A Unified Approach for the Discovery of Web and Peer-to-Peer Services

M. Pantazoglou, A. Tsalgatidou, G. Athanasopoulos, T. Pilioura
Dept. of Informatics & Telecom. University of Athens (NKUA) Athens 15784, Greece
{michaelp, atsalga, gathanas, thomi} @ di.uoa.gr

Abstract

As the web service technology matures, other computing paradigms such as peer-to-peer gradually adopt the service-oriented approach and are beginning to expose functionality as services. Hence there will soon be a need for integration of these heterogeneous services, for the development of service-oriented applications. The first step to accomplish this is to establish a unified approach for service discovery. In this paper, we briefly present a query language along with its enacting service search engine which is used for effectively discovering web and p2p services in a unified manner.

1. Introduction

Web services are the prevailing instantiation of the *Service-Oriented Computing (SOC)* paradigm. Yet, other computing technologies such as peer-to-peer (p2p) are gradually shifting to the service-oriented approach by exposing functionality as services [1]. It is in our belief that it would be beneficial if all these diverse technologies were integrated and/or combined in building service-oriented applications. In this context, software developers will inevitably call for the appropriate languages and tools to leverage this integration. In this short paper we briefly present a service discovery framework comprising a service query language and its enacting search engine, which provides for transparently accessing and querying heterogeneous registries and networks and for applying matchmaking against heterogeneous service descriptions in a unified manner.

2. The Unified Service Query Language

The *Unified Service Query Language* (USQL) is an XML-based language that enables requesters to formulate their requirements for discovering web and p2p services in a unified manner, while at the same time it keeps technical details transparent. USQL defines two types of messages, namely the *USQLRequest* and *USQLResponse*. To better capture real-world requirements, USQL blends the flavors of syntactic, semantic and Quality-of-Service (QoS) search criteria which can be applied at the service, operation, and message levels. Moreover, the language specifies a well defined set of operators, which can be applied to the search criteria explicitly, and thus affect the conduction and results of the matchmaking process. This departure is rendered particularly useful when applying service discovery at design time, where requirements should be expressed in a more relaxed fashion. More details regarding the language specification can be found in [3].

3. The Unified Service Search Engine

The *Unified Service Search Engine* applies service discovery over heterogeneous registries and networks with the use of the USQL language. It is characterized by an extensible architecture that enables the smooth accommodation of heterogeneous registry and service description standards. More specifically, registry plug-ins are used to provide access to the various service registries and p2p networks (e.g. UDDI [1] and JXTA [2]), while appropriate handler components are employed to deal with the various syntactic, semantic and quality-of-service descriptions that the registries and p2p networks advertise. The overall service discovery process along with the involved internal components of the search engine are depicted in Figure 1.

Upon receiving a USQL request, the engine employs the *USQL Handler*. The functionally of this component is divided into three logical parts: 1) the *Validator*, responsible for the validation of USQL messages; 2) the *Request Processor*, responsible for processing the content of USQL request messages; and 3) the *Response Processor*, responsible for constructing and properly formatting the USQL response messages. The USQL Handler contributes significantly to the overall flexibility and maintainability of the engine; it abstracts the rest of the components from language-specific details, thus making them resilient to potential changes in the USQL specification.

After the USQL request has been found to be valid, the request processor is activated to extract the service domain value from the message. The specified domain is used by the *Registry Selector* component, in order to identify the appropriate registries and/or networks to forward the query. This identification is realized through a lookup to an upper ontology maintained by the engine, which associates registries with application domains.

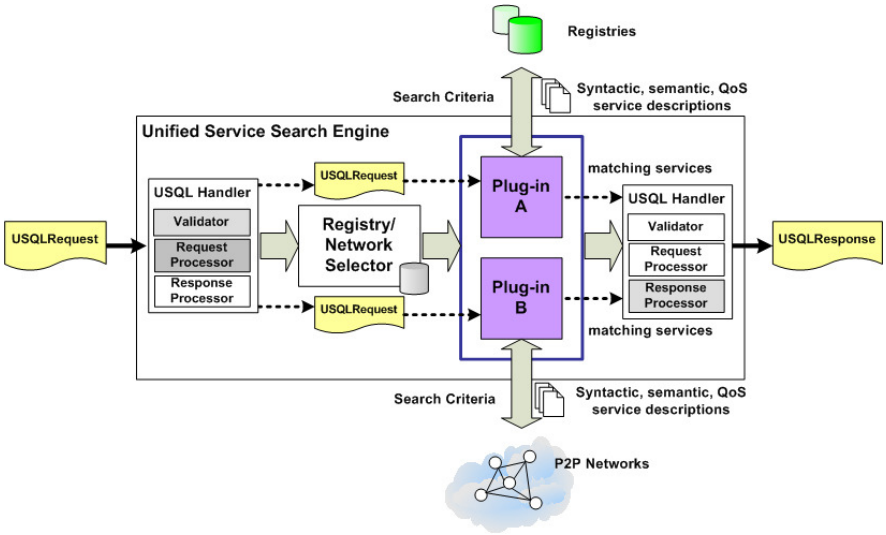

Figure 1. Internal structure and functionality of the search engine

To access the selected registries and p2p networks, the engine configures and instantiates the respective *plug-ins*. The employed plug-ins accept the USQL request as input and run in separate threads to improve the engine's performance. Each registry plug-in retrieves a set of advertisements which are checked against the requirements in the USQL request. The matching services constitute the output of the plug-in. Finally, the USQL Handler employs the response processor to consolidate the output from all plug-ins and formulate the corresponding USQL response message. The content of the USQL response can be used for the immediate invocation of the contained services. Hence, heterogeneous service composition middleware such as JOpera [4] could be extended to exploit the output of the search engine and execute compositions of heterogeneous web and p2p services.

4. Discussion and Future Work

Many efforts have targeted the area of software service discovery over the last few years and a number of novel service search engines have been proposed [6], [7]. Besides, the p2p architecture has been leveraged in a number of cases to enhance the various web service activities [5]. Yet, to the best of our knowledge, there is no approach other than the one presented in this paper, in terms of unified web and p2p service discovery.

Both the USQL language and its enacting engine presented in this paper are evolving. So far, we have developed plug-ins to support service discovery in UDDI and JXTA. Our future plans include the development of plug-ins that will enable service discovery against other registry and p2p network types (e.g. ebXML [8], P2PS [9], etc).

Acknowledgement. This work is partially supported by the Special Account of Research Funds of the National and Kapodistrian University of Athens under contract 70/4/5829 and by the European Commission under contract IST-FP6-004559 for the SODIUM project [10].

References

[1] UDDI Specifications, http://www.uddi.org
[2] JXTA Project, http://www.jxta.org
[3] A. Tsalgatidou, M. Pantazoglou, G. Athanasopoulos: *Specification of the Unified Service Query Language (USQL)*, Technical Report (2006), available at http://cgi.di.uoa.gr/~michaelp/TR/20060515-TR-usql-1.0-spec.pdf
[4] Cesare Pautasso, Gustavo Alonso: *From Web Service Composition to Megaprogramming*, In Proc. of the 5th VLDB Workshop on Technologies for E-Services (TES-04), Toronto, Canada, August 29-30, 2004.
[5] Verma, K., Sivashanmugam, et al: *METEOR-S WSDI: A Scalable Infrastructure of Registries for Semantic Publication and Discovery of Web Services*, Journal of Information Technology and Management (2005) Vol. 6 (1) pp 17-39
[6] Xin Dong et al: *Similarity Search for Web Services*, Procs of VLDB 2004, Canada
[7] Cibran, M. A., Verheecke, B. et al: *Automatic Service Discovery and Integration using Semantic Descriptions in the Web Services Management Layer*, In Proceedings of Third Nordic Conference on Web Services, Vaxjo, Sweden, November 2004.
[8] ebXML, Electronic Business using eXtensible Markup Language, http://www.ebxml.org
[9] Peer-to-Peer Simplified, http://www.trianacode.org/p2ps/
[10] SODIUM Project, http://www.atc.gr/sodium