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

Specification URIs:
This Version:
http://docs.oasis-open.org/dss-x/profiles/ebxml/v1.0/cd01/oasis-dss-1.0-profiles-ebxml-cd01.doc
(Authoritative)
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.ede>
Stefan Drees, Individual Member, <stefan@drees.name>.

Editor(s):
Pim van der Eijk, *Sonnenglanz Consulting BV,* <pvde@sonnenglanz.net>
Ernst Jan van Nigtevecht, *Sonnenglanz Consulting BV,* <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

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

Mappings from Digital Signature Services (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 1.0 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 either version 2.0 **[ebMS 2.0]** or version 3.0 **[ebMS 3.0]**.

1.1 Benefits and intended use

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

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

1.2 Scope

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

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

1.3 Terminology

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

1.4 Namespaces

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	**[ebCPA 2.0]**

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]**

In the context of this profile, the differences between version 2.0 and version 3.0 of ebXML Messaging are limited. The generic prefix *eb* will be used to refer to either version in situations where elements are 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]**

1.5 Normative References

[DSS Core] S. Drees et al., *Digital Signature Service Core Protocols, Elements and Bindings*. Version 1.0. http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.pdf. OASIS Standard, 11 April 2007.

[ebMS 2.0] *OASIS ebXML Messaging Services*. Version 2.0. http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS_v2_0.pdf. OASIS Standard, 1 April 2002.

[ebMS 3.0] P. Wenzel et al., *OASIS ebXML Messaging Services v3.0: Part 1, Core Features*. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/ebms_core-3.0-spec-os.pdf. OASIS Standard, 1 October 2007.

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

[RFC 2246] T Dierks, C. Allen., *The TLS Protocol Version 1.0.* http://www.ietf.org/rfc/rfc2246.txt. IETF RFC 2246, January 1999.

[RFC 2822] P. Resnick, *Internet Message Format*. http://www.ietf.org/rfc/rfc2822.txt. IETF RFC 2822, April 2001.

[SOAP 1.1] D. Box et al., *Simple Object Access Protocol (SOAP) 1.1*. http://www.w3.org/TR/2000/NOTE-SOAP-20000508/. W3C Note, 08 May 2000.

[SOAP 1.2] M. Gudgin et al., *SOAP Version 1.2 Part 1: Messaging*. http://www.w3.org/TR/2003/REC-soap12-part1-20030624/. W3C Recommendation, 24 June 2003.

[SOAPATTACH] J. Barton et al., *SOAP Messages with Attachments*. http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211. W3C Note, 11 December 2000.

1.6 Non-Normative References

[ebCPA 2.0] *ebXML Collaboration-Protocol Profile and Agreement*. Version 2.0. http://www.oasis-open.org/committees/ebxml-cppa/documents/ebcpp-2.0.pdf. OASIS Standard, 23 September 2002.

[RFC 4346] T Dierks, C. Allen., *The Transport Layer Security (TLS) Protocol Version 1.1.* http://www.ietf.org/rfc/rfc4346.txt. IETF RFC 4346, April 2006.

[SOAP 1.2(2nd)] M. Gudgin et al., *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)*. http://www.w3.org/TR/2007/REC-soap12-part1-20070427/. W3C Recommendation, 27 April 2007.

2 Exchanging DSS Messages using the ebXML Messaging Service

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

2.1 DSS message exchanges

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

- A signature *generation* message exchange. A DSS client can send a message to invoke a `SignRequest` action on a DSS server. The DSS server responds by returning a `SignResponse` DSS response message.
- A signature *verification* message exchange. A DSS client can send a message to invoke a `VerifyRequest` action on a DSS server. The DSS server responds by returning a `VerifyResponse` DSS response message.

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

2.1.1 Element <MessageId> and <RefToMessageId>

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

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

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

2.1.2 Element <ConversationId>

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

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

2.2 Other Header Elements

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

2.2.1 Service

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

2.2.2 Action

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

- SignRequest
- SignResponse
- VerifyRequest
- VerifyResponse

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

2.2.3 Role

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

- DSSClient for the DSS client partner
- DSSServer for the DSS server partner

2.3 Packaging

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

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

```
<dss:Document ID="doc1">
    <dss:Base64Data MimeType="application/pdf"> ... </dss:Base64Data>
</dss:Document>
```

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

2.3.1 Packaging in ebXML Messaging version 2.0

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

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

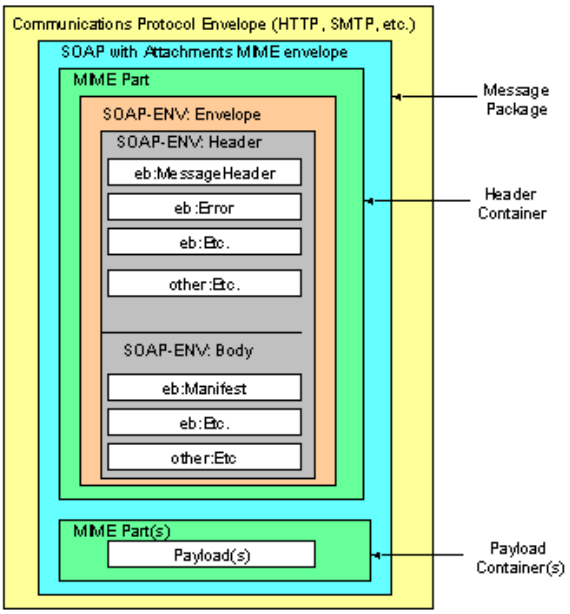


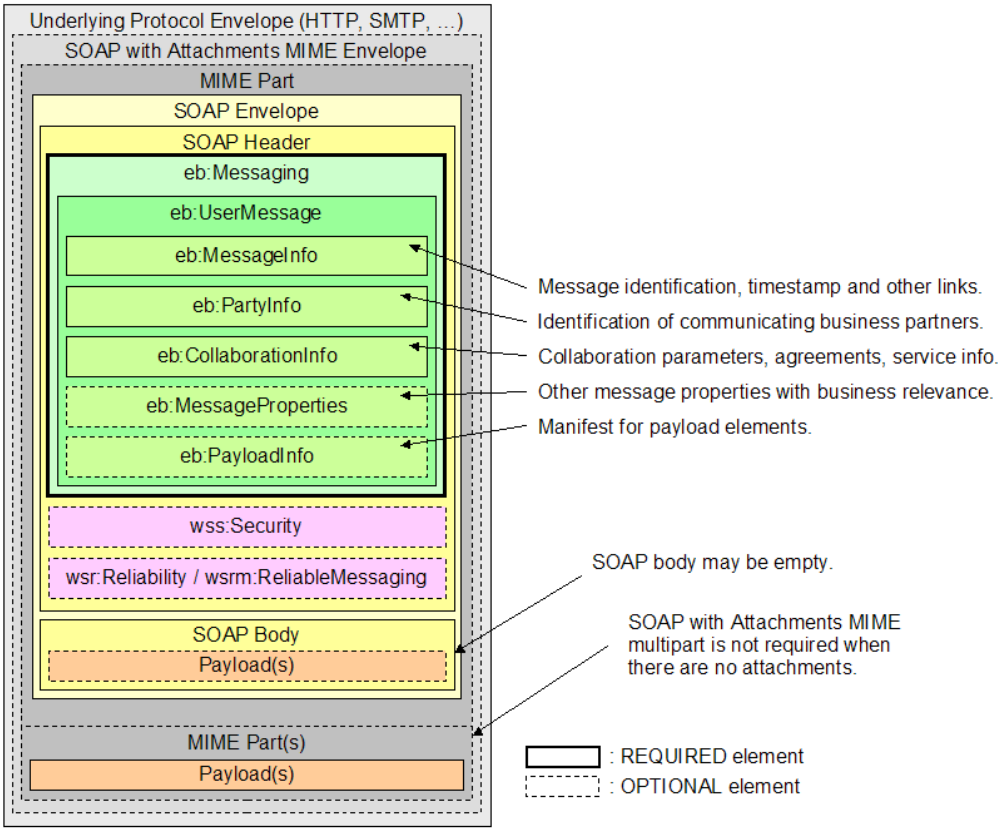
Figure 1 ebXML Messaging version 2 message structure

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

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

2.3.2 Packaging in ebXML Messaging version 3.0

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

3 Security Binding

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.

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]**.

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.

4 Conformance

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

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

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

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

4.1 Conformance as a DSS Client using ebMS 2.0

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

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

4.2 Conformance as a DSS Server using ebMS 2.0

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

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

4.3 Conformance as a DSS Client using ebMS 3.0

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

4.4 Conformance as a DSS Server using ebMS 3.0

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

A. Sample ebXML Messaging 2.0 SOAP envelopes

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

A.1 Sample SOAP envelope for DSS SignRequest message

```
<?xml version="1.0" encoding="UTF-8"?>
<s11:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
http://www.oasis-open.org/committees/ebxml-msg/schema/envelope.xsd">
    <s11:Header xmlns:eb2="http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd"
        xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd
        http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd">
        <eb2:MessageHeader eb2:id="ID1889004118372724932 3localhost"
eb2:version="2.0"
            s11:mustUnderstand="1">
            <eb2:From>
                <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:0088">1234567890123</eb2:PartyId>
                <eb2:Role>DSSClient</eb2:Role>
            </eb2:From>
            <eb2:To>
                <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:0088">3210987654321</eb2:PartyId>
                <eb2:Role>DSSServer</eb2:Role>
            </eb2:To>
            <eb2:CPAId>CPAID_3210987654321_1234567890123_0001</eb2:CPAId>
            <eb2:ConversationId>89612fb6-08ba-49c6-aff2-
8ddf81988495</eb2:ConversationId>
            <eb2:Service>urn:oasis:names:tc:dss:1.0</eb2:Service>
            <eb2:Action>SignRequest</eb2:Action>
            <eb2:MessageData>
                <eb2:MessageId>d8afbc35-2bc1-11dc-8605-
000c29eb4f66@localhost.localdomain</eb2:MessageId>
                <eb2:Timestamp>2007-07-06T13:07:29.314Z</eb2:Timestamp>
            </eb2:MessageData>
            <eb2:DuplicateElimination/>
        </eb2:MessageHeader>
        <eb2:AckRequested eb2:id="ID13720699118372724931 7localhost"
eb2:signed="false" eb2:version="2.0"
            s11:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
s11:mustUnderstand="1"/>
    </s11:Header>
    <s11:Body xmlns:eb="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-
header-2_0.xsd"
        xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd">
        <eb2:Manifest eb2:id="ID5997570118372724932 5localhost" eb2:version="2.0">
            <eb2:Reference eb2:id="ID1778054711837272493 23localhost"
                xlink:href="cid:A1183727249322.85513@localhost_cn"
xlink:type="simple"/>
        </eb2:Manifest>
```

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

A.2 Sample SOAP envelope for DSS SignResponse message

```
<?xml version="1.0" encoding="UTF-8"?>
<s11:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
    http://www.oasis-open.org/committees/ebxml-msg/schema/envelope.xsd">
    <s11:Header xmlns:eb="http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd"
        xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd
        http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd">
        <eb2:MessageHeader eb2:id="ID2839629611837315655OOVega" eb2:version="2.0"
            s11:mustUnderstand="1">
            <eb2:From>
                <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:0088">3210987654321</eb2:PartyId>
                <eb2:Role>DSSServer</eb2:Role>
            </eb2:From>
            <eb2:To>
                <eb2:PartyId eb2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:0088">1234567890123</eb2:PartyId>
                <eb2:Role>DSSClient</eb2:Role>
            </eb2:To>
            <eb2:CPAId>CPAID_3210987654321_1234567890123_0001</eb2:CPAId>
            <eb2:ConversationId>89612fb6-08ba-49c6-aff2-
8ddf81988495</eb2:ConversationId>
            <eb2:Service>urn:oasis:names:tc:dss:1.0</eb2:Service>
            <eb2:Action>SignResponse</eb2:Action>
            <eb2:MessageData>

<eb2:MessageId>M1183731565500.20294@vega_cn669245569088929317 5</eb2:MessageId>
                <eb2:Timestamp>2007-07-06T14:19:25.500Z</eb2:Timestamp>
                <eb2:RefToMessageId>d8afbc35-2bc1-11dc-8605-
000c29eb4f66@localhost.localdomain</eb2:RefToMessageId>
            </eb2:MessageData>
            <eb2:DuplicateElimination/>
        </eb2:MessageHeader>
        <eb2:AckRequested eb2:id="ID1875456111837315655OOVega" eb2:signed="false"
eb2:version="2.0"
            s11:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
s11:mustUnderstand="1"/>
    </s11:Header>
    <s11:Body xmlns:eb="http://www.oasis-open.org/committees/ebxml-msg/schema/msg-
header-2_0.xsd"
        xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd http://www.oasis-open.org/committees/ebxml-
msg/schema/msg-header-2_0.xsd">
        <eb2:Manifest eb2:id="ID1186312111837315655OOVega" eb2:version="2.0">
            <eb2:Reference eb2:id="ID6765681111837315655OOVega"
                xlink:href="cid:A1183731565500.20296@vega_cn"
xlink:type="simple"/>
        </eb2:Manifest>
    </s11:Body>
</s11:Envelope>
```

B. Sample Collaboration Protocol Agreement

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

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

```
<?xml version="1.0" encoding="UTF-8"?>
<cpa2:CollaborationProtocolAgreement xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:cpa2="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:axsl="http://www.w3.org/1999/XSL/TransformAlias"
    xsi:schemaLocation="http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd
http://www.oasis-open.org/committees/ebxml-cppa/schema/cpp-cpa-2_0.xsd"
    cpa2:cpaid="CPAID_3210987654321_1234567890123_0001" cpa2:version="1.0">
    <cpa2:Status cpa2:value="agreed"/>
    <cpa2:Start>2007-06-12T00:00:00Z</cpa2:Start>
    <cpa2:End>2008-06-12T00:00:00Z</cpa2:End>
    <cpa2:PartyInfo cpa2:partyName="Company X"
        cpa2:defaultMshChannelId="Client_defaultDeliveryChannel_ProfileReliableMessaging"
        cpa2:defaultMshPackageId="defaultPackage_Profile">
        <cpa2:PartyId cpa2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:0088">1234567890123</cpa2:PartyId>
        <cpa2:PartyRef xlink:href="http://www.company_X.com/"/>
        <cpa2:CollaborationRole>
            <cpa2:ProcessSpecification cpa2:name="Digital Signature Services" cpa2:version="0.1"
                xlink:href="http://docs.oasis-open.org/dss/v1.0/"
                cpa2:uuid="http://docs.oasis-open.org/dss/v1.0/"/>
            <cpa2:Role cpa2:name="DSSClient" xlink:href="http://docs.oasis-open.org/dss/v1.0/"/>
            <cpa2:ServiceBinding>
                <cpa2:Service>urn:oasis:names:tc:dss:1.0</cpa2:Service>
                <cpa2:CanSend>
                    <cpa2:ThisPartyActionBinding cpa2:id="Client_S_SignRequest"
                        cpa2:action="SignRequest" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Server_R_SignRequest</cpa2:OtherPartyActionBinding>
                </cpa2:CanSend>
                <cpa2:CanSend>
                    <cpa2:ThisPartyActionBinding cpa2:id="Client_S_VerifyRequest"
                        cpa2:action="VerifyRequest" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Server_R_VerifyRequest</cpa2:OtherPartyActionBinding>
                </cpa2:CanSend>
```

```
                <cpa2:CanReceive>
                    <cpa2:ThisPartyActionBinding cpa2:id="Client_R_SignResponse"
                        cpa2:action="SignResponse" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Server_S_SignResponse</cpa2:OtherPartyActionBinding>
                </cpa2:CanReceive>
                <cpa2:CanReceive>
                    <cpa2:ThisPartyActionBinding cpa2:id="Client_R_VerifyResponse"
                        cpa2:action="VerifyResponse" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Client_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Server_S_VerifyResponse</cpa2:OtherPartyActionBinding>
                </cpa2:CanReceive>
            </cpa2:ServiceBinding>
        </cpa2:CollaborationRole>
        <cpa2:Certificate cpa2:certId="X_ServerCert">
            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
               <!-- Details omitted -->
        </cpa2:Certificate>
        <cpa2:Certificate cpa2:certId="X_ClientCert">
            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
               <!-- Details omitted -->
        </cpa2:Certificate>
        <cpa2:SecurityDetails cpa2:securityId="X_TransportSecurity"/>
        <cpa2:DeliveryChannel
            cpa2:channelId="Client_defaultDeliveryChannel_ProfileReliableMessaging"
            cpa2:docExchangeId="Client_ReliableMessaging"
            cpa2:transportId="Client_transport_HTTPS">
            <cpa2:MessagingCharacteristics cpa2:syncReplyMode="none" cpa2:ackRequested="always"
                cpa2:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
                cpa2:ackSignatureRequested="never" cpa2:duplicateElimination="always"/>
        </cpa2:DeliveryChannel>
        <cpa2:Transport cpa2:transportId="Client_transport_HTTPS">
            <cpa2:TransportSender>
                <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
                <cpa2:TransportClientSecurity>
                    <cpa2:TransportSecurityProtocol
cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
                    <cpa2:ClientCertificateRef cpa2:certId="X_ClientCert"/>
                    <cpa2:ServerSecurityDetailsRef cpa2:securityId="X_TransportSecurity"/>
                </cpa2:TransportClientSecurity>
            </cpa2:TransportSender>
            <cpa2:TransportReceiver>
                <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
                <cpa2:Endpoint cpa2:uri="http://company_X.com:4080/exchange/1234567890123"
                    cpa2:type="allPurpose"/>
                <cpa2:TransportServerSecurity>
                    <cpa2:TransportSecurityProtocol
cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
                    <cpa2:ServerCertificateRef cpa2:certId="X_ServerCert"/>
                    <cpa2:ClientSecurityDetailsRef cpa2:securityId="X_TransportSecurity"/>
                </cpa2:TransportServerSecurity>
            </cpa2:TransportReceiver>
        </cpa2:Transport>
```

```
        <cpa2:DocExchange cpa2:docExchangeId="Client_ReliableMessaging">
            <cpa2:ebXMLSenderBinding cpa2:version="2.0">
                <cpa2:ReliableMessaging>
                    <cpa2:Retries>8</cpa2:Retries>
                    <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
                    <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
                </cpa2:ReliableMessaging>
                <cpa2:PersistDuration>P1D</cpa2:PersistDuration>
            </cpa2:ebXMLSenderBinding>
            <cpa2:ebXMLReceiverBinding cpa2:version="2.0">
                <cpa2:ReliableMessaging>
                    <cpa2:Retries>8</cpa2:Retries>
                    <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
                    <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
                </cpa2:ReliableMessaging>
                <cpa2:PersistDuration>P1D</cpa2:PersistDuration>
            </cpa2:ebXMLReceiverBinding>
        </cpa2:DocExchange>
    </cpa2:PartyInfo>
    <cpa2:PartyInfo cpa2:partyName="Company Y"
        cpa2:defaultMshChannelId="Server_defaultDeliveryChannel_ProfileReliableMessaging"
        cpa2:defaultMshPackageId="defaultPackage_Profile">
        <cpa2:PartyId cpa2:type="urn:oasis:names:tc:ebxml-cppa:partyid-
type:0088">3210987654321</cpa2:PartyId>
        <cpa2:PartyRef xlink:href="http://www.company_y.com"/>
        <cpa2:CollaborationRole>
            <cpa2:ProcessSpecification cpa2:name="Digital Signature Services" cpa2:version="0.1"
                xlink:href="http://docs.oasis-open.org/dss/v1.0/"
                cpa2:uuid="http://docs.oasis-open.org/dss/v1.0/"/>
            <cpa2:Role cpa2:name="DSSServer" xlink:href="http://docs.oasis-open.org/dss/v1.0/"/>
            <cpa2:ServiceBinding>
                <cpa2:Service>urn:oasis:names:tc:dss:1.0</cpa2:Service>
                <cpa2:CanSend>
                    <cpa2:ThisPartyActionBinding cpa2:id="Server_S_SignResponse"
                        cpa2:action="SignResponse" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Client_R_SignResponse</cpa2:OtherPartyActionBinding>
                </cpa2:CanSend>
                <cpa2:CanSend>
                    <cpa2:ThisPartyActionBinding cpa2:id="Server_S_VerifyResponse"
                        cpa2:action="VerifyResponse" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Client_R_VerifyResponse</cpa2:OtherPartyActionBinding>
                </cpa2:CanSend>
                <cpa2:CanReceive>
                    <cpa2:ThisPartyActionBinding cpa2:id="Server_R_SignRequest"
                        cpa2:action="SignRequest" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>
```

```
<cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Client_S_SignRequest</cpa2:OtherPartyActionBinding>
                </cpa2:CanReceive>
                <cpa2:CanReceive>
                    <cpa2:ThisPartyActionBinding cpa2:id="Server_R_VerifyRequest"
                        cpa2:action="VerifyRequest" cpa2:packageId="defaultPackage_Profile">
                        <cpa2:BusinessTransactionCharacteristics cpa2:isAuthenticated="transient"
                            cpa2:isAuthorizationRequired="true" cpa2:isConfidential="transient"
                            cpa2:isIntelligibleCheckRequired="false"
                            cpa2:isNonRepudiationReceiptRequired="false"
                            cpa2:isNonRepudiationRequired="false" cpa2:isTamperProof="transient"
                            cpa2:timeToAcknowledgeReceipt="PT8H" cpa2:timeToPerform="P2D"/>

<cpa2:ChannelId>Server_defaultDeliveryChannel_ProfileReliableMessaging</cpa2:ChannelId>
                    </cpa2:ThisPartyActionBinding>

<cpa2:OtherPartyActionBinding>Client_S_VerifyRequest</cpa2:OtherPartyActionBinding>
                </cpa2:CanReceive>
            </cpa2:ServiceBinding>
        </cpa2:CollaborationRole>
        <cpa2:Certificate cpa2:certId="Y_ServerCert">
            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
               <!-- Details omitted -->
        </cpa2:Certificate>
        <cpa2:Certificate cpa2:certId="Y_ClientCert">
            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
               <!-- Details omitted -->
        </cpa2:Certificate>
        <cpa2:SecurityDetails cpa2:securityId="Y_TransportSecurity"/>
        <cpa2:DeliveryChannel
            cpa2:channelId="Server_defaultDeliveryChannel_ProfileReliableMessaging"
            cpa2:docExchangeId="Server_ReliableMessaging"
            cpa2:transportId="Server_transport_HTTPS">
            <cpa2:MessagingCharacteristics cpa2:syncReplyMode="none" cpa2:ackRequested="always"
                cpa2:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
                cpa2:ackSignatureRequested="never" cpa2:duplicateElimination="always"/>
        </cpa2:DeliveryChannel>
        <cpa2:Transport cpa2:transportId="Server_transport_HTTPS">
            <cpa2:TransportSender>
                <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
                <cpa2:TransportClientSecurity>
                    <cpa2:TransportSecurityProtocol
cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
                    <cpa2:ClientCertificateRef cpa2:certId="Y_ClientCert"/>
                    <cpa2:ServerSecurityDetailsRef cpa2:securityId="Y_TransportSecurity"/>
                </cpa2:TransportClientSecurity>
            </cpa2:TransportSender>
            <cpa2:TransportReceiver>
                <cpa2:TransportProtocol cpa2:version="1.1">HTTP</cpa2:TransportProtocol>
                <cpa2:Endpoint cpa2:uri="http://company_y.com:4080/exchange/3210987654321"
                    cpa2:type="allPurpose"/>
                <cpa2:TransportServerSecurity>
                    <cpa2:TransportSecurityProtocol
cpa2:version="1.0">TLS</cpa2:TransportSecurityProtocol>
                    <cpa2:ServerCertificateRef cpa2:certId="Y_ServerCert"/>
                    <cpa2:ClientSecurityDetailsRef cpa2:securityId="Y_TransportSecurity"/>
                </cpa2:TransportServerSecurity>
            </cpa2:TransportReceiver>
        </cpa2:Transport>
        <cpa2:DocExchange cpa2:docExchangeId="Server_ReliableMessaging">
            <cpa2:ebXMLSenderBinding cpa2:version="2.0">
                <cpa2:ReliableMessaging>
                    <cpa2:Retries>8</cpa2:Retries>
                    <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
                    <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
                </cpa2:ReliableMessaging>
                <cpa2:PersistDuration>P1D</cpa2:PersistDuration>
            </cpa2:ebXMLSenderBinding>
```

```
            <cpa2:ebXMLReceiverBinding cpa2:version="2.0">
                <cpa2:ReliableMessaging>
                    <cpa2:Retries>8</cpa2:Retries>
                    <cpa2:RetryInterval>PT3H</cpa2:RetryInterval>
                    <cpa2:MessageOrderSemantics>NotGuaranteed</cpa2:MessageOrderSemantics>
                </cpa2:ReliableMessaging>
                <cpa2:PersistDuration>P1D</cpa2:PersistDuration>
            </cpa2:ebXMLReceiverBinding>
        </cpa2:DocExchange>
    </cpa2:PartyInfo>
    <cpa2:SimplePart cpa2:id="DefaultSimplePart" cpa2:mimetype="application/xml"/>
    <cpa2:Packaging cpa2:id="defaultPackage_Profile">
        <cpa2:ProcessingCapabilities cpa2:parse="true" cpa2:generate="true"/>
        <cpa2:CompositeList>
            <cpa2:Composite cpa2:id="DefaultComposite" cpa2:mimetype="type=text/xml">
                <cpa2:Constituent cpa2:idref="DefaultSimplePart"/>
            </cpa2:Composite>
        </cpa2:CompositeList>
    </cpa2:Packaging>
</cpa2:CollaborationProtocolAgreement>
```

C. Revision History

[optional; should not be included in OASIS Standards]

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.