WSMX Deliverable

D13.4 v0.2
WSMX Architecture

WSMX Working Draft – 21st January 2005

Authors:
Michal Zaremba and Matthew Moran

Editors:
Michal Zaremba

This version:
http://www.wsmo.org/2004/d13/d13.4/v0.2/20050118
Latest version:
http://www.wsmo.org/2004/d13/d13.4/
Previous version:
http://www.wsmo.org/2004/d13/d13.4/v0.1/

Abstract

This deliverable describes Web Services Execution Environment Architecture - WSMX Architecture. The document provides both a high-level overview of the necessary system components and interactions between them (conceptual architecture) as well as a low-level definition of components interfaces, connectivity, applied events mechanism, etc. Through this deliverable we want to present the vision and provide the conceptual model of a distributed systems, where semantically described components can be plugged in and executed following dynamic execution semantics. While we assume that for the Semantic Web Services (SWS) infrastructure, we need a set of components such as for example discovery, mediator, invoker etc, we would like our system to be capable to host any additional components, which functionality is unknown to designers of WSMX system. Using analogy to Eclipse system, WSMX architecture can be used to plug any components without changing and recompiling the core engine of the system. These components can be distributed while behavior of the system (execution semantics) can be deployed and instantiated in running instance of the system. This deliverable provide the conceptual description and framework of the open Web Services architecture.
On the other hand we want to provide the design of the concrete implementation of this architecture (events mechanism, interfaces definition, deployment descriptors), which is used as a reference implementation of such a system. With evolving of this deliverable, system, managed as open source implementation at sourceforge, will evolve as well.

Contents

1 Introduction **4**

2 Conceptual Architecture versus Implementation Architecture **4**
2.1 Why Conceptual Architecture 4
2.2 Why Implementation Architecture 4

3 Architecture Components and Interfaces **4**
3.1 Architecture 4

4 Acknowledgement **5**

1 Introduction

The Web Services Modelling Execution Environment (WSMX) is an execution environment for dynamic discovery, mediation and invocation of Web Services. WSMX is based on the Web Services Modelling Ontology (WSMO) [2], an ontology for describing various aspects related to Semantic Web Services. A web service can be registered with WSMX by describing it in terms of WSMO, using the Web Service Modelling Language (WSML) [1], and then by invoking a registration interface provided by WSMX. A service requester with a goal to achieve, submits their goal to WSMX. The WSMX environment takes responsibility for matching the requester goal to capabilities of web services registered to WSMX, selecting the most appropriate Web Service, mediating between the ontologies of service requester and provider, and finally, invoking the selected web service.

2 Conceptual Architecture versus Implementation Architecture

2.1 Why Conceptual Architecture

2.2 Why Implementation Architecture

3 Architecture Components and Interfaces

3.1 Architecture

References

[1] Jos de Bruijn. D16 The WSML Specification. Technical report, WSML Working Draft, 2004. Available from `http://www.wsmo.org/2004/d16/v0.2/20041029/`.

[2] D. Roman, H. Lausen, and U. Keller. Web Service Modeling Ontology Standard. WSMO Working Draft v02, 2004.

4 Acknowledgement

The work is funded by the European Commission under the projects DIP, Knowledge Web, InfraWebs, SEKT, SWWS, ASG and Esperonto; by Science Foundation Ireland under the DERI-Lion project; by the Vienna city government under the CoOperate programme and by the FIT-IT (Forschung, Innovation, Technologie - Informationstechnologie) under the projects RW^2 and TSC.

The editors would like to thank to all the members of the WSMO, WSML, and WSMX working groups for their advice and input into this document.

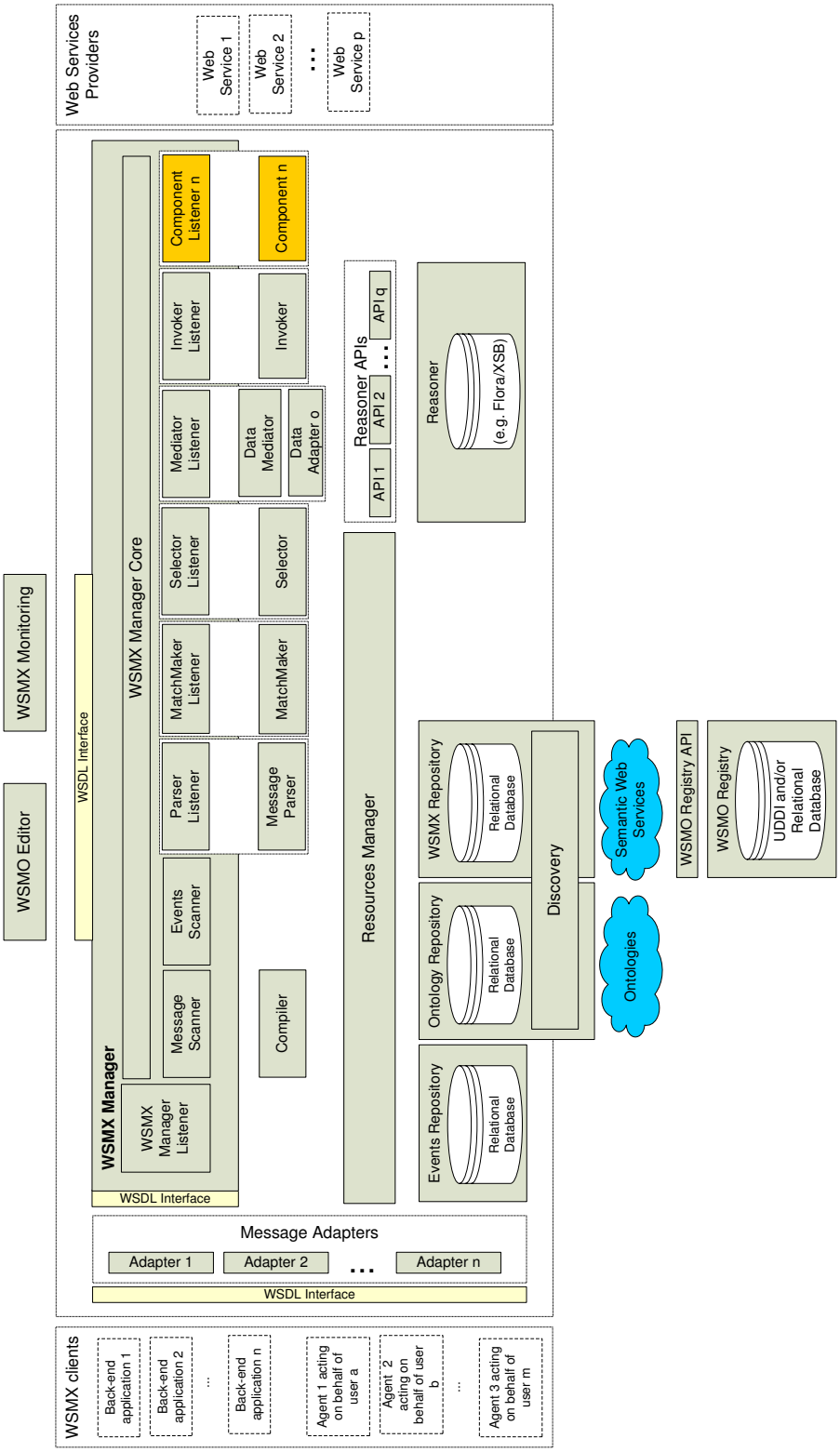

Figure 1: WSMX Architecture