

Utilizing Ontologies for enhancing Mobile Service Registries: The ONAR approach

Dimitrios Tektonidis
 ALTEC S.A.,
 6 M.Kalou,
 54629, Thessaloniki, Greece
dte@altec.gr

Albert Bokma
 University of Sunderland,
 St Peter Campus,
 SR6 0DD, Sunderland, United Kingdom
Albert.Bokma@sunderland.ac.uk

Abstract— The significant increase in the number and the diversity of mobile service arises problems of discovery, categorization and description to the service oriented architecture of mobile services systems. In DESYME project aims to develop a collaboration environment for mobile service services. In this paper we present the methodology and the framework (ONAR) used for providing the necessary infrastructure to enrich existing service registries (UDDI –based) used from mobile service frameworks (OSA/PARLAY X) with semantics.

Index Terms— Semantically enriched services, Ontologies , mobile service requirements, Enterprise Application Integration.

I. INTRODUCTION

The utilization of mobile service inside a network of collaborated party (company or developers) can be approached as a specialization of Service Oriented Application Integration[5] (henceforth SOAI). Although there are significant differences, in principal there is much communality since the utilization of the provided web-services is an integration with the back office of the mobile services system.

Of course, contrary to the Enterprise Application integration in the field of the mobile services there are frameworks like PARLAY X[21] that offers the necessary infrastructure for services integration. However the problem as it is defined in DESYME project[22] is that the diversity of the service that can potentially be used from a mobile network can not be covered from the technical specification of a framework such as Parlay.

The Ontologies Based Enterprise Application Integration (ONAR) Framework [7], as it is used inside DESYME Project, utilizes web ontologies to create semantic conceptualizations of the business concepts that exist inside a system or a network of mobile services. This conceptualization is used for the creation and the registration of the Web services in a UDDI based registry.

In this paper we present our work in the direction of semantic

description of mobile services and the utilization of ontologies for enhancing UDDI based registries.

II. ONAR MOBILE SERVICES CONCEPTUALIZATION CYCLE

The creation of Semantics for definition of mobile service is process where the semantic concepts and their relations are defined. Regardless the structure and the architecture of the existing Information System (in this case PARLAY X and Service Oriented Architecture) ONAR Framework enables the definition of semantics. This ONAR Conceptualization Cycle consists of the following phases:

1. **Conceptualization phase** where the existing applications are conceptualized to ontologies based on the requirements of the integration.
2. **Association phase.** The conceptual schema is associated to the legacy system resources that exist in the system repository.
3. **Design phase.** The design of a software instance (Web-Service) is based on the information that is contained (in the conceptual and technical level) in the ontologies. The consultant chooses the parts of the ontologies that need to be included inside the software instance.
4. **Deployment phase** where the software instance is deployed to the web-server.
5. **Publication phase.** The framework uses a UDDI registry enriched with OWL-S profile features (from semantic invocation) to publish its software instances. This registration takes place in the publication phase.

In this paper is presented the application of ONAR framework and its adaptation to enhance service registries for mobile services.

A. Mobile system conceptualization using OWL

Service Oriented Applications, like PARLAY X, follows a component based logic that has many differences to object or frame based logic that modern ontology languages use. During our research

[2][3] we have found that languages like the Web Ontology [12] are capable to provide us with formal definitions for business concepts as OWL classes even in non object oriented structure like in PARLAY X interface.

OWL descriptiveness enables the definition of semantics for the business concepts and their attributes. The entities inside the ONAR conceptualization methodology inherit all the features of the OWL classes. However there are certain problems foreseen for permitting abstract definition of domains that can lead to the definition of conceptualizations.

The venture of having an ontology language for defining information that is exchange between information systems has been confronted with the definition of a restricted usage of OWL. In this attempt, graphical environment was developed in order the definition of the OWL ontologies to be constrained by certain rules that would prevent from the false usage of OWL. ONAR Concepts and Services Designer [7] enables the creation of conceptualization ontologies (OWL) providing the business consultant a reduced OWL-Lite ontology [12].

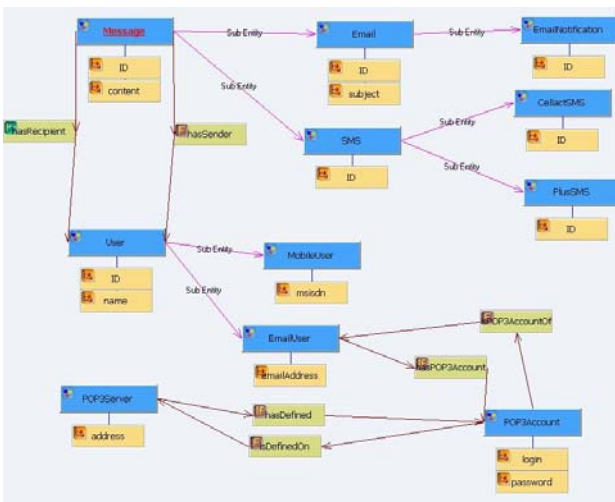


Figure 1: The graph of a conceptualization ontology produced by ONAR concepts and services designer representing concepts of a mobile service.

Figure 1 presents the conceptualization of the summary of Mobile Messages ontology used in DESYME project. This ontology contains all the concepts (OWL classes) and their properties (OWL Datatype properties) as well as their relation (OWL object properties).

The graphs appears that does not represent a formal OWL structure since for example there are properties (OWL Data-types) like the property "ID" that is presented more than once in the graph. Our framework uses OWL-Lite as ontology language however for reasons of presentation based on the end users feedback we support that this kind of ontology representation is more convenient and less complex to user that are not familiar to ontology engineering.

Summarizing the conceptualization capabilities of our framework¹ we are using the OWL-Lite and more precisely:

- *Inheritance* is permitted allowing some class features to be inherited from one class to another. The relation between the parent class and child classes follows the same rules that both object oriented engineering and frame-based logic supports, however multiple inheritance is not permitted. This constraint is due to the

fact that in complex information system polymorphism of some concepts will unduly increase the complexity of conceptualization.

- Entities are allowed to have *properties* that correspond to OWL datatype properties. A property can belong to more than one entity and is contrary to the usual principles of object oriented modelling. This principle derives from the frame based logic of OWL ontologies and extends the object oriented engineering where a property belongs to only one class.

Relations between concepts can be defined using:

- *Functional property*: that defines the "one to many" relation between the instances (objects) of two concepts.
- *Inverse functional property*: complementary to functional property in order to facilitate lexical descriptions.
- *Symmetric property*: Symmetric properties are used to declare the one to one elation between the instances (objects) of two concepts.
- *Transitive properties*: Contrary to the definition of OWL in ONAR framework Transitive properties are used for the division of complex concepts. For example the concept "message" can have properties regarding its content features (recipient, message content), managerial features like (delivery status, delivery date) etc. Using transitive properties the concept can be divided to more than one concept according to its semantic.

However we have to limit the syntax of OWL which permits the existence in a relation of more than one concept as ranges and domains. In our approach (in a relation) if the domains contain more than one concept then the range should be defined from only one concept and vice versa. This ensures that in the association of the logical model to the system resources will not create indeterminism.

Both entities and properties are enriched with (a) notional properties like lexical descriptions, (b) the ability to define the inverse of a concept and (c) the ability that a property can potentially be derived from another property. These enrichments are used to increase the inference for the semantics of the concepts and their properties inside a model following the OWL principles.

The most important reason for the utilization of ontologies is the capability to define semantic relations between the concepts of a domain. Mecella et al [13] introduced the concept of application wrapper as meta-layer enables the redefinition of concepts of the application in order to be used from other heterogeneous applications. The lack of semantics limits the capabilities of EAI approaches like Application Wrappers and Object Oriented Middleware (OOM) techniques.

Defining information using data definition methods like UML Class diagrams or Entity Relation (E-R) schemas the produced conceptualizations unavoidably follow the data structure of an application. This is the actual limitation that our work needs to overcome. Software engineering methods have limited capabilities for semantics definition. For example, the semantics in the relation between two concepts that both in UML or in E-R schemas do not have formal methods to define.

Our work has focused on creating semantics in the relations between concepts. The existing semantics of relations that are expressed in object oriented class diagrams in UML or Entities Relations (E-R) diagrams pose limits to the semantic two concepts may have at the logical level. Therefore adapting the four types of relations (object relations) that are defined in OWL we are achieved to increase the inference capabilities of the conceptualizations. The

¹ Further information is available on ONAR project Web Site online documentation (<http://research.altec.gr/onar>)

relations between concepts in this level represent semantic relation between the concepts. These concepts may have a syntactic relation however this is not requisite. This detaches the definition of the concepts from their data realizations. Thus the definition of the concepts is based to the integration requirements.

Concepts relations are defined based on their domain and their range. The domain is consisted of set of concepts that subject of the relation. The range for a relation is the object or the objects of the relation. According to Wache et al [14] the ontological models do not limit the number the domains and ranges that are related to a concepts. However in our research based on e-business [1], e-Government [19] and e-Health [20] use cases the simultaneous existence of multiple domain and range ontology created problems regarding the comprehension of the ontology as well as in the association of the concepts to the repository of the information system.

B. Wrapping Service Oriented Mobile service systems

ONAR approach aims to be independent from that application logic of each application necessitates the existence of mechanisms and technique to associate logical conceptualizations (OWL ontologies) to the real application repository. In addition our approach is independent also from the type of the repository (whether is a RDBMS or an Application server).

In our attempt to enrich OWL with elements that would enable as to associate the conceptualization to the system we have discovered that these enrichments would create a complex Ontology file that will carry both semantic and syntactic elements. Another important issue was the portability of the ontologies from different kind of systems. Therefore we have created an association ontology that enables the mapping of the conceptualization ontology to the system repository.

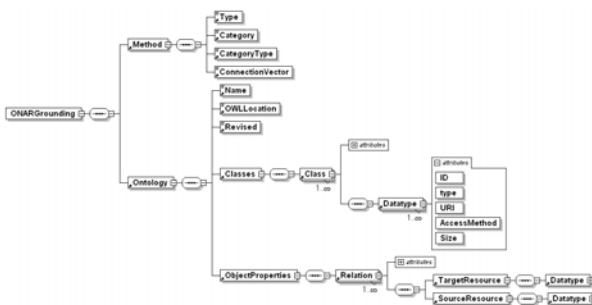


Figure 2: The generic structure of the association ontology.

The association ontology is related to the system structure and associates the concepts of one conceptualization ontology to the system repository. The ontology enables the definition of the:

- *Category of repository*: Category of the repository defines level of the repository (e.g. whether is a flat file system or an RDBMS or an Application server)
- *Type of repository*: defines the technology of the repository (e.g. DCOM application server or JavaBean application server, or Web service)
- *OWL Ontology location*: the conceptualization that the association ontology is related
- *Mapping information*: information that relates every logical concept to the resources of the repository like the access method or technical information like the data type and the size.

With the disassociation of the conceptualization from the repository of the system and the usage of the association ontology we have achieved:

- To increase the *reusability* of the conceptualizations ontologies: The same conceptualization ontologies can be used to different information system.
- To reduce cost of *maintainability* when changes happened in to the application repository.
- To enable the creation of *common vocabularies* that can be shared between heterogeneous applications.

In our work so far we have achieved the application of the ONAR framework to relational databases, ODBC compliant repositories and application server based on web services (WSDL). In the near future we aim to include other type of repositories

III. CREATING META-WEB SERVICES USING CONCEPTUALIZATION

The ONAR Conceptualization cycle including the design, the deployment and registration of the web services to the UDDI registry is related to the conceptualization ontology. The design of the software instance is the process where elements of a conceptualization are used to define the input and the output of the Web service. The design of a web service consists of its syntactic definition and its semantic description.

The syntactic definition defines the input and the output as well as the behaviour of the Web service. This definition is based one conceptualization of the conceptual schema. Starting from one basic entity the Web service designer can use all the concepts that are necessary for his definition. The syntactic definition (ONAR Service Model) of the service contains all the semantic and syntactic relations between the basic entity (primary concept) and the secondary concepts.

The ONAR Service Model (figure 3) is an XML document contains the functional description of the service based on the conceptualization ontology

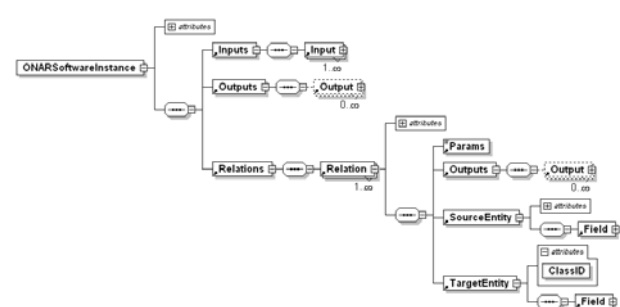


Figure 3: ONAR Service Model

The reason of having a new definition document for the syntactic definition of the service, apart from maintenance purposes, is to increase the portability of the web services. We have implement the necessary functionality that enable us to automatically generate the source code (C#) of a Web service based only to the conceptualization ontology the association ontology and the ONAR Service Model. Therefore if two applications share the same conceptualization ontologies but have different structures (different association ontologies) they can exchange Web services.

IV. CREATING SEMANTIC DESCRIPTION FOR SUPPORTING SOAI

The lack of the Web Service Definition Language (WSDL) to define semantic characteristics of the web services is considered a serious handicap in SOA. The use (or consumption) of a web service requires before the technical implementation and an understanding of its functionality. The capabilities of the WSDL at this level are limited as it was not designed to serve this purpose. Instead other protocols have been produced in order to support Web services discovery and description.

Our work has been based on M.Paolucci attempts [15][16] to integrate a powerful UDDI discovery and categorization mechanisms with the semantics of OWL-S [17]. We have extended UDDI using OWL-S elements in order to include semantic description of the web services. Apart from the ONAR service model that a service designer defines with the definition of the behavior of the web service (Inputs, Outputs and effects); the designer should describe the web service behavior at the semantic level.

This Service Publication Profile (figure 4) defines the semantic characteristics of the web services that are the following:

- **Categorization Information:** Using the UDDI categorization schemas the web service designer defines the categories that apply to the web service.
- **Contact Information:** A list of persons that are associated with the service and their type and responsibilities
- **Custom Parameterization:** Special features of the services that do not belong to a particular categorization schema.



Figure 4: Service Publication Profile

Inside the Service Publication Profile we have also included the elements of the Service Model (Inputs, Outputs and Effects). These elements that are parts of certain conceptualizations carry also semantic descriptions that are also registered to the UDDI registry.

These semantic descriptions of the web service leverage the semantic discovery of the web services reducing the errors of incorrect usage of the web services. This is considered essential for effective SOAI solutions regarding the number of the web services required to support complex integrations and the complexity of each Web service.

V. EXTENDING UDDI REGISTRIES TO SUPPORT SEMANTICS

In order to include semantic meta-information to UDDI registries we have extended both the UDDI and OWL-S profiles. Our work

was based on M.Paolucci work in this domain and also some recommendations from [16]. The ONAR Service Publication Profile is based on OWLS profile syntax with some additions that can be summarized (further information can be found on [5]) as follows:

- Enhancement of the OWL-S input and OWL-S output class in order to define the related classes and relations.
- Enhancement of the OWL-S Service Category and Service Parameter in order to exploit all the UDDI description capabilities
- Additional Models to support OWL-S concepts like Input, Output, Effect, Actor and Service Parameter

This integration between the UDDI and OWL profile leverage the search capabilities that can be based on concepts of conceptualizations. We have developed a search facility that enables the user to choose semantic filters in the level of class properties or relation between classes apart from the typical UDDI inquiry. The search can use both of the techniques lead to more accurate results.

VI. CONCLUSIONS AND FUTURE WORK

The utilization of ontologies for the conceptualization of existing system based on an integration centric approach requires validation and evaluation of our framework. Thus we aim first of all to validate whether ONAR framework can be applied to service oriented mobile services systems as a part of DESYME[22] system. In this direction over this last year we are collecting feedback from the registered user of the framework. Currently we have examined e-Business[1] e-Health[20] and e-Government[19] integration cases.

In addition we aim to create an evaluation framework that will enable us to evaluate ONAR to existing approaches based on the its (a) maintainability (b) complexity (c) portability (d) adaptability in order to assess quantitatively its performance to other integration frameworks.

Finally we are currently working on technical improvements and enrichments of the framework mainly for its adaptability feature in order ONAR to be used from mobile services systems.

VII. REFERENCES

- [1] D. Tektonidis, A. Vontas, U. Hess, J. Meschonat "Handling the Shop-Floor of an Industry through ERP systems – A functional Integration Model", International Conference on E-business & E-Work 2002, Prague 2002
- [2] D. Tektonidis, A. Bokma, G. Oatley & M. Salampasis: "ONAR: An ontologies-based service oriented application integration framework", 1st International Conference on Interoperability of Enterprise Software and Applications, 23-25 February 2005, Geneva, Switzerland
- [3] D. Tektonidis & A. Bokma "The utilization of ontologies for e-Health systems integration", 10th International Symposium on Health Information Management Research, 22-24 September 2005, Thessaloniki, Greece
- [4] M. Themistocleous, Z. Irani, R. O'Keefe & R. Paul "ERP Problems and Application Integration Issues: An Empirical Survey" Proc. of the 34th Hawaiian International Conference on Systems Science, Hawaii Big Island 7-10 January 2001, USA, IEEE 0-7695-0981-9/01
- [5] David Linthicum "Next Generation Application Integration: From Simple Information to Web Services", Addison-Wesley 2003, ISBN: 0201844567, USA
- [6] W3C, Semantic Web Description, 2001, available at <http://www.w3.org/2001/sw/>, as accessed in 10 Nov. 2002

- [7] ONAR, Ontologies based Enterprise Application Integration Framework, 2004, available at <http://research.altec.gr/ONAR> , as accessed in 10 Oct. 2004
- [8] Li min Liu & M. Halper - "Incorporating semantic relationships into an object oriented database system" Proc. of the 32th Hawaiian International Conference on Systems Science, Hawaii Big Island 5-8 January 1999, USA, IEEE 0-7115-1405-9/99 Baldonado, M., Chang, C.-C.K., Gravano, L., Paepcke, A.: The Stanford Digital Library Metadata Architecture. Int. J. Digit. Libr. 1 (1997) 108–121
- [9] Chia-Chu Chiang - "Wrapping legacy systems for use in heterogeneous computing environments", Information and Software Technology, Volume 43, Issue 8, July 2001, Pages 497-507
- [10] O. Corcho, M. Fernández-López and A. Gómez-Pérez "Methodologies, tools and languages for building ontologies. Where is their meeting point?" , Data & Knowledge Engineering, Volume 46, Issue 1, July 2003, pages 41-64
- [11] V. Ratnakar, and Y. Gil, "A Comparison of (Semantic) Mark-up Languages", Proceedings of the 15th International FLAIRS Conference, Special Track on Semantic Web, Pensacola, Finland , May 2002.
- [12] OWL 2005, W3C (2001b) OWL 1.0 Syntax Specification, available at <http://www.w3.org/TR/owl-ref/> , as accessed in 15 May, 2005
- [13] M. Mecella & C. Batini "Cooperation of Heterogeneous Legacy Information Systems: a Methodological Framework" Proceedings of the 4th International Enterprise Distributed Object Computing Conference, Makuhari, Japan, 2000
- [14] Wache, H., Vögele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., & Hübner, S. "Ontology-based Integration of Information - A Survey of Existing Approaches", In: Proceedings of the IJCAI-01 Workshop: Ontologies and Information Sharing, Seattle, USA, pages 108-117.
- [15] M. Paolucci, T. Kawamura, T. R. Payne, K. Sycara. "Semantic Matching of Web Services Capabilities", International Semantic Web Conference (ISWC), 2002, pp 333-347
- [16] M. Paolucci, T. Kawamura, T. R. Payne, and K. Sycara. Importing the Semantic Web in UDDI. In Proceedings of E-Services and the Semantic Web Workshop, 2002
- [17] DARPA, DARPA Agent Markup Language Web Services (DAML-S) Version 0.7 Specification, 2002, available at <http://www.daml.org/services/daml-s/0.7/> , as accessed in 10 Nov. 2002
- [18] Dogac, A., Kabak, Y., Laleci, G., "Enriching ebXML Registries with OWL Ontologies for Efficient Service Discovery", 14th International Workshop on Research Issues on Data Engineering, Boston, USA , March 28-29, 2004.
- [19] Eu-Publi.com IST-Project, "Facilitating Cooperation amongst European Public Administration Employees", available at <http://www.eu-publi.com> , as accessed in 10 Nov. 2005
- [20] Artemis IST-Project, "A Semantic Web Service-based P2P Infrastructure for the Interoperability of Medical Information Systems", available at <http://www.srdc.metu.edu.tr/webpage/projects/artemis/> , as accessed in 10 Nov. 2005
- [21] The Parlay Group, <http://www.parlay.org>
- [22] DESYME (Development System for Mobile Services) project, <http://www.desyme.org>