# Table of Contents

4 1 Overview ..................................................................................................................................5  
5 2 Introduction ................................................................................................................................5  
6 3 URI.............................................................................................................................................5  
7 4 targetNamespace ..........................................................................................................................6  
8 5 Electronic document instance files ............................................................................................7  
9 6 File naming convention ..............................................................................................................8  
10 7 ENTSO-E electronic document Conformity ..............................................................................9  
11 7.1 ENTSO-E Schema additions to cater for Local variations ....................................................10  
12 7.2 Core Component Schema ......................................................................................................10  
13 7.3 Code List Schema .................................................................................................................11  
14 7.4 Local code list extensions ......................................................................................................12  
15 7.5 Local Document use overview ..............................................................................................13  
16 8 Annex 1: Modifications to the schemas defined before 2012 ....................................................14  
17 8.1 Etso-code-lists.xsd ................................................................................................................14  
18 8.2 Etso-core-cmpts.xsd ................................................................................................................14  
19 8.3 Electronic document Schemas ...............................................................................................14  
20 9 Annex 2: Latest version of entso-e schemas .............................................................................15  

## List of figures

22 Figure 1 - XML Schema extract ....................................................................................................7  
23 Figure 2 - Instance document respecting the XML Schema in Figure 1 ......................................7  
24 Figure 3 - Recommended structure of an instance document ......................................................8  
25 Figure 4 - ENTSO-E Compliant schema ......................................................................................9  
26 Figure 5 - ENTSO-E Document extensions ...............................................................................10  
27 Figure 6 - ENTSO-E Codelist structure ....................................................................................11  
28 Figure 7 - Overall picture of local code use ................................................................................13  

– Page 2 of 15 –
Copyright notice:

Copyright © ENTSO-E. All Rights Reserved.

This document and its whole translations may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, except for literal and whole translation into languages other than English and under all circumstances, the copyright notice or references to ENTSO-E may not be removed.

This document and the information contained herein is provided on an "as is" basis.

ENTSO-E DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Maintenance notice:

This document is maintained by the ENTSO-E WG EDI. Comments or remarks are to be provided at EDI.Library@entsoe.eu
## Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Release</th>
<th>Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>2011-03-16</td>
<td>First version of the document. Document approval by the Market Committee</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2012-10-11</td>
<td>New harmonized approach for defining the namespace as well as the management of extension/restriction of code list. Document approved by the Market Committee on 2012-10-23.</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>2016-04-19</td>
<td>Cancellation of the use of the &quot;restricted codelist&quot;. Thus all the codes defined in the ENTSO-E codelist or added in the &quot;local extension&quot; could be used in an exchange if this is agreed between the parties. &quot;VersionRelease&quot; referring to the version of the ENTSO-E core component has been removed as the ENTSO-E core component are freezed. This is as per decision of ENTSO-E to use standards based on CIM and published as European norm (IEC/EN norms). Added note in the overview chapter stating that ENTSO-E schemas based on ENTSO-E components are freezed.</td>
</tr>
</tbody>
</table>
1 Overview

XML namespace is the method to qualify element and attribute names used in XML Schema through their association with a unique reference that is identified by a URI (Uniform Resource Identifier) reference.

XML namespace use is defined in the W3C document “Namespaces in XML” (Third Edition) and can be located at the following URL: http://www.w3.org/TR/xml-names/

Please note that as ENTSO-E’s WG EDI work products are standardised in the IEC and reflected in the IEC 62325 series, the International Standard IEC 62325-450 “Profile and context modelling rules” has become the reference for modelling purposes, methodology wise. Thus, ENTSO-E schemas based on ENTSO-E components are freezeed.

2 Introduction

With the introduction of the possibility of extend ENTSO-E code lists in addition to the increasing use of web services and its requirement for WSDL (Web Services Description Language), ENTSO-E WG EDI agreed that time has come to introduce namespace use within the ENTSO-E XML Schema.

The principle objective of these namespaces will be to identify each electronic document in a manner that ensures its uniqueness.

An XML namespace is identified by a URI reference. All schema element and attribute names may be placed in an XML namespace.

The element name taken in isolation (for example “DocumentIdentification”) is known as a local name. However, in order to completely understand the meaning behind a local name it is necessary to know to which namespace (i.e. environment where the word is used) it belongs. The combination of both names (i.e. the local name and namespace name) is known as an expanded name.

To understand an XML element name it is necessary to prefix the namespace name to the local name (for example, “ScheduleDocument:DocumentIdentification”; where “ScheduleDocument” corresponds to the namespace name and “DocumentIdentification” corresponds to the element name). However, since namespace names can be very long and can even contain characters not allowed in names, a “prefix” is used in their place to represent them. The combination of the prefix and the local name is known as a qualified name.

The prefix, usually of three characters, appears in the place of the namespace name (for example, ess:DocumentIdentification where “ess” has been associated with the namespace “ScheduleDocument”).

The use of namespaces may also allow a TSO to tailor a specific codelist to a specific set of exchanges. Consequently the TSO has the possibility to give different meanings to the internal code “Z01” in a codelist for different information exchanges.

3 URI

A Uniform Resource Identifier (URI) provides a simple and extensible means for identifying a resource. The term "resource" is used here in a general sense for whatever might be identified by a URI. In the case of ENTSO-E it will be used to identify the different contexts in which XML

---

1 Definition: An expanded name is a pair consisting of a namespace name and a local name.

2 Definition: A qualified name is a name subject to namespace interpretation.
names are declared. This, more specifically enables the definition of a namespace for each of
the ENTSO-E processes.

It is important to note that the URI only provides identification of a resource; access to the
resource is neither guaranteed nor is it implied by the presence of a URI.

In order to provide a generic and stable means of declaring a URI for ENTSO-E it will be
composed in the following manner:

\[ \text{urn:entsoe.eu:<ENTSO-E Domain>[:<process>][:<document>][:<Version>[:<Release>]]} \]

where:

- \text{urn:entsoe.eu} shall be the stem of all ENTSO-E namespaces.
- \text{<ENTSO-E Domain>} identifies the organisation or group of organisations within ENTSO-E
  that owns the object being referenced. In the case of WG EDI this shall be “wgedi”. In the
  case of a TSO it could be “ree”, “rte”, “ceps”, “seps”, etc.
- \text{<process>} when possible identifies the specific process within the ENTSO-E Domain where
  the object is situated. In the case of WG EDI this shall be the processes for which electronic
  documents have been defined (for example, ess, ecan, errp, etc.).
- \text{<document>} optionally identifies the electronic document.
- \text{<Version>} optionally identifies the version of the document.
- \text{<Release>} optionally identifies the release of the document.

Examples of specific instances of ENTSO-E URIs:

- \text{urn:entsoe.eu:wgedi:acknowledgement:acknowledgementdocument:6:1}
- \text{urn:entsoe.eu:wgedi:errp:activationdocument:4:0}
- \text{urn:entsoe.eu:wgedi:esp:energyaccountreport:1:1}
- \text{urn:entsoe.eu:wgedi:ess:schedulingimplementationguide:4:0}
- \text{urn:entsoe.eu:wgedi:components}
- \text{urn:entsoe.eu:wgedi:codelists}
- \text{etc.}

It is to be noted that for simplicity and to avoid naming conflicts all the characters used in a
namespace shall be in lower case. A namespace name is case sensitive and this will avoid
problems due to the wrong use of a character case.

4 targetNamespace

Within an XML Schema a targetNamespace is used to specify to which namespace all the
schema constructs belong (e.g. element declarations, attribute declaration and type definitions).
This in fact enables one to define the namespace of the schema that can be used for validation.
The targetNamespace attribute can be used to place the elements and attributes of the schema
into a specific namespace. This ensures that their use and meaning is qualified by the schema
in which they belong.

The default XML namespace will be used within the ENTSO-E XML schema to ensure that all
the elements defined in the schema document are restricted to it by default. Consequently they
will not require a specific namespace prefix.

For example in the ScheduleDocument XML schema the information will be as presented as
outlined in Figure 1.
The default XML namespace may be used within the ENTSO-E XML instance documents. Elements belonging to that namespace would not require a specific namespace prefix.

The introduction of a Namespace slightly changes the beginning of the XML instance document to take the new attributes requirements into account. An example of such an instance document is shown in Figure 2.

The XML instance header information changes in two basic manners:

a) The optional introduction of the attribute "schemaLocation" to provide a hint of where the XML Schema file that can be used to validate xml elements associated with the provided namespace can be found.

b) The introduction of a default namespace definition to indicate that elements without a prefix belong to the provided namespace.

5 Electronic document instance files

In the creation of an instance document that is strictly compliant with the XML Schema it is necessary to identify the namespace being used, generally through the use of the default namespace in the heading of the instance document as indicated in Figure 2.

However, prefixes should not be used for the elements since a default ENTSO-E namespace is provided.
Prefixes could be used for local purposes not explicitly described in ENTSO-E namespace recommendation. In this case an agreement is required.

It may also be helpful to identify the location of the schema with the xsi:schemaLocation instruction as can also be seen in Figure 2.

The use of the xsi:schemaLocation generally provides the receiver of the document with a “hint” of where the schema for a specific namespace itself may be found. For example the instruction as shown in Figure 2:

\[\text{xsi:schemaLocation} = \text{urn:entsoe.eu:wgedi:ess:scheduledocument:4:2} \]
\[\text{urn:entsoe.eu-wgedi-ess-scheduledocument-4-2.xsd} \]

Implies that the schema itself may perhaps be found at:

\[\text{"urn:entsoe.eu-wgedi-ess-scheduledocument-4-2.xsd"} \]

However this is not always the case and in general an agreement is required.

In order to enable flexibility, it is recommended that the schema location instruction (and xsi definition) in the schema compliant instance should not be used.

Removing such instructions from the instance document results in an XML document that does not specifically identify a location for the XML Schema to which it is compliant. It is consequently up to the receiving party to use the appropriate schema that defines the provided namespace to validate the information instance.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="urn:entsoe.eu:wgedi:ess:scheduledocument:4:2">
<Identification xId="a"/>
<DocumentVersion v="1"/>
<DocumentType v="A94"/>
</ScheduleDocument>
```

**Figure 3 - Recommended structure of an instance document**

6 File naming convention

In order to facilitate in a consistent manner the way files containing schemas should be named it is recommended to make use of the URI that is defined for the schema in question. However since the colon (":" cannot be used in a filename it is recommended to replace each colon and dot ("." with a dash ("-").

Consequently the file name for the schema with the URI "urn:entsoe.eu:wgedi:ess:scheduledocument:4:2" would be "urn-entsoe.eu-wgedi-ess-scheduledocument-4-2.xsd".
7 ENTSO-E electronic document Conformity

The standard ENTSO-E electronic document XML schema is composed of the following standardized elements as outlined in Figure 4:

a) The “Document Schema” itself that has been automatically generated from the UML model as described in the implementation guide. The electronic document XML schema file is named “urn:entsoe-eu-wgedi-xyz-electronicdocument-n-m.xsd” where “xyz-electronicdocument” corresponds to the process and name of the electronic document and “n-m” corresponds to the version and release of the document in question.

b) The “Document Schema” imports the basic set of ENTSO-E “Core Components” in order to provide the complete set of XML datatypes that are required within the ENTSO-E environment. This Schema is a generic schema that is valid for all ENTSO-E electronic document schemas. The core component xml schema file is named “urn:entsoe-eu-wgedi-components.xsd”.

c) The “Core Components” schema itself imports the set of codelists that have been approved within ENTSO-E. This Schema is a generic schema that is valid for all ENTSO-E electronic document schemas. The code list xml schema file is named “urn:entsoe-eu-wgedi-codelists.xsd”.

The Document Schema defines an electronic document that has been clearly documented within an ENTSO-E process (ESS, ESP, ERRP, etc.).

An electronic document instance shall be considered ENTSO-E compliant if it respects the information provided by the above “Document Schema”.

Figure 4 - ENTSO-E Compliant schema

Note: the term “xyz-electronicdocument-n-m” should be replaced by the specific process and name, version and release of the document in question.
7.1 ENTSO-E Schema additions to cater for Local variations

An ENTSO-E local variation consists of an ENTSO-E provided schema with additional new valid local codes to satisfy local market needs. The specific local codes are defined in the Local extension. No other changes are permitted.

7.2 Core Component Schema

The Core Component Schema defines all the data types that are used within the ENTSO-E environment. No ENTSO-E electronic document can be generated using data types that are not referenced in the Core Component Schema. This ensures that all ENTSO-E electronic documents make use of a consistent set of information blocks.

The Core Component Schema is divided into two parts:

a) A set of unqualified data types that have no specific constraints applied to the information content other than the basic structural rules within their definition.

b) A set of qualified data types that represent the complete set of code lists defined for use within ENTSO-E electronic documents. The codelists are in general of a qualifying nature and provide additional consistent semantics to the electronic document content. These codelists are defined in the Standard ENTSO-E codelist set package.

The Core Component Schema along with the ENTSO-E standard codelist set comprise the basic building blocks necessary for the creation of all ENTSO-E electronic documents.
7.3 Code List Schema

As can be seen in Erreur ! Source du renvoi introuvable, the standard codelist set is itself divided into two basic parts:

a) The list of "standard" codelists that consists of all the codes that have been approved within WG-EDI.

b) A list of "local" codelists that are assigned by individual System Operators for internal local market use. These codes are local to a given market and are managed by the local System Operator in order to satisfy local market constraints. These codes cannot be used outside the market in which they have been designed.

In order to provide a stable core component library, ENTSO-E makes available the library structure that includes an "empty" local set of codelists. This "empty" set of codelists in fact contains by default the first "standard" code that can be found within the "standard" codelist.

This ensures that within the value space of the "local" codelist there is no possibility of validating codes that do not exist. This local codelist schema has a standardised name within the ENTSO-E environment which is "urn-entsoe-eu-local-extension-types.xsd".

Each System Operator then replaces the "empty" local set of codelists with the "local" codelists that are needed within its local marketplace. It is also the responsibility of the System Operator to ensure the distribution of the "local" codelist to the Market Participants within the local market area.
The ENTSO-E XML Schema environment “merges” the two codelist sets in order to provide at
the electronic document level the complete set of codes possible (i.e. the union of the “standard”
and “local” codes).

7.4 Local code list extensions.
The “Code List” XML Schema is composed of all the codes for each individual ENTSO-E codelist
(BusinessTypes, ProcessTypes, etc). Within the XML Schema, each of the codelists is named
with the prefix “Standard” (StandardBusinessType, StandardProcessType, etc.).

The “Code List” Schema also includes an XML Schema with an equivalent set of local codelists
that may be defined for specific local use. Within the local schema they have the prefix “Local”
(LocalBusinessType, LocalProcessType, etc.).

The standard “Code List” XML schema has in addition one other entry that provides the junction
of the standard and local codelists. The name of the codelist has no prefix (BusinessType,
ProcessType, etc.) and it merely consists of the union of the local and the standard codelists
(e.g. StandardBusinessType, LocalBusinessType).

The complete codelist is always considered to be the union of the local and the standard
codelist.

ENTSO-E shall always provide a local code list under the name “urn-entsoe-eu-local-extension-
types.xsd” that contains all the codelists as found in the standard file, with the prefix “Local”
(LocalBusinessType, LocalProcessType, etc.).

By default, every local entry has a single enumeration that corresponds to the first entry that is
found in the standard codelist.

If a local code is to be defined for the local entry, the default entry shall be replaced by the local
entry. It is local market responsibility to maintain this electronic document and to ensure that it
remains up to date.

A local extension of the code list would require the replacement of one standard entry:

```
<xsd:simpleType name="LocalAllocationModeType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:minLength value="3"/>
    <xsd:maxLength value="3"/>
    <xsd:enumeration value="A01"/>
  </xsd:restriction>
</xsd:simpleType>
```

By the local requirement:

```
<xsd:simpleType name="LocalAllocationModeType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:minLength value="3"/>
    <xsd:maxLength value="3"/>
    <xsd:enumeration value="Z01"/>
    <xsd:enumeration value="Z02"/>
  </xsd:restriction>
</xsd:simpleType>
```

Every time the ENTSO-E codelist is updated it will be necessary to replace the standard “urn-
entsoe-eu-local-extension-types.xsd” by the locally managed one to ensure that the local codes
are integrated into the extended codelist.
7.5 Local Document use overview

The diagram in Figure 7 provides the overall picture of the use of standard and local codelists within the ENTSO-E environment. Where there are XML schema files the standardised format of the filenames has been highlighted.

Note: The term "xyz-electronic-document" should be replaced by the specific process and name of the document in question.
8 Annex 1: Modifications to the schemas defined before 2012

XML Schemas developed prior to the introduction of namespaces will require the changes as outlined in the following paragraphs.

8.1 Etso-code-lists.xsd

The name of the xsd file will be changed to “urn:entsoe-eu-wgedi-codelists.xsd” to bring it into line with the file naming convention as outlined in section 6.

There are no other changes other than the modification of the specific core component namespace which is within the codelist schema.

8.2 Etso-core-cmpts.xsd

The name of the xsd file will be changed to “urn:entsoe-eu-wgedi-components.xsd” to bring it into line with the file naming convention as outlined in section 6.

There are no other changes other than the modification of the specific core component namespace.

8.3 Electronic document Schemas

The only change required is to the schema heading as follows:

```xml
<xsd:schema
    xmlns: ecc="urn:entsoe.eu:wgedi:components"
    xmlns: xsd="http://www.w3.org/2001/XMLSchema"
    xmlns="urn:entsoe.eu:wgedi:ess:schedule:4:0"
    targetNamespace="urn:entsoe.eu:wgedi:ess:schedule:4:0"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified"
    ecc:VersionRelease="19.0">
  <xsd:import namespace="urn:entsoe.eu:wgedi:components" schemaLocation="../urn-entsoe-eu-wgedi-components.xsd"/>
  <xsd:include schemaLocation="urn:entsoe-eu-wgedi-schedul edocument-4-0-restricted-codes.xsd"/>
</xsd:schema>
```

The schema heading will contain the targetNamespace and with it the URI identifying the electronic document. It also requires the introduction of a default namespace (xmlns="urn:entsoe.eu:wgedi:ess:scheduledocument:4:0") in order to ensure the connection with the elements that are specific to the XML Schema in question. In addition the core component namespace is aligned with the new namespace specification.

There are no other changes required to the electronic document schemas.

It should be noted that the introduction of a default namespace in the instance document that corresponds to the default namespace in the XML schema will also ensure that a specific namespace prefix is not required. This does not preclude its use if necessary and is agreed between the parties.
9 Annex 2: Latest version of entso-e schemas

In the latest version of the entso-e schemas the following actions have been taken place:

- Cancellation of the use of the "restricted codelist". Thus all the codes defined in the ENTSO-E codelist or added in the "local extension" could be used in an exchange if this is agreed between the parties.

- "VersionRelease" referring to the version of the ENTSO-E core component has been removed as the ENTSO-E core component are freezed. This is as per decision of ENTSO-E to use standards based on CIM and published as European norm (IEC/EN norms).

These changes are reflected in the following fragment using bold text:

```xml
<xsd:schema
    xmlns:ecc="urn:entsoe.eu:wgedi:components"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns="urn:entsoe.eu:wgedi:ess:schedul
document:4:2"
    targetNamespace="urn:entsoe.eu:wgedi:ess:schedul
document:4:2"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified"
    ecc:VersionRelease="19.0">
</xsd:schema>
```