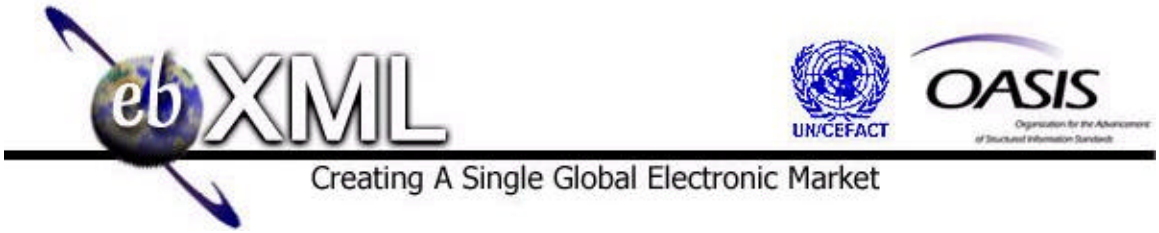


ebXML Transport, Routing & Packaging Message Envelope Specification

Working Draft 26-May-2000

This version:
ebXML Message Envelope Specification v0-5.doc
Latest version:
N/A
Previous version:
ebXML Transport, Routing & Packaging Message Envelope Specification (STRAWMAN 0.4)
Editor:
Dick Brooks <dick@8760.com>
Authors:
Dick Brooks <dick@8760.com>
Nick Kassem <nick.kassem@sun.com>
Contributors:
See Acknowledgments

Abstract

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

This document defines the structure (or envelope) used to encapsulate data for transport between parties, following the specifications defined by ebXML. Every attempt has been made to ensure that ebXML requirements, related to transport, routing and packaging are addressed within this specification. 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.

Status of this Document

This document is a draft for Public Comment. The document represents work in progress and no reliance should be made.

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 IETF RFC 2119.

Table of Contents

1 Introduction

This specification defines the 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. This main goal of this specification is to define an enveloping structure to encapsulate any digitally encoded payload for transport over any data communication mechanism. No limitation is implied relative to processing mode, the structures defined in this specification can be used in one-way, broadcast, request/response (RPC) or full messaging mode communications between parties.

1.1 Purpose and Scope

This document provides software practitioners with sufficient detail to develop software used in the packaging, exchange and processing of information following ebXML Transport, Routing and Packaging specifications. This document defines the enveloping specifications used to represent ebXML messages and encapsulate ebXML message headers and digital payloads for transport over a data communication mechanism. There are other aspects of ebXML messaging that are not addressed in this document, for example: Content and Semantics of Message Headers, payload structure, business processes, choreography of message exchanges and error handling. These are addressed in other ebXML specification documents.

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

1.1.1 Goals

The goals of this specification are:

- Meet the requirements specified by the ebXML Transport, Routing and Packaging Overview and Requirements Document -version 0.92 [2]
- Meet the requirements identified by the packaging sub-group (the people responsible for creating this specification)
- Compatible with other ebXML specifications
- Leverage existing industry standards
- Implementable in a prototype by the May meeting of ebXML group
- Enable parties to "package" very simple to very complex combinations of headers and payloads
- Payload Neutral
- Transport Neutral
- Support corporate security policies and business practices

1.2 Relationship to other specifications

This specification is one of a set of related specifications, for details see ebXML Transport, Routing & Packaging Document Control and Index [1]

1.3 Specification Structure

This specification is organized around four main topics:

- Packaging and Other Requirements
- Candidate Packaging Technologies and Selection Process
- Packaging Specifications
- Security Considerations

2 Packaging and Other Requirements

The packaging sub-group began development of the ebXML envelope specification (this document) by first identifying requirements from the Transport Routing and Packaging Overview and Requirements [2] that directly affect enveloping. Secondly, the group identified requirements, specific to enveloping that were not included in [2]. This combined list of requirements was used by the group to evaluate candidate packaging technologies and would ultimately serve as the "checklist" for choosing a solution. The combined list of requirements considered by the packaging sub-group includes:

- Able to handle large documents
- Able to envelope any document type
- Minimize intrusion to payload (special encoding or alterations)
- Minimize potential for abnormal termination caused by envelopes
- Facilitate a migration path for existing installed base and technologies
- Low processing overhead
- Support for recursive documents
- Able to preserve digital signatures
- Able to unambiguously identify signed data
- Documents, expressed either in XML or other electronic formats, must be able to be wrapped inside a *message envelope* for transporting between the *parties* involved in that want to execute an eCommerce - *Service* or *Transaction* [2]
- Multiple *document*s, whether related or not, may be transportable within a single *message envelope* [2]
- *Messages* can be transported over many network protocols (e.g. HTTP, SMTP, CORBA, JMQ, MQSeries, MSMQ, etc.) [2]
- *Messages* can be sent using a variety of methods: [2]
  - to a single *party*, e.g. by specifying a URL
  - to multiple *parties*, e.g. by specifying a list of URLs
  - to an agent or intermediary for forwarding to the next *party*
- Individual *messages* must be capable of routing serially or in parallel with other related *messages* [2]
- Publish and Subscribe[2]

- *Messages* may be distributed to the members of a list of *parties* using a "Publish and Subscribe"
- mechanism
- the anonymity of the subscriber may optionally be maintained

- *Documents* and/or *message headers* may be digitally signed [2]
- The signature over the *documents* or *message headers* should be independent of the transport protocol used [2]
- A single digital signature may be used to bind together *documents* either: [2]
  - within the same *message*
  - in another *message*
  - somewhere else (for example the content at a URL)
- Signatures on digitally signed *documents* can be used to: [2]
  - verify the authenticity of the *party* that is the sender,
  - provide non-repudiation of origin or receipt, and
  - ensure that the content of the message has not changed
- All or part of the *documents* in a *message* may be encrypted prior to sending [2]
- *messages* may be encrypted during transportation using a transport protocol [2]
- documents may be time stamped securely with a digital signature [2]
- Platform Independent Interoperability [2]
  - Servers/systems that support the exchange of documents can be treated as "black boxes"
  - The method used to transport documents is completely independent of:
  - the hardware used by the server/services at each end
  - the software or systems architecture of the server/services at each
  - the language used for implementation of systems and *applicatio*ns.
  - Support for a *service* can be expressed solely in terms of the type and sequence in which *documents* (and their *message envelope*s) can be exchanged
  - The approach must be suitable for implementation on hardware that varies from a very simple device to a large multi-processor/system complex
- The protocol must be extensible to support: [2]
  - additional types of data in message headers and message routing information
  - new values for codes
  - new ways and methods of exchanging data
- enable any party to carry out integrated eCommerce transactions with any other party anywhere in the world using their hardware and software vendor of choice [2]
- attract a wide variety of vendors to implement the approach [2]
- to not reinvent the wheel - re-use where possible [2]
- to enable existing "messaging" solutions to "bridge" to the ebXML solution [2]
- to scale from SMEs to large companies [2]

- to scale from low power to high end solutions [2]

3 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

3.1 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.

3.2 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.

3.3 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.

3.4 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]

The packaging sub-group did find that the deficiencies listed above that caused XML to be excluded were directly related to XML's immaturity relative to MIME. It was the sub-groups opinion that XML is a powerful technology, with great potential and the ebXML group should continue to monitor XML's progress in these areas. It is expected that XML will overcome these issues and may one day provide a future packaging solution suitable for ebXML.

The ebXML executive committee should consider sending this document to the W3C for consideration as a set of requirements to be used by a W3C workgroup in the creation of an XML based packaging solution.

4 Packaging Specification

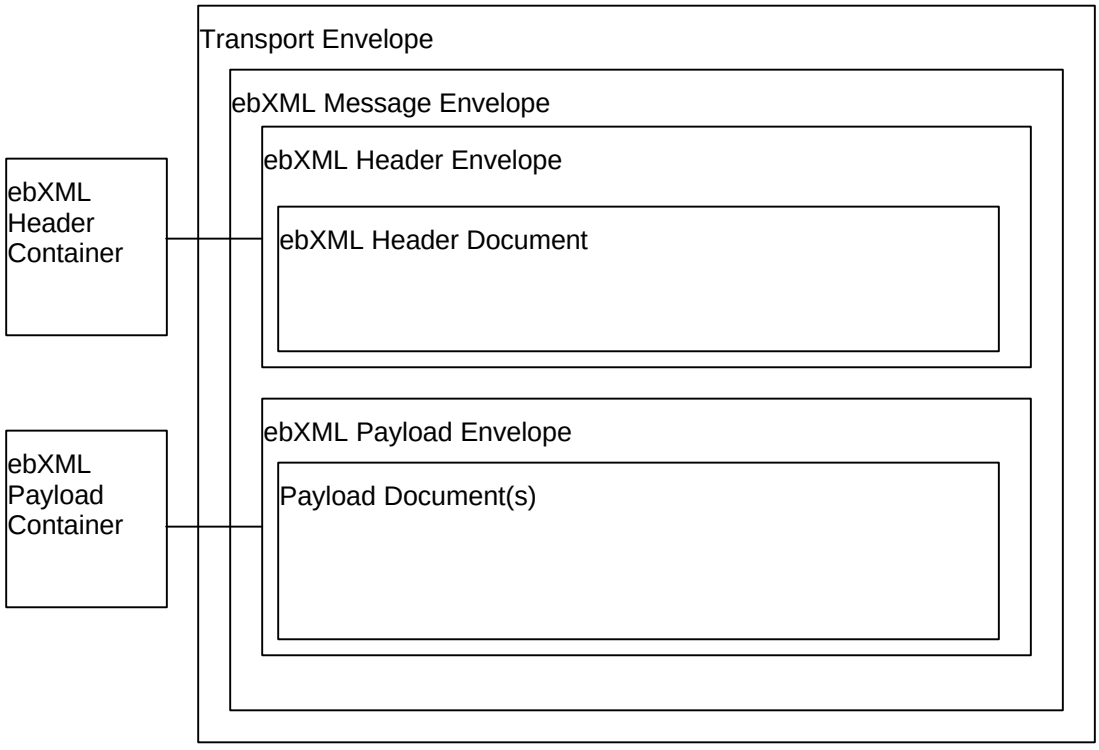
4.1 General Conventions

- All headers, attributes and values defined in this specification are to be handled in a case insensitive fashion, regardless of the way information is presented in this document.
- All messages following the ebXML standard must follow the specifications for packaging defined in this document, regardless of message type (request, response, error, et al).
- Values associated with MIME header attributes are valid in both "quoted" and unquoted form, for example both of the following forms are valid: (type="ebxml" or type=ebxml)

4.2 Message Structure

A Message Consists of:

- a conditional outer **Transport Envelope**, such as HTTP or SMTP,
- a transport independent **Message Envelope**, for example MIME multipart/related, that contains the two main parts of the Message:
  - a **Header container** that is used to envelope one ebXML header document, and
  - an optional **Payload container** that is used to envelope the real payload of the Message



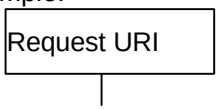
4.3 Transport Envelope

This document does NOT define any requirements affecting the structure of transport level envelopes. It is expected that existing transport systems, such as SMTP, HTTP, FTP and others can be used to send/receive ebXML compliant messages, without modification. The only requirement ebXML has on the transport envelope is the ability to identify a specific "handler" to receive incoming ebXML messages. All transports known at the time of creating this specification support this requirement.

A transport envelope is only required in those cases requiring such structures. In the case of HTTP or SMTP transport envelopes are REQUIRED, however in the case of FTP no transport envelope is needed.

In summary, an implementer of software to process ebXML messages must be aware of transport specific requirements relative to transport envelopes.

Implementers are expected to provide an *ebXML handler* to process all incoming ebXML requests for service contained within an ebXML message. This handler may be a dispatch process or an actual application with the capability to process the message. There will be at least one *ebXML handler* for each supported transport protocol. In the SMTP case this ebXML handler could be associated with a particular mailbox (e.g. ebxmlhandler@mycompany.com). In the HTTP case the ebXML handler is contained in the Request-URI on a POST operation, for example:




```
POST /ebxmlhandler HTTP/1.1
```

Implementers must provide a means to communicate the name of an ebXML handler for all trading partners to use when sending ebXML messages. Implementers should consider using a common identifier, such as "ebxmlhandler", or, alternatively provide a "discovery mechanism" to be used by a sender to determine the ebXML handler to receive service request.

OPEN ISSUE 1.0

When ebXML messages are transported using MIME aware transports, such as SMTP and HTTP, the ebXML message envelope MIME headers are sent as part of the transport layer MIME headers. Concerns over a receivers ability to separate ebXML message envelope MIME headers from the transport layer MIME headers have been raised as this could affect a receivers ability to create a digitally signed receipt of the "received" ebXML message, including the ebXML message envelope MIME headers. The issue to be resolved is:

"What portion of an ebXML message should be considered within the scope of a signature block when creating signed receipts?"

If the ebXML message envelope MIME headers are within the "signature block" used to create a signed receipt then software implementers may not be able to reconstruct the original ebXML message, complete with ebXML message MIME headers, using the CGI or servlet interface used within many commercial web server products. If ebXML message envelope MIME headers are NOT within the signature block used to create signed receipts then there is no issue. Alternatively, ebXML implementers could develop their own HTTP processor, which could provide access to the original stream of data sent by the sender.

4.4 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

4.4.1 Content-type

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.

The Content-type header also contains three attributes:

1. type
2. version
3. boundary

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

```
Content-type: multipart/related; type="ebxml"
```

OPEN ISSUE 2.0

ebXML's use of the type parameter, as described above, is in conflict with the usage of this parameter as defined in RFC 2387, which states "The type parameter must be specified and its value is the MIME media type of the "root" body part."

Several alternatives have been discussed:

Option 1 - Add a new parameter to the multipart/related media type (possibly "subtype" or another name) which would be used to identify the message as an "ebxml" compliant message.

Option 2 - Register a new MIME media type "application/vnd.ebxml" which would be used to identify the Content-type of the ebXML header body part (root body part). The type parameter within the multipart/related message header would contain "application/vnd.ebxml", which conforms with RFC 2387.

The version attribute is used to identify the particular version of ebxml message envelope being used. 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.

Currently, there are no version-less message envelopes defined, therefore all message headers SHOULD USE "0.1". The following is an example usage of version:

```
Content-type: multipart/related; type="ebxml"; version="0.1"
```

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="ebxml"; version="0.1";
boundary="-------8760"
```

4.4.2 Content-Length

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

```
Content-Length: 9841
```

4.4.3 Complete ebXML Message Envelope Example

An example of a complete ebXML compliant Message Envelope appears as follows:

```
Content-type: multipart/related; type="ebxml"; version="0.1";
boundary="-------8760"
Content-Length: 9841
```

4.5 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 ebXML Message Header Specification [3]. 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 content portion contains a ebXML header document as defined by [3]. 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 header of the ebXML Message Envelope and ebXML Header envelope.

4.5.1 Content-ID

The Content-ID MIME header is used to uniquely identify this container as the ebXML header envelope. There is only one possible value to associate with this header "ebxmlheader". An example usage follows:

```
Content-ID: ebxmlheader
```

OPEN ISSUE 3.0

ebXML's use of the Content-ID header field, as described above, is in conflict with the usage of this header field as defined in RFC 2045, which states, "Like the Message-ID values, Content-ID values must be generated to be world-unique."

One possible solution is to replace the Content-ID header field with the Content-Description header field and use as specified above.

4.5.2 Content-Length

The Content-Length header contains a decimal value used to identify the total number of 0CTETS contained in the ebXML header document residing in the content portion of the container. Example:

```
Content-Length: 4208
```

4.5.3 Content-Type

The Content-type for an ebXML header is identified with the value "application/xml", as defined in RFC2376. An example of this content-type is:

```
Content-type: application/xml
```

4.5.3.1 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 RFC 2015.

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

If XML Dsig is used then implementers are expected to follow the specifications contained in the W3C Recommendation for XML Dsig.

4.5.4 Complete Example of an ebXML Header container

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

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

NOTE: A REAL ebXML Header example here prior to final release as an approved specification.

4.6 Payload Container Specifications

The payload container of Message is optional. 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 contains no entries then 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:

4. Content-ID
5. Content-Length
6. Content-Type

The content portion contains whatever structure and content two consenting parties agree to exchange. 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 implementers decision.

4.6.1 Content-ID

The Content-ID MIME header is used to uniquely identify this container as the ebXML payload envelope. There is only one possible value to associate with this header "ebxmlpayload". An example usage follows:

```
Content-ID: ebxmlpayload
```

OPEN ISSUE 4.0

ebXML's use of the Content-ID header field, as described above, is in conflict with the usage of this header field as defined in RFC 2045, which states "Like the Message-ID values, Content-ID values must be generated to be world-unique."

One possible solution is to replace the Content-ID header field with the Content-Description header field and use as specified above.

4.6.2 Content-Length

The Content-Length header contains a decimal value used to identify the total number of OCTETS contained in the content portion of the payload container. Example:

```
Content-Length: 5012
```

4.6.3 Content-Type

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

```
Content-Type: application/xml
```

4.6.3.1 Optional Support for Signed and Encrypted Payloads

Implementers are free to support encrypted and digitally signed payloads. Digitally signed and/or encrypted 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 RFC 2015.

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

If XML Dsig is used then implementers are expected to follow the specifications contained in the W3C Recommendation for XML Dsig.

4.6.4 Complete Example of an ebXML Payload container

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

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

4.7 Complete Example of an ebXML Message enveloped using multipart/related content-type sent via HTTP POST

NOTE: The following example is representative of the ebXML packaging structure ONLY. ebXML headers used in this example are not to be construed as ACTUAL ebXML header structures. The "official" format and content of ebXML headers is defined in ebXML Transport, Routing and Packaging: Message Header Specification version x.x. Published dd mmmm 2000 [3].

Following is a complete example of an ebXML Message sent via HTTP POST method:

```
POST /ebxmlhandler HTTP/1.1
Accept: multipart/related
Accept-Language: en-us
Content-Type: multipart/related; type=ebxml; version=0.1; boundary=----------------------
------7d02a82e5f8
Accept-Encoding: gzip, deflate
```

```
User-Agent: Group 8760 InsideAgent
Host: localhost:9090
Content-Length: 9293
Connection: Keep-Alive

----------------------------7d02a82e5f8
Content-ID: ebxmlheader
Content-Length: 211
Content-Type: application/xml

<?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
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--
```

4.8 Complete Example of an ebXML Message enveloped using multipart/related content-type sent via SMTP

NOTE: The following example is representative of the ebXML packaging structure ONLY. ebXML headers used in this example are not to be construed as ACTUAL ebXML header structures. The "official" format and content of ebXML headers is defined in ebXML Transport, Routing and Packaging: Message Header Specification version x.x. Published dd mmmm 2000 [3].

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

Following is a complete example of an ebXML Message sent via SMTP:

```
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="ebxml"; version="0.1";
  boundary="----=_NextPart_000_0005_01BFB846.BF7FABA0"

------=_NextPart_000_0005_01BFB846.BF7FABA0
Content-Type: application/xml
Content-ID: ebxmlheader
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
```

```
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--
```

5 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.

6 References

[1] ebXML Transport, Routing & Packaging Document Control and Index

[2] ebXML Transport, Routing and Packaging: Overview and Requirements

[3] ebXML Transport, Routing and Packaging: Message Header Specification

[4] XMTP - Extensible Mail Transport Protocol

`http://www.openhealth.org/documents/xmtp.htm`

7 Acknowledgments

Jonathan Borden - Author of XMTP

Jon Bosak - Sun

David Burdett - Commerce One

Rik Drummond - Drummond Group

Christopher Ferris - Sun

Ian Jones - British Telecom

Henry Lowe - OMG

Jim McCarthy - webXI

Bob Miller - GEIS

Dale Moberg - Sterling Commerce

Prasad Yendluri - Vitria

9 Authors' Address

Dick Brooks
Group 8760
110 12th Street North
Suite F103
Birmingham, Alabama 35203

Telephone: 205-250-8053
E-mail: dick@8760.com

991 Nicholas Kassem
992 Java Software, Sun Microsystems
993 901 San Antonio Road, MS CUP02-201
994 Palo Alto, CA 94303-4900

995 Telephone: 408-863-3535
996 E-mail: Nick.Kassem@eng.sun.com