

Basic Ground Rules For Model Documentation



Version 6.0

Document Version: 1

Table of Contents

24		
25		
26	1 Model Detail.....	3
27	2 Basic Ground Rules.....	4
28	2.1 Introduction	4
29	2.2 Time identification	4
30	2.2.1 Values for hour definition	4
31	2.2.2 Daylight saving time	4
32	2.3 Edig@s document uniqueness.....	5
33	2.3.1 Changing previously issued messages.....	5
34	2.3.2 Document uniqueness during the nomination process	5
35	2.3.2.1 Day ahead nomination.....	5
36	2.3.2.2 Within day nomination	5
37	2.4 Document header validity period scope	5
38	2.5 Units used in messages.....	5
39	2.6 Code values	5
40	2.7 Message examples.....	6
41	2.8 Debit and credit definition.....	6
42	2.9 The definition of input and output time series.....	6
43	2.9.1 Example of direction use	6
44	2.10 Flow direction convention.....	7
45	2.10.1 Use case.....	7
46	2.10.2 Direction switch	7
47	2.11 Quantity assignment.....	7
48	2.11.1 Decimal mark	8
49	2.11.2 Measure assignment	8
50	2.11.3 Zero quantity values	8
51	2.12 Internal and external account definitions.....	8
52	2.13 Rules concerning the use of a particular coding scheme for party identifications	8
53	2.14 Edig@s packages	8
54	2.15 Multiple versions of Edig@s messages.....	8
55	2.16 The use of “_”, “.” and upper and lower camelcase in XML schema tag names	9
56	2.16.1 Contextual model	10
57	2.16.2 Assembly model.....	10
58	2.17 Use of message restriction XSDs	11
59	2.17.1 Schema setup that is Edig@s compliant.....	11
60	2.17.2 Local code extension schema	11
61	2.18 Use of net values	12
62	3 Document Change Log.....	13
63	3.1 Version	13
64		

1 Model Detail

COPYRIGHT & LIABILITY

The Edig@s Workgroup (EASEE-Gas Message and Workflow Design Working Group) disclaims and excludes, and any user of the Edig@s Workgroup Implementation Guidelines acknowledges and agrees to the Edig@s Workgroup disclaimer of, any and all warranties, conditions or representations, express or implied, oral or written, with respect to the guidelines or any part thereof, including any and all implied warranties or conditions of title, non-infringement, merchantability, or fitness or suitability for any particular purpose (whether or not the Edig@s Workgroup knows, has reason to know, has been advised, or is otherwise in fact aware of any such purpose), whether alleged to arise by law, by reason of custom or usage in the trade, or by course of dealing. Each user of the guidelines also agrees that under no circumstances will the Edig@s Workgroup be liable for any special, incidental, exemplary, punitive or consequential damages arising out of any use of, or errors or omissions in, the guidelines.

2 Basic Ground Rules

2.1 Introduction

The basic ground rules outlined in this document are applicable for all the messages developed for version 6 of the *Edig@s* Message Implementation Guidelines.

2.2 Time identification

It is mandatory to use UTC as the standard time metrology in the *Edig@s* messages as recommended in the CBP 2003-002/01.

When parties involved are located in different time zones this will largely simplify the correct definition and understanding of the time indications. Additionally the annual switch to and from daylight saving time does not affect UTC.

Dates and times shall use the standardised format from ISO 8601 (i.e. a date and time shall be expressed as:

YYYY-MM-DDThh:mm:ssZ

whereas a date and time interval shall be expressed as

YYYY-MM-DDThh:mmZ/ yyyy-mm-ddThh:mmZ

In the above formulae the following terms mean:

YYYY = Year;

MM = month;

DD = day;

hh = hour;

mm = minutes;

ss = seconds.

ATTENTION:

It is mandatory that all times included in a message are provided with the same time definition.

2.2.1 Values for hour definition

The identification of an hourly period in the *Edig@s* messages is defined respecting the following rules:

- Values for hours range from **00h** to **23h**

i.e.: the last hour of a day is from **23h00** to **00h00**

The first hour of the next day is from **00h00** to **01h00**

It is thus clear that **the first hour** in a combination **is always inclusive** and **the last hour** of a combination **is always exclusive**.

- Values for minutes range from 00 to 59

E.g.: 03:00, 03:01, 03:02, ..., 03:58, 03:59, 04:00

2.2.2 Daylight saving time

The use of UTC, mandatory in *Edig@s*, makes the messages independent of any impact due to Daylight Saving Time. However as a convention used in the gas and electricity industries on the change to daylight saving time (summertime) the new day in question has only 23 hours. On contrary on the change from daylight saving time (wintertime) the day in question has 25 hours.

The first gas day therefore in the case of CEST summertime day (last weekend of March between Saturday to Sunday) is as follows:

The time interval on the day of the changeover is AAAA-MM-DDT05:00Z/ AAAA-MM-DD+1T04:00Z (i.e. 23 hours)

For the rest of the summer the time interval corresponds to:

AAAA-MM-DDT04:00Z/AAAA-MM-DD+1T04:00Z Note: the first hour is inclusive; the last hour is exclusive.

The first gas day in the case of CET Summertime day (last weekend of October between Saturday to Sunday) is as follows:

The time interval on the day of the changeover is AAAA-MM-DDT04:00Z/ AAAA-MM-DD+1T05:00Z (i.e. 25 hours)

For the rest of the winter the time interval corresponds to:

AAAA-MM-DDT05:00Z/AAAA-MM-DD+1T05:00Z Note: the first hour is inclusive; the last hour is exclusive.

2.3 Edig@s document uniqueness

Each Edig@s electronic document that is not a version of a previous transmission shall have an identification which shall be unique over time.

An Edig@s document shall be identified by the concatenation of the identification of the issuer of the message, the document identification and the document version.

2.3.1 Changing previously issued messages

A previously issued message has a unique document identification as well as a version. If a change takes place, and the revised message has to be issued, then the document identification remains unchanged and the version is increased. The new version of the document completely replaces the previous version of the document.

The first version number for a given document identification shall normally be 1.

The document version number must be incremented for each retransmission of a document that contains changes to the previous version.

The receiving system shall only accept a document with a version number which is greater than the previous version number of the same document.

2.3.2 Document uniqueness during the nomination process

The nomination process is broken down into 2 phases:

1. The transmission of a day ahead nomination for the beginning of a gas day.
2. The transmission of within day nominations for all the rest of the day respecting the market area submission rules.

2.3.2.1 Day ahead nomination

There is only one day ahead nomination for a given gas day. The nomination covers the full gas day. The nomination shall be identified by the issuer of the nomination, a document identification that is unique for the day ahead nomination and a version number. If prior to the first period for the gas day it is necessary to correct the document, it shall have the same identifiers (issuer and document) as the initial submission with the exception of the version which shall be superior to the previous version.

2.3.2.2 Within day nomination

During the course of the day within day nominations may be made. The first within day nomination may have a new document identification with an initial version but it is possible to also use the day ahead nomination document identification with a new version.

As within day nominations progress during the course of the day the same document identification shall be maintained but the version must evolve.

2.4 Document header validity period scope

The concatenation of the time intervals in an electronic document shall cover the complete validity period. Any exceptions to this rule shall be explicitly defined in the documentation of the relevant implementation guides.

2.5 Units used in messages

In line with the EASEE-Gas Recommendation “CBP Harmonisation of Units” *Edig@s* recommends the use of the following unit standards in the *Edig@s* message:

Pressure: bar

Energy: kWh (with a combustion reference temperature of 25°C)

Volume: m3 (at 0°C and 1.01325 bar) (normal m3)

Gross Calorific Value: kWh/m3 (normal m3), with a combustion reference temperature of 25°C

In case other unit references or scales are used within *Edig@s* messages, a bilateral operating agreement between the sending and receiving parties shall state the references used. No conversion parameters shall be included in the *Edig@s* message in case other references are used for pressure, energy, volume or Gross Calorific Value.

2.6 Code values

When coded entries are required, the valid code values are found in the electronic document implementation guide decision table.

The definition of the various code values is only provided in the relevant list in the *Edig@s* Code list.

Attention:

Missing code values should be reported to the *Edig@s* Workgroup who will arrange for adequate action.

See the *Edig@s* Maintenance Procedure.

2.7 Message examples

Examples, where provided, illustrate how message templates can be implemented. Those examples are fictitious and they DO NOT necessarily represent an actual operational situation. Under actual operational conditions those messages may be more complex while involving more information. Since such examples are only illustrative it MUST NEVER be used as a basis for programming or implementing this message.

2.8 Debit and credit definition

A debit refers to a quantity that decreases a balance account.

A credit refers to a quantity that increases a balance account.

2.9 The definition of input and output time series

An **Input quantity is the quantity entering a system operated by a System Operator** and an **Output quantity is the quantity exiting a system operated by a System Operator**.

This information is provided in the Attribute “Direction” which is tightly coupled to the quantity itself and is normally found in the Period class.

Consequently as long as there is no difference in the characterising information on which the Period class is dependent the direction, Z02 for input and Z03 for output may be mixed in the same Period set. For example the first hourly period could be Z02, the second hour Z03, the third hour Z03, the fourth hour Z02, etc..

This rule may vary if it is necessary to convey the direction as submitted to a System Operator. In such a case this condition shall be defined in the relevant implementation guide.

2.9.1 Example of direction use

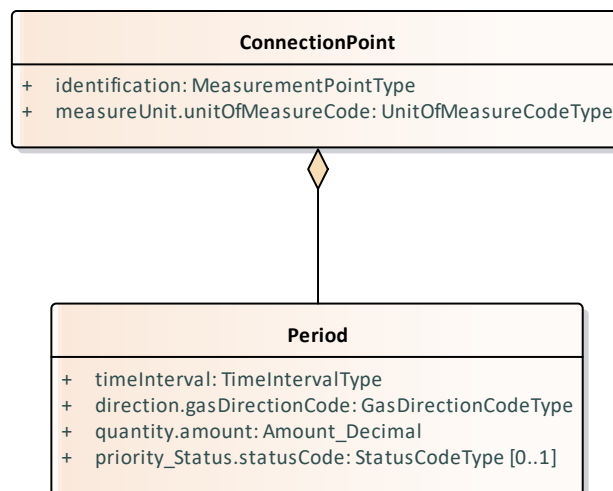


Figure: 1 **Example of direction use**

In the above example the Period class is dependent on the Connection Point class. Consequently as long as there is the same connection point, and associated attributes, the Period class may be repeated. Otherwise a new Connection Point class with different attributes will be required.

This implies that there is only one Connection Point class where its attributes remain the same.

2.10 Flow direction convention

2.10.1 Use case

In case the quantities do not directly relate to a system operated by a System Operator, for instance under Gas Sales Agreements or Service Agreements, Input quantities are quantities put into the contract and Output quantities are quantities delivered out of the contract. Quantities provided by a Seller to a Buyer are Output quantities (delivered out of the sales agreement). Quantities provided by a Shipping user to a Shipping provider are Input Quantities (into the shipping agreement).

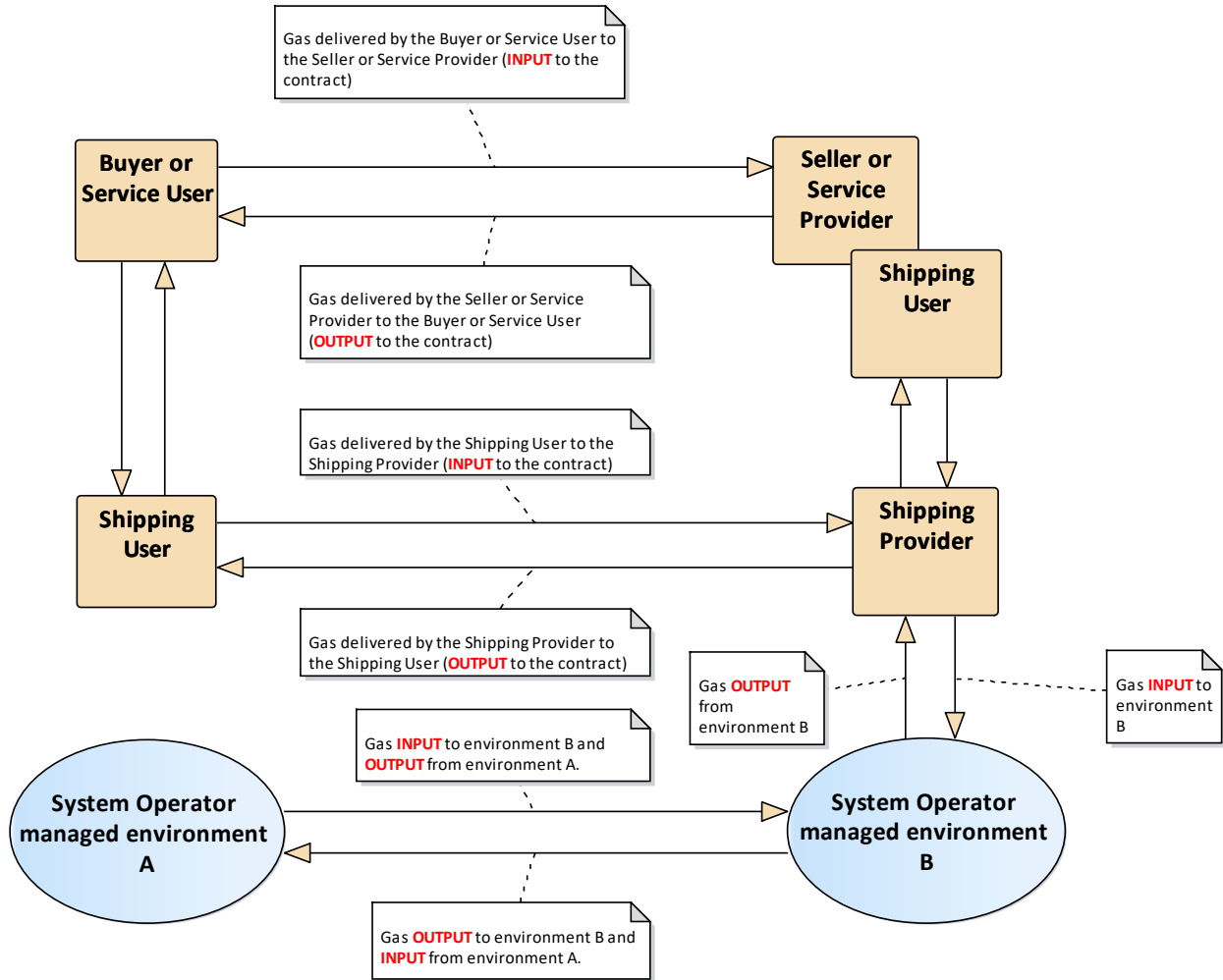


Figure: 2 Use case

2.10.2 Direction switch

A direction switch between 2 Periods (between two hours) is implemented directly where the first Period has an input direction (Z02) and a direction switch to output occurs in the second period. It is sufficient to change the direction for the second period to output (Z03). This indicates that the direction has changed from an input direction to an output direction.

2.11 Quantity assignment

All quantities assigned to the Quantity attribute shall only be expressed as a numeric value using the characters in the range 0 to 9 in addition to a single decimal sign. Only a negative decimal sign is permitted.

In addition all quantities with a unit of measure such as Kilowatts shall not contain a decimal sign since the sign is not an integral part of the quantity value and is generally used merely to express an additional concept. Negative values are permissible for quantities with a unit of measure such as temperature since the sign is an integral part of the quantity value and expresses no additional concept.

2.11.1 Decimal mark

The decimal mark is the point (“.”).

2.11.2 Measure assignment

All values assigned to the MeasureUnit attribute shall respect the same rules defined for Quantity assignment.

2.11.3 Zero quantity values

If a quantity has a value of zero then the following rules shall be followed to define the direction of the zero value:

1. If the complete time series is for an input direction then the zero quantity shall be input
2. If the complete time series is for an output direction then the zero quantity shall be output
3. If the time series is a mix of input and output directions then the zero quantity may be indifferently an input or an output direction.

In the case of debit and credit zero values the same basic rules shall apply, namely:

1. If the complete time series is for debit values then the zero quantity shall be debit
2. If the complete time series is for credit values then the zero quantity shall be credit
3. If the time series is a mix of debit and credit values then the zero quantity may be indifferently a debit or a credit value.

2.12 Internal and external account definitions

An Internal Account corresponds to an account defined by a System Operator responsible for the area covered for a Balance Responsible Party.

An External Account corresponds to an account defined by a System Operator that is not responsible for the area and is known to both System Operators.

2.13 Rules concerning the use of a particular coding scheme for party identifications

For international trade all party identifications shall respect the CBP 2020-001/01 “Energy identifier coding“. The recommended coding scheme shall in this case always be the Energy Identification Coding scheme (EIC).

For internal trade it is recommended to use the same coding scheme but local rules may dictate the use of other schemes.

2.14 Edig@s packages

An Edig@s package provides a homogeneous set of processes defined within the Edig@s scope. Each process and electronic document is identified by the targetNamespace URN which is broken down as follows:

“urn:easee-gas.eu:edigas:[Process]:[ElectronicDocument]:6:0:n” where “n” provides the version of the electronic document implementation guide. This enables an electronic document may evolve to cater for corrections or minor evolutions.

The package in the targetNamespace is identified with the indication “6.0”.

Parties opting to use the Edig@s standard should always implement the most recently published package. All parties should make provisions to be able to handle at least two versions of the standard, one being the latest published package.

Which package will be used between two parties should be agreed bilaterally in the Operational Agreement or in the Interconnection Agreement.

An Edig@s revision of a package will only occur at a minimum of two years.

2.15 Multiple versions of Edig@s messages

In order to be usable a standard needs to evolve and to adapt to changing operating environments as well as to evolving and new market requirements. This is why there are multiple versions of an *Edig@s* message. *Edig@s* shall only make changes or evolutions to the last version of a message that has been published on the *Edig@s* website.

New requirements resulting in a change of an existing message model (name, multiplicity, datatype, constraint, relation, role, etc.) will result in a new model being issued and a new release of the message structure in the current package.

There are two levels of version and release information provided in an XML schema instance: The version and release of the document set of implementation guides (i.e. Version 6 Release 0). The information is provided in the default namespace of the XML schema instance:

(i.e. urn:easee-gas.eu:edigas:[Process]:[ElectronicDocument]:6:0:n). The last character, "n", corresponds to the "document version" that can be found on the cover page of the implementation guide where the document is defined. This information may be found in every *Edig@s* schema in the targetNamespace attribute.

For example:

```
<AuctionBid_Document
```

```
xmlns="urn:easee-gas.eu:edigas:capacitytrading:auctionbiddocument:6:0:1"
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xsi:schemaLocation="urn:easee-gas.eu:edigas:capacitytrading:auctionbiddocument:6:0:1 urn:easee-gas.eu:edigas-capacitytrading-auctionbiddocument-6-0:1.xsd">
```

```
<identification>a</identification>
```

In the case of a corrigendum the following release structure shall be used:

- [document version][C][corrigendum version]
- Document version = existing document version of the process
- C = corrigendum
- Corrigendum version = latest corrigendum version.

For example if there is a first corrigendum for document version "2" the release value shall be "2C1".

2.16 The use of "_", "." and upper and lower camelcase in XML schema tag names

XML Schema tag names are constructed from the tag names provided in the document assembly model. The document assembly model is automatically generated from the document contextual model. The assembly process integrates several contextual classes into a more concise class whenever possible. The naming convention provided by the assembly process enables in a precise manner to be able to identify the referenced object in the document contextual model.

The basic assembly rules are:

A class name of a contextual model class (Aggregate Business Information Entity – ABIE) begins with an upper case letter. In a multi word name, the beginning letter of every succeeding name is uppercase.

- An attribute name of an ABIE contextual class always begins with a lower case letter. In a multi word name, the beginning letter of every succeeding name is uppercase.
- The role name of the "association end" to an ABIE contextual class always begins with an upper case letter. In a multi word name, the beginning letter of every succeeding name is uppercase.
- An underscore "_" separates a qualified name from a basic name that it is contextualising.

A class name of an assembly model class (Message Business Information Entity – MBIE) begins with an upper case letter. In a multi word name, the beginning letter of every succeeding name is uppercase.

- An attribute name of an MBIE assembly class always begins with a lower case letter. In a multi word name, the beginning letter of every succeeding name is uppercase.
- The role name of the "association end" to an MBIE assembly class always begins with an upper case letter. In a multi word name, the beginning letter of every succeeding name is uppercase.
- An underscore "_" separates a qualified name from a basic name that it is contextualising.

When at assembly level, two aggregated ABIE contextual classes have been grouped into one MBIE assembly class:

- The name of the assembly class is the name of the aggregating contextual class.
- All of the attributes of the aggregating contextual class become attributes of the assembly class.

The attributes of the aggregated class that have become attributes of the assembly class have the following naming rule:

- The name of the assembly class attribute is the name of the aggregated contextual class attribute prefixed with the contextual aggregation end role name followed by a period "." and the first letter of the role name is reverted to lowercase.

A typical example of the application of these rules can be found in every assembly document header with the tag name "issuer_MarketParticipant.marketRole.code".

For example where the ABIE contextual model indicates the following:

2.16.1 Contextual model

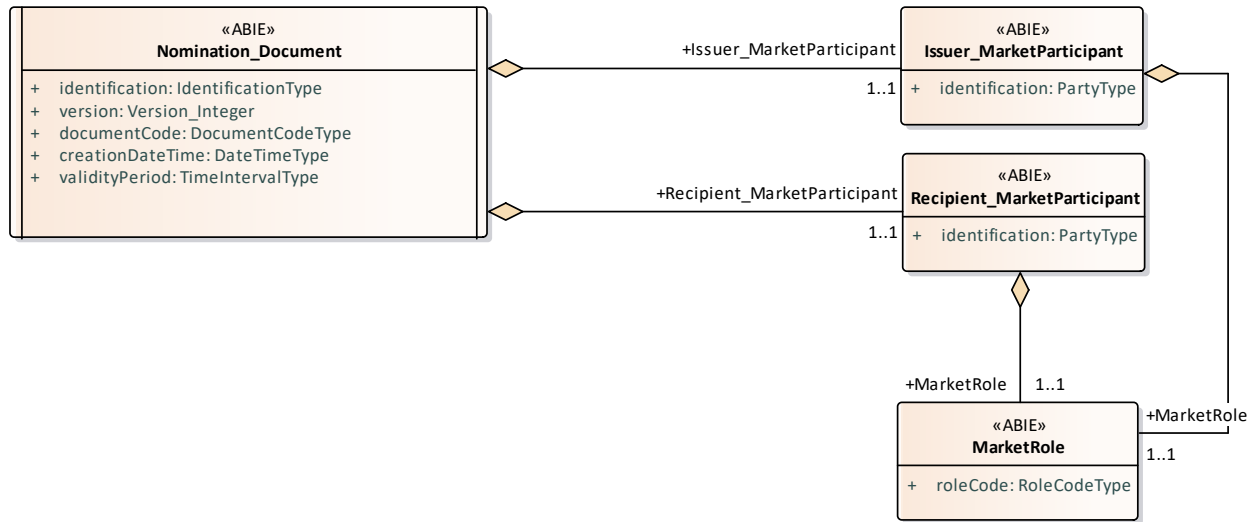


Figure: 3 Contextual model

This is transformed automatically into the following assembly model:

2.16.2 Assembly model

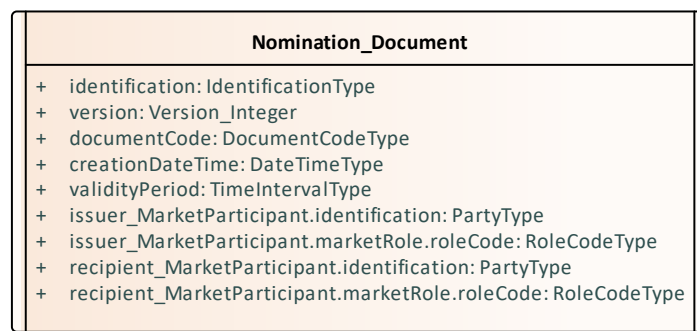


Figure: 4 Assembly model

Where the four attributes:

- issuer_MarketParticipant.identification;
- issuer_MarketParticipant.marketRole.roleCode;
- recipient_MarketParticipant.identification;
- recipient_MarketParticipant.marketRole.roleCode;

have all been aggregated from the two classes MarketParticipant and MarketRole as well as the three roles Issuer, Recipient and MarketRole as follows:

- The generic class "MarketParticipant" has been assembled into the class "MarketDocument" and becomes an attribute of MarketDocument".

- The role “Issuer” qualifies the generic class “MarketParticipant”. However, because it is now an attribute of the class “MarketDocument” the first letter is transformed to lowercase.
- The generic class “MarketRole” has been assembled into the class “MarketParticipant” and becomes an attribute of “MarketParticipant”. As an attribute of “MarketParticipant” the first letter of the class name is transformed to lowercase since it is not qualified and it is separated from “MarketParticipant” with a period.
- “roleCode” is an attribute of the class “MarketRole” and is separated from it with a period.”.

2.17 Use of message restriction XSDs

All Edig@s message XSDs include in the base schema a schema that restricts the Edig@s codelist set to allow only the codes permitted in the message in question. The Figure below shows the case of an Edig@s message that is compliant with the Edig@s standard.

2.17.1 Schema setup that is Edig@s compliant

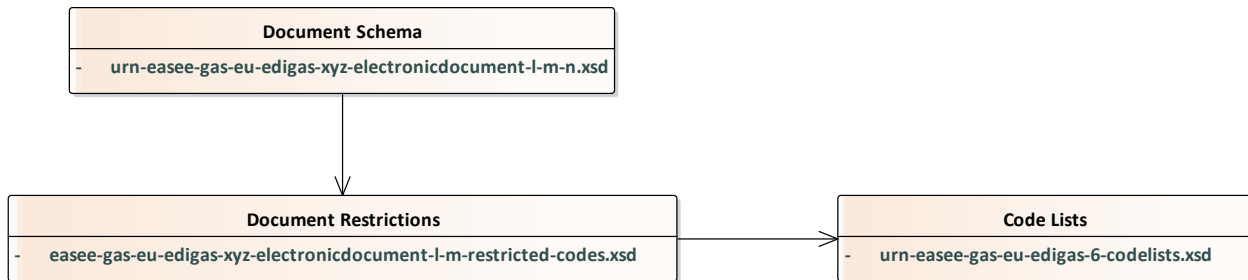


Figure: 5 **Schema setup that is Edig@s compliant**

The standard Edig@s document schema includes within it a document restriction schema that contains all the codes from the Standard Edig@s codelist that are permitted for use in the message. The Edig@s schema is named “urn-easee-gas-eu-edigas-xyz-electronicdocument-n-m.xsd” Where this corresponds to the target namespace of the document in question and where the “:”s have been replaced by a “-”.

The electronic document restriction schema for a given message is named “urn-easee-gas-eu-edigas-xyz-electronicdocument-l-m-restricted-codes.xsd” where the part “xyz-electronicdocument-l-m” is the same as that of the message schema in which it is included.

The Edig@s schema design also permits a Transmission System Operator to add for local market use additional codes over and above the permitted set of codes. In order to enable this possibility the local code schemas as outlined in figure 3 have been added.

2.17.2 Local code extension schema

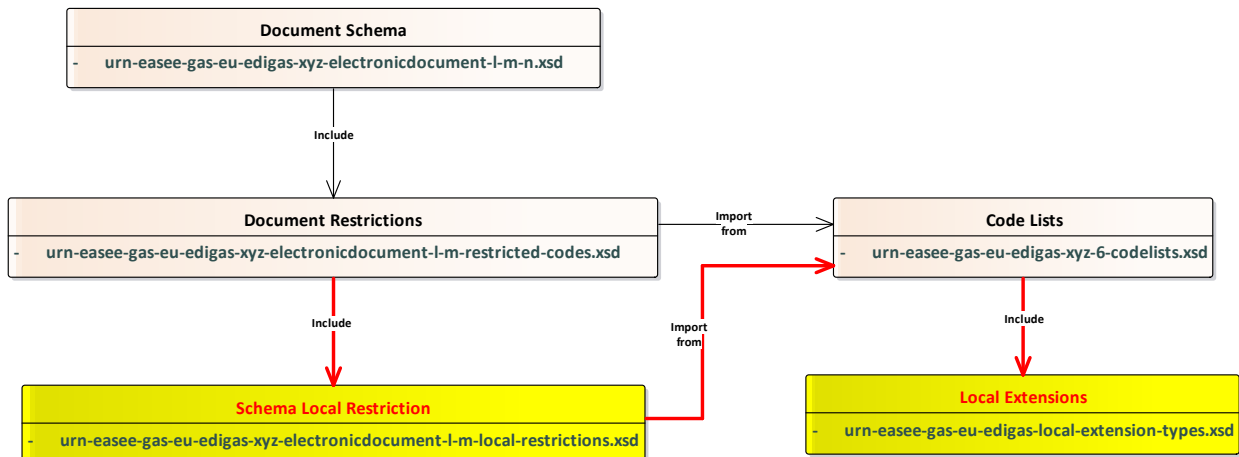


Figure: 6 **Local code extension schema**

In order to enable a Transmission System Operator to add local codes exclusively for use within the local market two codelist schema have been integrated into the standard schema set.

The first extension schema is associated with the standard set of codelists and contains all the codes that a Transmission System Operator wishes to use within the local market. This codelist “extends” in fact the standard codelist and any local codes that are to be permitted for use within the Edig@s document set must be added to this list.

The second extension schema is associated with the standard document restriction schema and enables the list of standard permitted restricted codes to be extended with the local codes. All codes used within this list must appear in the general local extension codelist. This schema is related to a specific message and has the name in the form “urn-easee-gas-eu-edigas-xyz-electronicdocument-l-m-local-restrictions.xsd” where “xyz-electronicdocument-l-m” corresponds to the message schema in question.

The release of a given set of electronic document schema will always contain these two codelist schema which by default only contain codes within the permitted codelist set. It is the responsibility of the Transmission System Operator to maintain these codelists up to date with any local variants that have been added for the local market. The addition of codes to these codelists makes the use of the Edig@s messages non-compliant with the standard.

Some local market conditions may restrict the standard restricted codelist even further. This is carried out through the modification of the “Document restrictions” XSD file. It should however be noted this makes the standard codelist restrictions non-compliant.

The use of the functions outlined in yellow and red in the diagram above make the implementation non standard since these introduce local requirements which deviate from a harmonised implementation approach.

2.18 Use of net values

In Edig@s 6 it has been decided to only permit net-values in nomination messages. This means that for a given timeseries (for the same connection point) in a Nomination & Matching message (NOMINT, NOMRES, DELORD, DELRES) only one value and one direction code should be present.

There is an exception for messages that are exchanged between systems that matches with different granularity. When a nomination message is exchanged from a daily system to a system that is matching on an hourly basis (and both input and output nominations is possible), the daily message needs to have both directions present in the message.

The Edig@s version 6 does not provide any provisions for validation of the usage of net values in the XSD, so even though it is possible to provide both directions in the messages through the Edig@s schemas, the guidelines presented here are mandatory to follow.

3 Document Change Log

3.1 Version

Version	Description
Version 1 2020-06-29	Initial release