Specification

OpenPEPPOL AISBL

Transport Infrastructure Coordinating Community

ICT - Models

PEPPOL Business Message Envelope (SBDH)

Version: 1.2
Status: In Use
**Statement of originality**

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

**Statement of copyright**

![Creative Commons License](http://creativecommons.org/licenses/by-nc-nd/4.0/)

This deliverable is released under the terms of the Creative Commons Licence accessed through the following link: http://creativecommons.org/licenses/by-nc-nd/4.0/.

You are free to:

* **Share** — copy and redistribute the material in any medium or format.

The licensor cannot revoke these freedoms as long as you follow the license terms.
Contributors
Martin Forsberg, ESV
Markus Gudmundsson, Unimaze Software
Jostein Frømyr, Difi/Edisys Consulting
Steinar Overbeck Cook
Oriol Bausà, Invinet
Sven Rasmussen, DIGST
Stefano Monti, EPOCA/IntercentER
Philip Helger, Bundesrechenzentrum
Erlend Klakegg Bergheim, Difi
Bård Langøy, Pagero
Jerry Dimitriou, OpenPEPPOL Operating Office
Risto Collanus, Visma
Hans Berg, Tickstar

Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Change log</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0.0</td>
<td>2014-01-15</td>
<td>Initial version</td>
</tr>
<tr>
<td>1.1</td>
<td>2018-08-31</td>
<td>Added the possibility to specify document type identifier scheme and process identifier scheme</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added the possibility to specify additional attributes</td>
</tr>
<tr>
<td>1.1.1</td>
<td>2018-09-28</td>
<td>Fixed error in chapter 2.5 in the example of an additional attribute without a value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added note on attribute case sensitivity in chapter 2.5</td>
</tr>
<tr>
<td>1.2</td>
<td>2019-02-01</td>
<td>Added section for non-XML payloads</td>
</tr>
</tbody>
</table>

1 Introduction
The PEPPOL Message Envelope is a customization of the UN/CEFACT Standard Business Document Header (SBDH) [SBDH]. The customization represents a true subset of the standard XML Schemas and any instance conformant to this specification is also conformant to the SBDH.

The PEPPOL Message Envelope makes it possible for Access points to:

- Route messages without having to access to the business message/data
- Always use the same way of identifying sender/receiver, document type and process
- Overcome issues with namespace or versioning of the payload
- Provide additional attributes that help processing the payload

The Message Envelope can also carry some of the infrastructure elements when using protocols like AS2 or AS4. The creation of the Message Envelope is RECOMMENDED to be done already in the system issuing the
business document but it may also be created by a service provider who is preparing the document for 
transportation to the receiver’s Access Point. This specification does not recommend any particular setup 
with regard to this when the Message Envelope is not created in the issuing system.

1.1 Terminology

The keywords "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 [RFC2119].

1.2 Normative references

<table>
<thead>
<tr>
<th>Reference</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[RFC2119]</td>
<td>Key words for use in RFCs to Indicate Requirement Levels,</td>
</tr>
<tr>
<td></td>
<td><a href="https://www.ietf.org/rfc/rfc2119.txt">https://www.ietf.org/rfc/rfc2119.txt</a></td>
</tr>
<tr>
<td>[PEPPOL_Policy3]</td>
<td>PEPPOL Policy for use of Identifiers v3.x,</td>
</tr>
<tr>
<td>[PEPPOL_Policy4]</td>
<td>PEPPOL Policy for use of Identifiers v4.x,</td>
</tr>
</tbody>
</table>

1.3 When to use the envelope

Unless other policies are decided for the PEPPOL infrastructure, the following principals describe when the 
envelope is to be applied.

- Business Message Envelope MUST be applied for all messages exchanged with AS2
- Business Message Envelope MUST be applied for all messages exchanged with AS4

2 SBDH Usage

2.1 Party identifiers

The required Receiver party identifier in the Message Envelope header is the one that corresponds to a 
PEPPOL Participant registered in the SML/SMP. Also the Sender party identifier is required. The structure of 
the identifier MUST follow the:

- "PEPPOL Policy for use of Identifiers v3.x" [PEPPOL_Policy3] for BIS v1 and v2 documents
- "PEPPOL Policy for use of Identifiers v4.x" [PEPPOL_Policy4] for BIS v3 or later

In cases where the sender is not registered in SML/SMP the identifier of the sender MUST be used as if the 
sender would be registered.

Non-normative example:

```xml
<Sender>
  <Identifier Authority="iso6523-actorid-upis">0088:7315458756324</Identifier>
</Sender>
<Receiver>
  <Identifier Authority="iso6523-actorid-upis">0088:4562458856624</Identifier>
</Receiver>
```
2.2 XML considerations

Since the envelope and included business document becomes one single XML instance, both the envelope and the business document MUST have the same character encoding. The included business document MUST be well-formed. The Message Envelope MUST NOT contain another Message Envelope.

2.3 Non-XML Payloads

Several processes that are supported in PEPPOL require the transportation of binary data and non-XML text as payload. In order for PEPPOL to support the transmission of non-XML payloads, an XML wrapper has been defined that MUST be used for wrapping these payloads.

The XML wrapper defined in this document MUST NOT be used to wrap another XML wrapper neither as binary nor as text payload.

The following picture depicts the XML schema of the XML wrapper (see chapter 3.2 for the full XML schema):
2.3.1 Binary Payloads

In order to support the transmission of binary payloads they should be transformed and packaged as follows:

1. The binary payload must be Base64-encoded.
2. The encoded payload MUST be included inside the XML element BinaryContent. The XML namespace URI for this element MUST be http://peppol.eu/xsd/ticc/envelope/1.0.
3. The attribute mimeType MUST be set to the respective payload MIME type.
4. For text based payloads, the optional encoding attribute MUST be used if the source encoding is different than the encoding of the surrounding XML document. At least the “UTF-8” encoding MUST be supported.

Non-normative example:

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
  ...
  <BinaryContent xmlns="http://peppol.eu/xsd/ticc/envelope/1.0"
                  mimeType="application/vnd.etsi.asic-e+zip"
                  encoding="UTF-8">
    ABCD45678922 ...
  </BinaryContent>
</StandardBusinessDocument>
```

2.3.2 Non-XML Text Payloads

For text data, there is no need of a container, as it can be placed directly as payload inside a TextContent XML element. The XML namespace URI for this element must be http://peppol.eu/xsd/ticc/envelope/1.0. The attribute mimeType MUST be set to the respective payload MIME type.

Note:

- If the text payload contains XML special characters (e.g. ‘<’ or ‘>’), they MUST be escaped using XML encoding or alternatively the data needs to be wrapped inside a CDATA element so the XML remains well formed.
- If a text payload is embedded inside the TextContent element, it MUST use the same character encoding as the surrounding XML, otherwise the BinaryContent data element SHOULD be used.

Non-normative example:

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
  ...
  <TextContent xmlns="http://peppol.eu/xsd/ticc/envelope/1.0"
               mimeType="Application/EDIFACT">
    EAN008+7'UNT+4+186453437'UNZ+1+EW861380947'
  </TextContent>
</StandardBusinessDocument>
```

2.4 PEPPOL Process ID and Document Type ID

The values of Process ID and Document Type ID are necessary in the SML/SMP discovery process to retrieve the relevant service metadata. Both values should be mapped to the element located at:
The respective identifier schemes are to be located in the following element (new in v1.1):

For backwards compatibility reasons (from version 1.1 to 1.0) – if the identifier schemes are missing – the default process scheme identifier cenbii-procid-ubl and the default document type identifier scheme busdox-docid-qns MUST be used.

The qualifier located at

is used to distinguish the meaning of the values by using codes: DOCUMENTID (for a document type identifier) and PROCESSID (for process identifiers).

Non-normative example without identifier schemes:

Non-normative example including identifier schemes (possible since v1.1 of this specification):

2.5 Additional attributes

Additional attributes MAY be provided that can be used to support the processing of the payload. These additional attributes are represented as key-value-pairs.

Each additional attribute is represented as a

element.

The attribute key must be contained in the child element Type. All attribute keys listed in chapter 2.5.1 are reserved and cannot be used as an additional attribute key. The attribute key MUST be unique within an SBDH. The attribute key MUST be handled case sensitive.

The attribute value must be contained in the child element InstanceIdentifier. The attribute value MAY be omitted.

Non-normative example with two additional attributes:
Non-normative example with one additional attribute that has no value:

```xml
<BusinessScope>
  <!-- Document type and process ID -->
  <!-- Scope -->
  <Type>IndicatorAttribute</Type>
  <InstanceIdentifier/>
</Scope>
</BusinessScope>
```

### 2.5.1 Reserved attributes

The following additional attribute keys are reserved for internal use in the PEPPOL network and MUST NOT be used for other purposes than the intended ones.

<table>
<thead>
<tr>
<th>Attribute key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOCUMENTID</td>
<td>Specifies the PEPPOL Document Type Identifier value (see chapter 2.3)</td>
</tr>
<tr>
<td>PROCESSID</td>
<td>Specifies the PEPPOL Process Identifier value (see chapter 2.3)</td>
</tr>
<tr>
<td>TECHNICAL_VALIDATION_URL</td>
<td>Reserved for potential future use.</td>
</tr>
<tr>
<td>TECHNICAL_VALIDATION_REQUIRED</td>
<td>Reserved for potential future use.</td>
</tr>
</tbody>
</table>

### 2.6 Message Envelope Schema

...
<table>
<thead>
<tr>
<th>Element/Attribute</th>
<th>Annotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>StandardBusinessDocument</td>
<td>Type StandardBusinessDocument</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>StandardBusinessDocumentHeader</td>
<td>Type StandardBusinessDocumentHeader</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>HeaderVersion</td>
<td>Type xs:string</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>Sender</td>
<td>Type Partner</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>Receiver</td>
<td>Type Partner</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>Authority</td>
<td>Type xs:string</td>
</tr>
<tr>
<td></td>
<td>Use required</td>
</tr>
<tr>
<td>DocumentIdentification</td>
<td>Type DocumentIdentification</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>Standard</td>
<td>Type xs:string</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>TypeVersion</td>
<td>Type xs:string</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>InstanceIdentifier</td>
<td>Type xs:string</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
<tr>
<td>Type</td>
<td>Type xs:string</td>
</tr>
<tr>
<td></td>
<td>Occurrence 1 .. 1</td>
</tr>
</tbody>
</table>

**Element/Attribute Description**

- **HeaderVersion**: Always value 1.0
- **Sender**: Use the format XXXX:AAAAAAAA where XXXX is the type of identifier (such as 0088 for GS1 GLN) and AAAAAAAA the actual identifier.
- **Authority**: Use fixed value "iso6523-actorid-upis"
- **Receiver**: Use the format XXXX:AAAAAAAA where XXXX is the type of identifier (such as 0088 for GS1 GLN) and AAAAAAAA the actual identifier.
- **DocumentIdentification**: Use fixed value "iso6523-actorid-upis"
- **Standard**: The standard of the enveloped business message, normally described by use of the XML namespace of the business message root element (such as urn:oasis:names:specification:ubl:schema:xsd:Order-2)
- **TypeVersion**: The version number of the enveloped business message (such as the value "2.1" for OASIS UBL 2.1 or "2.0" for OASIS UBL 2.0)
- **InstanceIdentifier**: An informative unique ID created by the issuer of the envelope. The InstanceIdentifier MUST be unique for each Business Message Envelope being created. This ID is not the same as the ID of the business message (such as the Invoice Number). It is not the same as a transmission Message ID generated by the application sending the message (as defined in AS2 or AS4). The InstanceIdentifier MUST be globally unique and it is RECOMMENDED to use UUID (such as 118e3040-51d2-11e3-8f96-0800200c9a66)
- **Type**: Message type - mandatory in SBDH. XML local element name of the root-element in the business message.
## Element/Attribute

<table>
<thead>
<tr>
<th>Element/Attribute</th>
<th>Occurrence</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreationDateAndTime</td>
<td>1 .. 1</td>
<td>xs:dateTime</td>
<td>The date and time for when this envelope was created. It is NOT necessarily the same as the issue date of the business document (such as the invoice) being enveloped. It is NOT necessarily the date time for transmission. The format of the value of this MUST include timezone information. Use this format for UTC: 2014-01-17T09:30:00Z (Where the “Z” indicates UTC) Or specify offset from UTC by adding the time difference: 2014-01-17T09:30:00+02:00 (Where +02:00 indicates 2 hours positive offset to UTC)</td>
</tr>
<tr>
<td>BusinessScope</td>
<td>1 .. 1</td>
<td>BusinessScope</td>
<td>Elements used to identify the ProcessID and DocumentID. Repeat at least twice - once for DocumentID once for ProcessID</td>
</tr>
<tr>
<td>Scope</td>
<td>1 .. 1</td>
<td>Scope</td>
<td>Repeat at least twice - once for DocumentID once for ProcessID</td>
</tr>
<tr>
<td>ScopeAttributes</td>
<td>1 .. 1</td>
<td>xs:string</td>
<td>Qualifier of how to understand the InstanceIdentifier element. Codes. Applicable Codes DOCUMENTID PROCESSID</td>
</tr>
<tr>
<td>InstanceIdentifier</td>
<td>1 .. 1</td>
<td>xs:string</td>
<td>The ProcessID (profile ID) or DocumentID corresponding to PEPPOL SMP for which the enveloped payload is intended to be used for. (the type of value is qualified by the ScopeAttributes/Type element) For senders - this value can be used to retrieve the correct set of PEPPOL service metadata. For receivers - this value can be used to verify that the receiving PEPPOL Participant has published support for this DocumentID or ProcessID.</td>
</tr>
<tr>
<td>Identifier</td>
<td>0 .. 1</td>
<td>xs:string</td>
<td>Identification scheme used for the Document type identifier/Process identifier. Use value &quot;busdox-docid-qns&quot; for Document type identifier Identification scheme for Process identifier may differ depending on the Document type.</td>
</tr>
<tr>
<td>xs:any</td>
<td>1 .. 1</td>
<td></td>
<td>Business message goes here!</td>
</tr>
</tbody>
</table>

## Appendix

3 Appendix
### 3.1 Example instance document (non-normative)

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <StandardBusinessDocumentHeader>
    <HeaderVersion>1.0</HeaderVersion>
    <Sender>
      <Identifier Authority="iso6523-actorid-upis">0088:7315458756324</Identifier>
    </Sender>
    <Receiver>
      <Identifier Authority="iso6523-actorid-upis">0088:4562458856624</Identifier>
    </Receiver>
    <DocumentIdentification>
      <TypeVersion>2.1</TypeVersion>
      <InstanceIdentifier>123123</InstanceIdentifier>
      <Type>Invoice</Type>
      <CreationDateAndTime>2019-02-01T15:42:10Z</CreationDateAndTime>
    </DocumentIdentification>
    <BusinessScope>
      <Scope Type="DOCUMENTID">
        <Identifier>budox-docid-qns</Identifier>
      </Scope>
      <Scope Type="PROCESSID">
        <InstanceIdentifier>urn:fdc:peppol.eu:2017:poacc:billing:01:1.0</InstanceIdentifier>
        <Identifier>cenbii-procid-ubl</Identifier>
      </Scope>
    </BusinessScope>
  </StandardBusinessDocumentHeader>
<!-- reduced instance file -->
</Invoice>
</StandardBusinessDocument>
```

### 3.2 Message Envelope Extension XML Schema

The normative version of the Message Envelope Extension XML Schema can be found at [https://github.com/OpenPEPPOL/documentation/tree/master/TransportInfrastructure](https://github.com/OpenPEPPOL/documentation/tree/master/TransportInfrastructure)