



Resource Representation SOAP Header Block

W3C Recommendation 25 January 2005

This version:

<http://www.w3.org/TR/2005/REC-soap12-rep-20050125/>

Latest version:

<http://www.w3.org/TR/soap12-rep/>

Previous version:

<http://www.w3.org/TR/2004/PR-soap12-rep-20041116/>

Editors:

Anish Karmarkar, Oracle Corp.
Martin Gudgin, Microsoft
Yves Lafon, W3C

Please refer to the [errata](#) for this document, which may include normative corrections.

See also [translations](#).

[Copyright](#) © 2005 [W3C](#)® ([MIT](#), [ERCIM](#), [Keio](#)), All Rights Reserved. W3C [liability](#), [trademark](#) and [document use](#) rules apply.

Abstract

This document describes the semantics and serialization of a SOAP header block for carrying resource representations in SOAP messages.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](#) at <http://www.w3.org/TR/>.

This document is a Recommendation of the W3C. It has been reviewed by W3C Members and other interested parties and has been endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited as a normative reference from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread

deployment. This enhances the functionality and interoperability of the Web.

This document has been produced by the [XML Protocol Working Group](#) (WG) as part of the W3C [Web Services Activity](#). The English version of this specification is the only normative version. However, for translations of this document, see <http://www.w3.org/2003/03/Translations/byTechnology?technology=soap12-rep>.

Please report errors in this document to xm1p-comments@w3.org ([archive](#)). The errata list for this edition is available at <http://www.w3.org/2005/01/soap12-rep-errata>

This document is based upon the [Resource Representation SOAP Header Block Proposed Recommendation](#) of 16 November 2004. Feedback received during that review resulted in no changes. Evidence of interoperation between at least two implementations of this specification are documented in the [Implementation Summary](#). Changes between these two versions are described in a [diff document](#).

This document has been produced under the [24 January 2002 CPP](#) as amended by the [W3C Patent Policy Transition Procedure](#). An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) with respect to this specification should disclose the information in accordance with section 6 of the [W3C Patent Policy](#). Patent disclosures relevant to this specification may be found on the Working Group's [patent disclosure page](#).

A list of current W3C Recommendations and other technical documents can be found at <http://www.w3.org/TR/>.

Table of Contents

- 1 [Introduction](#)
 - 1.1 [Notational Conventions](#)
 - 1.2 [Relation to other specifications](#)
 - 1.2.1 [Relationship to the SOAP Processing model](#)
- 2 [SOAP Feature Name](#)
- 3 [SOAP Module Name](#)
- 4 [Representation Header Block](#)
 - 4.1 [Introduction](#)
 - 4.2 [Representation header block Constructs](#)
 - 4.2.1 [rep:Representation element](#)
 - 4.2.2 [resource attribute](#)
 - 4.2.3 [reinsert attribute](#)
 - 4.2.4 [rep:Data element](#)
 - 4.3 [Extensibility of the Representation header block](#)
 - 4.3.1 [SOAP header block Attributes](#)
 - 4.3.2 [Specifying the media type](#)
 - 4.3.3 [Extension example: HTTP resolver extension](#)

Appendices

- A [References](#)
- B [Acknowledgements](#) (Non-Normative)

1 Introduction

This document describes the semantics and serialization of a SOAP header block for carrying resource representations in SOAP messages.

1.1 Notational Conventions

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [\[RFC 2119\]](#).

This specification uses a number of namespace prefixes throughout; they are listed in [\[Prefixes and Namespaces used in this specification.\]](#). Note that the choice of any namespace prefix is arbitrary and not semantically significant (see XML Infoset [\[XMLInfoSet\]](#)).

Prefixes and Namespaces used in this specification.

Prefix	Namespace
	Notes
env	"http://www.w3.org/2003/05/soap-envelope"
	A normative XML Schema [XML Schema Part 1: Structures Second Edition] , [XML Schema Part 2: Datatypes Second Edition] document for the "http://www.w3.org/2003/05/soap-envelope" namespace can be found at http://www.w3.org/2003/05/soap-envelope .
rep	"http://www.w3.org/2004/08/representation"
	A normative XML Schema [XML Schema Part 1: Structures Second Edition] , [XML Schema Part 2: Datatypes Second Edition] document for the "http://www.w3.org/2004/08/representation" namespace can be found at http://www.w3.org/2004/08/representation .
xs	"http://www.w3.org/2001/XMLSchema"
	The namespace of XML Schema data types (see [XML Schema Part 2: Datatypes Second Edition]).
xmlmime	"http://www.w3.org/2004/11/xmlmime"
	The namespace for representing MIME media-types in XML (see [Assigning Media Types to Binary Data in XML]).

All parts of this specification are normative, with the exception of examples and sections explicitly marked as "Non-Normative".

1.2 Relation to other specifications

This document along with [\[XML-binary Optimized Packaging\]](#) and [\[SOAP Message Transmission Optimization Mechanism\]](#) has been produced in conjunction with the development of requirements, embodied in the requirements document [\[SOAP](#)

[Optimized Serialization Use Cases and Requirements](#)].

This document defines a SOAP Feature, and a SOAP Module that realizes the SOAP Feature, as specified by SOAP 1.2 [\[SOAP Version 1.2 Part 1: Messaging Framework\] 3.1 SOAP Features](#) and [\[SOAP Version 1.2 Part 1: Messaging Framework\] 3.3 SOAP Modules](#).

Note: The Resource Representation header block is designed to optimize well when used with the SOAP Message Transmission Optimization Mechanism [\[SOAP Message Transmission Optimization Mechanism\]](#).

Note: The Resource Representation header block is designed, but not required, to be used in conjunction with the Assigning Media Types to Binary Data in XML specification [\[Assigning Media Types to Binary Data in XML\]](#).

1.2.1 Relationship to the SOAP Processing model

This feature makes no changes to the SOAP processing model.

2 SOAP Feature Name

The Resource Representation header block defined by this document embodies the SOAP feature [\[SOAP Version 1.2 Part 1: Messaging Framework\] 3.1 SOAP Features](#) identified by the URI:

- "http://www.w3.org/2004/08/representation"

The above SOAP feature URI can be used to identify the semantics and serialization of the Resource Representation header block.

3 SOAP Module Name

The SOAP Module [\[SOAP Version 1.2 Part 1: Messaging Framework\] 3.3 SOAP Modules](#) that realizes the SOAP feature defined in [2 SOAP Feature Name](#) is identified by the URI:

- "http://www.w3.org/2004/08/representation"

The above SOAP Module URI can be used to identify the semantics and serialization of the Resource Representation header block.

4 Representation Header Block

This section describes a SOAP header block, the Representation header block, that allows a SOAP message to carry representations of Web resources.

4.1 Introduction

The Representation header block is designed to allow applications to carry a

representation of a Web resource in a SOAP message. Applications of this header include cases where the receiver has limited ability to get the representation using other means, for example because of access restrictions or because the overhead would be unacceptable. The Representation header block is also useful when multiple references to the same resource are required but duplication of the resource is undesirable. See UC2 and UC6 [\[SOAP Optimized Serialization Use Cases and Requirements\]](#) for details.

The meaning of the Representation header block, when present in a SOAP message, is to make available the contained representation of the resource it carries to the processing SOAP node. The SOAP node MAY use this representation when dereferencing the URI of the resource instead of making a network request to obtain a representation of the resource. Note that implementations MAY need to process a Representation header block before processing other header blocks that require dereferencing of a URI whose representation is carried in the Representation header block.

Multiple occurrences of the Representation header block MAY be present in the same SOAP Message to carry representation of multiple Web resources or multiple representations of the same Web resource.

Several occurrences of the Representation header block having the same value for the `role` and `resource` *attribute information item* (see [4.2.2 resource attribute](#)) MAY be present in the same SOAP Message. Such Representation header blocks SHOULD NOT have the same metadata (such as media-type). If such Representation header blocks have the same metadata then any one of them may be used.

URIs that are character for character identical MUST be considered equal when using a representation header to resolve a web reference; URIs that are considered equal according to the URI scheme of the URI SHOULD be considered equal.

An example SOAP Envelope using the Representation header block is given below.

Example: Representation header block in a SOAP Envelope

```
<soap:Envelope xmlns:soap='http://www.w3.org/2003/05/soap-envelope'
  xmlns:rep='http://www.w3.org/2004/08/representation'
  xmlns:xmlmime='http://www.w3.org/2004/11/xmlmime' >
  <soap:Header>
    <rep:Representation resource='http://example.org/me.png'>
      <rep>Data xmlmime:contentType='image/png'>/aWKKapGGyQ=</rep>Data>
    </rep:Representation>
  </soap:Header>
  <soap:Body>
    <x:MyData xmlns:x='http://example.org/mystuff'>
      <x:name>John Q. Public</x:name>
      <x:img src='http://example.org/me.png' />
    </x:MyData>
  </soap:Body>
</soap:Envelope>
```

4.2 Representation header block Constructs

4.2.1 `rep:Representation` element

The `Representation` *element information item* has:

- A [local name] of `Representation`.
- A [namespace name] of "<http://www.w3.org/2004/08/representation>".
- One or more *attribute information items* amongst its [attributes] as follows:
 - A mandatory `resource` *attribute information item* (see [4.2.2 resource attribute](#)).
 - An optional `reinsert` *attribute information item* (see [4.2.3 reinsert attribute](#)).
 - Zero or more namespace qualified *attribute information items* whose [namespace name] is not "<http://www.w3.org/2004/08/representation>".
- One or more *element information items* in its [children] property in order as follows:
 1. A mandatory `Data` *element information item* (see [4.2.4 rep:Data element](#)).
 2. Zero or more *element information items* whose [namespace name] is not "<http://www.w3.org/2004/08/representation>".

The `rep:Representation` *element information item* contains a representation of a Web resource. The value of the `resource` *attribute information item* is the identifier of the Web resource. The value of the `rep:Data` *element information item* is a base64-encoded representation of the Web resource.

4.2.2 `resource` attribute

The `resource` *attribute information item* has:

- A [local name] of `resource`.
- An empty [namespace name].
- A [specified] property with a value of "true".

The type of the `resource` *attribute information item* is `xs:anyURI`. The value of the `resource` *attribute information item* identifies the Web resource whose representation is carried in the `rep:Representation` *element information item* parent of the `resource` *attribute information item*.

4.2.3 `reinsert` attribute

The `reinsert` *attribute information item* has:

- A [local name] of `reinsert`.
- An empty [namespace name].

- A [specified] property with a value of "true".

The type of the `reinsert` *attribute information item* is `xs:boolean`. When this attribute is specified on the Representation header block with a value of "true", it indicates that a SOAP forwarding intermediary node processing the header block must reinsert the header block. This means that when used in conjunction with the `relay` attribute, defined in [\[SOAP Version 1.2 Part 1: Messaging Framework\] 5.2.4 SOAP Relay Attribute](#), with a value of "true", the Representation header block will always be relayed by a SOAP forwarding intermediary. When this attribute is specified on the Representation header block with a value of "false", the behavior of the SOAP node processing the header block is the same as that when the attribute is not specified, and normal SOAP processing rules apply. The presence of this attribute has no effect on the processing of a Representation header by a SOAP endpoint.

4.2.4 `rep:Data` element

The `Data` *element information item* has:

- A [local name] of `Data`.
- A [namespace name] of "http://www.w3.org/2004/08/representation".
- Zero or more namespace qualified *attribute information items* whose [namespace name] is not "http://www.w3.org/2004/08/representation".
- Any number of *character information item* in its [children] property. No other type of *information item* in its [children] property.

The type of a `rep:Data` *element information item* is `xs:base64Binary`. The value of this *element information item* is a base64-encoded representation of the Web resource carried in the `rep:Representation` *element information item* parent of the `resource` *attribute information item*.

4.3 Extensibility of the Representation header block

The Representation header block is built to be extensible. This section describes several possible usages of this extensibility.

4.3.1 SOAP header block Attributes

Attributes defined in [\[SOAP Version 1.2 Part 1: Messaging Framework\] 5. SOAP Message Construct](#) for SOAP header blocks MAY be used with the Representation header block.

Adding a `env:mustUnderstand` *attribute information item* with a value of "true" in the [attributes] property of the `rep:Representation` *element information item* ensures that the SOAP receiver is aware that the Web resource representation is available to it.

A `env:role` *attribute information item* in the [attributes] property of the `rep:Representation` *element information item* indicates the SOAP node for which the

Web resource representation is intended.

4.3.2 Specifying the media type

An `xmlmime:contentType` *attribute information item* (See [\[Assigning Media Types to Binary Data in XML\]](#)) MAY be used to convey the media type of the representation conveyed by a header. Media type information can be useful in determining whether a given representation is suitable for processing and if it is, how best to interpret the representation provided. If used, the `xmime:contentType` *attribute information item* MUST appear within the [attributes] property of the `rep:Data` *element information item*. If the media type identified by the value of an `xmlmime:contentType` *attribute information item* is a text based media type then the value of the `xmlmime:contentType` *attribute information item* SHOULD include a charset parameter.

An example that uses the `xmime:contentType` *attribute information item* is shown in [Example](#).

4.3.3 Extension example: HTTP resolver extension

A receiving SOAP node MAY act as a resolver, with all the rules pertaining to HTTP caches, for some or all of the `http:` scheme URIs for which representations have been provided. To enable this, one or more *element information items* MAY be added to the [children] property of the `rep:Representation` *element information item* to transmit the information needed at the HTTP level.

To avoid requiring that all SOAP senders understand the HTTP caching mechanism, all the data required by a processor that wants to act as a local cache needs to be carried along with the message. This includes the complete request, reply as well as the time the original HTTP request has been sent and the time the HTTP response has been received.

Example: HTTP extension

```
<soap:Envelope xmlns:soap='http://www.w3.org/2003/05/soap-envelope'
  xmlns:rep='http://www.w3.org/2004/08/representation'
  xmlns:xmlmime='http://www.w3.org/2004/11/xmlmime'>
  <soap:Header>
    <rep:Representation resource='http://example.org/me.png'>

      <rep:Data xmlmime:contentType='image/png'>/aWKKapGGyQ=</rep:Data>
      <htx:env xmlns:htx="http://www.w3.org/2004/08/representation/http">
        <htx:request>
          <htx:request-line name="GET" version="HTTP/1.1">
            /me.png
          </htx:request-line>
          <htx:header name="Host">
            example.org
          </htx:header>
          <htx:header name="Accept">
            image/png,image/jpeg,image/gif
          </htx:header>
          <htx:header name="Accept-Encoding">
            gzip,deflate,compress;q=0.9
          </htx:header>
        </htx:env>
      </rep:Representation>
    </soap:Header>
  </soap:Envelope>
```



```

    <htx:header name="Date">
      Fri, 13 Feb 2004 11:23:28 GMT
    </htx:header>
    [...]
    <htx:time>
      Fri, 13 Feb 2004 11:23:28 GMT
    </htx:time>
  </htx:request>
  <htx:reply>
    <htx:status-line version="HTTP/1.1" status="200">
      OK
    </htx:status-line>
    <htx:header name="Content-Type">
      image/png
    <htx:header>
    <htx:header name="Date">
      Fri, 13 Feb 2004 11:23:28 GMT
    </htx:header>
    [...]
    <htx:time>
      Fri, 13 Feb 2004 11:23:32 GMT
    </htx:time>
    </htx:reply>
  </htx:env>

</rep:Representation>
</soap:Header>
<soap:Body>
  <x:MyData xmlns:x='http://example.org/mystuff'>
    <x:name>John Q. Public</x:name>
    <x:img src='http://example.org/me.png' />
  </x:MyData>
</soap:Body>
</soap:Envelope>

```

Note that if the clocks of the SOAP sender and the SOAP recipient are not synchronized, all the expiration/age computed at the receiving side will not accurately reflect what could have been computed at the SOAP sender side.

A References

SOAP Version 1.2 Part 1: Messaging Framework

[SOAP Version 1.2 Part 1: Messaging Framework](#), Marc Hadley, Noah Mendelsohn, Jean-Jacques Moreau, *et. al.*, Editors. World Wide Web Consortium, 24 June 2003. This version is <http://www.w3.org/TR/2003/REC-soap12-part1-20030624/>. The [latest version](#) is available at <http://www.w3.org/TR/soap12-part1/>.

XML-binary Optimized Packaging

[XML-binary Optimized Packaging](#), Mark Nottingham, Noah Mendelsohn, Martin Gudgin, and Hervé Ruellan, Editors. World Wide Web Consortium, 25 January 2005. This version is <http://www.w3.org/TR/2005/REC-xop10-20050125/>. The [latest version](#) is available at <http://www.w3.org/TR/xop10/>.

SOAP Message Transmission Optimization Mechanism

[SOAP Message Transmission Optimization Mechanism](#), Hervé Ruellan, Noah Mendelsohn, Martin Gudgin, and Mark Nottingham, Editors. World Wide Web Consortium, 25 January 2005. This version is <http://www.w3.org/TR/2005/REC-soap12-mtom-20050125/>. The [latest version](#) is

available at <http://www.w3.org/TR/soap12-mtom/>.

SOAP Optimized Serialization Use Cases and Requirements

[*SOAP Optimized Serialization Use Cases and Requirements*](#), Tony Graham, Mark Jones, and Anish Karmarkar, Editors. World Wide Web Consortium, 08 June 2004. This version is <http://www.w3.org/TR/2004/WD-soap12-os-ucr-20040608/>. The [latest version](#) is available at <http://www.w3.org/TR/soap12-os-ucr/>.

XML Information Set (Second Edition)

[*XML Information Set \(Second Edition\)*](#), Richard Tobin and John Cowan, Editors. World Wide Web Consortium, 04 Feb 2004. This version is <http://www.w3.org/TR/2004/REC-xml-infoset-20040204>. The [latest version](#) is available at <http://www.w3.org/TR/xml-infoset>.

XML Schema Part 1: Structures Second Edition

[*XML Schema Part 1: Structures Second Edition*](#), David Beech, Murray Maloney, Henry S. Thompson, and Noah Mendelsohn, Editors. World Wide Web Consortium, 28 October 2004. This version is <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/>. The [latest version](#) is available at <http://www.w3.org/TR/xmlschema-1/>.

XML Schema Part 2: Datatypes Second Edition

[*XML Schema Part 2: Datatypes Second Edition*](#), Ashok Malhotra and Paul V. Biron, Editors. World Wide Web Consortium, 28 October 2004. This version is <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>. The [latest version](#) is available at <http://www.w3.org/TR/xmlschema-2/>.

RFC 2119

[*Key words for use in RFCs to Indicate Requirement Levels*](#), S. Bradner, Editor. IETF, March 1997. This RFC is available at <http://www.ietf.org/rfc/rfc2119.txt>.

RFC 2396

[*Uniform Resource Identifiers \(URI\): Generic Syntax*](#), T. Berners-Lee, R. Fielding and L. Masinter, Editors. IETF, August 1998. This RFC is available at <http://www.ietf.org/rfc/rfc2396.txt>.

RFC 2732

[*Format for Literal IPv6 Addresses in URL's*](#), R. Hinden, B. Carpenter and L. Masinter, Editors. IETF, December 1999. This RFC is available at <http://www.ietf.org/rfc/rfc2732.txt>.

Assigning Media Types to Binary Data in XML

[*Assigning Media Types to Binary Data in XML*](#), Ümit Yalçınalp and Anish Karmarkar, Editors. World Wide Web Consortium, 02 November 2004. This version is <http://www.w3.org/TR/2004/WD-xml-media-types-20041102>. The [latest version](#) is available at <http://www.w3.org/TR/xml-media-types>.

B Acknowledgements (Non-Normative)

This specification is the work of the W3C XML Protocol Working Group.

Participants in the Working Group are (at the time of writing, and by alphabetical order): David Fallside (IBM), Tony Graham (Sun Microsystems), Martin Gudgin (Microsoft Corporation, formerly of DevelopMentor), Marc Hadley (Sun Microsystems), Gerd Hoelzing (SAP AG), John Ibbotson (IBM), Anish Karmarkar (Oracle), Suresh Kodichath (IONA Technologies), Yves Lafon (W3C), Michael Mahan (Nokia), Noah Mendelsohn (IBM, formerly of Lotus Development), Jeff Mischkin (Oracle), Jean-Jacques Moreau

(Canon), Mark Nottingham (BEA Systems, formerly of Akamai Technologies), David Orchard (BEA Systems, formerly of Jamcracker), Herve Ruellan (Canon), Jeff Schlimmer (Microsoft Corporation), Pete Wenzel (SeeBeyond), Volker Wiechers (SAP AG).

Previous participants were: Yasser alSafadi (Philips Research), Bill Anderson (Xerox), Vidur Apparao (Netscape), Camilo Arbelaez (webMethods), Mark Baker (Idokorro Mobile, Inc., formerly of Sun Microsystems), Philippe Bedu (EDF (Electricite De France)), Olivier Boudeville (EDF (Electricite De France)), Carine Bournez (W3C), Don Box (Microsoft Corporation, formerly of DevelopMentor), Tom Breuel (Xerox), Dick Brooks (Group 8760), Winston Bumpus (Novell, Inc.), David Burdett (Commerce One), Charles Campbell (Informix Software), Alex Ceponkus (Bowstreet), Michael Champion (Software AG), David Chappell (Sonic Software), Miles Chaston (Epicentric), David Clay (Oracle), David Cleary (Progress Software), Dave Cleary (webMethods), Ugo Corda (Xerox), Paul Cotton (Microsoft Corporation), Fransisco Cubera (IBM), Jim d'Augustine (Excelon Corporation), Ron Daniel (Interwoven), Glen Daniels (Macromedia), Doug Davis (IBM), Ray Denenberg (Library of Congress), Paul Denning (MITRE Corporation), Frank DeRose (TIBCO Software, Inc.), Mike Dierken (DataChannel), Andrew Eisenberg (Progress Software), Brian Eisenberg (DataChannel), Colleen Evans (Sonic Software), John Evdemon (XMLSolutions), David Ezell (Hewlett Packard), James Falek (TIBCO Software, Inc.), Eric Fedok (Active Data Exchange), Chris Ferris (Sun Microsystems), Daniela Florescu (Propel), Dan Frantz (BEA Systems), Michael Freeman (Engenia Software), Dietmar Gaertner (Software AG), Scott Golubock (Epicentric), Mike Greenberg (IONA Technologies), Rich Greenfield (Library of Congress), Hugo Haas (W3C), Mark Hale (Interwoven), Randy Hall (Intel), Bjoern Heckel (Epicentric), Frederick Hirsch (Zolera Systems), Erin Hoffmann (Tradia Inc.), Steve Hole (MessagingDirect Ltd.), Mary Holstege (Calico Commerce), Jim Hughes (Fujitsu Limited), Oisin Hurley (IONA Technologies), Yin-Leng Husband (Hewlett Packard, formerly of Compaq), Ryuji Inoue (Matsushita Electric Industrial Co., Ltd.), Scott Isaacson (Novell, Inc.), Kazunori Iwasa (Fujitsu Limited), Murali Janakiraman (Rogue Wave), Mario Jeckle (DaimlerChrysler Research and Technology), Eric Jenkins (Engenia Software), Mark Jones (AT&T), Jay Kasi (Commerce One), Jeffrey Kay (Engenia Software), Richard Koo (Vitria Technology Inc.), Jacek Kopecky (Systinet), Alan Kropp (Epicentric), Julian Kumar (Epicentric), Peter Lecuyer (Progress Software), Tony Lee (Vitria Technology Inc.), Michah Lerner (AT&T), Bob Lojek (Intalio Inc.), Henry Lowe (OMG), Brad Lund (Intel), Matthew MacKenzie (XMLGlobal Technologies), Murray Maloney (Commerce One), Richard Martin (Active Data Exchange), Alex Milowski (Lexica), Kevin Mitchell (XMLSolutions), Nilo Mitra (Ericsson), Ed Mooney (Sun Microsystems), Dean Moses (Epicentric), Highland Mary Mountain (Intel), Don Mullen (TIBCO Software, Inc.), Rekha Nagarajan (Calico Commerce), Raj Nair (Cisco Systems), Masahiko Narita (Fujitsu Limited), Mark Needleman (Data Research Associates), Art Nevarez (Novell, Inc.), Eric Newcomer (IONA Technologies), Henrik Nielsen (Microsoft Corporation), Conleth O'Connell (Vignette), Kevin Perkins (Compaq), Jags Ramnaryan (BEA Systems), Andreas Riegg (DaimlerChrysler Research and Technology), Vilhelm Rosenqvist (NCR), Marwan Sabbouh (MITRE Corporation), Waqar Sadiq (Vitria Technology Inc.), Rich Salz (Zolera Systems), Krishna Sankar (Cisco Systems), George Scott (Tradia Inc.), Shane Sesta (Active Data Exchange), Lew Shannon (NCR), John-Paul Sicotte (MessagingDirect Ltd.), Miroslav Simek (Systinet), Simeon Simeonov (Macromedia), Aaron Skonnard (DevelopMentor), Nick Smilonich (Unisys), Seumas Soltysik (IONA Technologies), Soumitro Tagore (Informix Software), James Tauber (Bowstreet), Anne Thomas Manes (Sun Microsystems), Lynne Thompson (Unisys), Patrick Thompson (Rogue Wave), Jim

Trezzo (Oracle), Asir Vedamuthu (webMethods), Randy Waldrop (WebMethods), Fred Waskiewicz (OMG), David Webber (XMLGlobal Technologies), Ray Whitmer (Netscape), Stuart Williams (Hewlett Packard), Yan Xu (DataChannel), Amr Yassin (Philips Research), Susan Yee (Active Data Exchange), Jin Yu (MartSoft Corp.).

This document has its genesis as a separate section in the [SOAP Message Transmission Optimization Mechanism \(MTOM\)](#) document. The editors of the MTOM document, specifically Mark Nottingham and Hervé Ruellan are gratefully acknowledged.

The people who have contributed to discussions on xml-dist-app@w3.org are also gratefully acknowledged.