ODESGS Framework, Knowledge-based Markup for Semantic Grid Services

Carole Goble
Department of Computer Science, University of Manchester, Manchester, U.K.
carole@cs.man.ac.uk

Asunción Gómez-Pérez
Ontology Engineering Group, UPM, Madrid, Spain,
asun@fi.upm.es

Rafael González-Cabero
Ontology Engineering Group, UPM, Madrid, Spain,
rgonza@fi.upm.es

Maria S. Pérez-Hernández
DATSI, UPM
Madrid, Spain,
mperez@fi.upm.es

ABSTRACT

The convergence of the Semantic Web and Grid technologies has resulted in the Semantic Grid. The Semantic Grid should be service-oriented, as the Grid is, so the formal description of Grid Services (GS) turns to be a crucial issue. In this paper we present our approach for this issue. ODESGS Framework will enable the annotation of all the aspects of a GS and the design, discovery and composition Semantic Grid Services (SGS).

Categories and Subject Descriptors

I.2.4 Knowledge Representation Formalisms and Methods

General Terms

Languages, Theory.

Keywords

Knowledge-based markup, Semantic Grid Services, Semantic Web Services, Problem-solving Methods, Ontologies

INTRODUCTION

The ODESGS Framework is an ongoing work carried out in the Ontogrid Project (FP6-511513). It is the extension of the ODESWS Framework [1] developed in the context of the Esperonto Project (IST-2001-34372). It is being developed for the markup of GS and creation of new complex SGS from these annotated GS, to enable their discovery and (semi)automatic composition. ODESGS Framework will also formalize Virtual Organizations (VO), originally defined as a set individuals/institutions defined by a set of resource sharing rules [4]. Now, since the appearance of OGSA [5], VO became defined by the services that they operate and share, due to the wrapping of resources. Therefore, VO description is closely attached to the descriptions made to each GS individually.

In this paper we will enumerate the ODESGS Framework design elements and we will include a detailed description of the set of ontologies that it uses.

K-CAP'05, October 2–5, 2005, Banff, Alberta, Canada.
ACM 1-59593-163-5/05/0010.

ODESGS FRAMEWORK

ODESGS Framework main assumptions are: the use of Problem-Solving Methods (PSM) and ontologies for the description in a formal and explicit way of GS; VO will be defined as the sum of SGS, plus some additional information about the hierarchy of roles of each SGS inside the VO; and some security and provenance related issues. This framework thus should provide a) service and stateful resource ontologies, rich enough to express the semantics required for service discovery and composition in a Grid environment; b) a set of rules to check whether the proposed design (for both complex SGS and VO) is correct; and c) a way to translate from this design into a concrete implementation once the SGS has been designed.

According to all these requirements, the following elements have been identified a) *ODESGS Ontology*, to describe the features of a VO, SGS, Grid resources, etc. a set of ontologies will be used (the ontology will be described in detail later); b) *Instance Model*., designing SGS or VO means to instantiate each of the ontologies of the stack and its relations; c) *Checking model,* once the instance model has been created, it is necessary to guarantee that such model does not present inconsistencies; d) *Design Rules* will be needed to check this, particularly when ontology instances have been created automatically; and e) *Translation Model*, because SGS must be translated into different representational languages to enable programs and external agents to access their capabilities.

ODESGS ONTOLOGY

Our aim is to come up with a service and data ontology, rich enough to express the semantics required for VO and SGS discovery and composition. This means that the VO and SGS features should be explicitly and formally described. With this purpose we propose the use of a stack of ontologies. The stack will be composed of the following ontologies:

KR Ontology and DT Ontology

The Knowledge Representation (KR) Ontology describes the primitives and elements of the KR model used in our descriptions. The KR Ontology is constructed on top of the ontology that describes the types of the attributes, the Data Type (DT) Ontology. It will be based on the XML Schema Datatypes.

SGS Ontology

The SGS ontology presumes that a SGS is decomposed in a set of operations. Each of these operations will be related to their corresponding Choreography, Model and Profile. More precisely a) the *Profile* stores both functional and non-functional properties of the SGS operations (for describing the functional properties, the profile concept establishes *hasTask* relationships with the Task concept of the PSM ontology); b) the *Model,* which defines a relationship *hasMethod* with an element of the concept Method of the PSM Ontology, a service operation will be described by a method that solves or decomposes the task associated with the profile of the operation; and c) the *Choreography* that describes the interaction that should be made to invoke the operation in a formal way, it describes both the messages interchanged and the roles of those that send and receive those messages.

PSM Ontology

In order to decouple the functional features of a service from its internal specification, we propose to apply PSM [2] for modeling SGS. A PSM is defined as a domain-independent and knowledge-level specification of a problem solving behavior [2]. Our ontology for the description of PSM is based on the Unified Problem-solving Method Language (UPML)[3]. Its main elements are:

- *Task.* It describes an abstract domain independent operation to be solved, specifying the input/output parameters and the task competence, which composed of: preconditions and postconditions, assumptions and effects. This description is independent of the method used for solving the task.
- *Method***.** It details the abstract domain independent reasoning process which to achieve a task, describing both the decomposition of the general tasks into sub-tasks and the coordination of those sub-tasks to reach the required result. As UPML does not impose a language for describing the reasoning processes we use a minimal set of programming primitives which allows us to derive several basic workflow-like patterns.
- *Adapter.* It specifies mappings among the knowledge components of a PSM, adapting a task to a method and refining tasks and methods to generate more specific components.
- *Domain Model.* It introduces knowledge about a concrete application.

VO Ontology

VO descriptions will initially be a set of SGS descriptions. But there are still open issues that an additional formalism should solve. More precisely we will decompose a VO description in:

- *Metadata Properties*. Non-functional information about the VO (security and trust information, geographical issues, date of creation, etc.).
- *Roles Models*. We will define roles of SGS in the VO by means of roles taxonomies and a set of restrictions for each role. This tree-shaped structure (or structures) contains the possible roles of the services (or external agents) that may interact or belong to the VO. A set of different restrictions for belonging to a role will be defined for each of them. These restrictions could be on both functional and non-functional properties of the different elements of the stack of ontologies. These restrictions, we may will a) know if a SGS can be added to a certain VO; b) know, in that case that, which of the different roles it may play; and c) use these roles to annotate the actors that appear in each SGS Choreography, relating thus the interaction of a concrete service with the other SGS that compose the VO.
- *Provenance Model*. We will initially follow the ideas carried out in myGrid Project (for a detailed explanation we remit the reader to. Provenance information provides the origin and metadata information with a concrete enactment of a Grid service so as to be able to interpret the results.

ACKNOWLEDGMENTS

This work has been partially financed by the Ontogrid Project (FP6-511513) and by a grant provided by the Community Autonomous of Madrid.

REFERENCES

[1] Gómez-Pérez, A., González-Cabero, R., Lama, M. (2004), A Framework for Design and Composing Semantic Web Services", IEEE Intelligent Systems, vol. 16, pp. 24–32

[2] Motta, E. (1999), Reusable Components for Knowledge Modelling, IOS Press

[3] Fensel D.et al., (2003), The Unified Problem-Solving Method Development Language. Knowledge and Information Systems (KAIS)

[4] Foster I., Kesselman C., Tuecke S. (2001) The anatomy of the Grid. LNCS 2150

[5] Foster I., C. Kesselman, J. N., and Tuecke S., (2002) Grid Services for Distributed System Integration Computer, vol.