Service Integration through Structure-Preserving Semantic Matching

Fiona McNeill, University of Edinburgh, Scotland

Paolo Besana, University of Edinburgh, Scotland

Juan Pane, University of Trento, Italy

Fausto Giunchiglia, University of Trento, Italy

EXECUTIVE SUMMARY

The problem of integrating services is becoming increasingly pressing. In large, open environments such as the Semantic Web, huge numbers of services are developed by vast numbers of different users. Imposing strict semantics standards in such an environment is useless; fully predicting in advance which services one will interact with is not always possible as services may be temporarily or permanently unreachable, may be updated or may be superseded by better services. In some situations, characterised by unpredictability, such as the emergency response scenario described in this case, the best solution is to enable decisions about which services to interact with to be made on-the-fly. We propose a method of doing this using matching techniques to map the anticipated call to the input that the service is actually expecting. To be practical, this must be done during run-time. In this case, we present our structure-preserving semantic matching algorithm (SPSM), which performs this matching task both for perfect and approximate matches between calls. In addition, we introduce the OpenKnowledge system for service interaction which, using the SPSM algorithm, along with many other features, facilitates on-the-fly interaction between services in an arbitrarily large network without any global agreements or pre-run-time knowledge of who to interact with or how interactions will proceed. We provide a preliminary evaluation of the SPSM algorithm within the OpenKnowledge framework.

Keywords: Ontologies, Semantic Matching, Service Integration, Structural Matching, Web Services

BACKGROUND

The problem of automated integration of services is key to the successful realisation of the Semantic Web, or any other system where services interact with one another. So far, this has proved difficult. Global ontologies allow different services to be expressed using the same terms,

which are thus understandable to all. But there are significant difficulties with the notion of a global ontology: both the relevance of terms and appropriate categorisation of those terms is very context dependent. An ontology that included all terms that could be relevant to any situation would be impossible to build, impossible to reason with and would allow no flexibility for different interpretations of situations.

However, integration of services using different ontologies is difficult. The difficulties arise at two levels: in the structure of the invocation to the service and in the values passed with the invocation. A service will expect some input parameters and will return an output. Consider for example, the web service measurement, whose WSDL description is shown in Figure 1. Its purpose it to provide the level of water registered by a particular sensor on a grid of sensors on a particular river-side area, which can be used during an emergency to assess the conditions. It expects as the input message the location, defined as the node identifier in the grid, and the id of the sensor, and returns in the output message the measured water level and the timestamp of the measurement.

The structure, or signature, provided by input parameters and output values must be respected by a process invoking the service. However, the invoking process may have a different signature for the caller function (parameters may have different names or

they may have a different structure). For example, a caller process could be a BPEL workflow, originally developed to invoke a service called reading, that does not have the concept of location, but only of the reporter and node identities, and expects the level to be named differently. The invocation needs to be adapted to the new called service.

Even after the structural adaptation has been performed, the terminology used in the parameters may be defined in different ontologies in the caller process and in the service. This may cause misunderstandings or failure: for example, the water level may be returned in meters when the caller expected feet. Translation is required.

This case focuses on the problem of matching the signature of service invocation with that of the service when they are expressed in different ontologies. It is perfectly possible to solve this problem by manually matching the expected inputs and outputs of two services – the one that the caller expected and the one that is actually called – prior to interaction. For example, Altova MapForce[1] is a system which facilitates this manual mapping. However, performing this is time consuming and not scalable. Additionally, this presupposes that one knows in advance what calls will be necessary. This is perhaps feasible in a small, static system, but in a large, dynamic system where services may be temporary, may be updated, may suffer from occasional communication breakdown, and so on, we do not wish to limit the number of services with which it is possible to interact. A better solution in this sort of environment is to automatically integrate services on-the-fly as the need becomes apparent.

Using our matching approach we are able to map between the invocation of the service, written in the ontology of the caller, and the call the service is expecting, written in the ontology of the service. The goal of the matching is two-fold:

- to establish whether these services (the expected and the called) are *similar enough*: if the service is being asked to perform an action that is too different to the function it is equipped to perform, then the correct response is to refuse to interact;
- if the call is judged to be similar enough, then an adaptor is generated to bridge beween the caller and the invoked service .

19 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/teaching-case/service-integration-through-structure-preserving/37362

Related Content

Main Component and Architecture of the Semantic-Based Process Mining and Analysis Framework (SPMaAF)
(2020). *Applications and Developments in Semantic Process Mining (pp. 111-120).*
www.irma-international.org/chapter/main-component-and-architecture-of-the-semantic-based-process-mining-and-analysis-framework-spmaaf/253006

Applying an Organizational Uncertainty Principle: Semantic Web-Based Metrics
Joseph Wood, Hui-Lien Tung, Tina Marshall-Bradley, Donald A. Sofge, James Grayson, Margo Bergmanand W.F. Lawless (2009). *Handbook of Research on Social Dimensions of Semantic Technologies and Web Services (pp. 469-488).*
www.irma-international.org/chapter/applying-organizational-uncertainty-principle/35743

Information Retrieval by Semantic Similarity
Angelos Hliaoutakis, Giannis Varelas, Epimenidis Voutsakis, Euripides G.M. Petrakisand Evangelos Milios (2006). *International Journal on Semantic Web and Information Systems (pp. 55-73).*
www.irma-international.org/article/information-retrieval-semantic-similarity/2824

A Model for Predicting Physical Health of College Students Based on Semantic Web and Deep Learning Under Cloud Edge Collaborative Architecture
Yu Wang, Zhiyi Zhang, Peng Tangand Shiyao Bian (2024). *International Journal on Semantic Web and Information Systems (pp. 1-19).*
www.irma-international.org/article/a-model-for-predicting-physical-health-of-college-students-based-on-semantic-web-and-deep-learning-under-cloud-edge-collaborative-architecture/340379

Quantifying the Connectivity of a Semantic Warehouse and Understanding its Evolution over Time

Michalis Mountantonakis, Nikos Minadakis, Yannis Marketakis, Pavlos Fafaliosand Yannis Tzitzikas (2016). *International Journal on Semantic Web and Information Systems (pp. 27-78).*