

Business Interoperability Specification



OpenPEPPOL AISBL



**Post Award Coordinating
Community**

ICT - Models

**BIS 36A – Message
Level Response**



**Version: 2.00_A
Status: In use**



Statement of copyright

This PEPPOL Business Interoperability Specification (BIS) document is based on the CEN CWA prepared by the BII workshop specified in the Introduction below.
The original CEN CWA document contains the following copyright notice which still applies:

© 2012 CEN All rights of exploitation in any form and by any means reserved worldwide for CEN national Members.

The CEN CWA documents and profiles prepared by the BII workshop are not specific to a business area. Subject to agreement with CEN, customizations have been made by PEPPOL to establish the PEPPOL BIS, detailing and adding further guidance on the use of BII profiles.

OpenPEPPOL AISBL holds the copyright in the customizations made to the original document. The customizations appear from the corresponding conformance statement which is attached to this document. For the purpose of national implementations, customizations covered by the conformance statement may be further refined and detailed by PEPPOL Authorities and/or other entities authorized by OpenPEPPOL AISBL, provided that interoperability with PEPPOL BIS is ensured.

This PEPPOL BIS document may not be modified, re-distributed, sold or repackaged in any other way without the prior consent of CEN and/or OpenPEPPOL AISBL.

Table of Contents

1	INTRODUCTION TO OPENPEPPOL AND BIS	5
1.1	AUDIENCE.....	5
2	REFERENCES	6
3	DOCUMENT HISTORY	7
3.1	REVISION HISTORY	7
3.2	CONTRIBUTORS.....	7
4	CHANGES IN THIS VERSION	8
5	PRINCIPLES AND PREREQUISITES.....	9
5.1	MESSAGE LEVEL RESPONSE MESSAGE IN GENERAL	9
5.2	MESSAGE LEVEL RESPONSE – SCOPE.....	10
5.3	PARTIES AND ROLES	11
5.4	PROCESS REQUIREMENTS.....	12
6	BUSINESS REQUIREMENTS	13
7	CODE LISTS	14
7.1	CODE LISTS FOR CODED ELEMENTS	14
7.2	CODE LIST FOR IDENTIFIER SCHEMES	14
8	BUSINESS RULES.....	15
8.1	BII RULES	15
8.2	PEPPOL SPECIFIC RULES	15
8.3	CODE LISTS BUSINESS RULES	15
9	PROCESS AND TYPICAL SCENARIOS	16
9.1	LEGEND FOR BPMN DIAGRAMS.....	16
9.2	ACTORS AND ROLES	16
9.3	PROCESS IN GENERAL	16
9.4	TYPICAL USE CASES	17
9.4.1	<i>Use Case 1 – Response with acceptance.....</i>	<i>17</i>
9.4.2	<i>Use Case 2 – Response with acknowledgment</i>	<i>18</i>
9.4.3	<i>Use Case 3 – Response with reject – violation of business rules.....</i>	<i>18</i>
10	DESCRIPTION ON IDENTIFIER USAGE	20
10.1	PARTY IDENTIFICATION	20
10.2	DOCUMENT TYPE IDENTIFIERS/CUSTOMIZATION ID.....	20
10.3	PROCESS/PROFILE IDENTIFICATION	20
11	DESCRIPTION OF SELECTED PARTS OF THE ENVELOPE	21
11.1	PARTY IDENTIFICATION	21
11.1.1	<i>Sender Party</i>	<i>21</i>
11.1.2	<i>Receiver Party.....</i>	<i>21</i>
11.2	DOCUMENT TYPE IDENTIFIERS/CUSTOMIZATION ID.....	21
11.2.1	<i>Document identification</i>	<i>21</i>
11.2.2	<i>Document type identification</i>	<i>21</i>
11.3	PROCESS/PROFILE IDENTIFICATION	22
11.4	OVERVIEW OF ENVELOPE MAPPING.....	22
12	DESCRIPTION OF SELECTED PARTS OF THE MLR MESSAGE	23
12.1	NAMESPACE URIS.....	23
12.2	UBL VERSION ID	23

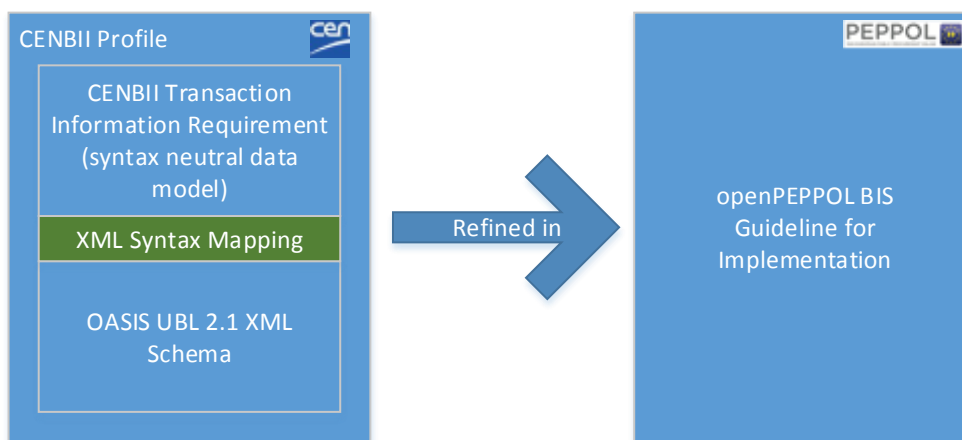
12.3	CUSTOMIZATION ID	23
12.4	PROFILE ID	23
12.5	PARTY IDENTIFICATION	23
12.5.1	<i>Sender Party</i>	23
12.5.2	<i>Receiver Party</i>	23
12.6	DOCUMENT RESPONSE	24
12.7	DOCUMENT REFERENCE	24
12.8	LINE RESPONSE	24
12.9	OVERVIEW OF MLR MESSAGE MAPPING	25
13	XML SCHEMA GUIDELINE AND INFORMATION CONTENT	26
13.1	XML SCHEMA GUIDELINE	26
13.2	INFORMATION CONTENT	27
14	APPENDICES	31
14.1	APPENDIX A – XML FOR USE CASES	31
14.1.1	<i>Use case 1 – Response with acceptance</i>	31
14.1.2	<i>Use case 2 – Response with acknowledgment</i>	31
14.1.3	<i>Use case 3 – Response with reject – violation of business rules</i>	31
14.2	APPENDIX B – CONFORMANCE STATEMENT	31

1 Introduction to OpenPEPPOL and BIS

This BIS is a result of work within PEPPOL project and is published as part of PEPPOL specifications.

This PEPPOL BIS provides a set of specifications for implementing a PEPPOL business process. The document is concerned with clarifying requirements for ensuring interoperability of pan-European Public eProcurement and provides guidelines for supporting these requirements and how to implement them. This PEPPOL BIS is based on the CEN WS/BII2 Profile “Profile BII36 Message Level Response”.

The purpose of this document is to describe a common format for the message level response message in the European market, and to facilitate an efficient implementation and increased use of electronic collaboration regarding the message level response process based on this format.



1.1 Audience

The audience for this document is organizations wishing to be PEPPOL enabled for exchanging electronic business documents, and/or their ICT-suppliers. These organizations may be:

- ▶ Service providers
- ▶ Contracting Authorities
- ▶ Economic Operators
- ▶ Software Developers

More specifically it is addressed towards the following roles:

- ▶ ICT Architects
- ▶ ICT Developers
- ▶ Business Experts

For further information on PEPPOL/OpenPEPPOL please see [COMMON_BIS].

2 References

[PEPPOL]

<http://www.peppol.eu/>

[PEPPOL_PostAward]

<http://www.peppol.eu/ressource-library/technical-specifications/post-award/mandatory>

[PEPPOL_Transp]

<http://www.peppol.eu/ressource-library/technical-specifications/infrastructure-resources>

[PEPPOL_Ident]

https://joinup.ec.europa.eu/svn/peppol/TransportInfrastructure/PEPPOL_Policy%20for%20use%20of%20identifiers-300.pdf

[PEPPOL_Env]

https://joinup.ec.europa.eu/svn/peppol/TransportInfrastructure/ICT-Transport-OpenPEPPOL-Envelope_Specification-100_2014-01-15.pdf

[CEN_BII2]

<http://www.cenbii.eu>

[COMMONS_BIS]

<https://joinup.ec.europa.eu/svn/peppol/Validation%20Resources/BIS%20v2/PEPPOL%20BIS%20Common%20text%20and%20introduction%20-%20over%201%202014-04-14.pdf>

[BII_MessageLevelResp]

ftp://ftp.cen.eu/public/CWAs/BII2/CWA16558/CWA16558-Annex-M-BII-Profile-36-MessageLevelResponse-V1_0_0.pdf

ftp://ftp.cen.eu/public/CWAs/BII2/CWA16558/CWA16558-Annex-G-BII-CodeLists-V2_0_4.pdf

A browsable HTML version: http://spec.cenbii.eu/BII2/fxhtml/Trdm071-MessageLevelResponse/g_1.htm?http://spec.cenbii.eu/BII2/fxhtml/Trdm071-MessageLevelResponse/g_5.htm

[UBL]

<http://docs.oasis-open.org/ubl/UBL-2.1.html>

[UBL_ApplicationResp]

<http://docs.oasis-open.org/ubl/os-UBL-2.1/xsd/maindoc/UBL-ApplicationResponse-2.1.xsd>

[Schematron]

<http://www.schematron.com>

[XSLT]

<http://www.w3.org/TR/xslt20/>

[EIF]

European Interoperability Framework 2.0, found at:

http://ec.europa.eu/isa/library/index_en.htm

http://ec.europa.eu/isa/documents/isa_annex_ii_eif_en.pdf

[UN/EDIFACT 4343]

<http://www.unece.org/trade/untidd/d16a/tred/tred4343.htm>

3 Document history

3.1 Revision history

Version	Date	Author	Organisation	Description
1.0	2013-11-11	Project team	OpenPEPPOL	Public review
2.0	2017-03-30	Bård Langøy Martin Forsberg Philip Helger	Pagero ESV BRZ	- Deleted all usage policy-related requirements and rules - Added possibility to just acknowledge a received message (and not inform about conformance test result)

3.2 Contributors

Olav Astad Kristiansen, DIFI, NO
Jostein Frømyr, DIFI, NO
Per Martin Jøraholmen, DFØ
Martin Forsberg, SFTI, SE
Mette Theisen, EVRY, NO
Philip Helger, BRZ, AT
Oriol Bausa, Invinet, ES
Randi Haftorn, Jernbaneverket, NO
Christian Druschke, Cap Gemini, NO
Imran Muhammad Shafiq, ISS, NO
Erik Gustavsen, DIFI, NO
Bård Langøy, Pagero, SE

4 Changes in this version

This chapter describes the major changes between this version and the previous version of the BIS. Semantical changes:

- Changed “Issue type code” to “Status reason code”

BII Rules Changes:

- Removed BII2-T71-R010 (replaced by rule EUGEN-T71-R006)
- Removed BII2-T71-R012 (replaced by rule EUGEN-T71-R008)

PEPPOL Specific Rules Changes:

- Removed EUGEN-T71-R001 (business document reference is irrelevant because reference must now be taken from the envelope)
- Added EUGEN-T71-R006 (replacing rule BII2-T71-R010)
- Added EUGEN-T71-R007 (extension)
- Added EUGEN-T71-R008 (replacing rule BII2-T71-R012)

Code List Business Rules changes:

- Removed CL-071-R001 (business document reference is irrelevant because reference must now be taken from the envelope)
- Changed text of OP-T71-R003 (clarification)

Changes to the XML representation:

- Removed /Note
- Made /SenderParty/EndpointID mandatory
- Removed /SenderParty/PartyIdentification
- Removed /SenderParty/PartyName
- Made /ReceiverParty/EndpointID mandatory
- Removed /ReceiverParty/PartyIdentification
- Removed /ReceiverParty/PartyName
- Removed /DocumentResponse/DocumentReference/DocumentTypeCode
- Removed /DocumentResponse/DocumentReference/VersionID
- Added /DocumentResponse/LineResponse/LineReference/DocumentReference/ID
- Removed /DocumentResponse/LineResponse/Response/ResponseCode
- Added /DocumentResponse/LineResponse/Response/ReferenceID

5 Principles and prerequisites

This chapter describes the principles and assumptions that underlie the use of the PEPPOL Message Level Response message. It is based on the CEN BII2 Message Level Response.

A Message Level Response message can be used in the choreography of the exchange of a business document to improve reliability by allowing a receiver of a business document to inform the sender about the results of receivers validations and, in case of negative results, to inform the sender about the nature of the errors as well as their details. They may allow the sender of the document to take appropriate action.

A Message Level Response message can be used with any document type suitable for usage in the PEPPOL transport infrastructure.

5.1 Message Level Response message in general

Through the start to end flow of a message exchange; from the creation of an electronic message, down the transport line that goes through one or more transport networks to the designated receiver and all way through the eventual processing of the message content, there may be need to give responses to the relevant parties up-line about the status or results of the actions that the message goes through. These responses are of different nature but for the purpose of this document they can be divided into the following main groups.

Transport acknowledgements

These are messages that are exchanged within the transport network(s) to inform about the process of carrying a message down the transport line. These responses may inform someone up-line that the delivery to a given point was successful or not and may contain details about issues that are relevant such as why a delivery was not successful. The key nature of these responses is that they do not in any way act on result of validation or processing of the content of the payload that is being transported. These response messages are commonly called “acks”.

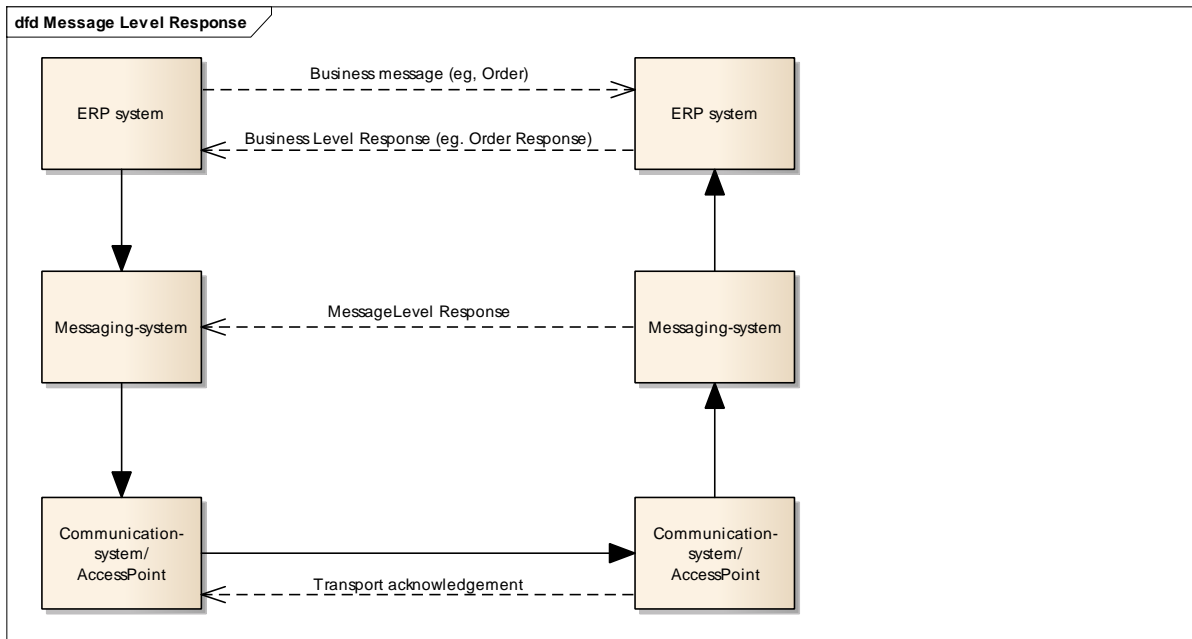
Message Level Responses

When a message has reached a given point in the transport line its content may be validated according to agreed specifications that may be both syntactical and semantic. The outcome of these validations may be reported to a relevant party up-line, informing him whether the validation was successful or not as well as giving some details. An example could be that an order message that is received is rejected because it is missing a closing tag (syntax error) or because its amounts don't add up according to what is specified in the relevant syntax specification. A key nature of these messages is that they report on the message content on the basis of the technical specifications that apply.

Business Level Responses

A message that has been received and accepted for processing may call for an action on the receiver's behalf. That receiver's action may need to be reported back up-line to a relevant party. An example is that a technically correct order may be received but the receiver decides to reject the order for any business reason such as out-of-stock situation, expired contract etc. The key nature of these responses is that they report a business decision that is made on the message instance received.

This specification is only concerned with the Message Level Responses.



5.2 Message Level Response – Scope

The Message Level Response message is intended to inform the issuer of the following situations:

1. The received message contained errors according to the relevant conformance rules.
 - The message will not be processed any further.
2. The received message passed the validation of conformance rules without any fatal errors.
 - The message will be processed further.
3. The received message is not validated for conformance but the receiver acknowledges that it has been received and identified as a business message.
 - The message will be processed further.

The following errors are **within the scope** for a negative/rejecting Message Level Response:

- XML schema validation error
- Standard Compliance violations (e.g. empty elements not being allowed by UBL 2.1)
- Validation error of type fatal error
- Validation error of type warning. Warnings alone must NOT cause rejection of the business document (but they may be reported in addition to fatal errors)
- Wrong version of business document (Will be handled like validation error of type fatal error)

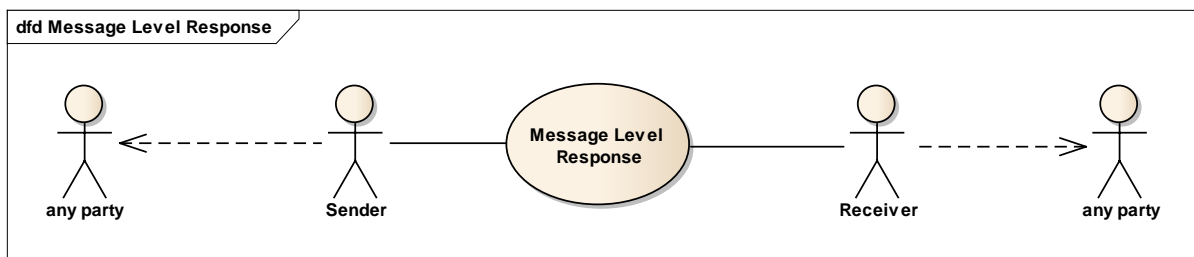
The following errors are **outside the scope** of the message level response:

- Unknown sender (in scope of transport acknowledgement)
- Unknown receiver (in scope of transport acknowledgement)
- Wrong version of envelope (in scope of transport acknowledgement)
- XML schema validation error – envelope (in scope of transport acknowledgement)
- XML not well formed (in scope of transport acknowledgement)
- Non supported encoding (in scope of transport acknowledgement)
- Wrong value (after database look-up) in reference fields (in scope of Business Level Response). Examples:
 - Wrong order number in invoice
 - Wrong project or customer reference in invoice
 - Wrong contract ID reference in catalogue

5.3 Parties and roles

The table below gives the definitions of the parties and roles of the message level response process. The sender and receiver of the Message Level Response message should be extracted from the envelope, i.e. the PEPPOL participants.

Party / Role	Definition
Sender	The party sending an electronic message level response message back to the sending party of the business document.
Receiver	The party an electronic message level response was addressed to and who is supposed to process the message level response. This is the same party as the sender of the business document.



5.4 Process requirements

A message level response document should support the following requirements:

- ▶ The message and its use should not be linked to any specific infrastructure implementation.
- ▶ At most one message level response can be sent per received message.
- ▶ The response message should convey either an “accept”, “reject” or “acknowledge” of the instance received. If accepted or acknowledged, no fatal errors should be reported. If rejected, the reason **MUST** be stated.
- ▶ A rejection implies that the received message will not be further processed by the receiver of the business document.
- ▶ An acceptance or acknowledge implies that the received message will be further processed
- ▶ The specification assumes that any service provider acts on behalf of either the sender or the receiver.
- ▶ The response message should provide for coded responses in order to facilitate automation in processing the message response.
- ▶ MLRs may be sent to different endpoints, for a single participant, depending on the used process.
- ▶ MLRs **MUST NOT** be sent as a response to receiving an MLR.

6 Business requirements

ID	Requirement
tbr71-001	It must be possible to give the response message a unique identifier. The identifier is issued by the sender of the MLR and can be used to uniquely identify a message instance. ¹
tbr71-002	It must be possible to state the date and time when the response message is issued. The date must always be given but the time (hours, minutes and seconds) is optional to use.
tbr71-003	It must be possible to state a free text note used to inform the receiver about information that is not explicitly given in any dedicated structure. The information is meant to be manually read/assessed by the receiver.
tbr71-004	It must be possible to specify the Party sending the response.
tbr71-005	It must be possible to specify the Party receiving the response.
OP-tbr71-006	It must be possible to specify the Response to a previously received message referring to the document by referring to its envelope/messageID
OP-tbr71-007	It must be possible to give the response as a code (response code) for the whole document (on header-level). A response code list is required in order to facilitate automated processing of message responses. Examples of response codes may be Accepted/Rejected/Acknowledged.
OP-tbr71-009	It must be possible to provide a list of validation errors and which type of violation they represent.
tbr71-010	A response document must be able to clearly indicate whether the received document was accepted or not.
tbr71-011	It must be possible to sign the response document in order to provide for non-repudiation.
OP-tbr71-014	The message should allow for location identifiers/paths to indicate the location of the errors in the received instance.

¹ For the MLR UBL-Binding this refers to an identifier the in the ApplicationResponse document.

7 Code lists

7.1 Code lists for coded elements

Table of the code lists used in the message level response transaction:

Business Term	Source	Subset	XPath	listID
Response code	[UN/EDIFACT 4343]	PEPPOL	ApplicationResponse/cac:DocumentResponse/cac:Response/cbc:ResponseCode	UNCL4343, Values: RE - Rejected AP - Accepted AB – Message acknowledgement
Status reason code	[BII_Message LevelResp]		ApplicationResponse/cac:DocumentResponse/cac:LineResponse/cac:Response/cac:Status/cbc:StatusReasonCode	StatusReason, Values: SV – Syntax violation RVF – Rule violation, fatal RVW – Rule violation, warning

7.2 Code list for identifier schemes

Table of the code lists used to constrain the values of `schemeID` for identifiers in the message level response transaction:

Business Term	Allowed SchemeID	Applicable XPath	Note
Party Identifier	See [PEPPOL_Ident]	cbc:EndpointID/@schemeID cac:PartyIdentification/cbc:ID/@schemeID	

8 Business rules

8.1 BII rules

The list below describes the business rules valid for the Message Level Response message. The MLR message itself will not be validated to prevent possible infinite loops.

RuleID	Rule	Target	Errorlevel
BII2-T71-R001	A message level response MUST have a profile identifier	message level response	Fatal
BII2-T71-R002	A message level response MUST have a customization identifier	message level response	Fatal
BII2-T71-R003	A message level response MUST contain the date of issue	message level response	Fatal
BII2-T71-R004	A message level response MUST contain the response identifier	message level response	Fatal
BII2-T71-R005	The party sending the message level response MUST be specified	message level response	Fatal
BII2-T71-R006	The party receiving the message level response MUST be specified	message level response	Fatal

8.2 PEPPOL Specific rules

RuleID	Rule	Target	Errorlevel
EUGEN-T71-R002	A response code MUST have a 'listID' attribute value of 'UNCL4343'.	message level response	Fatal
EUGEN-T71-R003	A status reason code MUST have a 'listID' attribute value of 'PEPPOLSTATUS'	message level response	Fatal
EUGEN-T71-R004	An endpoint identifier MUST have a 'schemeID' attribute.	message level response	Fatal
EUGEN-T71-R005	A party identifier MUST have a 'schemeID' attribute.	message level response	Fatal
EUGEN-T71-R006	A message level response MUST contain a document reference pointing towards the envelope of the business document that the response relates to.	message level response	Fatal
EUGEN-T71-R007	A response document MUST clearly indicate the reason for rejection.	message level response	Fatal
EUGEN-T71-R008	A response document MUST indicate whether the received document was accepted or not.	message level response	Fatal

8.3 Code Lists Business Rules

RuleID	Rule
CL-071-R002	A response code MUST be from the UNCL 4343 PEPPOL Subset code list
OP-T71-R001	An Endpoint Identifier Scheme MUST be from the list of PEPPOL Party Identifiers described in the "PEPPOL Policy for using Identifiers".
OP-T71-R002	A Party Identifier Scheme MUST be from the list of PEPPOL Party Identifiers described in the "PEPPOL Policy for using Identifiers".
OP-071-R003	A status reason code MUST be from PEPPOL Status code list (see chapter 7.1)

9 Process and typical scenarios

9.1 Legend for BPMN diagrams

The diagrams are expressed in the BPMN notation. The diagram below serves as an explanation for the diagrams used in the process descriptions.

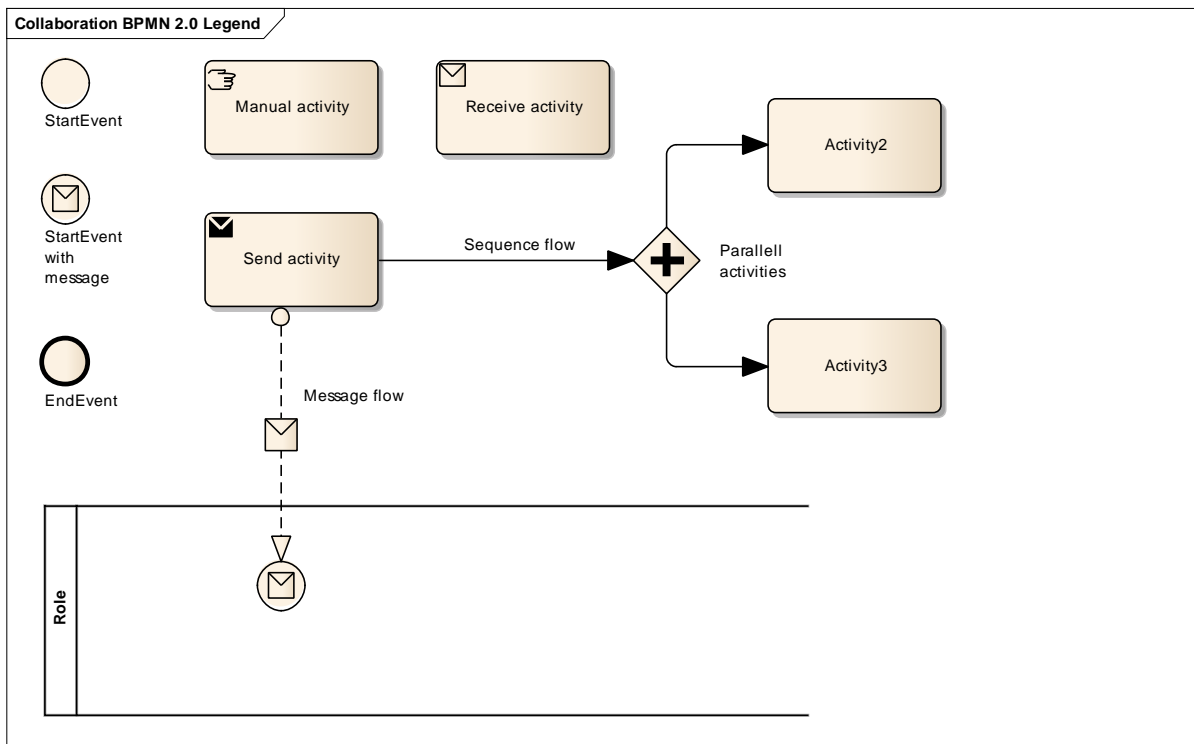


Figure 1: BPMN diagram elements

The following section and diagrams show the choreography of the business process involving various parties.

9.2 Actors and roles

In the use case descriptions below the following terms are used:

- *BusinessDocumentSender*: Sender party in the role of sending a business document
- *BusinessDocumentReceiver*: Receiving party in the role of receiving a business document
- *MLRSender*: Sender party in the role of sending a MLR document
- *MLRReceiver*: Receiver party in the role of receiving a MLR document

In the following use cases the *BusinessDocumentSender* is the same physical party as the *MLRReceiver* and the *BusinessDocumentReceiver* is the same physical party as the *MLRSender*.

9.3 Process in general

The process starts when a *BusinessDocumentSender* is preparing an electronic business document and then sends it. The *BusinessDocumentReceiver* receives the business document and potentially validates syntax and business rules.

If the *BusinessDocumentSender* has requested an MLR back², the *BusinessDocumentReceiver* either:

1. Validates the business document and based on the result returns either an accept (no fatal errors) or a reject (fatal errors found)
2. Doesn't validate the business document and just sends a MLR with a response code indicating the message is acknowledged.

If the *BusinessDocumentSender* hasn't requested an MLR back, the *BusinessDocumentReceiver* either:

1. Validates the business document and if fatal errors are found, returns a reject.
2. Doesn't respond with a MLR in case no fatal errors are found.

If a MLR is returned the *MLRReceiver* must be notified and take appropriate action. If the response is positive the *MLRReceiver* may update the status of the business document or simply ignore the MLR.

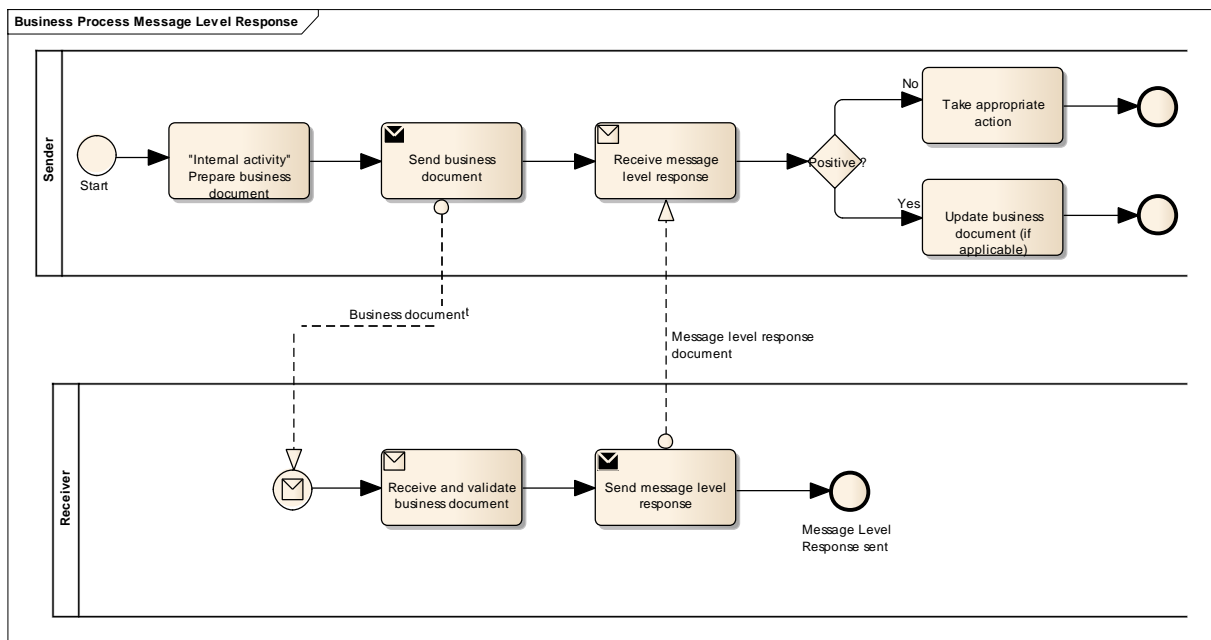


Figure 2: Process in general

9.4 Typical Use Cases

9.4.1 Use Case 1 – Response with acceptance

Use Case number	1
Use Case Name	Response with acceptance
Use Case Description	This use case describes a scenario where a business document has been received, validated and no errors are found. The BusinessDocumentSender has indicated that a MLR is requested and a response indicating an acceptance is sent back.
Parties involved	BusinessDocumentSender, MLRReceiver BusinessDocumentReceiver, MLRSender
Assumptions	<ol style="list-style-type: none"> 1. The BusinessDocumentReceiver has received an electronic business document from the BusinessDocumentSender. The BusinessDocumentSender has indicated that a response is requested. 2. The BusinessDocumentReceiver has validated the business document from the BusinessDocumentSender. 3. The result of the validation is OK, no fatal errors.
The flow	1. The BusinessDocumentSender has prepared and sent an electronic business document to

² This specification does not prescribe how a request for MLR is made or communicated.

	<p>the BusinessDocumentReceiver.</p> <ol style="list-style-type: none"> The BusinessDocumentReceiver has received the business document. The BusinessDocumentReceiver has validated the business document. The MLRSender has sent a message level response message back to the BusinessDocumentSender with response code indicated that business document was accepted (it follows the conformance rules) The MLRReceiver has received and processed the message level response message.
Result	The message level response message helped the BusinessDocumentSender to confirm that the business document was received and validated with no errors by the BusinessDocumentReceiver.
XML example file	See Appendix A for a sample file illustrating Use Case 1.

9.4.2 Use Case 2 – Response with acknowledgment

Use Case number	2
Use Case Name	Response with acknowledgment
Use Case Description	This use case describes a scenario where a business document has been received, but no validation is carried out. The BusinessDocumentSender has indicated that a MLR is requested and a response indicating an acknowledgment of receipt is sent back
Parties involved	BusinessDocumentSender, MLRReceiver BusinessDocumentReceiver, MLRSender
Assumptions	<ol style="list-style-type: none"> The BusinessDocumentReceiver has received an electronic business document from the BusinessDocumentSender. The BusinessDocumentSender has indicated that a response is requested. The BusinessDocumentReceiver don't validate incoming messages.
The flow	<ol style="list-style-type: none"> The BusinessDocumentSender has prepared and sent an electronic business document to the BusinessDocumentReceiver. The BusinessDocumentReceiver has received the business document. The MLRSender has sent a message level response message back to the BusinessDocumentSender with response code indicated that business document was acknowledged. The MLRReceiver has received and processed the message level response message.
Result	The message level response message helped the BusinessDocumentSender to confirm that the business document was received by the BusinessDocumentReceiver.
XML example file	See Appendix A for a sample file illustrating Use Case 2.

9.4.3 Use Case 3 – Response with reject – violation of business rules

Use Case number	3
Use Case Name	Response with reject– violation of validation rules
Use Case Description	This use case describes a scenario where a business document has been received and validated. The business document contained fatal errors and a MLR is returned to the MLRReceiver indicating the business document is rejected.
Parties involved	BusinessDocumentSender, MLRReceiver BusinessDocumentReceiver, MLRSender
Assumptions	<ol style="list-style-type: none"> The BusinessDocumentReceiver has received an electronic business document from the BusinessDocumentSender. The BusinessDocumentSender did not request an MLR. The BusinessDocumentReceiver has validated the business document from the BusinessDocumentSender. The result of the validation is not OK due to violation of validation rules.

The flow	<ol style="list-style-type: none"> 1. The BusinessDocumentSender has prepared and sent an electronic business document to the BusinessDocumentReceiver. 2. The BusinessDocumentReceiver has received the business document. 3. The BusinessDocumentReceiver has validated and rejected the business document. 4. The MLRSender has sent a message level response message back to the BusinessDocumentSender. 5. The MLRReceiver has received and processed the message level response message and performed appropriate action due to the rejection.
Result	<p>The message level response message helped the BusinessDocumentSender to confirm that the business document was received and rejected by the BusinessDocumentReceiver. The BusinessDocumentSender must take appropriate action to correct and resend the business document.</p>
XML example file	<p>See Appendix A for a sample file illustrating Use Case 3.</p>

10 Description on identifier usage

This chapter describes the general usage of identifiers in an MLR document. The rules stated in this chapter apply both to the envelope (e.g. SBDH) as well as to the payload document (e.g. UBL). For all identifiers the Policies 1, 2, 3 and 4 of [PEPPOL_Ident] apply.

10.1 Party identification

Party identifiers need to be handled according to Policies 5, 6, 7 and 8 of [PEPPOL_Ident].

10.2 Document type identifiers/Customization ID

Document type and Customization identifiers need to be handled according to Policies 10, 11, 12 and 13 of [PEPPOL_Ident]

10.3 Process/Profile identification

Process/Profile identifiers need to be handled according to Policies 15 and 16 of [PEPPOL_Ident].

11 Description of selected parts of the envelope

The MLR envelope MUST adhere to the [PEPPOL_Env] specification.

11.1 Party identification

11.1.1 Sender Party

The element `StandardBusinessDocumentHeader/Sender/Identifier` MUST contain the party identification of the receiver of the original Business Document.

Example:

```
<Sender>
  <Identifier Authority="iso6523-actorid-upis">9908:981915550</Identifier>
</Sender>
```

11.1.2 Receiver Party

The element `StandardBusinessDocumentHeader/Receiver/Identifier` MUST contain the party identification of the sender of the original Business Document.

Example:

```
<Receiver>
  <Identifier
    Authority="iso6523-actorid-upis">0088:7315458756324</Identifier>
</Receiver>
```

11.2 Document type identifiers/Customization ID

11.2.1 Document identification

The element `StandardBusinessDocumentHeader/DocumentIdentification/Standard` MUST have the value

```
urn:oasis:names:specification:ubl:schema:xsd:ApplicationResponse-2
(corresponds to the root namespace URI of the UBL document)
```

The element `StandardBusinessDocumentHeader/DocumentIdentification/TypeVersion` MUST have the value 2.1 (corresponds to the used UBL version).

The element `StandardBusinessDocumentHeader/DocumentIdentification/Type` MUST have the value `ApplicationResponse` (corresponds to the UBL document type).

Example:

```
<DocumentIdentification>
  <Standard>
    urn:oasis:names:specification:ubl:schema:xsd:ApplicationResponse-2
  </Standard>
  <TypeVersion>2.1</TypeVersion>
  <InstanceIdentifier>...</InstanceIdentifier>
  <Type>ApplicationResponse</Type>
  <CreationDateAndTime>...</CreationDateAndTime>
</DocumentIdentification>
```

11.2.2 Document type identification

The element

```
StandardBusinessDocumentHeader/BusinessScope/Scope[Type/text()='DOCUMENTID']/InstanceIdentifier
```

 MUST contain the document type identifier of the MLR.

The document type identifier to be used is:

```
urn:oasis:names:specification:ubl:schema:xsd:ApplicationResponse-
2::ApplicationResponse##urn:www.cenbii.eu:transaction:biitrns0071:ver2.0:ex
tended:urn:www.peppol.eu:bis:peppol36a:ver2.0::2.1
```

Example:

```
<Scope>
  <Type>DOCUMENTID</Type>
  <InstanceIdentifier>
urn:oasis:names:specification:ubl:schema:xsd:ApplicationResponse-
2::ApplicationResponse##urn:www.cenbii.eu:transaction:biitrns0071:ver2.0:ex
tended:urn:www.peppol.eu:bis:peppol36a:ver2.0::2.1
  </InstanceIdentifier>
</Scope>
```

11.3 Process/Profile identification

The element

StandardBusinessDocumentHeader/BusinessScope/Scope [Type/text ()='PROCESSID'] /InstanceIdentifier **MUST** contain the process identifier of the MLR.

The process identifier to be used is:

```
urn:www.cenbii.eu:profile:bii36a:ver2.0
```

Example:

```
<Scope>
  <Type>PROCESSID</Type>
  <InstanceIdentifier>
urn:www.cenbii.eu:profile:bii36a:ver2.0
  </InstanceIdentifier>
</Scope>
```

11.4 Overview of envelope mapping

The following table summarizes the sources of the MLR envelope values.

MLR envelope element	Described in	Value taken from
Sender Party	11.1.1	Receiver Party of original envelope
Receiver Party	11.1.2	Sender Party of original envelope
Document identification	11.2.1	Constant values
Document type identifier	11.2.2	Based on Document type identifier of original envelope
Process identification	11.3	Based on Profile identifier of original envelope

12 Description of selected parts of the MLR message

This document requires that the MLR document is encoded as a UBL 2.1 document of type ApplicationResponse.

12.1 Namespace URIs

The target namespace for the UBL 2.1 Application Response which the PEPPOL Message Level Response is based on is:

`urn:oasis:names:specification:ubl:schema:xsd:ApplicationResponse-2`

12.2 UBL Version ID

This BIS is using the UBL 2.1 syntax. The namespace of the XML-message does always communicate the major version number. Since it is important for the receiver to also know what minor version of the syntax that is used, the element `cbc:UBLVersionID` must be stated with the value 2.1.

Example:

```
<cbc:UBLVersionID>2.1</cbc:UBLVersionID>
```

12.3 Customization ID

The PEPPOL Customization ID identifies the specification of content and rules that apply to the transaction. The element `cbc:CustomizationID` MUST have the value

`urn:www.cenbii.eu:transaction:biitrns071:ver2.0:extended:urn:www.peppol.eu:bis:peppol36a:ver2.0`.

Example:

```
<cbc:CustomizationID  
schemeID="PEPPOL">urn:www.cenbii.eu:transaction:biitrns071:ver2.0:extended:  
urn:www.peppol.eu:bis:peppol36a:ver2.0</cbc:CustomizationID>
```

12.4 Profile ID

The Profile ID identifies the process the business document is part of. PEPPOL BIS uses the identification system according to BII. The element `cbc:ProfileID` MUST have the value

`urn:www.cenbii.eu:profile:bii36a:ver2.0`.

Example:

```
<cbc:ProfileID>urn:www.cenbii.eu:profile:bii36a:ver2.0</cbc:ProfileID>
```

12.5 Party identification

12.5.1 Sender Party

The element `cac:SenderParty/cbc:EndpointID` MUST contain the party identification of the receiver of the original envelope.

Example:

```
<cac:SenderParty>  
  <cbc:EndpointID schemeID="NO:ORGNR">981915550</cbc:EndpointID>  
</cac:SenderParty>
```

12.5.2 Receiver Party

The element `cac:ReceiverParty/cbc:EndpointID` MUST contain the party identification of the sender of the original envelope.

Example:

```
<cac:ReceiverParty>
  <cbc:EndpointID schemeID="GLN">7315458756324</cbc:EndpointID>
</cac:ReceiverParty>
```

12.6 Document response

The document response is used to indicate the result of business document validation. See chapter 7.1 for all code list values. Exactly one `cac:DocumentResponse` element MUST be present. The element `cac:DocumentResponse/cac:Response/cbc:ResponseCode` MUST contain the overall result code. In case the document is rejected, the `cbc:Description` element MUST be set.

Example for acceptance:

```
<cac:Response>
  <cbc:ResponseCode listID="UNCL4343">AP</cbc:ResponseCode>
</cac:Response>
```

Example for acknowledging:

```
<cac:Response>
  <cbc:ResponseCode listID="UNCL4343">AB</cbc:ResponseCode>
</cac:Response>
```

Example for rejection:

```
<cac:Response>
  <cbc:ResponseCode listID="UNCL4343">RE</cbc:ResponseCode>
  <cbc:Description languageID="en">The document was rejected because ....
  This is a human readable text.... </cbc:Description>
</cac:Response>
```

12.7 Document reference

The document reference is used to provide a reference to the envelope of the business document on which the message level response is based. The message level response message may only cover exactly one business document. The element `cac:DocumentResponse/cac:DocumentReference/cbc:ID` MUST contain the instance identifier of the envelope of the original business document.

Example:

```
<cac:DocumentReference>
  <cbc:ID>uuid:9a2cefeb-2d2a-462e-bb2d-5898b7929b0c</cbc:ID>
</cac:DocumentReference>
```

12.8 Line response

In case of a negative response (rejection), the `cac:LineResponse` element is used to specify the errors in the business document. The `cbc:LineID` element is mandatory in UBL 2.1 but not relevant since the line responses are listing errors on both header level and line level from the original document and should have the value: NA (“Not Applicable”, must be all uppercase). The `cbc:XPath` element should be used to indicate where in the business document the error occurred (or other method for non-xml syntaxes) to reference the element causing the error. The `cbc:Description` element must be expressed in English only.

Example:

```
<cac:LineResponse>
  <cac:LineReference>
    <cbc:LineID>NA</cbc:LineID>
  <cac:DocumentReference>
    <cbc:ID>NA</cbc:ID>
```



```

    <cbc:XPath>
      Invoice/cac:LegalMonetaryTotal/cbc:TaxExclusiveAmount
    </cbc:XPath>
  </cac:DocumentReference>
</cac:LineReference>
<cac:Response>
  <cbc:ReferenceID>BIIRULE-T10-R012</cbc:ReferenceID>
  <cbc:Description>Invoice tax exclusive amount MUST equal the sum of
lines plus allowances and charges on header level</cbc:Description>
  <cac:Status>
    <cbc:StatusReasonCode listID="PEPPOLSTATUS">RVF</cbc:StatusReasonCode>
  </cac:Status>
</cac:Response>
</cac:LineResponse>

```

12.9 Overview of MLR message mapping

The following table summarizes the sources of the MLR message values.

MLR message element	Described in	Value taken from
Namespace URI	12.1	Constant value
UBL Version ID	12.2	Constant value
Customization ID	12.3	Constant value
Profile ID	12.4	Constant value
Sender Party	12.5.1	Receiver Party of original envelope
Receiver Party	12.5.2	Sender Party of original envelope
Document reference	12.7	Instance Identifier of original envelope

13 XML Schema Guideline and information content

13.1 XML Schema Guideline

Occurrence	Element/Attribute	BII Business Term	Business req.
ApplicationResponse			
1 .. 1	cbc:UBLVersionID	UBL version ID	1
1 .. 1	cbc:CustomizationID	Customization identifier	tir71-006
1 .. 1	cbc:ProfileID	Profile identifier	tir71-005
1 .. 1	cbc:ID	Response identifier	tir71-001
1 .. 1	cbc:IssueDate	Response issue date	tir71-002
0 .. 1	cbc:IssueTime	Response issue time	tir71-003
1 .. 1	cac:SenderParty		
1 .. 1	cbc:EndpointID	Electronic address	tir71-009
1 .. 1	cac:ReceiverParty		
1 .. 1	cbc:EndpointID	Electronic address	tir71-012
1 .. 1	cac:DocumentResponse		
1 .. 1	cac:Response		
1 .. 1	cbc:ResponseCode	Message response code	tir71-014
0 .. 1	cbc:Description	Response textual notes	tir71-004
1 .. 1	cac:DocumentReference		
1 .. 1	cbc:ID	Document identifier	tir71-018
0 .. unbounded	cac:LineResponse		
1 .. 1	cac:LineReference		
1 .. 1	cbc:LineID		
0 .. 1	cac:DocumentReference		
1 .. 1	cbc:ID		
1 .. 1	cbc:XPath	Section identification (XPATH)	tir71-013
1 .. 1	cac:Response		
0 .. 1	cbc:ReferenceID	Applied specification (RuleID)	tir71-017
1 .. 1	cbc:Description	Issue description	tir71-015
1 .. 1	cac:Status		
1 .. 1	cbc:StatusReasonCode	Issue type coded	tir71-016

13.2 Information Content

Element/Attribute	Description	Usage/Rules/Code lists
ApplicationResponse	Type ApplicationResponseType	
— cbc:UBLVersionID	Occurrence 1 .. 1 Type cbc:UBLVersionIDType Info req.ID 1	Term name UBL version ID BII Usage <i>The version of UBL the Message Level Response is based on (2.1)</i>
— cbc:CustomizationID	Occurrence 1 .. 1 Type cbc:CustomizationIDType Info req.ID tir71-006	Term name Customization identifier BII Usage <i>Identifies the specification of content and rules that apply to the transaction.</i> Rules BII2-T71-R002 - A message level response MUST have a customization identifier
— cbc:ProfileID	Occurrence 1 .. 1 Type cbc:ProfileIDType Info req.ID tir71-005	Term name Profile identifier BII Usage <i>Identifies the BII profile or business process context in which the transaction appears.</i> Rules BII2-T71-R001 - A message level response MUST have a profile identifier
— cbc:ID	Occurrence 1 .. 1 Type cbc:IDType Info req.ID tir71-001 Bus req.ID tbr71-001	Term name Response identifier BII Usage <i>An transaction instance must contain an identifier. The identifier enables positive referencing the transaction instance for various purposes including referencing between transactions that are part of the same process.</i> Rules BII2-T71-R004 - A message level response MUST contain the response identifier
— cbc:IssueDate	Occurrence 1 .. 1 Type cbc:IssueDateType Info req.ID tir71-002 Bus req.ID tbr71-002	Term name Response issue date BII Usage <i>The date on which the transaction instance was issued.</i> Rules BII2-T71-R003 - A message level response MUST contain the date of issue
— cbc:IssueTime	Occurrence 0 .. 1 Type cbc:IssueTimeType Info req.ID tir71-003 Bus req.ID tbr71-002	Term name Response issue time BII Usage <i>The time at which the transaction instance was issued.</i>
— cac:SenderParty	Occurrence 1 .. 1 Type cac:PartyType	

Element/Attribute	Description	Usage/Rules/Code lists
<pre> graph TD Root[] --- cbc1[cbc:EndpointID] Root --- scheme1[schemeID] Root --- cac1[cac:ReceiverParty] Root --- cac2[cac:DocumentResponse] cac1 --- cbc2[cbc:EndpointID] cac1 --- scheme2[schemeID] cac2 --- cac3[cac:Response] cac3 --- cbc3[cbc:ResponseCode] cac3 --- listID[listID] </pre>	<p>Occurrence 1 .. 1 Type cbc:EndpointIDType Info req.ID tir71-009 Bus req.ID tbr71-004</p>	<p>Term name Electronic address BII Usage A response may contain the party electronic address. The address can be of any format and the format should be identified in the message.</p> <p>Rules EUGEN-T71-R004 - An endpoint identifier MUST have a 'schemeID' attribute.</p>
<p>schemeID</p>	<p>Type xs:normalizedString Use required</p>	<p>Rules OP-T71-R001 - An Endpoint Identifier Scheme MUST be from the list of PEPPOL Party Identifiers described in the "PEPPOL Policy for using Identifiers".</p>
<p>cac:ReceiverParty</p>	<p>Occurrence 1 .. 1 Type cac:PartyType</p>	
<p>cbc:EndpointID</p>	<p>Occurrence 1 .. 1 Type cbc:EndpointIDType Info req.ID tir71-012 Bus req.ID tbr71-005</p>	<p>Term name Electronic address BII Usage A response may contain the party electronic address. The address can be of any format and the format should be identified in the message.</p> <p>Rules EUGEN-T71-R004 - An endpoint identifier MUST have a 'schemeID' attribute.</p>
<p>schemeID</p>	<p>Type xs:normalizedString Use required</p>	<p>Rules OP-T71-R001 - An Endpoint Identifier Scheme MUST be from the list of PEPPOL Party Identifiers described in the "PEPPOL Policy for using Identifiers".</p>
<p>cac:DocumentResponse</p>	<p>Occurrence 1 .. 1 Type cac:DocumentResponseType</p>	
<p>cac:Response</p>	<p>Occurrence 1 .. 1 Type cac:ResponseType</p>	
<p>cbc:ResponseCode</p>	<p>Occurrence 1 .. 1 Type cbc:ResponseCodeType Info req.ID tir71-014 Bus req.ID tbr71-010, tbr71-012</p>	<p>Term name Message response code BII Usage An indicator stating whether the referenced message was cleared through validation and advanced to the next step in the process. A negative response states that the document was not processed because of identified issues.</p> <p>Rules CL-071-R002 - A response code MUST be from the UNCL 4343 PEPPOL Subset code list. EUGEN-T71-R008 - A response document MUST indicate whether the received document was accepted or not.</p>
<p>listID</p>	<p>Type xs:normalizedString Use required</p>	<p>Rules EUGEN-T71-R002 - A response code MUST have a list identifier attribute 'UNCL4343'.</p>

Element/Attribute	Description	Usage/Rules/Code lists
cbc:Description	Occurrence 0 .. 1 Type cbc:DescriptionType Info req.ID tir71-004 Bus req.ID tbr71-003	Term name Response textual notes BII Usage <i>Used to make any comments or instructions relevant to the response. The use of this element requires manual assessment by the receiver.</i>
cac:DocumentReference	Occurrence 1 .. 1 Type cac:DocumentReferenceType	
cbc:ID	Occurrence 1 .. 1 Type cbc:IDType Info req.ID tir71-018 Bus req.ID tbr71-006	Term name Document identifier BII Usage <i>Identifies the document being referred to.</i> Rules EUGEN-T71-R006 - A message level response MUST contain a document reference pointing towards the envelope of the business document that the response relates to
cac:LineResponse	Occurrence 0 .. unbounded Type cac:LineResponseType	
cac:LineReference	Occurrence 1 .. 1 Type cac:LineReferenceType	
cbc:LineID	Occurrence 1 .. 1 Type cbc:LineIDType	Rules Must be value NA
cac:DocumentReference	Occurrence 0 .. 1 Type cac:DocumentReferenceType	
cbc:ID	Occurrence 1 .. 1 Type cbc:IDType	Rules Must be value NA
cbc:XPath	Occurrence 1 .. 1 Type cbc:XPathType Info req.ID tir71-013 Bus req.ID tbr71-009, tbr71-013	Term name Section identification (XPATH) BII Usage <i>Identifies the section of the document to which the reported issue applies.</i>
cac:Response	Occurrence 1 .. 1 Type cac:ResponseType	
cbc:ReferenceID	Occurrence 0 .. 1 Type cbc:ReferenceIDType Info req.ID tir71-017 Bus req.ID tbr71-008, tbr71-013	Term name Applied specification (RuleID) BII Usage <i>An identification of the specification that was applied when the reported issue was identified.</i>
cbc:Description	Occurrence 1 .. 1 Type cbc:DescriptionType Info req.ID tir71-015 Bus req.ID tbr71-008, tbr71-013, tbr71-014	Term name Issue description BII Usage <i>The description of the issue identified in the transaction document.</i>
cac:Status	Occurrence 1 .. 1	

Element/Attribute	Description	Usage/Rules/Code lists	
<div style="border-left: 1px solid black; border-top: 1px solid black; padding-left: 5px;"> <p>cbc:StatusReasonCode</p> </div>	<p>Type cac:StatusType</p> <p>Occurrence 1 .. 1</p> <p>Type cbc:StatusReasonCodeType</p> <p>Info req.ID tir71-016</p> <p>Bus req.ID tbr71-007, tbr71-013</p>	<p>Term name</p> <p>Bill Usage</p> <p>Rules</p>	<p>Issue type coded</p> <p><i>A codified version of the issue description that describes the nature of the issue e.g. Syntax violation, Business rule violation, etc.</i></p> <p>OP-071-R003 - An status reason code MUST be from PEPPOL Status code list</p> <p><i>EUGEN-T71-R003 - A status reason code MUST have a 'listID' attribute value of 'PEPPOLSTATUS'</i></p>
<div style="border-left: 1px solid black; border-top: 1px solid black; padding-left: 5px;"> <p><i>listID</i></p> </div>	<p><i>Type</i> xs:normalizedString</p> <p><i>Use</i> required</p>	<p><i>Rules</i></p>	

14 Appendices

14.1 Appendix A – XML for Use cases

14.1.1 Use case 1 – Response with acceptance

14.1.2 Use case 2 – Response with acknowledgment

14.1.3 Use case 3 – Response with reject – violation of business rules

- Example with a single violation
- Example with 2 violations

14.2 Appendix B – Conformance statement