

OWL-S based Autonomic Services for Grid Computing

Narendranadh Jabisetti
School of Computing and Engineering
University of Missouri - Kansas City
njn88@umkc.edu

Yugyung Lee
School of Computing and Engineering
University of Missouri - Kansas City
leeyu@umkc.edu

Abstract

Recent advances in the distributed computing infrastructure like Web, Grid, and Pervasive computing environment accompany demands for a more powerful and autonomous service framework. We believe it is feasible to achieve service discovery and matching in an automated manner and perform the service execution not in restricted environments. For the purpose, we developed an autonomous service framework, called ASF, which allows autonomous service to be published, discovered and executed in distributed environments based on the autonomous service specification, extension of the Web Service Ontology (OWL-S) to incorporate physical/logical resources and resource policy.

1. Introduction

Grid computing recently received a lot of attention due to its great potentials for distributed computing. The semantic Grid [5] is to automate most of the services that are defined in scientific computing like high-end computing jobs, storage and retrieval of large datasets, etc. The distribution of resources and the variety of applications that execute on Grid are a suitable candidate for service oriented architecture (e.g., Open Grid Services Architecture and Open Grid Services Infrastructure). The interoperability and incompatibility issues are also the open issues for the Grid services architecture. Tangmunarunkit et al. [5] attempt to create ontology for resources. In [2], the semantic matching capability for Web services based on OWL-S was described. However, the direct use of OWL-S for Grid service markup is not desirable since the principles and requirements of the underlying architectures (i.e., Web and Grid) are different. Moreover, the interaction between user service and its assignment to the underlying Grid services is complicated and inefficient.

In the current Grid infrastructures a user specifies a service name with the associated attributes, and the registry gives access to a matched Grid service file. As the resource

description is extended with rich semantics, there are many resource description languages (e.g., Resource Specification Language, Condor based Class-ads and LDAP based network registries). None of them support the symmetric resource matching and are developed without considering underlying system requirements.

We focus on defining the Grid service requirements so that the service can be discovered automatically and assigned efficiently to the underlying resources. We specify the autonomous service based on the extended model of the Ontology Web Service Ontology (OWL-S). It describes both the services and the resource requirements to make them autonomous. Also, we developed a framework for the autonomous service, called ASF (Autonomous Service framework) that is an infrastructure for diverse services such as Web services, Grid services. The ASF framework defines for the distributed community and virtual organizations, in which Web services and Grid services can be interoperable. The intelligent spaces can be part of virtual organizations or can be connected to the Web community.

2 Autonomous Service Framework

We define autonomous service as a computational entity that includes the following elements in its description: the service name and type, the service physical and logical resource requirements, the service policy to the OWL-S¹ (the service grounding, the service inputs, outputs, preconditions and effects, the service process model). Service quantification describes the required CPU power, physical memory, network bandwidth for a fixed input size/type.

The physical resource model describes the required resources for the service to execute in its original or leased environment. Physical resources we mean individual or aggregated computing and storage systems. Typically, a service can be executed on a single processor system or a multi processor system. The specified computing resource abstraction should clearly specify the compatibility of the service

¹OWL-S Website: <http://www.daml.org/services/daml-s/0.9/>

with the specified computing systems architecture. A service with strict sequentially executable subprocess modules can only be executed on a single processor system with required memory and processor types. Some services can be executed on a multi-processor system, since the sub modules are not strictly sequentially executed. These services can tolerate a small delay in subsequent module executions, however these small delays cannot be exploited for multi computer executions of these services. In each of these architectures, the only requirements for autonomous service description are the name of computing system, processor type, required physical and logical memory, installed operating system and operating system version.

In many of the Grid application, the service selection and performance depend on the software programs and tools, data repositories and algorithms used in the service execution [3]. The end users can define a policy for sharing their resources. The policy defines which resources are shared and how they are shared. The policy also allows users to set preferences into particular communities. The policy would be used while searching Grid application to run on the host computer. For example, a user may specify that s/he wishes to share only 50% of his/her computer memory.

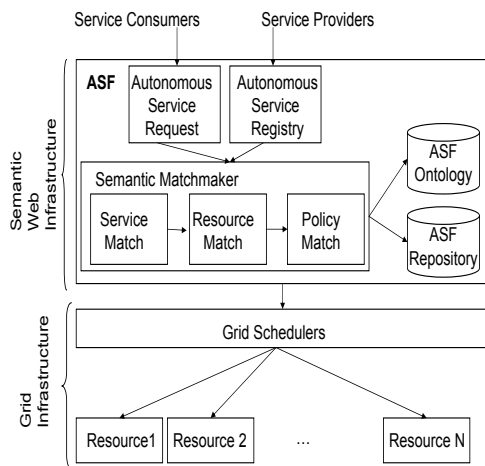


Figure 1. ASF Infrastructure

In our ASF framework the autonomic service registration is based on a semantic approach to overcome the limitations of the LDAP and UDDI registries. We also adopted the TeraGrid approach² to represent service capability in the registry. So, the service selection mechanism in the ASF framework discovers the appropriate service depending on the service requirements and resource capabilities.

In the ASF framework, the service matchmaking entirely depends on the user abstraction the framework provides. For the high level of abstraction where the service name, inputs and expected outputs are specified, the framework

should take necessary steps to find an appropriate service for a given user's task. The whole purpose of service autonomy is to provide as high level of user abstraction as possible. In many cases the user want to find a service using a specific kind of data repository or algorithm. For such request, the matchmaking needs semantics (i.e., domain ontologies), rather than syntactic information.

In the ASF framework, we link the services to their required resources by specifying resource requirements specification with the service specification. During the resource matching, the service provider's resource description is matched with the resource provider's resource description. Our service discovery is semantic based (i.e., subsumption reasoning) rather than syntactic. Then what we mean semantic matching and how semantic matching supports autonomy of Grid services. Semantic matching of Web services is well known phenomenon and has variety of approaches. [4] uses service input and output matching (i.e. it uses subsumption reasoning to match the user service input/output requirements with the potential service input/output requirements). In the matching process the output matching plays a major role than the input matching. Other approaches use classification of services based on some domain ontology and uses concept matching [1]. We believe that our solution is expected to meet the requirements of extended matching capabilities for the extended resources.

References

- [1] Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, Katia P. Sycara, Importing the Semantic Web in UDDI. *Web Services, E-Business, and the Semantic Web 2002*, 225-236
- [2] Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, Katia P. Sycara, Semantic Matching of Web Services Capabilities. *International Semantic Web Conference 2002*: 333-347
- [3] D. de Roure, M. A. Baker, N. R. Jennings, and N. R. Shadbolt. The evolution of the Grid. In F. Berman, G. Fox, and T. Hey, editors, *Grid Computing: Making the Global Infrastructure a Reality*, pages 65–100. Wiley & Sons, 2003.
- [4] Katia P. Sycara, Massimo Paolucci, Anupriya Ankolekar, Naveen Srinivasan, Automated discovery, interaction and composition of Semantic Web services. *Journal of Web Semantics* 1 (1): 27-46 (2003)
- [5] Hongsuda Tangmunarunkit, Stefan Decker, Carl Kesselman, Ontology-Based Resource Matching in the Grid - The Grid Meets the Semantic Web. *International Semantic Web Conference 2003*: 706-721

²<http://www.teragrid.org/>