

IS&T procedure

Import / Export Process

1 Scope

The exchanges of information between our partners are increasing specifically due to the de-concentration process. Those exchanges mainly concern the concepts of Broadcasts, Works, Rights, Rightsholders and Conflicts. They have to be managed in Import as well as in Export.

A standard way of importing/exporting those information will reduce the processing lead-time and increase the reliability while minimizing the global effort.

2 Detailed process/Work instructions

In the import process, we only add records. To simplify the treatment a full retransmission of the file is required when records are rejected at the data control step.

An import/export file is composed by a block header, a bloc content and a block footer.

2.1 Principles

- The format is XML or a tab delimited flat text file;
- The file is structured as a message containing a block header (described further in this document), a block content (made of records constituting a consistent set of data in accordance with the set of rules defined in separate documents) and a block footer (described further in this document);
- If format is flat text, each line (also called a record) is a tab delimited list of fields beginning by a record type;
- If format is XML, each line and information is delimited by XML tags described further in this document for the header and footer, and in each block content document for the block content tags depending of the exchange to be done.
- The UTF-8 **symbol set** must be used in order to provide a common way to represent special characters (characters with diacritical marks and other special characters).
The standard record separators are **CR** (Carriage Return: 13^D, 0D^H) and **LF** (Line Feed: 10^D, 0A^H);
- The UTF-8 **symbol set** must be used in order to provide a common way to represent special characters (characters with diacritical marks and other special characters).
The standard record separators are **CR** (Carriage Return: 13^D, 0D^H) and **LF** (Line Feed: 10^D, 0A^H);
- Although XML standards allow to have all information on one line, our system cannot process such data. Therefore, it is recommended to separate each sets of XML tags by the standart record separations within an XML file.
- A record is a list of character strings separated by a **Tab** character (09^D, 09^H);
- Any empty line will be ignored;
- As stated in the W3C XML standard, the ampersand character (&) and the left angle bracket (<) **MUST NOT** appear in their literal form. If they are needed elsewhere, they **MUST** be escaped using the strings " & " and " < " respectively. The right angle bracket (>) may be represented using the string " > ", and **MUST**, for compatibility, be escaped using either " > " or a character reference when it appears in the string "]> " in content, when that string is not marking the end of a CDATA section.

IS&T procedure

Import / Export Process

- Depending on the data type, the corresponding character string are formatted as follows:

Data type	Corresponding character string
Character string: xxxx	XXXX The trailing spaces are meaningless and can be omitted. The maximum length of a character field is 255 characters. All characters may be used in the fields of this type, except the tab character (09^D, 09^H) and the double quote (") .
Integer: 00003423	3423 leading zeros are meaningless and are removed
Decimal: 0024.5600	24.56 leading and trailing zeros are meaningless and are removed
Logical/Boolean True or Yes or 1 False or No or 0	1 0
Date	YYYY/MM/DD Where YYYY is the year including the century, MM is the month number with a leading zero if necessary, DD is the day number with a leading zero if necessary.
Time	HH:MM[:SS] where <ul style="list-style-type: none"> ▪ HH are the Hours expressed on a 24h/day base with a leading zero if necessary; ▪ MM are the Minutes with a leading zero if necessary; ▪ SS is optional and corresponds to the Seconds with a leading zero if necessary.
Record Type	Character string, maximum 4 characters; Even though you will see record types as numeric values, they, in fact, should be considered as an alphabetical field. It's an identifier for a row. It tells you what kind and how many fields the row contains.

IS&T procedure

Import / Export Process

2.2 Block header

The block header is always structured as follows:

Data field	Format	Description
Record Type	Text (max 4)	Mandatory fixed value = "00". Record type.
Message reference & version number	Text (max. 20)	Mandatory. Message identifier with its corresponding version number.
From (company name)	Text (max. 50)	Mandatory. Name of the company who is sending this file.
From (person name)	Text (max. 20)	Optional. Name of the person who generated the file, or the name of the contact person, if it was batch generated.
To (company name)	Text (max. 20)	Mandatory. Name of the company who will receive this file.
To (person name)	Text (max. 20)	Optional. Name of the person the files are addressed to.
Character set	Text (max. 20)	Mandatory fixed value = "utf-8". Character set identifier
Beg Date	Text (fix 10)	Mandatory. Beginning date of Message creation.
Beg Time	Text (fix 8)	Mandatory. Beginning time of Message creation.
Extensions	Text (max 50)	Optional. Comments

2.3 Block Content

The block content is specific to the transmission subject will be described in a separate document.

2.4 Block Footer

The block footer is always structured as follows:

Data field	Format	Description
Record Type	Text (max 4)	Mandatory fixed value = "ZZ". Record type.
Number of records	Integer (max 7)	Mandatory. Number of records contained in the block content (thus excluding the block header and footer).
End Date	Text (fix 10)	Mandatory. Ending date of Message creation.
End Time	Text (fix 8)	Mandatory. Ending time of Message creation.
Extensions	Text (max 100)	Optional. Comments

IS&T procedure

Import / Export Process

1 XML format

This section describes the XML tags that should be used when generating or interpreting data exchanged in XML format.

Be aware that XML element names are **case sensitive**.

2.5 Beginning of file

XML Files always start with XML version number and the encoding character set:

`<?xml version='1.0' encoding='utf-8'?>`

The root XML tag corresponding to the whole data block is: **Data**

2.6 'Header' data block

The XML tag for this data block is: **Header**

The XML tags for the data element of this data block are listed below:

#	Data element name	Corresponding XML Tag	Remark
1	Record Type	-	This data element is useless in the XML file
2	Message reference & version number	Version	
3	From (company name)	FromCompany	
4	From (person name)	FromPerson	
5	To (company name)	ToCompany	
6	To (person name)	ToPerson	
7	Character set	-	This data element is useless in the XML file because the information is already present in the "encoding" statement at the beginning of the file.
8	Beg Date	BegDate	
9	Beg Time	BegTime	
10	Extensions	Extensions	

IS&T procedure

Import / Export Process

2.7 'Footer' data block

The XML tag for this data block is: **Footer**

The XML tags for the data element of this data block are listed below:

#	Data element name	Corresponding XML Tag	Remark
1	Record Type	-	This data element is useless in the XML file
2	Number of records	RecCount	
3	End Date	EndDate	
4	End Time	EndTime	
5	Extensions	Extensions	

2.8 Example

This is an example of a footer and a header in XML format.

```
<?xml version='1.0' encoding='utf8'?>
<Data>
  <Header>
    <Version>WRI 02.02</Version>
    <FromCompany>AGICOA</FromCompany>
    <FromPerson>pco</FromPerson>
    <ToCompany>Agicoa</ToCompany>
    <ToPerson/>
    <BegDate>2002/07/24</BegDate>
    <BegTime>14:50:16</BegTime>
  <Extensions>By using WRI, declarant recognizes having read and accepted the terms and conditions of
the Mandates: http://www.agicoa.org/english/rightsholder/wri/WRI\_mandate\_v1\_8\_0.pdf, as selected
for each work hereby declared. </Extensions>  </Header>
  <work>
    Here work informations in XML format
  </work>
  <Footer>
    <RecCount>1</ RecCount >
    <EndDate>2002/07/24</EndDate>
    <EndTime>14:50:16</EndTime>
    <Extensions/>
  </Footer>
</Data>
```