

A Grid Service Discovery Matchmaker based on Ontology Description

Simone A. Ludwig
Department of Electronic and Computer Engineering, Brunel University
Uxbridge, Middlesex, UK

Peter van Santen
Department of Electronic and Computer Engineering, Brunel University
Uxbridge, Middlesex, UK

An ontology defines a common vocabulary that facilitates the sharing of information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and their relations. In the Grid environment (GE) where so many different implementations are available, the need for semantic matching based on a defined ontology becomes increasingly important. Especially for service discovery (SD), an ontology description is very useful in supporting a customised SD process. This paper proposes a SDmatchmaking framework for GEs based on a well-defined ontology. The matching of service requests is discussed and a matchmaking mechanism is described in detail. The proposed framework allows for close and customised SD matches, that current service discovery systems (SDSs) cannot provide as their SD process is based on type matching only.

1. INTRODUCTION

Locating a network service or a device on demand is a challenging task for pervasive computing. A variety of service discovery systems (SDSs) exist that enable an application to discover a service. Most of these systems support an attribute-based discovery as well as a simple name lookup, also called matchmaking, to locate a service. Usually only a set of primitive attribute types, such as string and integer, are used to characterise a service. The service discovery (SD) process is therefore primarily done by type matching, based on string or integer comparison. All existing SDSs lack the ability of inexact matching [1]. This is especially the case in Grid Environments (GEs), where so many different implementations of services are available that might vary in name and functionality. The need for semantic matching becomes increasingly important as more and more services are developed.

Grid computing has been receiving a lot of attention in recent years. A GE typically comprises heterogeneous resources over a wide-area network. These resources and services are available for Grid applications and therefore, Grid SD is an important issue and presently, there is effort underway to standardise services. The Open Grid Services Architecture (OGSA) [2] was created to define a base framework of services to achieve interoperability between different Grid implementations.

Though, OGSA provides interoperability and defines discovery of services between different Grid domains, semantic matching based on ontologies will provide a much better SD. The term ontology is useful in the Grid Service (GS) context, where it describes the need for the provider of a service and the user of that service to share a common understanding of what capabilities the service offers and how they can be put to use. This is described in more detail in the next section.

This paper is organised as follows. Section 2 gives an introduction to ontology and its development, section 3 describes the SD framework including the matchmaking mechanism, and finally, section 4 presents our conclusions.

2. BACKGROUND TO ONTOLOGY

When two or more parties seek a common understanding of something, they must work together to ensure that there is a high degree of correlation and similarity between the details of their respective descriptions and definitions of what they are trying to agree on [3]. This implies that shared understanding requires shared definitions. Particularly, for GSs where SD is a significant issue, the need to share a common ontology becomes very important.

A prerequisite for widespread use of ontologies is a joint standard for their description and exchange. RDF(S) (Resource Description Framework Schema) itself is an ontology/knowledge representation language that contains classes and properties (binary relations), range and domain constraints (on properties) and subclass and subproperty (subsumption) relations. RDF(S) is a relatively primitive language, however, more expressive power would clearly be necessary and desirable to describe resources in sufficient detail [4]. These considerations led to the development of the Ontology Inference Layer (OIL) [5] and later to the design of DAML+OIL [6]. DAML+OIL is a more recent

proposal for an ontology representation language that has emerged from work under DARPA's Agent Markup Language (DAML) initiative along with input from leading members of the OIL consortium. DAML+OIL is based on the original OIL language, but differs in a number of ways. As part of the DARPA Agent Markup Language program an ontology of services, called DAML-S [7], has been developed, which provides a set of basic concepts and relations for declaring and describing services, by utilising the ontology structuring mechanisms provided by DAML. The DAML-S service is characterised in three types which are service profile, service model and service grounding [8].

3. SERVICE DISCOVERY FRAMEWORK

The resources in the Grid are heterogeneous and geographically distributed with varying availability and variety of usage and cost policies for diverse users at different times. The management of resources and services in such a large distributed environment is a complex task. Therefore, SD plays an important role in such environments.

The following is a description of a proposed SD framework for GEs. The framework relies on an ontology description that allows semantic matching, and it is based on the LARKS matchmaker proposed by Sycara et al. [9]. The matching mechanism comprises of three filter stages. These are context, syntactic and semantic matching, whereas the service ontology database provides the knowledge-base. The difference between the LARKS and the Grid SD matchmaker is that the Grid SD relies on DAML-S and its ontologies for the matchmaking, while LARKS only uses the set of filters that progressively restrict the number of advertisements which are candidates for a match. The Grid SD achieves semantic matching as it relies on the set of filters and also allows the flexibility of close matches by providing an ontology knowledge-base. A description of the matching process in the SD matching framework [10] is described below and is followed by a description of the framework.

3.1 Matching Mechanism

An advertisement matches a request, when the advertisement describes a service that is sufficiently similar to the service requested [11]. The problem of this definition is to specify what "sufficiently similar" means. Basically, it means that an advertisement and a request are "sufficiently similar" when they describe exactly the same service. This definition is too restrictive, because providers and requesters have no prior agreement on how a service is represented and additionally, they have very different objectives.

It is necessary to allow matching engines to perform flexible matches, those that recognise the degree of similarity between advertisements and requests in order to provide a softer definition of "sufficiently similar". Service requesters should be allowed to decide the degree of flexibility that they grant to the system. If they allow little flexibility, they reduce the likelihood of finding services that match their requirements, which means they minimise the false positives while increasing the false negatives. On the other hand, by increasing the flexibility of a match, they achieve the opposite effect, that is, they reduce the false negatives at the expense of an increase of false positives.

Considering all these criteria, the matching engine should satisfy the following [11]:

- a) The matching engine should support flexible semantic matching between advertisements and requests based on the ontology available to the service and the matching engine.
- b) Despite the flexibility of the match, the matching engine should minimise false positives and false negatives.
- c) The matching engine should encourage providers and requesters to be precise with their descriptions at the cost of either not being matched or being matched inappropriately.
- d) The matching process should be efficient which means that it should not burden the requester with excessive delays that would prevent its effectiveness.

The algorithm proposed aims to satisfy all four requirements. Semantic matching is based on DAML ontologies. The advertisements and requests refer to DAML concepts and the associated semantic. By using DAML, the matching process can perform implications on the subsumption hierarchy leading to the recognition of semantic matches despite their syntactical differences.

3.2 Service Discovery Matchmaker

Three components are necessary for the SD matchmaking. These are Grid service provider, service requester and service matchmaker. The sequence of interactions is as follows:

- 1) The service provider (GS) registers its service description into the service registry database.
- 2) The Grid application requests a GS and sends the request to the SD matchmaker.
- 3) The matchmaker returns the matches to the service requester.

4) The service requester decides then which service to use depending on the client's need.

GSs are services that are provided in a GE. These services are authentication and authorisation, job submission, distributed job scheduling, resource optimisation, data management, wide-area data transfer, file replication, resource management and resource monitoring.

The service requester consumes services offered by the GS providers in the system. A request for any GS has to be sent to the matchmaker. In a GE, service requesters are the applications which use the GSs. These are, for example, the LHC-HEP (Large Hadron Collider - High Energy Physics) experiments such as Alice [12], Atlas [13], CMS [14] and CDF/D0 [15]. These LHC-HEP experiments want to access the GSs in order to process their petabytes of data necessary for their experimental evaluations.

The service matchmaker mediates between service requesters and service providers for some mutually beneficial cooperation. Each provider must first register with a registry, also called matchmaker. Service provider advertises their capabilities (advertisements) by sending some appropriate messages describing the kind of service they offer. Every request a matchmaker receives will be matched with his actual set of advertisements. If the match is successful, the matchmaker returns a ranked set of appropriate service provider and the relevant advertisements to the requester.

The matchmaker process a received request in the following three basic steps:

- a) Comparison of the request with all advertisements in the service registry database.
- b) Decision of the service provider whose capabilities match best with the request depending on the specified algorithm and the defined service ontology. Every pair of request and advertisement has to go through several different filters during the matchmaking process.
- c) Providing information to the service requester by sending a contact address and related capability descriptions of the relevant service provider.

The architecture fulfils the matching criteria listed in section 3.1 as follows. The Grid SD matchmaker supports flexible semantic matching between advertisements and requests based on the ontology available. Minimising false positives and false negatives is achieved with three filter stages in combination with the well-defined ontology. The filter stages restrict the false positives and the ontology restricts the false negatives. The matching engine should encourage providers and requesters to be precise with their descriptions. To achieve this, the service provider follows an XML-based description to advertise its services and the service requester generates a query in a specified format. Defining the ontology and the filter stages precisely allows the matching process to be efficient. This is the next step for further research which needs to be investigated and defined in detail.

3.3 Component Description of the Matchmaker

When a Service Requester requests a service the request goes through the parser. The parser breaks data into smaller elements, according to a set of rules that describe its structure.

The matching process of the matchmaker is designed with respect to the criteria listed in section 3.1. Matching a given request into a set of advertisements consists of the following five filters that are organised in three consecutive filtering stages:

- 1) Context Matching: Selects those advertisements in the auxiliary database that can be compared with the request in the same or similar context.
- 2) Syntactic Matching: This filter compares the request with any advertisement selected by the context matching in three steps that are comparison of profiles, similarity matching and signature matching. The request and advertisement profile comparison uses a weighted keyword representation for the specifications and a given term frequency based similarity measure. The service ontology provides the service model and the service grounding. The last two steps focus on the input/output constraints and declaration parts of the specification.
- 3) Semantic Matching: This final filter checks if the input/output constraints of any pair of request and advertisement logically match.

For reasons of efficiency, the context filter roughly prunes off advertisements that are not relevant for a given request. In the following two filtering stages (syntactic and semantic matching) the remaining advertisements in the auxiliary database of the matchmaker are checked in more detail. All filters are independent from each other and each of the filters narrows the set of matching candidates with respect to a given filter criteria.

The service registry database contains all GSs in form of the DAML-S service profile. The auxiliary database comprises a database for word pairs and word distances, basic type hierarchy and internal data. The service ontology database contains the service model and service grounding of the DAML-S service ontology. The service registry database is linked to the service ontology database and both contain the DAML-S service ontology. The main benefits of providing a local ontology are as follows.

The user can specify in more detail what is being requested or advertised, and particularly the matchmaker is able to make automated inferences on these additional semantic descriptions. This improves the overall quality of the matching process.

In this section, the SD matchmaking framework based on ontology knowledge was described. It revealed how the matchmaking process is done and how close and customised matching of services is achieved.

4 CONCLUSION

In the GE, where so many different implementations are available, the need for semantic matching based on a defined ontology becomes increasingly important in order to provide close and customised service request matches. This paper proposed a new semantic SD framework for GEs. It proposes a new matchmaking mechanism based on ontology knowledge. Using this matchmaking framework allows for a better SD and close matches in a flexible way based on the defined ontology. With this framework, Grid applications are able to specify the criteria a service request should be matched with.

ACKNOWLEDGMENT

Part of this research is funded by the IST Program of the European Union (Grant IST-2001-32459). The authors would like to thank Dr. M. Reyhani for his helpful discussions and useful suggestions.

REFERENCES

- [1] Ludwig, S.A., "Review of Service Discovery Systems". Technical Report, TR-DGRG695, Department of Electrical and Computer Engineering, Brunel University, UK, 2002.
- [2] Foster, I., Kesselman, C., Nick, J.M. and Tuecke, S. "The Physiology of the Grid", Computer, 35(6), 2002.
- [3] W3C Working Draft, "Requirements for a Web Ontology Language".
<http://www.w3.org/TR/webont-req/>.
- [4] Bechhofer, S. and Goble, C., "Towards Annotation using DAML+OIL". K-CAP 2001 Workshop on Knowledge Markup and Semantic Annotation, Victoria B.C., October 2001.
- [5] Fensel, D., Horrocks, I., Van Harmelen, F., Decker, S., Erdmann, M., and Klein M., "OIL in a nutshell". In Proceedings of EKAW-2000, LN AI, 2000.
- [6] DAML. Darpa Agent Markup Language Program.
<http://www.daml.org>.
- [7] Ankolekar, A., Burstein, M., Hobbs, J.R., Lassila, O., Martin, D.L., McIlraith, S.A., Narayanan, S., Paolucci, M., Payne, T.R., Sycara, K., and Zeng, H., "DAML-S: Semantic Markup for Web Services". International Semantic Web Workshop (SWWS), 2001.
- [8] Payne, T.R., Paolucci, M. and Sycara, K., "Advertising and Matching DAML-S Service Descriptions". Semantic Web Working Symposium (SWWS), 2001.
- [9] Sycara, K., Klusch, M., Widoff, S., Lu, J., "Dynamic Service Matchmaking Among Agents in Open Information Environments". Journal ACM SIGMOD Record, 1999.
- [10] Ludwig, S.A., "An Ontology-based Service Discovery Matchmaking Framework". Technical Report, TR-DGRG697, submitted for publication to FGCS, Department of Electrical and Computer Engineering, Brunel University, UK, 2002.
- [11] Paolucci, M., Kawamura, T., Payne, T.R., Sycara, K., "Semantic Matching of Web Services Capabilities". Proceedings International Semantic Web Conference (ISWC 02), 2002.
- [12] ALICE.
<http://alice.web.cern.ch/Alice/>.
- [13] ATLAS.
<http://atlasinfo.cern.ch/Atlas/Welcome.html>.
- [14] CMS.
<http://cmsdoc.cern.ch/cms/outreach/html/index.shtml>.
- [15] CDF/D0.
<http://www-cdf.fnal.gov/>.