ebXML Messaging Protocol Binding for RegRep Version 1.0

Committee Specification 01

09 March 2021

This stage:
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/cs01/ebrr-ebms-v1.0-cs01.odt (Authoritative)
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/cs01/ebrr-ebms-v1.0-cs01.html
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/cs01/ebrr-ebms-v1.0-cs01.pdf

Previous stage:
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/csd01/ebrr-ebms-v1.0-csd01.odt (Authoritative)
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/csd01/ebrr-ebms-v1.0-csd01.html
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/csd01/ebrr-ebms-v1.0-csd01.pdf

Latest stage:
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/ebrr-ebms-v1.0.odt (Authoritative)
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/ebrr-ebms-v1.0.html
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/ebrr-ebms-v1.0.pdf

Technical Committee:
OASIS ebXML Core (ebCore) TC

Chairs:
Pim van der Eijk (pvde@sonnenglanz.net), Sonnenglanz Consulting
Sander Fieten (sander@chasquis-consulting.com), Individual member

Editors:
Nikola Stojanovic (nikola.stojanovic@acm.org), Individual member
Pim van der Eijk (pvde@sonnenglanz.net), Sonnenglanz Consulting

Related work:
This specification is related to:

Abstract:
The OASIS ebXML Messaging Protocol Binding for RegRep Version 1.0 specifies a messaging protocol binding for the Registry Services of the OASIS ebXML RegRep Version 4.0 OASIS Standard. This binding is compatible with both the versions 2.0 and 3.0 of ebMS as well as the AS4 profile and complements the existing protocol bindings specified in OASIS RegRep Version 4.0.
Status:
This document was last revised or approved by the OASIS ebXML Core (ebCore) TC on the above date. The level of approval is also listed above. Check the "Latest stage" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ebcore#technical.

TC members should send comments on this specification to the TC’s email list. Others should send comments to the TC’s public comment list, after subscribing to it by following the instructions at the "Send A Comment" button on the Technical Committee’s web page at https://www.oasis-open.org/committees/ebcore/.

This specification is provided under the RF on Limited Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this Work Product, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC’s web page (https://www.oasis-open.org/committees/ebcore/ipr.php).

Note that any machine-readable content (Computer Language Definitions) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product’s prose narrative document(s), the content in the separate plain text file prevails.

Citation format:
When referencing this Work Product the following citation format should be used:

[RegRep-ebMS-v1.0]
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/cs01/ebrr-ebms-v1.0-cs01.html. Latest stage:
https://docs.oasis-open.org/ebcore/ebrr-ebms/v1.0/ebrr-ebms-v1.0.html.
# Table of Contents

1 Introduction ........................................................................................................................................ 5
   1.1 Overview ..................................................................................................................................... 5
   1.2 IPR Policy ................................................................................................................................... 5
   1.3 Terminology ............................................................................................................................... 5
   1.4 Normative References .................................................................................................................. 5
   1.5 Non-Normative References ......................................................................................................... 6

2 Binding for Registry Services ........................................................................................................... 7
   2.1 Introduction .................................................................................................................................. 7
   2.2 Layering Approach ...................................................................................................................... 7
   2.3 Registry Interface ......................................................................................................................... 7
   2.4 Collaboration Information ............................................................................................................. 8
   2.5 Exceptions .................................................................................................................................. 9
   2.6 Correlation .................................................................................................................................. 9
   2.7 Packaging .................................................................................................................................... 10
   2.8 Versions ...................................................................................................................................... 10

3 Safety, Security, and Data Protection Considerations ....................................................................... 12

4 Conformance ..................................................................................................................................... 13

Appendix A  Example Messages (Non-Normative) .................................................................................. 14
   Appendix A.1 Query Request Message ............................................................................................... 14
   Appendix A.2 Query Response Message .............................................................................................. 15

Appendix B  Acknowledgments (Non-Normative) .................................................................................. 17

Appendix C  Revision History (Non-Normative) ..................................................................................... 18
# 1 Introduction

## 1.1 Overview

The OASIS ebXML Messaging Protocol Binding for RegRep Version 1.0 specifies a messaging protocol binding for the OASIS ebXML RegRep Version 4.0 OASIS Standard [regrep-overview-v4.0]. It supports all the ebXML RegRep service interfaces specified in the OASIS ebXML RegRep Version 4.0 Part 2: Services and Protocols (ebRS) specification [regrep-rs-v4.0]. This specified binding is compatible with the version 2.0 [EBMS2] of ebMS, the core specification of version 3.0 of ebMS [EBMS3CORE] and the AS4 profile of ebMS3 [AS4-Profile]. It complements the existing protocol bindings specified in the OASIS ebXML RegRep Registry Services specification [regrep-rs-v4.0]. The goal of this specification is to allow users of RegRep to take advantage of the superior security, reliability and other advanced features of ebXML Messaging, and to allow users of ebXML Messaging to take advantage of the capabilities provided by RegRep.

## 1.2 IPR Policy

This specification is provided under the RF on Limited Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. 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 TC’s web page ([https://www.oasis-open.org/committees/ebcore/ipr.php](https://www.oasis-open.org/committees/ebcore/ipr.php)).

## 1.3 Terminology

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “NOT RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in BCP 14 [RFC2119] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 1.4 Normative References

1.5 Non-Normative References

[EBCOREISSUES] Issue tracker of the OASIS ebCore TC.
https://issues.oasis-open.org/projects/EBCORE/issues/

[EBMSISSUES] Issue tracker of the OASIS ebXML Messaging TC
https://issues.oasis-open.org/browse/EBXMLMSG/issues/
2 Binding for Registry Services

2.1 Introduction
This binding concerns the RegRep Registry Services [regrep-rs-v4.0]. It specifies:

- The approach adopted to layering the ebXML Messaging and RegRep functionality (see section 2.2).
- A mapping of RegRep operations to Message Exchange Patterns (MEPs; see section 2.3).
- A specification of predetermined values for some values in the eb:Messaging header and the P-Mode parameters that control their values (section 2.4).
- The way exceptions are exchanged (see section 2.5).
- Correlation of requests and responses (see section 2.6).
- Use of the SOAP-with-attachments envelope for the transmission of repository objects (see section 2.7).
- Versions of ebXML Messaging supported (see section 2.8).

Except where specified in this specification, this binding places no constraints on the use of ebXML Messaging features. In particular:

- WS-Security [WSSSMS] and using Transport Layer Security [RFC8446] MAY be used to provide message integrity, authentication, non-repudiation and confidentiality.
- This binding MAY also be used in conjunction with the SAML conformance clause for ebMS [ebMS-saml-conformance].
- Reliable messaging MAY be used to ensure guaranteed message delivery.
- Message transfer MAY use Push or Pull transport channel binding and MAY use HTTP or SMTP transport protocol bindings.

2.2 Layering Approach
This specification is based on a loose coupling between the RegRep and ebXML Messaging layers.

- The binding supports implementation using unmodified ebXML Message Service Handler (MSH) implementations, provided that in deployment in the MSH the Processing Mode definitions ([EBMS3CORE], chapter 4 and Appendix D) are configured in accordance with this specification.
- The binding supports exchange of RegRep payloads based on unmodified RegRep XML schemas defined in the RegRep standard [regrep-rim-v4.0,regrep-rs-v4.0]. This means all functionality of an existing RegRep implementation other than the message protocol binding (Web Services or REST) can be reused without modification.

As a consequence of this approach, each of RegRep or ebMS layers contains identifiers and correlation identifiers. These values are set independently (see section 2.6).

2.3 Registry Interface
The Registry Services, which are defined in the RegRep Registry Services specification [regrep-rs-v4.0], can be mapped to ebXML Message Exchange Patterns, as defined in section 2.2 of the ebMS3 Core Specification [EBMS3CORE]. In that specification, exchange patterns are classified as One Way exchanges or as Two Way exchanges.
The following ebRS interfaces map to Two Way Message Exchange Patterns:

- Query Manager Interface ([regrep-rs-v4.0], section 2).
- Lifecycle Manager Interface ([regrep-rs-v4.0], section 3).
- Validator Interface ([regrep-rs-v4.0], section 5).
- Cataloger Interface ([regrep-rs-v4.0], section 6).

The following ebRS interface maps to a One Way Exchange Pattern:

- Notification ([regrep-rs-v4.0], section 7).

For the Registry Services interfaces, which are modeled as operations in the RegRep WSDL definitions [regrep-wsdl-v4.0], the mapping is as follows:

- The *input* of the operation maps to the first leg of the ebMS exchange.
- The *output or fault* of the operation (if present) maps to the second leg of the ebMS exchange.

This specification does not constrain the transport channel and transport protocol bindings for the RegRep interface. Some examples of bindings are the following:

- For the interface that uses the One Way MEP, one possible binding uses an SMTP binding. Another possibility would be to use the One Way Push binding with HTTP.
- For the interfaces that use the Two Way MEP, one possible binding is to follow the synchronous SOAP protocol binding specified in section 13 of the RegRep registry services specification [regrep-rs-v4.0] and use a synchronous exchange over the HTTP protocol, in which the second leg uses the HTTP backchannel.
- For the interfaces that use the Two Way MEP, another possible binding involves the so-called Two-Way/Push-and-Push MEP (see section 2.2.8 in [EBMS3CORE]) that composes the choreographies of two One-Way/Push MEPs in opposite directions.

Constraints on channels and transport protocol bindings MAY be specified in profiles of this specification. Note that the AS4 profile of ebMS3 is limited to asynchronous exchanges and to the HTTP protocol. It therefore does not support the Two Way Sync binding. Many available AS4 implementations do support the Two-Way/Push-and-Push MEP.

This specification does not constrain values for the *eb:From/eb:Role and eb:To/eb:Role* elements.

### 2.4 Collaboration Information

In ebXML version 3.0, user messages have a mandatory *eb:CollaborationInfo* section that includes mandatory *eb:Service* and *eb:Action* elements. This profile specifies recommended values for these elements and for the *type* attribute of *eb:Service*.

- The value of the *type* attribute of the *eb:Service element in eb:CollaborationInfo SHOULD be set to `urn:oasis:names:tc:ebcore:ebrs:ebms:binding:1.0`. This indicates that the exchange uses the ebXML Messaging Protocol Binding for RegRep Version 1.0 as defined in this specification. These values can be configured using the *PMode[1].BusinessInfo.Service* and *PMode[2].BusinessInfo.Service* processing mode parameters.
- The content of the *eb:Service* and the *eb:Action* elements for the Registry Services interfaces SHOULD be set to the values specified in table 1.

1. The *Service* column specifies content of the *eb:Service element*. These values can be configured using the *PMode[1].BusinessInfo.Service* and *PMode[2].BusinessInfo.Service* processing mode parameters.
2. The Action [1] column specifies the value of the eb:Action element that SHOULD be used in the first leg message. This value can be configured for exchanges using the PMode[1]. BusinessInfo.Action processing mode parameter.

3. For Two Way MEPs, the Action [2] column specifies the value of the eb:Action element that SHOULD be used in the second leg message in the case of a successful exchange. This value can be configured for exchanges using the PMode[2].BusinessInfo.Action processing mode parameter. This column is not applicable in the case of One Way message exchanges.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Query Manager</td>
<td>QueryManager</td>
<td>ExecuteQueryRequest</td>
<td>ExecuteQueryResponse</td>
</tr>
<tr>
<td>LifeCycle Manager</td>
<td>LifecycleManager</td>
<td>RemoveObjectsRequest</td>
<td>RemoveObjectsResponse</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SubmitObjectsRequest</td>
<td>SubmitObjectsResponse</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UpdateObjectsRequest</td>
<td>UpdateObjectsResponse</td>
</tr>
<tr>
<td>Validator</td>
<td>Validator</td>
<td>ValidateObjectsRequest</td>
<td>ValidateObjectsResponse</td>
</tr>
<tr>
<td>Cataloger</td>
<td>Cataloger</td>
<td>CatalogObjectsRequest</td>
<td>CatalogObjectsResponse</td>
</tr>
<tr>
<td>Notification</td>
<td>Notification</td>
<td>OnNotificationRequest</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 1: Registry Services BusinessInfo Values

The Action [2] column does not cover the case of an exchange that resulted in an error situation. See section 2.5 on exchange of exceptions.

In all interfaces that are mapped to Two Way MEP, the second leg MUST be correlated with the first leg using cross-references, as explained in 2.6 below.

### 2.5 Exceptions

If an error is encountered by the receiving RegRep registry services handler during the processing of a RegRep request, the server MUST return an exception. To report the exception, it MUST be encoded as a rs:Exception child element of a rs:RegistryResponse element. This rs:RegistryResponse element MUST be set to be the message payload of an AS4 eb:UserMessage element which MUST be returned as an ebMS3 response user message. An error relating to RegRep registry services processing MUST NOT be returned as a rs:RegistryException SOAP Fault.

To differentiate error responses that contain rs:Exception elements from successful responses, the value of the eb:Action element in the eb:CollaborationInfo container in the response user message MUST be set to the value ExceptionResponse.

### 2.6 Correlation

Response messages contains independent correlation information at both the RegRep and ebMS layers.

- The value of the requestId attribute on rs:RegistryResponse MUST be set to the value of the id attribute on the registry request that the response relates to.
• The value of the \texttt{eb:RefToMessageId} element in the \texttt{eb:MessageInfo} container in the response user message MUST be set to the value of the \texttt{eb:MessageId} element in the \texttt{eb:MessageInfo} container in the request user message that it relates to.

This correlation information enables appropriate processing of response, also in situations where asynchronous transport protocol bindings are used and responses may arrive out-of-order, and at unpredictable times.

2.7 Packaging

The ebXML Messaging specification uses the SOAP-with-attachments specification to encode ebXML messages and their payloads. A single ebXML user message can have multiple payloads in arbitrary formats, including native binary formats that are not base64 encoded. This feature is useful for:

• Responses to queries (using the QueryManager interface) that carry registry objects.
• Submissions to the registry (using the LifecycleManager interface) that carry data to be registered.

In the QueryManager interface, the response message uses the \texttt{query:QueryResponse} object that contains a \texttt{rim:RegistryObjectList} that contains \texttt{rim:RegistryObject} elements. A \texttt{rim:RegistryObject} of type \texttt{rim:ExtrinsicObjectType} can include an embedded object or a \texttt{rim:RepositoryItemRef}. When using SOAP-with-attachments as used in ebXML Messaging, a repository item reference can link to a repository item that is carried as a separate MIME part. This has several advantages:

• The external payload, if it is in a non-textual binary format, can be carried in its native format, obviating the need for Base64 encoding and therefore reducing the size of the message.
• When using AS4, the external payload can be compressed, also reducing the size of the message payload.

An ebXML Message MUST include \texttt{eb:PartInfo} elements in the \texttt{eb:PayloadInfo} section of the \texttt{eb:UserMessage} for all payload parts. If a message contains both RegRep XML content and separate payload parts, the \texttt{eb:PayloadInfo} section MUST include \texttt{eb:PartInfo} elements for both the RegRep XML content and all other payload parts. The \texttt{eb:PartInfo} element relating to the RegRep XML content MUST precede any other \texttt{eb:PartInfo} elements in the \texttt{eb:PayloadInfo} section.

If the RegRep XML content is carried in the SOAP Body, the \texttt{href} attribute on the related \texttt{eb:PayloadInfo} MUST be absent as specified in section 5.2.2.13 of ebMS3 Core [EBMS3CORE].

In ebMS3 and AS4, the SOAP Body MAY remain empty. RegRep XML content MAY therefore also packaged in a separate MIME part. In that case, the related \texttt{eb:PayloadInfo} MUST contain an \texttt{href} attribute that references the MIME part containing the RegRep XML content using its MIME Content Identifier.

The syntax and semantics of RegRep messages, as defined in the RegRep Registry Services [regrep-rs-v4.0] and Registry Information Model [regrep-rim-v4.0] specifications, are not affected by this specification.

Note that ebXML Messaging MAY use XML Encryption to secure messages [XMLENC-CORE, XMLENC-CORE1]. When using XML Encryption, the content of MIME payload parts is in binary encrypted form.

2.8 Versions

This specification defines version 1.0 of ebXML Messaging Protocol Binding for RegRep. Messages using this version can be distinguished from any future versions of this specification can be recognized at runtime by checking the value for the \texttt{type} attribute on \texttt{eb:Service} element.

This specification covers ebXML Messaging Version 3.0 [EBMS3CORE] including the AS4 profile [AS4-Profile]. This specification MAY also be used in conjunction with version 2.0 of ebXML Messaging.
[EBMS3CORE] instead of version 3.0 or AS4, by applying the v2.0-v3.0 compatibility mapping specified in Appendix F of [EBMS3CORE].

Note that the SAML Conformance Clause for ebMS is specified for version 3.0 of ebMS only [ebMS-saml-conformance].
3 Safety, Security, and Data Protection Considerations

For data security and privacy reasons, message exchange based on this specification SHOULD use message layer security (Web Services Security, [WSSSMS]) and/or transport layer security ([RFC8446]).
4 Conformance

In order to claim conformance to the ebXML Messaging Protocol Binding for RegRep Version 1.0, an implementation:

- MUST be a conformant implementation of one of the supported versions of ebXML messages specified in section 2.8.
- MUST be a conformant implementation of the RegRep Registry Services [regrep-rs-v4.0].
- MUST satisfy all the mandatory requirements specified in section 2 of this specification.
Appendix A  Example Messages (Non-Normative)

This appendix contains a sample RegRep AS4 request and corresponding response. It is assumed WS-Security is used to secure the message, but details of the wsse:Security header are omitted for brevity.

In both examples, the RegRep XML contained is carried in the SOAP Body. As explained in section 2.7, the ebMS3 and AS4 specifications also allow the RegRep XML content to be carried in a separate MIME part.

Appendix A.1  Query Request Message

The following is a simplified example of an ebMS3 Query Request message.

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">    
  <env:Header>       
    <eb:Messaging xmlns:eb="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" env:mustUnderstand="true" wsu:Id="_210bca51-e9b3-4ee1-81e7-226949ab6ff6">       
      <eb:UserMessage mpc="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/defaultMPC">                      
        <eb:MessageInfo>                          
          <eb:Timestamp>2020-07-24T14:07:36.000Z</eb:Timestamp>        
          <eb:MessageId>8196c8e2@requester.example.com</eb:MessageId>      
        </eb:MessageInfo>                     
        <eb:PartyInfo>                         
          <eb:From>                            
            <eb:PartyId type="urn:oasis:names:tc:ebcore:partyid-type:iso6523:0088">1234567890</eb:PartyId>  
            <eb:Role>http://example.com/roles/Initiator</eb:Role>            
          </eb:From>                           
          <eb:To>                              
            <eb:PartyId type="urn:oasis:names:tc:ebcore:partyid-type:iso6523:0088">0987654321</eb:PartyId>  
            <eb:Role>http://example.com/roles/Responder</eb:Role>         
          </eb:To>                            
        </eb:PartyInfo>                      
        <eb:CollaborationInfo>                
          <eb:Service type="urn:oasis:names:tc:ebcore:ebrs:ebms:binding:1.0">QueryManager</eb:Service>  
          <eb:Action>ExecuteQueryRequest</eb:Action>                        
          <eb:ConversationId>6C24403E35E6</eb:ConversationId>            
        </eb:CollaborationInfo>              
        <eb:PayloadInfo>                    
          <eb:PartInfo/>                      
          <eb:PayloadInfo/>                  
        </eb:PayloadInfo>                         
      </eb:UserMessage>                        
    </eb:Messaging>                           
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">  
      <!-- Details omitted -->  
    </wsse:Security>                           
  </env:Header>                            
  <env:Body>
```
Appendix A.2 Query Response Message

The following is a simplified example of an ebMS3 Query Response message.

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <eb:Messaging xmlns:eb="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  </env:Header>
  <env:Body>
    <query:QueryRequest id="c4369c4d-740e-4b64-80f0-7b209a66d629"
    <query:ResponseOption returnType="LeafClassWithRepositoryItem"/>
    <query:Query queryDefinition="a_query">
      <query:QueryRequest id="c4369c4d-740e-4b64-80f0-7b209a66d629"
    </query:Query>
    <query:ResponseOption returnType="LeafClassWithRepositoryItem"/>
    <query:Query queryDefinition="a_query">
      <query:QueryRequest id="c4369c4d-740e-4b64-80f0-7b209a66d629"
    </query:Query>
  </env:Body>
</env:Envelope>
```
Appendix B  Acknowledgments (Non-Normative)

This specification was created in the OASIS ebCore Technical Committee, whose voting members at the
time of writing included the following individuals:

Berntzen, Mr. Sigbjorn, Directorate of Labour and Welfare Norway
Kirschner, Mr. Torsten, Directorate of Labour and Welfare Norway
Fieten, Sander, Individual
Kramer, Theo, Individual
Stojanovic, Mr. Nikola, Individual
Bergheim, Mr. Erlend Klakegg, Norwegian Digitalisation Agency
Eijk, Mr. Pim van der, Sonnenglanz Consulting
Moberg, Dr. Dale, Sonnenglanz Consulting
van Nigtevecht, Mr. Ernst Jan, Sonnenglanz Consulting
## Appendix C Revision History (Non-Normative)

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>WD01</td>
<td>2020-06-05</td>
<td>PvdE</td>
<td>Initial working draft.</td>
</tr>
<tr>
<td>WD02</td>
<td>2020-06-08</td>
<td>NS, PvdE</td>
<td>Updates from Nikola.</td>
</tr>
<tr>
<td>WD03</td>
<td>2020-06-10</td>
<td>NS</td>
<td>To be shared with ebCore TC</td>
</tr>
<tr>
<td>WD04</td>
<td>2020-07-24</td>
<td>PvdE</td>
<td>Updates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Used type attribute of Service, version information is recorded there.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Service and Action values simplified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Explained the layering principles and loose coupling.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Complete examples</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Conformance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Acknowledgments</td>
</tr>
<tr>
<td>WD05</td>
<td>2020-08-03</td>
<td>NS, PvdE</td>
<td>Updates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Improved consistency in naming of actions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Editorial.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Mention potential use of XML Encryption in packaging.</td>
</tr>
<tr>
<td>WD06</td>
<td>2020-08-06</td>
<td>PvdE</td>
<td>Updates:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Clarified that the RegRep content, in ebMS3/AS4, can go either in the SOAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Body or into a separate MIME part.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Clarified href attribute presence and value.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Corrected error in PayloadInfo in examples.</td>
</tr>
</tbody>
</table>