



---

# OASIS ebXML Messaging Transport Binding for Digital Signature Services Version 1.0

## Committee Draft

**9 May 2008**

**Document Identifier:**

[oasis-dss-1.0-profiles-ebxml-cd01](http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/cd01/oasis-dss-1.0-profiles-ebxml-cd01)

**Specification URIs:**

**This Version:**

[\(Authoritative\)](http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/cd01/oasis-dss-1.0-profiles-ebxml-cd01.doc)

<http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/cd01/oasis-dss-1.0-profiles-ebxml-cd01.html>  
<http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/cd01/oasis-dss-1.0-profiles-ebxml-cd01.pdf>

**Previous Version:**

N/A

**Latest Version:**

<http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/oasis-dss-1.0-profiles-ebxml.doc>  
<http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/oasis-dss-1.0-profiles-ebxml.html>  
<http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/oasis-dss-1.0-profiles-ebxml.pdf>

**Technical Committee:**

OASIS Digital Signature Services eXtended (DSS-X) TC

**Chair(s):**

Juan Carlos Cruellas, *Centre d'aplicacions avanades d'Internet*, <[cruellas@ac.upc.edu](mailto:cruellas@ac.upc.edu)>  
Stefan Drees, Individual Member, <[stefan@drees.name](mailto:stefan@drees.name)>.

**Editor(s):**

Pim van der Eijk, *Sonnenglanz Consulting BV*, <[pvde@sonnenglanz.net](mailto:pvde@sonnenglanz.net)>  
Ernst Jan van Nigtevecht, *Sonnenglanz Consulting BV*, <[ejvn@sonnenglanz.net](mailto:ejvn@sonnenglanz.net)>

**Related work:**

This specification is related to:

- OASIS Digital Signature Service Core Protocols, Elements and Bindings. Version 1.0.
- OASIS ebXML Messaging Services version 2.0
- OASIS ebXML Messaging Services version 3.0

**Abstract:**

Mappings from DSS messages into standard communication protocols are called DSS *bindings*. A *transport binding* specifies how DSS messages are encoded and carried using a transport protocol. The DSS Core standard [**DSS Core**] specifies two transport bindings. This document specifies an alternative transport binding that uses the OASIS ebXML Messaging Service. This profile supports is compatible with both the version 2.0 [**ebMS 2.0**] and version 3.0 [**ebMS 3.0**] ebXML Messaging OASIS standards.

**Status:**

This document was last revised or approved by the OASIS DSS-X TC on the above date. The level of approval is also listed above. Check the "Latest Version" or "Latest Approved Version" location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "Send A Comment" button on the Technical Committee's web page at <http://www.oasis-open.org/committees/dss-x/>.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (<http://www.oasis-open.org/committees/dss-x/ipr.php>).

The non-normative errata page for this specification is located at <http://www.oasis-open.org/committees/dss-x/>.

---

## Notices

Copyright © OASIS ® 2008. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS", [insert specific trademarked names and abbreviations here] are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

---

## Table of Contents

1	Introduction.....	5
1.1	Benefits and intended use .....	5
1.2	Scope.....	5
1.3	Terminology .....	5
1.4	Namespaces.....	5
1.5	Normative References .....	6
1.6	Non-Normative References .....	7
2	Exchanging DSS Messages using the ebXML Messaging Service.....	8
2.1	DSS message exchanges .....	8
2.1.1	Element <MessageId> and <RefToMessageId>.....	8
2.1.2	Element <ConversationId>.....	8
2.2	Other Header Elements .....	9
2.2.1	Service.....	9
2.2.2	Action.....	9
2.2.3	Role .....	9
2.3	Packaging .....	9
2.3.1	Packaging in ebXML Messaging version 2.0 .....	9
2.3.2	Packaging in ebXML Messaging version 3.0 .....	11
3	Security Binding.....	13
4	Conformance .....	14
4.1	Conformance as a DSS Client using ebMS 2.0.....	14
4.2	Conformance as a DSS Server using ebMS 2.0 .....	14
4.3	Conformance as a DSS Client using ebMS 3.0.....	14
4.4	Conformance as a DSS Server using ebMS 3.0 .....	14
A.	Sample ebXML Messaging 2.0 SOAP envelopes.....	15
A.1	Sample SOAP envelope for DSS SignRequest message .....	15
A.2	Sample SOAP envelope for DSS SignResponse message.....	16
B.	Sample Collaboration Protocol Agreement.....	17
C.	Revision History.....	22

---

# 1 Introduction

2 Mappings from Digital Signature Services (DSS) messages into standard communication protocols are  
3 called DSS *bindings*. A *transport binding* specifies how DSS messages are encoded and carried using a  
4 transport protocol. The DSS 1.0 Core standard [**DSS Core**] specifies two transport bindings. This  
5 document specifies an alternative transport binding that uses the OASIS ebXML Messaging Service. This  
6 profile supports either version 2.0 [**ebMS 2.0**] or version 3.0 [**ebMS 3.0**].

## 7 1.1 Benefits and intended use

8 The benefits of a DSS transport binding for ebXML Messaging include the following:

- 9 • An application area for DSS is to sign electronic business documents, such as electronic  
10 invoices and order documents. The ebXML Messaging service is designed to support  
11 electronic business. This profile allows user communities that use ebXML messaging to  
12 exchange electronic business documents to use the same message transport protocol to  
13 interface with DSS service providers.
- 14 • ebXML Messaging supports asynchronous messaging, using elements in the ebXML business  
15 document header to correlate requests and asynchronous responses. It therefore naturally  
16 supports use cases of DSS that require, or benefit from, an asynchronous messaging  
17 capability, where a signature may not be returned until hours after it was requested.
- 18 • ebXML Messaging includes functionality for reliable messaging. It therefore provides a more  
19 robust message channel between DSS clients and servers, which facilitates the use of DSS in  
20 automated workflows.

## 21 1.2 Scope

22 There are currently two versions of the ebXML Message Service specification, which are very similar from  
23 a user functionality perspective but are not interoperable. For the purpose of this profile, the differences  
24 between these two versions are unimportant as the relevant ebXML message header elements affected  
25 by this profile have similar syntax and identical semantics. This profile therefore defines how to use either  
26 version 2.0 [**ebMS 2.0**] or version 3.0 [**ebMS 3.0**] of the ebXML Messaging Service as transport protocol  
27 for DSS messages.

28 Unlike other DSS profiles that constrain or extend the DSS XML messages to support particular uses of  
29 DSS, this profile is limited to being a transport binding. It does not constrain or extend the use of DSS  
30 itself. It is therefore compatible, and can be used in conjunction with, other profiles that are defined for  
31 particular business uses of DSS.

## 32 1.3 Terminology

33 The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD  
34 NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described  
35 in [**RFC 2119**].

## 36 1.4 Namespaces

37 This following table lists the namespaces referenced in this specification.

Prefix	Namespace	Specification(s)
cpa2	http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd	[ <b>ebCPA 2.0</b> ]

dss	urn:oasis:names:tc:dss:1.0:core:schema	[DSS Core]
eb2	http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd	[ebMS 2.0]
eb3	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/	[ebMS 3.0]
s11	http://schemas.xmlsoap.org/soap/envelope/	[SOAP 1.1]
s12	http://www.w3.org/2003/05/soap-envelope	[SOAP 1.2]

38

39 In the context of this profile, the differences between version 2.0 and version 3.0 of ebXML Messaging  
 40 are limited. The generic prefix *eb* will be used to refer to either version in situations where elements are  
 41 used that have the same syntax and semantics in both versions:

Prefix	Namespace	Specification(s)
eb	http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd or http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/	[ebMS 2.0] or [ebMS 3.0]

42

## 43 1.5 Normative References

- 44 [DSS Core] S. Drees et al., *Digital Signature Service Core Protocols, Elements and Bindings*.  
 45 Version 1.0. <http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.pdf>.  
 46 OASIS Standard, 11 April 2007.
- 47 [ebMS 2.0] OASIS ebXML Messaging Services. Version 2.0.  
 48 [http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS\\_v2\\_0.pdf](http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS_v2_0.pdf).  
 49 OASIS Standard, 1 April 2002.
- 50 [ebMS 3.0] P. Wenzel et al., *OASIS ebXML Messaging Services v3.0: Part 1, Core Features*.  
 51 [http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/ebms\\_core-3.0-spec-os.pdf](http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/ebms_core-3.0-spec-os.pdf). OASIS Standard, 1 October 2007.
- 53 [RFC 2119] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*,  
 54 <http://www.ietf.org/rfc/rfc2119.txt>, IETF RFC 2119, March 1997.
- 55 [RFC 2246] T Dierks, C. Allen., *The TLS Protocol Version 1.0*.  
 56 <http://www.ietf.org/rfc/rfc2246.txt>. IETF RFC 2246, January 1999.
- 57 [RFC 2822] P. Resnick, *Internet Message Format*. <http://www.ietf.org/rfc/rfc2822.txt>. IETF  
 58 RFC 2822, April 2001.
- 59 [SOAP 1.1] D. Box et al., *Simple Object Access Protocol (SOAP) 1.1*.  
 60 <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>. W3C Note, 08 May 2000.
- 61 [SOAP 1.2] M. Gudgin et al., *SOAP Version 1.2 Part 1: Messaging*.  
 62 <http://www.w3.org/TR/2003/REC-soap12-part1-20030624/>. W3C  
 63 Recommendation, 24 June 2003.
- 64 [SOAPATTACH] J. Barton et al., *SOAP Messages with Attachments*.  
 65 <http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211>. W3C Note, 11  
 66 December 2000.

69 **1.6 Non-Normative References**

- 70     **[ebCPA 2.0]**     *ebXML Collaboration-Protocol Profile and Agreement*. Version 2.0.  
71         <http://www.oasis-open.org/committees/ebxml-cppa/documents/ebcpp-2.0.pdf>.  
72         OASIS Standard, 23 September 2002.
- 73     **[RFC 4346]**     T Dierks, C. Allen., *The Transport Layer Security (TLS) Protocol Version 1.1*.  
74         <http://www.ietf.org/rfc/rfc4346.txt>. IETF RFC 4346, April 2006.
- 75     **[SOAP 1.2(2<sup>nd</sup>)]** M. Gudgin et al., *SOAP Version 1.2 Part 1: Messaging Framework (Second  
76         Edition)*.  
77         <http://www.w3.org/TR/2007/REC-soap12-part1-20070427/>. W3C  
78         Recommendation, 27 April 2007.
- 79
- 80

---

## 81   **2 Exchanging DSS Messages using the ebXML** 82   **Messaging Service**

83 This profile specifies how the ebXML messaging service can be used to transport DSS messages.  
84 Section 2.1 defines DSS message exchanges and describes how these map to ebXML message  
85 exchanges. The profile constrains the use of various elements in the ebXML message header (section  
86 2.2) and defines how DSS XML messages and documents that are transmitted along with these  
87 messages are packaged into ebXML messages (section 2.3).

### 88   **2.1 DSS message exchanges**

89 The DSS protocol defines two two-way message exchanges involving four DSS XML messages.

- 90   • A signature *generation* message exchange. A DSS client can send a message to invoke a  
91    SignRequest action on a DSS server. The DSS server responds by returning a  
92    SignResponse DSS response message.
- 93   • A signature *verification* message exchange. A DSS client can send a message to invoke a  
94    VerifyRequest action on a DSS server. The DSS server responds by returning a  
95    VerifyResponse DSS response message.

96 Depending on bilaterally agreed bindings, this profile allows these message exchanges to be executed  
97 either as a single synchronous ebXML SOAP request-response interaction, or asynchronously as two  
98 separate ebXML SOAP one way messages.

#### 99   **2.1.1 Element <MessageId> and <RefToMessageId>**

100 The elements `<eb:MessageId>` and `<eb:RefToMessageId>` in the ebXML header are used to  
101 correlate ebXML request and response message. In an ebXML message containing a DSS response  
102 document, the value of `<eb:RefToMessageId>` MUST be set to the value of the `<eb:MessageId>`  
103 element of the ebXML message that contained the corresponding DSS request document.

104 At the level of DSS XML messages, correlation of DSS requests and response can be expressed by  
105 using the optional attribute `@RequestID` on the `<dss:SignRequest>`, `<dss:SignResponse>`,  
106 `<dss:VerifyRequest>` and `<dss:VerifyResponse>` XML elements. If this attribute is set on a  
107 `<dss:SignRequest>` or a `<dss:VerifyRequest>` DSS message, the DSS server MUST return it in  
108 the response `<dss:SignResponse>` or `<dss:VerifyResponse>` DSS documents, respectively.

109 If a DSS server receives an ebXML message containing a DSS request and subsequently sends back an  
110 ebXML message containing a DSS response, and if both DSS messages have the `@RequestID` set,  
111 then the value of these attributes MUST be the same and the value of the ebXML element  
112 `<eb:RefToMessageId>` in the ebXML response message MUST be identical to the value of the element  
113 `<eb:MessageId>` in the ebXML request message. The actual value of the `@RequestID` attribute in a  
114 DSS XML request document does not have to be the same as the value of the `<eb:MessageId>`  
115 element. The type of the `@RequestID` attribute is `xs:string`, whereas the value of the `<eb:MessageId>`  
116 element MUST match the *msg-id* production defined in [RFC 2822] and therefore MUST contain the “@”  
117 character.

#### 118   **2.1.2 Element <ConversationId>**

119 The element `<eb:ConversationId>` in the ebXML business document header identifies the (possibly  
120 long-running) conversation that a particular message takes part in. Conversations may span multiple one  
121 way or two way ebXML message exchanges. The value of `<eb:ConversationId>` in a sign request  
122 message and in the corresponding sign response message MUST be the same. Similarly, the value of

123 <eb:ConversationId> in a verify request and corresponding verify response message MUST also be  
124 the same.

## 125 **2.2 Other Header Elements**

126 The ebXML messaging service provides a general purpose business document header that contains  
127 ebXML extension elements. This profile constrains the values for some of these values.

### 128 **2.2.1 Service**

129 The value of the ebXML <eb:Service> element MUST have the fixed value  
130 urn:oasis:names:tc:dss:1.0.

### 131 **2.2.2 Action**

132 The value of the ebXML <eb:Action> element MUST have one of the following four fixed values:

- 133 – SignRequest
- 134 – SignResponse
- 135 – VerifyRequest
- 136 – VerifyResponse

137 Each of these values corresponds to the four DSS message types. An ebXML message conforming to  
138 this profile MUST have the value SignRequest (or SignResponse, VerifyRequest,  
139 VerifyResponse, respectively) for the <eb:Action> ebXML header element if, and only if, the  
140 business document transmitted using the message is a valid DSS XML document that has the root  
141 element <dss:SignRequest> (or <dss:SignResponse>, <dss:VerifyRequest>,  
142 <dss:VerifyResponse>, respectively).

### 143 **2.2.3 Role**

144 The value of the ebXML <eb:Role> element MUST have one of the following values:

- 145 – DSSClient for the DSS client partner
- 146 – DSSServer for the DSS server partner

## 147 **2.3 Packaging**

148 The ebXML message service is based on SOAP and MIME enveloping and provides a number of ebXML  
149 extension elements. The packaging differs due to the differences in message structure in versions 2.0  
150 and 3.0 of ebXML Messaging.

151 DSS Core [**DSS Core**] itself also provides mechanisms to either include business documents in the DSS  
152 XML structure or reference business documents in a MIME structure. The following example shows a  
153 <dss:Document> element containing a PDF business document included in the DSS XML in base64  
154 encoded form. The structure then has a format like:

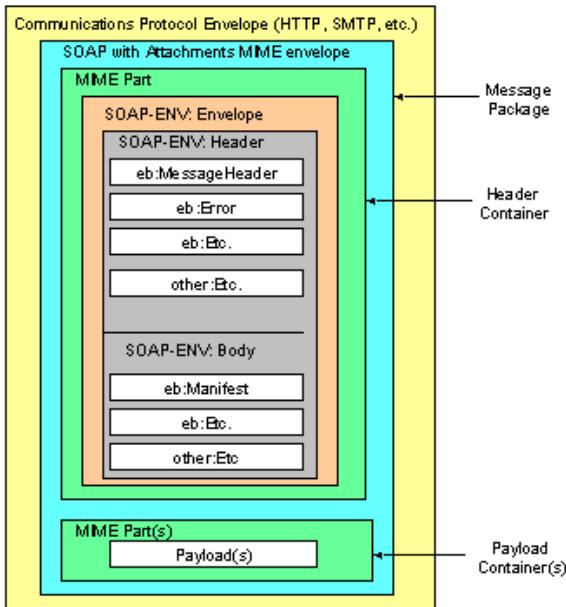
```
155 <dss:Document ID="doc1">
156   <dss:Base64DataMimeType="application/pdf"> ... </dss:Base64Data>
157 </dss:Document>
```

158 Alternatively, DSS also allows documents to be transported in attachments and referenced from a  
159 <dss:AttachmentReference> element, which is similar to the ebXML extension elements  
160 <eb2:Manifest> and <eb3:PayloadInfo>.

### 161 **2.3.1 Packaging in ebXML Messaging version 2.0**

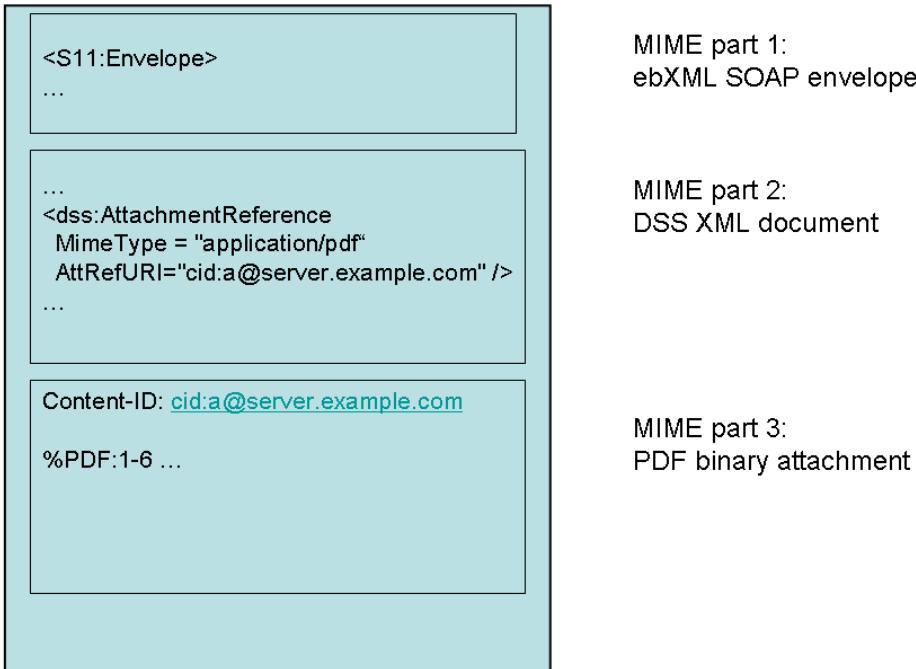
162 Version 2.0 of ebXML Messaging [**ebMS 2.0**] is based on SOAP 1.1 [**SOAP 1.1**] and provides extension  
163 elements from the eb2: namespace to both the SOAP header and SOAP body. Version 2.0 is also based

164 on the SOAP with attachments specification [**SOAPATTACH**]. The SOAP 1.1 envelope, including ebXML  
165 version 2.0 extension elements, is transported as the first part of a MIME envelope. Any business  
166 documents transported in a version 2.0 ebXML message are not contained in the SOAP envelope, but  
167 rather included in separate MIME parts. These documents are referenced from the SOAP envelope using  
168 an ebXML <eb2:Manifest> extension element. A version 2.0 ebXML message that carries  $n$  business  
169 documents therefore always consists of  $n+1$  MIME parts. This is shown in Figure 1 ebXML Messaging  
170 version 2 message structure, from [**ebMS 2.0**].



171  
172 *Figure 1 ebXML Messaging version 2 message structure*

173 When a DSS message is transported with version 2.0 of ebXML messaging, the message MIME  
174 envelope MUST contain at least two MIME parts: a first part containing a SOAP 1.1 envelope including  
175 ebXML version 2.0 extension elements and a second part containing a DSS XML document. If the  
176 document(s) referenced from the DSS message are not included in the DSS XML document, they are  
177 packaged in a third or subsequent MIME part. The following diagram illustrates this enveloping. It  
178 contains an attached PDF document which is referenced from a DSS XML document, which itself is  
179 referenced from the ebXML manifest.

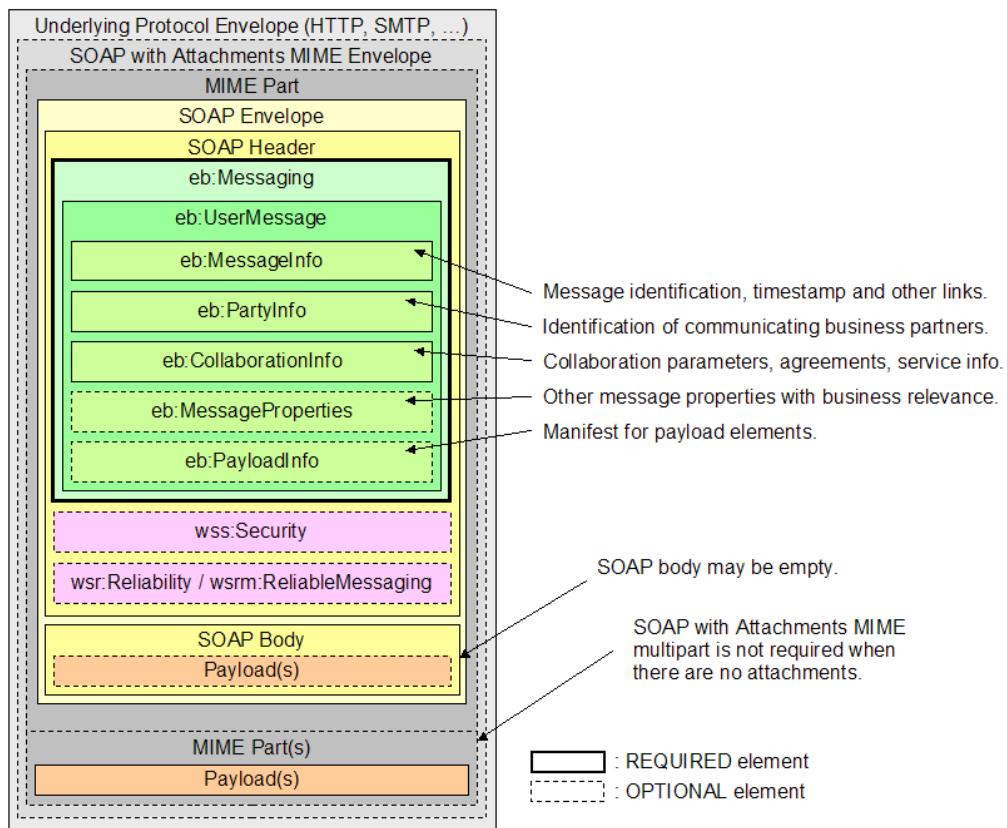


180

181 *Figure 2 ebXML message containing a DSS XML document and a PDF document*

### 182 **2.3.2 Packaging in ebXML Messaging version 3.0**

183 Version 3.0 of ebXML messaging can use either SOAP 1.1 [**SOAP 1.1**] or SOAP 1.2 [**SOAP 1.2**]. It  
 184 provides a number of elements from the eb3: namespace included in an <eb3: Messaging> structure,  
 185 included in the SOAP header. Any business documents transported in a version 3.0 ebXML message can  
 186 be contained in the SOAP body, or contained in separate MIME parts, when using SOAP with  
 187 attachments. These business documents are referenced from the SOAP envelope using an ebXML  
 188 <eb3:PayloadInfo> extension element structure that has a similar function to the <eb2:Manifest>  
 189 element. The use of SOAP with attachments is optional with version 3.0 of ebXML Messaging, and only  
 190 needed in situations when the business document is not contained in the SOAP body. This is shown in  
 191 Figure 3 ebXML Messaging version 3.0 message structure, from [**ebMS 3.0**].



192

193 *Figure 3 ebXML Messaging version 3.0 message structure*

194 When a DSS message is transported with version 3.0 of ebXML messaging, it can be included as the  
 195 second MIME part as in version 2.0. An alternative packaging option is for the DSS XML document to be  
 196 included in the SOAP body. If any documents referenced from the DSS document are base64 included in  
 197 the DSS XML structure, there is no need to use a SOAP with attachments MIME envelope as all data can  
 198 be included in the SOAP envelope.

199

200

---

201

### 3 Security Binding

202

This profile is based on the security bindings defined in section 6 of the DSS Core specification [\[DSS Core\]](#). Specifically, the ebXML message exchange between the DSS client and DSS server SHOULD use TLS 1.0 [\[RFC 2246\]](#) to provide message confidentiality, integrity and authentication.

205

Note: Although at the time of writing this profile TLS 1.0 is an obsoleted standard, which is superseded by TLS 1.1 [\[RFC 4346\]](#), it is still used as normative reference to keep it aligned with the DSS Core specification [\[DSS Core\]](#).

208

The DSS Core protocol defines a mechanism to carry data authenticating the claimed identity of the entity on whose behalf the DSS services are invoked, such as the use of SAML tokens. The use of such additional security mechanisms is out of the scope for this transport binding profile, but may be required by profiles that use this transport binding for particular DSS applications.

---

## 212 4 Conformance

213 Any implementation of this profile is not conformant with this specification if it fails to satisfy one or more  
214 of the MUST or REQUIRED level requirements defined in this specification.

215 An implementation of this profile provides DSS client functionality, DSS server functionality, or both DSS  
216 client and DSS server functionality.

217 An implementation of this profile provides support for either version 2.0 or 3.0 of the OASIS ebXML  
218 Messaging Service specification, or supports both.

219 Any implementation of this profile SHOULD support the DSS security binding described in section 3.

### 220 4.1 Conformance as a DSS Client using ebMS 2.0

221 An implementation of this profile conforms to this profile as a *DSS Client using ebMS 2.0* if it meets the  
222 following requirements:

- 223 • Supports sending of `SignRequest` signature generation request messages and `VerifyRequest`  
224 signature verification request messages with syntax and semantics defined in section 2 of this  
225 specification.
- 226 • Supports receiving of `SignResponse` signature generation response messages and  
227 `VerifyResponse` signature verification response message with syntax and semantics defined in  
228 section 2 of this specification.
- 229 • Supports version 2.0 of the ebXML Messaging Service version 2.0 OASIS Standard [**ebMS 2.0**].

### 230 4.2 Conformance as a DSS Server using ebMS 2.0

231 An implementation of this profile conforms to this profile as a *DSS Server using ebMS 2.0* if it meets the  
232 following requirements:

- 233 • Supports receiving of `SignRequest` signature generation messages and `VerifyRequest` signature  
234 verification messages with syntax and semantics defined in section 2 of this specification.
- 235 • Supports sending of `SignResponse` signature generation messages and `VerifyResponse`  
236 signature verification messages with syntax and semantics defined in section 2 of this specification.
- 237 • Supports version 2.0 of the ebXML Messaging Service version 2.0 OASIS Standard [**ebMS 2.0**].

### 238 4.3 Conformance as a DSS Client using ebMS 3.0

239 The conformance requirements for a *DSS Client using ebMS 3.0* are the same as for a *DSS Client using  
240 ebMS 2.0*, except that it support version 3.0 of the ebXML Messaging Service version 3.0 OASIS  
241 Standard [**ebMS 3.0**] instead of version 2.0 [**ebMS 2.0**].

### 242 4.4 Conformance as a DSS Server using ebMS 3.0

243 The conformance requirements for a *DSS Server using ebMS 3.0* are the same as for a *DSS Server  
244 using ebMS 2.0*, except that it support version 3.0 of the ebXML Messaging Service version 3.0 OASIS  
245 Standard [**ebMS 3.0**] instead of version 2.0 [**ebMS 2.0**].

247

## A. Sample ebXML Messaging 2.0 SOAP envelopes

248 This non-normative appendix contains sample SOAP envelopes with version 2.0 ebXML extension  
 249 elements that reference a DSS <SignRequest> message (section A.1) and the response ebXML  
 250 message containing the DSS <SignResponse> message (section A.2). The enveloping MIME structures  
 251 and the DSS messages are omitted.

252 **A.1 Sample SOAP envelope for DSS SignRequest message**

```

253 <?xml version="1.0" encoding="UTF-8"?>
254 <s11:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" 
255   xmlns:xlink="http://www.w3.org/1999/xlink"
256   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
257   xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
258   http://www.oasis-open.org/committees/ebxml-msg/schema/envelope.xsd">
259     <s11:Header xmlns:eb2="http://www.oasis-open.org/committees/ebxml-
260   msg/schema/msg-header-2_0.xsd"
261       xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
262   msg/schema/msg-header-2_0.xsd
263       http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd">
264       <eb2:MessageHeader eb2:id="ID18890041183727249323localhost"
265         eb2:version="2.0"
266           s11:mustUnderstand="1">
267             <eb2:From>
268               <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
269   type:0088">1234567890123</eb2:PartyId>
270               <eb2:Role>DSSClient</eb2:Role>
271             </eb2:From>
272             <eb2:To>
273               <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
274   type:0088">3210987654321</eb2:PartyId>
275               <eb2:Role>DSSServer</eb2:Role>
276             </eb2:To>
277             <eb2:CPAId>CPAID_3210987654321_1234567890123_0001</eb2:CPAId>
278             <eb2:ConversationId>89612fb6-08ba-49c6-aff2-
279   8ddf81988495</eb2:ConversationId>
280             <eb2:Service>urn:oasis:names:tc:dss:1.0</eb2:Service>
281             <eb2:Action>SignRequest</eb2:Action>
282             <eb2:MessageData>
283               <eb2:MessageId>d8afbc35-2bc1-11dc-8605-
284   000c29eb4f66@localhost.localdomain</eb2:MessageId>
285               <eb2:Timestamp>2007-07-06T13:07:29.314Z</eb2:Timestamp>
286             </eb2:MessageData>
287             <eb2:DuplicateElimination/>
288           </eb2:MessageHeader>
289           <eb2:AckRequested eb2:id="ID137206991183727249317localhost"
290             eb2:signed="false" eb2:version="2.0"
291               s11:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
292             s11:mustUnderstand="1"/>
293           </s11:Header>
294           <s11:Body xmlns:eb="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-
295   header-2_0.xsd"
296             xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
297   msg/schema/msg-header-2_0.xsd http://www.oasis-open.org/committees/ebxml-
298   msg/schema/msg-header-2_0.xsd">
299             <eb2:Manifest eb2:id="ID59975701183727249325localhost" eb2:version="2.0">
300               <eb2:Reference eb2:id="ID177805471183727249323localhost"
301                 xlink:href="cid:A1183727249322.85513@localhost_cn"
302               xlink:type="simple"/>
303             </eb2:Manifest>

```

```
304     </s11:Body>
305 </s11:Envelope>
```

## 306 A.2 Sample SOAP envelope for DSS SignResponse message

```
307 <?xml version="1.0" encoding="UTF-8"?>
308 <s11:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" 
309   xmlns:xlink="http://www.w3.org/1999/xlink"
310   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
311     xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
312     http://www.oasis-open.org/committees/ebxml-msg/schema/envelope.xsd">
313   <s11:Header xmlns:eb="http://www.oasis-open.org/committees/ebxml-
314 msg/schema/msg-header-2_0.xsd"
315     xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
316 msg/schema/msg-header-2_0.xsd
317     http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd">
318     <eb2:MessageHeader eb2:id="ID283962961183731565500Vega" eb2:version="2.0"
319       s11:mustUnderstand="1">
320       <eb2:From>
321         <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
322 type:0088">3210987654321</eb2:PartyId>
323         <eb2:Role>DSSServer</eb2:Role>
324       </eb2:From>
325       <eb2:To>
326         <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
327 type:0088">1234567890123</eb2:PartyId>
328         <eb2:Role>DSSClient</eb2:Role>
329       </eb2:To>
330       <eb2:CPAId>CPAID_3210987654321_1234567890123_0001</eb2:CPAId>
331       <eb2:ConversationId>89612Fb6-08ba-49c6-aff2-
332 8ddf81988495</eb2:ConversationId>
333       <eb2:Service>urn:oasis:names:tc:dss:1.0</eb2:Service>
334       <eb2:Action>SignResponse</eb2:Action>
335       <eb2:MessageData>
336
337     <eb2:MessageId>M1183731565500.20294@vega_cn6692455690889293175</eb2:MessageId>
338       <eb2:Timestamp>2007-07-06T14:19:25.500Z</eb2:Timestamp>
339       <eb2:RefToMessageId>d8afbc35-2bc1-11dc-8605-
340 000c29eb4f66@localhost.localdomain</eb2:RefToMessageId>
341       <eb2:MessageData>
342       <eb2:DuplicateElimination/>
343     </eb2:MessageHeader>
344     <eb2:AckRequested eb2:id="ID187545611183731565500Vega" eb2:signed="false"
345 eb2:version="2.0"
346       s11:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
347     s11:mustUnderstand="1"/>
348   </s11:Header>
349   <s11:Body xmlns:eb="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-
350 header-2_0.xsd"
351     xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
352 msg/schema/msg-header-2_0.xsd http://www.oasis-open.org/committees/ebxml-
353 msg/schema/msg-header-2_0.xsd">
354     <eb2:Manifest eb2:id="ID118631211183731565500Vega" eb2:version="2.0">
355       <eb2:Reference eb2:id="ID67656811183731565500Vega"
356         xlink:href="cid:A1183731565500.20296@vega_cn"
357         xlink:type="simple"/>
358       </eb2:Manifest>
359     </s11:Body>
360 </s11:Envelope>
```

## B. Sample Collaboration Protocol Agreement

362 The OASIS ebXML Collaboration Protocol Profiles and Agreements version 2.0 technical specification  
 363 [**ebCPA 2.0**] is an OASIS standard that defines an XML language to encode and exchange the technical  
 364 configuration parameters used to exchange ebXML version 2.0 messages. Implementations of ebXML  
 365 messaging may use CPAs to configure ebXML message handlers.

366 The following non-normative sample CPA defines the agreement between the sample DSS client and  
 367 server used to create the sample messages displayed in section A. The certificate details have been  
 368 omitted to simplify the example.

369

```

370 <?xml version="1.0" encoding="UTF-8"?>
371 <cpa2:CollaborationProtocolAgreement xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
372   xmlns:cpa2="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
373   xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xlink="http://www.w3.org/1999/xlink"
374   xmlns:axsl="http://www.w3.org/1999/XSL/TransformAlias"
375   xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
376 http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
377   cpa2:cpaid="CPAID_3210987654321_1234567890123_0001" cpa2:version="1.0">
378     <cpa2:Status cpa2:value="agreed"/>
379     <cpa2:Start>2007-06-12T00:00:00Z</cpa2:Start>
380     <cpa2:End>2008-06-12T00:00:00Z</cpa2:End>
381     <cpa2:PartyInfo cpa2:partyName="Company X"
382       cpa2:defaultMshChannelId="Client_defaultDeliveryChannel_ProfileReliableMessaging"
383       cpa2:defaultMshPackageId="defaultPackage_Profile">
384       <cpa2:PartyId cpa2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
385 type:0088">1234567890123</cpa2:PartyId>
386       <cpa2:PartyRef xlink:href="http://www.company_X.com/" />
387       <cpa2:CollaborationRole>
388         <cpa2:ProcessSpecification cpa2:name="Digital Signature Services" cpa2:version="0.1"
389           xlink:href="http://docs.oasis-open.org/dss/v1.0/"
390           cpa2:uuid="http://docs.oasis-open.org/dss/v1.0/"/>
391         <cpa2:Role cpa2:name="DSSClient" xlink:href="http://docs.oasis-open.org/dss/v1.0/" />
392         <cpa2:ServiceBinding>
393           <cpa2:Service>urn:oasis:names:tc:dss:1.0</cpa2:Service>
394           <cpa2:CanSend>
395             <cpa2:ThisPartyActionBinding cpa2:id="Client_S_SignRequest"
396               cpa2:action="SignRequest" cpa2:packageId="defaultPackage_Profile">
397               <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
398                 cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
399                 cpa2:isIntelligibleCheckRequired="false"
400                 cpa2:isNonRepudiationReceiptRequired="false"
401                 cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
402                 cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>
403
404             <cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
405             </cpa2:ThisPartyActionBinding>
406
407             <cpa2:OtherPartyActionBinding>Server_R_SignRequest</cpa2:OtherPartyActionBinding>
408               </cpa2:CanSend>
409               <cpa2:CanSend>
410                 <cpa2:ThisPartyActionBinding cpa2:id="Client_S_VerifyRequest"
411                   cpa2:action="VerifyRequest" cpa2:packageId="defaultPackage_Profile">
412                   <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
413                     cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
414                     cpa2:isIntelligibleCheckRequired="false"
415                     cpa2:isNonRepudiationReceiptRequired="false"
416                     cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
417                     cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>
418
419             <cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
420             </cpa2:ThisPartyActionBinding>
421
422             <cpa2:OtherPartyActionBinding>Server_R_VerifyRequest</cpa2:OtherPartyActionBinding>
423               </cpa2:CanSend>
```

```

424
425     <cpa2:CanReceive>
426         <cpa2:ThisPartyActionBinding cpa2:id="Client_R_SignResponse"
427             cpa2:action="SignResponse" cpa2:packageId="defaultPackage_Profile">
428             <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
429                 cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
430                 cpa2:isIntelligibleCheckRequired="false"
431                 cpa2:isNonRepudiationReceiptRequired="false"
432                 cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
433                 cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>
434
435 <cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
436     </cpa2:ThisPartyActionBinding>
437
438 <cpa2:OtherPartyActionBinding>Server_S_SignResponse</cpa2:OtherPartyActionBinding>
439     </cpa2:CanReceive>
440     <cpa2:CanReceive>
441         <cpa2:ThisPartyActionBinding cpa2:id="Client_R_VerifyResponse"
442             cpa2:action="VerifyResponse" cpa2:packageId="defaultPackage_Profile">
443             <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
444                 cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
445                 cpa2:isIntelligibleCheckRequired="false"
446                 cpa2:isNonRepudiationReceiptRequired="false"
447                 cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
448                 cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>
449
450 <cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
451     </cpa2:ThisPartyActionBinding>
452
453 <cpa2:OtherPartyActionBinding>Server_S_VerifyResponse</cpa2:OtherPartyActionBinding>
454     </cpa2:CanReceive>
455     </cpa2:ServiceBinding>
456     </cpa2:CollaborationRole>
457     <cpa2:Certificate cpa2:certId="X_ServerCert">
458         <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
459             <!-- Details omitted -->
460         </cpa2:Certificate>
461         <cpa2:Certificate cpa2:certId="X_ClientCert">
462             <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
463                 <!-- Details omitted -->
464             </cpa2:Certificate>
465             <cpa2:SecurityDetails cpa2:securityId="X_TransportSecurity"/>
466             <cpa2:DeliveryChannel
467                 cpa2:channelId="Client_defaultDeliveryChannel_ProfileReliableMessaging"
468                 cpa2:docExchangeId="Client_ReliableMessaging"
469                 cpa2:transportId="Client_transport_HTTPS">
470                 <cpa2:MessagingCharacteristics cpa2:syncReplyMode="none" cpa2:ackRequested="always"
471                     cpa2:actor="urn: oasis:names:tc:ebxml-msg:actor:toPartyMSH"
472                     cpa2:ackSignatureRequested="never" cpa2:duplicateElimination="always"/>
473             </cpa2:DeliveryChannel>
474             <cpa2:Transport cpa2:transportId="Client_transport_HTTPS">
475                 <cpa2:TransportSender>
476                     <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
477                     <cpa2:TransportClientSecurity>
478                         <cpa2:TransportSecurityProtocol
479                             cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
480                             <cpa2:ClientCertificateRef cpa2:certId="X_ClientCert"/>
481                             <cpa2:ServerSecurityDetailsRef cpa2:securityId="X_TransportSecurity"/>
482                     </cpa2:TransportClientSecurity>
483                 <cpa2:TransportReceiver>
484                     <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
485                     <cpa2:Endpoint cpa2:uri="http://company_X.com:4080/exchange/1234567890123"
486                         cpa2:type="allPurpose"/>
487                     <cpa2:TransportServerSecurity
488                         <cpa2:TransportSecurityProtocol
489                             cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
490                             <cpa2:ServerCertificateRef cpa2:certId="X_ServerCert"/>
491                             <cpa2:ClientSecurityDetailsRef cpa2:securityId="X_TransportSecurity"/>
492                     </cpa2:TransportServerSecurity>
493                 </cpa2:TransportReceiver>
494             </cpa2:Transport>

```

```

495     <cpa2:DocExchange cpa2:docExchangeId="Client_ReliableMessaging">
496         <cpa2:ebXMLSenderBinding cpa2:version="2.0">
497             <cpa2:ReliableMessaging>
498                 <cpa2:Retries>8</cpa2:Retries>
499                 <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
500                 <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
501             </cpa2:ReliableMessaging>
502             <cpa2:PersistDuration>PID</cpa2:PersistDuration>
503         </cpa2:ebXMLSenderBinding>
504         <cpa2:ebXMLReceiverBinding cpa2:version="2.0">
505             <cpa2:ReliableMessaging>
506                 <cpa2:Retries>8</cpa2:Retries>
507                 <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
508                 <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
509             </cpa2:ReliableMessaging>
510             <cpa2:PersistDuration>PID</cpa2:PersistDuration>
511         </cpa2:ebXMLReceiverBinding>
512     </cpa2:DocExchange>
513 </cpa2:PartyInfo>
514 <cpa2:PartyInfo cpa2:partyName="Company Y"
515     cpa2:defaultMshChannelId="Server_defaultDeliveryChannel_ProfileReliableMessaging"
516     cpa2:defaultMshPackageId="defaultPackage_Profile">
517     <cpa2:PartyId cpa2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
518 type:0088">3210987654321</cpa2:PartyId>
519     <cpa2:PartyRef xlink:href="http://www.company_y.com"/>
520     <cpa2:CollaborationRole>
521         <cpa2:ProcessSpecification cpa2:name="Digital Signature Services" cpa2:version="0.1"
522             xlink:href="http://docs.oasis-open.org/dss/v1.0/"
523             cpa2:uuid="http://docs.oasis-open.org/dss/v1.0/" />
524         <cpa2:Role cpa2:name="DSSServer" xlink:href="http://docs.oasis-open.org/dss/v1.0/" />
525         <cpa2:ServiceBinding>
526             <cpa2:Service>urn:oasis:names:tc:dss:1.0</cpa2:Service>
527             <cpa2:CanSend>
528                 <cpa2:ThisPartyActionBinding cpa2:id="Server_S_SignResponse"
529                     cpa2:action="SignResponse" cpa2:packageId="defaultPackage_Profile">
530                     <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
531                         cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
532                         cpa2:isIntelligibleCheckRequired="false"
533                         cpa2:isNonRepudiationReceiptRequired="false"
534                         cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
535                         cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D" />
536
537 <cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
538     </cpa2:ThisPartyActionBinding>
539
540 <cpa2:OtherPartyActionBinding>Client_R_SignResponse</cpa2:OtherPartyActionBinding>
541     </cpa2:CanSend>
542     <cpa2:CanSend>
543         <cpa2:ThisPartyActionBinding cpa2:id="Server_S_VerifyResponse"
544             cpa2:action="VerifyResponse" cpa2:packageId="defaultPackage_Profile">
545             <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
546                 cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
547                 cpa2:isIntelligibleCheckRequired="false"
548                 cpa2:isNonRepudiationReceiptRequired="false"
549                 cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
550                 cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D" />
551
552 <cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
553     </cpa2:ThisPartyActionBinding>
554
555 <cpa2:OtherPartyActionBinding>Client_R_VerifyResponse</cpa2:OtherPartyActionBinding>
556     </cpa2:CanSend>
557     <cpa2:CanReceive>
558         <cpa2:ThisPartyActionBinding cpa2:id="Server_R_SignRequest"
559             cpa2:action="SignRequest" cpa2:packageId="defaultPackage_Profile">
560             <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
561                 cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
562                 cpa2:isIntelligibleCheckRequired="false"
563                 cpa2:isNonRepudiationReceiptRequired="false"
564                 cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
565                 cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D" />

```

---

```

566
567 <cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
568     </cpa2:ThisPartyActionBinding>
569
570 <cpa2:OtherPartyActionBinding>Client_S_SignRequest</cpa2:OtherPartyActionBinding>
571     </cpa2:CanReceive>
572     <cpa2:CanReceive>
573         <cpa2:ThisPartyActionBinding cpa2:id="Server_R_VerifyRequest"
574             cpa2:action="VerifyRequest" cpa2:packageId="defaultPackage_Profile">
575             <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
576                 cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
577                 cpa2:isIntelligibleCheckRequired="false"
578                 cpa2:isNonRepudiationReceiptRequired="false"
579                 cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
580                 cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>
581
582 <cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
583     </cpa2:ThisPartyActionBinding>
584
585 <cpa2:OtherPartyActionBinding>Client_S_VerifyRequest</cpa2:OtherPartyActionBinding>
586     </cpa2:CanReceive>
587     </cpa2:ServiceBinding>
588     </cpa2:CollaborationRole>
589     <cpa2:Certificate cpa2:certId="Y_ServerCert">
590         <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
591             <!-- Details omitted -->
592         </cpa2:Certificate>
593         <cpa2:Certificate cpa2:certId="Y_ClientCert">
594             <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
595                 <!-- Details omitted -->
596             </cpa2:Certificate>
597             <cpa2:SecurityDetails cpa2:securityId="Y_TransportSecurity"/>
598             <cpa2:DeliveryChannel
599                 cpa2:channelId="Server_defaultDeliveryChannel_ProfileReliableMessaging"
600                 cpa2:docExchangeId="Server_ReliableMessaging"
601                 cpa2:transportId="Server_transport_HTTPS">
602                 <cpa2:MessagingCharacteristics cpa2:syncReplyMode="none" cpa2:ackRequested="always"
603                     cpa2:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
604                     cpa2:ackSignatureRequested="never" cpa2:duplicateElimination="always"/>
605             </cpa2:DeliveryChannel>
606             <cpa2:Transport cpa2:transportId="Server_transport_HTTPS">
607                 <cpa2:TransportSender>
608                     <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
609                     <cpa2:TransportClientSecurity>
610                         <cpa2:TransportSecurityProtocol
611                             cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
612                             <cpa2:ClientCertificateRef cpa2:certId="Y_ClientCert"/>
613                             <cpa2:ServerSecurityDetailsRef cpa2:securityId="Y_TransportSecurity"/>
614                         </cpa2:TransportClientSecurity>
615                     </cpa2:TransportSender>
616                     <cpa2:TransportReceiver>
617                         <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
618                         <cpa2:Endpoint cpa2:uri="http://company_y.com:4080/exchange/3210987654321"
619                             cpa2:type="allPurpose"/>
620                         <cpa2:TransportServerSecurity>
621                             <cpa2:TransportSecurityProtocol
622                                 cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
623                                 <cpa2:ServerCertificateRef cpa2:certId="Y_ServerCert"/>
624                                 <cpa2:ClientSecurityDetailsRef cpa2:securityId="Y_TransportSecurity"/>
625                             </cpa2:TransportServerSecurity>
626                         </cpa2:TransportReceiver>
627                     </cpa2:Transport>
628                     <cpa2:DocExchange cpa2:docExchangeId="Server_ReliableMessaging">
629                         <cpa2:ebXMLSenderBinding cpa2:version="2.0">
630                             <cpa2:ReliableMessaging>
631                                 <cpa2:Retries>8</cpa2:Retries>
632                                 <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
633                                 <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
634                             </cpa2:ReliableMessaging>
635                             <cpa2:PersistDuration>P1D</cpa2:PersistDuration>
636                         </cpa2:ebXMLSenderBinding>

```

---

```
637      <cpa2:ebXMLReceiverBinding cpa2:version="2.0">
638          <cpa2:ReliableMessaging>
639              <cpa2:Retries>8</cpa2:Retries>
640              <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
641              <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
642          </cpa2:ReliableMessaging>
643          <cpa2:PersistDuration>P1D</cpa2:PersistDuration>
644      </cpa2:ebXMLReceiverBinding>
645      </cpa2:DocExchange>
646  </cpa2:PartyInfo>
647  <cpa2:SimplePart cpa2:id="DefaultSimplePart" cpa2:mimetype="application/xml"/>
648  <cpa2:Packaging cpa2:id="defaultPackage_Profile">
649      <cpa2:ProcessingCapabilities cpa2:parse="true" cpa2:generate="true"/>
650      <cpa2:CompositeList>
651          <cpa2:Composite cpa2:id="DefaultComposite" cpa2:mimetype="type=text/xml">
652              <cpa2:Constituent cpa2:idref="DefaultSimplePart"/>
653          </cpa2:Composite>
654      </cpa2:CompositeList>
655  </cpa2:Packaging>
656</cpa2:CollaborationProtocolAgreement>
```

657

## C. Revision History

658

[optional; should not be included in OASIS Standards]

659

Revision	Date	Editor	Changes Made
0.1	2007-11-16	Pim van der Eijk, Ernst Jan van Nigtevecht	Initial Draft.
0.2	2008-02-18	Pim van der Eijk	Updated and restructured based on review by Juan Carlos Cruellas
0.3	2008-04-14	Pim van der Eijk, TC Chairs	Edits to reflect Committee Draft status
0.4	2008-04-27	Stefan Drees	Minor edits in normative references (obsoleted RFC, SOAP and applying bibliographic style)
0.5	2008-04-28	Stefan Drees	Added a note on referencing obsoleted or historical standards as alignment to dss core.
0.6	2008-05-09	Pim van der Eijk	Feedback from OASIS TC admin; conformance section added.

660

661