SOAP 1.2 Attachment Feature

W3C Working Group Note 8 June 2004

This version:
http://www.w3.org/TR/2004/NOTE-soap12-af-20040608/
Latest version:
http://www.w3.org/TR/soap12-af/
Previous version:
http://www.w3.org/TR/2002/WD-soap12-af-20020924/

Editors:
Henrik Frystyk Nielsen, Microsoft
Hervé Ruellan, Canon

Abstract

This document defines a SOAP feature that represents an abstract model for SOAP attachments. It provides the basis for the creation of SOAP bindings that transmit such attachments along with a SOAP envelope, and provides for reference of those attachments from the envelope. SOAP attachments are described using the notion of a compound document structure consisting of a primary SOAP message part and zero or more related documents parts known as attachments.

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 the work of the W3C XML Protocol Working Group (WG). The Attachment Feature document has been superceded by the SOAP Message Transmission Optimization Mechanism document which describes attachment related features along with some implementation details. The XMLP WG does not intend to do any further work on the Attachment Feature document.

Discussion of this document takes place on the public xml-dist-app@w3.org mailing list (public archive) under the email communication rules in the XML Protocol Working Group Charter .

The XML Protocol Working Group is part of the Web Services Activity.

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.

Publication as a Working Group Note does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

Table of Contents

Appendices

1 Introduction

SOAP 1.2 part 1 (see [SOAP Part 1]) provides a flexible and extensible envelope for describing structured information intended for exchange between SOAP nodes. Even though SOAP 1.2 is described in terms of [XML Infoset], it is expected that [XML 1.0] will be a widely used representation for SOAP data.

The following problems can arise when using such an [XML 1.0] representation for SOAP data:

1. Encapsulation of binary data in the form of image files etc. cannot be done without additional encoding/decoding of the data. The overhead of encoding binary data in a form acceptable to XML (for example using base64 as defined by [RFC 2045]) is often significant both in terms of bytes added because of the encoding as well as processor overhead performing the encoding/decoding.

2. Encapsulation of other XML documents as well as XML fragments is cumbersome within a SOAP message--especially if the XML parts do not use the same character encoding.

3. Although SOAP messages inherently are self-delimiting, the message delimiter can only be detected by parsing the complete message. This can imply a significant overhead in generic message processing as well as parsing and memory allocation.

The compound message structures provided by this specification MAY be used to create SOAP bindings, or other specifications to be used by bindings, that avoid some or all such problems.

The purpose of this specification is not to define an actual representation of SOAP attachments but rather to define an abstract SOAP 1.2 feature which can be used as the basis for defining SOAP bindings that support the transmission of messages with attachments. The feature describes characteristics common to all such implementations.

1.1 Notational Conventions

The key words "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].

1.2 Conformance

This document describes a SOAP attachment feature which is an abstract model, and conformance is a property of SOAP binding specifications or of SOAP modules that use this model.

A SOAP binding specification or a SOAP module using this model is conformant if it follows all the requirements of this specification (see in particular **6 Implementation**).

2 SOAP Feature Name

This Attachment Feature is identified by the URI

(see [SOAP Part 1] SOAP Protocol Binding Framework)

:

- "http://www.w3.org/2002/06/soap/features/attachment" .

3 Terminology

Compound SOAP structure

A compound SOAP structure consists of a primary SOAP message part and zero or more related secondary parts.

Primary SOAP message part

A SOAP message that provides the processing context for the compound SOAP structure as a whole including the secondary parts.

Secondary Part

A document or entity related to the primary SOAP message part in some manner. A secondary part is a resource in the sense that it has identity and is identified by a URI. The representation of the resource can be of any type and size. Secondary parts are informally referred to as attachments.

Note: the use of the term "part" in this specification is independent of its use in other specifications and should not be assumed to be identical.

4 Compound SOAP Structure Model

Throughout this document, the components of a compound document structure are called parts. Accordingly, a compound SOAP structure consists of a primary SOAP message part and zero or more related secondary parts that are distinct from the primary SOAP message but related to it in some manner.

For example secondary parts can be used to contain data that naturally represents a resource in its own right or which is cumbersome to represent within the primary SOAP message part. The latter can be due to the size, type, or format of the data--a secondary part may be an audio clip, an image, or a very large view of a database, for example.

Secondary parts are often informally referred to as "attachments". While the attachment relationship is expected to be commonly used, the model makes no assumption about the nature of the semantic relationship between the primary SOAP message part and secondary parts, or between secondary parts.

It is important to note that the compound SOAP structure model does not modify or supersede the message envelope concept defined by SOAP. Nor does it define a processing model for any of the parts of a compound SOAP structure including the primary SOAP message part. The processing model for the primary SOAP message part is defined by SOAP. The application-defined semantics of the SOAP message provides the processing context for the secondary part(s).

Each part is identified by one or more URIs (typically one) that can be used to reference it from other parts. The URI(s) identifying a part can be of any URI scheme. It is

RECOMMENDED that only IANA registered URI schemes be used.

The compound SOAP structure model does not require that a SOAP receiver process, dereference, or otherwise verify any secondary parts of a compound SOAP structure. It is up to the SOAP receiver to determine, based on the processing context provided by the primary SOAP message part, which operations must be performed (if any) on the secondary part(s).

5 Attachment Feature properties

The Attachment Feature defines a set of properties described in Property definition for the Attachment FeatureProperty NameProperty Description SOAPMessage An abstract structure that represents the primary SOAP message part of the compound SOAP structure. SecondaryPartBag An abstract structure that represents the compound SOAP structure's secondary part(s). This structure is a bag that contains representations of each of the compound SOAP structure's secondary part(s). A secondary part representation can be a URI referencing this secondary part, an abstract structure representing the secondary part itself, or both. .

Note: the base URI for all the properties defined in this section is `http://www.w3.org/2002/06/soap/features/attachment`. In this section, property names are sometimes given using a URI relative to this base URI.

Property definition for the Attachment Feature

Property Name	Property Description
`SOAPMessage`	An abstract structure that represents the primary SOAP message part of the compound SOAP structure.
`SecondaryPartBag`	An abstract structure that represents the compound SOAP structure's secondary part(s). This structure is a bag that contains representations of each of the compound SOAP structure's secondary part(s). A secondary part representation can be a URI referencing this secondary part, an abstract structure representing the secondary part itself, or both.

Both these properties are instantiated inside the scope of a message exchange context. If several messages are exchanged inside the scope of this message exchange context, each instantiation of those properties is linked to a particular message.

Note: the

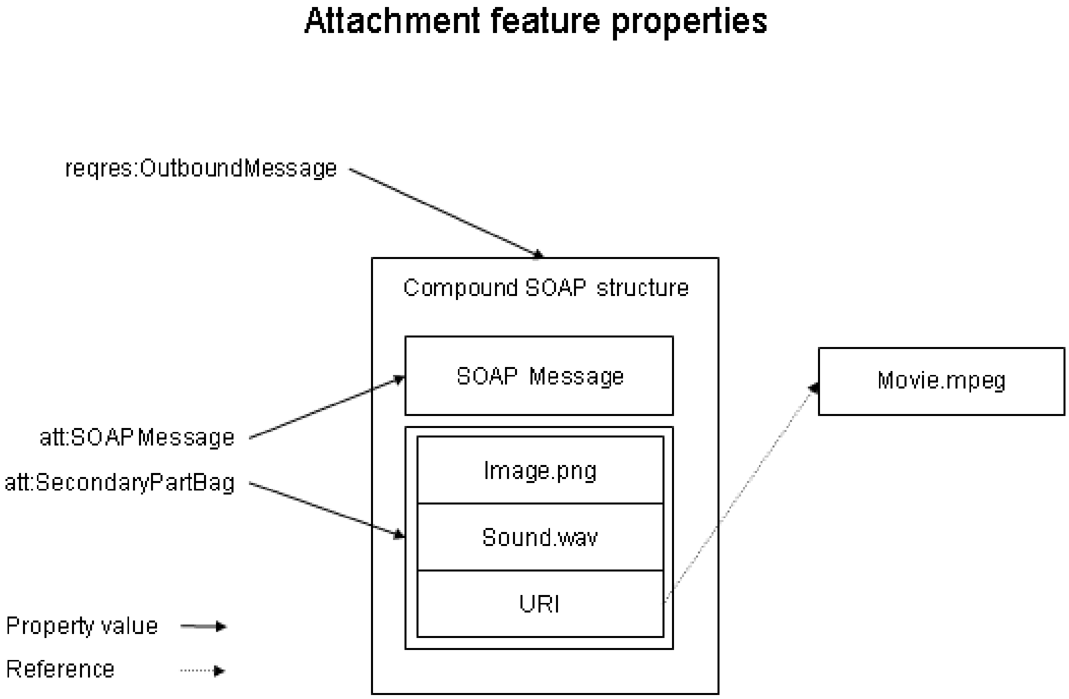
```
http://www.w3.org/2002/06/soap/features/attachment/SOAPMessage
```

and

```
http://www.w3.org/2002/06/soap/features/attachment/SecondaryPartBag
```

properties may interact with or affect the contents of other properties (from a MEP or another feature) defining the message sent. It is up to the implementation to specify how those properties interact.

For example, the Request-Response Message Exchange Pattern in the [SOAP Part 2] specification defines a reqres:OutboundMessage property that represents the current outbound message in the message exchange. If the Request-Response Message Exchange Pattern is used in conjunction with this feature, then the reqres:OutboundMessage property is initialized to represent the compound SOAP Structure (see diagram below).



5.1 Sending a compound SOAP structure

To use this feature, a SOAP node sending a message instantiates a local message exchange context. Property definition for the Attachment FeatureProperty NameProperty Value SOAPMessage A representation of the primary SOAP message part of the outbound message. SecondaryPartBag A representation of all the secondary parts of the outbound message. describes how the context is initialized.

Property definition for the Attachment Feature

Property Name	Property Value
`SOAPMessage`	A representation of the primary SOAP message part of the outbound message.
`SecondaryPartBag`	A representation of all the secondary parts of the outbound message.

There may be other properties related to the operation of the message exchange context instance. Such properties are initialized according to their own feature specifications.

5.2 Receiving a compound SOAP structure

When the SOAP protocol binding instance at the receiving SOAP node starts to receive an inbound message using this feature, it (logically) instantiates a message exchange context. Property definition for the Attachment FeatureProperty NameProperty Value SOAPMessage A representation of the primary SOAP message part of the inbound message. SecondaryPartBag A representation of all the secondary parts of the inbound message. describes the properties that the binding initializes as part of the context's instantiation.

Property definition for the Attachment Feature

Property Name	Property Value
`SOAPMessage`	A representation of the primary SOAP message part of the inbound message.
`SecondaryPartBag`	A representation of all the secondary parts of the inbound message.

There may be other properties related to the operation of the message exchange context instance. Such properties are initialized according to their own feature specifications.

6 Implementation

The compound SOAP structure model is abstract in the sense that it does not define an actual means by which compound SOAP structures are represented or transmitted by a SOAP binding. This section describes the requirements on any binding that uses this feature to transmit SOAP compound structures; the definition of any particular binding, or of particular representations of compound structures to be used by such bindings, is outside the scope of this specification.

A binding that supports the transmission of compound SOAP structures MUST provide the following:

- A means by which the primary and secondary parts are made available to the receiving party. Typically, this is achieved by transmitting all of the parts from the sender to the receiver, using binding-specified means. One option for achieving such transmission is to use an encapsulation mechanism (e.g. DIME or MIME) to prepare a single data stream containing all of the parts, and to then transmit the encapsulation.
- A mechanism by which each part is identified using one (or more) URI(s)

  (see [SOAP Part 1] Use of URIs in SOAP)

  . The URI scheme used MAY but need not be the same for all parts. The URI scheme used for multiple identifiers of a single part MAY but need not be the same.

  Note: the ability to identify a single part with multiple URIs is provided because, in general, the Web architecture allows such multiple names for a single resource. It is anticipated that most bindings will name each part with a single URI, and through the use of base URIs, provide for absolute and/or relative URI references to that URI.

The compound SOAP structure model is independent of the underlying protocol used for transmitting the primary SOAP message part and any of the secondary parts. That is, there is no requirement that all parts of a compound SOAP structure representation be transmitted within the same unit of the underlying protocol. Note that in some cases, the underlying protocol may also provide the functionality of the encapsulation mechanism.

As an example of possible representations that a binding might implement, consider a compound SOAP structure consisting of a primary SOAP message part containing an insurance claim for a damaged car and a secondary part containing a JPEG image of the car. Such a compound structure can be represented in several ways including but not limited to the following:

1. The primary SOAP message part and the JPEG image may be encapsulated in a single DIME message and transmitted using an underlying protocol such as TCP or HTTP ([RFC 2616]) (see [WS-Attachments]).

2. The primary SOAP message part and the JPEG image may be encapsulated in a single MIME Multipart/Related message ([RFC 2387]) and transmitted using an underlying protocol such as HTTP ([RFC 2616]) (see [SOAP with attachments]).

3. The primary SOAP message part may be exchanged using the HTTP protocol binding without any further encapsulation

   ,

   and the JPEG image

   retrieved at the application's discretion

   using a separate HTTP GET request.

A binding that supports this feature MUST provide a means by which receivers can distinguish the primary SOAP part from the secondary parts. A SOAP receiver that supports this feature MUST process the primary SOAP message part according to the rules for processing SOAP messages (see [SOAP Part 1]).

Compound SOAP structures MAY be nested by having a secondary part of a compound SOAP structure contain another compound SOAP structure. In this case, the primary SOAP message part is the primary SOAP message part of the outermost compound SOAP structure and as for any other secondary part, the semantics of the primary SOAP message part provides the processing context for the nested compound SOAP structure(s).

While a binding that supports this feature MAY provide mechanisms for verifying the integrity and enumerating the parts of a compound SOAP structure, this is not a requirement of this feature.

7 Intermediaries

A SOAP message can travel through zero or more SOAP intermediaries. This section describes the requirements posed on SOAP intermediaries supporting this specification.

A SOAP intermediary MUST be able to access any secondary part.

A forwarding SOAP intermediary MUST in general forward every secondary parts contained in the incoming SOAP message, except when the specification for a processed SOAP header block calls for the part to be removed or changed. An active SOAP intermediary MAY change or remove any secondary part even in the absence of such a mandate.

A SOAP intermediary MAY insert new secondary parts.

The integrity of references (i.e. URIs) to secondary parts MUST be maintained accross SOAP intermediaries. That is, a URI which resolves to a secondary part in an inbound SOAP message MUST continue to resolve to that part in the outbound message, unless that part was removed by the SOAP intermediary.

8 Internationalization Considerations

This specification does not introduce internationalization considerations beyond what is introduced by [SOAP Part 1], and URIs ([RFC 2396]). This specification refers to the specific considerations described by those specifications.

9 Security Considerations

Implementers should pay special attention to the security implications of any payload types that can cause the remote execution of any actions in the recipient's environment. Before accepting payloads of any type, an application should be aware of the particular security implications associated with that type.

Security considerations for media types in general are discussed in [RFC 2048] and in the context of the "application/postscript" and the "message/external-body" media type in [RFC 2046].

To address concerns about integrity and confidentiality, secondary parts can be digitally signed and encrypted. Typically, a compound SOAP structure that contains signed or encrypted secondary parts would include security information about the secondary parts in a security header of the primary SOAP message part. This specification does not provide a means to protect a message by encrypting and/or digitally signing a body, a header, a secondary part, or any combination of them (or parts of them). Other specifications (see for example [WS-Security]) can provide such means.

10 IANA Considerations

This specification does not describe any components that require registration by IANA.

11 References

11.1 Normative References

RFC 2026
IETF "The Internet Standards Process -- Revision 3", S. Bradner, October 1996. (See http://www.ietf.org/rfc/rfc2026.txt.)

RFC 2045
IETF "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", N. Freed, N. Borenstein, November 1996. (See http://www.ietf.org/rfc/rfc2045.txt.)

RFC 2046
IETF "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", N. Freed, N. Borenstein, November 1996. (See http://www.ietf.org/rfc/rfc2046.txt.)

RFC 2048
IETF "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", N. Freed, J. Klensin, J. Postel, March 1997. (See http://www.ietf.org/rfc/rfc2048.txt.)

RFC 2119
IETF "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997. (See http://www.ietf.org/rfc/rfc2119.txt.)

RFC 2387
IETF "The MIME Multipart/Related Content-type", E. Levinson, August 1998. (See http://www.ietf.org/rfc/rfc2387.txt.)

RFC 2396
IETF "Uniform Resource Identifiers (URI): Generic Syntax", T. Berners-Lee, R. Fielding, L. Masinter, August 1998. (See http://www.ietf.org/rfc/rfc2396.txt.)

RFC 2616
IETF "Hypertext Transfer Protocol -- HTTP/1.1", R. Fielding, J. Gettys, J. Mogul, H.F. Nielsen, L. Masinter, P.Leach, T. Berners-Lee, June 1999. (See http://www.ietf.org/rfc/rfc2616.txt.)

XML 1.0
W3C Recommendation "Extensible Markup Language (XML) 1.0 (2nd ed)", T. Bray, J. Paoli, C.M. Sperberg-McQueen, E. Maler, October 2000. (See http://www.w3.org/TR/2000/REC-xml-20001006)

XML Infoset
W3C Recommendation "XML Information Set", J. Cowan, R. Tobin, October 2001. (See http://www.w3.org/TR/2001/REC-xml-infoset-20011024/.)

SOAP Part 1
W3C Working Draft "SOAP Version 1.2 Part 1: Messaging Framework", M. Gudgin, M. Hadley, N. Mendelsohn, J-J. Moreau, H. F. Nielsen, June 2002. (See http://www.w3.org/TR/2002/WD-soap12-part1-20020626/.)

SOAP Part 2
W3C Working Draft "SOAP Version 1.2 Part 2: Adjuncts", M. Gudgin, M. Hadley, N. Mendelsohn, J-J. Moreau, H. F. Nielsen, June 2002. (See http://www.w3.org/TR/2002/WD-soap12-part2-20020626/.)

11.2 Informative References

SOAP with attachments
W3C Note "SOAP Messages with Attachments", J. Barton, S. Thatte, H.F. Nielsen, December 2000. (See http://www.w3.org/TR/SOAP-attachments)

WS-Attachments
Internet draft "WS-Attachments", H.F. Nielsen, E. Christensen, J. Farrell, June 2002.

WS-Security
"Web Services Security (WS-Security)", C. Kaler, B. Atkinson, G. Della-Libera, S. Hada, M. Hondo, P. Hallam-Baker, J. Klein, B. LaMacchia, P. Leach, J. Manferdelli, H. Maruyama, A. Nadalin, N. Nagaratman, H. Prafullchandra, J. Shewchuk, D. Simon, April 2002 (See http://www-106.ibm.com/developerworks/webservices/library/ws-secure/.)

A Contributors

E. Christensen, Microsoft

J. Farrell, IBM

B History

B.1 5 december 2002

- Solved 385 by creating section **1.2 Conformance**.

B.2 4 november 2002

- Solved 387 by updating section **2 SOAP Feature Name**.
- Added direct reference to Request-Response MEP in section **5 Attachment Feature properties**
- Solved 391 by incorporating proposal.
- Solved 277 by updating section **5 Attachment Feature properties**. Properties are now named using URIs instead of QNames. This also solves 386.

B.3 31 october 2002

- Solved 392 by adding section **7 Intermediaries**.
- Solved 388 by incorporating proposal.

B.4 31 July 2002

- Incorporated Noah's and Asir's modifications.
- Further modified last sentence of first para of **1 Introduction**.
- In last paragraph of **3 Terminology** changed 'also' to 'informally'.
- Changed last sentence of Note in **6 Implementation** to be reversed.
- Changed att:SOAPMessage value definition in **5.2 Receiving a compound SOAP structure** to be in sync with value definition in **5.1 Sending a compound SOAP structure**.
- Changed last paragraph of **9 Security Considerations** by replacing 'attachment' by 'secondary part', and by some rewording.
- Added some text in fifth paragraph of **1 Introduction** to say that this spec can be used directly in SOAP bindings or indirectly through another specification.

B.5 30 July 2002

- Added 'part' 2 times in the last sentence of first paragraph of abstract.

- Changed Last paragraph of **1 Introduction** as per Noah's suggestion.

- Changed 'may' to 'can' in **2 SOAP Feature Name**.

- Changed third paragraph of **4 Compound SOAP Structure Model** as per Noah's suggestion (added 'semantic').

- Changed fourth paragraph of **4 Compound SOAP Structure Model** as per Noah's suggestion (added 'envelope').

- Changed **6 Implementation** as per Noah's suggestion.

B.6 22 July 2002

- Reformatted spec as a WD.

- Added text in abstract to say that attachment is an informal term for part.

- Removed sentence "a priori with any MEP" from **5 Attachment Feature properties**.

- Added warning in **5 Attachment Feature properties** about potential properties conflicts. Added a figure.

- Removed note from **5.1 Sending a compound SOAP structure**.

- Removed note from **5.1 Sending a compound SOAP structure**.

- Changed NEED NOT to lowercase in **6 Implementation**.

- Added example mentioning MIME multipart in **6 Implementation**.

- Clarified last sentence of last paragraph in **9 Security Considerations** to read as an example.