

The future is open

Peppol

Peppol model for Tax Data Document

Table of Contents

Link to main site of documentation . . . 2
- Introduction . . . 3
  - Scope . . . 3
  - Audience . . . 3
  - Benefits . . . 3
- 1. Business processes . . . 5
  - 1.1. Parties and roles . . . 5
  - 1.2. Tax Data Document process . . . 6
  - 1.3. Tax Data Document functionality . . . 7
  - 1.4. Tax Data Document Status functionality . . . 7
- 2. Rules . . . 9
  - 2.1. Custom content . . . 9
  - 2.2. Source document . . . 9
- 3. Technical details . . . 10
  - 3.1. BIS Identifiers . . . 10
  - 3.2. Datatypes . . . 10
  - 3.3. XML schemas and namespaces . . . 13
  - 3.4. Glossary . . . 13

OpenPeppol AISBL, Post-Award Coordinating Community v1.0.5

Link to main site of documentation

Main documentation site

Introduction

The purpose of this document is to specify a Tax Data Document (TDD) that may be used in the United Arab Emirates to report issued and received invoices to the UAE Federal Tax Authority (FTA) for the purpose of VAT reporting using continuous transaction control (CTC) methods.

Statement of copyright

OpenPeppol AISBL holds the copyright of this Peppol specification.

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

Scope

This document is concerned with clarifying the structure and functionality of the TDD transaction and how it shall be applied.

Audience

The audience for this document is organisations wishing to be Peppol enabled for CTC tax reporting in the UAE, and/or their ICT-suppliers. These organisations may be:

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

More specifically, roles addressed are the following:

- ICT Architects
- ICT Developers
- Business Experts

For further information on Peppol/OpenPeppol see https://peppol.org

Benefits

The TDD transaction may be used by either an invoice issuer or an invoice receiver, or a service provider on their behalf, to report the sending or reception of an invoice or a self-billed invoice to FTA, for the purpose of VAT declaration.

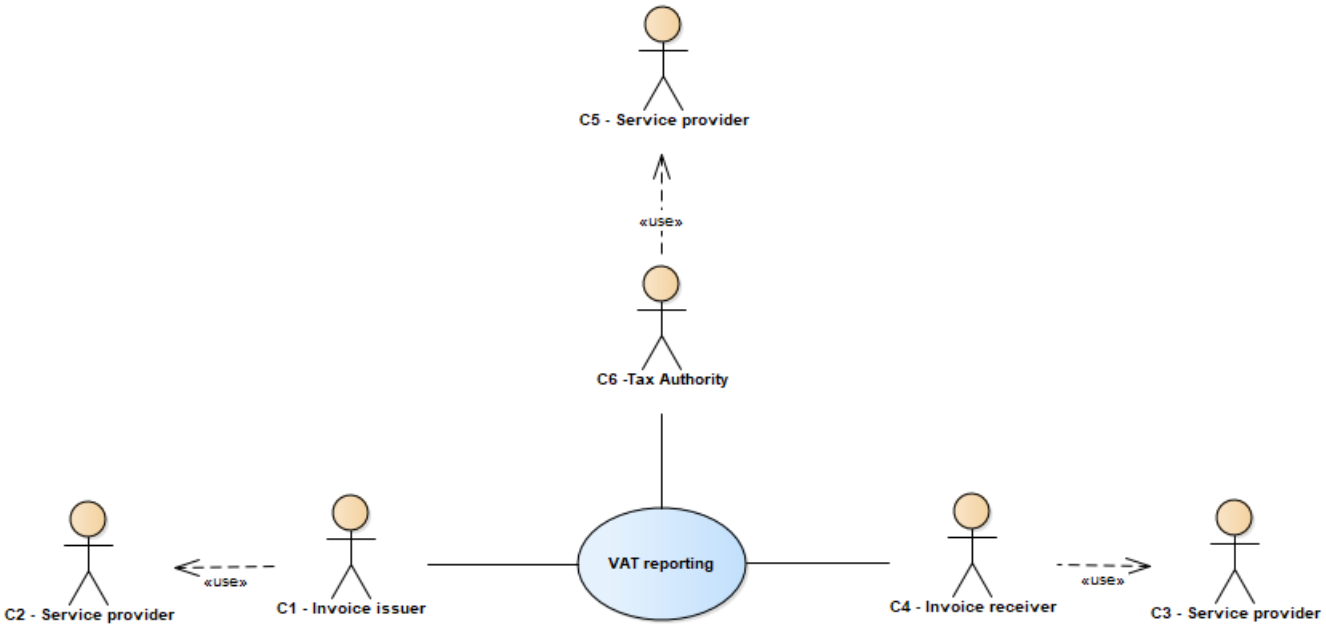
- The TDD can be used by either the invoice issuer or the invoice receiver.

- It provides information about the TDD message itself.
- It supports management of the TDD message such as submit, update, withdraw, that may be independent of the invoice exchange between the invoice issuer and the invoice receiver.
- It provides for different levels of detail about the invoice ranging from simple extract of basic information from the invoice to full inclusion of the invoice.
- It allows for full inclusion of an invoice either as attachment or in expanded form.

Chapter 1. Business processes

1.1. Parties and roles

The use-case diagram below shows the use case of VAT reporting and that the parties who interact with it are the invoice issuer (C1), invoice receiver (C4) and the Federal Tax Authority (FTA). The invoice issuer and the invoice receiver are the taxable persons that are responsible for reporting to FTA. Those parties will use the services of service providers (C2, C3 and C5).



1.1.1. Parties

Invoice issuer C1

The party who issues an invoice to an invoice receiver. This is usually the supplier but in a self-billing situation this is the buyer. The invoice may contain VAT that the invoice issuer, as a taxable person, is responsible for reporting to FTA.

Service provider C2

A party who provides electronic messaging services that include preparing and transmitting invoices and TDD transactions on behalf of the invoice issuer, C1.

Service provider C3

A party who provides electronic messaging services that include preparing and transmitting TDD transactions on behalf of the invoice receiver, C4, and receiving and forwarding the invoice to the invoice receiver, C4.

Invoice receiver C4

The party who receives the invoice from the invoice issuer. This is usually the buyer but in a self-billing situation this is the supplier. The invoice may contain VAT that the invoice receiver, as a taxable person, is responsible for reporting to FTA.

Service provider C5

A party who provides electronic messaging services that include receiving and managing TDD transactions on behalf of a tax authority, C6.

Federal Tax Authority C6

The UAE authority that is responsible for collecting VAT and enforcing relevant legislation.

1.1.2. Roles

Taxable person

An organization or an individual who is responsible for reporting and settling taxes in accordance with applicable legislation.

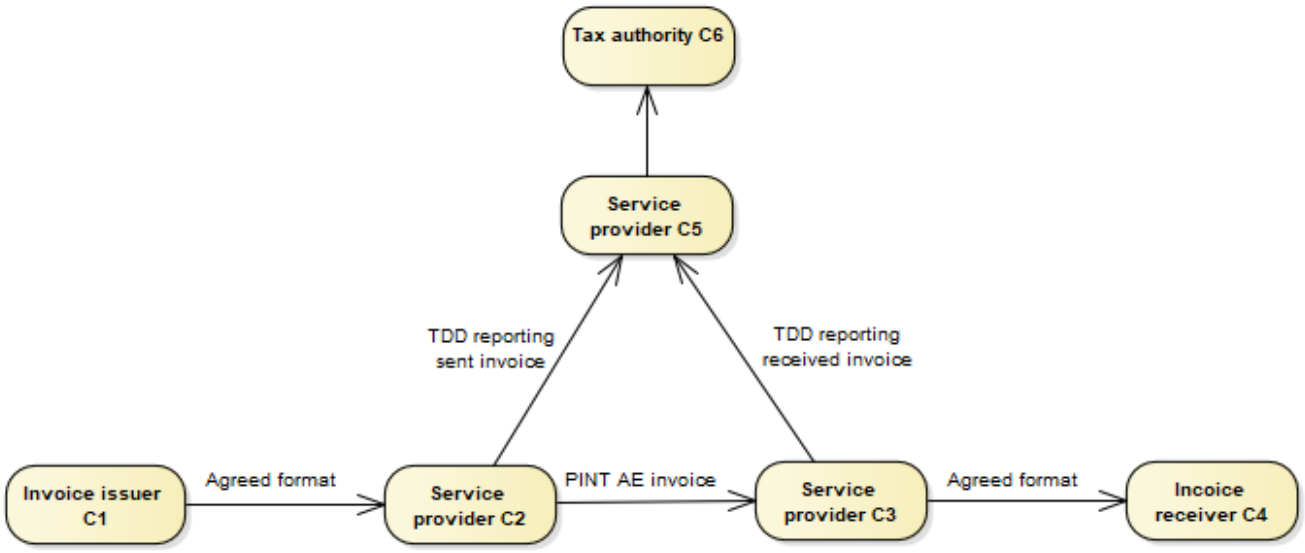
Service provider

A party who assists another party in carrying out his responsibilities.

1.2. Tax Data Document process

The TDD process is dependent on the legislation of UAE and the objectives that the invoicing process is intended to achieve in the UAE. This affects the content, the sequence and the timing of the TDD in the overall process.

The process using the TDD is described in the following diagram.



The invoice issuer (C1) provides the data of the invoice to his service provider (C2), using the data format agreed between them. With this data the C2 service provider creates a PINT-AE compliant invoice and transmits that to the invoice receiver (C4) through the receiver's service provider (C3) by using the Peppol network.

With sending the invoice the invoice issuer's service provider (C2) creates and sends a TDD document, reporting the issued invoice to FTA through its service provider (C5) by using the Peppol network.

The receiver's service provider (C3) forwards the invoice to the invoice receiver (C4) using the data

format agreed between them. Following that the invoice receiver's service provider (C3) creates and sends a TDD document reporting the reception of the invoice to FTA through its service provider (C5), using the Peppol network.

The FTA's service provider (C5) forwards the TDD document to FTA using a data format agreed between them.

1.3. Tax Data Document functionality

The TDD document contains three main sections:

- TDD document meta data
- Data extracts from the reported invoice
- Inclusion of the full invoice

1.3.1. TDD meta data

The TDD is a report that is based on its own specification. It is exchanged with FTA rather than between an invoice issuer and an invoice receiver and it may be sent at a different time.

The first part of the TDD document contains metadata that identifies the specification, the relevant parties and dates and time. It also provides codes that define the role of the parties in the exchange and the function of each instance.

- A given party may send TDD's either in the role of an invoice issuer or invoice receiver.
- There may be more than one instance of a TDD for the same invoice that have different functionality such as **submit**, **update** and **withdraw**.

1.3.2. Data extracts

Key data values shall be extracted from the reported invoice into the TDD.

1.3.3. Full invoice

The full reported invoice shall be included in the TDD as extended XML. See section Source document for details.

1.4. Tax Data Document Status functionality

The TDS document is a TDD used in the scenario where a (C2) declared PINT-AE compliant invoice is not validated correctly by C3 i.e. is not recognized by C3 as a PINT-AE compliant invoice. Instead of a normal TDD, C3 shall send a TDS to C5. Since the invoice has failed validation, it might not be possible to make the Data extracts from the reported invoice.

1.4.1. TDS meta data

A TDS is identified as a TDD with:

- TDT-006 Tax Data Document type code = 'F'

1.4.2. Data extracts

Key data values may be extracted from the reported invoice into the TDD i.e.

- TDG-02 REPORTED DOCUMENT is optional.

Chapter 2. Rules

The information given in a TDD shall comply to a set of rules on the content of the business terms as well as the relationship between them.

2.1. Custom content

Custom, jurisdiction specific content shall be inserted into the following element of the TDD XML:

XPath of the element that contains the custom content

```
/pxs:TaxData/pxs:ReportedTransaction/pxs:CustomContent
```

2.2. Source document

The source document (full invoice) shall be inserted into the following element of the TDD XML:

XPath of the element that contains the source document

```
/pxs:TaxData/pxs:ReportedTransaction/pxs:SourceDocument/cec:ExtensionContent
```

The source document shall not contain any attachment content. This means that the XML element `EmbeddedDocumentBinaryObject` (incl. all its attributes) shall not be part of the inserted source document.

Chapter 3. Technical details

The following section provides 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 Business process type (tdt-002) and Specification identifier (tdt-001). Business process type (tdt-002) identifies what business process a given message is part of, and Specification identifier (tdt-001) identifies the kind of message and the rules applied.

Valid document instances shall contain corresponding Business process type (tdt-002) and Specification identifier (tdt-001).



Specification identifier (tdt-001) is a string without spaces. The list below contains spaces in Specification identifier (tdt-001) to make them easier to read. **Make sure to remove any spaces before use.**

In the table below you will find the values to be used as the specification identifier (tdt-001) and the business process type (tdt-002) for this profile

Type	**Element** `cbc:CustomizationID`	**Element** `cbc:ProfileID`
Tax Data	urn:peppol:taxdata:ae-1	urn:peppol:taxreporting

UBL example of profile identifier

```
<cbc:CustomizationID>urn:peppol:taxdata:ae-1</cbc:CustomizationID>
<cbc:ProfileID>urn:peppol:taxreporting</cbc:ProfileID>
```

3.1.2. Document type identifier scheme

The `busdox-docid-qns` Document Type Identifier scheme shall be used in accordance with the current version of 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 days, as defined in ISO 8601.
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
currencyID (attribute)	Mandatory	String	AED

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

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	0847976000005
Scheme identifier	Optional	String	0088
Scheme version identifier	Optional	String	3

0088 is the ISO 6523 (ICD) code for the GLN identifier scheme.

Date

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



Dates shall not include time zone information.

Table 1. 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 shall include time zone information.

Table 2. EN 16931 Time Type

Component	Use	Primitive Type	Allowed forms
Content	Mandatory	String	13:20:00Z (UTC)
			13:20:00-05:00 (UTC-5)

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

3.3. XML schemas and namespaces

The primary XML schema used is

- OpenPeppol TDD, with the target namespace `urn:peppol:schema:taxdata:1.0`

As this XML Schema is defined by OpenPeppol itself, it is handled slightly differently from, for example, the schema for UBL Invoice. The TDD XML Schema reuses as many common components from OASIS UBL 2.1 as possible but having its own primary identity.

Used XML namespaces in the TDD XML schema are:

Prefix	URL
pxs	`urn:peppol:schema:taxdata:1.0`
cac	`urn:oasis:names:specification:ubl:schema:xsd:CommonAggregateComponents-2`
cbc	`urn:oasis:names:specification:ubl:schema:xsd:CommonBasicComponents-2`
cec	`urn:oasis:names:specification:ubl:schema:xsd:CommonExtensionComponents-2`
udt	`urn:oasis:names:specification:ubl:schema:xsd:UnqualifiedDataTypes-2`

For technical reasons, there are no deep integrations with the required XML schemas but instead it is expected that an XML Catalog is used to resolve the required XML schemas. The referenced XML schemas can be downloaded from the OASIS UBL 2.1 publication.

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 shall 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 shall be understood and carefully weighed before choosing a different course.

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

may - is truly optional.