

Using Semantic Web Technologies to Discover Resources within the Intranet of an Organization

Sabin C. Buraga

Faculty of Computer Science

“A.I.Cuza” University of Iasi, Romania

<http://www.infoiasi.ro/~busaco/>

Teodora Rusu

“P.Poni” Institute of Macromolecular Chemistry, Iasi

teia@icpp.ro



Agenda

- Introduction
- ITW – General Architecture
 - Overall Presentation
 - ITW Web Services
 - ITW Web Agents
 - ITW User Interface
- ITW – Implementation Prototype
- Conclusions & Further Work

Intro

- **Web** \equiv data storage & presentation (Berners-Lee, 1989)
- **Semantic Web** (Berners-Lee *et al.*, 2001)
 - Relations between objects that can be identified \Rightarrow unified management of distributed information
 - Information needs no longer to be intended for human readers only, but also for machine processing



Intro

- **Semantic Web applications:**
 - Intelligent information services
 - Personalized Web sites
 - Semantically empowered search engines
 - Intelligent collaborative Web environments (Weblogs, wikis,...)
 - Flexible user-interaction
 - ...

Intro

- One important issue is **resource discovery within enterprise intranet**
- We'll present a solution:
ITW – a multi-language distributed platform
 - exploits the temporal relations established between Web resources
 - uses a RDF/XML-based model for semantic representation of metadata and additional information (users, computers, time, etc.)



ITW

- **Goal:** to offer a heterogeneous interoperable infrastructure, based on Web components, for resource discovery within an intranet
 - Multi-platform (Linux & Windows)
 - Multi-language (C++, C#, Java, Perl)
 - Uses Web agents, (semantic) Web services, and other Web components
 - Communication between components is exclusively based on **XML (Extensible Markup Language)** family

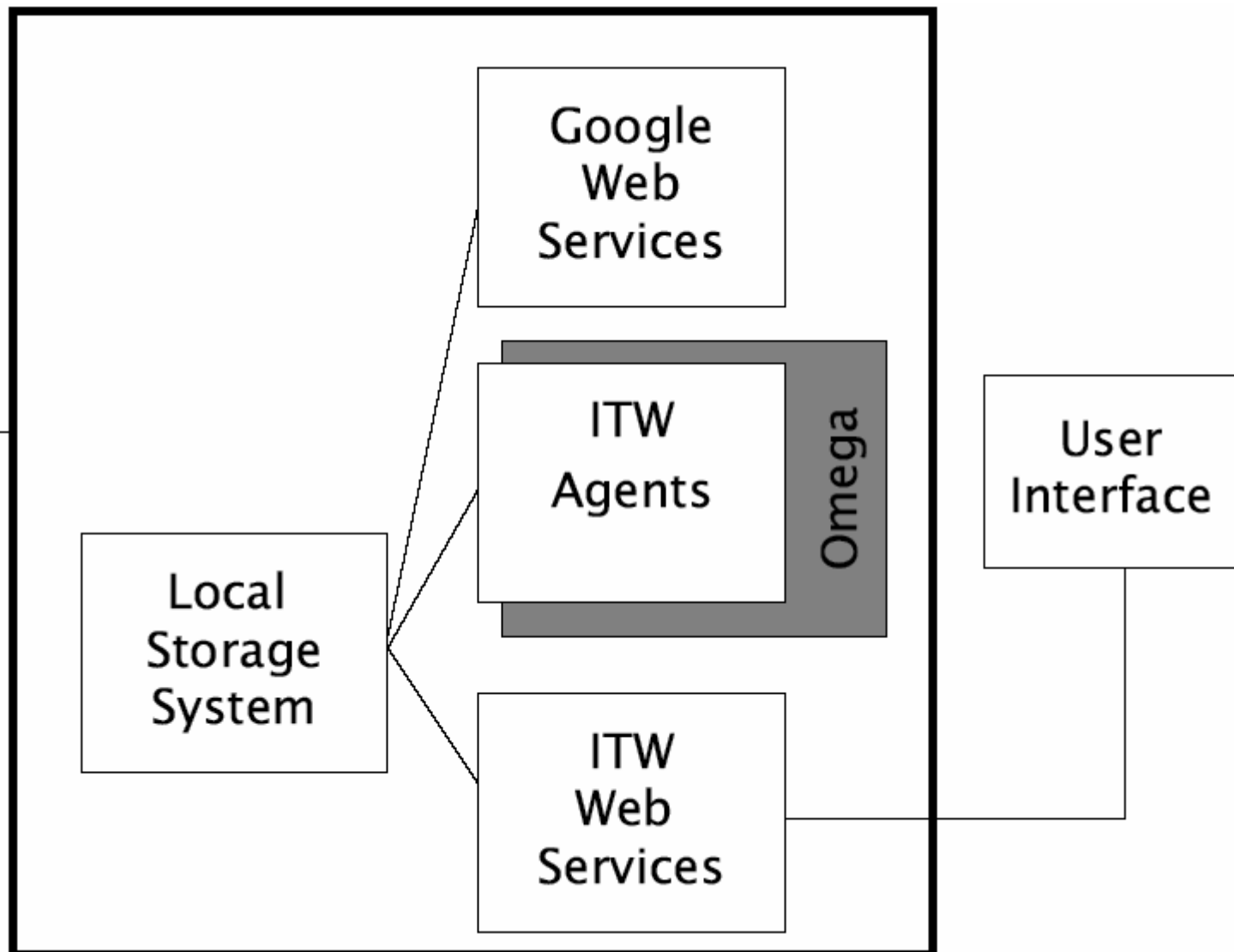


ITW

- Through a Web portal-like interface, the user will be able to formulate complex queries that involve time
- The information and the associated **RDF** (**Resource Description Framework**) metadata offered by ITW system is stored on independent servers



Web
(local)





ITW Web Services

- **Local Web services**
 - Provide information about the resources stored on the local Web (i.e. the intranet, the extranet or the public Web site)
 - Give access to these resources
- **External Web services**
 - developed by third-party organizations (e.g., Google)

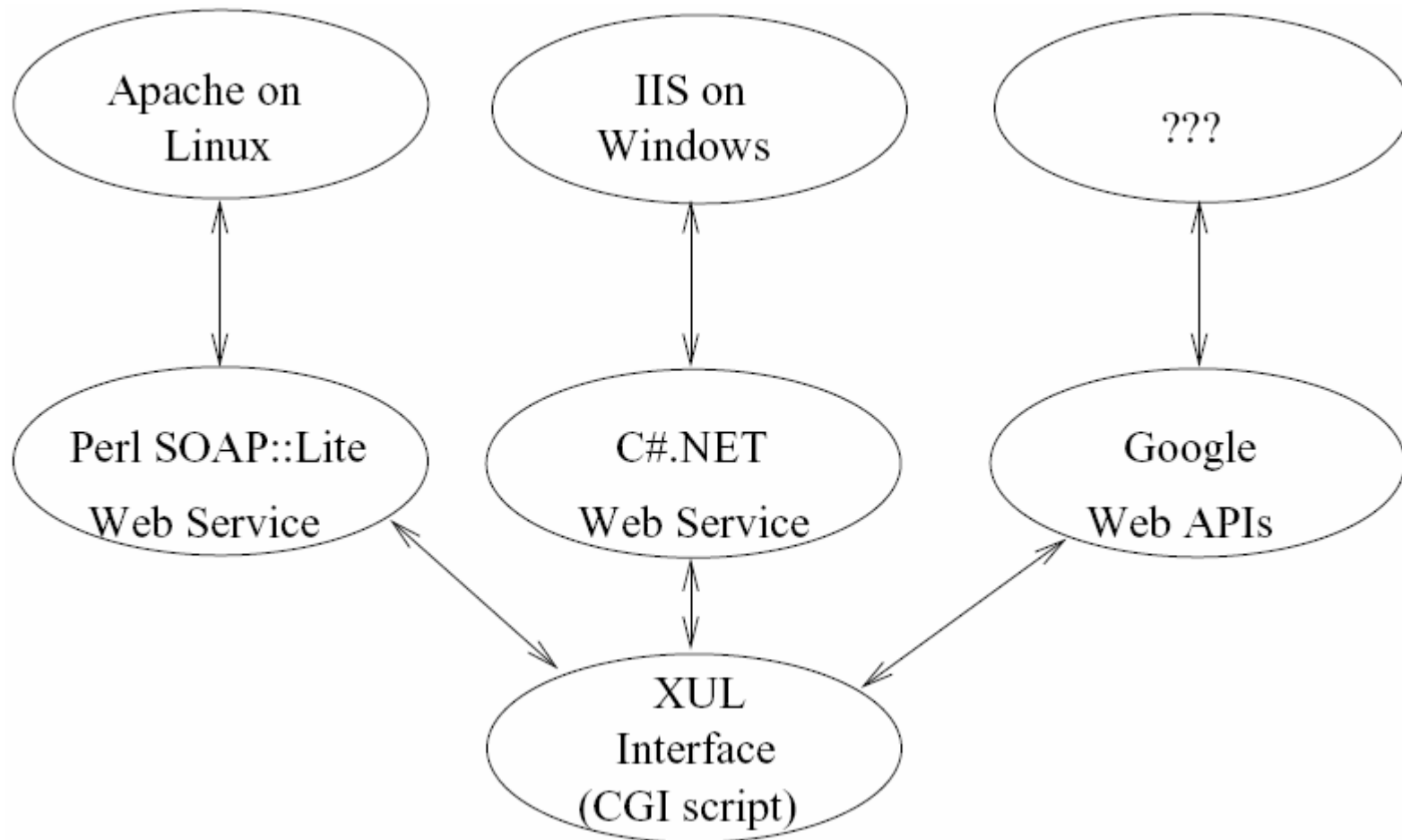


ITW Web Services

- The physical localization and execution of the ITW Web services are transparent for the final user
- We can consider these Web services as one entity (very similar with a computational Grid)
- To attach semantic descriptions for each Web service, **OWL-S** is used – process ontology expressed by **OWL (Web Ontology Language)** constructs



ITW Web Services



ITW

- **ITW local storage system**
 - Store RDF documents that contain metadata and relations between found resources
 - Automatically generated by ITW agents
 - Interrogated by ITW Web services/agents
 - Based on database relational management systems

ITW Web Agents

- Developed within a multi-agent platform:
 - Two XML-based modular approaches:
Omega (Alboaie & Buraga, 2004)
ADF – Agent Development Framework (Nichifor & Buraga, 2004; Hritcu & Buraga, 2005)

ITW Web Agents

- **Omega**

- Framework that offers an addressing space for the Web objects and a mechanism for remotely accessing the Web distributed resources (that can be viewed as objects)
- For each object, different metadata constructs (expressed in RDF) are attached to specify semantic properties
 - We are using different XML vocabularies

ITW Web Agents

- **Omega** agents' tasks
 - For each Web resource, a RDF document is generated in order to:
 - attach metadata: location, type (e.g., XHTML page, JPEG image, XSL stylesheet file, etc.), owner, access method, timestamp of last modification
 - **XFiles** (Buraga, 2002) language based on XML
 - express temporal information regarding the relationship between two resources – **TRSL** – **Temporal Relation Specification Language** (Buraga & Ciobanu, 2002; Buraga & Gabureanu, 2003)
 - based on ITL (Interval Temporal Logic)

ITW Web Agents

- Internal behavior modeled by BDI^K_{CTL} logic (Rao & Wooldridge, 1995, Wooldridge, 1996)
- Data serialization uses XML and SOAP (Alboaie & Buraga, 2003)
- Agent properties are specified by XML/RDF assertions (Buraga & Alboaie, 2005)



ITW Web Agents

- **ADF (Agent Development Framework)**
 - Second, more recent, open-source Java-based approach
 - Built as a **Service-Oriented Architecture (SOA)** to assure loose coupling between the interacting software agents
 - Agent communication is message-oriented and compatible with **FIPA (Foundation for Intelligent Physical Agents)** model
 - **<http://adf.sourceforge.net/>**

ITW User-Interface

- One CGI script that uses **XUL (Extensible Markup Language)**
 - provides a flexible query user-interface
- Queries are stored into **WQFL (Web Query Formulating Language)** format
 - denotes additional information about the search (e.g., relation with another resource, method of access, resource type, etc.) – see (Buraga & Rusu, 2000; Buraga & Brut, 2002)



ITW Actual Implementation

- 1 CGI script written in Perl language
 - generates XUL interface
- 2 ITW Web services
 - Linux (SOAP::Lite, Perl, Apache)
 - Windows (C#, .NET, IIS)
- 1 external Web service
 - provided by Google to discover world-wide resources
- ITW agents first implemented in C++ using Omega, then ADF (J2EE, under JBoss)
- 2 storage systems: MySQL (under Windows) and PostgreSQL (under Linux)

ITW

- Several tests were arranged for execution within an organizational intranet
- ITW can be viewed as a modular semantic Web-based open-source platform for discovering hypermedia information
- ITW uses different XML vocabularies in order to represent metadata and relations between Web resources

Further Work

- Instead of XUL, a set of XSL transformations can be applied
 - to offer a flexible interface for each Web client
- Storage system can use a native XML database system (e.g., eXist, Xindice,...)
- Deployment tests must be executed into a real life enterprise – following directions expressed by (Cioca & Buraga, 2003)
- Incorporating the ITW system into a Grid-like architecture – **tuBiG** (Alboaie & Buraga, 2003, Buraga & Alboaie, 2004)

Summary

- Introduction
- ITW – Internal Architecture
 - General Presentation
 - ITW Web Agents
 - ITW Web Services
 - ITW User Interface
- ITW – Actual Implementation
- Further Work



Online Resources

- **Agent Development Framework (ADF):**
<http://adf.sourceforge.net/>
- **Foundation for Intelligent Physical Agents (FIPA):**
<http://www.fipa.org/>
- **Semantic Web:** <http://www.semanticweb.org/>
- **Web Services:** <http://www.webservices.org/>
- **World-Wide Web Consortium:**
<http://www.w3.org/>

- **Our publications:**
<http://www.infoiasi.ro/~busaco/pub.html>



**Thank you
for your attention!**