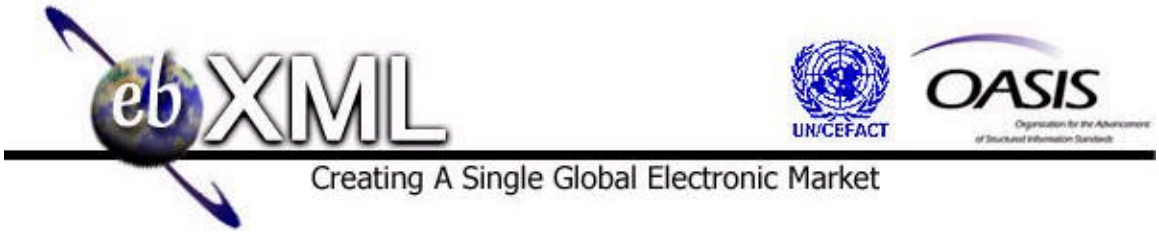


ebXML Transport, Routing & Packaging Messaging Service Specification

Working Draft 10-August-2000

This version:
ebXML Messaging Service Specification v0-1.doc
Latest version:
N/A
Previous version:
ebXML Transport, Routing & Packaging Message Envelope Specification 0.6
ebXML Message Header Specification v0-63.doc
Editor:
Dick Brooks dick@8760.com
David Burdett <david.burdett@commerceone.com>
Authors:
Dick Brooks <dick@8760.com>
Nick Kassem nick.kassem@sun.com
David Burdett <david.burdett@commerceone.com>
John Ibbotson <john_ibbotson@uk.ibm.com>
Christopher Ferris <chris.ferris@sun.com>

Contributors:
The members of the Transport Routing and Packaging Project Team

---

Abstract

This document is a draft proposal whose purpose is to solicit additional input and convey the current state of the ebXML message structure recommendations.

This document defines the envelope and header structure used to encapsulate data for transport between parties. Every attempt has been made to ensure that ebXML requirements as stated in the ebXML Transport, Routing and Packaging: Overview and Requirements, Version 0.96, are addressed. Adherence to industry standards, consideration of existing business-to-business practices and support for small and medium enterprises were key factors influencing the direction of this specification.

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 Key Words for Use in RFC's to Indicate Requirement Levels (RFC 2119).

Terms in *Italics* are defined in the ebXML Glossary of Terms [3]. Terms listed in ***Bold Italics*** represent the element and/or attribute content of the ebXML Message Header.

Status of this Document

This document represents work in progress upon which no reliance should be made.

41 # Table of Contents

97

1 Introduction

This specification defines the ebXML message structure used to encapsulate ebXML message headers and payloads for transport between parties. No assumption or dependency is made relative to transport protocol or type of payload. The specifications contained here are both payload and transport agnostic. The main goal of this specification is to define an enveloping structure and ebXML message header elements used to encapsulate any digitally encoded payload for transport over any data communication mechanism.

1.1 Purpose and Scope

This document provides sufficient detail to develop software for the packaging, exchange and processing of ebXML messages. This document defines the enveloping and ebXML message header structure used to transfer ebXML messages over a data communication mechanism.

Software practitioners MAY use this document in combination with other ebXML specification documents when creating ebXML compliant software.

NOTE: Message security, extensibility, service interface, reliability, and versioning will be addressed in future versions of this document.

1.1.1 Goals

The goals of this specification are to:

- Meet the requirements as specified by the ebXML Transport, Routing and Packaging: Overview and Requirements, Version 0.96 [1]
- Be compatible with other ebXML specifications
- Leverage existing industry standards
- Enable parties to "package" very simple to very complex payloads
- Be payload neutral
- Be transport neutral

1.2 Related Transport, Routing and Packaging Specifications

The following set of related specifications will be delivered in phases:

- **ebXML Messaging Service Specification** (this document) **-** defines the ebXML message structure and is a composite of the following documents:
  - ebXML Transport, Routing & Packaging Message Envelope Specification 0.6
  - ebXML Message Header Specification v0-63.doc
- **ebXML Reliable Messaging Specification** (under development) - defines a method to achieve robust reliable once-only delivery of ebXML message.
- **ebXML Messaging Security and Signature Specification** (under development) - describes the use of IETF/W3C XML Digital Signatures, S/MIME and PGP with ebXML messages.

- **ebXML Messaging Audit Trail Specification** (future development) - defines an audit trail method.

1.3 Specification Structure

This specification is organized around the following main topics:

- **Packaging Specification** - A description of how to package an ebXML message and associated parts. This section includes specifications for the various structures and containers.
- **Message Headers** - A description of the elements REQUIRED.
- **ebXML Message Types** - An introduction to the different sequences in which ebXML messages of different *Message Types* are exchanged

1.4 General Conventions

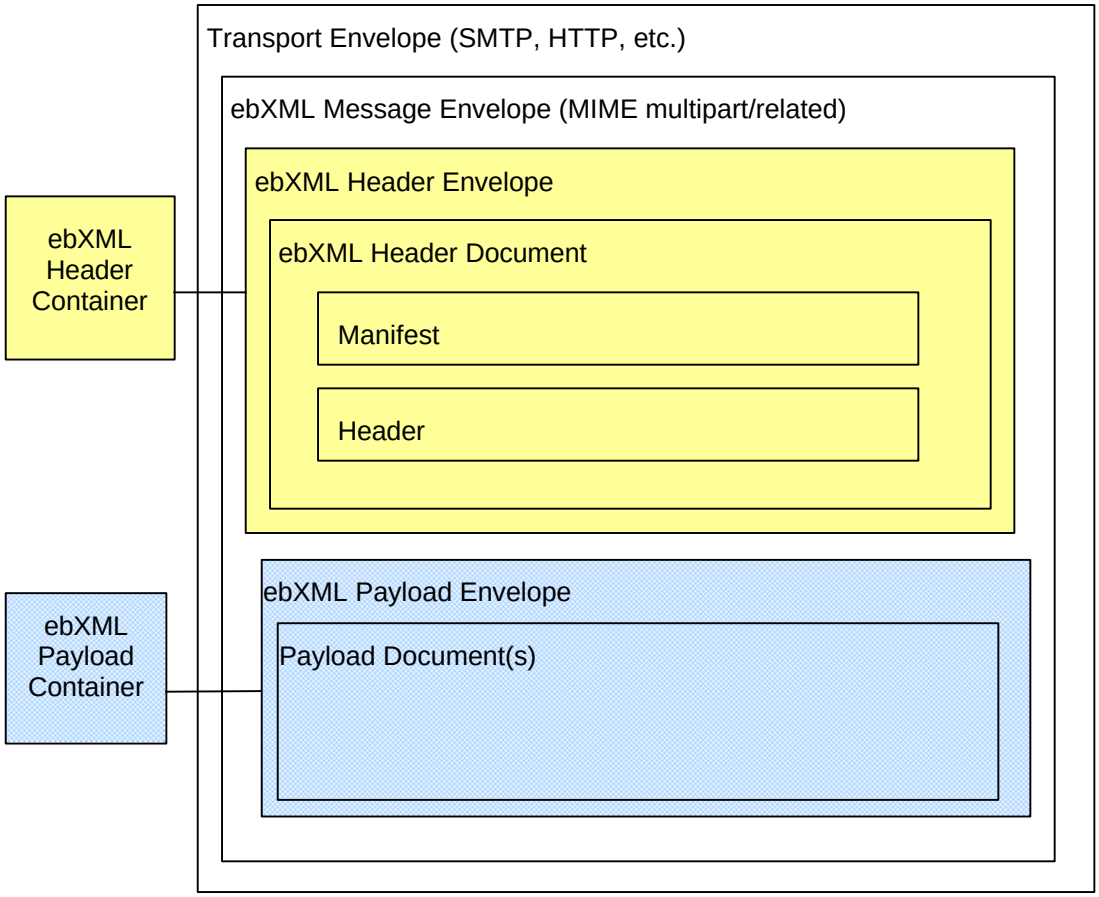
- All headers, attributes and values defined in this specification are to be handled in a case-insensitive fashion.
- For all messages following the ebXML standard, a single message structure is defined, regardless of message type.
- Values associated with MIME header attributes are valid in both quoted and unquoted form. For example, the forms type="ebxml" and type=ebxml are both valid.

2 Packaging Specification

2.1 ebXML Message Structure

An ebXML message consists of:

- a conditional outer Transport Envelope, such as HTTP or SMTP,
- a transport independent Message Envelope, for example MIME multi-part/related, that contains the two main parts of the Message:
  - a Header Container that is used to envelope one ebXML header document, and
  - a Payload Container MUST be used to envelope the real payload of the Message when the payload is present



2.2 Transport Envelope

This document does NOT define any requirements affecting the structure of transport level envelopes. Existing transport protocols, such as SMTP, HTTP, FTP and others, MAY be used to send and receive ebXML compliant messages, without modification. The only requirement ebXML has on the transport envelope is the ability to identify a specific *ebXML handler* to receive incoming ebXML messages.

A transport envelope is only REQUIRED in those cases requiring such structures. In the case of HTTP or SMTP, transport envelopes are REQUIRED. However, a transport envelope is not needed for FTP. Implementers of software to process ebXML messages MUST be aware of transport specific requirements relative to transport envelopes.

2.3 Message Envelope Specifications

The message envelope is used to identify the message as an ebXML compliant structure and encapsulates the header and payload body parts. A message envelope MUST HAVE two MIME headers:

1. Content-Length
2. Content-Type

2.3.1 Content-Type

The multi-part/related Content-Type SHALL be used for all ebXML message envelopes. See Appendix **X** for selection rationale.

The Content-Type header contains four attributes:

1. type
4. boundary

2.3.1.1 type Attribute

The type attribute is used to identify the message envelope as an ebXML compliant structure. There is only one valid value for this attribute: "application/vnd.eb+xml". The following is an example usage of the type attribute:

```
Content-Type: multipart/related; type="application/vnd.eb+xml"
```

2.3.1.2 boundary Attribute

The boundary attribute is used to identify the body part separator used to identify the start and end points of each body part contained in the message. The boundary SHOULD be chosen carefully to insure that it does not occur within the content area of a body part. Example usage of the boundary attribute:

```
Content-Type: multipart/related; type="application/vnd.eb+xml"; version="0.1";
boundary="-------8760"
```

2.3.2 Content-Length

The Content-Length header is a decimal value used to identify the total number of OCTETS contained in all constituent message body parts, including body part boundaries. Example:

```
Content-Length: 9841
```

2.3.3 ebXML Message Envelope Example

An example of a ebXML compliant message envelope appears as follows:

```
Content-Type: multipart/related; type=" application/vnd.eb+xml "boundary="-------8760"
Content-Length: 9841
```

2.4 ebXML Header Container Specifications

The ebXML Header Container is a MIME body part used to encapsulate an ebXML header document. The ebXML header document is described in section **3** of this document. There MUST be one ebXML header document associated with every ebXML Message. The ebXML Header Container consists of a MIME Header portion (referred to as the ebXML Header envelope) and a content portion.

The ebXML Header envelope, consists of three MIME headers:

1. Content-ID
2. Content-Length
3. Content-Type

The ebXML header document within the content portion of the container MAY be enhanced during transport, provided it has not been digitally signed. Any change in the size of the ebXML header document MUST be reflected in Content-Length attribute of the ebXML Message Envelope and ebXML Header envelope.

2.4.1 Content-ID

The Content-ID MIME header identifies this instance of an ebXML message header body part. The value for Content-ID SHOULD be a unique identifier, in accordance with RFC 2045. An example usage follows:

```
Content-ID: ebxmlheader-com-8760-2000-0722-161201-123456789
```

2.4.2 Content-Length

The Content-Length header is a decimal value used to identify the total number of OCTETS contained in all constituent message body parts, including body part boundaries. Example:

```
Content-Length: 4208
```

2.4.3 Content-Type

The Content-Type for an ebXML header is identified with the value "application/vnd.eb+xml". Content-Type SHALL contain two attributes: ***version*** and ***charset***. The charset attribute shall agree with the value passed in the ***encoded*** attribute.

- ***version*** – a value indicating the actual version of the ebXML message structures . There are currently two valid values for version:
    1. “0” indicating a version-less message; all ebXML implementations MUST support version-less messages.
    2. “0.1” indicating the current version of ebXML message structures.

- ***charset*** - a value used to identify the character set used to create the message. The list of valid values can be found at http://www.iana.org/. The default charset value is "iso-8895-1".

An example of this Content-Type is:

```
Content-Type: application/vnd.eb+xml; version="1.0"; charset = "UTF-8"
```

2.4.4 Optional Support for Signed Headers

Implementers are free to support digitally signed ebXML header documents. Digitally signed ebXML headers MUST be identified with the appropriate Content-Type and structure-appropriate for the cryptographic tool used. In the case of S/MIME, the Content-Type MUST contain the correct value and attributes as specified in RFC 2633. In the case of *OpenPGP*, the Content-Type MUST contain the correct values and attributes specified in MIME Security with PGP (RFC 2015).

Implementers MUST follow the guidelines specified in RFC 2633 and RFC 2015 for creating and processing digitally signed objects.

The XML Signature Syntax and Processing [4] MUST be followed when implementers use XML Digital Signatures.

2.4.5 Example of an ebXML Header container

The following represents an example of an ebXML Header envelope and ebXML Header document:

```
Content-ID: ebxmlheader-123      -------------|                              |
Content-Length: 2048                          |   ebXML Header Envelope      |
Content-Type: application/vnd.eb+xml ---------|                              |  ebXML
                                                                             |  Header
<ebXMLHeaderDocument>            -------------|                              |  Container
  <MessageHeader>........                     |   ebXML header Document      |
  </MessageHeader>                            |                              |
</ebXMLHeaderDocument>           -------------|                              |
```

A complete example of the ebXML Header envelope is presented in Appendix **X**.

2.5 Payload Container Specifications

A single Payload Container of an ebXML message MUST be used to envelop the payload(s) when one or more payloads is/are present. The ebXML header document contains a *Message Manifest* that identifies whether a Payload Container is present or not. If the *Message Manifest* of the ebXML header does not contain any entries, the ebXML Payload Container will not be present in the ebXML Message. However, if the Message Manifest of the ebXML header indicates that a payload is present, it will consist of a MIME header portion (referred to as the *ebXML Payload envelope*) and a content portion.

The ebXML Payload envelope, consists of three MIME headers:

1. Content-ID
2. Content-Length
3. Content-Type

ebXML makes no provision nor limits in any way the structure or content of payloads. Payloads MAY contain simple plain text object or complex nested multipart objects. This is the implementer's decision.

2.5.1 Content-ID

The Content-ID MIME header identifies this instance of an ebXML is used to uniquely identify an instance of an ebXML message payload body part. The value for Content-ID SHOULD be a unique identifier, in accordance with Multipurpose Internet Mail Extensions (RFC 2045). An example usage follows:

```
Content-ID: ebxmlpayload-com-8760-2000-0722-161201-123456789
```

2.5.2 Content-Length

The Content-Length header is a decimal value used to identify the total number of OCTETS contained in the content portion of the Payload Container. Example:

```
Content-Length: 5012
```

2.5.3 Content-Type

The Content-Type for an ebXML header is determined by the implementer and is used to identify the type of data contained in the content portion of the Payload Container.

```
Content-Type: application/vnd.eb+xml
```

2.5.4 Optional Support for Signed and Encrypted Payloads

Implementers are free to support digitally signed ebXML payloads. Digitally signed ebXML payloads MUST be identified with the appropriate Content-Type and structure-appropriate for the cryptographic tool used. In the case of S/MIME, the Content-Type MUST contain the correct value and attributes as specified in RFC 2633. In the case of *OpenPGP*, the Content-Type MUST contain the correct values and attributes specified in MIME Security with PGP (RFC 2015).

Implementers MUST follow the guidelines specified in RFC 2633 and RFC 2015 for creating and processing digitally signed objects.

The XML Signature Syntax and Processing [4] MUST be followed when implementers use XML Digital Signatures.

2.5.5 Example of an ebXML Payload Container

The following represents an example of an ebXML Payload envelope and ebXML Payload document:

```
Content-ID: ebxmlpayload-123      -------------|                              |
Content-Length: 4096                           | ebXML Payload Envelope       |
Content-Type: application/xml     -------------|                              |  ebXML
                                               |                              |  Payload
<Invoice>                         -------------|                              |  Container
  <Invoicedata>........                        |   ebXML Payload              |
  </Invoicedata>                               |                              |
</Invoice>                        -------------|                              |
```

A complete example of the ebXML Payload Container is presented in Appendix **X**.

2.6 Message Digest Computation

Parties wishing to ensure the integrity of data exchanged using ebXML standards MAY wish to create a *Message Digest* of the information contained in an ebXML message. In order to ensure consistent usage of a *Message Digest*, it is imperative that all parties agree on the range of data to include when performing a *Message Digest* calculation. This range of data is referred to as the *Domain of Computation*. The *Domain of Computation* defines the beginning and ending boundaries (inclusive) of the information to be processed when calculating a *Message Digest*.

The *Domain of Computation* begins with the first "-" octet of the first boundary following the ebXML Message Envelope. The *Domain of Computation* ends with the last "-" of the final boundary of an ebXML message. All octets between and including these begin and end points *Domain of Calculation*. Parties wishing to create a *Message Digest* of an ebXML message SHOULD apply their hash algorithms over the *Domain of Computation*, as defined.

In the following example, the *Domain of Computation* begins with the first "-" of "--8760" and ends with the last "-" of "--8760--". The hash algorithm used to calculate a *Message Digest* SHOULD process all octets between and including these points.

```
Content-Type: multipart/related; type="ebxml"; version="0.1"; charset="iso-8859-1";
boundary="8760"
Content-Length: 9841
--8760

ebXML header and payload body parts removed for brevity

--8760--
```

3 Message Header

The ebXML Header is a single XML document with a number of principal *Header* elements within it where each *Header* element is a separate XML element. In general, separate header elements are used where:

- different software is likely to be used to generate that header element,
- the structure of the header element might vary independently of the other header elements, or
- the data contained in the header element MAY need to be digitally signed separately from the other header elements.

Using this principle, the following header elements have been identified:

- **Message Manifest** - contains a list of references to he other parts of the Message. This includes references to the documents, which comprise the *Payload* of the *Message.*
- **Message Header** - contains the information REQUIRED by the recipient to process the message. The message originator creates this information to which additional information MAY be added.

The Message *Header* and Message *Manifest* are REQUIRED elements in every *Message*.

NOTE: It MAY prove necessary that a separate document MAY be REQUIRED for header elements not yet defined. Every effort will be made to preclude this possibility.

3.1 Root Element

The root element of the ebXML Message Header document is ***ebXMLHeader***. It is comprised of three attributes and two subordinate elements.

The first attribute is the namespace declaration (***xmlns***) which has a REQUIRED value of "http://www.ebxml.org/namespaces/messageHeader".

The second attribute is the **Version** attribute. This attribute is required. Its purpose is to provide for future versioning capabilities. It has a default value of '1.0'.

The last of the ***ebXMLHeader*** attributes is the ***MessageType*** attribute. Its purpose is to enable ebXML-aware software to distinguish between normal and transport-specific messages, such as acknowledgment and error messages. This is described in more detail in section 4 below.

The ***MessageType*** is an enumeration consisting of three possible values:

- ***Normal*** – an application-generated message, distinct from a message generated by the infrastructure software providing the ebXML messaging service.
- ***Acknowledgment*** – a ebXML Messaging Service-specific acknowledgment message.
- ***Error*** – an ebXML Messaging Service-specific error message.

There are two subordinate elements of the ***ebXMLHeader*** root element:

- ***Manifest –*** identification of the payload contents
- ***Header –*** routing information

The following is a sample ***ebXMLHeader*** document fragment demonstrating the overall structure:

```
<?xml version="1.0"?>
<ebXMLMessageHeader xmlns="http://www.ebxml.org/namespaces/messageHeader"
    Version="1.0" MessageType="Normal">
    <Manifest>...</Manifest>
    <Header>...</Header>
</ebXMLMessageHeader>
```

3.2 Manifest

The required ***Manifest*** element is a composite element consisting of zero or more ***DocumentReference*** elements. Each ***DocumentReference*** element identifies data associated with the message, whether included as part of the message, or remote resources accessible via a URL. The ***Manifest*** SHALL be the first subordinate element in the ***ebXMLMessageHeader***. It identifies the payload document(s) contained in the ebXML message envelope. The purpose of the ***Manifest*** is to make it easier to directly extract a particular document associated with the Message.

3.2.1 DocumentReference

The ***DocumentReference*** element is a composite element consisting of two required subordinate elements as follows:

- ***DocumentLabel –*** a textual description of the document/resource referenced by:
- ***DocumentId –*** a URL of the Content-ID of a MIME body part, as defined in [RFC2111], representing payload data, or a remote URL to some external resource.

The following fragment demonstrates a typical ***Manifest*** for a message with a single payload MIME body part:

```
<Manifest>
    <DocumentReference>
        <DocumentLabel>PurchaseOrder</DocumentLabel>
        <DocumentId>cid:0987654321</DocumentId>
    </DocumentReference>
</Manifest>
```

3.3 Header

The ***Header*** element immediately follows the ***Manifest*** element. It is required in all ***ebXMLMessageHeader*** documents. The ***Header*** element is a composite element comprised of the following required subordinate elements:

- ***From*** – the logical address of the sender of the message.
- ***To*** – the logical address of the intended recipient of the message.
- ***TPAInfo*** – a composite set of information which relates to the *Trading Partner Agreement* under which the message is governed
- ***MessageData*** – a composite set of information which uniquely identifies the *Message*
- ***ReliableMessagingInfo*** - information which identifies the degree of reliability with which the message SHOULD be delivered

The following fragment demonstrates the structure of the ***Header*** element of the ***ebXMLMessageHeader*** document:

```
<Header>
    <From>...</From>
    <To>...</To>
    <TPAInfo>...</TPAInfo>
    <MessageData>...</MessageData>
    <ReliableMessagingInfo>...</ReliableMessagingInfo>
</Header>
```

3.3.1 From and To

The ***From*** element identifies the *Party* which originated the message. It is a logical identifier, which MAY take the form of a URN. An example of this would be a DUNS number. The ***From*** element consists of a ***PartyId*** element.

The ***To*** element identifies the intended recipient of the message. As with ***From***, it is a logical identifier which is comprised of a ***PartyId*** element.

The **PartyId** element has a single attribute; ***context*** and a text value. The purpose of the context attribute is to provide a context for the text value of the ***PartyId*** element. The following fragment demonstrates usage of the ***From*** and ***To*** elements of the ***ebXMLMessageHeader***.

```
<From>
    <PartyId context="DUNS">12345</PartyId>
</From>
<To>
    <PartyId context="DUNS">54321</PartyId>
</To>
```

3.3.2 TPAInfo

The ***TPAInfo*** element follows the ***From*** and ***To*** elements in the ***Header*** element structure. The ***TPAInfo*** element is a composite set of information which relates to the *Trading Partner Agreement* under which the message is governed. The ***TPAInfo*** element has four subordinate elements as follows:

- ***TPAId*** – a URI which identifies the *Trading Partner Agreement* which governs the processing of the message
- ***ConversationId*** – a URI which identifies the conversation instance of the *Trading Partner Agreement.*
- ***ServiceInterface*** – Identifies the Service Interface that SHOULD act on the payload in the message. It is unique within the domain of the ***Party*** to which the message is being sent. URN's MAY be considered suitable for the element content.
- ***Action*** – Identifies a process within a Service Interface, which processes the Message. ***Action*** SHALL be unique within the Service Interface in which it is defined.

The following example fragment demonstrates the usage of the ***TPAInfo*** element.

```
<TPAInfo>
         <TPAId context = "tpadb">12345678</TPAId>
         <ConversationId context = "tpadb">987654321</ConversationId>
         <ServiceInterface>QuoteToCollect</ServiceInterface>
         <Action>NewPurchaseOrder</Action>
</TPAInfo>
```

3.3.3 MessageData

The required ***MessageData*** element follows the ***TPAInfo*** element. The purpose of the ***MessageData*** element is to provide a means of uniquely identifying the message. It is a composite element which contains the following three subordinate elements:

- ***MessageId*** – a globally unique identifier for the message conforming to [RFC2111].
- ***TimeStamp*** – a value representing the time that the message header was created conforming to [ISO-8601]. The format of CCYYMMDDTHHMMSS.SSSZ is used. This time format is Coordinated Universal Time (UTC).
- ***RefToMessageId*** – a globally unique identifier which relates the current message to a previous message. It MUST contain the value of the ***MessageId*** of the related message or the value "Not Applicable".

The following example demonstrates the usage of the ***MessageData*** element.

```
<MessageData>
     <MessageId>UUID</MessageId>
     <TimeStamp>20000725T121905.000Z</TimeStamp>
     <RefToMessageId>UUID</RefToMessageId>
</MessageData>
```

3.3.4 ReliableMessagingInfo

The last element of the ***ebXMLMessageHeader*** is the ***ReliableMessagingInfo*** element. This element identifies the degree of reliability with which the message will be delivered. This element has a single subordinate attribute, ***DeliverySemantics.*** This attribute is an enumeration, which MAY have one of the following values:

- “AtMostOnce” – reliable messaging semantics, which specifies that a given message will be received by the Service Interface handler no more than once.
- “Unspecified” – reliable delivery semantics are not specified.

4 Security Considerations

Implementers SHOULD examine carefully the security features of each transport. In the case of HTTP, Implementers are encouraged to use Realm Security, using basic authentication for access controls and SSL to protect sensitive information.

Users of E-Mail based solution SHOULD ensure that anti-spamming services are in place and filtering is used to prevent unauthorized access to E-Commerce Servers.

5 References

[1] [Overview] ebXML Transport, Routing and Packaging: Overview and Requirements, Version 0.96, Published 25 May 2000

[2] XMTP - Extensible Mail Transport Protocol http://www.openhealth.org/documents/xmtp.htm

[3] ebXML Glossary, see ebXML Project Team Home Page

[4] XML Signature Syntax and Processing

[ISO 8601] International Standards Organization Ref. ISO 8601 Second Edition, Published 1997

[RFC 2111] IETF Request For Comments 2111 E. Levinson, Published March 1997

6 Acknowledgments

The author's wish to acknowledge the support of the members of the Transport, Routing and Packaging Project Team who contributed ideas to this specification by the group's discussion e-mail list, on conference calls and during face-to-face meetings.

Ralph Berwanger – bTrade.com

Jonathan Borden - Author of XMTP

Jon Bosak - Sun

Doug Bunting - Ariba

David Burdett - Commerce One

Lawrence Ding - WorldSpan

Rik Drummond - Drummond Group

Christopher Ferris - Sun

Ian Jones - BT

Henry Lowe - OMG

Jim McCarthy - webXI

Bob Miller - GEIS

Dale Moberg - Sterling Commerce

Kathy Spector – Extricity

Martha Warfelt - DaimlerChrysler

Prasad Yendluri - Vitria

7 Authors' Address

Contact information for the Authors of this specification follow.

David Burdett
Commerce One Inc
4400 Rosewood Drive 3rd Fl, Bldg 4,
Pleasanton, CA 94588,
USA
Telephone: (925) 520-4422 or (650) 623-2888
E-mail: david.burdett@commerceone.com

John Ibbotson
IBM UK Ltd
Hursley Park
Winchester
SO21 2JN
United Kingdom
Telephone: +44 (1962) 815188
E-mail: john_ibbotson@uk.ibm.com

Christopher Ferris
Sun Microsystems, Inc.
One Network Drive
Burlington, Ma 01803-0903
USA
Telephone: (781) 442-3063
E-mail: chris.ferris@sun.com

Dick Brooks
Group 8760
110 12th Street North
Suite F103
Birmingham, Alabama 35203
USA
Telephone: (205) 250-8053
E-mail: dick@8760.com

Nicholas Kassem
Java Software, Sun Microsystems
901 San Antonio Road, MS CUP02-201
Palo Alto, CA 94303-4900

605 Telephone: (408) 863-3535
606 E-mail: Nick.Kassem@eng.sun.com

607

Appendix A Schemas and DTD Definitions

The following are definitions for validation of the ebXML message header structure.

A.1 XML Header DTD

```
<?xml version ="1.0"?>
<schema xmlns = "http://www.w3.org/1999/XMLSchema">
<!ELEMENT ebXMLHeader  (Manifest , Header )>
<!ATTLIST ebXMLHeader  Version     CDATA  #FIXED '1.0'
                    MessageType CDATA  #FIXED 'Normal' >
<!ELEMENT Manifest  (DocumentReference )+>
<!ELEMENT DocumentReference  (DocumentLabel , DocumentId )>
<!ELEMENT DocumentLabel  (#PCDATA )>
<!ATTLIST DocumentLabel  e-dtype NMTOKEN  #FIXED 'string' >
<!ELEMENT DocumentId  (#PCDATA )>
<!ATTLIST DocumentId  e-dtype NMTOKEN  #FIXED 'uri' >
<!ELEMENT Header  (From , To , TPAInfo , MessageData , ReliableMessagingInfo )>

<!ELEMENT ServiceInterface  (#PCDATA )>
<!ATTLIST ServiceInterface  e-dtype NMTOKEN  #FIXED 'string' >
<!ELEMENT Action  (#PCDATA )>
<!ATTLIST Action  e-dtype NMTOKEN  #FIXED 'string' >
<!ELEMENT TPAId  (#PCDATA )>
<!ATTLIST TPAId  context CDATA    'Undefined'
              e-dtype NMTOKEN  #FIXED 'uri' >
<!ELEMENT ConversationId  (#PCDATA )>
<!ATTLIST ConversationId  context CDATA    'Undefined'
                    e-dtype NMTOKEN  #FIXED 'uri' >
<!ELEMENT MessageData  (MessageId , TimeStamp , RefToMessageId )>
<!ELEMENT RefToMessageId  (#PCDATA )>
<!ATTLIST RefToMessageId  e-dtype NMTOKEN  #FIXED 'uuid' >
<!ELEMENT TimeStamp  (#PCDATA )>
<!ATTLIST TimeStamp  e-dtype NMTOKEN  #FIXED 'dateTime' >
<!ELEMENT MessageId  (#PCDATA )>
<!ATTLIST MessageId  e-dtype NMTOKEN  #FIXED 'uuid' >
<!ELEMENT From  (PartyId )>
<!ELEMENT To  (PartyId )>
<!ELEMENT PartyId  (#PCDATA )>
<!ATTLIST PartyId  context CDATA    'Undefined'
              e-dtype NMTOKEN  #FIXED 'uri' >
<!ELEMENT ReliableMessagingInfo EMPTY>
<!ATTLIST ReliableMessagingInfo  DeliverySemantics  (AtMostOnce | Unspecified )  #FIXED
'Unspecified' >
<!ELEMENT TPAInfo  (TPAId , ConversationId , ServiceInterface , Action )>
```

A.2 XML Header Schema Definition

```
<?xml version ="1.0"?>
<schema xmlns = "http://www.w3.org/1999/XMLSchema">
    <element name = "ebXMLHeader">
        <complexType content = "elementOnly">
            <sequence>
                <element ref = "Manifest"/>
                <element ref = "Header"/>
            </sequence>
            <attribute name = "Version" use = "fixed" value = "1.0" type = "string"/>
            <attribute name = "MessageType" use = "fixed" value = "Normal" type = "string"/>
        </complexType>
    </element>

    <element name = "Manifest">
        <complexType content = "elementOnly">
            <sequence minOccurs = "1" maxOccurs = "unbounded">
                <element ref = "DocumentReference"/>
            </sequence>
        </complexType>
    </element>

    <element name = "DocumentReference">
        <complexType content = "elementOnly">
            <sequence minOccurs = "1" maxOccurs = "unbounded">
                <element ref = "DocumentLabel"/>
                <element ref = "DocumentId"/>
            </sequence>
        </complexType>
    </element>

    <element name = "DocumentLabel" type = "string">
    </element>

    <element name = "DocumentId" type = "uri">
    </element>

    <element name = "Header">
        <complexType content = "elementOnly">
            <sequence>
                <element ref = "From"/>
                <element ref = "To"/>
                <element ref = "TPA"/>
                <element ref = "MessageData"/>
                <element ref = "ReliableMessagingInfo"/>
            </sequence>
        </complexType>
    </element>

    <element name = "BusinessServiceInterface" type = "string">
    </element>
```

```
<element name = "Action" type = "string"/>
<element name = "TPAId">
    <complexType base = "uri" content = "textOnly">
        <attribute name = "context" use = "default" value = "Undefined" type = "string"/>
    </complexType>
</element>

<element name = "ConversationId">
    <complexType base = "uri" content = "textOnly">
        <attribute name = "context" use = "default" value = "Undefined" type = "string"/>
    </complexType>
</element>

<element name = "MessageData">
    <complexType content = "elementOnly">
        <sequence>
            <element ref = "MessageId"/>
            <element ref = "TimeStamp"/>
            <element ref = "RefToMessageId"/>
        </sequence>
    </complexType>
</element>

<element name = "RefToMessageId" type = "uuid">
</element>

<element name = "TimeStamp" type = "dateTime">
</element>

<element name = "MessageId" type = "uuid">
</element>

<element name = "From">
    <complexType content = "elementOnly">
        <sequence>
            <element ref = "PartyId"/>
        </sequence>
    </complexType>
</element>

<element name = "To">
    <complexType content = "elementOnly">
        <sequence>
            <element ref = "PartyId"/>
        </sequence>
    </complexType>
</element>

<element name = "PartyId">
    <complexType base = "uri" content = "textOnly">
        <attribute name = "context" use = "default" value = "Undefined" type = "string"/>
    </complexType>
</element>
```

```
    <element name = "ReliableMessagingInfo">
        <complexType content = "empty">
            <attribute name = "DeliverySemantics" use = "fixed" value = "Unspecified">
                <simpleType base = "ENUMERATION">
                    <enumeration value = "AtMostOnce"/>
                    <enumeration value = "Unspecified"/>
                </simpleType>
            </attribute>
        </complexType>
    </element>

    <element name = "TPAInfo">
        <complexType content = "elementOnly">
            <sequence>
                <element ref = "TPAId"/>
                <element ref = "ConversationId"/>
                <element ref = "BusinessServiceInterface"/>
                <element ref = "Action"/>
            </sequence>
        </complexType>
    </element>

</schema>
```

Appendix B Examples

The following are complete examples of ebXML messages showing the structure as defined in this specification.

B.1 Complete Example of an ebXML Message enveloped using multipart/related Content-Type sent via HTTP POST

```
POST /ebxmlhandler HTTP/1.1
Accept: multipart/related
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Group 8760 InsideAgent
Host: localhost:9090
Connection: Keep-Alive
Content-Type: multipart/related; type=application/vnd.eb+xml; version=0.1;
boundary=----------------------------7d02a82e5f8
Content-Length: 9293

------------------------------7d02a82e5f8
Content-ID: ebxmlheader-9981
Content-Length: 211
Content-Type: application/vnd.eb+xml; charset="UTF-8";

<?xml version="1.0" encoding="UTF-8"?>
<ebXMLMessageHeader xmlns='http://www.xml.org/ebXMLStds/ebXMLMessageHeaderv1'>
  <Version>1.0</Version>
       <MessageType>Request</MessageType>
  <ServiceType>Payroll</ServiceType>
  <Intent>RecordCommission</Intent>
</ebXMLMessageHeader>
------------------------------7d02a82e5f8
Content-ID: ebxmlpayload-9981
Content-Length: 7517
Content-Type: text/xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v2.5  - http://www.xmlspy.com -->
<HITISMessage xmlns="" Version="1.0">
  <Header OriginalBodyRequested="false" ImmediateResponseRequired="true">
       <FromURI>http://www.pms.com/HITISInterface</FromURI>
       <ToURI>http://www.crs.com/HITISInterface</ToURI>
       <ReplyToURI>http://www.pms.com/HITISInterface</ReplyToURI>
       <MessageID>1234567890</MessageID>
       <OriginalMessageID>1234567890</OriginalMessageID>
       <TimeStamp>1999-11-10T10:23:44</TimeStamp>
       <Token>1234-567-8901</Token>
       <!--Token to be assigned in response to HITISRegister-->
  </Header>
  <Body>
       <HITISOperation OperationName="CommissionEventsUpdate">
            <CommissionEvents>
                 <CommissionEvent>
                      <ConfirmationID>18097YZ</ConfirmationID>
                      <ConfirmationOriginatorCode>DBZ223</ConfirmationOriginatorCode>
                      <CommissionOriginatorCode>3457YTXV</CommissionOriginatorCode>
                      <ReservationID>098787818097YZ</ReservationID>
                      <HotelReference>
                           <ChainCode>HI234</ChainCode>
                           <HotelCode>1234STL</HotelCode>
                      </HotelReference>
                      <OriginalBookingDate>19991223T17:53:22</OriginalBookingDate>
                      <StayDateRange>
```

```
                        <StartInstant>20000122</StartInstant>
                        <Duration>00000003T000000</Duration>
                    </StayDateRange>
                    <GuestNames>
                        <NameInfo>
                            <NamePrefix>Mr.</NamePrefix>
                            <NameFirst>John</NameFirst>
                            <NameMiddle>Q.</NameMiddle>
                            <NameSur>jones</NameSur>
                            <NameSuffix>Jr.</NameSuffix>
                            <NameTitle>Professor</NameTitle>
                            <NameOrdered>JohnJones</NameOrdered>
                        </NameInfo>
                        <NameInfo>
                            <NamePrefix>Mrs.</NamePrefix>
                            <NameFirst>Sally</NameFirst>
                            <NameMiddle>T.</NameMiddle>
                            <NameSur>Jones</NameSur>
                            <NameSuffix/>
                            <NameTitle/>
                            <NameOrdered>SallyJones</NameOrdered>
                        </NameInfo>
                    </GuestNames>
                    <ProfileCertification CertificationType="ARC">
                        <CertificationID>67TR901-AZ</CertificationID>
                    </ProfileCertification>
                    <ProfileReference>
                        <!--Profile to be inserted as a reusable component-->
                        <Profile/>
                    </ProfileReference>
                    <Commissions>
                        <Commission CommissionStatusType="Full">
                            <CommissionableAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>185.00</Amount>
                                </Currency>
                            </CommissionableAmount>
                            <PrepaidAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>12.00</Amount>
                                </Currency>
                            </PrepaidAmount>
                            <CommissionPercent>0.0525</CommissionPercent>
                            <FlatCommission>not applicable<Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>00.00</Amount>
                                </Currency>
                            </FlatCommission>
                            <Comment>Default percentage commission agreement</Comment>
                            <CommissionReasonCode>7930</CommissionReasonCode>
                            <BillToID>HOTEL7890</BillToID>
                            <HotelReference>
                                <ChainCode>HI234</ChainCode>
                                <HotelCode>1234STL</HotelCode>
                            </HotelReference>
                        </Commission>
                        <Commission CommissionStatusType="Partial">
                            <CommissionableAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>185.00</Amount>
                                </Currency>
                            </CommissionableAmount>
                            <PrepaidAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>00.00</Amount>
                                </Currency>
                            </PrepaidAmount>
                            <Comment>This commission per agreement with Travel Agents,
Inc.</Comment>
                            <CommissionPercent>00.00</CommissionPercent>
                            <FlatCommission>
```

```
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>10.00</Amount>
                                </Currency>
                            </FlatCommission>
                            <CommissionReasonCode>7930</CommissionReasonCode>
                            <BillToID>HOTEL7890</BillToID>
                            <HotelReference>
                                <ChainCode>HI234</ChainCode>
                                <HotelCode>1234STL</HotelCode>
                            </HotelReference>
                        </Commission>
                    </Commissions>
                </CommissionEvent>
                <CommissionEvent>
                    <ConfirmationID/>
                    <ConfirmationOriginatorCode/>
                    <CommissionOriginatorCode>3457YTXV</CommissionOriginatorCode>
                    <ReservationID>09878783276XY</ReservationID>
                    <HotelReference>
                        <ChainCode>BASS123</ChainCode>
                        <HotelCode>1234STL</HotelCode>
                    </HotelReference>
                    <OriginalBookingDate>19991223T17:53:22</OriginalBookingDate>
                    <StayDateRange>
                        <StartInstant>20000122</StartInstant>
                        <Duration>00000003T000000</Duration>
                    </StayDateRange>
                    <GuestNames>
                        <NameInfo>
                            <NamePrefix>Mr.</NamePrefix>
                            <NameFirst>Kevin</NameFirst>
                            <NameMiddle>R.</NameMiddle>
                            <NameSur>Smithson</NameSur>
                            <NameSuffix>Jr.</NameSuffix>
                            <NameTitle>Professor</NameTitle>
                            <NameOrdered> Kevin Smithson</NameOrdered>
                        </NameInfo>
                        <NameInfo>
                            <NamePrefix>Miss</NamePrefix>
                            <NameFirst>Mary</NameFirst>
                            <NameMiddle>T.</NameMiddle>
                            <NameSur>Smithson</NameSur>
                            <NameSuffix>esq.</NameSuffix>
                            <NameTitle>Professor</NameTitle>
                            <NameOrdered> MarySmithson</NameOrdered>
                        </NameInfo>
                    </GuestNames>
                    <ProfileCertification CertificationType="ARC">
                        <CertificationID>67TR901-AZ</CertificationID>
                    </ProfileCertification>
                    <ProfileReference>
                        <Profile/>
                    </ProfileReference>
                    <Commissions>
                        <Commission CommissionStatusType="Full">
                            <CommissionableAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>185.00</Amount>
                                </Currency>
                            </CommissionableAmount>
                            <PrepaidAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>12.00</Amount>
                                </Currency>
                            </PrepaidAmount>
                            <CommissionPercent>0.0525</CommissionPercent>
                            <FlatCommission>not applicable<Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>00.00</Amount>
                                </Currency>
                            </FlatCommission>
                            <Comment>Default percentage commission agreement</Comment>
```

```
                              <CommissionReasonCode>7930</CommissionReasonCode>
                              <BillToID>HOTEL7890</BillToID>
                              <HotelReference>
                                   <ChainCode>HI234</ChainCode>
                                   <HotelCode>1234STL</HotelCode>
                              </HotelReference>
                         </Commission>
                         <Commission CommissionStatusType="Partial">
                              <CommissionableAmount>
                                   <Currency>
                                        <CurrencyCode>USD</CurrencyCode>
                                        <Amount>185.00</Amount>
                                   </Currency>
                              </CommissionableAmount>
                              <PrepaidAmount>
                                   <Currency>
                                        <CurrencyCode>USD</CurrencyCode>
                                        <Amount>00.00</Amount>
                                   </Currency>
                              </PrepaidAmount>
                              <Comment>Flat commission per agreement with TA</Comment>
                              <CommissionPercent>00.00</CommissionPercent>
                              <FlatCommission>
                                   <Currency>
                                        <CurrencyCode>USD</CurrencyCode>
                                        <Amount>10.00</Amount>
                                   </Currency>
                              </FlatCommission>
                              <CommissionReasonCode>7930</CommissionReasonCode>
                              <BillToID>HOTEL7890</BillToID>
                              <HotelReference>
                                   <ChainCode>HI234</ChainCode>
                                   <HotelCode>1234STL</HotelCode>
                              </HotelReference>
                         </Commission>
                    </Commissions>
               </CommissionEvent>
          </CommissionEvents>
     </HITISOperation>
  </Body>
</HITISMessage>

----------------------------7d02a82e5f8--
```

B.2 Complete Example of an ebXML Message enveloped using multipart/related Content-Type sent via SMTP

The default Content-transfer-encoding type of 7BIT is being used in this message.

```
From dick@8760.com Sun May  7 17:01:14 2000
Received: from granger.mail.mindspring.net by alpha2000.tech-comm.com;
(8.8.5/1.1.8.2/05Jun95-1217PM)
  id RAA32702; Sun, 7 May 2000 17:01:13 -0500 (CDT)
Received: from gamma (user-33qt10l.dialup.mindspring.com [199.174.132.21])
  by granger.mail.mindspring.net (8.9.3/8.8.5) with SMTP id SAA11942
  for <ebxmlhandler@8760.com>; Sun, 7 May 2000 18:11:14 -0400 (EDT)
From: "Dick Brooks (E)" <dick@8760.com>
To: <ebxmlhandler@8760.com>
Subject: OTA Commission Event
Date: Sun, 7 May 2000 17:07:38 -0500
Message-ID: <NDBBIOBLMLCDOHCHIKMGKEEIDAAA.dick@8760.com>
MIME-Version: 1.0
X-Priority: 3 (Normal)
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook IMO, Build 9.0.2416 (9.0.2910.0)
Importance: Normal
X-MimeOLE: Produced By Microsoft MimeOLE V5.00.2314.1300
Content-Length: 8081
Content-Type: multipart/related; type="application/vnd.eb+xml"; version="0.1";
charset="iso-8859-1"; boundary="----=_NextPart_000_0005_01BFB846.BF7FABA0"
```

```
------=_NextPart_000_0005_01BFB846.BF7FABA0
Content-Type: application/vnd.eb+xml
Content-ID: ebxmlheader-9000
Content-Length: 272

<?xml version="1.0" encoding="UTF-8"?>
<ebXMLMessageHeader xmlns='http://www.xml.org/ebXMLStds/ebXMLMessageHeaderv1'>
<Version>1.0</Version>
<MessageType>Request</MessageType>
<ServiceType>Payroll</ServiceType>
<Intent>RecordCommission</Intent>
</ebXMLMessageHeader>
------=_NextPart_000_0005_01BFB846.BF7FABA0
Content-Type: text/xml
Content-ID: ebxmlpayload-9000
Content-Length: 7515

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v2.5  - http://www.xmlspy.com -->
<HITISMessage xmlns="" Version="1.0">
  <Header OriginalBodyRequested="false" ImmediateResponseRequired="true">
       <FromURI>http://www.pms.com/HITISInterface</FromURI>
       <ToURI>http://www.crs.com/HITISInterface</ToURI>
       <ReplyToURI>http://www.pms.com/HITISInterface</ReplyToURI>
       <MessageID>1234567890</MessageID>
       <OriginalMessageID>1234567890</OriginalMessageID>
       <TimeStamp>1999-11-10T10:23:44</TimeStamp>
       <Token>1234-567-8901</Token>
       <!--Token to be assigned in response to HITISRegister-->
  </Header>
  <Body>
       <HITISOperation OperationName="CommissionEventsUpdate">
           <CommissionEvents>
               <CommissionEvent>
                    <ConfirmationID>18097YZ</ConfirmationID>
                    <ConfirmationOriginatorCode>DBZ223</ConfirmationOriginatorCode>
                    <CommissionOriginatorCode>3457YTXV</CommissionOriginatorCode>
                    <ReservationID>098787818097YZ</ReservationID>
                    <HotelReference>
                         <ChainCode>HI234</ChainCode>
                         <HotelCode>1234STL</HotelCode>
                    </HotelReference>
                    <OriginalBookingDate>19991223T17:53:22</OriginalBookingDate>
                    <StayDateRange>
                         <StartInstant>20000122</StartInstant>
                         <Duration>00000003T000000</Duration>
                    </StayDateRange>
                    <GuestNames>
                         <NameInfo>
                              <NamePrefix>Mr.</NamePrefix>
                              <NameFirst>John</NameFirst>
                              <NameMiddle>Q.</NameMiddle>
                              <NameSur>jones</NameSur>
                              <NameSuffix>Jr.</NameSuffix>
                              <NameTitle>Professor</NameTitle>
                              <NameOrdered>JohnJones</NameOrdered>
                         </NameInfo>
                         <NameInfo>
                              <NamePrefix>Mrs.</NamePrefix>
                              <NameFirst>Sally</NameFirst>
                              <NameMiddle>T.</NameMiddle>
                              <NameSur>Jones</NameSur>
                              <NameSuffix/>
                              <NameTitle/>
                              <NameOrdered>SallyJones</NameOrdered>
                         </NameInfo>
                    </GuestNames>
                    <ProfileCertification CertificationType="ARC">
                         <CertificationID>67TR901-AZ</CertificationID>
                    </ProfileCertification>
                    <ProfileReference>
                         <!--Profile to be inserted as a reusable component-->
                         <Profile/>
                    </ProfileReference>
```

```
                    <Commissions>
                        <Commission CommissionStatusType="Full">
                            <CommissionableAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>185.00</Amount>
                                </Currency>
                            </CommissionableAmount>
                            <PrepaidAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>12.00</Amount>
                                </Currency>
                            </PrepaidAmount>
                            <CommissionPercent>0.0525</CommissionPercent>
                            <FlatCommission>not applicable<Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>00.00</Amount>
                                </Currency>
                            </FlatCommission>
                            <Comment>Default percentage commission agreement</Comment>
                            <CommissionReasonCode>7930</CommissionReasonCode>
                            <BillToID>HOTEL7890</BillToID>
                            <HotelReference>
                                <ChainCode>HI234</ChainCode>
                                <HotelCode>1234STL</HotelCode>
                            </HotelReference>
                        </Commission>
                        <Commission CommissionStatusType="Partial">
                            <CommissionableAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>185.00</Amount>
                                </Currency>
                            </CommissionableAmount>
                            <PrepaidAmount>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>00.00</Amount>
                                </Currency>
                            </PrepaidAmount>
                            <Comment>This commission per agreement with Travel Agents,
Inc.</Comment>
                            <CommissionPercent>00.00</CommissionPercent>
                            <FlatCommission>
                                <Currency>
                                    <CurrencyCode>USD</CurrencyCode>
                                    <Amount>10.00</Amount>
                                </Currency>
                            </FlatCommission>
                            <CommissionReasonCode>7930</CommissionReasonCode>
                            <BillToID>HOTEL7890</BillToID>
                            <HotelReference>
                                <ChainCode>HI234</ChainCode>
                                <HotelCode>1234STL</HotelCode>
                            </HotelReference>
                        </Commission>
                    </Commissions>
                </CommissionEvent>
                <CommissionEvent>
                    <ConfirmationID/>
                    <ConfirmationOriginatorCode/>
                    <CommissionOriginatorCode>3457YTXV</CommissionOriginatorCode>
                    <ReservationID>09878783276XY</ReservationID>
                    <HotelReference>
                        <ChainCode>BASS123</ChainCode>
                        <HotelCode>1234STL</HotelCode>
                    </HotelReference>
                    <OriginalBookingDate>19991223T17:53:22</OriginalBookingDate>
                    <StayDateRange>
                        <StartInstant>20000122</StartInstant>
                        <Duration>00000003T000000</Duration>
                    </StayDateRange>
                    <GuestNames>
                        <NameInfo>
```

```
                    <NamePrefix>Mr.</NamePrefix>
                    <NameFirst>Kevin</NameFirst>
                    <NameMiddle>R.</NameMiddle>
                    <NameSur>Smithson</NameSur>
                    <NameSuffix>Jr.</NameSuffix>
                    <NameTitle>Professor</NameTitle>
                    <NameOrdered> Kevin Smithson</NameOrdered>
                </NameInfo>
                <NameInfo>
                    <NamePrefix>Miss</NamePrefix>
                    <NameFirst>Mary</NameFirst>
                    <NameMiddle>T.</NameMiddle>
                    <NameSur>Smithson</NameSur>
                    <NameSuffix>esq.</NameSuffix>
                    <NameTitle>Professor</NameTitle>
                    <NameOrdered> MarySmithson</NameOrdered>
                </NameInfo>
            </GuestNames>
            <ProfileCertification CertificationType="ARC">
                <CertificationID>67TR901-AZ</CertificationID>
            </ProfileCertification>
            <ProfileReference>
                <Profile/>
            </ProfileReference>
            <Commissions>
                <Commission CommissionStatusType="Full">
                    <CommissionableAmount>
                        <Currency>
                            <CurrencyCode>USD</CurrencyCode>
                            <Amount>185.00</Amount>
                        </Currency>
                    </CommissionableAmount>
                    <PrepaidAmount>
                        <Currency>
                            <CurrencyCode>USD</CurrencyCode>
                            <Amount>12.00</Amount>
                        </Currency>
                    </PrepaidAmount>
                    <CommissionPercent>0.0525</CommissionPercent>
                    <FlatCommission>not applicable<Currency>
                            <CurrencyCode>USD</CurrencyCode>
                            <Amount>00.00</Amount>
                        </Currency>
                    </FlatCommission>
                    <Comment>Default percentage commission agreement</Comment>
                    <CommissionReasonCode>7930</CommissionReasonCode>
                    <BillToID>HOTEL7890</BillToID>
                    <HotelReference>
                        <ChainCode>HI234</ChainCode>
                        <HotelCode>1234STL</HotelCode>
                    </HotelReference>
                </Commission>
                <Commission CommissionStatusType="Partial">
                    <CommissionableAmount>
                        <Currency>
                            <CurrencyCode>USD</CurrencyCode>
                            <Amount>185.00</Amount>
                        </Currency>
                    </CommissionableAmount>
                    <PrepaidAmount>
                        <Currency>
                            <CurrencyCode>USD</CurrencyCode>
                            <Amount>00.00</Amount>
                        </Currency>
                    </PrepaidAmount>
                    <Comment>Flat commission per agreement with TA</Comment>
                    <CommissionPercent>00.00</CommissionPercent>
                    <FlatCommission>
                        <Currency>
                            <CurrencyCode>USD</CurrencyCode>
                            <Amount>10.00</Amount>
                        </Currency>
                    </FlatCommission>
                    <CommissionReasonCode>7930</CommissionReasonCode>
                    <BillToID>HOTEL7890</BillToID>
```

```
                                <HotelReference>
                                    <ChainCode>HI234</ChainCode>
                                    <HotelCode>1234STL</HotelCode>
                                </HotelReference>
                            </Commission>
                        </Commissions>
                    </CommissionEvent>
                </CommissionEvents>
            </HITISOperation>
    </Body>
</HITISMessage>
------=_NextPart_000_0005_01BFB846.BF7FABA0--
```

Appendix C Candidate Packaging Technologies and Selection Process

The packaging sub-group began its investigation of packaging technologies by identifying the technologies currently used for business-to-business message exchange or were being developed for this purpose. The following packaging technologies were identified:

- MIME - currently in use by companies exchanging business transactions using E-mail and HTTP
- XML - currently used by RosettaNet and Microsoft (BizTalk and SOAP) and others

C.1 Selection Process

Each candidate technology was evaluated based on its ability to meet the requirements listed in the section titled "Packaging and other Requirements" in this document. When necessary, specific parties were contacted to provide details describing how a technology was being used to meet specific requirements. The following parties were contacted to provide expert insight:

- Microsoft - David Turner, regarding use of XML packaging in BizTalk
- Develop Mentor - Don Box, regarding use of XML packaging in SOAP
- Vitria - Prasad Yendluri, regarding use of XML packaging in RosettaNet
- Jonathan Borden - author of XMTP [3], an XML to MIME transformation tool

The packaging sub-group considered the inputs of people from the ebXML Transport mailing list as well as the parties listed above, before making a selection.

C.2 MIME

Multipurpose Internet Mail Extensions (MIME) is an international standard created by the Internet Engineering Task Force. It has been implemented by numerous software vendors across the globe and has been used to exchange mixed type payloads, including XML, for several years. MIME was designed purely as a packaging (enveloping) solution to allow the transport of mixed payloads using Internet E-mail (SMTP). MIME is also being used by other transport technologies as a packaging technology, most notably HTTP.

C.3 XML

eXtensible Markup Language (XML) version 1.0 is a technical specification holding a RECOMMENDED status created by the World Wide Web Consortium. It has been implemented by numerous software vendors across the globe and has been used to describe a broad spectrum of document structures from very simple to very complex. XML is a very flexible markup language that can be used to represent virtually any type of document. XML can be used solely for packaging (enveloping) documents of any type, providing the data can be "transformed" into "legal" XML.

In some cases, XML documents MUST be placed into transport specific "envelopes" before being transported. For example, XML data MUST be placed in a MIME envelope when being transported via SMTP or HTTP.

C.4 Conclusion

The packaging sub-group examined the capabilities of both XML and MIME relative to the list of packaging requirements above. It's important to note that neither technology met all of the ebXML requirements and in the end it was the packaging sub-groups assessment of which technology came closest to meeting ALL of the ebXML requirements that determined which technology SHOULD be used.

MIME was chosen to serve as the ebXML packaging technology, over XML, based on the

information contained in following table:

Reason	Requirement(s) Satisfied
There is no formal packaging recommendation within IETF or W3C, based on XML. If ebXML were to choose XML as a packaging technology it would be required to define an XML packaging specification and submit this to IETF or W3C for adoption as a formal standard.	to not reinvent the wheel - re-use where possible [2]
XML requires that binary and other types of payload data including XML documents be base64 encoded in order to be encapsulated within a XML root document. Base64 encoding ensures that no illegal XML characters exist within a document and recursive XML documents are "hidden". Base64 encoding imposes a significant processing overhead and results in larger messages, which affect both transmission and processing times. Base64 encoding of binary data is required of MIME content when being transported by SMTP, but this is a transport level requirement, not a requirement imposed by MIME. Binary data can be packaged and transported without alteration when using MIME over HTTP	Minimize intrusion to payload (special encoding or alteration) Low processing overhead
At the time of defining this specification there is no industry standard way to package an encrypted message, or portion of a message, using XML.	All or part of the documents in a message MAY be encrypted prior to sending [2]
MIME could be used in conformance within existing IETF recommendations, no additions or changes are initially required to produce a functional envelope.	to not reinvent the wheel - re-use where possible [2]

Appendix D MIME Type discussion

Three MIME media types were considered to serve as Content-Type for the ebXML Message Envelope:

- Multipart/related
- Multipart/Mixed
- Multipart/form-data

The group selected the multipart/related media type to serve as the preferred message envelope Content-Type.

Note:

There was some discussion over the similarities of multipart/related and multipart/mixed, both of which appear to offer similar capabilities and both could meet stated requirements. However, the group converged on multipart/related, believing it to be more semantically appropriate for ebXML.

There was significant discussion over whether to support multipart/form-data as an alternate Content-Type for message-envelope, due to the large installed base of web browsers that support this Content-Type.

It was determined that multipart/related was a more generic Content-Type than multipart/form-data and the multipart/related Content-Type is the preferred Content-Type for ebXML message envelopes. Multipart/form-data Content-Type is typically associated with HTTP/HTML web forms, whereas multipart/related can be associated with any type of data.

Additionally, due to limitations in their handling of multipart ebXML payloads it was determined that existing web browsers are unable to support the full breadth of functions needed to package complex ebXML messages containing multipart payloads. Therefore browser vendors are encouraged to add support for the ebXML enveloping standard as specified in this document.