



XSLT processing of XML import files

KE EMu Documentation

Document Version 1.0

KE EMu Version 4.0



Contents

Overview	1
XSLT processing	3
Pre-configured XSLT files	8

Overview

The advent of XML (eXtensible Markup Language) has provided a standards based mechanism for exchanging data between computer systems. XML, as the name implies, is extensible; that is the format in which the data is stored can be adapted to suit the data source. While this is one of the strengths of XML it also causes problems when importing data from one system into another in which the data formats do not match exactly. For example, consider this XML snippet describing a work of art in an imaginary Catalogue:

```
<table name="ecatalogue">
  <tuple>
    <atom column="TitMainTitle">An imaginary work of Art</atom>
    <atom column="CreDateCreated">1995-07-02</atom>
    <table column="CreCreatorRef_tab">
      <tuple>
        <atom column="NamLast">Citizen</atom>
        <atom column="NamFirst">John</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

You receive this data from another institution using EMu and want to import it into your system, but there is a mismatch between some of the column names in your system and those in the originating institution. For example, in your Catalogue the *Title* column may be called *SumTitle* and the *Date Created* column may be called *SumDateCreated*. Before you can load the XML into your system it is necessary to *transform* it so that it appears like:

```
<table name="ecatalogue">
  <tuple>
    <atom column="SumTitle">An imaginary work of Art</atom>
    <atom column="SumDateCreated">1995-07-02</atom>
    <table column="CreCreatorRef_tab">
      <tuple>
        <atom column="NamLast">Citizen</atom>
        <atom column="NamFirst">John</atom>
      </tuple>
    </table>
  </tuple>
</table>
```

One way to make the change is to use a text editor and replace all instances of *TitMainTitle* with *SumTitle* and *CreDateCreated* with *SumDateCreated*. If the amount of data is small or if the import is to occur only once then this solution is feasible. If, however, a number of imports will occur in which the data will be supplied in the same format, it makes sense to use XSLT (eXtensible Stylesheet Language Transforms) to apply the changes before the data is loaded. XSLT is an XML-based scripting language used to manipulate XML.

For example, the following script can be used to perform the required column renaming outlined above:

```

<?xml version="1.0" encoding="iso-8859-1"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:map="urn:map" version="1.0">
  <!-- Output in XML format -->
  <xsl:output method="xml" encoding="utf-8"/>

  <!-- Mapping table of old names to new names -->
  <map:entries>
    <map:entry oldname="TitMainTitle" newname="SumTitle"/>
    <map:entry oldname="CreDateCreated" newname="SumDateCreated"/>
  </map:entries>
  <xsl:variable name="map" select="document('')/*/*map:entries/*"/>
  <!-- For every node we copy it over. Note that attributes
  are handled by the next template. -->
  <xsl:template match="*">
    <xsl:copy>
      <xsl:apply-templates select="@*|node()"/>
    </xsl:copy>
  </xsl:template>

  <!-- Special handling of attributes. -->
  <xsl:template match="@*">
    <xsl:variable name="entry" select="$map[@oldname =
current()]" />
    <xsl:choose>
      <xsl:when test="name() = 'column' and $entry">
        <xsl:attribute name="column">
          <xsl:value-of select="$entry/@newname" />
        </xsl:attribute>
      </xsl:when>
      <xsl:otherwise>
        <xsl:copy/>
      </xsl:otherwise>
    </xsl:choose>
  </xsl:template>
</xsl:stylesheet>

```

To execute the XSLT script an XSL engine is required. A number of products provide XSL engines that can be used to transform the XML for loading into EMu. One such product is Cooktop (<http://www.xmlcooktop.com>). When a file is received from an institution, it is only necessary to perform the transformation before importing the XML into EMu.

EMu 4.0.01 has streamlined the above process by adding XSLT processing as part of the Import tool for XML files: it is now possible to import an XML file and have it transformed as part of the Import process. The XSLT file used to transform the XML can be stored on your local machine (*local file*) or on the EMu server (*pre-configured file*). Files stored on the EMu server are available to all users. In general, the pre-configured files are "standard" transformations used to manipulate data from known sources. A known source can be:

- a standard format (e.g. Darwin Core or Dublin Core)
- a repeatable format (e.g. EMu export format, BRAHMS)

Using *repeatable formats* it is possible to define XSLT files that allow for easy import of data from other EMu clients for customised modules such as the Catalogue, Taxonomy and Collection Events.

XSLT processing

The EMu Import Wizard has been extended to provide XSLT processing for XML-based import files. The extensions are only available for files with a .xml file suffix. If you have XML files with a .txt suffix, it will be necessary to rename them in order to use the XSLT processor.

To access the XSLT processor, in the module in which records are to be imported:

1. Select **Tools>Import** from the Menu bar.

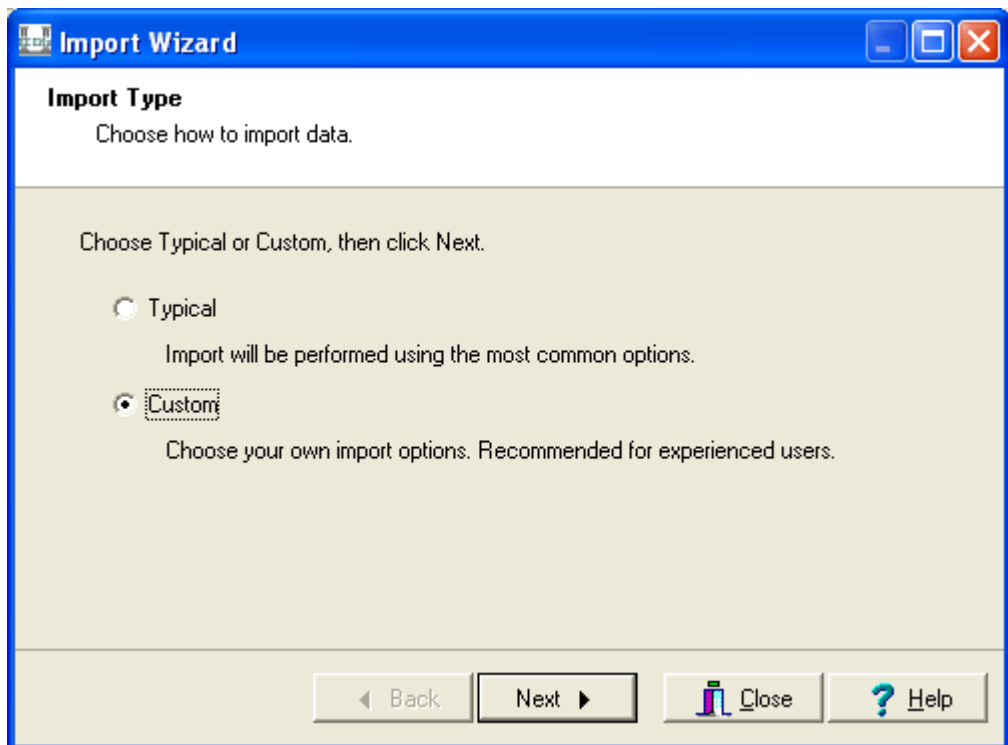


If the Import option is greyed out, you do not have the necessary permissions to use it.

The Select File To Import box displays.

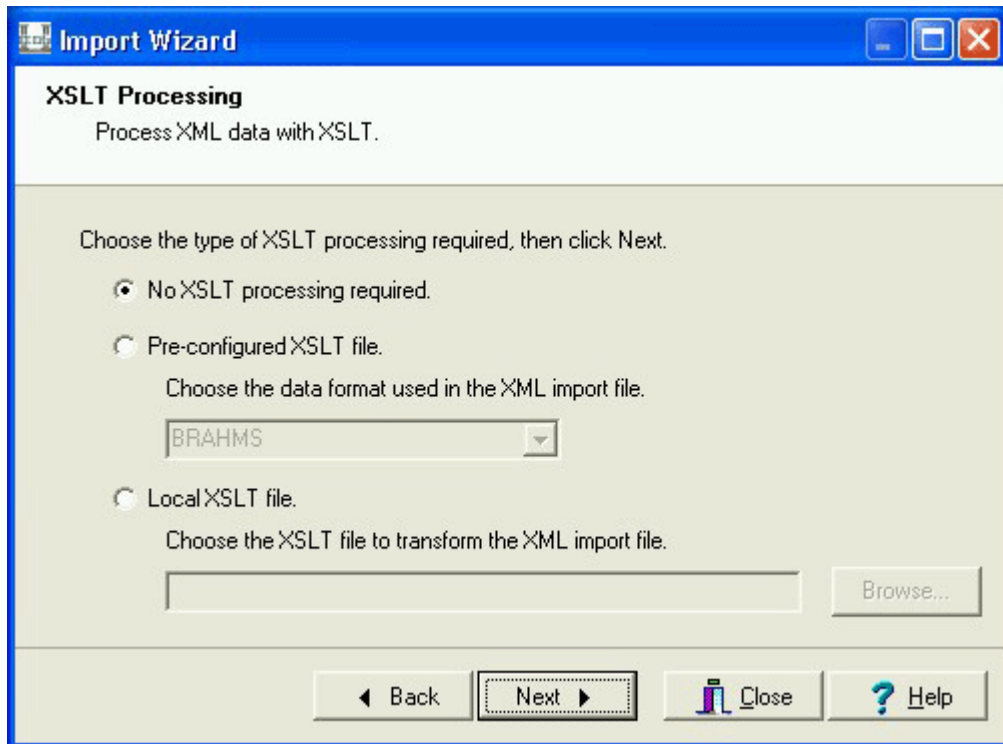
2. Navigate to the file that contains the data to be imported, select it and select **Open**.

The Import Wizard displays:



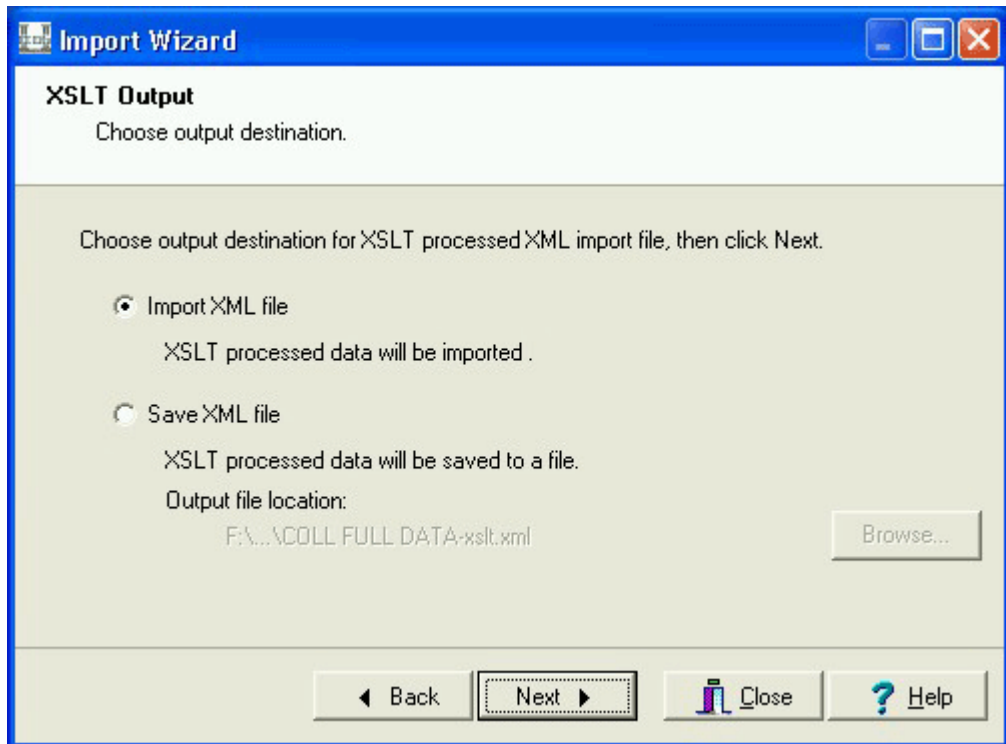
3. Select **Custom** from the Import Type screen and click **Next**.

The XSLT Processing screen displays:



Three options are available:

- **No XSLT processing required**
The XSLT processor is not invoked and the import data file is passed to the Import tool for loading.
 - **Pre-configured XSLT file**
A drop-list is populated with all the server side XSLT files. These files contain "standard" XSLT scripts used to transform known XML formats. Selecting this option and one of the pre-configured entries will result in the XSLT file being copied from the server to your local machine and executed by the XSLT processor.
 - **Local XSLT file**
If you want to use an XSLT file that resides on your local machine, choose this option and browse to the file.
4. To use the XSLT processor choose the second or third option and select **Next**.
The XSLT Output screen will display:



Two options are available:

- **Import XML file**

The output of the XSLT processor (the transformed XML) is fed into the Import facility for loading. The transformed XML is saved in a temporary file used by the Import tool. All error messages relating to the import refer to this temporary file. The name of the temporary file can be determined by using the **Verbose** option on the Logging screen. The temporary file is not removed until the **Finished** button is clicked on the Importing screen.

- **Save XML file**

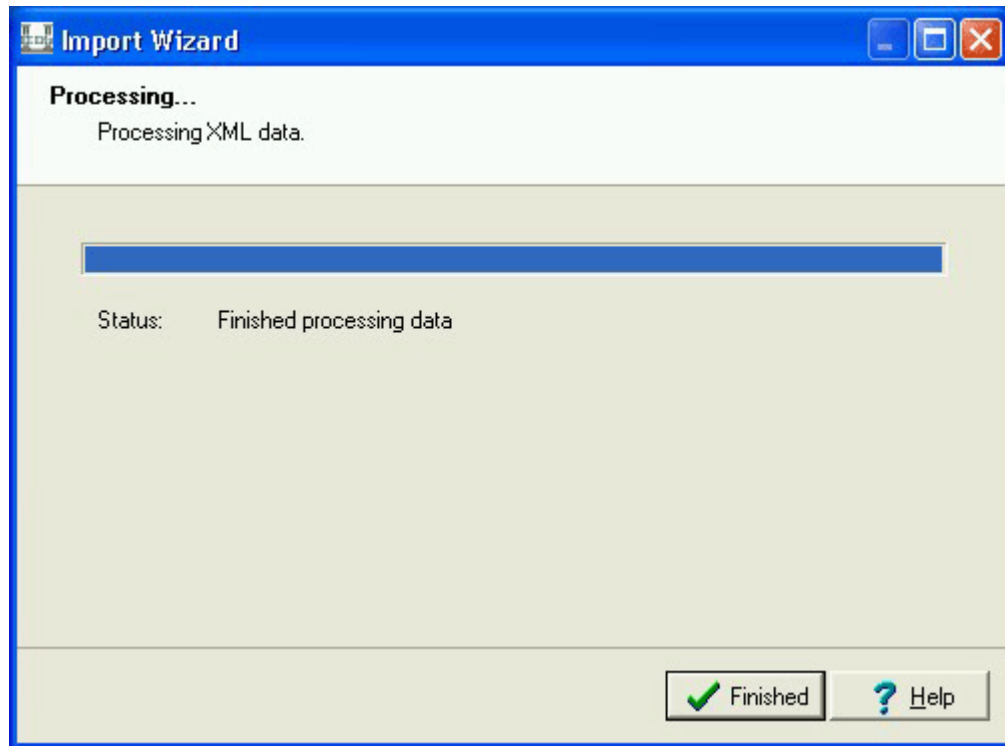
If you only want to run the XSLT processor and view the output of the transformation, use this option to select the name of the file into which the generated XML will be saved. The data import phase will not be run.

If **Save XML file** is selected, the level of logging can be set and the XSLT processing invoked; if the **Import XML file** option is selected, the normal Import sequence is followed (see the EMu Help for details).

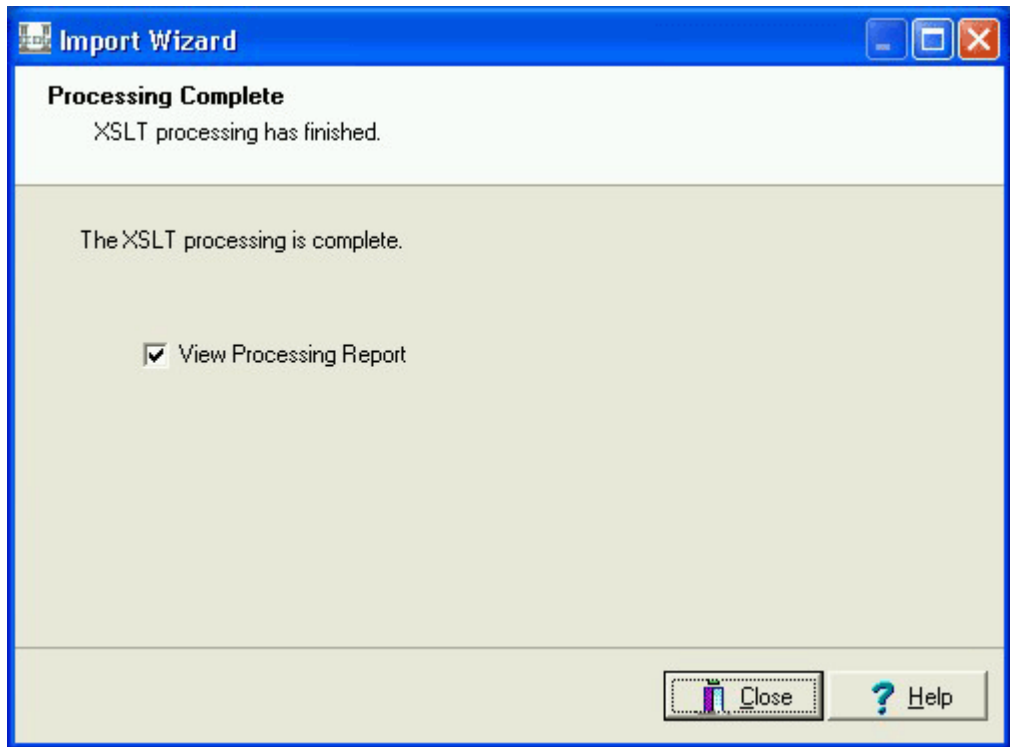
The table below indicates when the XSLT processor is invoked and whether the Import phase is executed:

Option	XSLT	Import
No XSLT processing required	✗	✓
Pre-configured XSLT File/Import XML file	✓	✓
Pre-configured XSLT File/Save XML file	✓	✗
Local XSLT File/Import XML file	✓	✓
Local XSLT File/Save XML file	✓	✗

When the XSLT processor is run a screen showing the status of the processing is displayed. Once the transformations are complete the Import phase will begin automatically for options that require the data to be imported (see the EMu Help for details). If the data is not imported (e.g. saving XML to a file), the processing screen will indicate that the transformations are complete:



- When the **Finished** button is clicked the final screen displays allowing the generated report to be viewed:



The EMu XSLT processor uses the Microsoft XML libraries (MSXML). In order to use the XSLT processor it is necessary to have MSXML 3.0 or later installed (Windows 2000 SP4 or Internet Explorer 6 or later, Windows XP, Windows Vista, Windows Server 2003).

Pre-configured XSLT files

As described above (page 2) it is possible to have pre-configured XSLT files stored on the EMu server. These files are accessible to all users and are listed in the drop-list below the **Pre-configured XSLT file** option. The files are stored in a per table directory in one of two locations:

- `etc/import/table`
Location of client independent XSLT scripts. These script typically load into the core EMu modules that do not vary from client to client (e.g. Parties, Loans, etc.). Clients should not add scripts to this location as these scripts are added by KE Software.
- `local/etc/import/table`
Location of client specific XSLT scripts. Any scripts that transform data for institution specific modules (e.g. Catalogue, Taxonomy) should be kept in this location. All client scripts should be added to this location.

When installing a script on the EMu server the `local/etc/import/table` directory may not exist, in which case it will be necessary to create it. For example, if you have a script called "BRAHMS.xslt" that transforms Brahms XML for loading into your EMu Catalogue module, you would store it under:

```
local/etc/import/ecatalogue/BRAHMS.xslt
```

The entry that appears in the drop-list in the Import wizard is the name of the file without its file suffix (e.g. BRAHMS for BRAHMS.xslt). The file name may contain spaces. XSLT scripts do not need to have an .xslt suffix, however this is the extension usually used.