



The future is open
Peppol

PINT A-NZ Self-billing BIS

Table of Contents

Link to main site of documentation	2
Introduction	3
Document Structure	3
Scope	4
Audience	4
1. Business processes	5
1.1. Parties and roles	5
1.2. PINT A-NZ Self-billing Process	5
1.3. Tax Information	6
1.4. Self-billed Invoice functionality	8
1.5. Self-billed Credit notes and negative invoices	10
2. Rules	11
3. Technical details	12
3.1. BIS Identifiers	12
3.2. Datatypes	13
3.3. UBL schemas and namespaces	17
3.4. Glossary	17

Link to main site of documentation

[Main documentation site](#)

Introduction

This Peppol Business Interoperability Specification (BIS) is concerned with clarifying requirements for ensuring interoperability, and provides a set of specifications for implementing a self-billing business process. It is based on the Peppol International (PINT) Billing specification and includes localisations to support Australian and New Zealand business and legal requirements.

This specification is optional and service providers can choose to support this as an additional service offering.

The purpose of this document is to describe the use of the self-billed invoice and self-billed credit note messages in Peppol, and to facilitate an efficient implementation and increased use of electronic collaboration regarding the self-billing process based on these formats. For simplicity only the differences to the [PINT A-NZ Billing specification](#) are included in this document, and otherwise the content from the [PINT A-NZ Billing specification](#) should be used for implementation.

Statement of copyright

This Peppol Business Interoperability Specification (Peppol BIS) document is a Country Specification based on the PINT. The restrictions on PINT implemented in this Peppol BIS are identified in the conformance statement provided in appendix A.

The copyright of PINT is owned by OpenPEPPOL and its members. OpenPEPPOL AISBL holds the copyright of this Peppol BIS.

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

Document Structure

This document is structured as follows:

- **Chapter 1** gives general information on the business processes, requirements and functionalities.
- **Chapter 2** provides information about rules that apply to the invoice content.
- **Chapter 3.1** describes the BIS identifiers.
- **Chapter 3.2** describes the semantical data types.
- **Chapter 3.3** gives external links to the relevant UBL schemas.

The data model and validation rules for PINT A-NZ billing and PINT A-NZ self-billing are largely identical. Details on the self-billing invoicing process and the main differences to the billing invoicing process are provided at [Section 1.2](#).

Scope

This document is concerned with clarifying requirements for PINT A-NZ self-billing. For simplicity only the differences to the [PINT A-NZ Billing specification](#) are included in this document, and otherwise the content from the [PINT A-NZ Billing specification](#) should be used as guidelines for the support and implementation of these requirements.

Audience

This document is for organisations that would like to exchange electronic invoices, and/or their ICT-suppliers. These organisations may be:

- Service providers
- Contracting Authorities (CA)
- Economic Operators (EO)
- Software Developers

More specifically, roles addressed are the following:

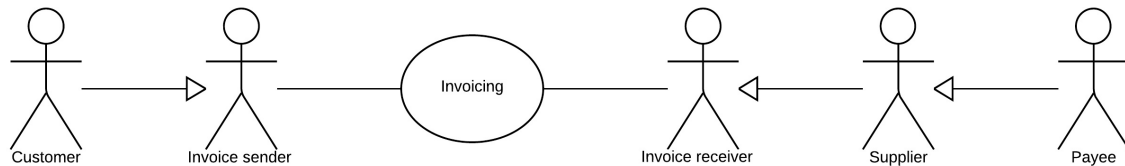
- ICT Architects
- ICT Developers
- Business Experts

For further information on Peppol/OpenPEPPOL see <http://peppol.org>

Chapter 1. Business processes

1.1. Parties and roles

The diagram below shows the roles involved in the self-billing invoice and self-billing credit note transactions. The customer and invoice sender is the same entity, as is the supplier and the invoice receiver.



1.1.1. Parties

Customer

The customer is the legal person or organisation who is in demand of a product or service. Examples of customer roles: buyer, consignee, debtor, contracting authority.

Supplier

The supplier is the legal person or organisation who provides a product or service.

1.1.2. Roles

Creditor

One to whom a debt is owed. The party that claims the payment and is responsible for resolving billing issues and arranging settlement. The party that receives the self-billed invoice or self-billed credit note. Also known as invoicee, accounts receivable or seller.

Debtor

One who owes debt. The party responsible for making settlement relating to a purchase. The party that sends the self-billed invoice or self-billed credit note. Also known as invoice issuer, accounts payable, or buyer.

1.2. PINT A-NZ Self-billing Process

The following sections describe how the PINT A-NZ self-billing specification is applied to support local business processes in Australia and New Zealand.

Self-billing describes the process where the customer values the goods or services and issues the invoice on the suppliers behalf. This process occurs based on mutual agreement between the supplier and customer, or the document can be used as the agreement itself.

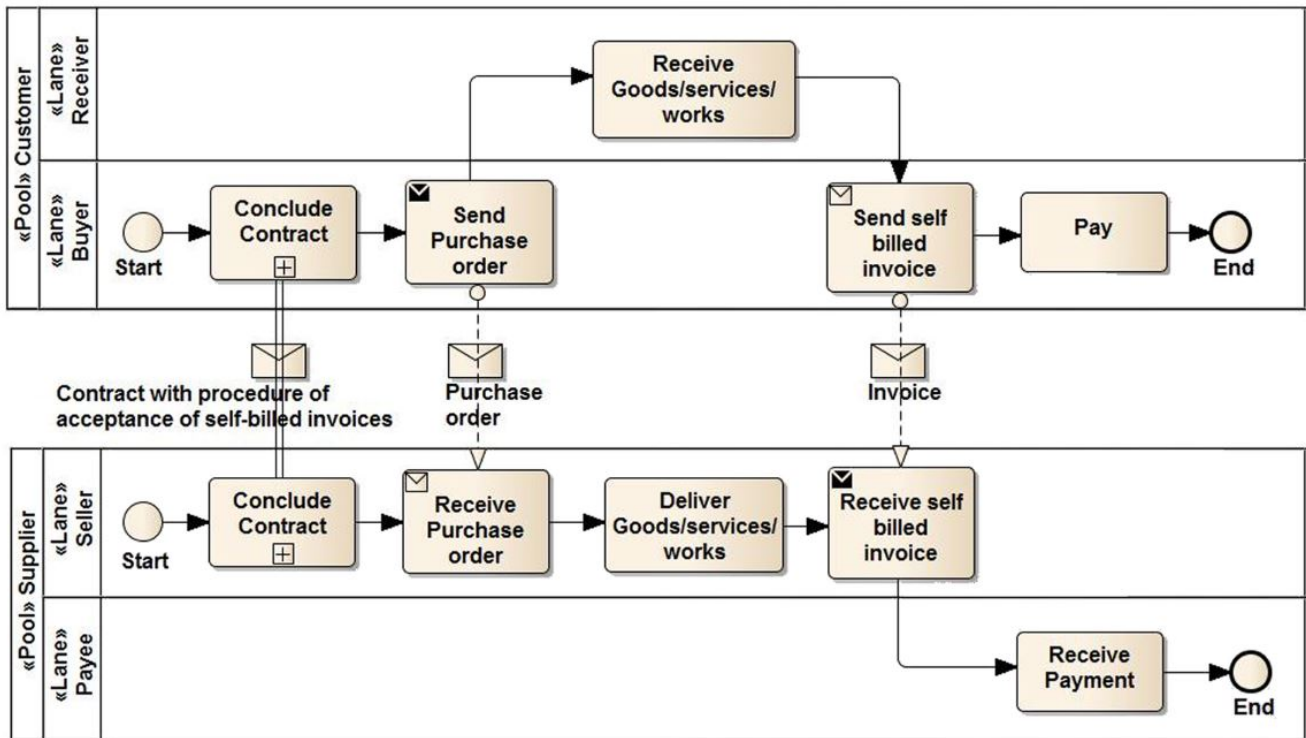
The document created in this process is known as:

- Recipient Created Tax Invoice (RCTI) in AU; or

- Buyer Created Tax Invoice (BCTI) in NZ

The data model and validation rules for the PINT A-NZ self-billing process are largely identical to the PINT A-NZ billing process. As per the PINT A-NZ billing process all supplier details shall still be provided within **cac:AccountingSupplierParty** (ibg-04) and all buyer details shall be provided within **cac:AccountingCustomerParty** (ibg-07). The main difference to note is that the self-billing process reverses the business initiation point. The switch of Party roles, with the customer sending the self-billed invoice to the supplier, is reflected in the message header (Standard Business Document Header, SBDH) level.

The diagram below illustrates the self-billing process in more detail.



As the customer issues the invoice and the supplier receives it, processing by the customer and supplier is different to the invoice billing process and requires self-billing uses a different specification identifier **cbc:CustomizationID** (ibt-024) and business process **cbc:ProfileID** (ibt-023) See section 2.1.1.

This allows separate recording of this capability in the Peppol network, and supports processing into the accounts receivable system of the supplier, which occurs during the self-billing process, rather than accounts payable system of the customer.

The self-billed invoice/credit note is optional and end-users and Service Providers can choose to support this process as a separate service. If supported a supplier will advertise their capability to receive the self-billed invoice from the buyer in the Service Metadata Publisher (SMP).

1.3. Tax Information

1.3.1. A-NZ Self-billing Tax Invoices

In Australia and New Zealand self-billing tax invoices are defined by law and have minimum requirements set out in relevant legislation. Self-billing tax invoices must be retained for a business to claim goods and services tax (GST) credits and report the amount of GST collected during the sale of taxable supplies.

Information relating to Australian (AU) RCTI requirements can be found [here](#).

Information relating to New Zealand (NZ) BCTI requirements can be found [here](#).

The free text `cbc:Note` (ibt-022) on document level can be used to meet the relevant legislative requirements of a RCTI or BCTI.



The Peppol eInvoicing standard can be used to issue an invoice that meets legal requirements. However, it does not guarantee or enforce compliance because requirements vary based on business scenarios. It is the trading entity's obligation to understand and comply with the legal and record keeping requirements.

UBL example of self-billing mutual agreement using note field

```
<!--code omitted for clarity-->
<cbc:DueDate>2019-08-30</cbc:DueDate>
<cbc:InvoiceTypeCode>389</cbc:InvoiceTypeCode>
<cbc:Note>Buyer-created tax invoice - Agreement between the buyer and
supplier.</cbc:Note>
<!--code omitted for clarity-->
```



The text used in the above UBL example should not be relied upon and is for demonstrative purposes only. This agreement (in free text format) will not be validated at runtime as this text may change due to legislative requirements.

1.3.2. Self-billing Invoice Document Type

This specification uses the [UBL 2.1 Invoice](#) document as the base schema, and uses a subset of the document name code [UN/CEFACT Code list 1001, D.16B](#) with the following allowed values: for `cbc:InvoiceTypeCode` (ibt-003) or `cbc:CreditNoteTypeCode` (ibt-003):

- 389 Self-billed invoice (subset: Invoice Type Code)
- 261 Self-billed credit note (subset: Credit Note Type Code)

UBL example of self-billing invoice type code in Australia

```
<cbc:InvoiceTypeCode>389</cbc:InvoiceTypeCode>
```

1.4. Self-billed Invoice functionality

A self-billed invoice may support functions related to a number of related (internal) business processes. This Peppol BIS shall support the following functions:

- Accounting
- Self-billed Invoice verification against the contract, the purchase order and the goods and service delivered
- Tax reporting
- Auditing
- Payment

In the following chapters an assessment is made of what information is needed for each of the functions listed above and whether it is in scope or out of scope for this Peppol BIS.

Explicit support for the following functions (but not limited to) is out of scope:

- Inventory management
- Delivery processes
- Customs clearance
- Marketing
- Reporting

1.4.1. Accounting

Recording a business transaction into the financial accounts of an organization is one of the main objectives of the self-billed invoice. According to financial accounting best practice and Tax rules every Taxable person shall keep accounts in sufficient detail for Tax to be applied and its application checked by the tax authorities. For that reason, a self-billed invoice shall provide for the information at document and line level that enables booking on both the debit and the credit side.

1.4.2. Self-billed Invoice verification

This process forms part of the suppliers internal business controls. The self-billed invoice shall refer to an authentic commercial transaction. Support for self-billed invoice verification is a key function of a self-billed invoice. The self-billed invoice should provide sufficient information to look up relevant existing documentation, electronic or paper, for example, and as applicable:

- the relevant purchase order
- the contract
- the call for tenders, that was the basis for the contract
- the Suppliers reference
- the confirmed delivery and information of the goods or services

A self-billed invoice should also contain sufficient information that allows the received self-billed invoice to be transferred to a responsible authority, person or department, for verification and approval.

1.4.3. Auditing

Companies audit themselves as means of internal control or they may be audited by external parties as part of a legal obligation. Accounting is a regular, ongoing process whereas an audit is a separate review process to ensure that the accounting has been carried out correctly. The auditing process places certain information requirements on a self-billed invoice. These requirements are mainly related to enable verification of authenticity and integrity of the accounting transaction.

Self-billed Invoices, conformant to this Peppol BIS support the auditing process by providing sufficient information for:

- identification of the relevant buyer and seller
- identification of the products and services traded, including description, value and quantity
- information for connecting the self-billed invoice to its payment
- information for connecting the self-billed invoice to relevant documents such as a contract and a purchase order

1.4.4. Tax Reporting

The self-billed invoice is used to carry Tax related information from the buyer to the seller to enable the buyer and seller to correctly handle Tax booking and reporting. A self-billed invoice should contain sufficient information to enable the seller and any auditor to determine whether the self-billed invoice is correct from a Tax point of view.

The self-billed invoice shall allow the determination of the Tax regime, the calculation and description of the Tax, in accordance with the relevant legislation.

1.4.5. Payment

A self-billed invoice represents a claim for payment. The issuance of a self-billed invoice may take place either before or after the payment is carried out. When a self-billed invoice is issued before payment it represents a commitment by the buyer to pay, in which case the self-billed invoice commonly contains information that enables the buyer, in the role of a debtor, to correctly initiate the transfer of the payment, unless that information is already agreed in prior contracts or by means of payment instructions separately lodged with the Buyer.

If a self-billed invoice is issued after payment, such as when the order process included payment instructions or when paying with a credit card, online or telephonic purchases, the self-billed invoice may contain information about the payment made in order to facilitate invoice to payment reconciliation on the seller side. A self-billed invoice may be partially paid before issuing such as when a pre-payment is made to confirm an order.

1.5. Self-billed Credit notes and negative invoices

Reverting a self-billed invoice that has been issued and received can be done in two basic ways. Either by issuing a self-billed credit note or a self-billed negative invoice.

- When crediting by means of a self-billed credit note, the `cbc:CreditNoteTypeCode` (ibt-003) is '261' (or its synonym), and the self-billed credit note quantities and extension/total amounts have the same sign (plus or minus) as the self-billed invoice that is being cancelled/credited. The document type code acts as an indicator that the given amounts are booked in reverse and cancel out the self-billed invoice amounts.
- When crediting by means of a self-billed negative invoice, the `cbc:InvoiceTypeCode` (ibt-003) is '389' (or its synonym), and the negative self-billed invoice quantities and extension/total amounts have the opposite sign (minus vs plus) as the self-billed invoice being cancelled/credited. It is the mathematical sign that indicates that when the amounts are booked they cancel out the original amounts. The Price Amount must always be positive.



A self-billed credit note may include negative amounts when cancelling a self-billed invoice that may have negative line items/amounts.

Chapter 2. Rules

The information given in a PINT self-billed invoice must comply with a set of rules on the content of the business terms as well as the relationship between them.

The rules for PINT A-NZ Self-billed invoices and credit notes are the same as for PINT A-NZ invoices and credit notes, except for the following:

Table 1. PINT A-NZ Self-billing rules that are different to PINT A-NZ Billing

PINT A-NZ Self-billing rule	PINT A-NZ Billing rule	Reason for difference
ibr-cl-01-aunz-sb	ibr-cl-01	Self-billing document type codes
aligned-ibrp-001-aunz-sb	aligned-ibrp-001-aunz	PINT A-NZ Self-billing specification identifier
aligned-ibrp-002-aunz-sb	aligned-ibrp-002	PINT Self-billing business process

Chapter 3. Technical details

Following section provide technical details.

3.1. BIS Identifiers

Peppol has a defined policy that specifies how to use identifiers in both its transport infrastructure and within the documents exchanged across that infrastructure. It also introduces principles for any identifiers used in the Peppol environment. The policies that apply to this BIS are the following:

3.1.1. Profiles and messages

All messages contain a Business process type (ibt-023) and a Specification identifier (ibt-024). A Business process type (ibt-023) identifies what business process a given message is part of, and a Specification identifier (ibt-024) identifies the kind of message and the rules applied.

Profiles are connected to one business process, and may contain multiple document types. Valid document instances shall contain corresponding Business process type (ibt-023) and Specification identifier (ibt-024).

3.1.2. Identifying the A-NZ Self-billing specification

The UBL element `cbc:ProfileID` (ibt-023) identifies the Business process type in which the transaction appears.

The UBL element `cbc:CustomizationID` (ibt-024) contains the Specification identifier.

In the table below you will find the values to be used as the Specification identifier (ibt-024) and the Business process type (ibt-023) for the self-billing profile.

Type	Element <code>cbc:CustomizationID</code>	Element <code>cbc:ProfileID</code>
Australian and New Zealand Self-Billing Tax Invoice and Self-Billing Credit Note	urn:peppol:pint:selfbilling-1@aunz-1	urn:peppol:bis:selfbilling

UBL example of customization & profile identifier

```
<cbc:CustomizationID>urn:peppol:pint:selfbilling-1@aunz-1</cbc:CustomizationID>  
<cbc:ProfileID>urn:peppol:bis:selfbilling</cbc:ProfileID>
```

Document Type Identifier Schemes

This BIS supports two document type identifier schemes, busdox-docid-qns and peppol-doctype-

wildcard, as described in the [Peppol Policy for Use of Identifiers](#). When performing a lookup to determine a match the busdox-docid-qns scheme takes precedence over peppol-doctype-wildcard.

Example matching customization identifiers as they may appear in SMP registrations:

Document Type Identifier Scheme	Customization Identifier
busdox-docid-qns	urn:peppol:pint:selfbilling-1@aunz-1
peppol-doctype-wildcard	urn:peppol:pint:selfbilling-1*

Note: Detailed instructions on how the customization ID corresponds to other identifiers used in the Peppol eDelivery network can be found in the eDelivery network specifications in the [Peppol Policy for use of Identifiers](#).

3.2. Datatypes

Semantic data types are used to bridge the gap between the semantic concepts expressed by the information elements defined in the semantic model and the technical implementation. The semantic data types define the allowed value domain for the content, and any additional information components (attributes) needed in order to ensure its precise interpretation.

3.2.1. Primitive types

Semantic data type content may be of the following primitive types. These primitive types were taken from ISO15000, Annex A.

Primitive type	Definition
Binary	A set of finite-length sequences of binary digits.
Date	Time point representing a calendar day on a time scale consisting of an origin and a succession of calendar ISO8601.
Decimal	A subset of the real numbers, which can be represented by decimal numerals.
String	A finite sequence of characters.

3.2.2. Semantic data types

The different semantic data types are described in the tables below, where various features such as attributes, format, and decimals as well as the basic type are defined for each semantic data type. They are based on {ISO15000}.

When used in an instance of an invoice, each data element will contain data. In the below tables this is identified as the “content”. Whenever a business term is used this term shall **always** have content and therefore the content is always mandatory.

Amount

An amount states a numerical monetary value. The currency of the amount is defined as a separate

business term.

Component	Use	Primitive Type	Example
Content	Mandatory	Decimal	10000

Unit Price Amount

A unit price amount states a numerical monetary amount value for data elements that contain item prices that may be multiplied by item quantities. The currency of the amount is defined as a separate business term.



Unit price amount does not set restrictions on number of decimals, as contrast to the Amount type

Component	Use	Primitive Type	Example
Content	Mandatory	Decimal	10000.1234

Percentage

Percentages are given as fractions of a hundred (per cent) e.g. the value 34.78 % in percentage terms is given as 34.78.



No restriction on number of decimals for percentages.

Component	Use	Primitive Type	Example
Content	Mandatory	Decimal	34.7812

Quantity

Quantities are used to state a number of units such as for items. The code for the Unit of Measure is defined as a separate business term.



No restriction on number of decimals for quantities.

Component	Use	Primitive Type	Example
Content	Mandatory	Decimal	10000.1234

Code

Codes are used to specify allowed values in elements as well as for lists of options. Code is different from Identifier in that allowed values have standardized meanings that can be known by the recipient.



Codes shall be entered exactly as shown in the selected code list of the applicable syntax.

Component	Use	Primitive Type	Example
Content	Mandatory	String	Abc123

Indicator

Indicators are used to give boolean values to state whether something is (true) or is not (false).



Indicators shall be used in lower case.



Default value is "false" and applies if the relevant business term is not used.

Component	Use	Primitive Type	Allowed values
Content	Mandatory	String	false
			true

Identifier

Identifiers (IDs) are keys that are issued by the sender or recipient of a document or by a third party.



The use of the attributes is specified for each information element.

Component	Use	Primitive Type	Example
Content	Mandatory	String	abc:123-DEF
Scheme identifier	Optional	String	GLN
Scheme version identifier	Optional	String	1.0

Date

Dates shall be in accordance to the “Calendar date complete representation” as specified by {ISO8601}, format **YYYY-MM-DD**.



Dates shall not include timezone information.

Table 2. EN 16931_ Date. Type

Component	Use	Primitive Type	Example
Content	Mandatory	Date	2017-12-01

Time

Time shall be according to UBL allowed format.



Time may include timezone information.

Table 3. EN 16931_ Date. Type

Component	Use	Primitive Type	Allowed forms
Content	Mandatory	Date	13:20:00 (1:30 PM)
			13:20:30.55 (30.55 sec)
			13:20:00Z (UTC)
			13:20:00-05:00 (UTC-5)
			00:00:00 (midnight)
			24:00:00 (midnight)

Time formats without time zone information (i.e. other than hh:mm:ssz and hh:mm:ss-h) shall be interpreted as being in the time zone of the sellers address and according to daylight saving status on the issue date of the invoice.

Document Reference

Document Reference Types are identifiers that were assigned to a document or document line by the Buyer, the Seller or by a third party.

Table 4. Document Reference. Type

Component	Use	Primitive Type	Example
Content	Mandatory	String	abc:123-DEF

Text

Text is the actual wording of anything written or printed. Line breaks in the text may be present, and any line breaks should be preserved and respected by the receiver's system

Component	Use	Primitive Type	Example
Content	Mandatory	String	5% allowance when paid within 30 days

Binary object

Binary object can be used to describe files which are transmitted together with the Invoice. The attachment functionality is not intended for attaching a copy of the invoice in an image format (such as PDF). Attaching an invoice copy is not in compliance with this specification.

Attachments shall be transmitted together with the Invoice. The binary object has two supplementary components: a Mime Code, which specifies the Mime type of the attachment and a Filename that is provided by (or on behalf of) the sender of the invoice or credit note.

Component	Use	Primitive Type	Example
Content	Mandatory	Binary	QmFzZTY0IGNvbnRlbnQgZXhhbXBsZQ==
Mime Code	Mandatory	String	image/jpeg
Filename	Mandatory	String	drawing5.jpg

A receiver of an invoice or credit note, shall accept and process attachments that are according to the Media type code list.

3.3. UBL schemas and namespaces

The XML schemas used are

- UBL Invoice 2.1, with the target namespace `urn:oasis:names:specification:ubl:schema:xsd:Invoice-2`
- UBL CreditNote 2.1 with the target namespace `urn:oasis:names:specification:ubl:schema:xsd:CreditNote-2`

3.4. Glossary

electronic invoice - invoice that has been issued, transmitted and received in a structured electronic format which allows for its automatic and electronic processing

semantic data model - structured set of logically interrelated information elements

information element - semantic concept that can be defined independent of any particular representation in a syntax

structured information element - information element that can be processed automatically

syntax - machine-readable language or dialect used to represent the information elements contained in an electronic document (e.g. an electronic invoice)

business term - label assigned to a given information element which is used as a primary reference

identifier - character string used to establish the identity of, and distinguish uniquely, one instance of an object within an identification scheme from all other objects within the same scheme Note 1 to entry: An identifier may be a word, number, letter, symbol, or any combination of those, depending on the identification scheme used.

identification scheme - collection of identifiers applicable for a given type of object governed under a common set of rules

compliant - some or all features of the invoice model are used and all rules of the invoice model are respected Note 1 to entry: Based on TOGAF definition of a compliant specification

conformant - all rules of the invoice model are respected and some additional features not defined

in the invoice model are also used

Optional Whether the option is used or not is the choice of the users independently from other data in the message.

Conditional Whether the option is used or not and in what way is dependent on other data in the message.

Mandatory The option must be used in all messages.

shall - the definition is an absolute requirement of the specification.

shall not - the definition is an absolute prohibition of the specification.

should - there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

should not - there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

may - is truly optional.