

# Secure, Reliable, Transacted; Innovation in Web Services Architecture

Martin Gudgin  
Microsoft Corp.  
One Microsoft Way  
Redmond, WA  
+1 425 882 8080

mgudgin@microsoft.com

## ABSTRACT

This paper discusses the design of Web Services Protocols paying special attention to composition of such protocols. The transaction related protocols are discussed as exemplars.

## 1. Introduction

Web Services are based on industry standards such as XML 1.0[1] and SOAP[2]. The former provides a standard representation format for information while the latter defines an extensible processing model for messages. Between them they provide a substrate for executing on constrained contracts between parties. As interest in Web Services has grown, so has the need to provide support for infrastructure services such as security, reliable delivery and transactions. This paper discusses the modeling of messages in XML, the SOAP processing model, and the protocol elements of the WS-Coordination[3], WS-AtomicTransaction[4] and WS-BusinessActivity[5] specifications.

## 2. XML Information Sets

The XML Information Set[6] defines an abstraction of the core information items found in XML data from the syntax used to serialize such data. Any XML tree has a corresponding information set. Such information sets can be serialized, typically using the serialization rules defined in XML 1.0.

## 3. SOAP

SOAP defines a message format and a processing model. Web Services use SOAP to exchange and process messages.

### 3.1 Message Format

SOAP defines a message format based on an XML envelope with header and body portions. The header portion can contain arbitrary XML elements known as header blocks. The body can also contain arbitrary XML. The distinction between the two pieces is that header blocks are a fundamental extensibility point in the specification allowing implementations to layer infrastructure level protocols on top of SOAP. Such protocols

include security protocols, reliable messaging and transactions. The body of the message is intended to carry information needed by the actual Web Service application.

SOAP messages are modeled as XML Information Sets. This allows SOAP to use different serialization techniques without changing the fundamental processing model. Today's Web Services use XML 1.0 as their serialization but in future other serialization such as MTOM[7], currently a work in progress, will be used. Regardless of which serialization is used, the information set remains the same and so the SOAP processing model can be applied.

### 3.2 Processing Model

A SOAP message travels from the initial sender, through zero or more intermediaries to an ultimate receiver. Header blocks provide information needed to perform some kind of processing either at a SOAP intermediary or at the ultimate receiver. The message body is always intended for processing by the ultimate receiver.

Header blocks can be marked as mandatory. If a recipient, be it an intermediary or the ultimate receiver of a message, does not recognize a mandatory header block then it must not perform any processing on the message body or other header blocks. This allows message senders to ensure that processing they deem important either happens, or none of the message is processed.

Header blocks can also be targeted at particular roles in a distributed system. Such roles may be played by intermediaries and the ultimate receiver. A given node may play multiple roles. An intermediary is required to process all header blocks that are targeted at any of the roles it plays. After such processing, the header blocks are typically removed. The message is then sent to the next node in the message path. This process continues until the message arrives at the ultimate receiver. At this point header blocks targeted at the ultimate receiver are processed along with the body of the message.

## 4. Composition

Different web services have different requirements with respect to security, reliability and transactional concerns amongst others. Any and all such protocols must fit into the SOAP processing model and, in addition, must compose with one another. The various web services protocol specifications defined thus far by Microsoft and its partners have been designed with this goal of composition very much at the forefront of the design.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGMOD 2004, June 13–18, 2004, Paris, France.

Copyright 2004 ACM 1-58113-859-8/04/06 ...\$5.00.

#### 4.1 Secure, Reliable, Transacted

Three of the main areas of interest in Web Services Protocols are Security, Reliable Messaging and Transactions. The protocol stack defines a set of security specifications that allow flexibility with respect to design of security protocols in the Web Services space. In addition a protocol for ensuring messages are reliably delivered has been defined. Lastly protocols for initiating and executing coordination between distributed parties are available. All these protocols are designed to work separately or together. Thus a Web Service can choose security along with transactional processing, security with reliable messaging or support for all three areas. As additional protocol layers are added to the Web Services stack over time, the existing protocols will compose with those new layers also.

#### 5. Web Services Coordination

The Web Services Coordination specification defines an extensible framework for coordinating activities using a coordinator and a set of coordination protocols. The framework does not constrain how a coordination protocol proceeds; rather it allows other specifications to specify such details. Transaction processing is but one area where such protocols can be defined. The specification defines how multiple parties can begin to interact according to some protocol. The protocol itself determines what actions the parties perform and how the parties determine when a given protocol has run to completion.

#### 6. Web Services Atomic Transaction

The Web Services Atomic Transaction specification defines several protocols suitable for short-running, high-trust interactions. The initiation of these protocols occurs based on constructs defined by WS-Coordination. The first such protocol, Completion, allows a user of a coordinator to request that a transaction complete. Completion can involve committing or aborting the transaction. The coordinator uses two other protocols, volatile 2-phase commit and durable 2-phase commit, in sequence, to determine the outcome in the commit case. Once

the transaction ends, the coordinator returns status to the initiator, indicating whether the transaction committed or aborted.

The two 2-phase commit protocols are identical but deal with different kinds of resources. A cache would be an example of a volatile resource while a database is an obvious example of a durable resource.

#### 7. Web Services Business Activity

The Web Services Business Activity Framework provides for the definition of agreement protocols based on long running activities. In addition it defines two such protocols; one based on participant oriented completion and the other based on coordinator oriented completion. These protocols are typically used for long-running activities where compensation rather than atomic rollback are the preferred error handling technique.

#### 8. REFERENCES

- [1] Bray, T., Paoli, J., Sperberg-McQueen, C.M., and Maler, E. Extensible Markup Language (XML) 1.0 (Second Edition). *W3C Recommendation 06 October 2000*.
- [2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H. SOAP Version 1.2 Part 1: Messaging Framework. *W3C Recommendation 24 June 2003*.
- [3] Cabrera, L.F. et.al. Web Services Coordination (WS-Coordination), September 2003.
- [4] Cabrera, L.F. et.al. Web Services Atomic Transaction (WS-Atomic Transaction), September 2003.
- [5] Cabrera, L.F. et.al. Web Services Business Activity Framework (WS-BusinessActivity), January 2004.
- [6] Cowan, J., and Tobin, R. XML Information Set. *W3C Recommendation 24 October 2001*.
- [7] Mendelsohn, N., Nottingham, M., and Ruellan, H. SOAP Message Transmission Optimization Mechanism. *W3C Working Draft 09 February 2004*.