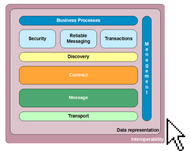
WS-Map: A Web-Based Survey Of Web Services Standards



http://web.ist.utl.pt/miguel.pardal/ws-map

Miguel Pardal

miguel.pardal@dei.ist.utl.pt

Department of Information Systems and Computer Engineering

Instituto Superior Técnico
Universidade Técnica de Lisboa

Interoperability is a critical requirement in enterprise software because most business processes cross organization and technology boundaries. **Web Services - WS -** technology greatly simplifies the integration between applications developed on different environments, like Java or Microsoft Dot Net.

A WS technology stack can be tailored by applications, to meet advanced requirements like **security**, **reliable messaging** or **transactions**. Each relevant standard must be well understood by itself and how it fits with others.

WS-Map is a survey on WS technology, available as a web site, offering a broad and vendor-independent view of horizontal WS standards. With it, developers, researchers and other practitioners can put the standards in perspective and make more informed technology decisions for their projects.

Web Services standards index

Business Processes

Business process standards leverage all the other Web Services technologies and define development abstractions closer to the needs of business.
WS-BPEL (Thatte, 2003) is an approach based on orchestration. The business process is represented by a graph, with the nodes being the business activities and the arcs being the information and control flows that enable composition of existing services.
WS-CDL (Kavantzas, 2004) is an approach based on choreography. The business process is described declaratively, by stating pre-conditions and post-conditions for the execution of activities. Process execution can change, as long as the stated conditions hold true.

Security

Security standards deal with message protection, access control, and configuration flexibility.
XML-Signature (Eastlake, 2002a) and **XML-Encryption** (Eastlake, 2002b) are the core standards.
WS-Security (Nadalin, 2004) deals with the protection of SOAP messages and the use of security tokens, like: cryptographic keys, digital certificates, assertions, etc. Tokens enable WS-Security to bind to existing security technologies, like **X.509** (Housley, 1999) and **Kerberos** (Kohl, 1993).
WS-SecurityPolicy (Kaler, 2005) extends WS-Policy with a vocabulary for security policies.
WS-Trust (Gudgin, 2005b) defines a trust brokering model.
WS-SecureConversation (Gudgin, 2005c) defines secure sessions encompassing several messages.
SAML assertions (Cantor, 2004) allow the exchange of authentication, authorization and attributes information between different security domains.

Reliable Messaging

Reliable messaging standards address the reliability of message exchanges in a transport independent way.
WS-Reliability (Iwasa, 2004) and **WS-ReliableMessaging** (Ferris, 2005) are two competing proposals for assured delivery, duplicate elimination and correct ordering in Web Services messaging.

Transactions

Transactions standards provide well-defined semantic properties for a group of operations, like ACID (atomicity, consistency, isolation and duration).
The underlying models assume temporary and recoverable fault models for machines and networks.
WS-Coordination (Feingold, 2005) and **WS-Composite Application Framework** (Little, 2003) are two alternative transaction frameworks.

Management

Management standards address the problem of keeping the services up-and-running.

Management has two complementary aspects:
- Management of the Web Services;
- Management of the machines, networks and other resources necessary for service execution.

Currently, the standardization efforts are focused on the latter kind of management. There are two competing standards: **WS Management** (Geller, 2004d) and **WS Distributed Management** (Sedukhin, 2005).

Discovery

Discovery standards define ways to publish and discover Web Services.
UDDI (Clement, 2004) defines a directory that allows dynamic registration and querying of Web Services.
WS-MEX (Curbera, 2004) enables Web Service self-description with a protocol to access XSD, WSDL, WS-Policy, and other meta-information.

Contract

Contract standards describe the data, functions and policy of a Web Service, enabling client-server binding.
The service interface is described with **WSDL** (Booth, 2005) and the data types are described with **XML Schema** (Fallside, 2004).
The WSDL contract is necessary but not sufficient to describe a service. **WS-Policy** (Schlimmer, 2006) states additional requirements that must be fulfilled by the client and by the service so that the interaction between them can occur.
For each non-functional requirement domain, like security, reliable messaging or transactions, there must be an extension library to implement it.

Message

Message standards define the structure of the communication units and the ways they can be exchanged between services.
SOAP (Gudgin, 2003) defines Web Services messages as XML documents with headers for platform data and body for application data.
WS-Addressing (Box, 2004) allows for the addressing and forwarding of SOAP messages in a transport-independent way.
MTOM/XOP (Gudgin, 2005a) is used to transport binary data in SOAP messages.
WS-Enumeration (Geller, 2004a) enables the creation of enumeration sessions.
WS-Eventing (Geller, 2004b) and **WS-Notification** (Graham, 2004) are competing standards for asynchronous event notification.
WS-Polling (Davis, 2005) specifies mechanisms for successive requests when asynchronous notifications are made impossible by a firewall.

Transport

Transport standards define ways to establish a communication channel between a client and a service.
The most widely used Web Services synchronous transport is **HTTP** (Fielding, 1999).
A common asynchronous transport is **SMTP** (Klensin, 2001).
There are also non-standard transport implementations using message queue systems.

Data representation

Data standards address the problem of heterogeneous representation i.e. how to represent data in a format that is identically understood by everyone.
XML (Bray, 2004) is a text-based tag language that allows the representation of data in a structured and self-describing way.
XSD (Fallside, 2004) is a grammar to define XML documents, including: elements, attributes, order, cardinality, data types and default values.

Interoperability

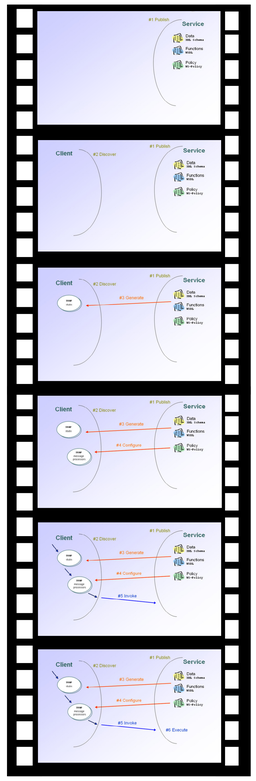
Interoperability profiles clear out ambiguities in standards. Each profile provides implementation guidelines, example applications, and compatibility testing toolkits.
WS-I is an organization that congregates the main vendors of Web Services tools and defines profiles like ***basic interaction*** (Ferris, 2004) and ***security*** (Barbir, 2005).
WS-DeviceProfile (Schlimmer, 2005) has a different scope, dealing with standards for services in devices with limited resources, like mobile phones.

Web Services in action

A **Web Service** is an access endpoint to data and functional resources of Enterprise applications.

Before a service can be invoked, the client and server must bind. The following action sequence describes the **binding** process.

Core technical principles:
- Message-orientation
- Encapsulation
- Autonomy
- Composition
- Interoperability



The service endpoint is published. Its network location and meta-data contracts are stored in a descriptor or service registry.
The data contract describes the data types used by the service interface, using XML Schema.
The interface contract describes the service functions through the input, output and fault messages, using WSDL.
The policy contract describes additional requirements for service interaction, using WS-Policy.

The client is an application that wants to use the service.
The client queries the descriptor or service registry to find the service's network location and its contracts.

The data and function contracts are used to automatically generate invocation stubs, that generate SOAP messages.

The policy is used to configure message processors and extension libraries, to add support for non-functional requirements like security, reliable messaging or transactions.

The client-service binding is complete.
The client invokes the service, using the stubs and configured processors to produce and send a message.

The service receives the message, verifies it and executes.
In some message exchange patterns, a response is returned.

Future work

WS-Map needs to keep pace with new developments of standards and implementations. To help in this, WS-Map should be upgraded to a Wiki-like application that enables user feedback to more easily become new content.

Regarding standards, a message queue transport specification is urgent, to take advantage of the features of these systems that are very useful for business processes modeling.

Also, the reliable messaging standards should be unified to improve transport over standard Internet protocols, when using message queues is difficult e.g. in cross organization integrations.

References

You can find a full reference listing in WS-Map's web site, readily available for download in BibTeX format.

You can also find the references in the publication:

Pardal, M.F.L., (2006).
Security of Enterprise Applications in Service Architectures,
Instituto Superior Técnico
http://mflpar.googlepages.com/MScMflp20060908.pdf