semantic definitions to registered services and executable processes;
7) develop personalized and easy-to-use interfaces for non-IT personnel;
8) supply automatic guidance for all sorts of users, especially for non-specialist ones.

The LD-CAST system [5] presented in this paper aims to support the development of private company initiatives, belonging either to private or public sectors, by cross border cooperation between European Chambers of Commerce (CCs). A one-stop portal at European level is dedicated to existent and new start-up enterprises, created by investors or entrepreneurs interested in an innovative line of business, based on trans-national development and growth. The semantic interoperability between different CCs acting as service providers is assured by an ontology containing common, core elements, specialized then on national-specific concepts. The technical interoperability stands in the framework developed for modeling, executing and monitoring business processes that seamlessly compose and localize services delivered by different institutions, from

Manuscript received March 2, 2009. This work was supported in part by the European Community, under the FP6 IST Programme, FP6-2004-IST-4-26919 project (LD-CAST) and in part by CNCSIS, Romania, under Grant A-37/2006.

A.D. Ionita is with the Computers and Industrial Informatics Department, University "Politehnica" of Bucharest, Spl. Independentei 313, 060042, Bucharest, Romania (phone: 4-021-4029113; fax: 4-021-3181014 ; e-mail: Anca.Ionita@mag.pub.ro).

M. Florea is with SIVECO ROMANIA SA, Victoria Park, Sos. Bucuresti-Ploiesti 73-81, 013685, Bucharest, Romania (e-mail: Monica.Florea@siveco.ro).

L. Jelea is with SIVECO ROMANIA SA, Victoria Park, Sos. Bucuresti-Ploiesti 73-81, 013685, Bucharest, Romania (e-mail: Lucian.Jelea@siveco.ro).

different countries, while respecting their specificities, procedures, as well as privacy and security rules and regulations.

The paper describes the LD-CAST system in chapter II, using UML 2 models [6], organized according to 4+1 views architecture [7]. It also presents the correlations between these views, outlining the interoperation between subsystems accessible to business domain users and those hidden to them, dedicated to actors responsible for the system configuration and maintenance (more details in chapter III). Chapter IV analyses some works that are related to LD-CAST approach, many of them also supported by European projects.

II. 4+1 Views for a Trans-National Business System

A. Actors and Use Cases

LD-CAST is mainly dedicated to serve entrepreneurs interests to obtain electronic business services, in order to save their time and money and to attain confidence related to unknown chambers of commerce and potential partners from abroad. Any *Guest* is able to obtain detailed information related to business cooperation, but one has to register and become an *End-User,* in order to get advantage of the business services published within the framework (see the use case diagram in Fig. 1).

However, LD-CAST system had to assure the dynamic discovery of web services, taking into account the client preferences, the common descriptions for services and processes, and the respect of the subsidiary principle, by allowing each chamber of commerce to follow its specific procedures. In order to acquire these requirements, the system needs specialists with various competences for its configuration and maintenance. A *Business Process (BP) Designer* is in charge of modeling specific processes, whose activities are to be implemented by web services (see Fig. 2). The late mapping of activities on concrete services is realized through the concepts and relationships of the business ontology, created and managed by the *Knowledge Engineer.* Thus, the process is executed with actual web services, selected from a repository where they had been published by persons playing the role of *Service Provider Administrator.* The security and integrity of LD-CAST are assured by an *LD-CAST Administrator*, who can also monitor the clients and the system performance.

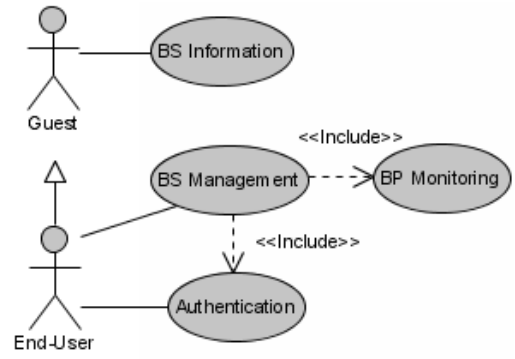

Fig. 1 Use case diagram with non-specialist actors

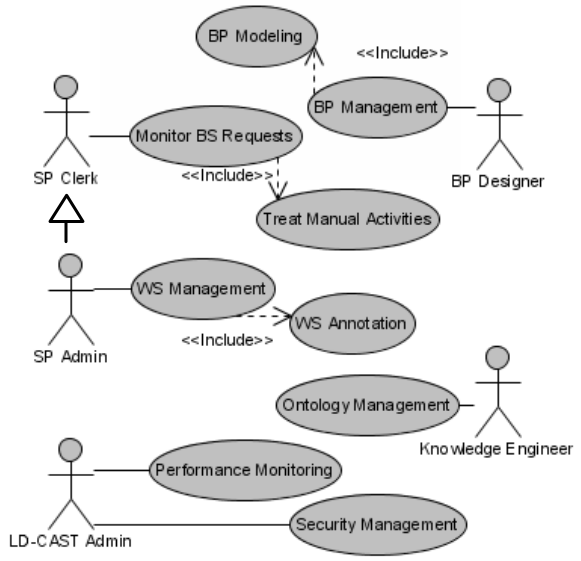

Fig. 2 Use case diagram with actors involved in LD-CAST configuration and monitoring

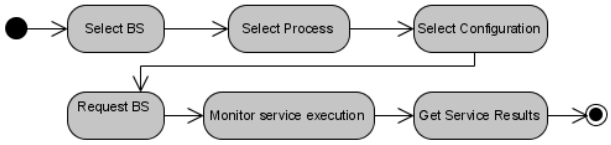

Fig. 3 Activity diagram for Business Service Management

Even if the system was conceived to automate the delivery of business services, the practice revealed that some of the tasks still have to be performed manually; therefore, a *Service Provider Clerk* was introduced, who can also monitor all the business service requests assigned to its service provider.

B. Typical Scenario for the End-User

The process view of LD-CAST is rather complex, so, for the purpose of the paper, one selected a typical scenario, correspondent to the *Business Service Management* use case and represented in Fig.3. First, the *End User* has to select the desired business service (for example *Company Legal Verification*) and the portal displays a list of processes that have been published for it, reflecting specificities characteristic for a certain country or even for a certain chamber of commerce. After selecting a process (for example *Release Ordinary Visura*, from Italy) the system searches the configurations of concrete web services that can be used for the actual execution and gives detailed information about them, as shown in Fig.4.

The *End User* selects the preferred configuration and starts the execution of that business process, then monitors it, by actively getting involved in certain activities. At the end, it is possible to download the expected results, which have been stored in a repository of the one-stop portal.

C. The Logical View of a Business Service in LD-CAST

From the logical point of view, behind the execution of the scenario above, there are three layers (see Fig. 5). First, there is a Modeling Layer, containing the concepts of Business Service (BS), which has associated Business Processes (BP), composed of activities. LD-CAST has identified the

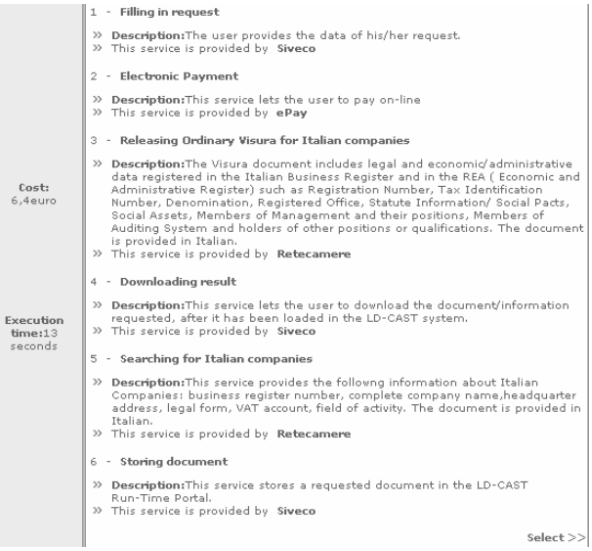

Fig. 4 Details related to a configuration of concrete services

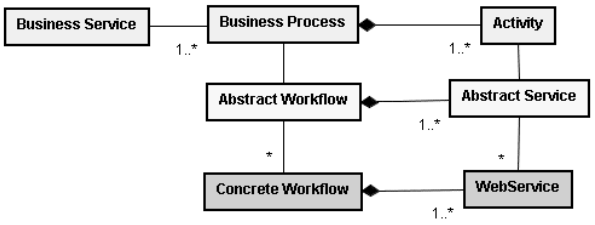

Fig. 5 The logical view with the execution layers [8]

following business services that can be delivered by the chambers of commerce: Partner search, Company legal verification, Company fiscal verification, Technical and quality standard verification, Agreement, Registration, Mediation, Arbitration and Supply subcontract. However, each of them may be delivered by different chambers, from different countries, involving a 1-to-many correspondence between a business service and its business processes, defined by a domain expert playing the role of *BP Designer*.

In order to automate these processes, they have to be transformed into executable workflows. As LD-CAST flexibility requirement imposes a late binding of web services, there are actually two layers of execution: an *Abstract Execution Layer* - where there are abstract workflows and abstract services, defined at design time - and a *Concrete Execution Layer* - containing concrete workflows, determined at run time, by semantically searching the available concrete web services that may implement the abstract activities.

D. LD-CAST Distributed Architecture

The LD-CAST architecture is service oriented and all the connections between its subsystems, depicted in Fig. 6, are realized through web services, using SOAP [9]. The subsystems accessed by each of the actors mentioned above are presented as follows.

The *Guest* and the *End User* work with the *Run Time Portal* for obtaining information and business services from LD-CAST. The portal is implemented with JEM (JBoss Enterprise Middleware) [10] and performs its tasks dynamically, by interacting with other subsystems.

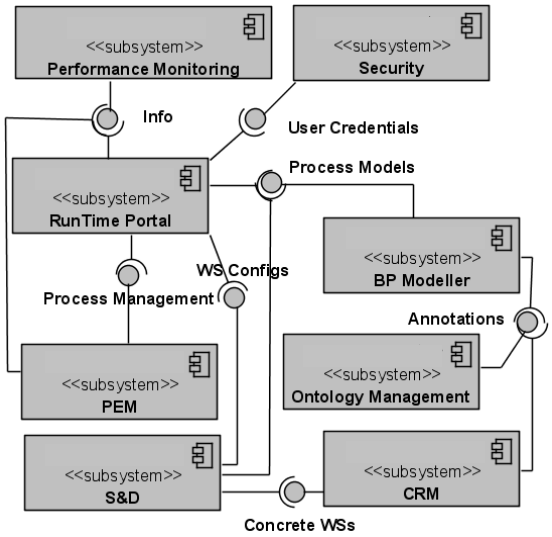

Fig. 6 LD-CAST subsystems and their connectors

The *Business Process Designer* uses the *Business Process Modeler* (BP Modeler) subsystem, for defining process models with ADOeGov [11] modeling language, which allows process activities to be linked with abstract workflows written in BPEL4WS (Business Process Execution Language for Web Services) [12].

The *Knowledge Engineer* uses the *Ontology Management* subsystem to create the reference business ontology, using OPAL (Object, Process, Actor Modeling Language) [12]. This ontology, related to the chambers of commerce domain, is necessary for semantically annotating business process activities and web services.

The *Service Provider Administrator* accesses the *Concrete Resource Management* (CRM) subsystem, in order to store information about the service provider, as well as to register, annotate and publish the delivered web services. Information is stored in a repository that may be accessed by the *Search and Discovery* (S&D) subsystem, which is able to perform a semantic matchmaking between the set of available web services and the abstract services corresponding to a requested business process. The configuration of actual web services or manual activities that implement the workflow activities is not fixed in advance, but established through a selection made by the *End User* at run time, according to their registered cost and time. Thus, one supports a late binding and the generation of concrete BPEL workflows, whose execution is performed by the *Process Execution Management* (PEM) subsystem, using the ActiveBPEL engine. S&D and PEM do not interact directly with any of the system actors.

The *Security* subsystem supports a federated identity solution for authentication and authorization, based on Shibboleth [14]. LD-CAST relies on the definition of a circle of trust with its local agencies - where *End Users* are registered - while the management of other groups of users is performed directly, by the *LD-CAST Administrator*; moreover, he or she can also verify the system and the user efficiency, with the help of the *Performance Monitoring* subsystem, which collects logs from other subsystems and supplies a centralized monitoring console for the desired criteria.

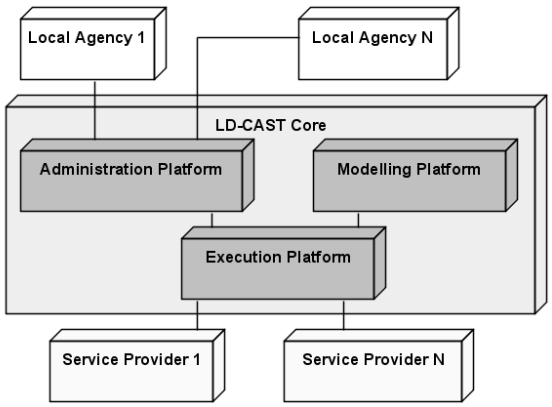

Fig. 7 LD-CAST deployment diagram

E. LD-CAST Deployment on Three Platforms

A primary reason for choosing SOA was the high degree of geographical distribution of LD-CAST, which is twofold. On one side, there is the deployment of the subsystems depicted in Fig. 6, which constitute the *LD-CAST Core* and for which the minimum number of nodes is 3, corresponding to the following platforms:

1) The *Modeling platform* is dedicated to all the tasks necessary for system set up and configuration, and can be accessed by actors like*: BP Designers*, *Knowledge Engineers*, and *SP Administrators*. The following subsystems are part of this platform: *Business Process Modeler, Ontology Management* and *Concrete Resource Management.*
2) The *Execution platform* supports the automatic execution of business services, by integrating the *Run Time Portal*, the *Process Execution Management* subsystem and the *Search and Discovery* one. Even if the portal is mainly dedicated to *End Users*, it also allows the *LD-CAST Administrator* to manage its content, and supports the delivery of non-automatic services by the *Service Provider Clerks*, after introducing the proper credentials.
3) The *Administration Platform* is used by the *LD-CAST Administrator* to manage the users of LD-CAST (by interacting with *Local Agencies* - for *End Users* - and directly - for the others); moreover, it controls the efficiency of the system operation, using the *Performance Monitoring* subsystem.

Actually, the prototype was deployed on more execution nodes, in respect with responsibilities assumed by different members of the Consortium for the development and maintenance of various subsystems, even inside the same platform. The SOA architecture allows this flexibility, in terms of optimizing the communication times of its connectors.

On the other side, there are also two other kinds of entities, with their correspondent deployment nodes (see Fig. 7):

- The *Local Agencies*: responsible for the payment and for the local management of users that have access to the LD-CAST Core services; one becomes an *End User* of LD-CAST and is able to request business services only after registering and identifying at a local agency, following its internal procedure; the agency acts as an identity provider in the federated network identity.
- The *Service Providers*: which provide manual and web services that will be orchestrated by LD-CAST Core, in order to satisfy an *End User* request; they register and publish their services using the LD-CAST Core subsystems, through the *SP Administrators*.

III. Support for System Configuration and Maintenance

For being able to support a typical scenario, as described above, the *Run Time Portal* requires services from:

- the *Security* subsystem, to get the user credentials;
- the *BP Modeler*, to get the list of business services and the descriptions of the processes published for them (an extract of the exchanged information in XML format is given in the Appendix);
- the *Process Execution Manager*, in order to start and monitor the execution of a concrete workflow, selected by the *End User*;
- the *Search and Discovery* subsystem, to get the list of configurations available for building a concrete workflow;
- the *Performance Monitoring* subsystem, in order to supply information about *End Users* operation and also for statistics related to the selected configurations, business processes and business services.

Thus, the *Run Time Portal* has the biggest coupling and LD-CAST system has to be properly configured in order to allow its coherent operation. On one side, for the desired business service, at least one process has to be modeled with *BP Modeler*, has to be annotated with concepts of the *Ontology* and has to be associated with an abstract workflow. On the other side, for assuring the executability, for each abstract activity there should be at least one concrete web service that has been registered at CRM, annotated similarly as the activity and published. Then, the business process can also be published, in terms of existing at least one concrete workflow corresponding to it.

LD-CAST has been designed as a system that is highly adaptable to change, as the system has to evolve in the rhythm of the business domain. Some important facilities for maintaining LD-CAST are: adding new concepts in the ontology (performed by the *Knowledge Engineer*); defining new processes (by the *BP Designer*); registering new web services (by the *SP Administrator*); accepting new local agencies and service providers (by the *LD-CAST Administrator*) [15]. The difficulty stands in keeping the system coherent, under the conditions of multiple subsystems and multiple actors involved in system extending. At the same time, the *End User* must be oblivious to the existence of all the subsystems behind, but should notice the benefits gained through the potential of increasing the number of available business services and configurations corresponding to them.

IV. Discussion

For avoiding predefined mappings between abstract and concrete services and for taking into account quality criteria and user choices, SOA generally uses Semantic Web Services. The role of semantic web for assuring interoperability inside highly distributed systems has also

been outlined by other European projects. BRITE project [16] uses ontologies to mitigate semantic heterogeneity of information, data and processes, supporting proactive information delivery and coping with multiple levels of formality on information and data levels. SEEMP [4] investigated interoperability between non-homogenous e-Government systems for the employment sector, pertaining to different countries among Europe. Their solution for resolving heterogeneity was to convert all the XML semantic content in terms of a reference ontology, shared by all partners, and then to lower it back to an XML corresponding to a different local ontology. Technologically, this is achieved using the WSMX Data Mediation [17].

SemanticGov project [18] utilizes the WSMO framework (implementing WSML ontologies) and Governance Enterprise Architecture (GEA) models for functionally restructuring public administration services and enhancing interoperability.

A difficulty encountered in LD-CAST was that the user interface cannot be designed statically, but has to be adapted in respect with the input and output formats of whatever new services registered to the system when it evolves. Moreover, service orchestration [19] involves creating composed services, involving composed input and output forms, whose configurations is not known in advance, but is decided by the *End User,* at run time. A similar requirement can be observed in SEGOF project [20], which proposes a way to generate e-Government forms semi-automatically, based on semantic descriptions of public administration services and their business rules. Data conforming to the ontology restrictions and represented as XML and RDF (Resource Description Framework) can be consumed by any application supporting the data interchange standard, including semantic web services.

Another important issue in LD-CAST was that business processes have to be defined at a high level of abstraction, by business domain experts, but they have to be rendered executable, in order to assure the proper user friendliness of the system. A possible solution, adopted for Apel [21], is to create a Domain Specific Language (DSL) for defining process models and to use the DSL interpreter as execution engine. Another solution (adopted in LD-CAST also) is to use a specific language for defining business processes and then to translate it into executable BPEL workflows, for which execution engines already exist. In the second case, the problem is that automatic generation of BPEL code encounters a number of limitations: different process engines require different workflow code and domain-specific models need complex graph transformation algorithms for coming from a graph-based to a block-based structure. Important achievements were obtained in AgilPro project [22], where one uses a generation framework for a component-based design, which fosters reuse and composition for parts of generation solutions.

V. Conclusion

LD-CAST system was designed to offer a common framework for automated or semi-automated business services realized by various processes, obtained by orchestrating late bound web services. One of the big challenges was to offer flexibility in respect with the clients' preferences, the national practices and the diverse terminology. The idea was that *End Users*, even if registered at *Local Agencies* that may be geographically distributed all around Europe, should retrieve a friendly system and should pass seamlessly through all the necessary steps for treating their business service requests, without perceiving the complexity behind.

By describing the use case, process, logical, implementation and deployment views of LD-CAST, the paper outlined how SOA, semantic web and Business Process Modeling have been combined for attaining requirements that have different priorities for different types of actors: user friendliness and security for the *End Users*, maintainability for the *Knowledge Engineers*, extensibility for the *Business Process Designers* and *Service Provider Administrators,* and traceability for the *LD-CAST Administrator*.

Appendix

One presents below a part of the XML description for the *Company Legal Verification* business service, as transmitted by the *BP Modeler* subsystem to the *Run Time Portal*, which subsequently displays information to *Guests* and *End-Users*.

<ATTRIBUTE name="PortalGroup" type="STRING"> Company Legal Verification,Italy</ATTRIBUTE>
<ATTRIBUTE name="gdShortName" type= "LONG STRING"> Company legal verification </ATTRIBUTE>
<ATTRIBUTE name="gdLongName" type="LONG STRING"> Check the legal status of a partner company, in order to verify its keeping of the operational current legal rules.</ATTRIBUTE>
<ATTRIBUTE name="gdDescription" type="LONG STRING"> The legal verification includes legal and economic/administrative data, such as Registration Number, Tax Identification Number, Denomination, Registered Office, Statute Information/ Social Pacts, Social Assets, Members of Management and their positions, Members of Auditing System and holders of other positions or qualifications - depending on the country. Various certificates supply information about the life and events related to the verified company.</ATTRIBUTE>
<ATTRIBUTE name="gdResults" type="LONG STRING"> The enterprise certificate of existence, joint stock, corporate form; Criminal record; Business History Certificates; Business profile information; Certificates of non-involvement in organized crime.</ATTRIBUTE>
<ATTRIBUTE name="gdNote" type="LONGSTRING"> The inputs and outputs differ from one country to another and also depend on the organization that supplies the service. The exact information may be obtained after registration to LD-CAST, through of its Local Agencies.</ATTRIBUTE>

Acknowledgment

We would like to thank to the LD-CAST Consortium, including both technical partners and chambers of commerce, for the experience we have gained together related to the interoperation between people and software, as well as related

to the delegation of responsibilities to multiple types of users and to developing the correspondent tools. We would also like to express special thanks to ELSAGDATAMAT (Roma, Italia) for their support for integrating and maintaining the system.

REFERENCES

[1] *The European Interoperability Framework.* Available: http://europa.eu.int/idabc/en/document/3797/194

[2] B. Fitzgerald and C. M. Olsson Eds. "The software and services challenge. Contribution to the preparation of the technology pillar", *Software, Grids, Security and Dependability of the 7th Framework Programme*. Ver 1.1. 30 Jan, 2006.

[3] D. Gašević, D. Djurić, V.Devedžić, *Model Driven Architecture and Ontology Development*, Springer-Verlag, 2006.

[4] E. Della Valle, D. Cerizza, I. Celino, J. Estublier, G. Vega, M. Kerrigan,J. Ram´ırez B. Villazon, P. Guarrera, G. Zhao, and G. Monteleone, "SEEMP: an semantic interoperability infrastructure for e-government services in the employment sector", *Lecture Notes in Computer Science*, Vol. 5419, 2007, pp. 220-234.

[5] *LD-CAST project*. Available: http://www.ldcastproject.com

[6] "UML 2.0 Superstructure Specification", Object Management Group, 2005. Available: http://www.omg.org

[7] "Applying 4+1 view architecture with UML2", *FCGSS White Paper*, 2007

[8] Based on a figure from A. D. Ionita, A. Catapano, S. Giuroiu, and M. Florea, "Service oriented system for business cooperation". In *Proceedings of ICSE Int. Workshop on Systems Development in SOA Environments* SDSOA '08, Leipzig, Germany, May 11 - 11, ACM, New York, 2008, pp. 13-18.

[9] SOAP Version 1.2, Available: http://www.w3.org/TR/soap12-part1/

[10] JBoss Enterprise Middleware, Available: http://www.jboss.com

[11] S. Palkovits, D. Orensanz, D. Karagiannis, "Process modelling in Egovernment – Living process modelling within a public organisation", *IADIS International e-Society*, 2004, pp. 3-10.

[12] OASIS. "Web Services Business Process Execution Language for Web Services" Version 2.0, 2007. Available: http://docs.oasis -open.org/wsbpel/2.0/ wsbpel-v2.0.pdf

[13] F. D'Antonio, M. Missikoff, F. Taglino, "Formalizing the OPAL eBusiness ontology design patterns with OWL. Business episodes and workflow integration: A use case in LD-CAST", *Third International Conference on Interoperability for Enterprise Applications and Softwar*e, I-ESA 2007, pp. 345-356.

[14] M. Erdos, S. Cantor, "Shibboleth Architecture", Draft version 5. Available: http://shibboleth.internet2.edu

[15] A. Catapano, A. D'Atri, V. Hrgovcic, A.D. Ionita, and K. Tarabanis, "LD-CAST: Local Development Cooperation Actions Enabled by Semantic Technology", In *Proceedings of Eastern European eGov Days Conference*, EEGOV, Prague, Czech Republic, April 23-25, 2008, pp. 22-27.

[16] T. Herborn, A. Mondorf, B. Mougouie, M. A. Wimmer, "Semantic interoperability in the BRITE project: Ontologies as a tool for collaboration, cooperation and knowledge management", In R. Meersman, Z. Tari (eds): "On the move to meaningful Internet systems", *Lecture Notes in Computer Science,* Monterrey, Mexico, Vol. 5331, Springer, 2008, pp. 475-483.

[17] A. Mocan, E. Cimpian, M. Kerrigan, "Formal model for ontology mapping creation". In "The Semantic Web - ISWC 2006", *Lecture Notes in Computer Scienc*e, Springer, Volume 4273, 2006, pp. 459-472

[18] V. Peristeras, and K. Tarabanis, "Reengineering the public administration modus operandi through the use of reference domain models and Semantic Web Service technologies", In *Proc. of AAAI Spring Symposium on The Semantic Web meets eGovernment* SWEG, Stanford University, California, USA Mar. 27-29, AAAI Press, 2006, pp. 56-63.

[19] D. Peltz, "Web services orchestration. A review of emerging technologies, tools and standards". *Hewlett Packard, Co*. January 2003.

[20] B. Stadlhofer, and P. Salhofer, "SeGoF: semantic e-government forms". In *Proceedings of the 2008 international Conference on Digital Government Research,* Montreal, Canada, May 18 - 21, 2008, ACM International Conference Proceeding Series, vol. 289. Digital Government Society of North America, 2008, pp. 427-428.

[21] J. Estublier, S. Sanlaville, "Extensible process support environments for web services orchestration", *International Journal of Web Services Practices* (IJWSP), Vol. 1, No. 1-2, 2005, pp. 30-39.

[22] St. Roser, F. Lautenbacher, and B. Bauer. "Generation of workflow code from DSMs". *Proceedings of the 7th OOPSLA Workshop on Domain-Specific Modeling*, Montreal, Canada, October 2007.