

Connecting OWL 2-aware Components via OWLlink



Tutorial at ISWC 2009

INTRODUCTION

A real-world semantic application typically is a heterogeneous system. Its distributed architecture is often implied by requirements such as reliability and exchangeability. Building an application out of OWL-aware components requires a flexible but standardized interface. OWLlink (www.owllink.org), the successor of the well-known DIG protocol, is an implementation-neutral communication interface for OWL. It relies on OWL 2 and is likewise defined by a structural specification. Accompanying bindings specify how messages can be encoded and exchanged by a transport protocols. A key feature of OWLlink is its extensibility which allows to add functionality required by specific applications to the core protocol.

TUTORIAL OVERVIEW

This half day tutorial at ISWC 2009 provides a practical introduction to OWLlink from the perspective of an application developer. It introduces the basic principles underlying OWLlink and explains how to manage reasoning engines, assert axioms, and to query inference results. Then two extensions are presented, one for retrieving previously told axioms and one for retracting axioms from the reasoner. Furthermore, an overview about existing OWLlink implementations is given and it is demonstrated how to employ the protocol on the code level using available OWLlink components. Finally, the tutorial describes how to specify your own OWLlink extension.

TARGET AUDIENCE AND PREREQUISITE KNOWLEDGE

This tutorial aims at people interested in designing as well as building semantic applications based on OWL. The tutorial is comprehensible even for participants with a basic understanding of OWL. For the code level demonstration part of the tutorial some programming experiences would be helpful.

PLACE

ISWC Conference at Westfields Conference Center (near Washington, DC).

DATE

Sunday, Oct. 25th 2009
Afternoon

CALL

This tutorial is open to all members of the ISWC community. Note that participation requires to register for the ISWC conference as well as to pay the tutorial fee.

ORGANIZATION

Thorsten Liebig
Ulm University
Ulm, Germany

Marko Luther
DOCOMO Eurolabs
Munich, Germany

Olaf Noppens
Ulm University
Ulm, Germany

owllink.org/tutorial09

TIMELINE

The tutorial has the following outline divided into two parts (90 min each):

1. **Introduction**
General motivation using typical application scenarios to identify deficits of related interfaces. Description of the aims of the protocol. Brief comparative analysis with respect to other APIs and a quick look at its development history.
2. **OWLink Core**
Presentation of the three basic blocks of the structural specification, namely resource management, axiom assertions, and basic queries.
3. **OWLink Bindings**
Introduction of the standard binding (XML over HTTP). Description of other bindings that have been defined as well as a brief look at potential further ones.
4. **OWLink Extensions (I)**
Description of the extension mechanism and a detailed introduction of two extensions, namely Retraction (of previously asserted axioms) and Told Data Access.
Break
5. **OWLink Extensions (II)**
Overview on further extensions and extension proposals. Practical explanation on how to define your own extension with the help of a simple example.
6. **OWLink Deployment**
Survey of available OWLink-aware components. Code level demonstration of the OWLink part for two simple client applications (at least one of them in Java).
7. **Conclusion**
Wrap-up, outlook and discussion of issues related to OWLink.

OBJECTIVE

The tutorial aims at describing the concepts and employment of the OWLink protocol for building semantic application out of OWL2-aware components. The goal of this tutorial is to enable participants to design and implement modular applications which utilize OWL reasoning services via OWLink as well as to create their own desired OWLink extension.

SCRIPT

A script of the presentation slides as hard-copy will be provided to participants at the tutorial. We also plan to make this script available for download after the tutorial.

CODE

The source of all code level examples will be provided at tutorial time. This includes two simple OWLink client applications (at least one of them will be in Java).

SYSTEMS

The code level demonstration makes use of systems or software libraries with OWLink support. We will provide links to these components at this place at a later point.

LINKS

All OWLink specification documents can be found at:

<http://www.owllink.org/>

A recent publication introduces OWLink:

T. Liebig, M. Luther, O. Noppens, M. Rodrigues, D. Calvanese, M. Wessel, R. Möller, M. Horridge, S. Bechhofer, D. Tsarkov, E. Sirin. **OWLink: DIG for OWL 2**. Fifth International Workshop on OWL: Experiences and Directions (OWLED08), Karlsruhe, Germany, October 2008.