

IMPORT MANAGER USER GUIDE

The logo for StiboSystems, featuring the company name in a white sans-serif font with a small crown-like icon above the 'i' in 'Stibo'. It is positioned on a large orange triangle that points to the right.

StiboSystems

STEP Trailblazer

Table of Contents

Table of Contents	2
Importing Data and Images	6
Import Data Wizard	6
Excel Files	6
CSV Files	7
Data Import Manager Wizard	8
Step 1 - Select Configuration	9
Step 2 - Select Data Source	11
Using FTP	11
Using ODBC	11
Using File	12
Step 3 - Select Format	13
BMEcat Format	14
CSV Format	16
ETIM Format	18
ETIM Versions	18
ETIM Classification	18
ETIM Feature Types	20
Valid Values	21
ETIM 5	21
ETIM 6	21
Valid Units	21
ETIM 5	23
ETIM 6	24
Excel Format	25

Excel Smartsheet Format	26
FixedWidth Format	27
Generic XML Format	28
Processing Instructions	29
Record Instruction	30
Source Instruction	31
Preventing Duplicate Field Names	31
MultiSource Instruction	32
SourceID Instruction	33
Repeated Instruction	35
DimensionPointID Instruction	37
Generic XML Import Simple Example	39
Input Document	39
Template	40
Conversion Preview	41
Generic XML Import Advanced Example	42
Input Document	42
Creating a Template	43
Remove Redundant Elements	43
Add Record and Source Instructions	44
Add Repeated Instruction	45
Add Repeated Instruction With a Filter	46
Add MultiSource Instruction	47
Conversion Preview	48
STEPXML Format	49
Schema or DTD Validation	49
Auto Approval	50
SingleUpdateMode	50

Deleting References	50
STEPXML Import Overview	51
Create Objects	51
Update and Delete Objects	51
Deleting Values and References / Links	51
STEP-ProductInformation Element Attributes	52
Available STEP-ProductInformation Replace Instructions	52
Available Replacement Rules Elements	53
Limitations	53
Deleting Objects	53
UNSPSC Format	55
Step 4 - Map Data	56
About CSV and Excel Files	56
Mapping Data Manually	58
IDs, Names, and Parent IDs	58
Changing Object Types	58
Mapping a Column Multiple Times	58
Delete Values	58
Dimension Dependent Data	59
Mapping Based on Data Type	59
Mapping the ID Column	60
Mapping Object ID	60
Mapping Keys	61
Mapping the Object Name Column	63
Mapping the Parent Column	65
Mapping an Attribute Column	67
Mapping a Constant Value	69

Removing a Mapped Column	70
Additional Column Mapping Options	71
Product Classification Link	72
Product Reference	72
Asset Reference	72
Entity Reference	72
Classification Reference	73
Reference Meta-Data	73
Auto Mapping Data	76
Predefined Header Text	76
Using Auto Map	76
Transformations	78
Transformation Examples	78
Available Transformations	78
Extract and Split Data Example	82
Extract and Split Identifiable Data Example	87
Extract, Split, and Convert Cryptic Data Example	91
Concatenate Column with Unit Data Example	95
Dimension Point for Single Dimension Data Example	99
Step 5 - Identify Objects	101
Importing With Object IDs	101
Importing Without Object IDs	102
Step 6 - Identify Destination	105
Step 7 - Select Business Rules	107
Adding Conditions and Actions	107
Removing Conditions and Actions	107

Business Rule Limitations	108	Illegal Unit for Attribute	137
Step 8 - Advanced Settings	109	Unknown Parent Product	137
General Advanced Settings	109	Maximum Length, Illegal Value	138
Remove Un-Mapped Values Setting	110	Out of Range, Illegal Value	139
Remove Un-Mapped References Settings ..	111	Not a Number, Illegal Value	139
Starting the Data Import Process	113	Object Existed with Another Object Type	140
Importing Assets	115	Unknown Classification	141
Audience	115	Not Privileged to Create Object	141
Importing Assets: A Checklist	115	Unknown Asset ID	141
Importing Assets: Special Settings	116	Array Operation Failed	142
Starting an Asset Import	117	Data Quality in the Import Manager	143
Select the Target Location	117	Discovering Data Import Quality Issues	143
Step 1 - Select Import Location	120	To Import a Data Profile	143
Step 2 - Select Asset Type	121	Overview	144
Step 3 - Select Context	122	Details	145
Step 4 - Link To Product	123	On the Fly Data Validation	146
Methods for Creating Reference Links	124	Using Transformations to Improve Data	
Step 5 - Overwrite Existing Assets	125	Quality	147
Step 6 - Import Overview	127	Concatenating Data Columns	147
Starting the Asset Import Process	128	To Merge Data into Single-Valued	
Importing an eCI@ss Hierarchy	129	STEP Data	147
Import an eCI@ss Hierarchy	129	To Merge Data into Multi-Valued Data ..	148
Map eCI@ss Attributes to STEP Attributes ..	130	To Concatenate Data Using Variables ..	151
Checking Data Imports	132	Transformation Lookup Tables	153
Monitor the Import Process	132	To Create Transformation Lookup Table	
Execution Report	132	Objects	153
Types of Error Messages	134	Using Transformation Lookup Tables	
Error Message Examples	137	When Importing Data	155
		Transform Using Attribute Transformation	156

The If Equals, Source Transformation 157
Handling Import Errors 157

Importing Data and Images

Data can be imported into STEP via the Import Manager, as described below, or with Integration Endpoints. (For more information, see the **Integration Endpoints for Data Exchange** documentation.)

The STEP Import Manager consists of two components:

- STEP Import Manager, a component for importing product, classification or entity data.
- STEP Import Assets, a component to load assets such as images.

Import Manager user should have the following skills:

- An understanding of the basic concepts of STEP, of maintaining attributes and their values, and of creating and maintaining products, classifications or entities.
- Knowledge of the STEP system equivalent to basic STEP training
- A good working knowledge of the character sets UTF-8, Windows-1252, and ISO 8859-1 (aka Latin-1).
- Working knowledge of STEPXML file format.
- Familiarity with the creation of attributes

Import Data Wizard

There are two Import Manager wizards: one for object data and one for assets, such as images and documents.

Before starting the import process, you should:

- Verify that the data in the columns being imported has been entered in a consistent manner.
- Verify that all attributes and reference / link types being imported already exist in STEP.

When working with Excel or CSV files, the STEP Import Manager lets you create and update objects of the types product, classification, entity, asset (objects, not content), and attributes (objects, definitions).

Only one node type / super type (products, entities, etc.) can be imported at a time. If the import file contains entities, products, and classifications, you must import the file once for each object type. This procedure is often used to build a new object (product) hierarchy.

Excel Files

- Formulas are not allowed in the XLS or XLSX file. Data columns that include calculations or formulas must be loaded as fixed values.
- When loading data that include special characters (such as trademark TM, registered trademark [®], or mathematical symbols), ensure that the load file conforms to the Unicode character set.
- Ensure that data originally entered as a fraction (e.g., 3/16) was not converted to a date by Excel.
- All Excel cells must be set to 'Text' format.

- Remove any formatting applied to numeric values in Excel. This allows the Excel value to be entered into STEP, not the value that Excel transforms it into for display purposes.
- To remove an attribute value, use the entry **[delete]** in your load file. Empty cells do not cause values to be deleted. If the value being deleted was inherited, the result is not a blank field, but the inherited value is restored.
- Enter attribute values and units in the same cell, with the units following the values. STEP separates the values from the units and validates them individually. If your attribute values and units are in separate columns, you can either use the concatenation method in Excel or use a database program to combine them.

CSV Files

- Use a character set that is supported by the Import Manager for CSV files: UTF-8, UTF-16, Windows-1252, or ISO 8859-1 (Latin-1).
- When the file does not use one of STEP's delimiter characters, you must convert the field delimiter character in your file to one that is accepted.

Important: When loading CSV files, a common setup is to use the tab character as delimiter. Other delimiter characters may conflict with attribute values that contain that delimiter character.

Data Import Manager Wizard

The Import Manager wizard involves a number of steps, but many data imports do not require all steps.

- [Step 1 - Select Configuration](#)—Specify if you want to use an existing import configuration or create a new one. If you frequently load data files manually, and they have the same data in the same columns, you can save a configuration with all necessary settings.
- [Step 2 - Select Data Source](#)—Select the source of the data. If the source is a file, open a browser and select the file that you want to use. You can also choose FTP or an ODBC connection.
- [Step 3 - Select Format](#)—Select the type of file to import and the basic parameters of the file.
- [Step 4 - Map Data](#)—Map the columns of data in the import file to the equivalent data in STEP.
- [Step 5 - Identify Objects](#)—Confirm if products in the input file are new or existing and specify to locate products by attribute value (when the file does not contain STEP IDs), by the product ID, or by name.
- [Step 6 - Identify Destination](#)—Specify a default object type and parent for new objects.
- [Step 7 - Select Business Rules](#)—Select business conditions and business actions to test / execute in relation to the data being imported. Data to be imported is processed based on the selected business rules.
- [Step 8 - Advanced Settings](#)—Set infrequently used operations, such as removing attribute values that are not listed in the input file or setting the approval status of imported objects.

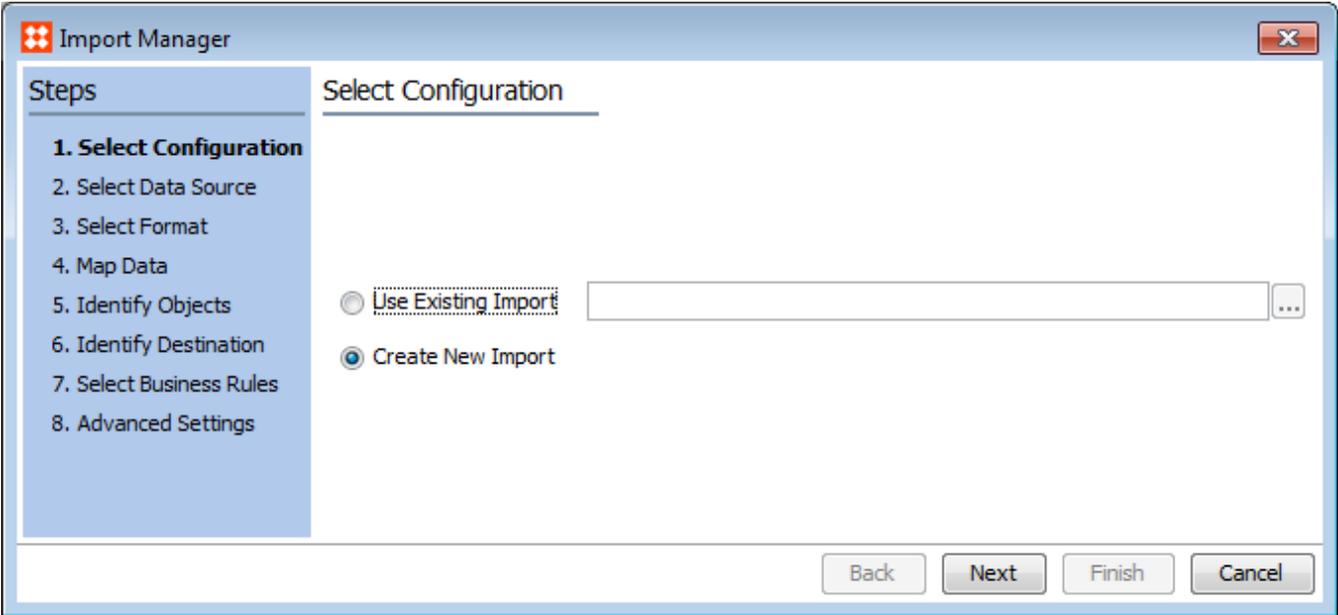
Step 1 - Select Configuration

First, open the Import Manager wizard.

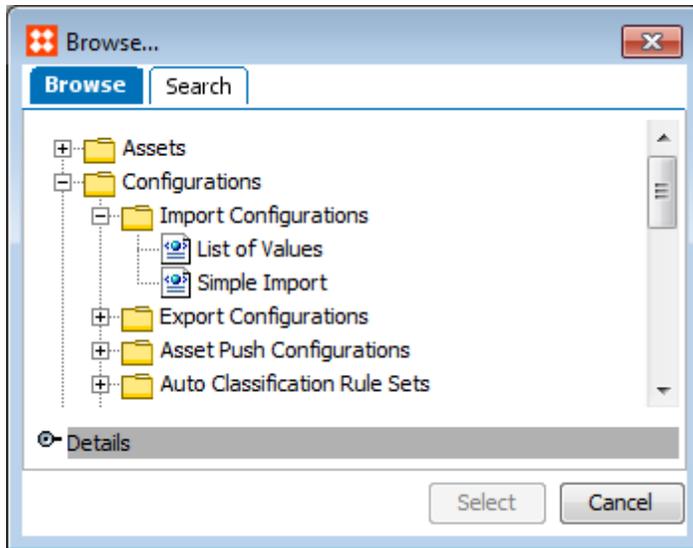
- From the **File** menu point to **Import**, and then choose **Data**. The **Import Manger** wizard opens.

In the first step, specify if you want to use an existing import configuration or create a new import configuration. If you create a new configuration, you can save it for use with other import files. A configuration file is stored as an asset in STEP and contains an XML representation of the options in Import Manager.

1. To create a new configuration, Select **Create New Import**.



To use an existing import configuration, select **Use Existing Import**, and then use the Browse or Search tab to locate the configuration in STEP.



2. Click **Next** to move on to the next step, or if you are using a configuration with no changes, click **Finish**.

Step 2 - Select Data Source

If you selected an import configuration in Step 1, this step is filled in with the saved data. If you are creating a new import configuration, you must select the source from the following options:

- FTP allows you to select a file from an FTP site.
- ODBC allows you to connect to an external database.
- File allows you to select a file from your local hard drive or from your network.

Using FTP

1. Select **FTP** from the Data Source list.

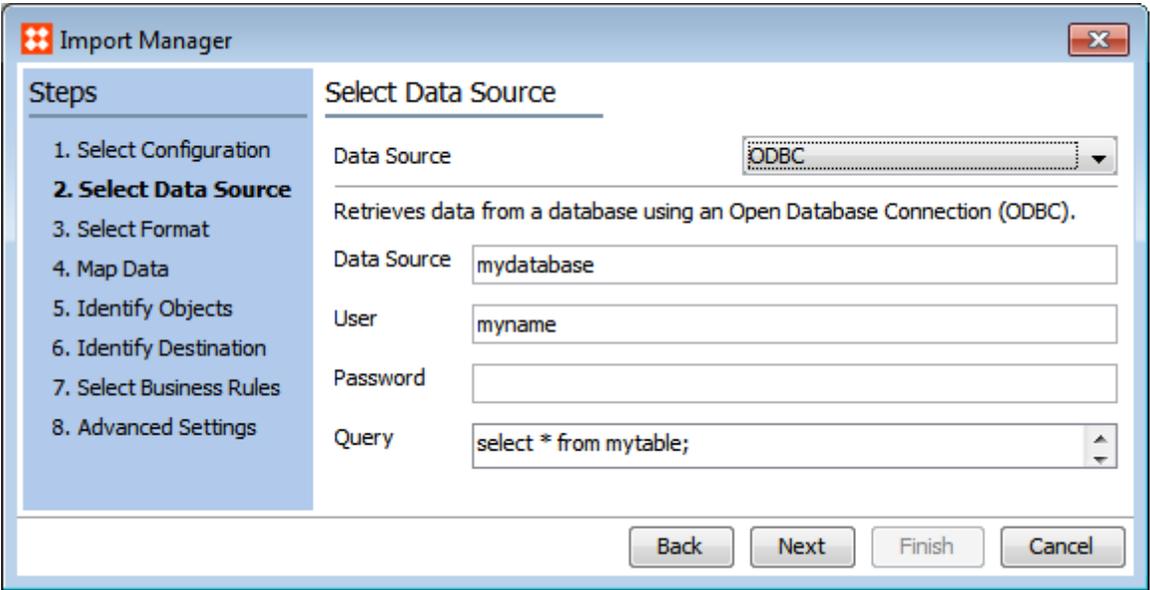
The screenshot shows the 'Import Manager' dialog box. On the left, a 'Steps' list includes: 1. Select Configuration, 2. Select Data Source (highlighted), 3. Select Format, 4. Map Data, 5. Identify Objects, 6. Identify Destination, 7. Select Business Rules, and 8. Advanced Settings. The main area is titled 'Select Data Source' and features a 'Data Source' dropdown menu currently set to 'FTP'. Below the dropdown, a descriptive text reads: 'Retrieves data from a remote server using the File Transfer Protocol (FTP)'. There are four input fields: 'Host' containing 'myserver.mycompany.com', 'User' containing 'myname', 'Password' which is empty, and 'Filename' containing '/myfile.txt'. At the bottom of the dialog, there are four buttons: 'Back', 'Next', 'Finish', and 'Cancel'.

2. Enter the **Host** name, the **User** name, and the **Password** (if any).
3. Enter the **Filename** including the path, and then click **Next**.

Using ODBC

When importing via the ODBC connection, STEP converts the data into XML automatically, and loads data into STEP without any modifications.

1. Select **ODBC** from the Data Source list.

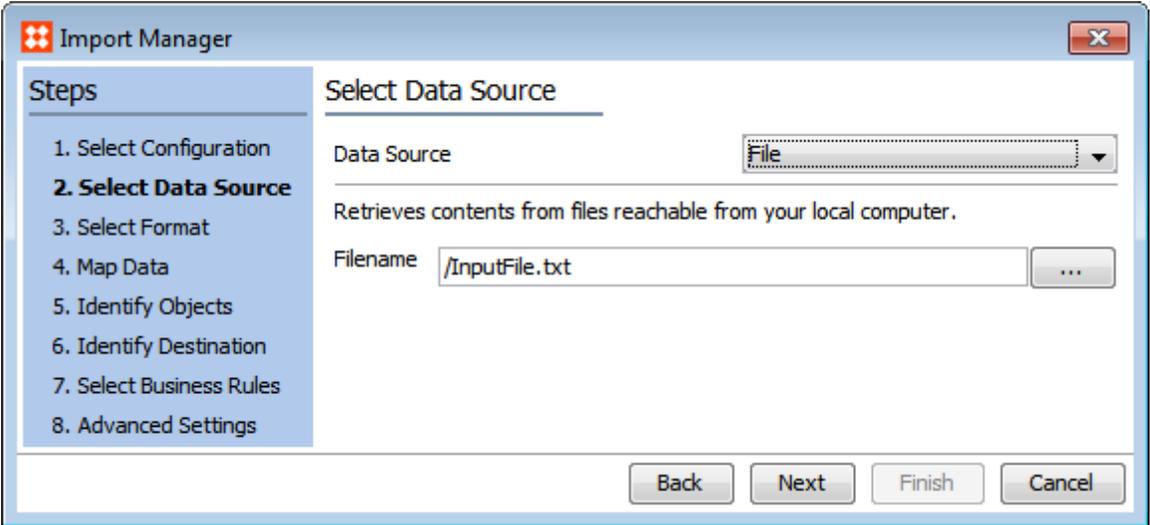


- 2. In the **Data Source** field, specify the target database.
- 3. Enter the database **User** ID and **Password** (if required).
- 4. In the **Query** field, write a SQL query that specifies which rows you want to import, and then click **Next**.

Using File

It is common practice to copy very large files to your local drive prior to importing.

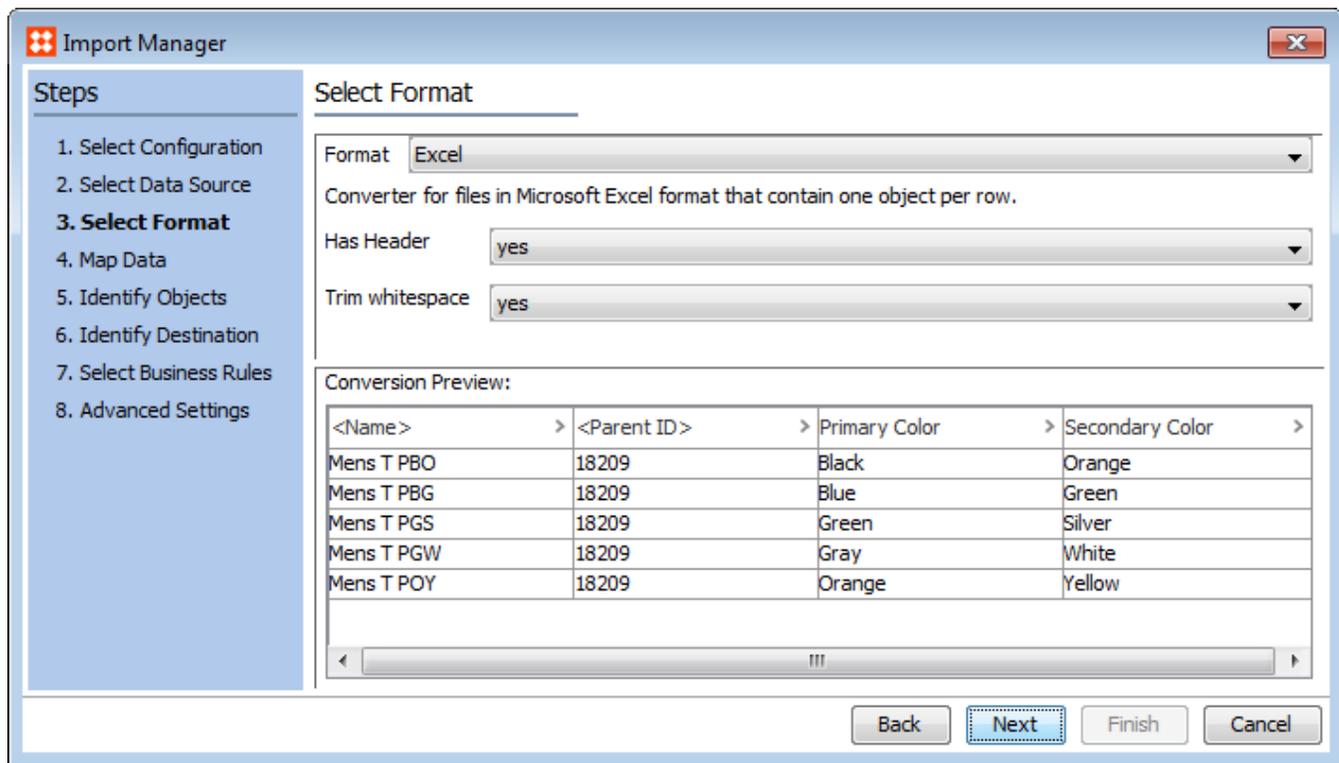
- 1. Select **File** from the Data Source list and click the ellipsis button (...) on the Filename field.



- 2. Locate the relevant file, and then click **Select File**, and then click **Next**.

Step 3 - Select Format

Select the format of the file manually if it is not automatically detected by the Import Manager. Depending on the format selected, a number of additional settings becomes available. Before proceeding, the Conversion Preview allows for visual inspection of how the settings affect the data.



The following formats are available:

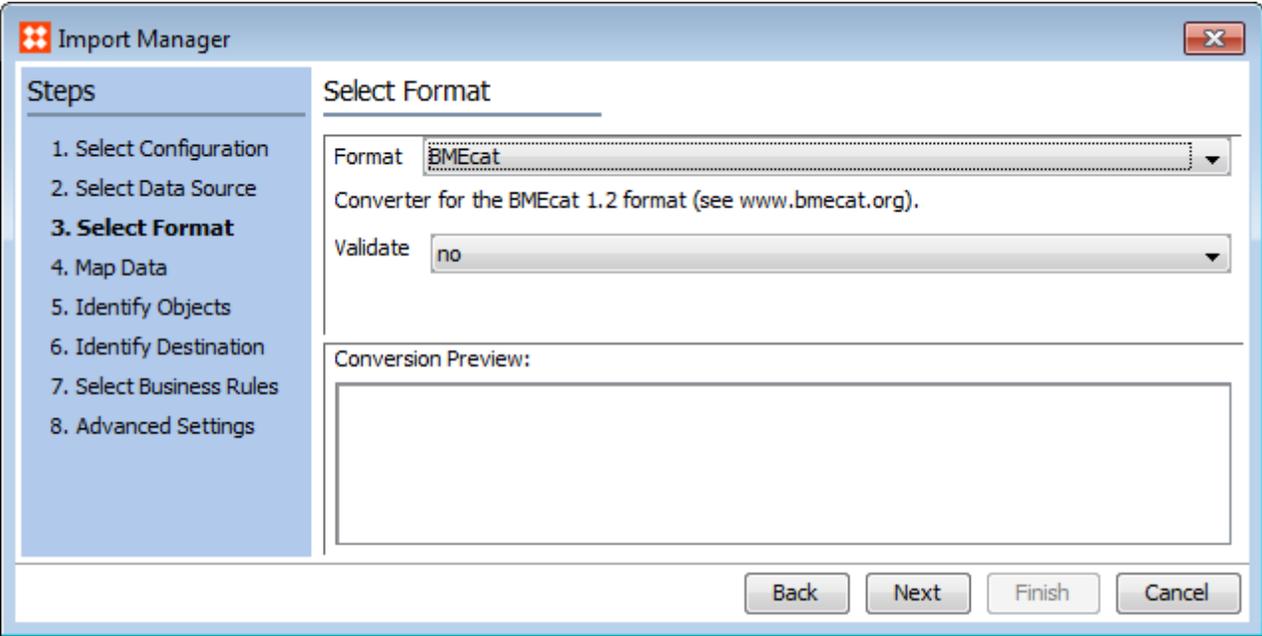
- [BMEcat Format](#)
- [CSV Format](#)
- [ETIM Format](#)
- [Excel Format](#)
- [Excel Smartsheet Format](#)
- [FixedWidth Format](#)
- [Generic XML Format](#)
- [STEPXML Format](#)
- [UNSPSC Format](#)

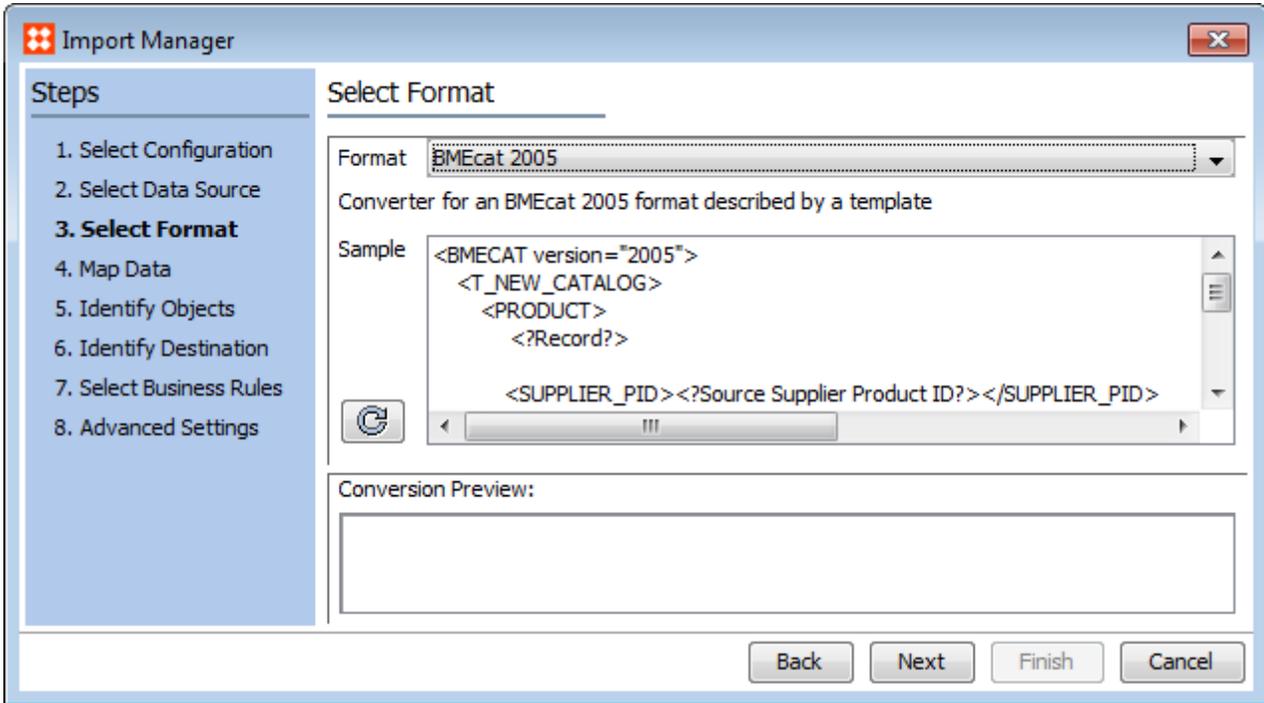
BMEcat Format

BMEcat and the 2005 version use an XML-based standard for electronic data transfer by electronic catalogs. For more information about the standard, see www.bmecat.org.

- 1. If the Import Manager did not detect the file format automatically, make that selection now.
- 2. In **Validate**, select whether or not you want the system check the validity of the XML file before loading any data, and then click **Next**.

Note: Conversion Preview information is not available for BMEcat format.





CSV Format

The importing process is very similar for CSV and Excel files. You can import new products, classifications, entities, attribute values and references. However, you cannot create new attributes or other system setup entries such as LOVs, users, reference types, and so on.

Note: Importing a CSV or Excel file that contains a mix of entities, classifications, and products requires several separate steps. Classification data must be imported first, then product data, and finally the entity data.

A typical CSV data import file could look like the following image, where the first row is a header and the data is delimited by a comma:

```
<Name>,<Parent ID>,Primary Color,Secondary Color
Mens T PBO,18209,Black,Orange
Mens T PBG,18209,Blue,Green
Mens T PGS,18209,Green,Silver
Mens T PGW,18209,Gray,white
Mens T POY,18209,Orange,Yellow
```

1. If the Import Manager did not detect the file format automatically, make that selection now.
2. In **Delimiter**, select a delimiter from the dropdown. A common setup is to use the tab character to avoid splitting the data on printable characters that may occur within the values. Delimiters other than those in the dropdown are not allowed for import. If the file uses a different delimiter, the file must be converted outside of STEP, prior to import.
3. In **Character Set**, select the relevant character set. This determines the characters that can be successfully imported. The default character set is windows-1252. If the file contains special symbols, like trademark symbols, Unicode character set UTF-8 or UTF-16 is required to import correctly.
4. In **Has Header**, select if the file has a header line. If the first line (row) of the CSV file has header information such as attribute names that match STEP, the Automap feature is available to map the columns of data to the appropriate object in STEP. A header row is not required for import, but without it, the user must be able to identify the data for manual mapping.
5. In **Trim whitespace**, select if you want the system to strip off leading and trailing spaces in values.
6. In **Allow Multi Line Values**, select 'yes' if the file includes data with double quoted values that split over several lines. Typically, the newline character is interpreted as the end of data, delimiting a data record. However, in some cases, data spans more than one line – that is, includes a newline character. In such cases, values must be quoted with the double quote character (") to be imported correctly, and the Allow Multi Line Values must be set to yes.

In **Conversion Preview**, a sample of the first few lines of the file is displayed. This allows you to verify that the options you selected are correct.

Import Manager [Close]

Steps

- 1. Select Configuration
- 2. Select Data Source
- 3. Select Format**
- 4. Map Data
- 5. Identify Objects
- 6. Identify Destination
- 7. Select Business Rules
- 8. Advanced Settings

Select Format

Format: **CSV**

Converter for character separated formats where each line contains data about one object in fields separated by a delimiter character

Delimiter: **,**

Character Set: **UTF-8**

Has Header: **yes**

Trim whitespace: **yes**

Allow Multi Line Values: **no**

Conversion Preview:

<Name>	>	<Parent ID>	>	Primary Color	>	Secondary Color	>
Mens T PBO		18209		Black		Orange	
Mens T PBG		18209		Blue		Green	
Mens T PGS		18209		Green		Silver	
Mens T PGW		18209		Gray		White	
Mens T POY		18209		Orange		Yellow	

[Back] [Next] [Finish] [Cancel]

ETIM Format

Electro-Technical Information Model (ETIM) is a classification structure used to standardize the electronic exchange of product data for electrical and electronic products and to enable the electronic trading of these products. The standard is designed to meet the requirements of the electrical industry and allows for a uniform, technical description of electrical goods and their assignment to a class of products. In order to support customers who need to work with data using the ETIM standard, the STEP Import Manager includes a ETIM formats to allow for easy processing and setup of ETIM classifications and their related data elements.

ETIM Versions

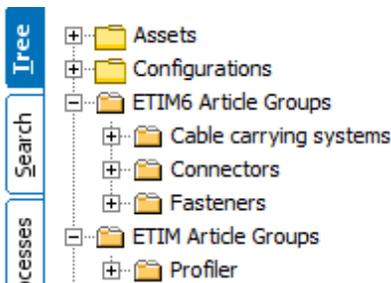
ETIM 5 and ETIM 6 can be imported into STEP using the Import Manager. The Import Manager cannot delete ETIM data or mark ETIM data as obsolete.

Both versions can exist in STEP simultaneously using two individual structures. Attributes, LOVs, Units, Object Types, and Classifications are not shared between ETIM versions.

- ETIM 5 uses CSV and each file contains a single language.
- ETIM 6 uses XML and includes multiple languages (English, German, Dutch, and a number of other European languages).

ETIM Classification

ETIM classifications are imported into STEP below the ETIM Article Groups node in the Tree.



Each imported ETIM article group will include the following Description attributes:

ETIM 5		ETIM 6	
Attribute Label	Description	Attribute Label	Description
ID	value displays ETIM preceding the ETIM ID	ID	value displays ETIM6 preceding the ETIM ID
Name	the ETIM Description of the article group	Name	the ETIM Description of the article group

ETIM 5		ETIM 6	
Attribute Label	Description	Attribute Label	Description
		ETIM6 Abbreviation	abbreviation from ETIM
ETIM Description	description from ETIM, matches Name attribute	ETIM6 Description	description from ETIM, matches Name attribute
ETIM ID	ID from ETIM, matches ID attribute	ETIM6 ID	ID from ETIM, matches ID attribute
ETIM Date	the date of the last change to the classification in ETIM	ETIM6 Date	the date of the last change to the classification in ETIM
ETIM Version	version from the ETIM system	ETIM6 Version	version from the ETIM system
ETIM Synonyms	multivalued and contains 0 or more ETIM synonyms for the article classification	ETIM6 Synonyms	multivalued and contains 0 or more ETIM synonyms for the article classification

ETIM Feature Types

ETIM operates with features that roughly translate to attributes and LOVs. An ETIM feature can be of one of the following types.

Type	Type Description	Result
Logic	The logical data type holds two values: true and false or yes and no.	A hard domain LOV with the ID ETIM Logical is created in STEP. It contains the values true and false. Features are imported as attributes using the LOV.
Alphanumeric	The alphanumeric data type contains a set of predefined values.	A hard domain LOV is created and contains the predefined set of values. Each value is provided with a value ID. Features are imported as attributes using the LOV.
Numeric	The numeric data type is a number that allows a unit	Features are imported as number-validated attributes with a unit.
Range	The numeric range that is possible with a valid unit	Features are imported as number-validated attributes with a unit.

Valid Values

In ETIM, the set of valid values for a feature depends on the classification of the linked article and the ETIM version.

ETIM 5

Typically, only one of the following ETIM 5 formats will be available in your STEP Import Manager:

- ETIM - this import converter creates a new attribute for each classification the feature is linked into. Each of the attributes (for the same ETIM feature) uses the same LOV but with an individual LOV filter of the valid values.
- ETIM v2 - this import converter avoids having multiple attributes for the same feature by moving the LOV filter specifying the valid values from the attribute to the attribute link. This should make it far easier to import, export, and maintain the attributes.

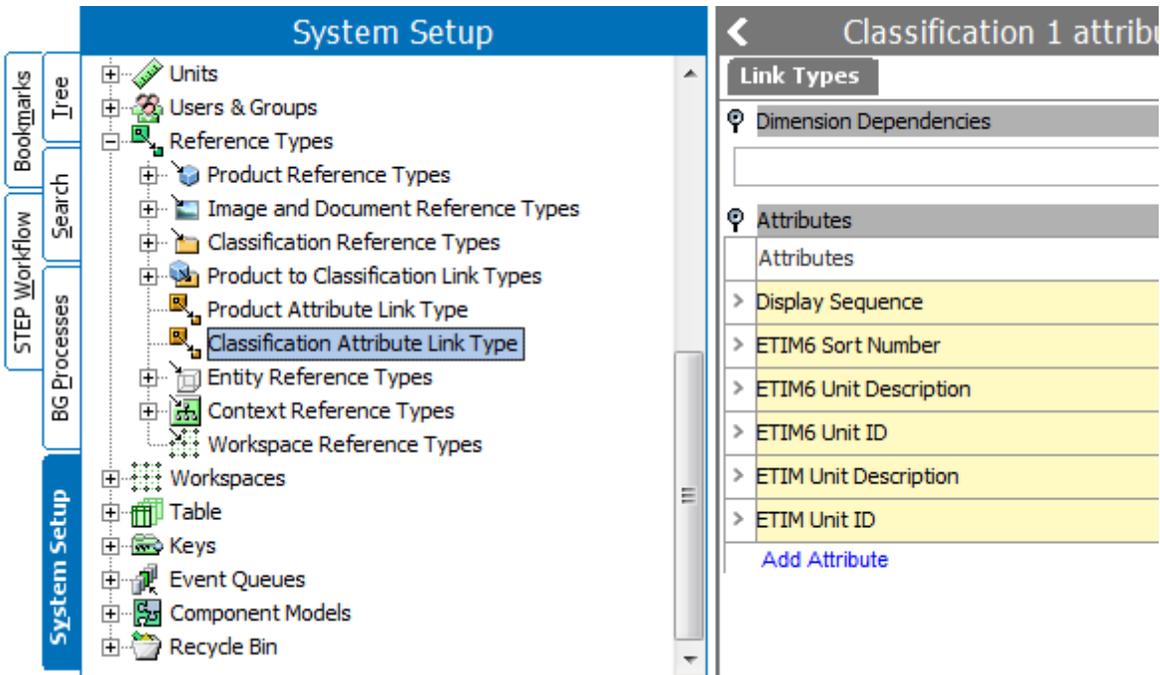
ETIM 6

- ETIM 6 - this import converter creates a new attribute for each classification the feature is linked into. Each of the attributes (for the same ETIM feature) uses the same LOV but with an individual LOV filter of the valid values.

Valid Units

In both ETIM 5 and ETIM 6, a feature does not have a valid unit. However, when it is linked into an article classification, one (and only one) valid unit can be specified. This is different from STEP where an attribute has the same valid units no matter where the attribute is used.

On the **Classification to Attribute Link**, the **ETIM Unit ID** and **Description** are stored as metadata to indicate that when used in this particular classification only the unit specified is valid.



Note: STEP does not include functionality to ensure only valid units are used when adding a value for an ETIM classification.

The ETIM standard currently specifies about 200 different units. A number of these units correspond to the units already available on STEP Workbench. When the ETIM classification system is imported, existing STEP units are reused if they correspond to the ETIM units. All of the ETIM unit values (new and existing) are populated for the metadata attributes ETIM Description and ETIM ID.

Note: It is possible to load various localizations of the ETIM classification system. For example, you can load an English version into the English context and a German version into the German context. While the IDs are the same across the various localizations, the descriptions of article groups, classifications, attributes, units, and lists of values are localized.

ETIM 5

Import Manager

Steps

1. Select Configuration
2. Select Data Source
- 3. Select Format**
4. Map Data
5. Identify Objects
6. Identify Destination
7. Select Business Rules
8. Advanced Settings

Select Format

Format: **ETIM v2**
 Imports an ZIP archive containing an ETIM database in CSV format. Will set up classification and attributes.

Article Group ID(s): **EG000015,EG000020, EG000023**

Character Set: **UTF-8**

Conversion Preview:

Description	Article Group ID
Accessories for lighting	EG000030
Alarm installations, emergency call and signalling	EG000054
Antenna and satellite technique	EG000033
Batteries and chargers	EG000053
Cabinet enclosures	EG000011
Cable and wire entry systems	EG000009
Cable carrying systems	EG000004
Cables and wires	EG000001
Circuit breakers and fuses	EG000020
Communication technique	EG000036
Connection devices	EG000048

Buttons: Back, Next, Finish, Cancel

1. Set **Format** based on the file to be imported. Based on your STEP configuration, the options displayed may include:
 - ETIM - creates duplicate attributes and multiple LOVs when more than one classification has multiple sets of valid values.
 - ETIM v2 - imports CSV and uses LOV filtering on the classifications, which avoids duplicate attributes and multiple LOVs when more than one classification has multiple sets of valid values.
2. Set **Article Group ID(s)** to display one or more classifications displayed in the Article Groups pane as follows:
 - A single Article Group ID to import one classification.
 - A comma-separated list of Article Group IDs to import several classifications.
 - Blank to import the full ETIM classification system.
3. In **Character Set** select the option that includes any special characters required by the data being imported.
4. Click **Next**.

ETIM 6

Import Manager

Steps

1. Select Configuration
2. Select Data Source
- 3. Select Format**
4. Map Data
5. Identify Objects
6. Identify Destination
7. Select Business Rules
8. Advanced Settings

Select Format

Format: **ETIM 6**
Imports an XML based file containing an ETIM database. Will set up Classifications, Attributes, LOVs, Units and Object Types.

Language: **en-GB**

Article Group ID(s): **EG000004, EG000049, EG000058**

Conversion Preview:

Description	Article Group ID
Accessories for lighting	EG000030
Alarm installations, emergency call and signalling	EG000054
Antenna and satellite technique	EG000033
Auxiliary material	EG000060

Buttons: Back, Next, Finish, Cancel

1. In **Format** ETIM 6 is automatically selected based on the Data Source selected. The Language field is displayed.
2. Set **Language** by selecting a single ISO Language Code to import. Multiple languages must be imported separately.

The Article Groups pane is updated to display the data available for import in the selected language.

Note: In Step 8 Advanced Settings you will select a Context for the import.

3. Set **Article Group ID(s)** to display one or more classifications displayed in the Article Groups pane as follows:
 - A single Article Group ID to import one classification.
 - A comma-separated list of Article Group IDs to import several classifications.
 - Blank to import the full ETIM classification system.
4. Click **Next**.

Excel Format

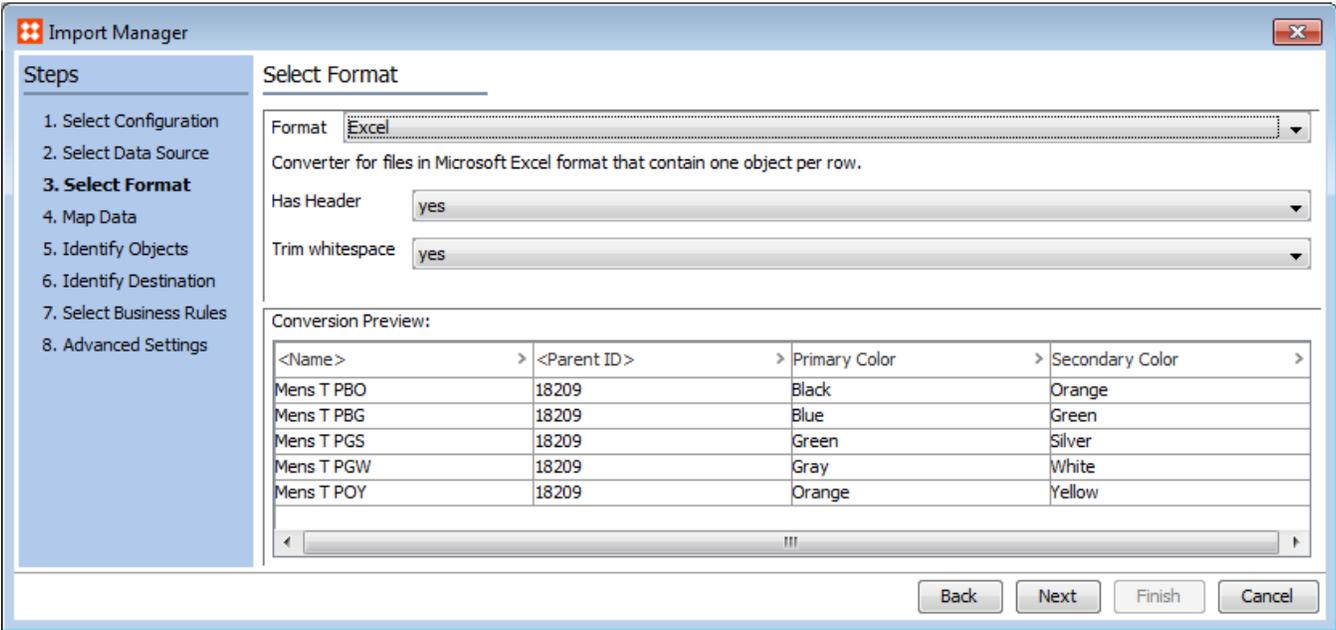
The importing process is very similar for CSV and Excel files. You can import new products, classifications, entities, attribute values and references. However, you cannot create new attributes or other system setup entries such as LOVs, users, reference types, and so on.

Note: Importing a CSV or Excel file that contains a mix of entities, classifications, and products requires several separate steps. Classification data must be imported first, then product data, and finally the entity data.

1. If the Import Manager did not detect file format automatically, make that selection now. Excel 2003 and Excel 2007 are supported
2. In **Has Header**, select whether the file has a header line. If the first line (row) of the Excel file has header information such as attribute names that match STEP, the Automap feature is available to map the columns of data to the appropriate object in STEP. A header row is not required for import, but without it, the user must be able to identify the data for manual mapping.
3. In **Trim whitespace**, select if you want the system to strip off leading and trailing spaces in values.

Note: Unicode is used as the character set for importing Excel data files.

In **Conversion Preview**, a sample of the first few lines of the file is displayed. This allows you to verify that the options you selected are correct.



Excel Smartsheet Format

When using the Import Manager to import a Smartsheet file, all relevant information is available from the Smartsheet. Selecting 'Excel Smartsheet' from the **Format** dropdown in step 3 of Import Manager is all that is required.

The import configuration can be saved and used for other Smartsheet imports where the same import options should be applied. This is useful when applying the same business rules and actions to multiple Smartsheet imports. The import configuration can be specified when exporting a Smartsheet using the Export Manager.

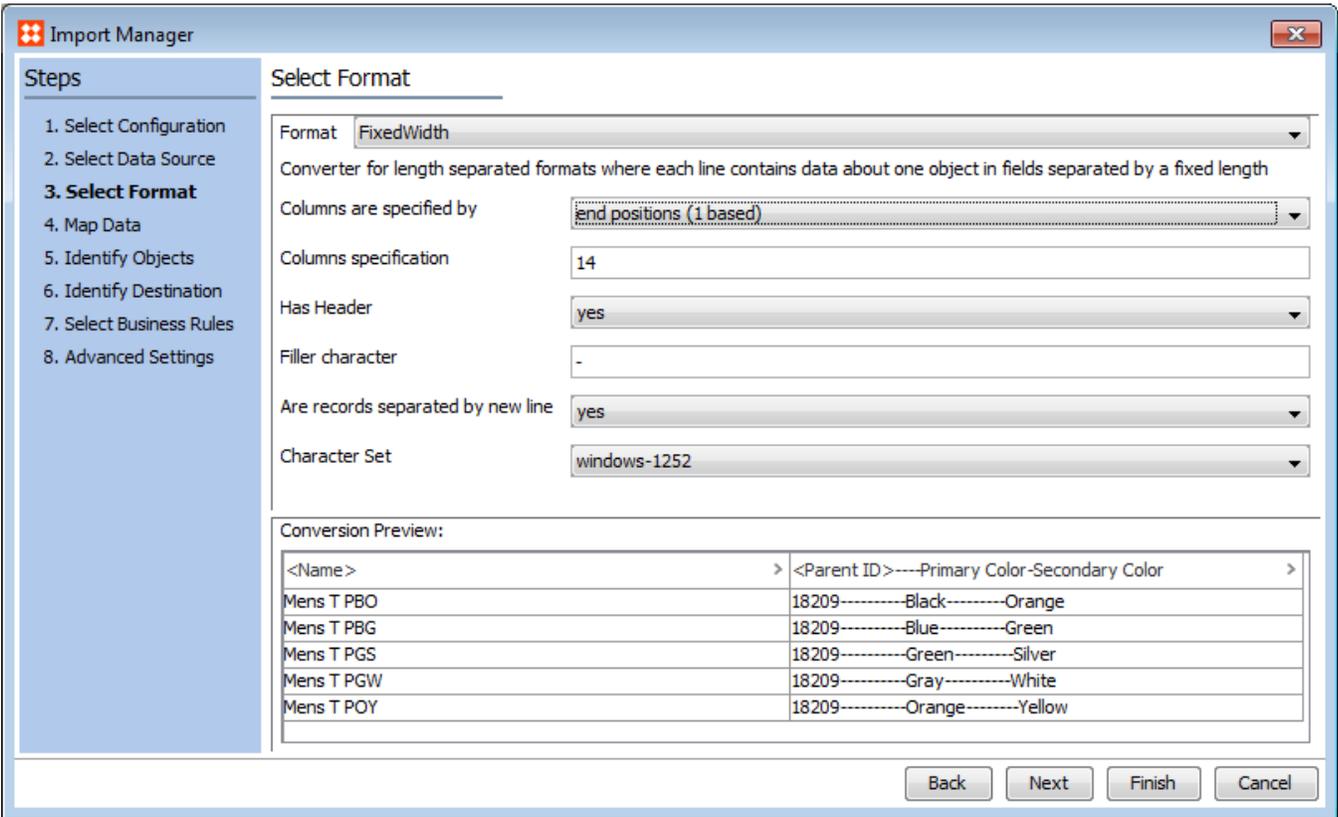
If an import configuration is not specified during import of the Smartsheet, a fallback import configuration can be specified on the SmartSheetImportAction in the Supplier portal.

For more information about Smartsheet import configurations, see [Importing Smartsheets with the Import Manager](#) in the Smartsheet documentation.

FixedWidth Format

A FixedWidth format is a text file with rows and columns, and each column has a fixed width.

1. If the Import Manager did not detect the file format automatically, make that selection now.
2. In **Columns are specified by** select how the initial element is identified. You have the following options:
 - end positions (zero-based)
 - end positions (one-based)
 - width
3. In **Columns specification**, specify the number of the last character of each column if you selected end positions in step 2. If you selected width in step 2, specify the width in characters of each column.
4. In **Has Header** select whether the file has a header line. If the first line (row) of the FixedWidth file has header information such as attribute names, you can use the Auto Map feature mapping the columns of data to STEP objects.
5. In **Filler Character**, specify the filler character that you want to remove from the fields - if any.
6. In **Are records separated by new line** select yes or no.
7. In **Character Set** select the relevant character set.



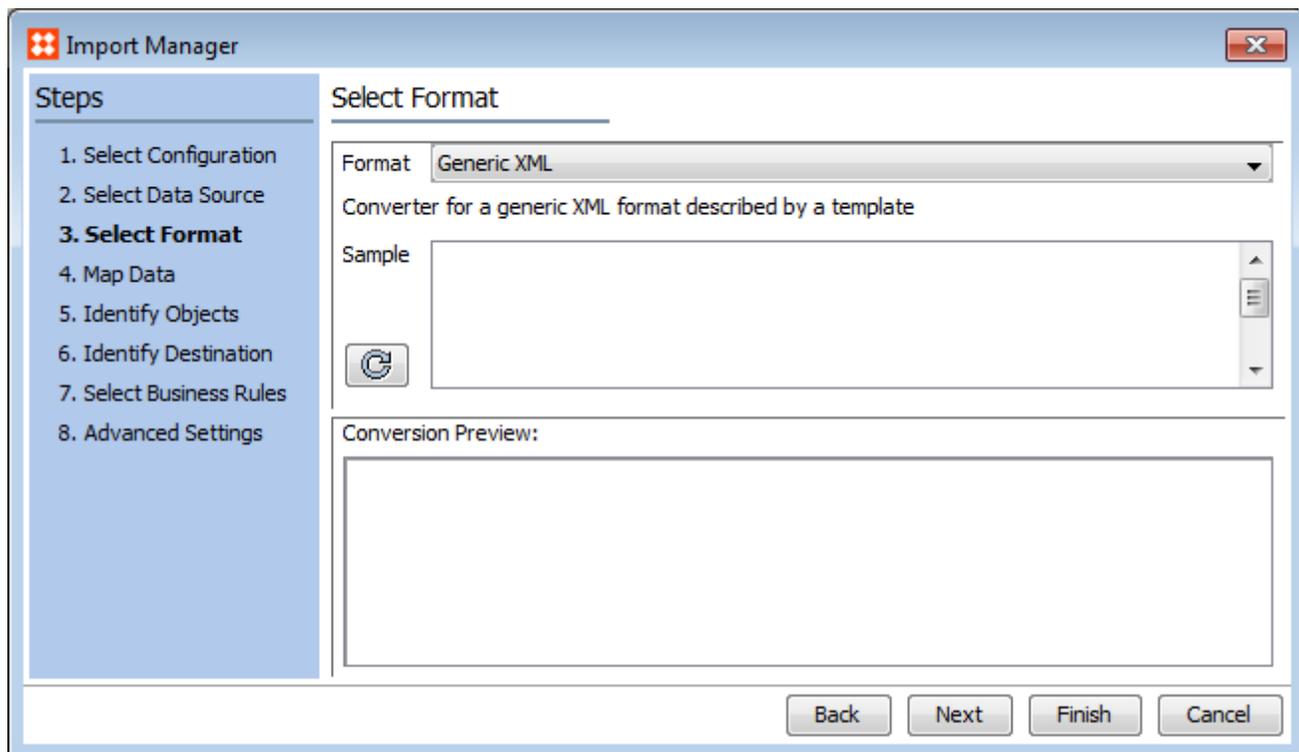
Generic XML Format

The Generic XML format in STEP allows you to import data from and export data to a variety of different XML formats without the need for extensions or customizations.

Generic XML is an XML-based language used for specifying how data can be extracted from a generic XML document. The basic idea is to parse the XML file via a processing instructions template that extracts data and represents it in a tabular format. This format can be handled with the standard Import Manager functionality, as if it was a CSV or Excel file.

The following description also applies to the IDoc MATMAS05 format.

- If the system did not detect file format automatically, make that selection now.



The following is an example of a Generic XML template:

```
<Products>
  <Product>
    <Name/>
  </Product>
</Products>
```

In the example, nodes are matched with the tag <Products> on the first level of the document. Below <Products>, the template matches nodes with the tag name <Product>.

Processing Instructions

Note: Only instructions included in both the document and the generic template are processed. Instructions not defined in the template are ignored.

Descriptions and examples of the available processing instructions used by STEP within Generic XML are covered in the following sections:

- Record Instruction
- Source Instruction
- MultiSource Instruction
- SourceID Instruction
- Repeated Instruction
- DimensionPointID Instruction

For an example of an input document, template and the conversion preview, see the **Generic XML Import Simple Example** or **Generic XML Import Advanced Example** sections of the **Importing Data and Images** documentation.

Record Instruction

The template must declare which element corresponds to a single record (product, entity, etc.). This is done using the `<?Record?>` processing instruction, which applies to the XML node that surrounds it. A template must contain exactly one `<?Record?>` processing instruction.

```
<Products>
  <Product>
    <?Record?>
      <Name />
    </Product>
  </Products>
```

In this example, STEP will extract data for each product object inside the `<Product>` and `<Products>` elements, regardless of the number that exist. The product node is the record node of the template. The record node must not be declared as a repeated node. Although this is a valid template, no data is being supplied by this template.

Source Instruction

To extract the content of an XML element as a field, place a `<?Source [OptionalIdentifier]?>` processing instruction within the `<?Record?>` scope of the template. By default, the heading (column) name of a field is the name of the surrounding parent node.

```
<Products>
  <Product>
    <?Record?>
      <Name><?Source?></Name>
      <Text><?Source Description?></Text>
    </Product>
  </Products>
```

In the above example, each product hierarchy encountered exports a row with data for the columns Name and Description.

To make the mapping step easier, use your own identifiers by defining an optional identifier. As shown in the `<Text>` instruction in the above example, the Text field will display with a heading of 'Description' since the optional identifier was used within the source tag. The optional identifier displays in the Map Data step.

You can also extract the value of an attribute by using the `[?Source [Optional Identifier]?]` instruction as follows:

```
<Products>
  <Product Name="[?Source?]" Text="[?Source Description?]">
    <?Record?>
  </Product>
</Products>
```

Preventing Duplicate Field Names

Fields extracted using the `<?Source?>` and `<?MultiSource?>` instructions can sometimes have the same names, either because of identical tag or attribute names, or because of duplication of names explicitly chosen by the template creator. In this situation, the Import Manager ensures that all field names are unique by adding a post-fix of the form (n), where n is the first integer ≥ 2 for which the combined field name is not already in use.

If a repeated scope contains multiple `<?Source?>` or `<?MultiSource?>` instructions, and one of them results in a name duplication, only that field will be renamed.

Note: Resolve name collisions by providing explicit, non-duplicating names.

MultiSource Instruction

Like the `<?Source?>` instruction, the `<?MultiSource?>` instruction also supports explicitly specifying a field name using an optional identifier. For more information, see the **Preventing Duplicate Field Names** section of the **Source Instruction** documentation.

To extract the value of a repeating XML element as a multi-value field that can be mapped to a multi-valued attribute / reference in STEP, place a `<?MultiSource [OptionalIdentifier]?>` instruction in either the node or the attribute of the `<?Record?>` scope:

```
<Products>
<Product ID=" [?Source?] ">
<?Record?>
<KeyWords>
<Keyword><?MultiSource?></Keyword>
</KeyWords>
</Product>
</Products>
```

Within the same scope, multiple nodes / attributes matching a `<?MultiSource [OptionalIdentifier]?>` declaration will contribute to the same multi-value field. The input document below, uses the template above.

```
<Products>
<Product ID="42">
<KeyWords>
<Keyword>Red</Keyword>
<Keyword>Green</Keyword>
<Keyword>Blue</Keyword>
</KeyWords>
</Product>
</Products>
```

It produces the fields 'ID' and 'Keyword' with values ID=42 and Keyword=Red|Green|Blue. Note that container-node, 'KeyWords' in this case, is not a requirement to extract a multi-value field. A document with the 'Keyword' nodes directly under the 'Product' node could be processed by using the following template:

```
<Products>
<Product ID=" [?Source?] ">
<?Record?>
<Keyword><?MultiSource Keyword?></Keyword>
</Product>
</Products>
```

SourceID Instruction

The `<?SourceID?>` is used to extract the identifier for a repeated element when the identifier is a nested element. `[?SourceID?]` is used to extract the identifier for a repeated element when the identifier is an attribute. `?SourceID?` instructions must be placed inside repeated scope. Only one `?SourceID?` is allowed for each scope.

The following is a Generic XML template for processing the input document above:

```
<Products>
  <Product ID="[?Source?]">
    <?Record?>
      <Values>
        <Value AttributeID="[?SourceID?]">
          <?Repeated?>
            <?Source?>
              </Value>
            </Values>
          </Product>
        </Products>
```

When used on the input document, this template outputs the fields 'ID', 'Value-52', 'Value-77'. In this case, the subtree from 'Value' and down is said to be in repeated scope.

The following is an example of an input document that requires the repeated node and ID source declarations to happen on different nodes:

```
<Products>
  <Product ID="EXA-5002-1004">
    <Attributes>
      <Attribute>
        <ID>52</ID>
        <Value>Some description text</Value>
      </Attribute>
      <Attribute>
        <ID>77</ID>
        <Value>53 kg</Value>
      </Attribute>
    </Attributes>
  </Product>
</Products>
```

The Generic XML template this time extracts the sourceID from a node value:

```
<Products>
  <Product ID="[?Source?]">
```

```
<?Record?>  
<Attributes>  
  <Attribute>  
    <?Repeated?>  
    <ID><?SourceID?></ID>  
    <Value><?Source?></Value>  
  </Attribute>  
</Attributes>  
</Product>  
</Products>
```

For the input document, this template generates the fields **ID**, **Value-52** and **Value-77**.

Repeated Instruction

Additional declarations are required when extracting data from repeated structures below the record node. Declaration of a repeated node happens with the `<?Repeated [OptionalSpaceSeparatedFilters]?>` processing instruction. The `<?Repeated?>` instruction must be placed inside `<?Record?>` scope. For example:

```
<Products>
  <Product ID="EXA-5002-1004">
    <Values>
      <Value AttributeID="52">Some description text</Value>
      <Value AttributeID="77">53 kg</Value>
    </Values>
  </Product>
</Products>
```

Note that the **Value** node is repeated. To extract data from both occurrences, the template must declare that a node is a repeated node, which means that a set of fields has to be generated for each node encountered. The template also has to declare an ID source on or below the repeated node. The ID source is used to map each occurrence of a repeated node to a specific set of fields of the form `[Identifier].[SourceTagName]`. This is required to ensure that related repeated nodes map to the same fields across different records. A repeated node may not be declared as a record node as well.

By default, a repeated scope produces a set of fields for each ID Source value encountered in the input document. Using the 'optional space separated filters' allows you to select values from specific ID keys or to guard against additional columns being generated as input documents evolve in the future.

```
<ARTICLE>
  <ARTICLE_PRICE_DETAILS>
    <DATETIME type="valid_start_date">
      <DATE>2010-01-01</DATE>
    </DATETIME>
  <PRODUCT_PRICE price_type="net_list">
    <PRICE_AMOUNT>11.08</PRICE_AMOUNT>
    <PRICE_CURRENCY>EUR</PRICE_CURRENCY>
    <TAX>0.19</TAX>
    <LOWER_BOUND>1</LOWER_BOUND>
  </PRODUCT_PRICE>
  <PRODUCT_PRICE price_type="nrp">
    <PRICE_AMOUNT>23.05</PRICE_AMOUNT>
    <PRICE_CURRENCY>EUR</PRICE_CURRENCY>
    <TAX>0.19</TAX>
    <LOWER_BOUND>1</LOWER_BOUND>
  </PRODUCT_PRICE>
  <PRODUCT_PRICE price_type="employee">
```

```

    <PRICE_AMOUNT>8.52</PRICE_AMOUNT>
    <PRICE_CURRENCY>EUR</PRICE_CURRENCY>
    <TAX>0.19</TAX>
    <LOWER_BOUND>1</LOWER_BOUND>
  </PRODUCT_PRICE>
</ARTICLE_PRICE_DETAILS>
</ARTICLE>

```

The price and currency information can be extracted for the net_list and nrp price types only by using the following template:

```

<ARTICLE>
  <?Record?>
  <ARTICLE_PRICE_DETAILS>
    <PRODUCT_PRICE price_type="[?SourceID?]">
      <?Repeated net_list nrp?>
        <PRICE_AMOUNT><?Source Amount?></PRICE_AMOUNT>
        <PRICE_CURRENCY><?Source Currency?></PRICE_CURRENCY>
      </PRODUCT_PRICE>
    </ARTICLE_PRICE_DETAILS>
  </ARTICLE>

```

This produces the fields: Currency-net_list=EUR, Amount-net_list=11,08, Currency-nrp=EUR, Amount-nrp=23.05. The <?Repeated?> instruction can take an arbitrary number of filter terms separated by white-space characters. When no terms are defined, the filtering functionality is disabled.

DimensionPointID Instruction

Values in generic XML for the same attribute in different languages could be represented using the same ID with an additional attribute stating the language. This normally causes a name collision in generic XML. However, generic XML can import such dimension dependent data using the `<?DimensionPointID?>` instruction. This instruction can be used in a `?Repeated?` block in combination with `?SourceID?` or alone. When the `?DimensionPointID?` instruction is used, the name of the field extracted will be appended with the value of the `?DimensionPointID?` in square brackets, thus avoiding the name collision.

It is important to note that the use of the `?DimensionPointID?` instruction, does not actually import into the specified dimension points - this has to be set up using transformations in the Map Data step see [Mapping Data Manually](#). For this reason the values appended to the column names using the `?DimensionPointID?` do not have to match the Dimension Point IDs, as this can be handed in the Map Data step of the import.

The following example shows how language dependent data can be extracted using generic XML. Note that the ID of the attribute is the same value, 52.

```
<Products>
  <Product ID="EXA-5002-1004">
    <Attributes>
      <Attribute>
        <ID>52</ID>
        <Lang>Eng</Lang>
        <Value>Some description text</Value>
      </Attribute>
      <Attribute>
        <ID>52</ID>
        <Lang>De</Lang>
        <Value>Einige beschreibenden Text</Value>
      </Attribute>
    </Attributes>
  </Product>
</Products>
```

The following Generic XML template extracts the sourceID from a node value, and appends the value inside the `<Lang>` tags in square brackets, e.g. '52[Eng]' and '52[De]' so they can be mapped to the respective languages.

```
<Products>
  <Product ID=" [?Source?] ">
    <?Record?>
      <Attributes>
        <Attribute>
          <?Repeated?>
            <ID><?SourceID?></ID>
            <Lang><?DimensionPointID?></Lang>
```

```
<Value><?Source?></Value>  
</Attribute>  
</Attributes>  
</Product>  
</Products>
```

Generic XML Import Simple Example

The following is an example of an import document and template that uses most of the features present in a Generic XML template.

Input Document

Copy the following text to a new file and save as an XML file. Open Import Manager and select the file as the Data Source Filename.

```
<Products>
  <Product>
    <Name>Pilot Pen</Name>
    <Market>
      <Priority>Low</Priority>
      <Attributes>
        <Attribute ID="42" UnitID="Pounds">3.068</Attribute>
        <Attribute ID="84" UnitID="Dollars">12.499</Attribute>
      </Attributes>
      <MarketID>USA</MarketID>
    </Market>
    <Market>
      <Priority>High</Priority>
      <Attributes>
        <Attribute ID="42" UnitID="Kg">1534</Attribute>
        <Attribute ID="84" UnitID="Euro">22.995</Attribute>
      </Attributes>
      <MarketID>EU</MarketID>
    </Market>
    <Colors>
      <Color>Red</Color>
      <Color>Green</Color>
      <Color>Blue</Color>
    </Colors>
  </Product>
  <Product>
    <Name>Parker Pen</Name>
    <Market>
      <Priority>Low</Priority>
      <Attributes>
        <Attribute ID="42" UnitID="Pounds">3.728</Attribute>
```

```

        <Attribute ID="84" UnitID="Dollars">24.199</Attribute>
    </Attributes>
    <MarketID>USA</MarketID>
</Market>
<Market>
    <Priority>Extreme</Priority>
    <Attributes>
        <Attribute ID="42" UnitID="Kg">1864</Attribute>
        <Attribute ID="84" UnitID="Euro">29.999</Attribute>
    </Attributes>
    <MarketID>EU</MarketID>
</Market>
<Colors>
    <Color>Gold</Color>
    <Color>Silver</Color>
</Colors>
</Product>
</Products>

```

Template

Copy the following text and paste it in the Import Manager Select Format step, Sample field.

```

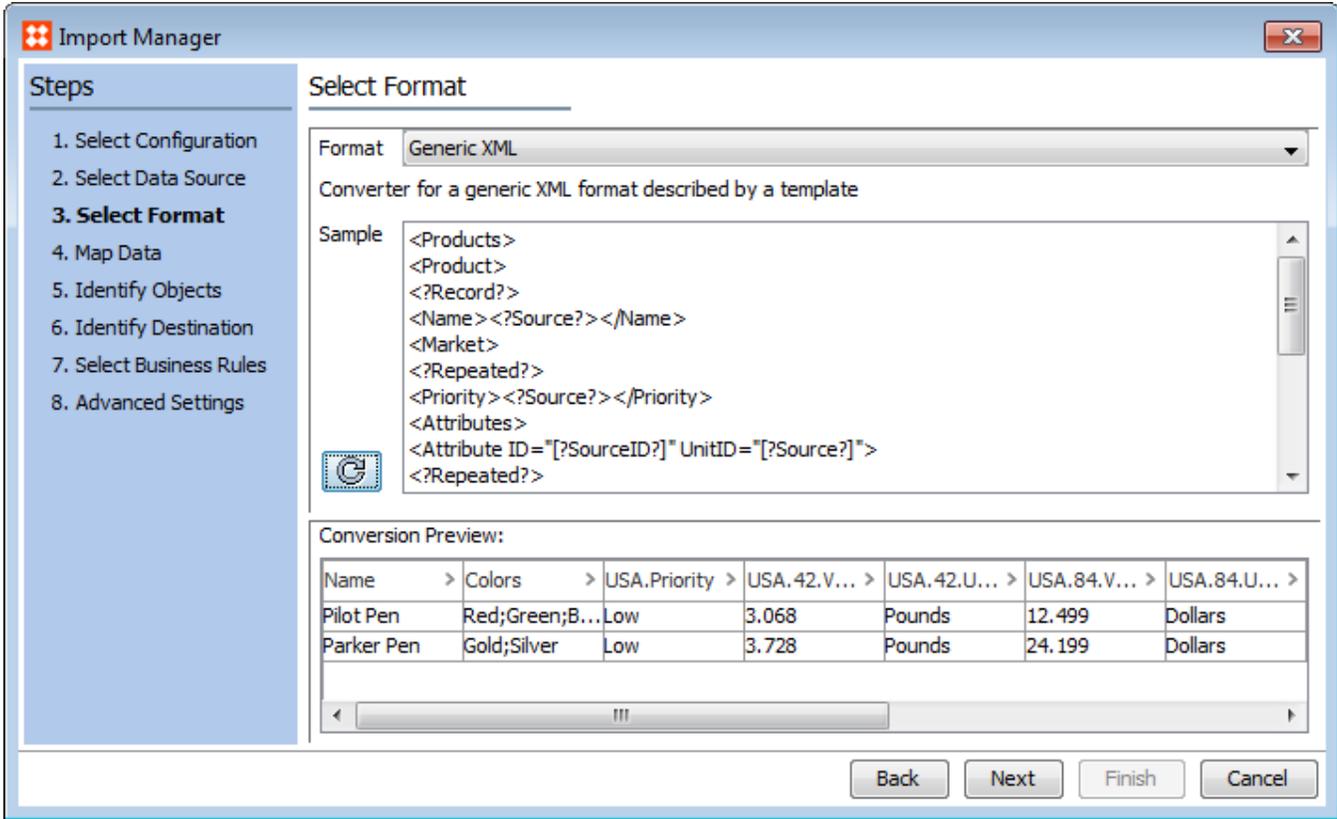
<Products>
  <Product>
    <?Record?>
      <Name><?Source?></Name>
      <Market>
        <?Repeated?>
          <Priority><?Source?></Priority>
          <Attributes>
            <Attribute ID="[?SourceID?]" UnitID="[?Source?]">
              <?Repeated?>
                <?Source Value?>
              </Attribute>
            </Attributes>
          <MarketID><?SourceID?></MarketID>
          </Market>
          <Colors>
            <Color><?MultiSource Colors?></Color>
          </Colors>
        </?Repeated?>
      </Name>
    </Product>
  </Products>

```

```
</Product>
</Products>
```

Conversion Preview

Click the Sample reload button . The Conversion Preview area displays the results of the input document against the provided template. The following Map Data step presents data in the same format that is used for and CSV / Excel import. Additionally, Generic XML data can be mapped and transformed in the same way as CSV / Excel data.



Import Manager

Steps

1. Select Configuration
2. Select Data Source
- 3. Select Format**
4. Map Data
5. Identify Objects
6. Identify Destination
7. Select Business Rules
8. Advanced Settings

Select Format

Format: **Generic XML**

Converter for a generic XML format described by a template

Sample

```
<Products>
<Product>
<?Record?>
<Name><?Source?></Name>
<Market>
<?Repeated?>
<Priority><?Source?></Priority>
<Attributes>
<Attribute ID="[?SourceID?]" UnitID="[?Source?]">
<?Repeated?>
```

Conversion Preview:

Name	Colors	USA.Priority	USA.42.V...	USA.42.U...	USA.84.V...	USA.84.U...
Pilot Pen	Red;Green;B...	Low	3.068	Pounds	12.499	Dollars
Parker Pen	Gold;Silver	Low	3.728	Pounds	24.199	Dollars

Buttons: Back, Next, Finish, Cancel

Generic XML Import Advanced Example

The following is an example of a more complex import document and template that uses most of the features present in a Generic XML template.

Input Document

Copy the following text to a new file and save as an XML file. Open Import Manager and select the file as the Data Source Filename. Note that any number of additional items could be included by repeating the data within the Item tag.

```
<?xml version="1.0" encoding="utf-8"?>
<ItemLoad>
  <Items>
    <Item>
      <EAN>2700524977488</EAN>
      <PrimarySpecs>
        <ProductInformation>
          <General>
            <ProductName>AC-UZ444</ProductName>
            <ConsumerShortDescription>Active 3D Glasses</ConsumerShortDescription>
            <AvailableFrom>2015-01-01 00:00:00</AvailableFrom>
          </General>
          <Variants>
            <Name>Color</Name>
            <Value>Gray</Value>
          </Variants>
          <Variants>
            <Name>Battery</Name>
            <Value>C2</Value>
          </Variants>
          <Manuals>
            <Manual type="ManualDE">Manual-9</Manual>
            <Manual type="ManualEN">Manual-10</Manual>
          </Manuals>
        </ProductInformation>
      </PrimarySpecs>
      <SyncProtocols>
        <Protocol>Acme</Protocol>
        <Protocol>Sony</Protocol>
        <Protocol>Sharp</Protocol>
      </SyncProtocols>
    </Item>
  </Items>
</ItemLoad>
```

```

    </SyncProtocols>
  </Item>
</Items>
</ItemLoad>

```

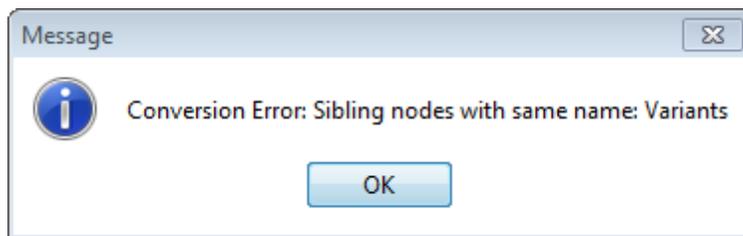
Creating a Template

To create a template from an import file, copy the file without the XML declaration (the first line that indicates the XML version), and paste into the Sample field on the Import Manager Select Format step. Then, remove any duplicate elements. Since there is only a single Item element in the example above, it can be pasted as is.

Note: When creating a template from a source file, ensure all of the elements that can occur, and that must be handled, are represented. Only instructions included in both the document and the generic template are processed. Instructions not defined in the template are ignored.

Remove Redundant Elements

Click the Sample reload button  and a message is displayed about sibling nodes with the same name:



Remove of repetitions of the 'Variants', 'Manual', and 'Protocol' elements, until only one of each remains. The Sample field now displays the following:

```

<ItemLoad>
  <Items>
    <Item>
      <EAN>2700524977488</EAN>
      <PrimarySpecs>
        <ProductInformation>
          <General>
            <ProductName>AC-UZ444</ProductName>
            <ConsumerShortDescription>Active 3D Glasses</ConsumerShortDescription>
            <AvailableFrom>2015-01-01 00:00:00</AvailableFrom>
          </General>
          <Variants>
            <Name>Color</Name>
            <Value>Gray</Value>
          </Variants>
        </ProductInformation>
      </PrimarySpecs>
    </Item>
  </Items>
</ItemLoad>

```

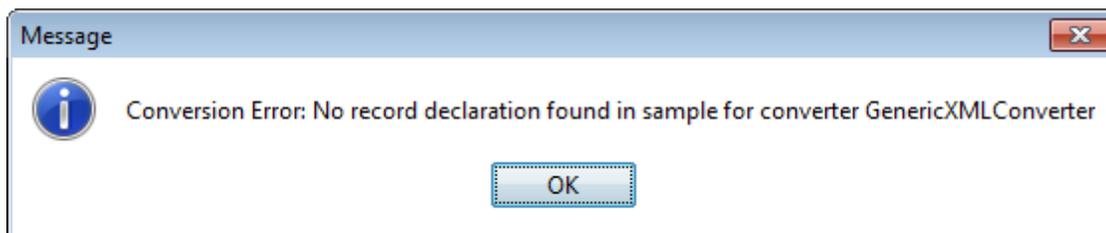
```

    <Manuals>
      <Manual type="ManualDE">Manual-9</Manual>
    </Manuals>
  </ProductInformation>
</PrimarySpecs>
<SyncProtocols>
  <Protocol>Acme</Protocol>
</SyncProtocols>
</Item>
</Items>
</ItemLoad>

```

Add Record and Source Instructions

Click the Sample reload button  and a message is displayed about a missing record declaration:



Use the `<?Record?>` and `<?Source?>` instructions to handle all the cases where you simply need to extract one value between a start and end tag.

```

<ItemLoad>
  <Items>
    <Item>
      <?Record?>
      <EAN><?Source?></EAN>
      <PrimarySpecs>
        <ProductInformation>
          <General>
            <ProductName><?Source?></ProductName>
            <ConsumerShortDescription><?Source?></ConsumerShortDescription>
            <AvailableFrom><?Source?></AvailableFrom>
          </General>
          <Variants>
            <Name>Color</Name>
            <Value>Gray</Value>
          </Variants>
        </ProductInformation>
      </PrimarySpecs>
    </Item>
  </Items>
</ItemLoad>

```

```

    <Manuals>
      <Manual type="ManualDE">Manual-9</Manual>
    </Manuals>
  </ProductInformation>
</PrimarySpecs>
<SyncProtocols>
  <Protocol>Acme</Protocol>
</SyncProtocols>
</Item>
</Items>
</ItemLoad>

```

Add Repeated Instruction

To see how each option affects the outcome, make the following updates in the Sample template and then click the Sample reload button  to update the Conversion Preview.

For the 'Variants' element, if we insert the <?Source?> instruction in the 'Value' element as shown below, the result will be the value for the first repetition, 'Gray'.

```

<Variants>
  <Name>Color</Name>
  <Value><?Source?></Value>
</Variants>

```

Conversion Preview:

EAN	ProductName	ConsumerShort...	AvailableFrom	Value
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Gray

To get values from both 'Variants' elements, we have two options: MultiSource and Source / Repeated / SourceID.

To get the Value from each repetition and be able to map them all to a multi valued attribute / multiple reference / links targets, we can use the <?MultiSource?> instruction.

```

<Variants>
  <Name>Color</Name>
  <Value><?MultiSource?></Value>
</Variants>

```

Conversion Preview:

EAN	ProductName	ConsumerShort...	AvailableFrom	Value
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Gray;C2

If the data should be mapped to different attributes, as is the case in our example, where the Name value of each repetition identifies the attribute, use these three instructions:

- `<?Source?>`
- one `<?Repeated?>` to indicate that Variants is a repeated element
- one `<?SourceID?>` to indicate that the Name value should be used as an identifier for each repetition

```
<Variants>
  <?Repeated?>
  <Name><?SourceID?></Name>
  <Value><?Source?></Value>
</Variants>
```

Conversion Preview:						
EAN	ProductName	ConsumerSh...	AvailableFrom	Color.Value	Battery.Value	
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Gray	C2	

The Import Manager displays the column header for each element using the pattern [Identifier].[SourceTagName].

Add Repeated Instruction With a Filter

To see how each option affects the outcome, make the following updates in the Sample template and then click the Sample reload button  to update the Conversion Preview.

The repeated Manual element is handled differently since the type of identifier is an attribute.

Again, start by placing the `<?Source?>` instruction to the value:

```
<Manuals>
  <Manual type="ManualDE"><?Source?></Manual>
</Manuals>
```

Conversion Preview:						
EAN	ProductName	ConsumerShortDes...	AvailableFrom	ManualDE	Color.Value	Battery.Value
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Manual-9	Gray	C2

Again, to get values for all repetitions, use the `<?Repeated?>` instruction and also add an identifier instruction. This time, since the identifier is an attribute value, use the square bracket version `[?SourceID?]`.

```
<Manuals>
  <Manual type="[?SourceID?]"><?Repeated?><?Source?></Manual>
</Manuals>
```

Conversion Preview:							
EAN	ProductName	ConsumerShort...	AvailableFrom	Color.Value	Battery.Value	ManualDE.Manual	ManualEN.Manual
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Gray	C2	Manual-9	Manual-10

To filter repeated elements so that only elements with specific identifiers are considered, add the identifier to the `<?Repeated?>` instruction. To get only ManualEN data, update the template as follows.

```
<Manuals>
  <Manual type="[?SourceID?]"><?Repeated ManualEN?><?Source?></Manual>
</Manuals>
```

Conversion Preview:

EAN	ProductName	ConsumerShortDescription	AvailableFrom	Color.Value	Battery.Value	ManualEN.Manual
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Gray	C2	Manual-10

Add MultiSource Instruction

For the repeated Protocol element there is no identifier, so it makes sense to use the `<?MultiSource?>` processing instruction. When this instruction is used, the `<?Repeated?>` instruction is not required.

```
<SyncProtocols>
  <Protocol><?MultiSource?></Protocol>
</SyncProtocols>
```

Conversion Preview:

EAN	ProductName	ConsumerShortDescription	AvailableFrom	Protocol	Color.Value	Battery.Value	ManualEN.Manual
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Acme;Sony;Sharp	Gray	C2	Manual-10

Now that all instructions have been added, the template is below.

```
<ItemLoad>
  <Items>
    <Item>
      <?Record?>
      <EAN><?Source?></EAN>
      <PrimarySpecs>
        <ProductInformation>
          <General>
            <ProductName><?Source?></ProductName>
            <ConsumerShortDescription><?Source?></ConsumerShortDescription>
            <AvailableFrom><?Source?></AvailableFrom>
          </General>
          <Variants>
            <?Repeated?>
            <Name><?SourceID?></Name>
            <Value><?Source?></Value>
          </Variants>
          <Manuals>
            <Manual type="[?SourceID?]"><?Repeated?><?Source?></Manual>
```

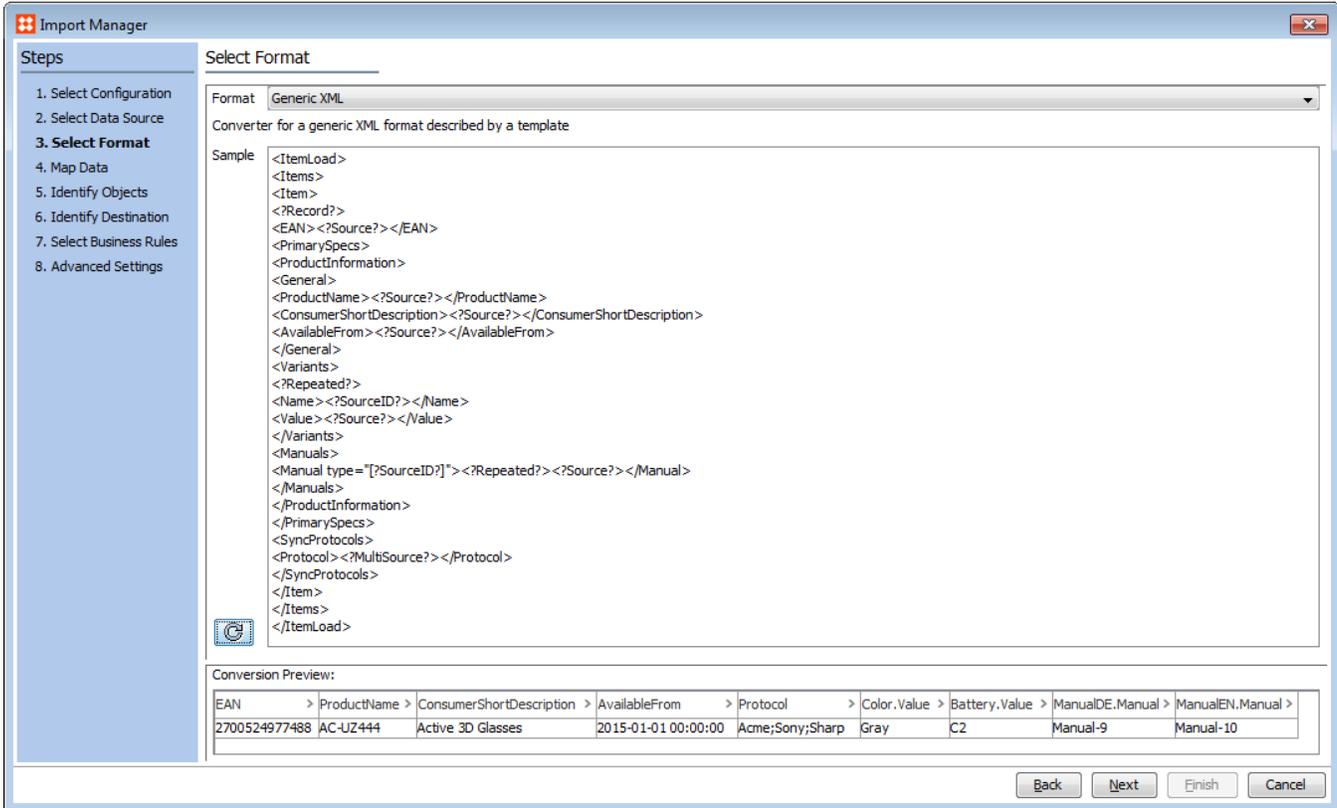
```

    </Manuals>
  </ProductInformation>
</PrimarySpecs>
<SyncProtocols>
  <Protocol><?MultiSource?></Protocol>
</SyncProtocols>
</Item>
</Items>
</ItemLoad>

```

Conversion Preview

Click the Sample reload button . The Conversion Preview area displays the results of the input document against the provided template.



Steps

- Select Configuration
- Select Data Source
- Select Format**
- Map Data
- Identify Objects
- Identify Destination
- Select Business Rules
- Advanced Settings

Select Format

Format: Generic XML

Converter for a generic XML format described by a template

Sample

```

<ItemLoad>
<Items>
<Item>
  <?Record?>
  <EAN><?Source?></EAN>
  <PrimarySpecs>
  <ProductInformation>
  <General>
  <ProductName><?Source?></ProductName>
  <ConsumerShortDescription><?Source?></ConsumerShortDescription>
  <AvailableFrom><?Source?></AvailableFrom>
  </General>
  <Variants>
  <?Repeated?>
  <Name><?SourceID?></Name>
  <Value><?Source?></Value>
  </Variants>
  <Manuals>
  <Manual type="[?SourceID?]"><?Repeated?><?Source?></Manual>
  </Manuals>
  </ProductInformation>
  </PrimarySpecs>
  <SyncProtocols>
  <Protocol><?MultiSource?></Protocol>
  </SyncProtocols>
</Item>
</Items>
</ItemLoad>

```

Conversion Preview:

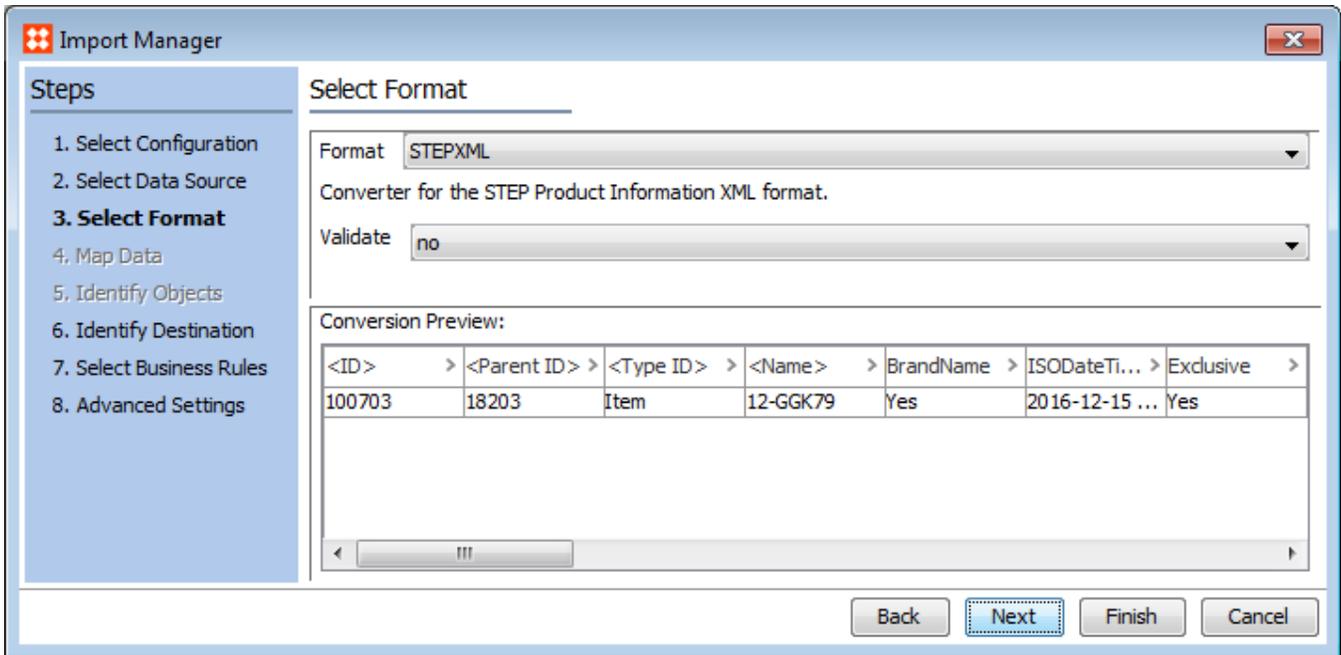
EAN	ProductName	ConsumerShortDescription	AvailableFrom	Protocol	Color.Value	Battery.Value	ManualDE.Manual	ManualEN.Manual
2700524977488	AC-UZ444	Active 3D Glasses	2015-01-01 00:00:00	Acme;Sony;Sharp	Gray	C2	Manual-9	Manual-10

Buttons: Back, Next, Finish, Cancel

STEPXML Format

Important: Not all import steps are required when importing a STEPXML file, since STEP assumes that all data in the file is to be imported without modifications.

1. If the Import Manager did not detect file format automatically, make that selection now.
2. In **Validate**, select if the system should check the validity of the XML file before loading any data. The header file states if it should be validated against schema or DTD.



Note: The character set for importing XML data files is always assumed to be Unicode.

Schema or DTD Validation

The STEP-ProductInformation tag in STEPXML file specifies if contents should be validated against DTD or schema. The import manager uses the validation specified in the tag as illustrated below. For more information on Schema and DTD, see the STEP API documentation. As an example, the following shows Schema validation:

```
<STEP-ProductInformation xmlns="http://www.stibosystems.com/step" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.stibosystems.com/step PIM.xsd" ExportTime="2011-10-18 13:45:04"
ExportContext="DE All All" ContextID="DE All All" WorkspaceID="Main">
...
</STEP-ProductInformation>
```

Auto Approval

To approve objects during import, inserting `AutoApprove="Y"` in the `STEP-ProductInformation` tag is required as illustrated in the following.

```
<STEP-ProductInformation xmlns="http://www.stibosystems.com/step" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.stibosystems.com/step PIM.xsd" ExportTime="2011-10-18 13:45:04"
ExportContext="DE All All" ContextID="DE All All" WorkspaceID="Main" AutoApprove="Y">
...
</STEP-ProductInformation>
```

SingleUpdateMode

Importing certain changes requires that the STEP system enters single update mode. This is to ensure that during import, only the import process is allowed to modify data in the database. When the import process enters single update mode, no user can write any data; they are restricted to only viewing data.

The following import changes require single update mode:

- Adding a dimension to attributes, object types, reference types, units, LOVs, etc.
- Changing attributes from being internally maintained to being externally maintained
- Changing validation base type settings on attributes or term object types.

To make the import process enter single update mode, insert `SingleUpdateMode="Y"` in the `STEP-ProductInformation` tag. If the import process cannot enter single update mode because users are working in STEP, the import process goes into a wait state until single update mode is possible. For example:

```
<STEP-ProductInformation xmlns="http://www.stibosystems.com/step" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.stibosystems.com/step PIM.xsd" ExportTime="2011-10-18 13:45:04"
ExportContext="DE All All" ContextID="DE All All" WorkspaceID="Main" SingleUpdateMode="Y">
```

Deleting References

You can delete references using the `Replacement Rules` tag (shown below), however, this removes **all** references of the indicated type for **all** loaded objects.

```
<ReplacementRules>
<ReplaceProductToProductCrossReference ReferenceTypeID="ProductToSupersededBy"/>
</ReplacementRules>
```

Note: The `DeleteProductReference` and `DeleteProductCrossReference` tags are only available for outbound XML when using event-based integration endpoints as discussed in **Deleting Product References with Advanced STEPXML** in the **Integration Endpoints for Data Exchange / Integration Endpoint User Guide** documentation.

STEPXML Import Overview

STEPXML can contain a number of different processing instructions for the import process. Some of these instructions are similar to the options available in the Import Manager wizard for the tabular formats.

Create Objects

New objects can be created in the Tree and in System Setup using the STEPXML import. Minimum requirements for information vary based on the object type. For example, to create a new product, the file must include ID, parent ID, and object type ID.

If an ID pattern is specified for the object type, the ID can be excluded. Also, a default parent and object type can be specified using the products tag attributes. The following XML would create a new product:

```
<?xml version="1.0" encoding="utf-8"?>
<STEP-ProductInformation ContextID="Context1" WorkspaceID="Main" >
<Products DefaultParentID="SalesItemFamily-12456" DefaultUserTypeID="SalesItem">
<Product/>
</Products>
</STEP-ProductInformation>
```

Update and Delete Objects

The STEP ID is required to update an existing object. However, for products, entities, classifications, and assets, a key can also be used to identify the objects. For more information about keys, see the **About Unique Keys** section of the **System Setup / STEP Super User Guide** documentation.

As with the tabular format imports, it is only necessary to include the information to be updated. By default, existing data that is not included in the import file will not be modified.

Deleting Values and References / Links

To delete an existing value, include the appropriate value element and leave the value part blank. The example below updates the product with ID 'L6576' by setting the value for 'Horizontal Pixels' to 1920 and removing any existing value for 'Manufacturer's Description'.

```
<?xml version="1.0" encoding="utf-8"?>
<STEP-ProductInformation ContextID="Context1" WorkspaceID="Main">
<Products>
<Product ID="L6576">
<Name>Panasonic Viera TC-P65VT50</Name>
<Values>
<Value AttributeID="Manufacturer's Description"/>
<Value AttributeID="Horizontal Pixels">1920</Value>
</Values>
</Product>
```

```
</Products>
</STEP-ProductInformation>
```

The same functionality cannot be used to remove references and links. Instead, use the applicable entry setting in the **STEP-ProductInformation** tag to indicate that existing references or links should be replaced. To specify the type, use the Replacement Rules section.

For example, the XML below removes all existing product-to-attribute links (using `ReplaceAttributeLinks="Y"` in the `STEP-ProductInformation` tag) from the product with ID 'UNSPSC000.52161505', so only the links in the file are present in STEP after the import. Additionally, for the same product, all existing asset references of the type 'Guides' are removed (using `<ReplaceProductToAssetCrossReference ReferenceTypeID="Guides"/>` in the ReplacementRules section). After the import, only the references in the file remain.

```
<?xml version="1.0" encoding="utf-8"?>
<STEP-ProductInformation ContextID="Context1" WorkspaceID="Main" ReplaceAttributeLinks="Y">
  <ReplacementRules>
    <ReplaceProductToAssetCrossReference ReferenceTypeID="Guides"/>
  </ReplacementRules>
  <Products>
    <Product ID="UNSPSC000.52161505">
      <Name>Televisions</Name>
      <AssetCrossReference AssetID="Mounting Guide" Type="Guides"/>
      <AttributeLink AttributeID="Aspect Ratio"/>
      <AttributeLink AttributeID="Diagonal Screen Size"/>
    </Product>
  </Products>
</STEP-ProductInformation>
```

STEP-ProductInformation Element Attributes

Some STEP-ProductInformation element attributes can be used to remove existing values that are not in the import file.

Available STEP-ProductInformation Replace Instructions

- ReplaceAttributeLinks
- ReplaceClassificationToClassificationCrossReferences
- ReplaceClassificationValues
- ReplaceOverrideToSubProductReferences
- ReplaceAttributeLinks
- ReplaceProductToClassificationReferences
- ReplaceProductToProductCrossReferences
- ReplaceProductValues

- ReplaceTerms
- ReplaceValidUserTypesForAttributes

Available Replacement Rules Elements

- ReplaceClassificationReferenceType
- ReplaceCrossReferenceType
- ReplaceProductToAssetCrossReferenceType
- ReplaceProductToProductCrossReferenceType
- ReplaceProductValuesForAttributeGroupType

Limitations

The following update cases cannot be handled by STEPXML:

- Removing or adding individual values for multivalued attributes
- Removing the legal source and target object types for reference / link type definitions
- Removing legal units for attributes

Deleting Objects

The following delete cases can be handled by STEPXML:

- Products
- Entities
- Classifications
- Assets
- Attributes
- LOV's
- Units

Delete actions vary based on the object type, as follows:

- For workspace revisable object types, delete moves the object to the Recycle Bin.
- For attributes, LOVs, units, and Global Revisable entities, delete purges the object.

The following XML moves four product objects to the Recycle Bin.

```
<?xml version="1.0" encoding="utf-8"?>
<STEP-ProductInformation ContextID="Context1" WorkspaceID="Main">
<DeleteProducts>
<DeleteProduct ID="L7629"/>
<DeleteProduct ID="L7321"/>
<DeleteProduct ID="L7847"/>
<DeleteProduct ID="L7623"/>
```

</DeleteProducts>

</STEP-ProductInformation>

UNSPSC Format

UNSPSC (United Nations Standard Products and Services Code®) is a global classification framework for products and services in all industry sectors. For detailed information, go to www.unspsc.org.

- 1. If the system did not detect the file format automatically, make that selection now.
- 2. In the **ID prefix** field, enter the prefix you want to add to the IDs that are imported. The default setting is UNSPSC000.
- 3. In the **Class Name** field, enter a name for the root node of the classification. The default setting is UNSPSC.
- 4. In the **Object Type** field, enter the object type of the root object. The default setting is UNSPSC.

Import Manager

Steps

- 1. Select Configuration
- 2. Select Data Source
- 3. Select Format**
- 4. Map Data
- 5. Identify Objects
- 6. Identify Destination
- 7. Select Business Rules
- 8. Advanced Settings

Select Format

Format: UNSPSC

Converts from the UNSPSC Excel format found on www.unspsc.org

ID Prefix: UNSPSC000.

Class Name: UNSPSC

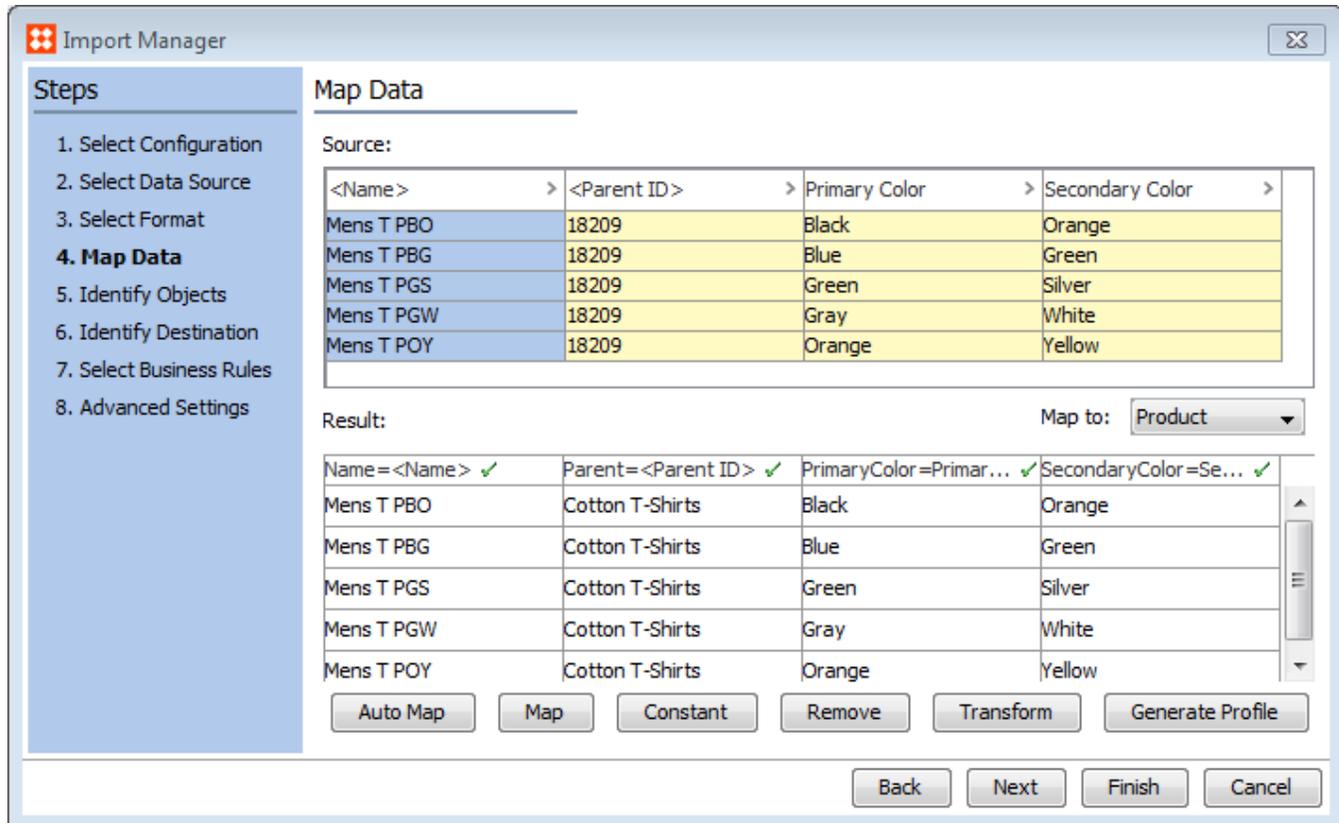
Object Type: UNSPSC

Conversion Preview:

Back Next Finish Cancel

Step 4 - Map Data

Mapping is required when loading an Excel or CSV file. Mapping relates the import data to the appropriate element in the STEP data model and specifies the desired object type.



Mapping can be accomplished using one of these three ways:

1. Map manually - select an import column and use the **Map** button to relate a STEP object. For more information, see the **Map Data Manually** section of the **Importing Data and Images** documentation.
2. Auto Map - click the **Auto Map** button, STEP reads the header row of the import file, and relates all matches to corresponding objects in STEP. For more information, see the **Auto Mapping Data** section of the **Importing Data and Images** documentation.
3. Auto Map and Map manually - after auto mapping an import file with a header row, manually update or correct the results.

Once mapped, source columns are shown with an yellow background. The result section displays a green check for a data column that has been validated for the object type.

About CSV and Excel Files

- Not every column of the input file has to be mapped. For example, if there are 5 pieces of data to load, but the import file has 20 columns, only map the 5 relevant columns. Mapped columns must correspond to a

selected STEP object. Columns that are not mapped are not loaded into STEP.

- Each row of an input file results in one object created or updated. STEP attempts to import every row of the file. Ensure your file contains only the objects that should be created or updated in STEP and remove any unwanted rows before importing.
- Every node and level within the Primary Product Hierarchy is regarded as being a STEP product. The term 'product' includes product families, subcategories, minor categories, major categories, and so on.
- Import file columns can be in any order. Mapping allows you to automatically match data based on the name or ID of an existing attribute, or you can map columns manually.
- Units are not required in the import file for values for STEP attributes with a default unit. For example, when loading values for the attribute Maximum Operating Temperature with the default unit of °F, there is no need to include the °F in the load file if they are all specified in Fahrenheit. The system imports the value and assigns the °F automatically.
- All mapped columns must be mapped against only one object type (product, asset, classification, entity, or attribute). If the file contains a more than one object type, you must import the same file several times, each time choosing the next object type. This procedure is often used when building a new product hierarchy.

Mapping Data Manually

When an Excel or CSV file does not contain a header row, you must map the data manually. Additionally, manual mapping can be used to modify the mapping generated with the Auto Map feature.

For information about automatic mapping, see the **Auto Mapping Data** section of the **Importing Data and Images** documentation.

IDs, Names, and Parent IDs

- IDs must be unique and can be up to 40 characters long. Once saved in STEP, an ID can never be changed.
- Names can be up to 80 characters long and should be used to properly identify the object (not as a description). Avoid using special characters in the name. Unique names are not required, but do aid in identifying data.
- When the object is a family, subcategory, or major category, etc., common setup is to use a number for the ID and use the actual category name as the name.
- Objects, parent objects, and referenced objects can also be identified using keys. Keys are automatically generated when the underlying attributes are mapped during import.

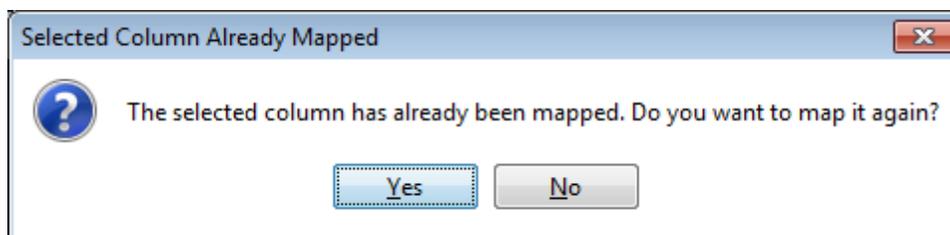
Changing Object Types

You can change the object type of products listed in one column of the input file to the object type listed in another column of the input file. You can only change products to valid object types.

Mapping a Column Multiple Times

Source columns can be mapped as many times as needed. For example, if an import column contains three dimensions for height, length, and width, you might want to map it three times so that the elements can be saved to three separate STEP attributes.

When you attempt to map a column that is already mapped, a warning displays. Click **Yes** to map the same column again, or click **No** and select a different column.



Delete Values

When updating existing objects, the import file must use the case-sensitive text [delete], as shown below, to remove data for already populated attributes. Then map the column to the appropriate STEP object. If the value being deleted was inherited, the result is not a blank field, but the inherited value is restored.

	A	B
1	<ID>	Primary Color
2	179915	[delete]

Dimension Dependent Data

When mapping data from a file to an attribute that is dependent on a single dimension (for example, language or country), transformations make it possible to specify which Dimension Point is required. This allows values for multiple languages or countries to be imported from a single file.

For information on using transformations, see the **Transformations Overview** and **Transformation Examples** sections of the **Importing Data and Assets** documentation.

Mapping Based on Data Type

See the following topics for examples on manually mapping different kinds of data:

- Mapping the ID Column
- Mapping the Object Name Column
- Mapping the Parent Column
- Mapping an Attribute Column
- Mapping a Constant Value
- Removing a Mapped Column

Mapping the ID Column

The STEP ID is the most common way to uniquely identify an object. It is required for updating values for existing objects. When creating new objects, and a STEP ID is not be included in the input file, you can match an existing object using its key instead. See the following steps for mapping the Object ID or the Key.

Mapping Object ID

When STEP is configured to generate IDs automatically, the import can also generate object IDs when they do not exist in the import file. In this case, a blank column in the import file will be mapped. When your object ID is an actual part number, do not use automatically generated IDs since it can cause confusion when identifying data. When updating existing products, you must specify the object ID as it exists in STEP.

1. In the Source section, select the column that contains the object ID.

Map Data

Source:

<ID>	<Name>	<Parent ID>	Primary Color	Secondary Color
MT18400	Mens T PBO	18209	Black	Orange
MT18401	Mens T PBG	18209	Blue	Green
MT18402	Mens T PGS	18209	Green	Silver
MT18403	Mens T PGW	18209	Gray	White
MT18404	Mens T POY	18209	Orange	Yellow

2. From the **Map to** dropdown, choose the option for the type of data being imported, and then click the **Map** button. In this example, products are being imported.

Map to: Product

- Product
- Asset
- Classification
- Entity
- Attribute

3. In the **Map To** window, select **ID**.

Map <ID> to

ID
 Name
 Attribute
 Product Classification Link

ID Aspect: ID

- Click **OK** and the **Result** section shows the mapped column with the ID of the objects to be created and/or have their attributes modified in STEP.

Result: Map to: Product ▾

ID=<ID> ✓
MT18400
MT18401
MT18402
MT18403
MT18404

Mapping Keys

If an existing object has a key that is active and has a specified object type, it can be mapped to identify an object during import. For more information, see the **About Unique Keys** section of the **System Setup / Super User Guide** documentation.

- In the Source section, select a column and click the **Map** button.

Map Data

Source:

<ID>	<Name>	<Parent ID>	Primary Color	Secondary Color
MT18400	Mens T PBO	18209	Black	Orange
MT18401	Mens T PBG	18209	Blue	Green
MT18402	Mens T PGS	18209	Green	Silver
MT18403	Mens T PGW	18209	Gray	White
MT18404	Mens T POY	18209	Orange	Yellow

- In the **Map to** window, select a radio button option to map to a key.

Selecting **ID** simply maps the current object via key. The additional options allow the user to map references, links, and parent objects via key. Most of these options require that additional selections be made, such as specifying the reference type to be applied to link the objects.

Valid radio button options include: **ID**, **Product Classification Link**, **Product Reference**, **Asset Reference**, **Classification Reference**, **Entity Reference**, **Reference Meta-Data**, and **Parent**.

3. In the **ID Aspect** drop down, select the appropriate key and then click **OK**.

Note: If no matching object is found (based on keys) and the import configuration allows for the creation of new objects, a new object can be created. To accomplish this, STEP IDs must be omitted from the import, and all attributes that make up the key definition must be mapped and populated on the current object. Additionally, the object type being created must use autogenerated IDs. Failing to provide the underlying key attributes will result in an error.

Mapping the Object Name Column

When updating attribute values on existing objects in STEP, mapping the object name is not required, even if it exists in the input file.

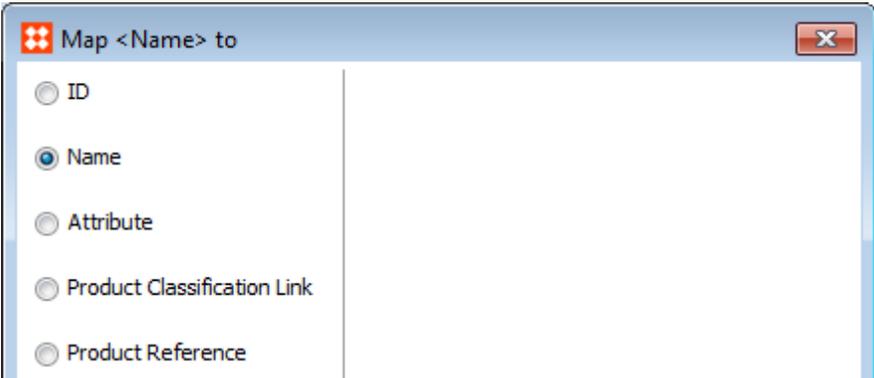
Common setup involves mapping the object name when one of the following conditions is true:

- new objects are being imported into STEP
 - existing names of all objects in the import file should be changed
 - the import file includes object names for a specific context and the objects are dimension dependent. For example, you sell the same product in three different markets, but identify them by a different part number in each market. Although you might typically hold a single product in STEP with a unique ID, in this case, for each market's context, you could enter the appropriate 'part number' into the Name.
1. In the Source section, select the data column to map to the object name, and then click **Map**.

Source:

<ID>	<Name>	<Parent ID>	Primary Color	Secondary Color
MT18400	Mens T PBO	18209	Black	Orange
MT18401	Mens T PBG	18209	Blue	Green
MT18402	Mens T PGS	18209	Green	Silver
MT18403	Mens T PGW	18209	Gray	White
MT18404	Mens T POY	18209	Orange	Yellow

2. In the **Map to** window, select the **Name** radio button.



3. Check the **Mandatory** option to require that imported objects have a name in the file. If unchecked, all rows are imported, even those without a name.



4. Click **OK** and the **Result** panel displays that the mapped column will specify the Name of the objects in the input file.

Result: Map to:

ID=<ID> ✓	Name=<Name> ✓	
MT18400	Mens T PBO	
MT18401	Mens T PBG	
MT18402	Mens T PGS	
MT18403	Mens T PGW	
MT18404	Mens T POY	

Mapping the Parent Column

The parent column indicates where to locate, or relocate, the imported object(s). When updating attribute values for existing objects, mapping the parent column is not required, even if it exists in the input file. Common setup involves mapping the parent column when one of the following conditions is true:

- new objects are being imported into STEP
 - existing objects within the hierarchy need to be relocated
1. In the Source section, select the column of data that includes valid STEP IDs for parents, and click the **Map** button.

Source:

<ID>	<Name>	<Parent ID>	Primary Color	Secondary Color
MT 18400	Mens T PBO	18209	Black	Orange
MT 18401	Mens T PBG	18209	Blue	Green
MT 18402	Mens T PGS	18209	Green	Silver
MT 18403	Mens T PGW	18209	Gray	White
MT 18404	Mens T POY	18209	Orange	Yellow

2. In the **Map to** window, select the **Parent** radio button. To use a Key, instead of the STEP ID, select the appropriate option from the Parent ID Aspect dropdown.

The screenshot shows the 'Map to' window with the following elements:

- Radio button options on the left:
 - Entity Reference
 - Reference Meta-Data
 - Parent
 - Object Type
 - Variable
 - Multivalued Variable
 - Overrides product
- 'Parent ID Aspect' dropdown menu on the right, showing:
 - Parent ID (selected)
 - Parent ID
 - Key: Key 2
 - Key: Supplier Part Number
 - Key: Key 3

3. Check the **Mandatory** option to require that imported objects have a parent in the file. If unchecked, all rows are imported, even those without a parent specified.

The screenshot shows the 'Map to' window with the following elements:

- Radio button option on the left:
 - Add child to override
- 'Mandatory' checkbox on the right, which is checked:
 - Mandatory
- 'Cancel' and 'OK' buttons at the bottom right.

4. Click **OK** and the **Result** panel displays your selection.

Result: Map to: Product ▼

ID=<ID> ✓	Name=<Name> ✓	Parent=<Parent ID> ✓	
MT 18400	Mens T PBO	Cotton T-Shirts	
MT 18401	Mens T PBG	Cotton T-Shirts	
MT 18402	Mens T PGS	Cotton T-Shirts	
MT 18403	Mens T PGW	Cotton T-Shirts	
MT 18404	Mens T POY	Cotton T-Shirts	

Note: If valid STEP IDs were not found in the first 200 rows of the file, the Result panel shows that the object is not found in STEP.

Mapping an Attribute Column

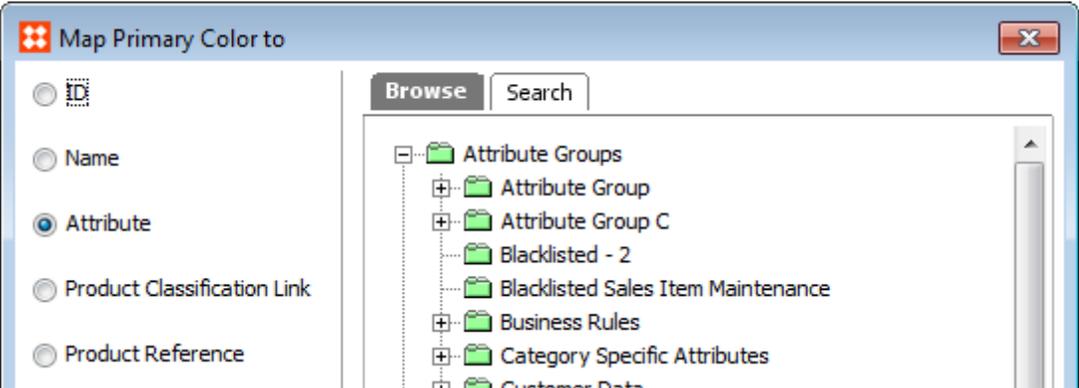
You can map a data column in your input file to an attribute that already exists in STEP.

- 1. In the Source section, select the column of data that you want to map to an attribute and click the **Map** button.

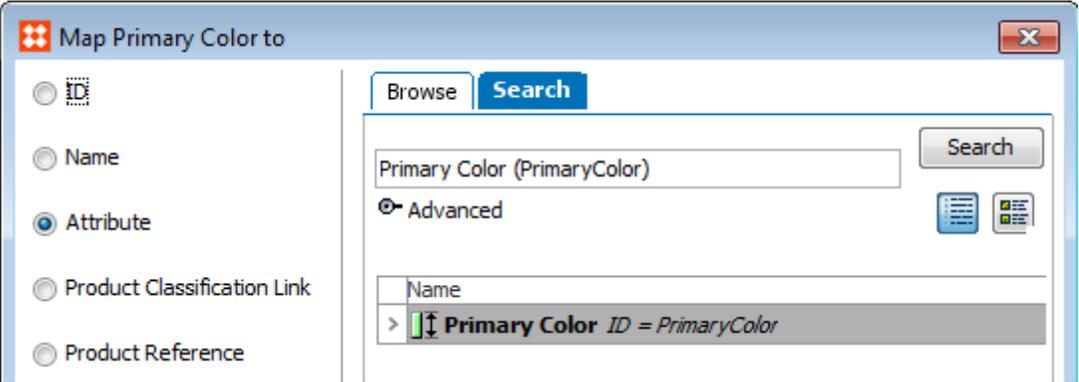
Source:

<ID>	<Name>	<Parent ID>	Primary Color	Secondary Color
MT18400	Mens T PBO	18209	Black	Orange
MT18401	Mens T PBG	18209	Blue	Green
MT18402	Mens T PGS	18209	Green	Silver
MT18403	Mens T PGW	18209	Gray	White
MT18404	Mens T POY	18209	Orange	Yellow

- 2. In the **Map to** window, select the **Attribute** radio button.



- 3. In the displayed attribute selector, use Browse or Search to select the STEP attribute to be mapped.



Search is available on all or part of the attribute’s name or ID. Click the advanced search flipper to use other options, such as searching for attributes with a specific value.

4. Check the **Mandatory** option to require that imported objects have a value in the file. If unchecked, all rows are imported, even those without a value.

Add child to override

Mandatory

Cancel OK

5. Click **OK** and the **Result** panel displays your selection of column and attribute.

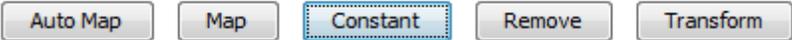
Result: Map to: Product

ID=<ID> ✓	Name=<Name> ✓	Parent=<Parent ID> ✓	PrimaryColor=Primary Color ✓
MT 18400	Mens T PBO	Cotton T-Shirts	Black
MT 18401	Mens T PBG	Cotton T-Shirts	Blue
MT 18402	Mens T PGS	Cotton T-Shirts	Green
MT 18403	Mens T PGW	Cotton T-Shirts	Gray
MT 18404	Mens T POY	Cotton T-Shirts	Orange

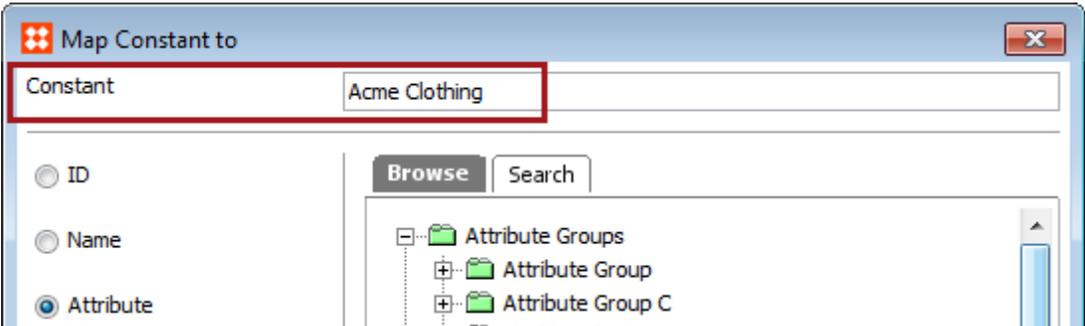
Mapping a Constant Value

You can add an attribute value to all records by mapping a constant value. Common setup uses this feature for filling some object attributes.

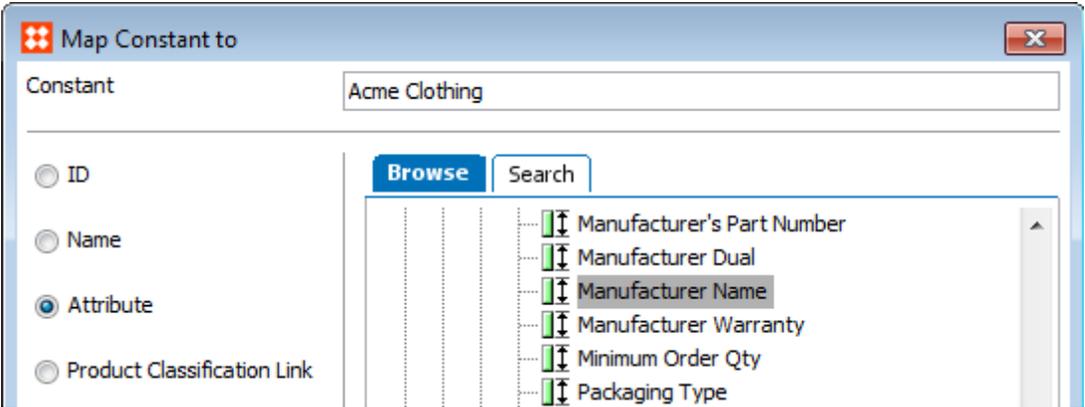
1. Below the **Result** panel, click the **Constant** button.



2. On the **Map Constant To** dialog, in the **Constant** field, enter text that should be assigned to all records.



3. Select the **Attribute** radio button and use **Browse** or **Search** to select the attribute to be modified.



4. The **Result** panel displays the constant value as if it exists in the input file.

Result: Map to: Product

ID=<ID> ✓	Name=<Name> ✓	Parent=<Parent ID> ✓	PrimaryColor=Primary ... ✓	ManufacturerName=A... ✓
MT18400	Mens T PBO	Cotton T-Shirts	Black	Acme Clothing
MT18401	Mens T PBG	Cotton T-Shirts	Blue	Acme Clothing
MT18402	Mens T PGS	Cotton T-Shirts	Green	Acme Clothing
MT18403	Mens T PGW	Cotton T-Shirts	Gray	Acme Clothing
MT18404	Mens T POY	Cotton T-Shirts	Orange	Acme Clothing

Removing a Mapped Column

If the Result panel displays a column that you do not want to import, you can remove it.

- 1. In the **Result** panel, select the mapped column to remove.

Result: Map to: Product

ID=<ID> ✓	Name=<Name> ✓	Parent=<Parent ID> ✓	PrimaryColor=Primary Color ✓	ManufacturerName=Acme Clot... ✓	SecondaryColor=Primary Color ✓
MT18400	Mens T PBO	Cotton T-Shirts	Black	Acme Clothing	Black
MT18401	Mens T PBG	Cotton T-Shirts	Blue	Acme Clothing	Blue
MT18402	Mens T PGS	Cotton T-Shirts	Green	Acme Clothing	Green
MT18403	Mens T PGW	Cotton T-Shirts	Gray	Acme Clothing	Gray
MT18404	Mens T POY	Cotton T-Shirts	Orange	Acme Clothing	Orange

- 2. Click the **Remove** button.



- 3. The **Result** panel no longer displays the column, and it will not be imported.

Result: Map to: Product

ID=<ID> ✓	Name=<Name> ✓	Parent=<Parent ID> ✓	PrimaryColor=Primary Color ✓	ManufacturerName=Acme ... ✓
MT18400	Mens T PBO	Cotton T-Shirts	Black	Acme Clothing
MT18401	Mens T PBG	Cotton T-Shirts	Blue	Acme Clothing
MT18402	Mens T PGS	Cotton T-Shirts	Green	Acme Clothing
MT18403	Mens T PGW	Cotton T-Shirts	Gray	Acme Clothing
MT18404	Mens T POY	Cotton T-Shirts	Orange	Acme Clothing

Additional Column Mapping Options

In addition to the mapping of ID, names, attributes, parents and object types, there are a number of other object, asset, classification, and entity column mapping options.

Before you create additional mappings, note the following:

- If you want to link objects to a classification, verify that the object and classification exists in STEP. You must also verify that you have specified the classification ID in the input file, and not the name.
- When making links from objects to classifications or to other objects or assets, you must map the product ID in the input file.
- When making product-to-product references, you must have at least two columns in your input file. One with the ID of the source product, and the other with the ID of the target product. When you map these columns, you can specify the reference type that should be used to link the products.
- When making product-to-product references, you cannot specify the product reference type in the input file. You must select the product reference type in the wizard screen. In a single load operation, you can only make one type of product-to-product reference. You can, however, load the same file again and specify a different product reference type.
- When making product-to-asset references, you cannot specify the asset reference type in the input file. You must select the asset reference type in the wizard screen. You can load the same file again and specify a different asset reference type.
- For product-to-product and product-to-asset references, the list of reference types listed in the wizard is the list of existing reference types. You cannot specify or create new reference types at this point.
- It is common setup to only change the object type of a list of objects when those objects are all at the same level in the STEP hierarchy.

Note: The mapping options Variable and Multivalued Variable are documented in the **Using Transformations to Improve Data Quality** section of the **Data Quality User Guide**.

Product	Classification	Entities	Assets	Attributes
<input type="radio"/> ID <input type="radio"/> Name <input type="radio"/> Attribute <input type="radio"/> Product Classification Link <input checked="" type="radio"/> Product Reference <input type="radio"/> Asset Reference <input type="radio"/> Entity Reference <input type="radio"/> Reference Meta-Data <input type="radio"/> Parent <input type="radio"/> Object Type <input type="radio"/> Variable <input type="radio"/> Multivalued Variable	<input checked="" type="radio"/> ID <input type="radio"/> Name <input type="radio"/> Attribute <input type="radio"/> Product Reference <input type="radio"/> Asset Reference <input type="radio"/> Classification Reference <input type="radio"/> Entity Reference <input type="radio"/> Reference Meta-Data <input type="radio"/> Parent <input type="radio"/> Object Type <input type="radio"/> Variable <input type="radio"/> Multivalued Variable	<input checked="" type="radio"/> ID <input type="radio"/> Name <input type="radio"/> Attribute <input type="radio"/> Product Reference <input type="radio"/> Asset Reference <input type="radio"/> Classification Reference <input type="radio"/> Entity Reference <input type="radio"/> Context Reference <input type="radio"/> Reference Meta-Data <input type="radio"/> Parent <input type="radio"/> Object Type <input type="radio"/> Variable <input type="radio"/> Multivalued Variable	<input checked="" type="radio"/> ID <input type="radio"/> Name <input type="radio"/> Attribute <input type="radio"/> Entity Reference <input type="radio"/> Parent <input type="radio"/> Object Type <input type="radio"/> Variable <input type="radio"/> Multivalued Variable	<input checked="" type="radio"/> ID <input type="radio"/> Name <input type="radio"/> Attribute <input type="radio"/> Attribute Info <input type="radio"/> Attribute Validation Info <input type="radio"/> Variable <input type="radio"/> Multivalued Variable

Product Classification Link

You can link a product in the input file to one or more classifications in STEP. Ensure that the import file contains IDs of the classifications required for linking to the products. The system automatically links the products using the valid product-to-classification link type. Note that each classification object type can have only one product-to-classification link type assigned to it. Therefore, the reference type is not selected during configuration as the system identifies the classifications by the provided IDs and applies the valid link type (of which only one will be available).

Product Reference

You can reference products, classifications, or entities listed in one column of the input file to another column of products in that file, and make a product reference between them in STEP. When you make this selection, you must specify which reference type to use. The available reference types are based on the product reference types that exist in STEP.

Asset Reference

You can reference products, classifications, or entities listed in one column of the input file to assets listed in another column of the input file, and make a reference between them in STEP. When you make this selection, you must specify which reference type to use. The available reference types are based on the asset reference types that exist in STEP.

Entity Reference

You can reference products, classifications, or entities listed in one column to entities listed in another column in the input file, and make a reference between them in STEP. When you make this selection, you must specify which reference type to use. The available reference types are based on the entity reference types that exist in

STEP.

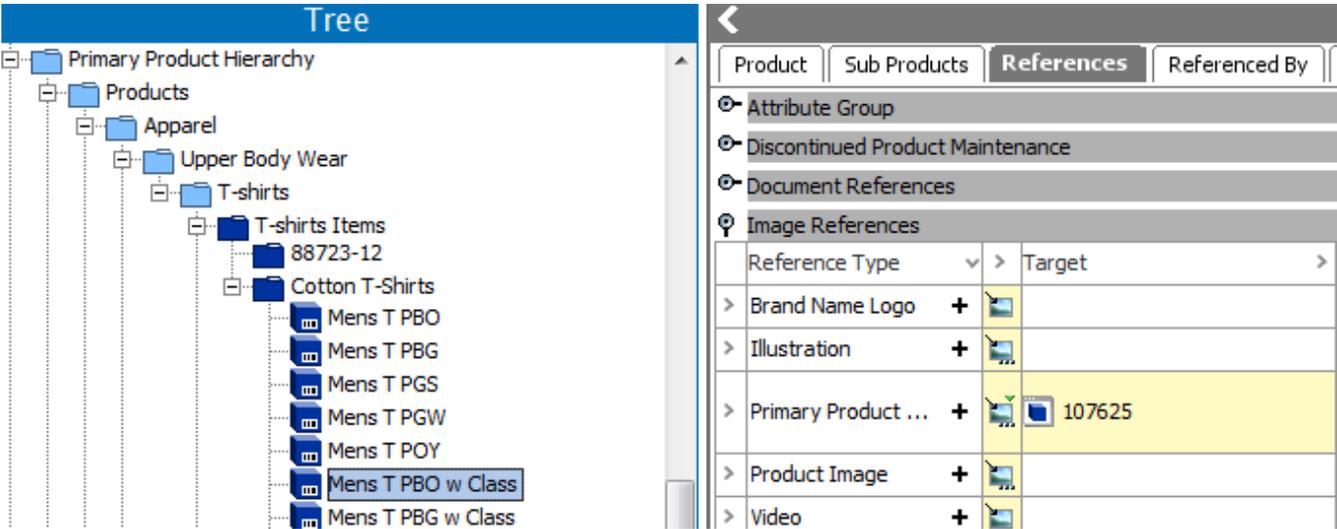
Classification Reference

You can reference classifications or entities listed in one column of the input file to classifications listed in another column of the input file, and make a reference between them in STEP. When you make this selection, you have to specify which reference type to use. The available reference types are based on the classification reference types that exist in STEP.

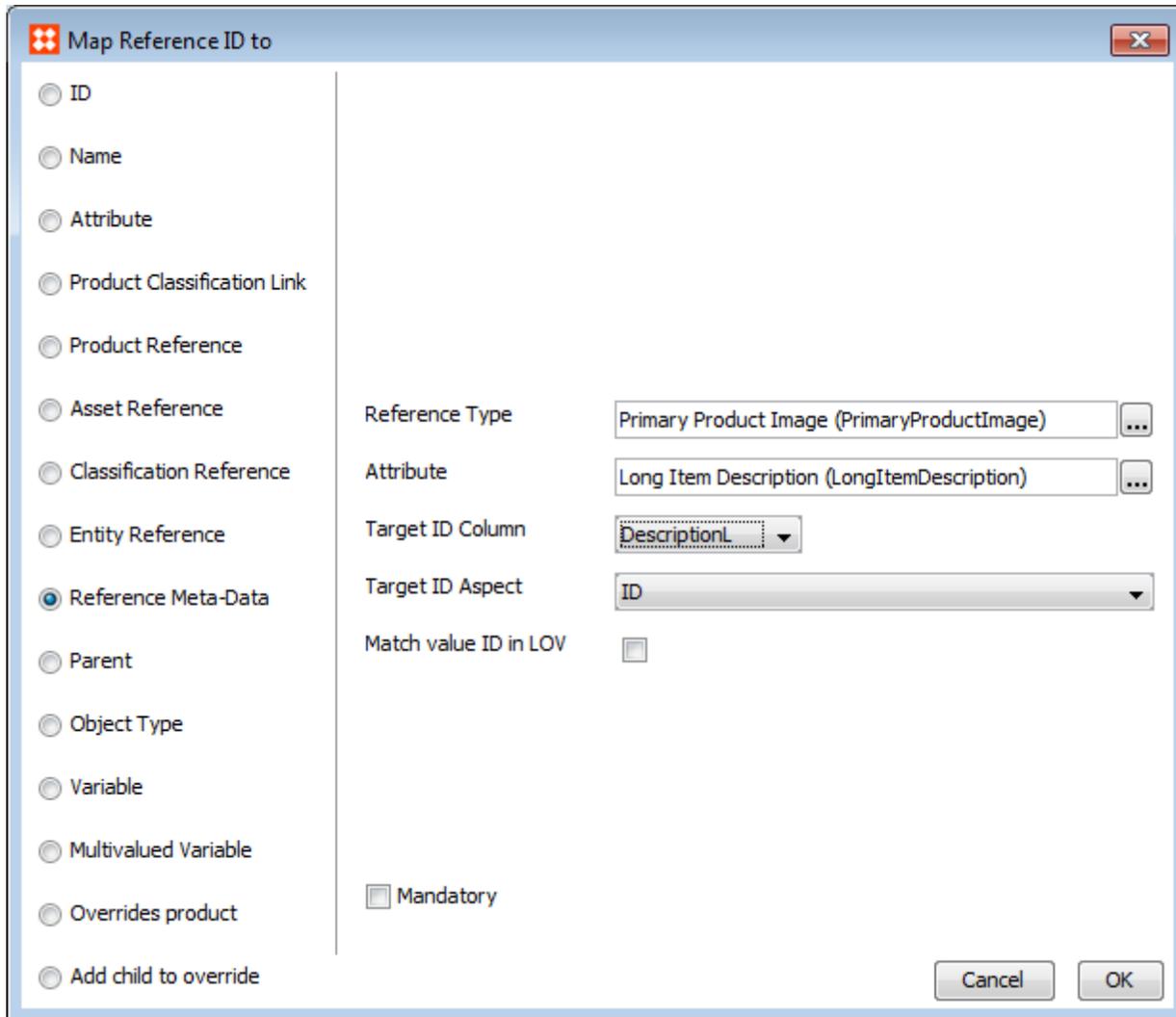
Reference Meta-Data

You can import values into meta attributes on references or links. You can import meta attribute values into all types of reference types or link types.

For example, the reference tab on a product has been selected.



To import meta attribute values into the attribute Description, your import file must contain a column with meta attribute values and a column with IDs of referenced objects. In this example, you need one column holding the Description values and another column holding the referenced target IDs.



- In **Reference Meta-Data** the reference type 'Primary Product Image' and the attribute 'Long Item Description' is selected. In the Excel sheet, the IDs of the referenced objects appear. The target ID Column is mapped to ReferenceID in the Excel sheet.
- In **Attribute Info** you can map to attribute details (multivalued, mandatory, externally maintained, attribute group links, object type links, etc.) for when importing attribute definitions.
- In **Attribute Validation Info** you can map to attribute validation details (base validation type, maximum length of values, usage of List Of Values, legal units, etc.).
- **Match value ID in LOV**: If the meta attribute holds a hard domain LOV with value IDs, the attribute column in the Excel sheet can hold the value IDs. The IDs are matched against the IDs in the LOV and values are applied based on the match.
- If you check **Mandatory**, an Excel row is skipped if a meta attribute value does not exist in the Excel sheet.

Additionally, the following attribute data is available for mapping Reference Meta-Data:

- <Attribute Type>
- <Attribute Multi Valued>
- <Attribute FullText>
- <Attribute Externally Maintained>
- <Attribute Calculated>
- <Attribute Calculated Value Template>
- <Attribute Calculated Unit Template>
- <Attribute Mandatory>
- <Attribute Completeness>
- <Attribute Default Unit ID>
- <Attribute Dimension Ref ID>
- <Attribute Object Type Ref ID>
- <Attribute Unit Ref ID>
- <Attribute Validation Base Type>
- <Attribute Validation Minimum value>
- <Attribute Validation Maximum value>
- <Attribute Validation Maximum length>
- <Attribute Validation Input mask>
- <Attribute Validation LOV ID>
- <AttributeGroup Ref ID>

Auto Mapping Data

Auto Map compares data in the input file's header row with the names of the attributes in STEP. When the Import Manager finds a match, the column is automatically mapped to the corresponding object attributes. The Auto map feature only works when the input file has a header row. You can add a header row if one does not exist, otherwise, you can manually map the data.

Predefined Header Text

Auto Map recognizes the following **case-sensitive** header text to indicate these special columns:

- <ID>
- <Name>
- <Parent ID>
- <Object Type ID>

The Import Manager attempts to auto map all other columns in the input file using Name or ID of existing attributes. Both successful and unsuccessful mappings are shown in the Result panel. If an auto mapping is unsuccessful, you can map that column manually.

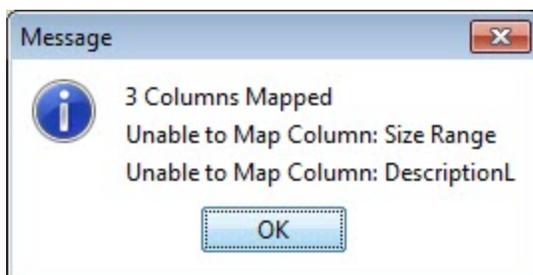
Using Auto Map

The import file includes following columns:

Source:

<Name>	<Parent ID>	Primary Color	Size Range	DescriptionL
Mens T PBO	18209	Black	XS - XXXL	Comfortable fitting 100% cotton tee.
Mens T PBG	18209	Blue	S - XL	Comfortable fitting 100% cotton tee.
Mens T PGS	18209	Green	XS - XXXL	Comfortable fitting tee of cotton/poly blend.
Mens T PGW	18209	Gray	S - XL	Comfortable fitting tee of cotton/poly blend.
Mens T POY	18209	Orange	S - XL	Comfortable fitting 100% cotton tee.

1. In the Import Manager > Map Data step, click the **Auto Map** button. After matching header text to the attributes in STEP, a message displays with the results.



2. Columns that are mapped successfully are displayed with a yellow background in the Source panel. Columns that are not mapped have a white background. The 'Unable to Map' error indicates that no attribute exists in

STEP with the name. Click **OK**.

Source:

<Name>	>	<Parent ID>	>	Primary Color	>	Size Range	>	DescriptionL	>
Mens T PBO		18209		Black		XS - XXXL		Comfortable fitting 100% cotton tee.	
Mens T PBG		18209		Blue		S - XL		Comfortable fitting 100% cotton tee.	
Mens T PGS		18209		Green		XS - XXXL		Comfortable fitting tee of cotton/poly blend.	
Mens T PGW		18209		Gray		S - XL		Comfortable fitting tee of cotton/poly blend.	
Mens T POY		18209		Orange		S - XL		Comfortable fitting 100% cotton tee.	

Result:

Map to:

Name = <Name> ✓	Parent = <Parent ID> ✓	PrimaryColor = Primary Color ✓
Mens T PBO	Cotton T-Shirts	Black
Mens T PBG	Cotton T-Shirts	Blue
Mens T PGS	Cotton T-Shirts	Green
Mens T PGW	Cotton T-Shirts	Gray
Mens T POY	Cotton T-Shirts	Orange

3. Review the **Result** panel. Make any adjustments needed.

If your system has multiple attributes with the same name, use caution when auto mapping. STEP will find a match, but it may not be the attribute you expected.

4. When the mapping is correct, click **Next**.

Transformations

There are numerous ways to use the transformation parameters to modify the data before it is loaded into STEP. With transformations, you can split one data column into multiple attribute values, append and prepend data to the values, perform search and replace, and so on.

- You can apply multiple transformations to the columns of data, but be careful about the order in which the transformations are applied.
- When setting up transformations that are implemented with a saved configuration, it is recommended that you work with smaller load files first. You can then build up the complexity of your transformations in a step-by-step mode.
- Several of the transformations allow you to manipulate units mm, kg. etc., including manipulations such as removing the unit from the value. Other transformations allow you to extract only the unit from a value. In these cases, the attribute that you are mapping to must be one that is allowed to have units and further, must be one that already has those units assigned to it.
- Some transformations refer to "embedded values" or "embedded numbers". These transformations are only relevant if the attribute you are mapping to is of the type "embedded_number". Otherwise, the transformation cannot function.
- When you use the transformation "Extract characters between two pieces of text in the value", the start and end text string cannot be the same.

Transformation Examples

- Extract and Split Data Example
- Extract and Split Identifiable Data Example
- Extract, Split, and Convert Cryptic Data Example
- Concatenate Column with Unit Data Example
- Dimension Point for Single Dimension Data Example

Available Transformations

The following transformations are available.

Transformations	Description
Append from multi-value source	
Append from source	

Transformations	Description
Append text	Adds the specified text, such as a suffix, to the end of the value.
Change case	Sets values to all upper case, all lower case, or mixed case. Mixed case sets the first letter of the value to upper case and the remainder to lower case.
Change hash function	
Convert fractions to parsable tags	Scans a text string for fractions. If a fraction is found, then the fraction is converted into parsable tags. A text string that contains the fraction 1/2 cm is transformed to: <code><StartNumerator/><bfdin/>1<FractionBar/><StartDenominator/>2<EndFraction/></code> cm You have to setup an output formatting to Indesign for <code><StartNumerator/></code> , <code><bfdin/></code> , <code><FractionBar/></code> , <code><StartDenominator/></code> and <code><EndFraction/></code> to mount the fractions.
Extract characters after a given text in the value	Extracts a part of the value, starting after the occurrence of the specified text string.
Extract characters at position	Extracts a specified number of characters after a specified position in a text string.
Extract characters before a given text in the value	Extracts a specified number of characters after a specified text in a text string.
Extract characters between two pieces of text in the value	Extracts the texts between the specified starting text string and the specified ending text string.
Extract prefix, suffix and value from an embedded number	
Extract unit from an embedded value	

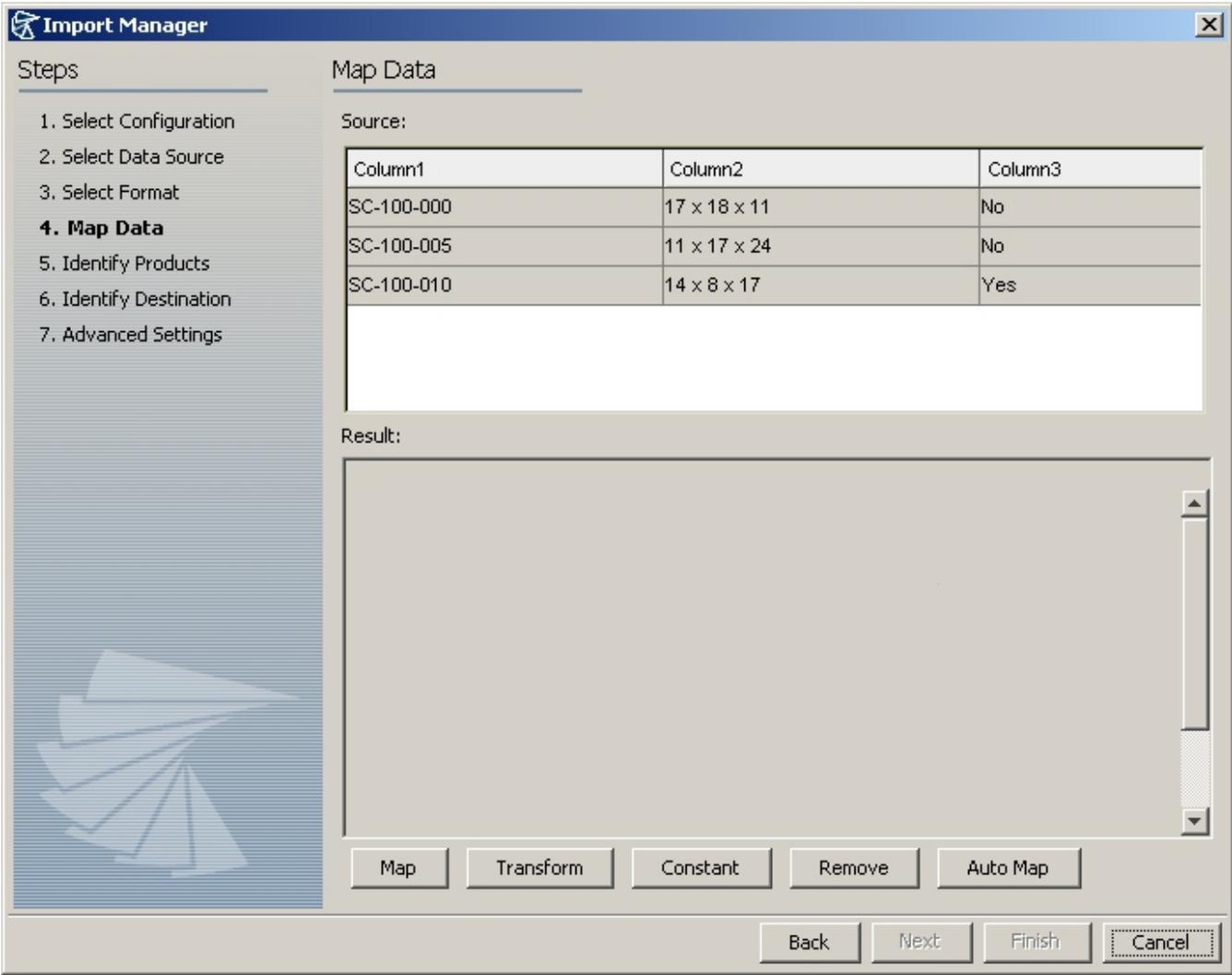
Transformations	Description
Extract unit part of value	
Extracts the last N characters of the value	
Format number	
Generate soundex code	
If equals, source	
Insert text at a given position	
Insert text before	
Math - Add Source	
Math - Divide by source	
Math - Multiply by source	
Math - Subtract source	
Math Operation (+, -, *, /)	Performs the standard math operations multiplication, division, addition, and subtraction on the value.
Parse address	
Prepend from source	

Transformations	Description
Remove unit from value	
Replace substrings of the value	Changes the specified text string with the replacement text string. All matches on the specified text are replaced.
Replace substrings of the value using a regular expression	Replaces sub-strings of text in the value with another string of text. Replaces the text on all matches or on the first match.
Replace the whole value	Replaces the specified value with a different value. You must enter the entire value. Only works on text-based attributes.
Replace the whole value using Lookup Table	
Replace words using Lookup Table	
Split and extract	Extracts data based on a specified search text string and its occurrence. You can, for example extract data from a value after the 4th occurrence of a hyphen, and before the 5th occurrence. So if the value is 179-245-Y-MYGR-TG105P-FDSJ-198, the result is TG105P.
Transform date by locale	<p>Transforms locale-formatted Date, ISO Date, or ISO Date and Time attributes. The locale format must match the context being used in Workbench at the time of import. If not, the 'Transform date by pattern' transformation should be used.</p> <p>Localized dates should follow the DateFormat.SHORT JavaScript formatting style. Times should be in the DateFormat.MEDIUM formatting style.</p>
Transform date by pattern	Transforms patterned dates (e.g., yyyy.MM.dd G 'at' HH:mm:ss z, EE, MMM d, "yy) to Date, ISO Date, or ISO Date and Time. Validation Base Type for the mapped attribute must match the 'Transform to' dropdown choice.
[Transform using attribute transformation]	Extracts data based on an attribute transformation that is already defined in System Setup under Attribute Transformations.

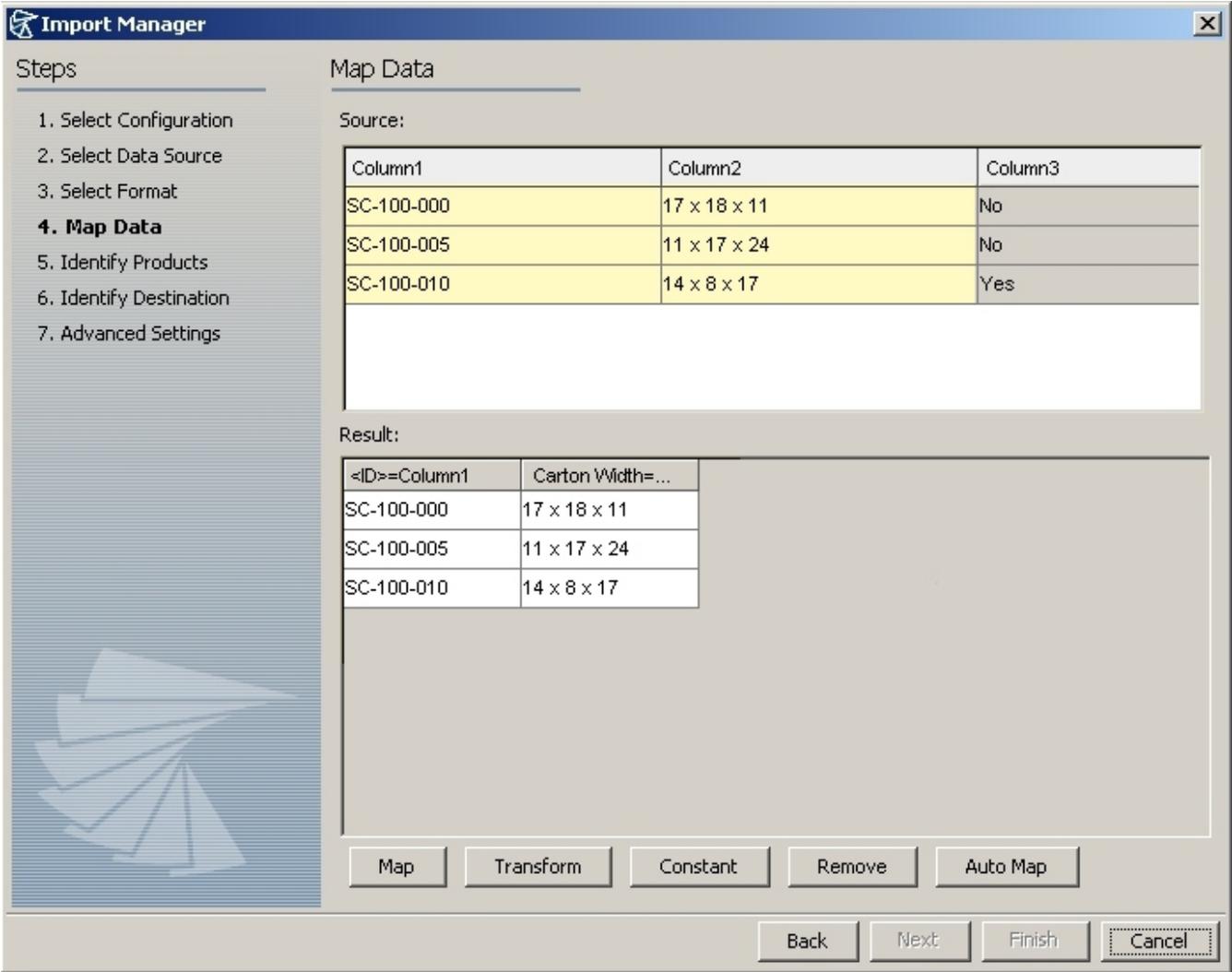
Extract and Split Data Example

In this example, you have a file that has dimensional data in one of the columns. Now, you want to split that dimensional information into three separate values for loading into STEP.

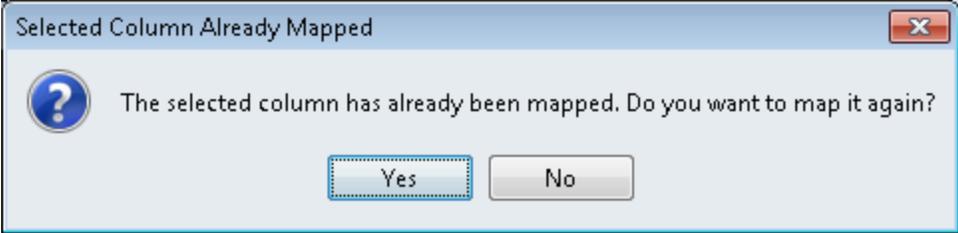
- 1. In Step 3 of the Import Manager, **Map Data**, select the column that holds the dimensional data. Column1 is mapped to the product ID, but Column2 is mapped to 3 separate attributes in the STEP. Therefore, select **Column 2**.



- 2. Manually map the column to the first attribute that will be populated with some part of Column2's data. The following shows that the Column1 was mapped to the ID and that Column2 was mapped to an attribute called "Carton Width".

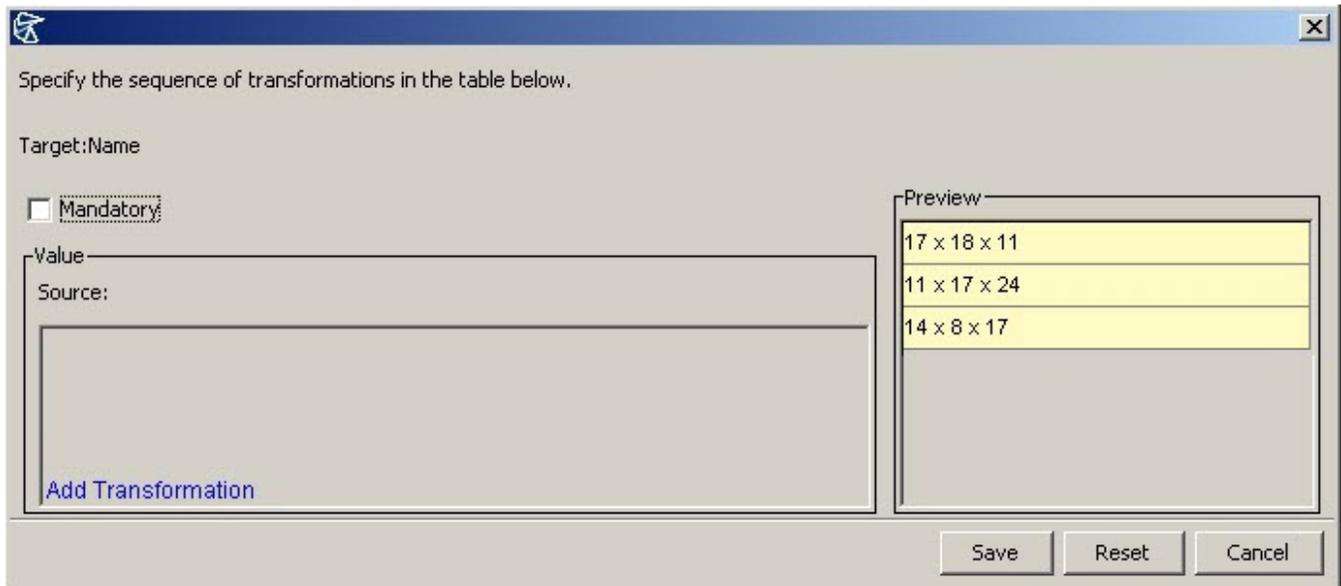


3. Map Column2 again to the next attribute is populated with some part of Column 2's data. And do the same for the third attribute. A warning is displayed saying that the source column has already been mapped and asking if you want to map it again. Click **Yes**.

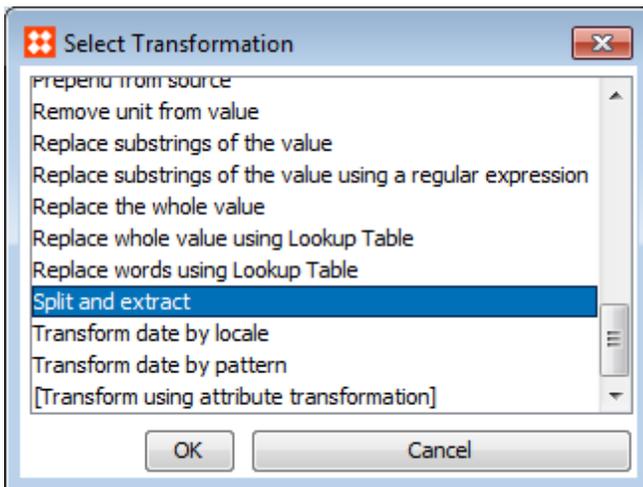


The same column has been mapped to three different attributes: Carton Width, Carton Depth, and Carton Height.

- In the **Results** pane, select the first attribute that you want to perform a transformation on. In this example that is "Carton Width". Click **Transform**. The following window appears.

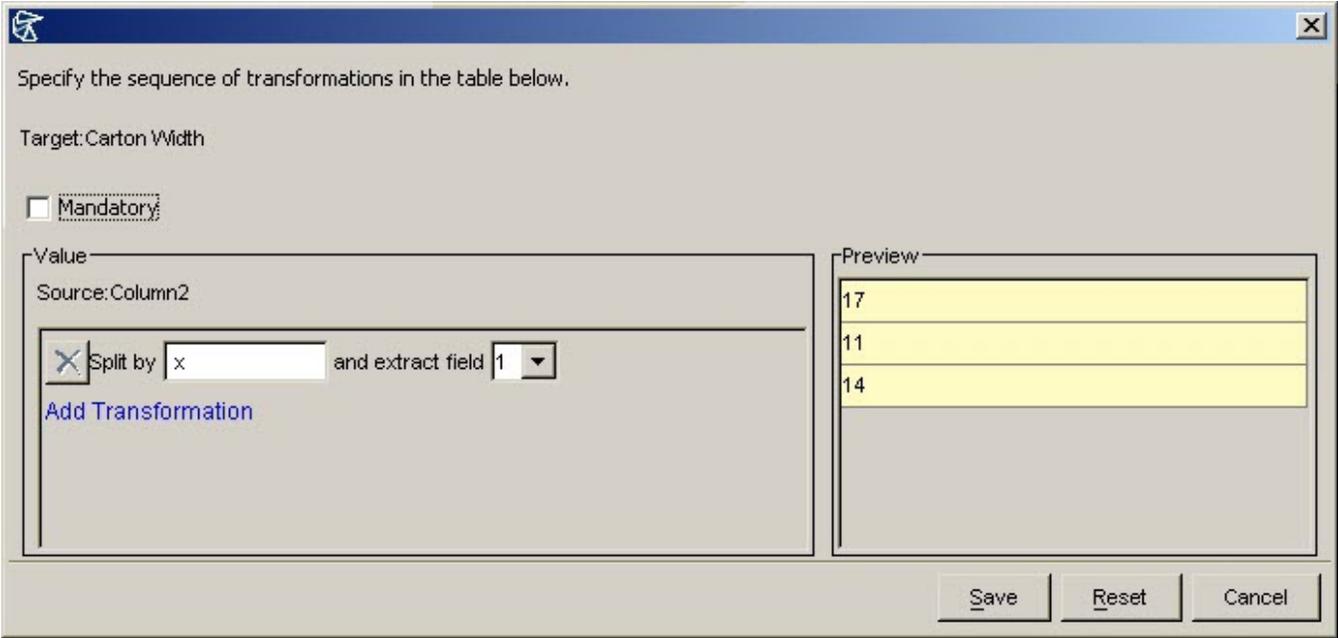


- Click **Add Transformation**. A list of available transformation functions appears.



- Select the **Split and extract** transformation.

In the transformation screen, you specify how the data in the original file's column 2 should be split. In this example, you want to split the data by looking at the information between the "x" (include the spaces), and extracting the data before the first occurrence of this string. Denoted by the extract field = 1. The right panel gives a preview of the data that is extracted. If the preview is not displayed, click F2.



- 7. Click **Save**. In the **Map Data** screen, the column for Carton Width shows a preview of the data that is loaded into STEP.

Import Manager

Steps

1. Select Configuration
2. Select Data Source
3. Select Format
- 4. Map Data**
5. Identify Products
6. Identify Destination
7. Advanced Settings

Map Data

Source:

Column1	Column2	Column3
SC-100-000	17 x 18 x 11	No
SC-100-005	11 x 17 x 24	No
SC-100-010	14 x 8 x 17	Yes

Result:

<ID>=Column1	Carton Width=(...	Carton Depth=...	Carton Height=...
SC-100-000	17	17 x 18 x 11	17 x 18 x 11
SC-100-005	11	11 x 17 x 24	11 x 17 x 24
SC-100-010	14	14 x 8 x 17	14 x 8 x 17

Map Transform Constant Remove Auto Map

Back Next Finish Cancel

8. Repeat the transformation steps for the attributes, 'Carton Depth' and 'Carton Height'.

Extract and Split Identifiable Data Example

This example describes how to extract and clean information from a legacy system and load it into separate attributes in STEP.

Note: The preview displays how the transformations function on a sample of data, typically the first 20 rows. Prior to starting the import, verify that the first 20 objects contain the entries with transformations you want to verify.

Map Data

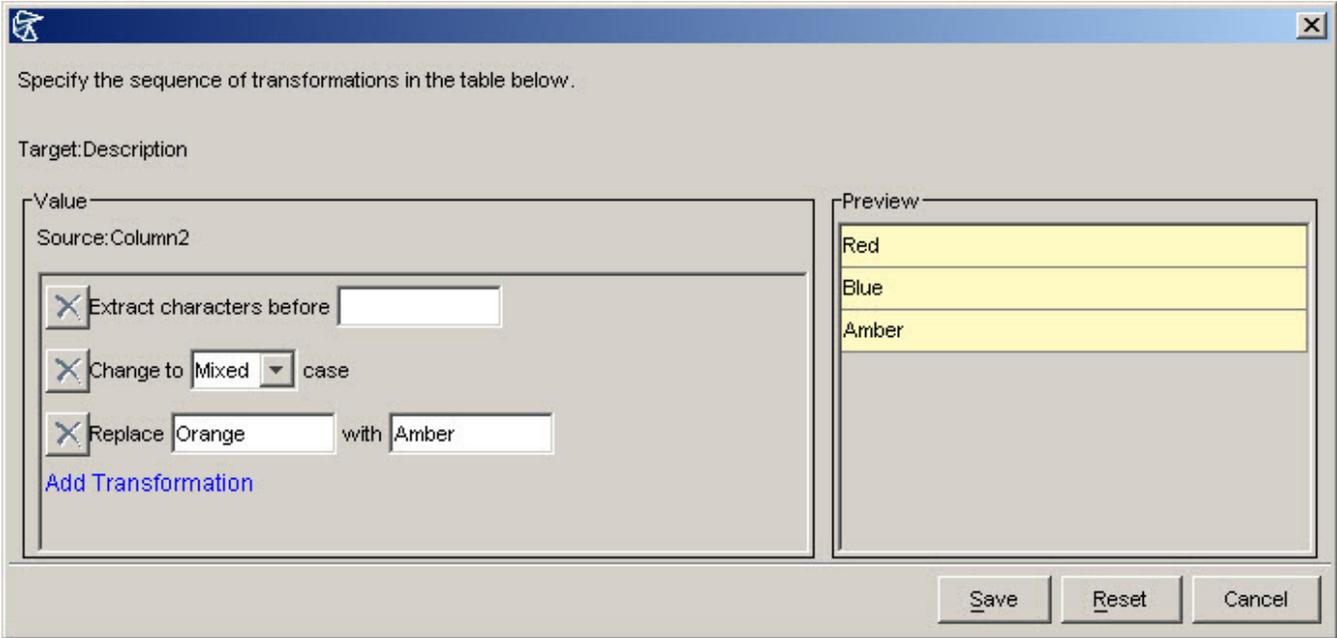
Source:

Column1	Column2	Column3
100-3526	RED 1/2" STRAIGHT WIDGET	UNICORP
100-2625	BLUE 1/4" ANGLED GADGET	MULTICORP
100-2827	ORANGE 3/4" ELL WHATSIT	ANYCORP

A single import column contains four pieces of data that we want to store separately in STEP. We will use transformations to create separate values for attributes: Color, Size, Type, and Description. For the first source row, we want to have Color = Red, Size = 1/2, Type = Straight, and Description = Widget.

First, map the same column to the four different STEP attributes.

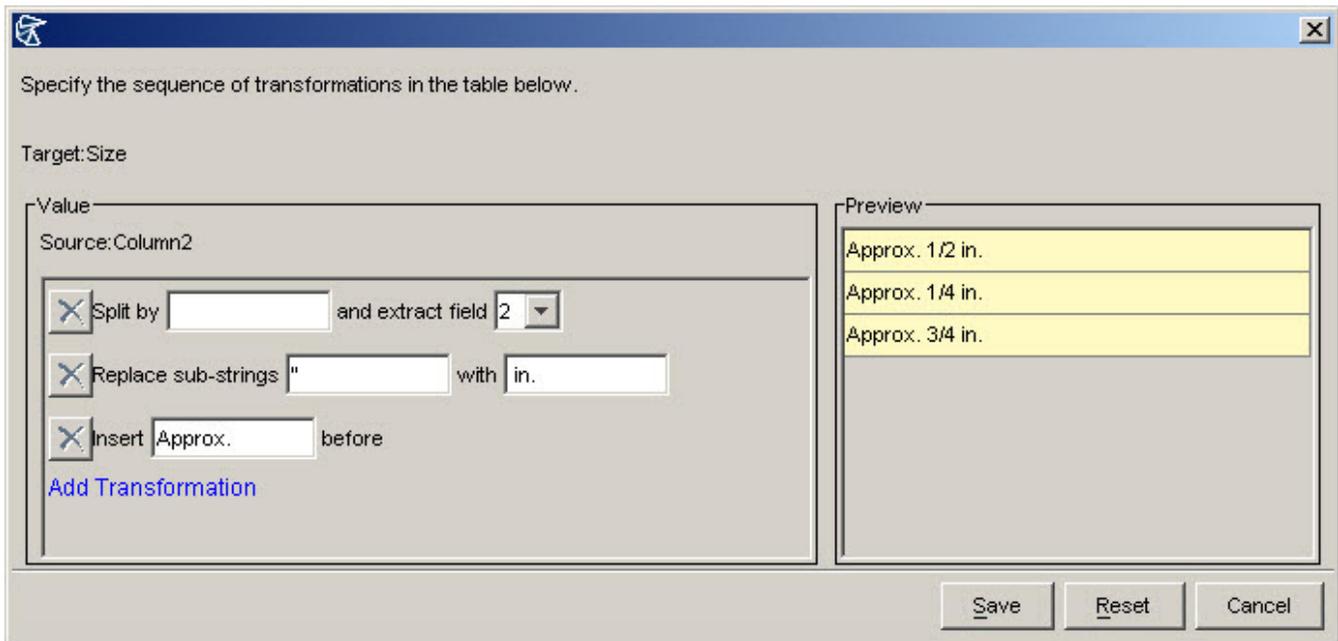
For the color portion of the attribute, the following transformations can be used:



- The **Extract** option gets all characters before the blank space (you cannot see the blank space in the image).
- The **Change case** option uses Mixed case to change the data to initial capitals.

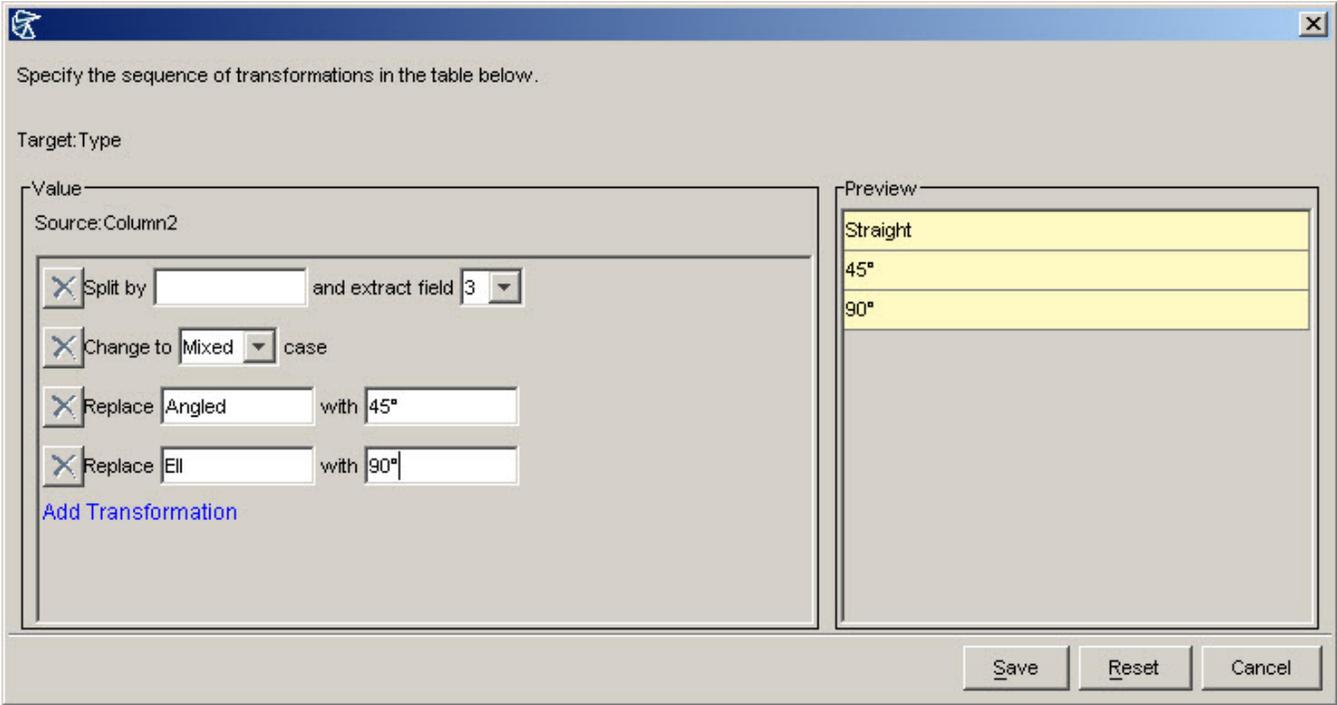
- The **Replace whole value** transformation updates Orange to Amber.

For the size portion of the attribute, the following transformations can be used:



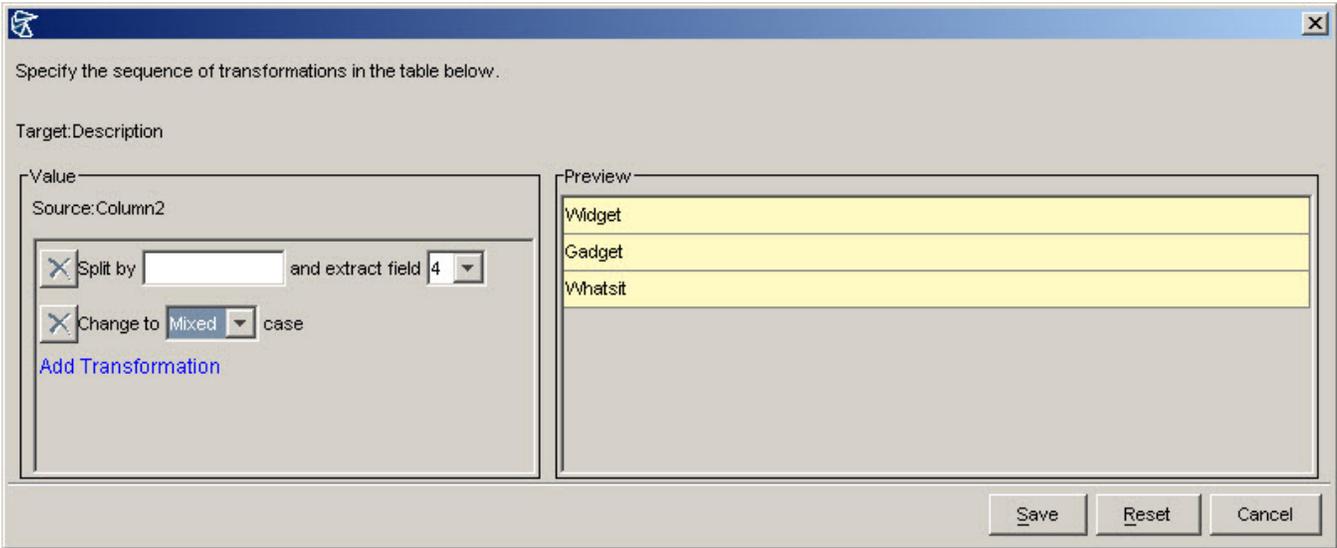
- The **Split and extract** operation works on the size field, using the blank space as the delimiter, and starting with the second field (or the text following the second blank space).
- The **Replace substring of the value** option replaces double quotes with the text string ' in.'.
- The **Insert text before** transformation adds the prefix 'Approx. ' to the results.

For the type portion of the attribute, the following transformations can be used:



- The **Split and extract** operation uses the space as the delimiter and finds the third field.
- The **Change case** option uses Mixed to change data to initial capitals.
- The first **Replace the whole value** transformation converts Angled to 45°.
- The second **Replace the whole value** transformation converts Ell to 90°.

For the description portion of the attribute, the following transformations can be used:



- The **Split and extract** operation gets the fourth piece of data, delimited by the blank space.
- The **Change Case** operation uses Mixed to change data to initial capitals.

The final mapping screen shows:

Map Data

Source:

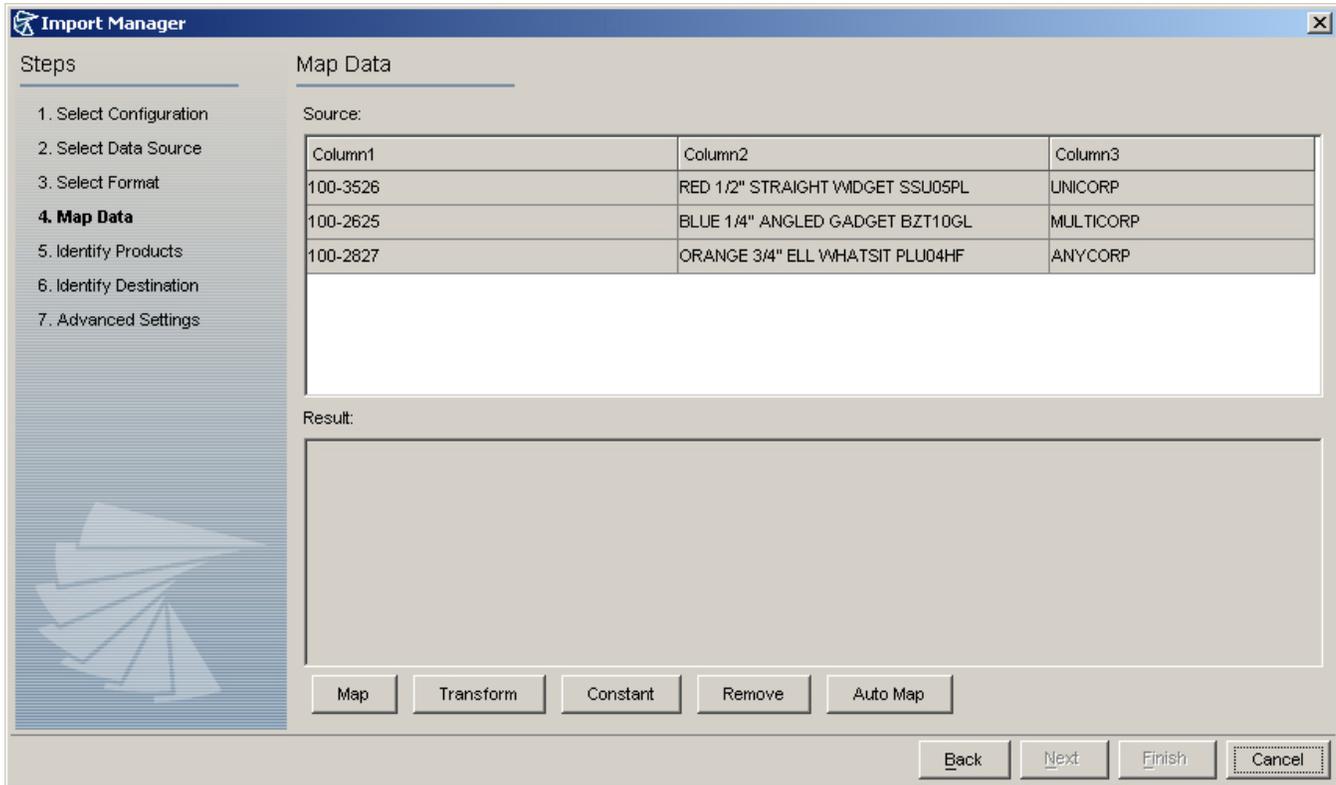
Column1	Column2	Column3
100-3526	RED 1/2" STRAIGHT WIDGET	UNICORP
100-2625	BLUE 1/4" ANGLED GADGET	MULTICORP
100-2827	ORANGE 3/4" ELL WHATSIT	ANYCORP

Result:

<ID>=Column1	Color=(Column2)	Size=(Column2)	Type=(Column2)	Description=(Column2)
100-3526	Red	Approx. 1/2 in.	Straight	Widget
100-2625	Blue	Approx. 1/4 in.	45°	Gadget
100-2827	Amber	Approx. 3/4 in.	90°	Whatsit

Extract, Split, and Convert Cryptic Data Example

This example builds on the previous one, but this time the original file contains additional information of a more cryptic character.



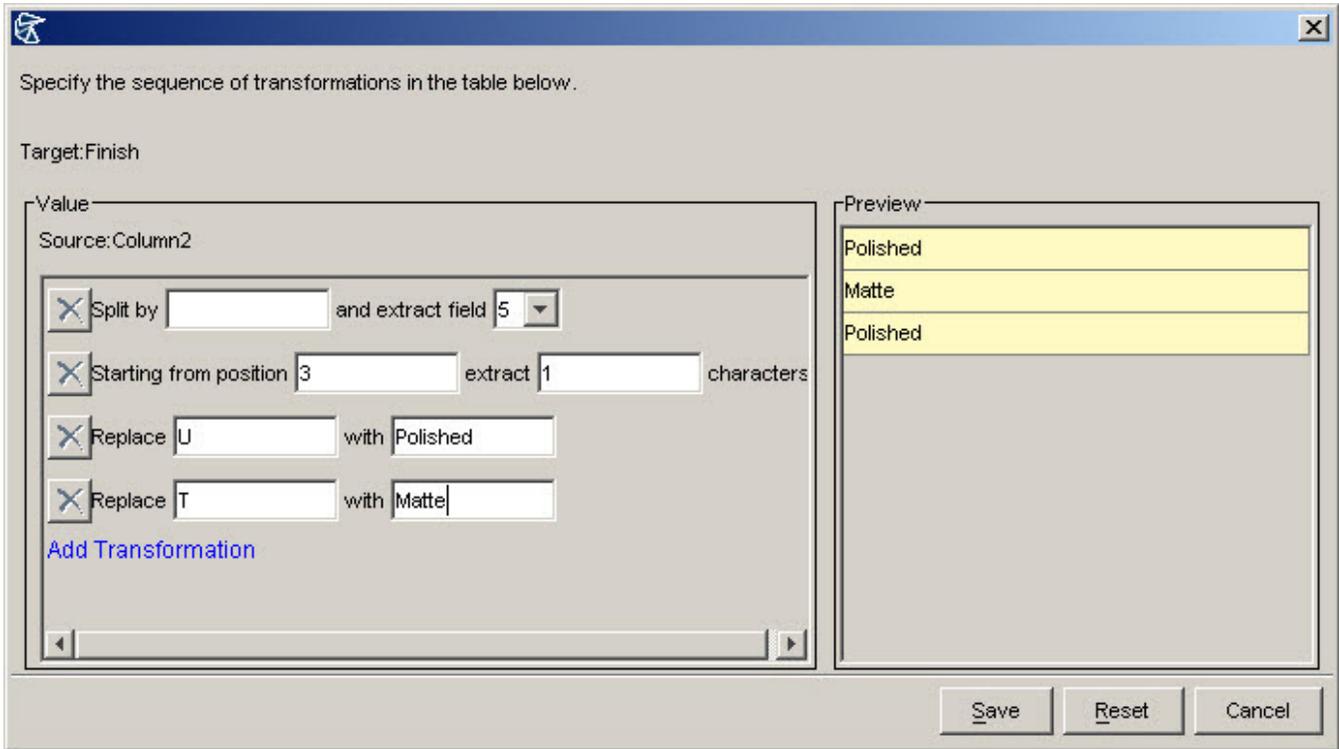
The last 6 characters of column 2 of the original input file contains information that we want to transform into "real" attribute values.

The 6 characters denote the Material, Finish, Warranty, Grade, and the Application of the product. For example: Material SS = Stainless Steel, Finish U = Polished, Warranty 05 = 5 years, Grade P = Professional, Application L = Domestic Plumbing

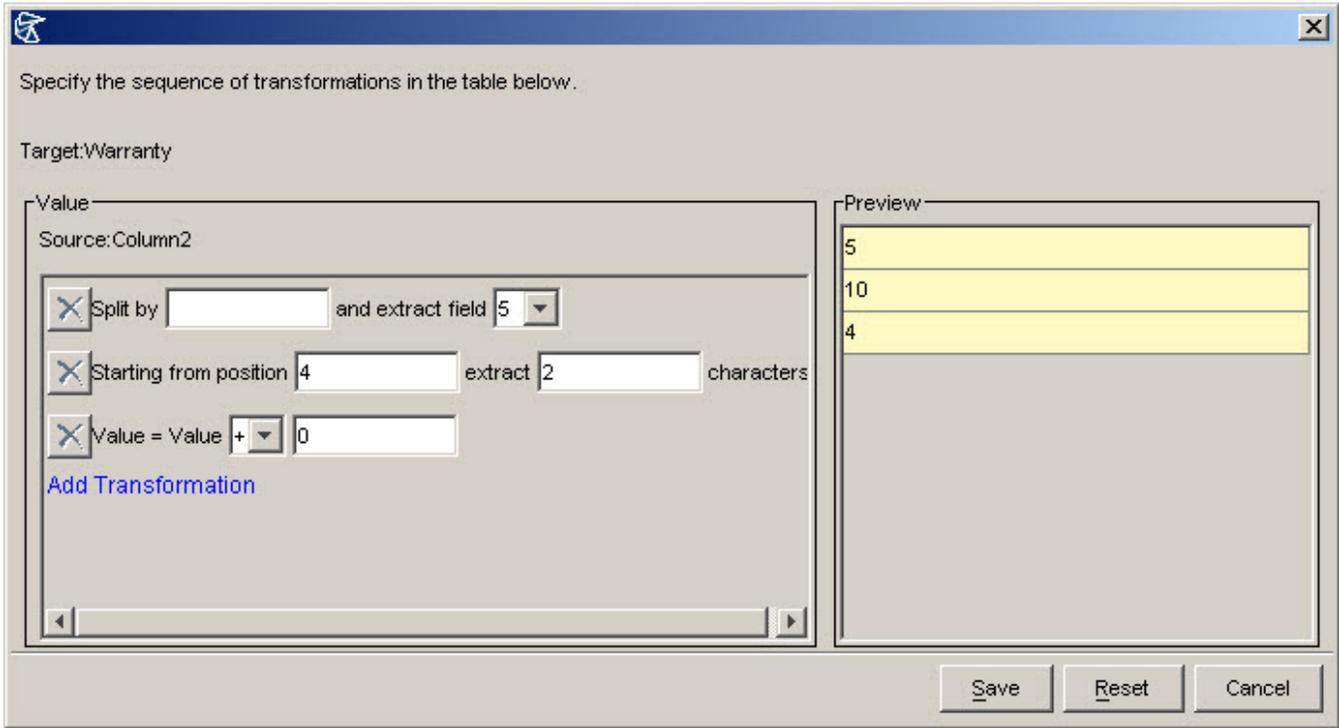
Five sets of transformations are applied to the five separate attributes.

- The first transformation extracts the last piece of data in column 2 of the original file using the blank space again. That data was SSU05PL, BZT10GL, and PLU04HF.
- The second transformation pulls the first two characters of those data, (SS, BZ, PL)
- The final transformation converts the two-letter material code to an actual text value.

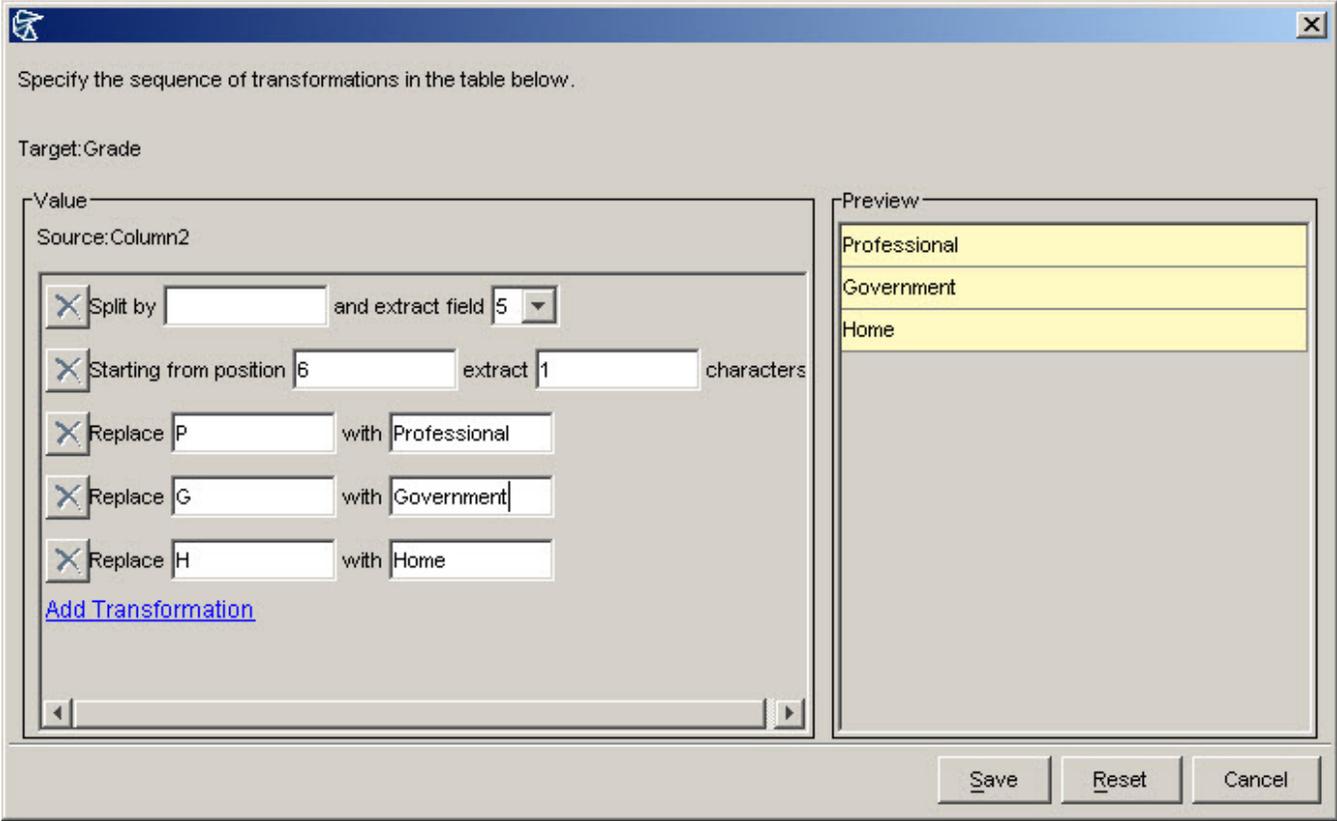
The following set of transformations apply to the Finish attribute. The first transformation pulls the same six characters as before (SSU05PL, etc.). The second transformation pulls a single character starting at position 3 of that resultant data (U, T, and U), and the remaining transformations convert the single character code into real text.



The following set of transformations apply to the Warranty attribute. The first transformation extracts the 6 characters from the original data stream (SSU05PL, etc.). The second transformation extracts just the two character year, (05, 10, 04), and the last transformation is a mathematical operation that is used to get rid of the leading zeros in the input.



The following set of transformations apply to the Grade attribute and are very similar to the transformations described for the other attributes.



The following set of transformations apply to the Application attribute and are very similar to the transformations described for the other attributes. The only exception is the second transformation where the last character is extracted.

Specify the sequence of transformations in the table below.

Target: Application

Value

Source: Column2

- Split by and extract field 5
- Extract the last 1 characters
- Replace L with Domestic Plumbir
- Replace F with Oil Pipelines

[Add Transformation](#)

Preview

Domestic Plumbing
Domestic Plumbing
Oil Pipelines

Concatenate Column with Unit Data Example

In this example, the source file is a CSV or XLS file with product data where one column has a value and another column has the corresponding units for that value.

One column has numbers such as 12, 15, 22, and another column has units like mm, cm, and inches. Two options are available to properly load this into STEP. One is an external option, the other takes place during the mapping phase.

Using the external procedure the two columns are concatenated in Excel using the standard Excel concatenation technique. Next, the column is mapped to the attribute.

However, if the file is too large for Excel, you can accomplish this concatenation using the following procedure in the mapping / transformation phase.

1. Select the column that contains the Product ID, as usual.
2. Select the column that contains the attribute values and map it to the appropriate attribute. In the following, the second column is mapped to the attribute "Gap Size".

The screenshot shows a 'Map Data' window with two tables. The 'Source' table has three columns: Column1, Column2, and Column3. The 'Result' table has two columns: <ID>=Column1 and Gap Size=(Col...). The source data is as follows:

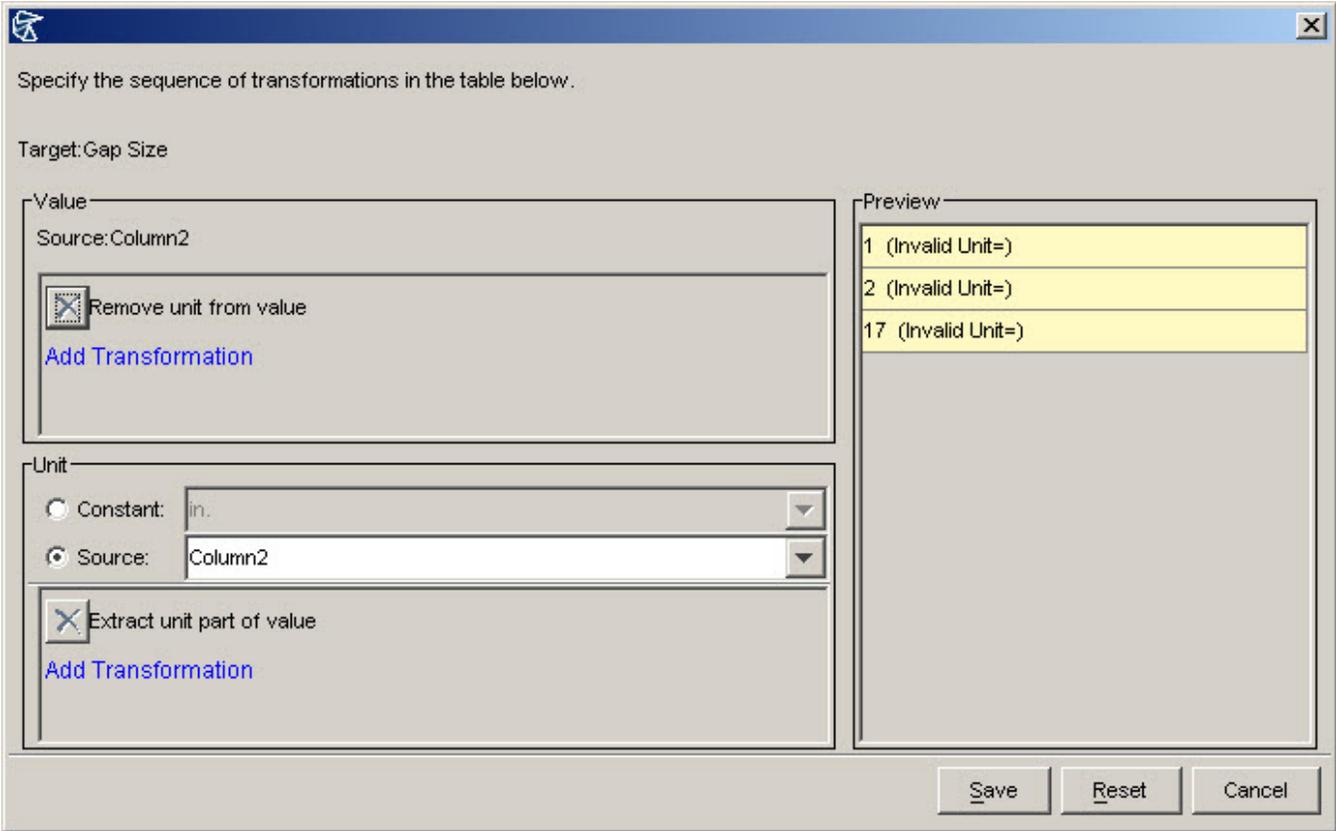
Column1	Column2	Column3
G100-352654	1	mm
J100-267225	2	mm
D100-289827	17	mm

The result data is as follows:

<ID>=Column1	Gap Size=(Col...
G100-352654	1 (Invalid Unit=)
J100-267225	2 (Invalid Unit=)
D100-289827	17 (Invalid Unit=)

Note: Note that Column 3 has the corresponding units, and that the **Result** panel shows an "Invalid Unit" error. The error disappears when the units column is concatenated.

3. In the **Result** panel, select the column that you want to concatenate the units into, and then click **Transform**.



4. In the **(Unit)** area, select **Source** and then the column that holds the corresponding units.

Specify the sequence of transformations in the table below.

Target: Gap Size

Value

Source: Column2

Add Transformation

Unit

Constant: in.

Source: Column3

Add Transformation

Preview

1 mm

2 mm

17 mm

Save Reset Cancel

The **Preview** panel on the right shows how the data will be imported.

5. Click **Save**, and you are done.

Note: The Units column does not have to be "clean". You can perform transformations on the Units column in exactly the same manner as described for the attribute values. You can also enter a Constant value as the Unit.

Specify the sequence of transformations in the table below.

Target: Application

Value

Source: Column2

- Split by and extract field
- Extract the last characters
- Replace with
- Replace with

[Add Transformation](#)

Preview

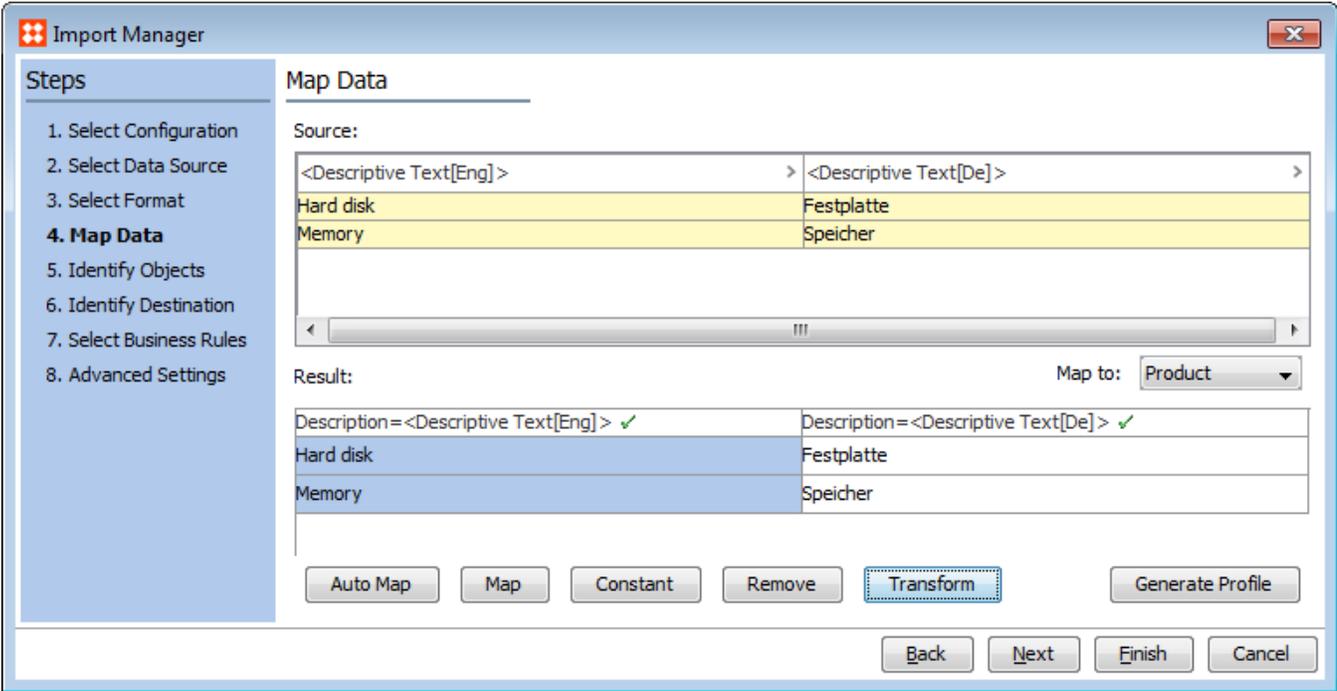
Domestic Plumbing
Domestic Plumbing
Oil Pipelines

Dimension Point for Single Dimension Data Example

This example demonstrates import of single dimension dependent data (language in this case) from either a CSV, Excel, or Generic XML file. The title of the column indicates the language of the values. For this case the following spreadsheet is used as input:

	A	B
1	<Descriptive Text[Eng]>	<Descriptive Text[De]>
2	Hard disk	Festplatte
3	Memory	Speicher

In the map step both columns are mapped to the language dependent attribute 'Description':



To map the first column to English, the column is selected and the **Transform** button is clicked:

Specify the sequence of transformations in the table below.

Target:Description

Mandatory

Value - Source: <Descriptive Text[Eng]>

Add Transformation

Dimension Point

Constant: English

Source: <Descriptive Text[Eng]>

Add Transformation

Preview

Description = <Descriptive Text[Eng]>	✓
Hard disk	
Memory	

Save Reset Cancel

Note the extra frame for Dimension Point, which is shown because the attribute has a single dimension dependency. Here the dimension point can either be mapped to a Constant or a Source. The Constant is used in case all values of the import are in the same language. The default behavior is to use the language of the import context. The 'Source' radio button is used in case the language is specified in a separate column. In that case transformations may be needed in order to map the values of the language column correctly to the dimension points.

In this example the Constant mapping meets the requirement. Select the **Constant** radio button and pick **English** in the drop down menu next to it. Finally, close the dialog by clicking **Save**. Using the same procedure, the German text is mapped to the German language. The remaining steps of the Import Wizard should be carried out as usual.

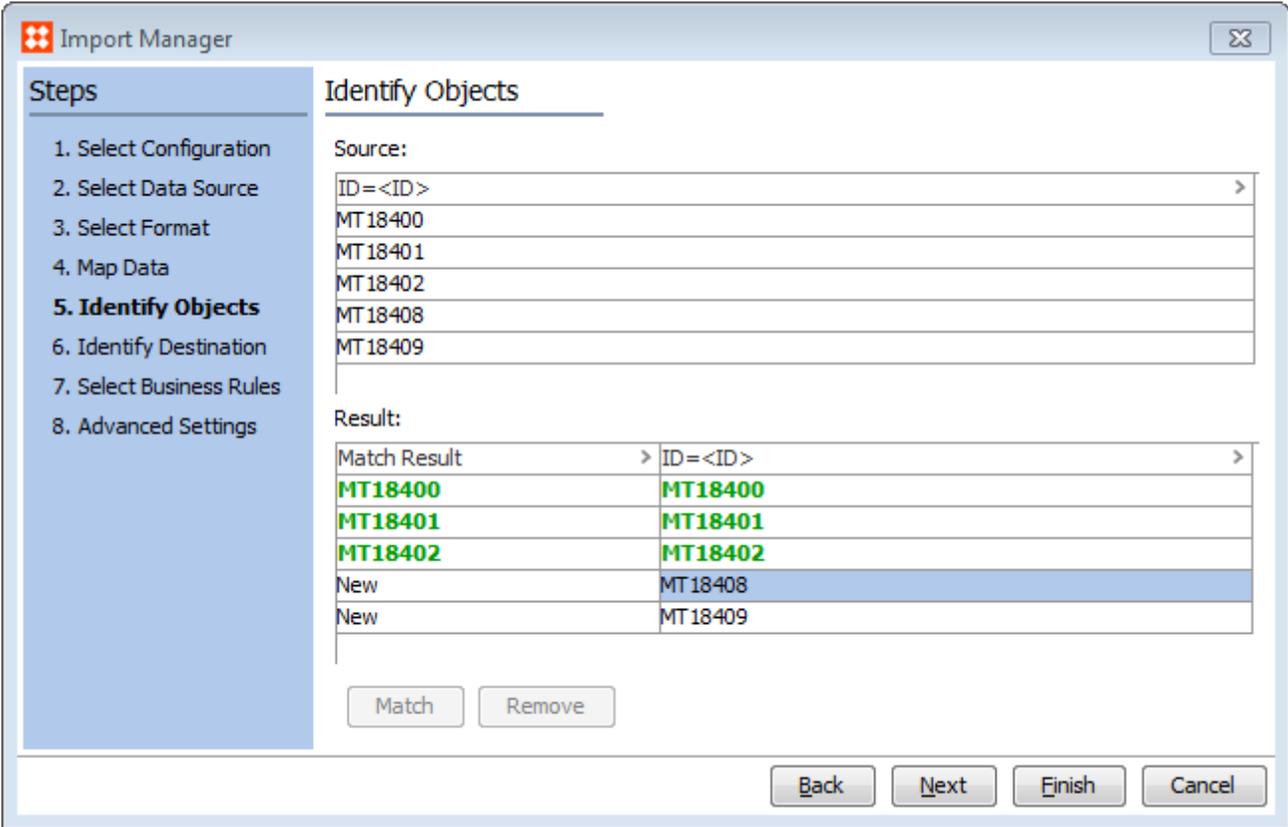
Step 5 - Identify Objects

Step 5 of the Import Manger wizard, Identify Objects, allows you to load existing product data into STEP by locating those products by an attribute value.

Importing With Object IDs

When the ID has been mapped, the Result panel displays the first 20 results:

- Objects found in STEP are displayed in green
- New objects display 'New'



You cannot introduce new products by using Identify Objects. All products must exist already in STEP. If there is no match of the attribute value to a Object ID, the record is ignored.

You can select any attribute that you have already mapped in the prior screen, the Map Data screen, but you can only match on one attribute. So if you have several attribute columns that are "alternate part numbers" or "replacement part numbers", etc. you cannot select multiple attributes at one time in order to find the actual Object ID in STEP.

Matching can only be done on one attribute at one time. However, you can try several columns one at a time by using the Remove button and by selecting another attribute column in the top panel.

Importing Without Object IDs

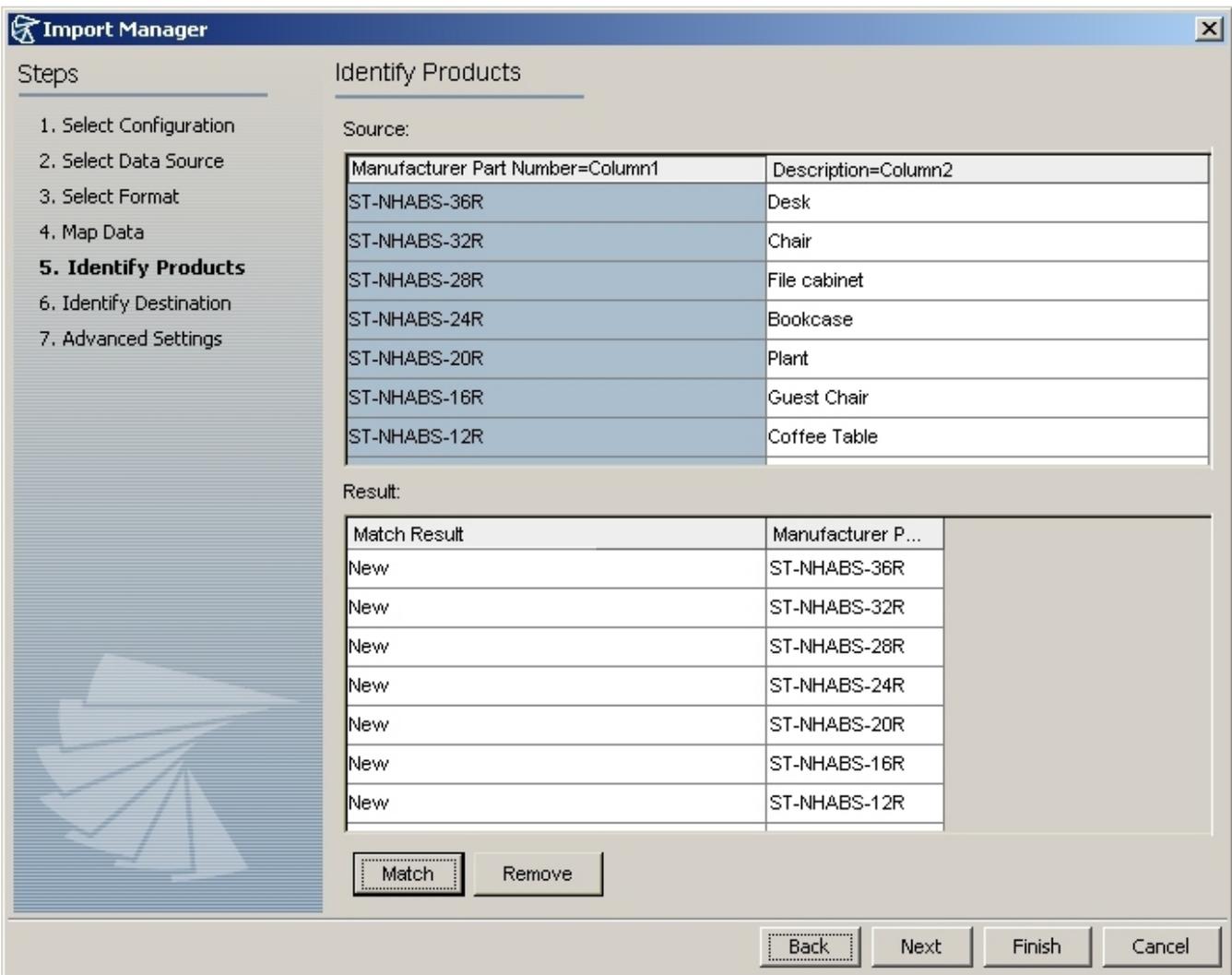
If new objects without IDs are to be imported, an ID Pattern must be set for the target Object Type. Otherwise the objects cannot be created.

The following example describes how to identify products by attribute values. The product data is loaded from a file with no product IDs listed but contains the manufacturer's own part numbers.

Normally, the manufacturer part number would be an attribute of your own product. If the manufacturer's part number is unique, you can then load the file by having the system locate the products by an attribute value – the manufacturer's part number - instead of mapping a column of data to the product ID as you would normally do.

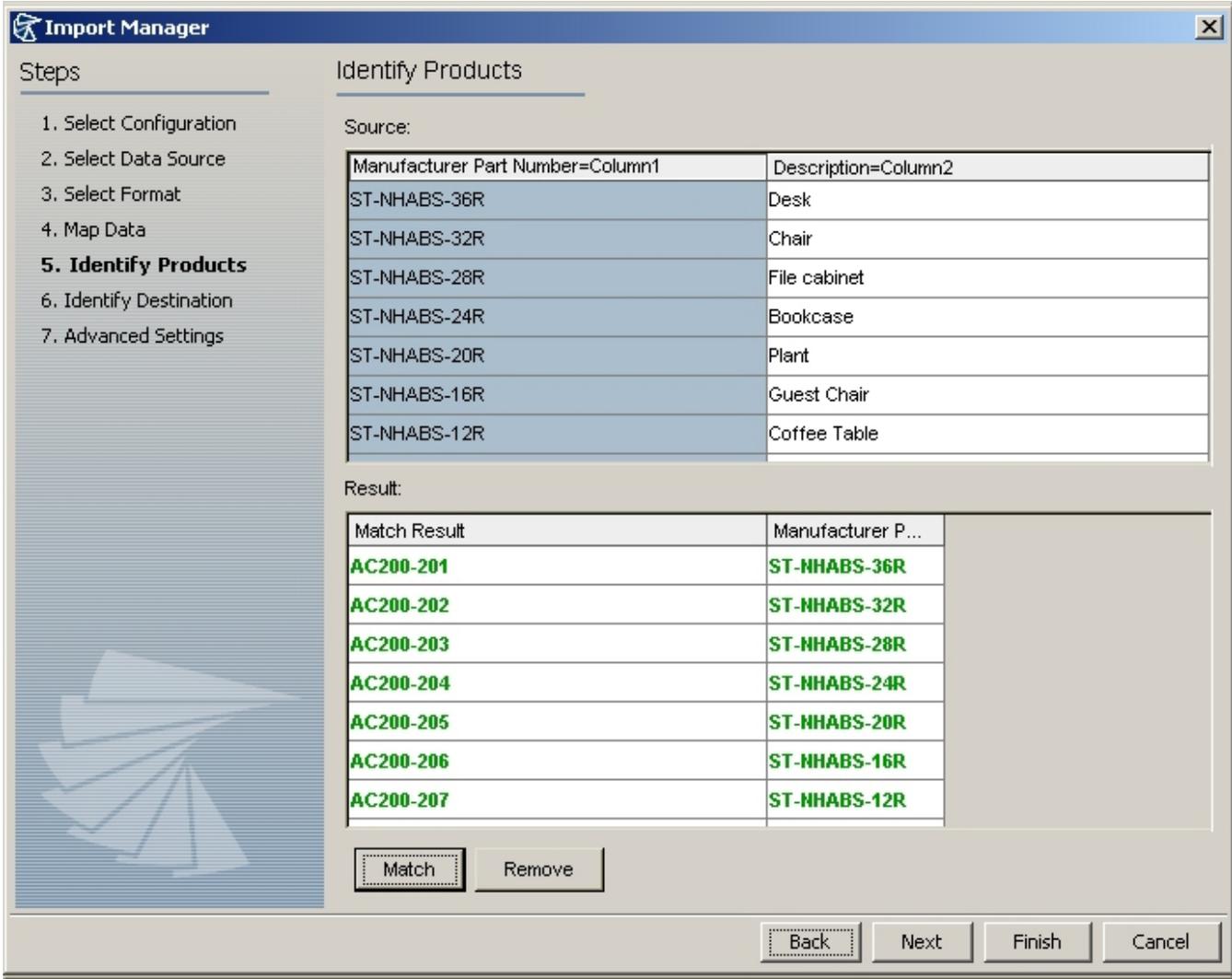
A file with two columns of data is imported to STEP. Both columns are mapped to attributes. The first one is mapped to the attribute Manufacturer Part Number, and the second is mapped to Description.

1. Select the first column in the top panel, and click **Match**. The system attempts to locate products by the values in that attribute.



Note: The Match button is only enabled if you have not mapped a column to a product ID in the Map Data step. STEP assumes that if you defined a column of data as the ID, no matching of attribute values to Objects IDs is required.

If the match is successful, the result is displayed in the **Results** pane.



The first column equals the ID of the product in STEP. The second column is the attribute value that matches that product.

Match only checks the first 20 products in the file. If it is unable to find a match it the result in the first column is "New".

New products are not loaded to STEP when they are located by attribute values. Even if you have set up the system to auto-generate an ID for new products, and even if you allow new products to be created, the system rejects any new products in this file when the products are located by attribute values.

2. Click **Next** to move on to the next step or click **Finish** if you have made all the selections that you need to make and want to launch the import process.

Step 6 - Identify Destination

When creating new objects, entities, assets, and classifications in STEP, object type and parent must be specified. When this data is missing in the import file, it can sometimes be derived via transformations, added by mapping a constant value in the Map Data step, or added using the Identify Destination settings, as shown below.

- **Approver** is used to specify the user name of the person that is allowed to approve the products after the import process is complete. When you click the ellipsis button (...), you can search or browse for the user. You are the default Approver. Normally, this is used in conjunction with a customized workflow.
- **Import Workspace** is used to specify which workspace the data is loaded into. You can use this feature if you have set up more than one maintenance type of workspace in your system, such as 'Staging'. The default is Main. You can never import data into the Approved workspace
- **Default Parent Product** is used to specify where new products should be placed in STEP. A parent is required to create new objects in STEP. Use this feature if no column has been mapped to the Parent ID, or if the entry in the Parent ID column is invalid (for example, if the parent product listed in the input file does not exist in STEP). The Import Manager wizard does not check all of the records in the load file. This means that if any of the records in your load file is a new product, and does not have a parent, it cannot be created in STEP. Similarly, even if you have mapped a column in the load file as the Parent ID, the Import Manager wizard has not validated all those entries before this point.

- **Default Object Type** is used to specify which object type to use when there are new products in the load file. An object type is required to create new objects in STEP and valid object types are based on the selected default parent product. It is not unusual to have different object types for categories, subcategories, families, or individual items. And it is also not unusual for objects to have several possible object types, depending on your system setup. Common setup is to use this feature when all products in the load file are of the same object type.
- **Batch Directory** is an obsolete option and selecting a directory has no effect on the import. Instead of using this option to link products to a classification folder, use the Map Data step to link a product to one or more classifications with the specified Reference / Link Type. For more information on importing to classifications, see the **Product Classification Link** heading of the **Additional Column Mapping Options** section in the **Importing Data and Assets** documentation.
- **Test Import Only** is used to load a small file into STEP without saving or committing any changes. The Import Manager wizard goes through all the steps of importing the file, but does not perform the final step of confirming the changes in STEP. This allows you to review the execution report so that you address any warnings or errors encountered when loading the file. You can then make appropriate changes to the load file and do the load again, until all errors have been resolved.
- **Reject New Products** is used to specify to reject any new products that are listed in the load file. If selected, new products are not loaded or created. The execution report lists the products that have been rejected as a result of this selection. This is useful, for example, when loading price files.
- **Reject Duplicate Products** is used to specify whether only new products should be imported from a load file that contains both new and existing products. This way you avoid updating existing products in STEP. Existing products are left untouched.

Step 7 - Select Business Rules

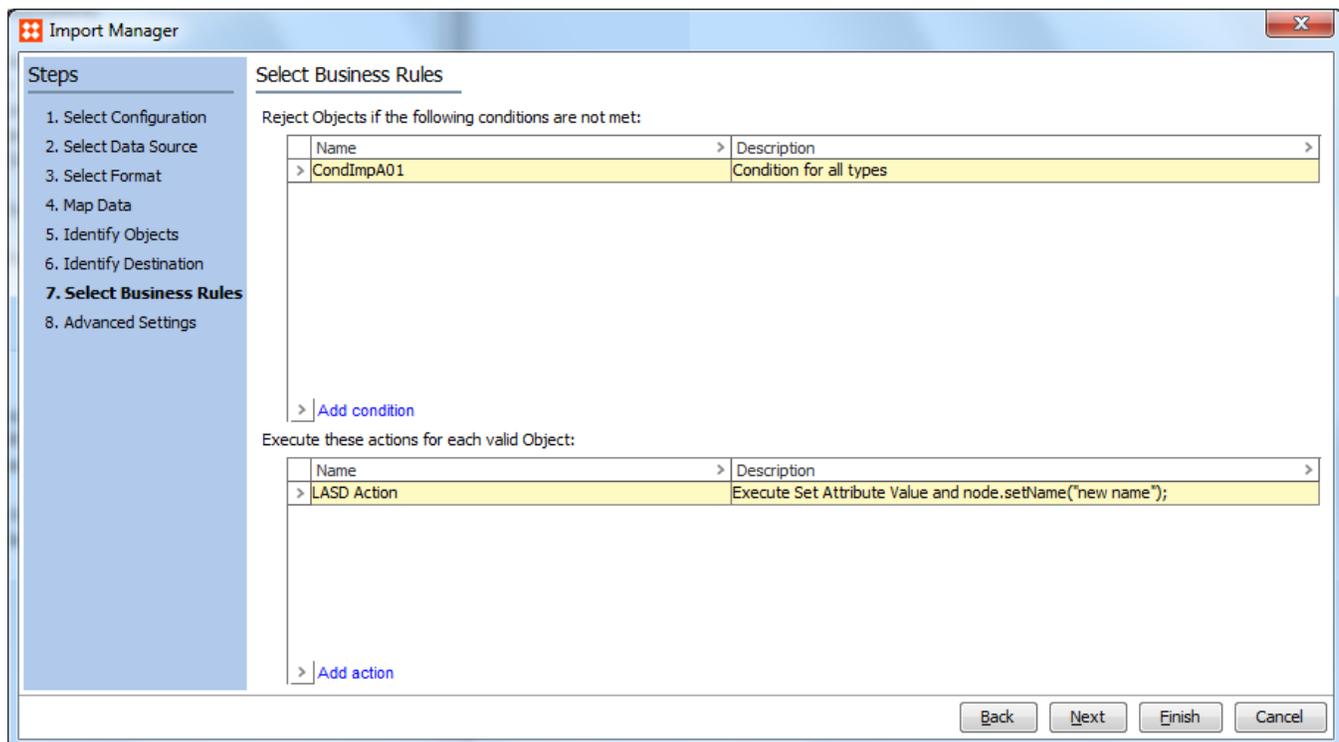
The import data is processed based on the selected business rules. The number of business rules that are available depend on your system setup. First, business conditions are validated against objects during the import process, and then business actions are executed on the valid objects. If a business rule results in an error, the Import Manager skips the object. Errors are logged in the Import Execution Report.

Adding Conditions and Actions

1. Click the **Add Condition** link, use Browse or Search to locate the condition, and then click **Select**. Repeat until all the relevant conditions are added.
2. Click the **Add Action** link, use Browse or Search to locate the action, and then click **Select**. Repeat until all the relevant actions are added.
3. Click **Next** for the next step or click **Finish** to launch the import process.

Removing Conditions and Actions

- To remove a condition, select the relevant condition, right-click, and then select Remove condition.
- To remove an action, select the relevant action, right-click, and then select Remove action.



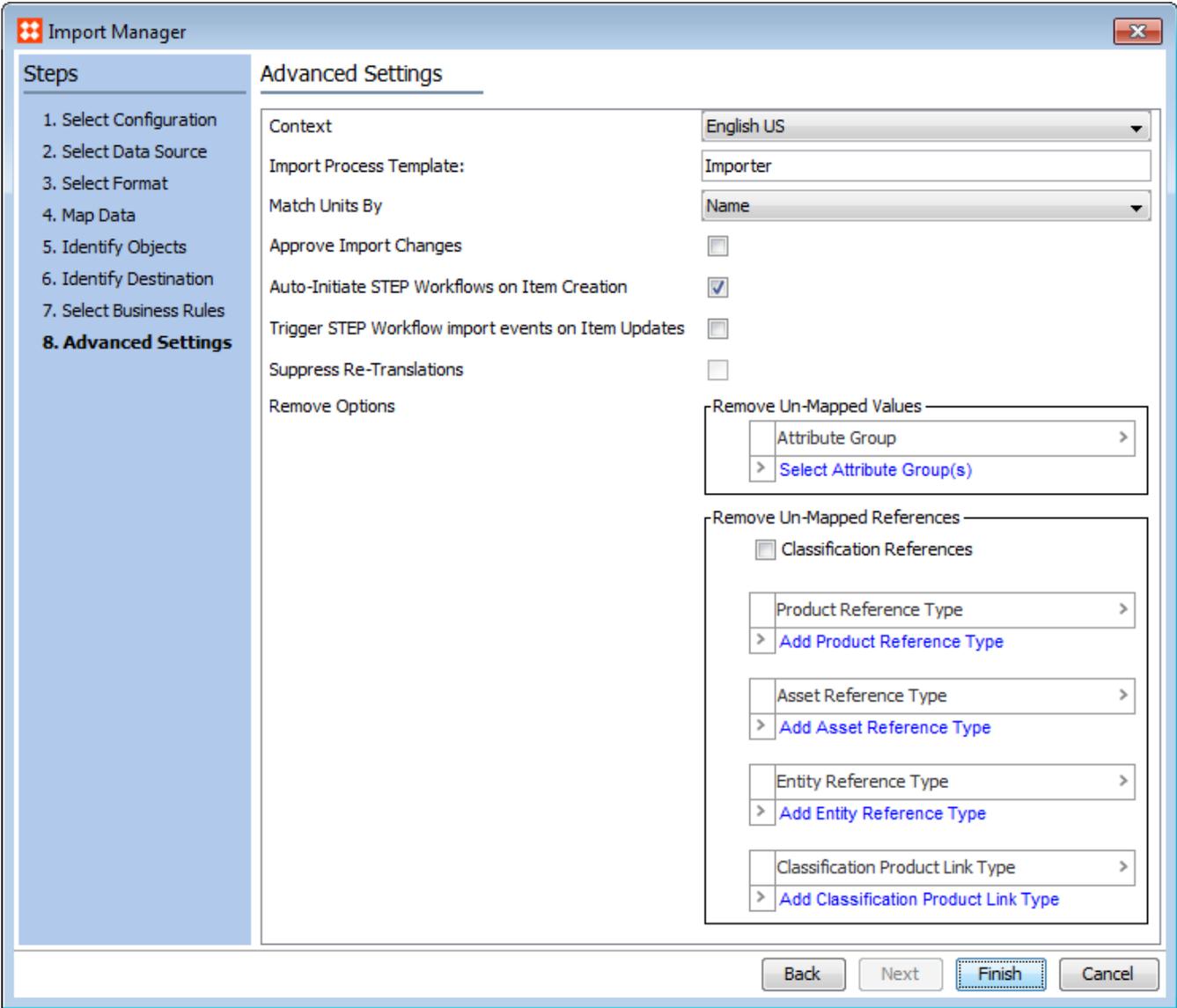
Business Rule Limitations

- Objects must be imported one at a time for the validation mechanism to work. It is therefore not possible to import nested STEPXML documents when you are using business rules.
- If one or more business rules are selected, the Import Manager uses domain mode.
- Changes are not detected for references and links. Objects with references in the import are therefore always reported as changed.
- Sometimes imported references are deferred if they depend on objects that have not been imported yet. As a consequence, the deferred parts of an imported object are not present while it is being validated. Therefore, you should be careful when using business conditions to validate references.
- When actions are used, the 'Approve Import Changes' on Advanced step is disabled - importer can no longer automatically auto approve imported data as side effects from business actions are unknown to importer. However it is still possible to approve imported objects - it can be done from within a business action.

Step 8 - Advanced Settings

The Advanced Settings screen is only available in licensed versions. It provides a series of options that are not required on a day-to-day basis for data imports.

Important: Advanced Settings options are mainly used to clean up data within STEP. However if you are not careful, you can cause data integrity issues. Double-check your load file and your System Setup before using any of these options. Common setup is to perform a test on the test server first, and start testing with small files.



General Advanced Settings

- **Context** - If you are creating a new configuration, the context shown here is the one that is currently selected in STEP. That is usually the safest way to import data. You can, however, select a context different than the

currently select one. If you selected an existing configuration, the context that was set up in that configuration is shown. Dimension Point specific mappings will overrule this setting. Consider the context setting when creating an Import Configuration that will be used repeatedly. You can only import to one context at a time, so if you have several languages, you will have to do several imports.

Note: When importing ETIM6, you selected a Language in Step 3. Verify that you choose the Context that should be updated with the selected language.

- **Import Process Template** - This is used to specify a workflow process. The default is STIMPMGR, which is the standard workflow.
- **Match Units By** - When the system loads data, it has the capability of trying to separate actual values from their units in the load file. You use this option to indicate to the system what to match the units on in the data file – the unit's ID or its name. The default is name.
- **Approve Import Changes** - This option is used to approve the changes that the import causes. When checked, the Force Domain Import check-box is automatically selected and disabled. If business rules are applied from previous step this option cannot be selected - instead it is possible to approve entire object from business rules.
- **Auto-Initiate STEP Workflows on Item Creation** - If a workflow is valid for either products, classifications, assets or entities, it is possible to have an instance of a given workflow created whenever an object of the valid object type(s) is created. That is, a workflow is automatically started when the object is created. This functionality is available for workflows set to Auto Initiate on Object Creation. Users with the privilege Disable STEP Workflow Auto-initiation in Imports are allowed to disable auto-initiation in the Import Manager (column based Imports) or directly in the XML for XML imports (by setting AutoInitiate="N" in the STEPProductInformation tag).
- **Trigger STEP Workflow import events on Item updates** - This option controls whether or not the import triggers events. In XML, the functionality is enabled / disabled using the STEPProductInformation tag attribute 'STEPWorkflowImportEvent' that can be set to "Y" or "N". The option is typically used in imports based on preconfigured Import Configurations.

Note: It is only possible to trigger Events from Imports running in Domain mode.

- **Suppress Re-Translations** - Checking Suppress Re-Translation, prevents products, classifications, entities or assets from getting the status **Re-Translation Needed**. This option only has effect if the data import is set to be approved too.
- **Auto Select Import Mode, Force Domain Import and Force Database Import**. This option is only available if you have the required privileges for using it. It is typically used in connections with large import files and results in a slower import and provides more information in the execution report. It is not recommend to use this option.

Remove Un-Mapped Values Setting

This option allows you to select one or more attribute groups. It enables you to remove attribute values from the selected attribute groups and replace them with the values specified in the load file, leave them blank, or return them to their inherited value.

Remove Un-Mapped Values

> Select Attribute Group(s)

Important: Use this option with care as it can both add and remove data from STEP.

For example, you have 200 products listed in your load file, and they are all gloves of one type or another. You use 8 product attributes to define these gloves. Now you want to use just four attributes. Further, all of the original 8 attributes are in their own attribute group. You load the 200 products using just the 4 attributes that you want to use, and you then select the attribute group under the Remove Un-Mapped Values option. STEP removes all attribute values in that attribute group for those 200 products only, and then loads the attribute values as specified in the load file. Thus, it has removed all 8 attribute values for all 200 products.

Remove Un-Mapped References Settings

Remove Un-Mapped References

Classification References

> Add Product Reference Type

> Add Asset Reference Type

> Add Entity Reference Type

> Add Classification Product Link Type

- **Classification References** - Check to remove all product to classification links on the products being imported and only create mapped product to classification links. A message is displayed asking you to confirm that you want to remove all unmapped classification references for the imported object.
- **Add Classification Product Link Type** - Click and select the specific **Product to Classification Link Types** to be removed. A message is displayed asking you to confirm that you want to remove all unmapped classification product links for the imported products.

Important: Use these options with care as it can both add and remove data from STEP.

Only use these options when you have a load file where you have mapped one column to the product ID and another column to a classification ID. That is, you have products that you want to link to a classification.

For example, if you have products that are linked multiple times to different classifications, you can use this option to clean up product to classification links not specified by the Import file. You therefore define certain products that are linked only to the classifications you list in your load file and not to any other classification.

Do that by listing the products and classifications in the load file, mapping them, and then selecting one of the two of the following options to remove remaining product to classification links.

- **Product Reference Type** - This option enables you to clean up your **Product to Product References**, **Classification to Products References**, or **Entity to Product References**.

For example, if you have products in your system reference other products using the reference type 'Replacement Product'. Over time, products that really are replacement products have been linked to other products using other reference types such as 'Obsolete Product' or 'Similar Product'. To clean up, you list the products and their valid reference types in your load file, map them, and then select the reference type (s) that you want to remove from these products.

The system then reads your load file and finds the source product, which is mapped to the ID will be linked to one or more target products). All links are then removed from that product to all other products that have the reference type(s) selected in this Import Wizard screen. The source product is linked in the load file to the target product(s) using the reference type that was selected when you specified the target product.

For example, Product 123 is linked to products 456, 567, 678, and 789 with the reference type of 'Obsolete'. All those links are incorrect. Therefore, you specify in your load file that 123 should link to 567 with the reference type of 'Replacement Product', and nothing else. There are no other entries for product 123 are in the load file. In this screen, you select the reference type 'Obsolete'. After this load file has been processed, there are no links from product 123 to any product with the reference type 'Obsolete', and there is a link from 123 to 567 with the reference type of 'Replacement Product'.

You can only unlink the reference types that you have specified. To unlink all references from a product , no matter what type, you must add each reference type.

To make two references between two products (and all the other product ID pairs listed in the input file), you must load the file twice, specifying one reference type the first time, and the other reference type the second time. Common setup is to deselect the Remove Un-Mapped References option the second time.

- **Asset Reference Type** - This is similar to the product reference type described earlier, except that it refers to **Product to Asset References**, **Classification to Asset References**, or **Entity to Asset References**. It enables you to clean up, for example Product to Asset References by removing some or all references between products and assets and reinstating only those listed in the load file.

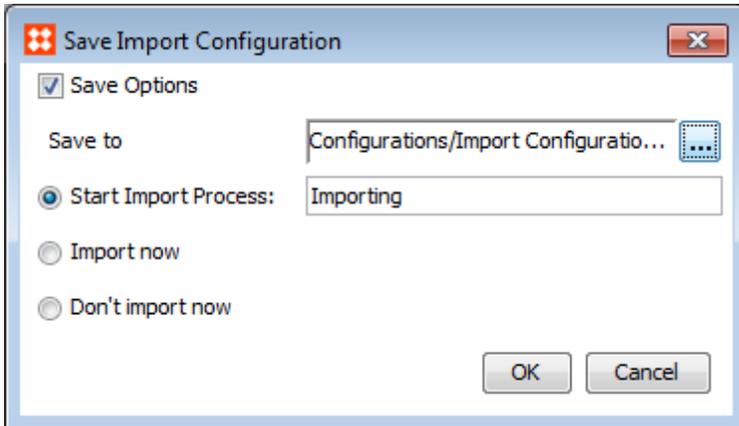
For example, in STEP you have products that are linked to images using the reference type 'Primary Image'. However, images that are really manufacturer logos have been linked to various products using this reference type, instead of using Vendor Logo. You therefore list the products and images in the load file together with the correct reference type, map the columns, and then select the reference type to be removed from the product / image. The system removes the selected reference type(s) and reinstates only the reference type(s) listed in the load file.

- **Entity Reference Type** - This is similar to the asset reference type described earlier, except that it refers to **Classification to Entity**, **Product to Entity References**, **Asset to Entity References**, or **Entity to Entity References**.

Starting the Data Import Process

You can start the Import process after completing some or all of the steps in the wizard.

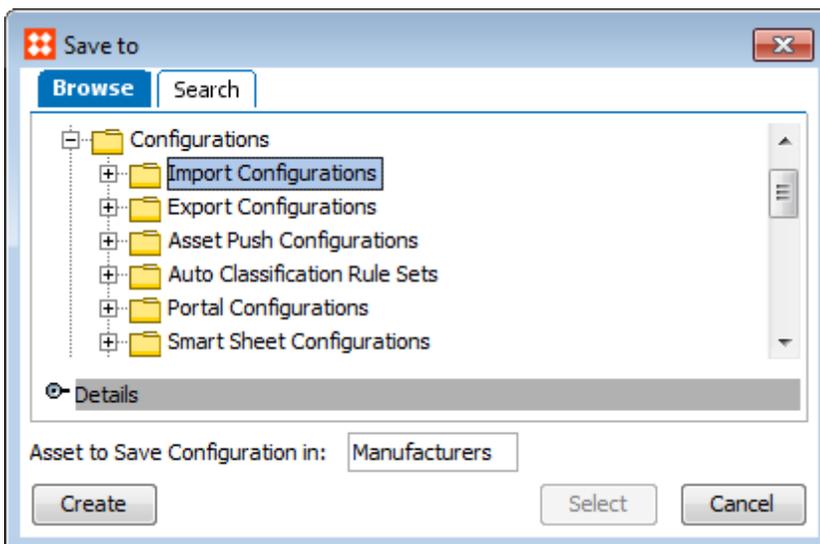
1. Click **Finish** to start the process. The **Save Import Configuration** window appears.



2. To save the configuration so that it can be used for other imports, check **Save Options**, click the ellipsis button (...), and then use the **Browse** or **Search** tab to select a location. Configurations are saved as a special type of asset and live in the classification hierarchy.

When updating an existing configuration, the path and name of the configuration that you selected on the first wizard screen is automatically displayed in the **Save To** box.

3. In the **Asset to Save Configuration in** field, enter a name for the configuration, and then click **Create**.



4. Optionally, you can add a name for the import process in the **Start Import Process** text field. This makes finding the process on the BG Processes tab easier.

5. Determine the final action:

- Select the **Start Import Process** radio button and click **OK** to send the job to a background process.
- Select the **Import Now** radio button, which is available if the import file is less than 100 KB on a default system. This allows you to import the file without use of a workflow. Click **OK** to send the job to a background process.
- Select **Don't Import Now** to only save the settings, assuming you selected the Save Options checkbox. Click **OK** to close without importing.

Note: No data is imported into STEP if, on the **Identify Destination** step, you checked the **Test Only Import** checkbox. Details of the test import are available on the **Import Report** dialog when selecting the Import Now option; details are available in the BG Process when selecting the Start Import Process option.

Importing Assets

Import Images and Documents wizard is used to import assets. Images are the most frequently imported asset, however, it is possible to import any electronic file using the same procedures as the ones used for images.

Audience

We recommend that users of the Import Images and Documents wizard have the following skills.

- An understanding of the basic concepts of STEP, of maintaining attributes and their values, and of creating and maintaining products, classifications or entities.
- Knowledge of the STEP system equivalent to basic STEP training
- Thorough knowledge of the files that are imported.
- Knowledge of dimensions and dimension points and whether or not you require assets to be dependent on dimensions, and how you would use such a set-up.
- Knowledge of MIME types.

Also, ensure that your system is set up to accept appropriate file formats for imports of assets, and ensure that each of these formats is assigned an appropriate icon.

Importing Assets: A Checklist

Before starting the import process, be aware of the following:

- You can load assets and use either the file name as ID or use an automatically generated number. In either case, the file name is used as the name of the asset in STEP.

Note: Asset IDs have a maximum length of 40 bytes, and names have a maximum of 80 bytes.

- Do not to use special characters or spaces in asset IDs or names.
- Although you can upload images of almost any file format, print publications the system accepts only TIFF or EPS file formats. You can upload other formats for non-print applications.
- You should check all your images for accuracy and suitability before uploading them to STEP. This includes file size, resolution, one clipping path maximum, compression type, actual image size at 100%, whether or not there is an embedded low-res Tiff in EPS images, and so on.
- If you have a Photoshop image that has text in it, you must outline that text. You can not have text with font calls within an image. For Illustrator files, you can also outline the text. If you choose not to do so, then you must select the option to embed the font(s) within the EPS file.
- To ensure a smooth upload of images of different file types, you are strongly advised to use extensions to the image file names that match the file type, e.g. .TIFF or .EPS.
- Avoid having assets in STEP that have different IDs but the same name. IDs must be unique, but names do not have to be unique. However, having different images with the same name can be confusing with the exception of country- or market specific assets. In this case you can import assets with the same ID and name into different contexts.

Importing Assets: Special Settings

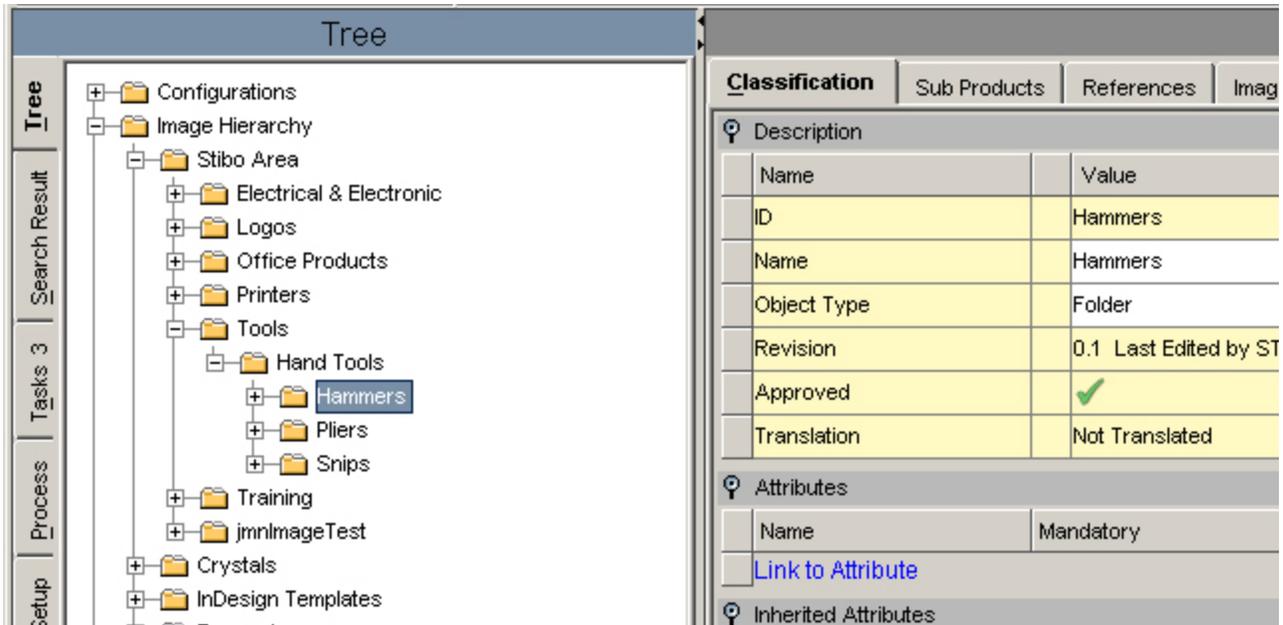
- You can upload multiple assets by selecting a folder and not just an individual asset. All assets in the selected folder, and any assets in any subsidiary folders to the selected folder are uploaded.
- When you load assets you can specify that you want to use the file name as both the ID and the name within STEP. Alternatively, you can ask the system to automatically generate an ID and use the file name for the name.
- You can select the dimension to which the assets should be loaded. In the case of loading images, this feature is most often used when loading illustrations that have text in them, and there is a different version of the illustration per language.
- You can link the assets you are loading to existing products in STEP, using one of several predefined parameters, such as matching the first x number of characters of the asset file name to the product number.
- You can either ask the system to try to recognize all the types of assets you are loading and assign them the appropriate object type, or you can force the system to load them all with a fixed object type.
- You can tell the system to automatically overwrite any asset in STEP that has the same ID (or not), and you can also set a minimum resolution, below which the system will optionally give you a warning if you attempt to load images that are below that specified resolution.

Starting an Asset Import

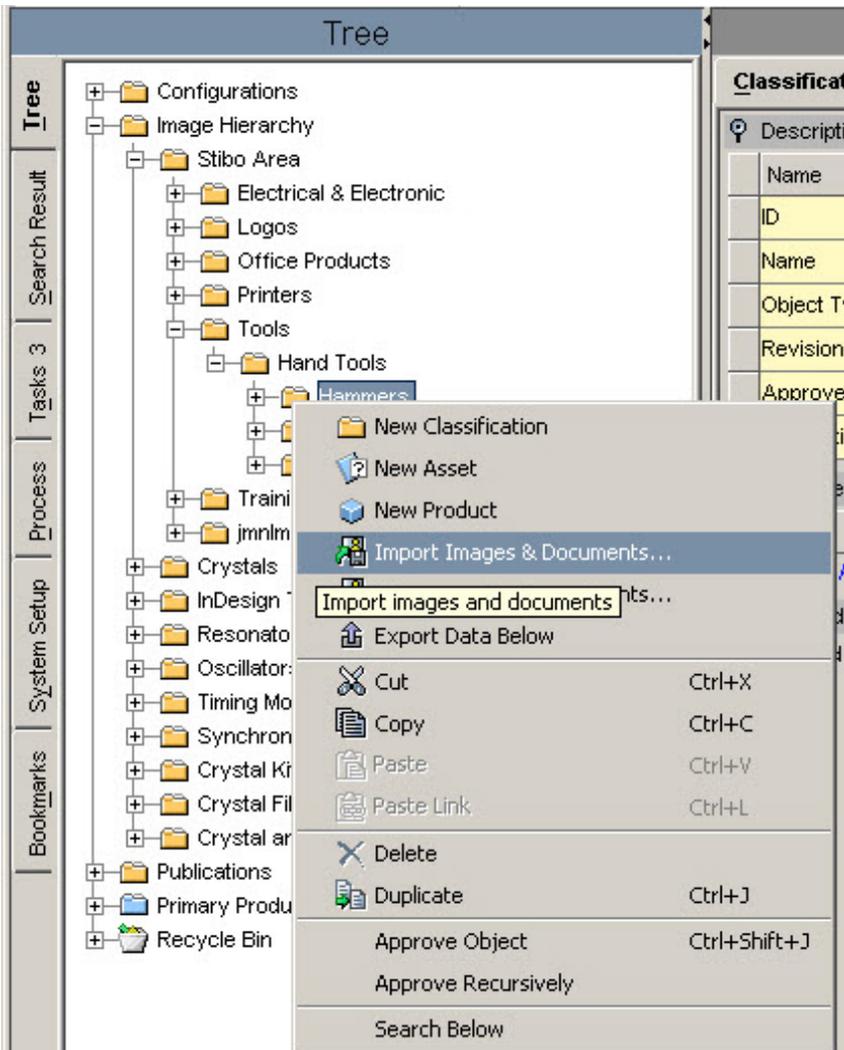
When you have checked the assets that you want to load, you are ready to start the Import Images and Documents wizard. You have to select a target location for the import to start the wizard.

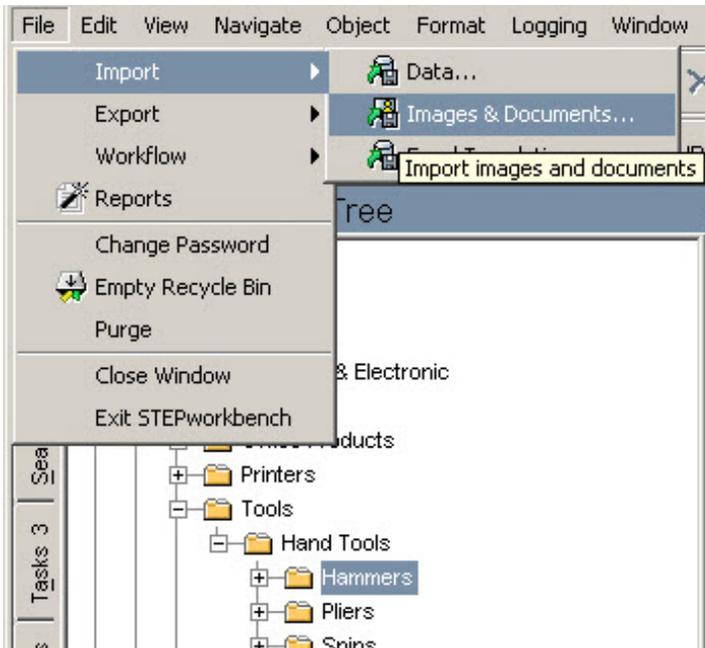
Select the Target Location

1. In the **Tree**, expand the relevant **Classification** hierarchy, and select the folder that you want to load the asset files into.



2. Right-click and select **Import Images & Documents** or from the **File** menu, point to **Import**, and then select **Images and Documents**.





3. The **Import Images and Document** wizard opens. The wizard guides you through 6 steps. It is not always necessary to go through all six steps. Often you can skip the later screen forms and proceed to the end to launch the import process.



Step 1 - Select Import Location

In the first screen you specify where the file or folder is located that contains the assets to be uploaded into STEP. This is also where you specify whether or not the asset file name should be used as the asset's ID in STEP (as well as the name).

1. In the **Specify ID** area, specify whether or not to use the asset's file name as the asset's ID in STEP.

The screenshot shows a Windows-style dialog box titled "Import Images and Documents". On the left, a "Steps" pane lists six steps: 1. Select Import Location (highlighted), 2. Select Asset Type, 3. Select Context, 4. Link to Product, 5. Overwrite Existing Assets, and 6. Import Overview. The main area is titled "Select Import Location" and contains the instruction "Specify ID(s) and location of the files to be imported". Below this, there are two sections: "Specify ID" with two radio buttons, "Use file name as ID" (unselected) and "Auto generate an ID" (selected); and "Location" with a text box containing "C:\Users\lotk\Pictures" and an ellipsis button (...). At the bottom, there are four buttons: "Back", "Next" (highlighted with a dashed border), "Finish", and "Cancel". A Stibo logo is visible in the bottom-left corner of the dialog.

2. For Location, click the ellipsis button (...) and use the file browser to navigate to the location of the assets that will be uploaded. When you have selected a folder, all assets within that folder are loaded to the location you specified before starting the wizard.
3. Click **Next** to continue.

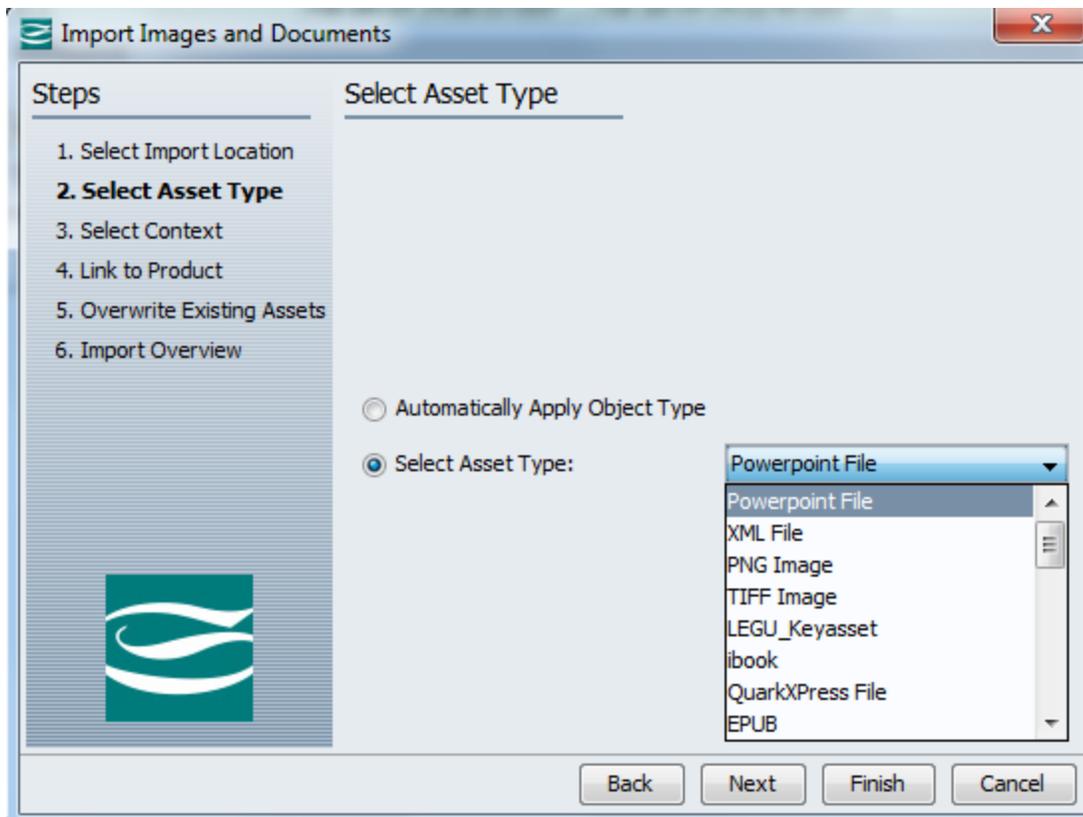
Step 2 - Select Asset Type

In the second wizard screen, you specify whether you want to manually apply an asset type to a specific object type, or if you want the system to determine each asset's object type.

1. Select **Automatically Apply Object Type** if you want the system to automatically apply an object type.

-OR-

Select **Select Asset Type** if you want to apply the asset type manually.



2. If you chose to select the asset type manually, select a type from the list. The available asset types depend on your system set up.
3. Click **Next** to continue to the next screen or **Finish** to start the import process without specifying any further parameters.

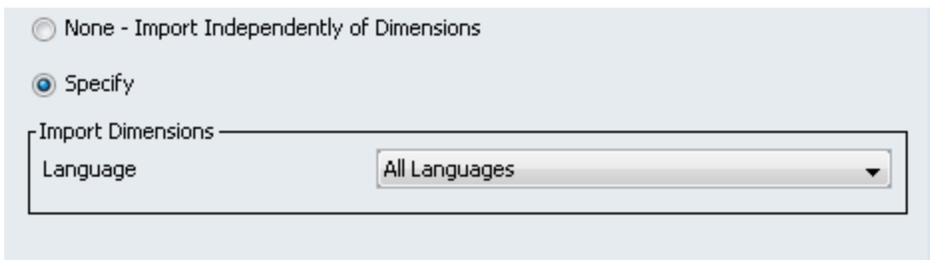
Step 3 - Select Context

In this screen you specify any language dimension dependencies. You typically specify a context when you are loading assets that are essentially the same but which are language-sensitive. Or you specify a context when publishing a catalog where there are illustrations with embedded texts, and there has to be different language versions of that illustration.

You must load these types of assets using the same asset ID. That is because when you do publication swaps in, for example, STEP'n'design, it can swap out not just product data, but any language-relevant images, too.

1. Choose **None** if you want to load the assets in a global context. Choose **Specify** if you want to select the context from the list in the **Import Dimensions** area.

It is standard to set up the language dimension for assets. However, set-ups may vary, and therefore you may be able to select more than one dimension. The list is taken from the language dimension point set-ups and you can choose each of the available dimension points, one per dimension.



The screenshot shows a dialog box with the following elements:

- Two radio buttons:
 - None - Import Independently of Dimensions
 - Specify
- A section titled "Import Dimensions" containing a dropdown menu labeled "Language" with "All Languages" selected.

2. Click **Next** to continue to the next screen or **Finish** to start the import process without specifying any further parameters.

Note: If you want to load an asset which is dependent on, for example, a country dimension, but you have only got the language dimension set up for assets in your system, you can upload that image for a context using a specific country. You do so in STEP from the Edit menu by using **Create Local Content of Asset** and then using the **Replace Asset Content**. This is done on an individual basis. It keeps the image IDs the same, and all the links from modules and/or products are maintained to the appropriate version of the image.

Note: When an asset has been replaced a major revision will be created with the changed image. The major revision enables the option to revert to an older revision of the asset.

Step 4 - Link To Product

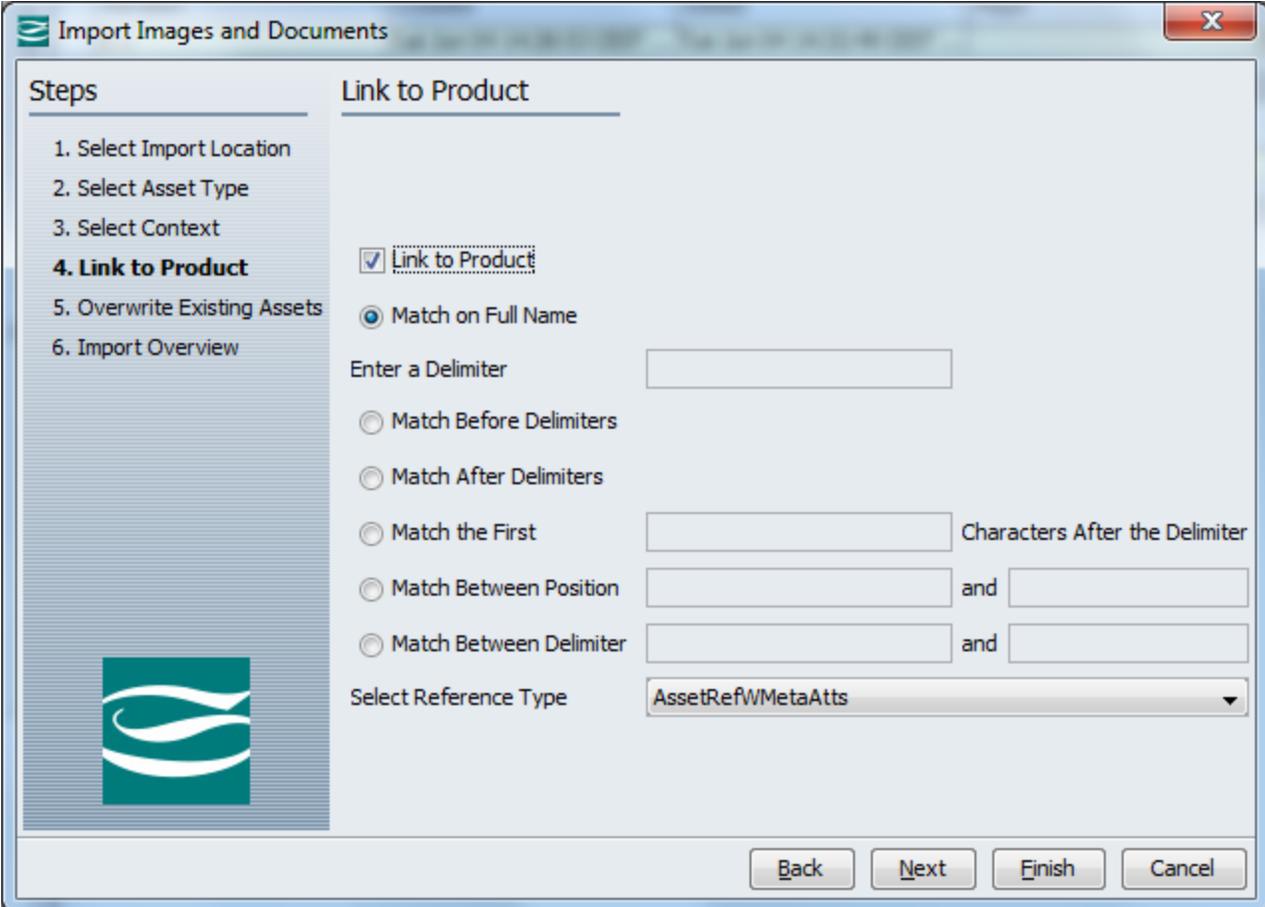
In this step you can create reference links between the assets you are loading and existing products in STEP.

1. If you want to create links, select **Link to Products**.
2. Choose **Match on Full Name**, if you want the system to try to match the full file name of the asset (minus the extension) to a product ID in STEP. For all successful matches, a link is created.

-OR-

In the **Enter a Delimiter** field, enter a delimiter that is used to delimit the assets file name.

3. If you chose **Enter a Delimiter**, choose one of the following options:
 - **Match Before Delimiters.** If the asset's file name is 17268_AS-56, and the delimiter is set to the underscore character (`_`), then the system tries to find the product with the ID of 17268.
 - **Match After Delimiters.** If the asset's file name is UK_177628, and the delimiter is set to the underscore character (`_`), then the system tries to find the product with the ID of 177628. If it finds that product it creates the link.
 - **Match the first x characters after the delimiter.** If the asset's file name is UK_9014514-ASH, and the delimiter is set to the underscore character (`_`), and the number of characters to match is set to 7, then the system tries to find the product with the ID of 9014514.
 - **Match Between Position.** Enter the start and ending character positions to use for making the match to a product ID. If the asset's file name is UK_447628ASP, and the starting position is set to 4 and the ending position is set to 11, then the system tries to find the product with the ID of 447628AS. If it finds that product it creates the link.
 - **Match Between Delimiter.** Enter the start and ending delimiters to use for making the match to a product ID. If the asset's file name is UK_44762877_EAS, and the starting delimiter is set to an underscore (`_`), and the ending delimiter is also set to the underscore, then the system tries to find the product with the ID of 44762877. If it finds that product it creates the link.



Import Images and Documents

Steps

1. Select Import Location
2. Select Asset Type
3. Select Context
- 4. Link to Product**
5. Overwrite Existing Assets
6. Import Overview

Link to Product

Link to Product

Match on Full Name

Enter a Delimiter

Match Before Delimiters

Match After Delimiters

Match the First Characters After the Delimiter

Match Between Position and

Match Between Delimiter and

Select Reference Type

4. From the **Select Reference Type** list, select the reference type that should be used for the link between the asset and the product.

The content of the list varies depending on the system setup. You can only choose one reference type, so all assets loaded in this session are linked to products using the selected reference type.

5. Click **Next** to continue to the next screen or **Finish** to start the import process without specifying any further parameters.

Methods for Creating Reference Links

- In STEP, you can manually link any asset to any existing product via any existing reference type in STEP.
- You can create a STEPXML file to create links between assets and products and load that file via the Import Data wizard.
- If you have the STEP Import Manager component, you can load a simple tab delimited or Excel file that holds the product ID and the asset ID, and create multiple reference links.

Step 5 - Overwrite Existing Assets

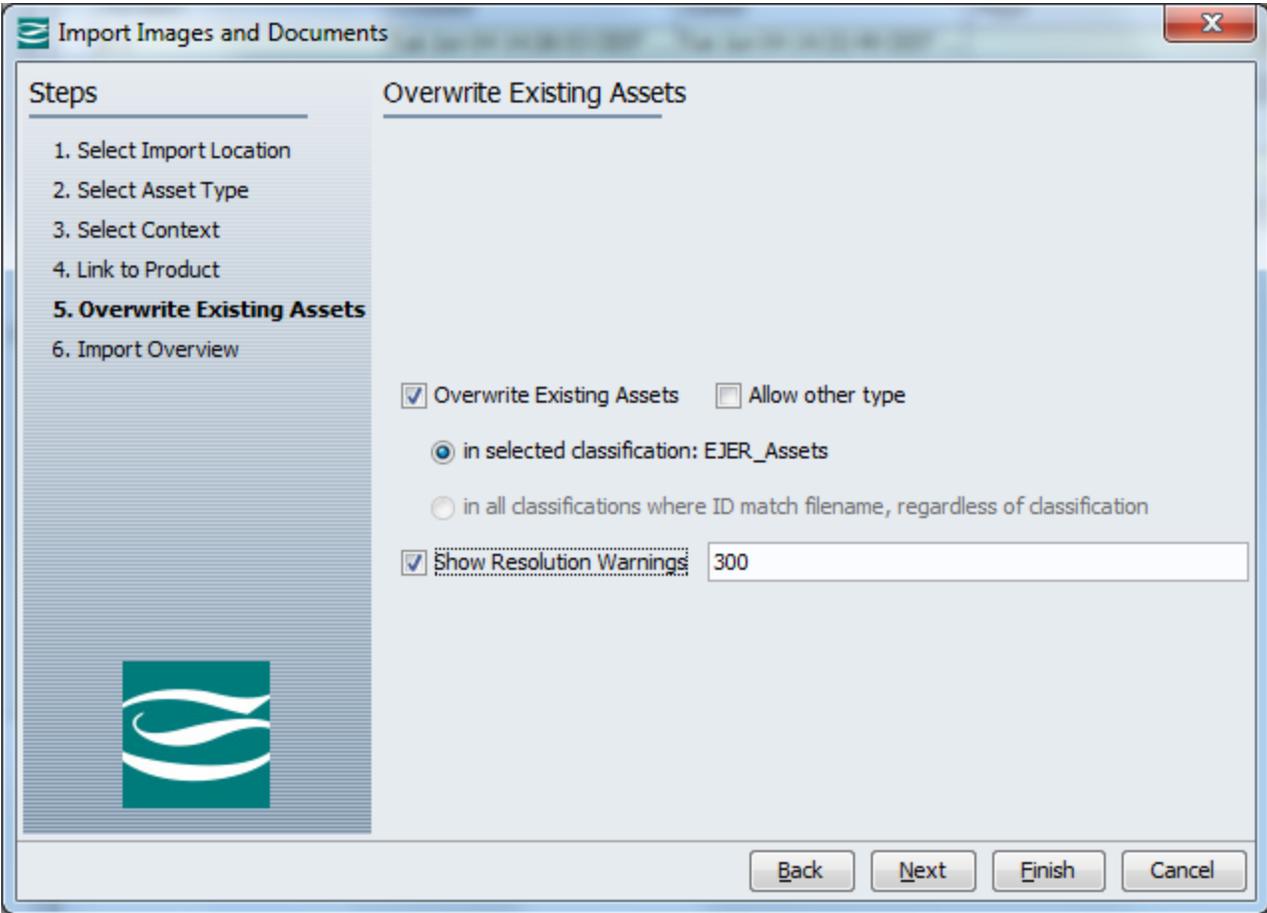
In this screen you specify whether the imported assets replace existing assets.

1. Select **Overwrite Existing Assets** option to replace images already existing in the database with images to be imported.
2. Select **Allow other type** to allow replacing images if the file extensions of the imported images do not match extensions of the images in the database. **An example**
3. Choose **in selected classification** to only replace existing images located in the classification the images are imported into.

-OR-

Choose **in all classifications where ID match filename, regardless of classification** to replace existing images located in any classification. Images already existing in the database are replaced with imported images no matter where the original images are located. The image is replaced in the classification where the original image is located.

4. If you want the imported images to meet a minimum resolution requirement, select **Show Resolution Warnings**, and then enter the preferred value. Note that all images are imported, but if the resolution of an image is lower than specified, a warning is reported.
5. Click **Next** to continue to the final screen.

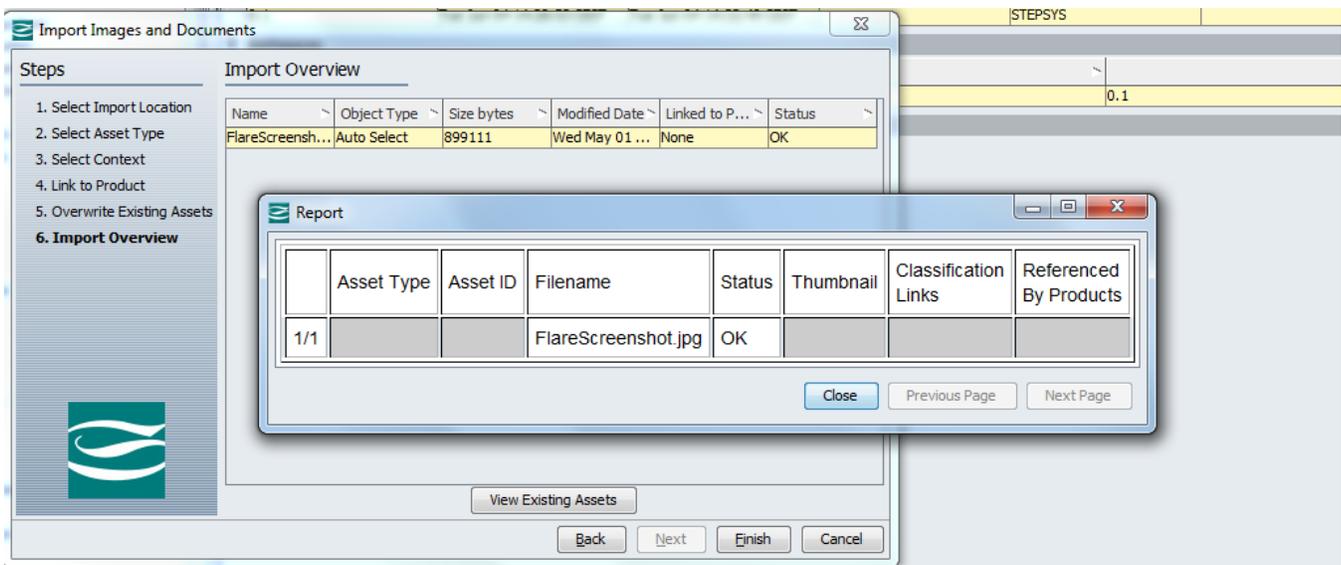


Step 6 - Import Overview

The final screen enables you to review some of the assets that you are about to load - provided that you selected a folder and not an individual file.

The **Import Overview** shows you the following information about the file about to be imported:

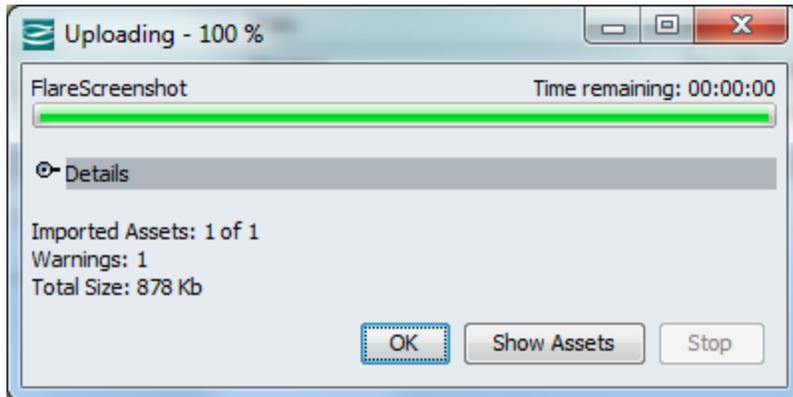
- The file name
 - The object type
 - The file size and date,
 - Any matching product ID
 - The status, that is, whether or not it will overwrite an existing asset.
1. To view a report the assets that are about to be imported as well as existing assets, click **View Existing Assets**. A report window appears.
 2. Click **Close** to close the report window.
 3. If you are satisfied with the overview, click **Finish** to start the import process.



Starting the Asset Import Process

- Click **Finish** to start the asset import process.

The **Import Status** window appears, and shows the progress of the process. As each asset is loaded, the screen updates with the percentage complete and the number of warnings.



- Click the **Details** flipper to view the status of each imported asset.
- Click **Show Assets**, to see a multi-view display including thumbnails of the imported assets.

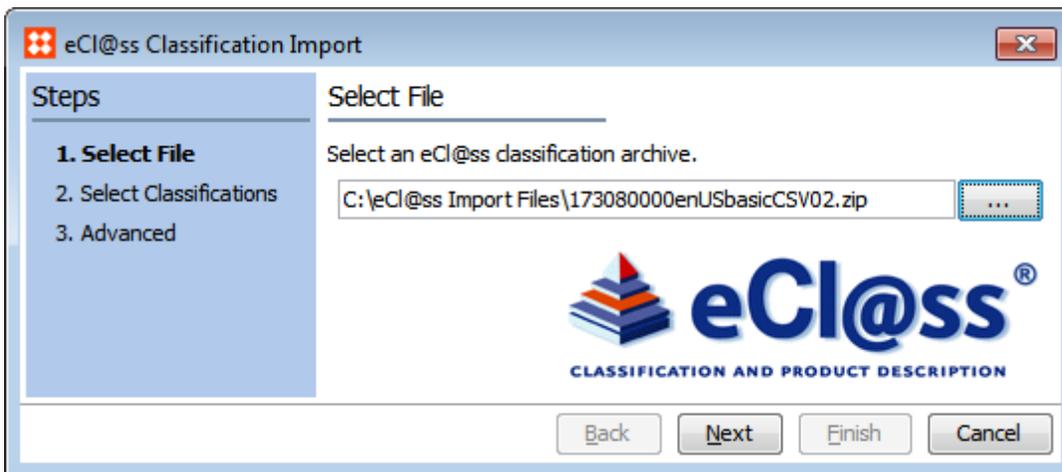
Importing an eCl@ss Hierarchy

eCl@ss is an international standard for the classification products and materials used for information exchange between suppliers and their customers.

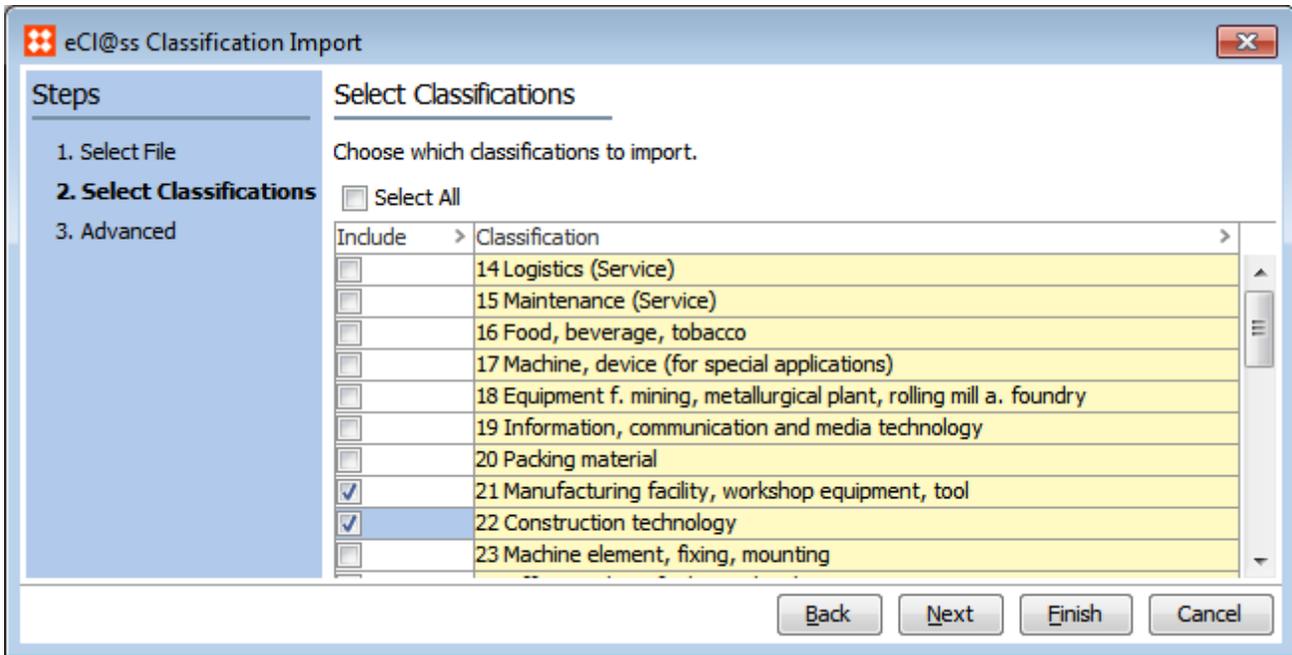
For detailed information about eCl@ss, go to www.eclass.de.

Import an eCl@ss Hierarchy

1. On the **File** menu point to **Import**, and then select **eCl@ss...**. The **eCl@ss Classification Import** wizard opens.
2. In step 1, **Select File**, click the ellipsis button (...), search or browse for the relevant archive, and click **Next**.

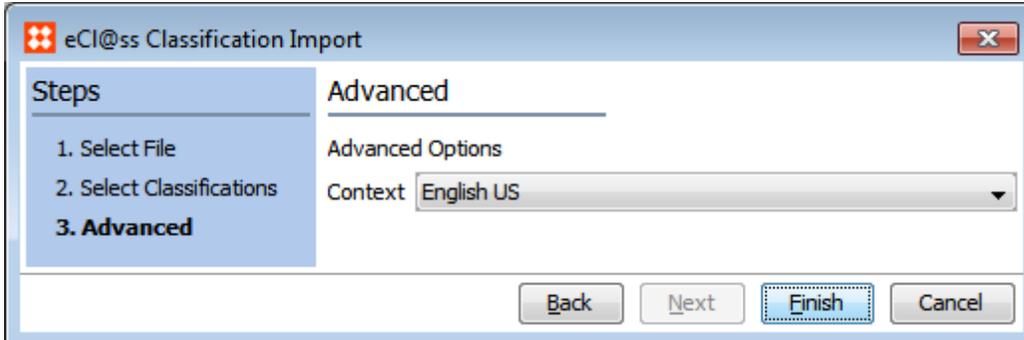


3. In step 2, **Select Classifications**, select the classifications you want to import, and click **Next**.



- Step 3, **Advanced**, enables you to populate the eClass classification with context dependent names that correspond with your available context.

Select the relevant context, and click **Finish**.



- If needed, refer to the information in **Starting the Import Process** to begin the import.

Map eCI@ss Attributes to STEP Attributes

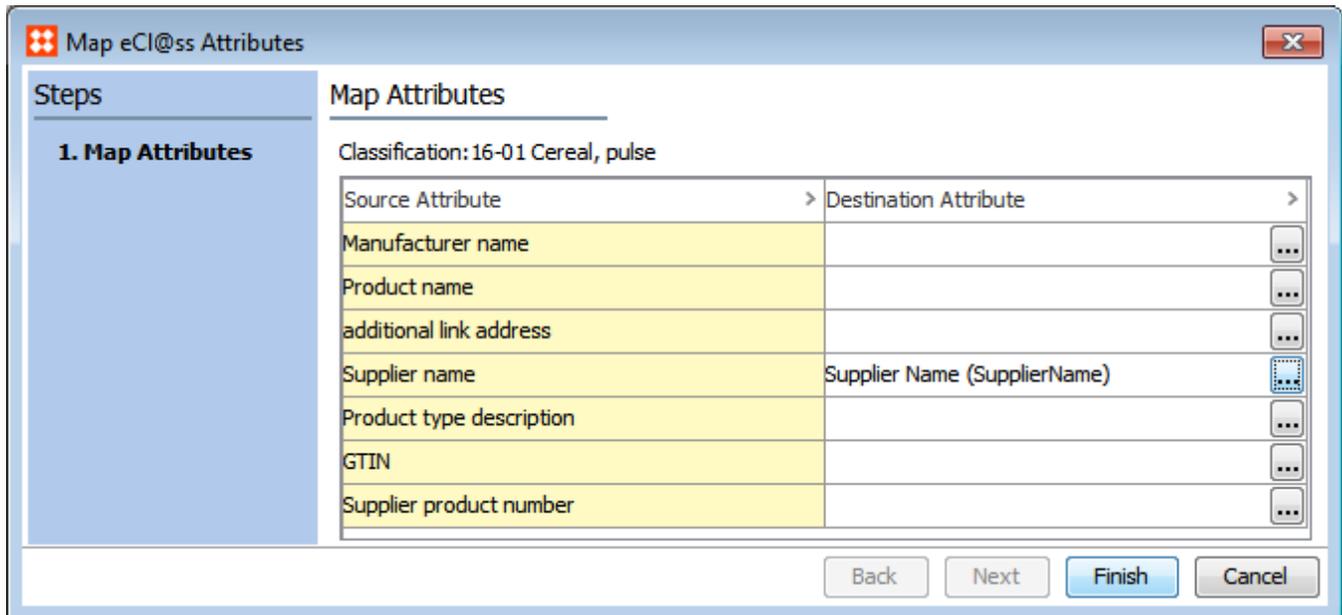
eCI@ss classifications include standard eCI@ss attributes that are inherited downwards in the hierarchical structure.

You can map eCI@ss attributes to STEP attributes. Mapping STEP attributes to the standard eCI@ss attributes merges the attributes. Therefore, there is no need to recreate already defined attributes to gain the benefits of using standard eCI@ss attributes.

The merge enables you to use the eCI@ss aspects of the attributes in your eCatalog by mapping the eCI@ss aspect of the attribute in your eCatalog configuration. For more information about eCatalog configurations, see [Creating an eCatalog Configuration](#) in the in the eCatalog documentation.

Note: The eCI@ss attribute will be deleted after the mapping since the mapping serves as a merge of attributes.

1. In the **Tree**, right-click the eCI@ss classification that you want to map attributes from, and then select **Map eCI@ss Attributes**. The Map eCI@ss Attributes wizard opens.



2. Next, click the ellipsis button (...) next to the relevant attribute, and the search or browse for the STEP attribute you want to map to.
3. Click **Select**, and then click **Finish**.

Checking Data Imports

When you launch a data import process, STEP starts a workflow process. It assigns a process number to that particular data import, and gives it the name you assigned when you saved the import.

Monitor the Import Process

When you want to monitor a particular data import process, you can:

- Click **Go To Process** in the **Starting Process** window, which appears when you launch the import.
- On the **BG Processes** tab, and select the relevant process. For data imports that have completed, look in **Ended Processes**. For data imports that are currently being performed, look in **Active Processes**. The process information is displayed.

Typically, you do not monitor the import process as it is being carried out. However, if the import process fails and you need to investigate why or if you need to contact Stibo Help Desk, the information can be useful.

The screenshot shows the 'BG Processes' window on the left and the 'Importing - Background Process' window on the right. The 'BG Processes' window has a tree view with categories like 'Queued Processes', 'Active Processes', 'Completed with Errors', and 'Ended Processes'. The 'Importing' category is selected. The 'Importing - Background Process' window shows a table of properties and an execution report.

Property	Value
Started by	USERJ
Id	BGP_179903
Description	Importing
Execution Server	doc-dev
Progress	Done
Status	succeeded
Created	Thu Jun 02 16:17:51 EDT 2016
Started	Thu Jun 02 16:17:56 EDT 2016
Finished	Thu Jun 02 16:17:57 EDT 2016
Processing Time	0 m 1 s
Time in Queue	0 m 5 s
# of warnings	0
# of errors	0

Execution Report

- 1 Retrieval started (Thu Jun 02 16:17:56 EDT 2016)
- 2 Retrieved 36864 bytes (Thu Jun 02 16:17:56 EDT 2016)
- 3 Conversion started (Thu Jun 02 16:17:56 EDT 2016)
- 4 Converted 5 objects (Thu Jun 02 16:17:56 EDT 2016)
- 5 Logged on
- 6 Mapping started (Thu Jun 02 16:17:56 EDT 2016)
- 7 Mapping completed (Thu Jun 02 16:17:56 EDT 2016)
- 8 Import Started (Thu Jun 02 16:17:56 EDT 2016)
- 9 Logged On
- 10 Using import mode "domain"
- 11 Starting first import pass (creating system setup objects)
- 12 Starting second import pass (importing data)

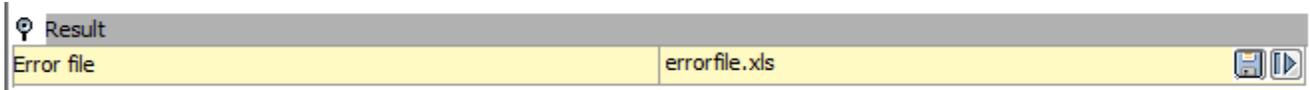
Execution Report

The execution report gives information about the data import, including validation errors found during import.

When the following errors are reported, the creation of new objects is skipped entirely:

- Specified parent object does not exist
- Specified Object Type is not valid below specified parent
- No ID is provided and no ID pattern is specified for the target Object Type

When an import process completes with errors, a file is created that contains just the rows with erroneous data. The file can be saved to disk or opened directly in the Import Manager. Use this error file to correct the data in the import file and run the import process again.



Typically, an execution report can be very long, and it can be useful to copy and paste the information into a standard text editor. You can then go through the file to locate errors and warnings.

When a data error is encountered, the error message reports the data it was attempting to load, including the attribute name and the object ID for the value, and the type of error encountered. The line number and column number are also given, but that is not always useful since it refers to the interim XML file that is created.

Types of Error Messages

The following table shows most of the different types of error message that you are likely to encounter, with a brief explanation of the most common reason for the error.

The error messages marked with an asterisk (*) are errors that you will only encounter when you load an XML file. That is because in an XML import, the wizard does not prevent you from entering an invalid value. However, when importing CSV or XLS files, since you often select values from a list, you cannot select an invalid value.

Error Message	Description
Illegal value x for attribute y in product z	The value in the load file does not meet the validity or other constraint that is set up for the attribute.
Not in legal values list	You tried to load an attribute value into an attribute that uses an LOV, and that value does not exist in the LOV, and it is set up not to accept additional entries.
Attribute not legal for object type	The product's object type is not valid for the attribute. That is, the attribute is not allowed to hold a value for products with the specified object type.
*Illegal unit x for attribute y in product z	In the XML file, you have tried to assign a non-existing unit to an attribute.
Unknown Classification x in product y.	In the XML file, you have probably requested a product-to-classification link using a specified reference type, but the classification you specified is not found.
Unknown Parent product x for product y	You tried to assign a product to a parent ID, but that parent ID cannot be found in the system.
*Unknown product reference target	You attempted to link one product to another via a reference type, but the target product does not exist.
*Unknown attribute	You specified an attribute in the XML file, but that attribute does not exist.
Unknown unit	The unit encountered in the load file does not exist in STEP.

Error Message	Description
Rejected new product x	You set the option to "Reject New Products" in the wizard, and a new product was encountered in the load file.
Illegal unit for attribute	The unit exists but is not valid for the attribute you are trying to load the value into.
*Illegal Object Type for product position	You tried to create a new product via an XML load and you specified an object type that is not a valid choice, given the parent product that you specified.
*Unknown Object Type	You specified an object type (e.g. for a product) in the XML file, but that object type does not exist.
*Unknown reference Type	You specified a reference type (e.g. for a product-to-product reference) in the XML file, but that reference type does not exist.
<ul style="list-style-type: none"> • Insufficient privileges to ... 	<p>You do not have permission to carry out one of the following actions (specified in the error message):</p> <ul style="list-style-type: none"> • set value • set name • create product • move product • modify Object Type • classify product • create product reference
Product x contains more than one reference to product y of Type z, reference data will not be imported.	You cannot make a product-to-product reference from the same source product to the same target product more than once.
Product x contains more than one reference to asset y of Type z, reference data will not be imported.	You cannot make a product-to-asset reference from the same source product to the same target asset more than once.
*Classification x contains more than one reference to	You cannot make a classification-to-classification

Error Message	Description
Classification y of Type z, reference data will not be imported.	reference from the same source classification to the same target classification more than once.
*Reference Type x is not valid for importing a product cross reference from a to b.	In your XML file, you have asked for a product reference that refers to a non-existent reference type.
*Reference Type x is not valid for importing an asset cross reference from a to b.	In your XML file, you have asked for an asset reference that refers to a non-existent reference type.
*Reference Type x is not valid for importing a Classification cross reference from a to b.	In your XML file, you have asked for a classification reference that refers to a non-existent reference type.
*Product x already contains a reference to Classification y but it is invisible in the specified context. The reference and any data on it will not be imported.	You have probably made a product-to-classification link dimension-dependent, and the context you are importing into cannot "see" that link, but it is still there.

Error Message Examples

The most common errors encountered during data import refer to an invalid attribute value. Sometimes there is a mismatch with the attribute's validation type, other times there is a mismatch with one of the other setups. For example, the attribute's constraints, such as masks, minimum and maximum values, maximum length, LOVs, the object's own object type, and so on. Therefore, when you encounter an error, look at the attribute setup and review the constraints.

Even if you load attribute values for an object, not all values will be automatically available in STEP. Attributes must be made valid for an object before the values can be accessed via references. STEP will always load attribute values if the object's type is valid for the attribute, and the attribute values meet the validity criteria. But, the attribute itself must be a valid attribute for that object, that is, linked somewhere in the object hierarchy or classification hierarchy where the object resides.

Use the following examples to assist in troubleshooting problems with data imports.

Illegal Unit for Attribute

The tab-delimited input file below had two columns mapped: the first column was mapped both to the Product ID and Product Name, and the second column was mapped to the attribute 'Description'.

```
T100-3526      RED 1/2" STRAIGHT WIDGET SSU05PL
T100-2625      BLUE 1/4" ANGLED GADGET BZT10GL
T100-2827      ORANGE 3/4" ELL WHATSIT PLU04HF
```

When the file was imported, the execution report showed three illegal unit errors:

- Illegal unit "1/2" STRAIGHT WIDGET SSU05PL" for attribute Description in product T100-3526 (UnknownUnit).
Source: Line=2, Column=Column2.
- Illegal unit "1/4" ANGLED GADGET BZT10GL" for attribute Description in product T100-2625 (Unknown Unit).
Source: Line=3, Column=Column2.
- Illegal unit "3/4" ELL WHATSIT PLU04HF" for attribute Description in product T100-2827 (Unknown Unit).
Source: Line=4, Column=Column2.

Upon inspection of the attribute with the name of 'Description', it was found that the assigned data validity check was 'Numeric_Text', and that it had a unit assigned to it: mm. Further, it was not selected as the default unit.

STEP could not handle the input text, since the attribute's validity was defined as being NumericText and had only one legal unit, and that unit could not be matched with anything in the attribute value provided by the import. In this case, either the attribute's setup is incorrect, or a different product attribute should be used that has an appropriate data validity type assigned.

Unknown Parent Product

The tab-delimited input file includes three columns. Column 1 was mapped to both Product ID and Product Name (new products), and Column 3 was mapped to the Parent ID.:

```
T100-3526      RED 1                               Hand Tools
T100-2625      1mm ANGLED GADGET                   Hand Tools
```

T100-2827 ORANGE 3mm Hand Tools

When the file was imported, the execution report showed the following three errors:

- Unknown Parent Product "Hand Tools" for product "T100-3526" (1 product(s) was rejected due to this error). Source: Line=2, Column=Column3.
- Unknown Parent Product "Hand Tools" for product "T100-2625" (1 product(s) was rejected due to this error). Source: Line=3, Column=Column3.
- Unknown Parent Product "Hand Tools" for product "T100-2827" (1 product(s) was rejected due to this error). Source: Line=4, Column=Column3.

STEP could not find the Parent IDs given in Column 3 of the input file, and therefore the new products were rejected. Even though a default Parent ID was selected in the Identify Destination screen of the import wizard, STEP accepts the value for the Parent ID in the input file as an override, and rejects the new product anyway.

A reason for this execution report error could be that one of the input file's columns was mapped to be the Parent ID, and STEP could not find it. If the product existed already, it would not be moved anywhere, it would stay with its current parent. If the product did not exist, it would not be created.

Maximum Length, Illegal Value

The input file had two columns mapped: the first column was mapped both to the Product ID and Product Name (these were new products), and the second column was mapped to the attribute "Manufacturer Part Number".

G100-352654	REDD-52626278-YDS-777777-1/FFGH00KK7733249-UHP
J100-267225	REDD-5262627
D100-289827	K7733249-UHP

The file was imported and the execution report showed the following three errors:

- Illegal value "REDD-52626278-YDS-777777-1/FFGH00KK7733249-UHP" for attribute Manufacturer Part Number in product G100-352654 (Length of the value exceeds max length of domain Is trying to insert a value with '46' characters into a domain that has a maximum length of '40' characters.). Source: Line=10, Column=Column2.
- Line= at line 9: : Length of the value exceeds max length of domain Is trying to insert a value with '46' characters into a domain that has a maximum length of '40' characters.
- Line= at line 9: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0

These errors are reporting a single problem. The first and second errors say that one of the imported part numbers had a value in the Manufacturer Part Number column that was too long, as defined by the attribute's setup. You are given the part number, the attribute name, and the value that was trying to be imported. The third error message is useful for a programmer.

Note: This type of error does not prevent the new part number from being created. The value for the Manufacturer Part Number, however, is left blank.

Out of Range, Illegal Value

Here is another example of an illegal value error:

- Illegal value "250" for attribute Focal Length (ft) in product G100-352654 (Value error: Value '250' is out of range). Source: Line=9, Column=Column2.
- Line= at line 8: : Value error: Value '250' is out of range
- Line= at line 8: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0

This group of three error messages are reporting a single problem. The first two messages indicate that a value of 250 is too large for the maximum value allowed for the attribute.

Not a Number, Illegal Value

The following tab-delimited input file had two columns mapped: the first column was mapped to the Product ID (the products already existed in STEP), and the second column was mapped to an attribute called 'Hole Diameter'.

```
G100-352654      1-1/2
J100-267225     2-3/4
D100-289827     17-3/16
```

The file was imported and the execution report showed the following nine errors:

- Illegal value "1-1/2" for attribute Hole Diameter in product G100-352654 (Value error: Value '1-1/2' is not a number). Source: Line=9, Column=Column2.
- Line= at line 8: : Value error: Value '1-1/2' is not a number
- Line= at line 8: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0
- Illegal value "2-3/4" for attribute Hole Diameter in product J100-267225 (Value error: Value '2-3/4' is not a number). Source: Line=14, Column=Column2.
- Line= at line 13: : Value error: Value '2-3/4' is not a number
- Line= at line 13: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0
- Illegal value "17-3/16" for attribute Hole Diameter in product D100-289827 (Value error: Value '17-3/16' is not a number). Source: Line=19, Column=Column2.
- Line= at line 18: : Value error: Value '17-3/16' is not a number

- Line= at line 18: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0

Each set of three errors addresses one of three problems. That is, different information is reported for the same error in three individual messages. The first message in each group is what is most useful to the end user. In the above series, notice that the first lines that indicate that the fractional value in the input file is not allowed for the attribute 'Hole Diameter' because the value 'is not a number'. In this case, the attribute's validation type was set to Number, which does not allow fractional values.

Here is another example:

- Illegal value "Hand tools" for attribute Focal Length (ft) in product J100-267225 (Value error: Value 'Hand tools' is not a number). Source: Line=14, Column=Column2.
- Line= at line 13: : Value error: Value 'Hand tools' is not a number
- Line= at line 13: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0

The first message in the group tells you that the attribute value of 'Hand tools' in the input file is not allowed since the attribute is set up to expect a number.

And another example:

- Illegal value "50,70" for attribute Focal Length (ft) in product D100-289827 (Value error: Value '50,70' is not a number). Source: Line=19, Column=Column2.
- Line= at line 18: : Value error: Value '50,70' is not a number
- Line= at line 18: : Array operation failed (1 times): Error during array operation: ORA-20291: Parent key not found (valuemap.edgeid) ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 18 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20291; SQL return code=0

The first message in the group tells you that the attribute value of '50,70' in the input file does not match the attribute's validation type of Number, which does not allow commas.

Object Existed with Another Object Type

The following tab-delimited input file had two columns mapped: the first column was mapped to the Product ID (the products already existed in STEP), and the second column was mapped to an attribute called "MFOB".

```
G100-352654    Yes
J100-267225   No
D100-289827   Maybe
```

The file was imported and the execution report showed the following three warnings and one error:

- Line=2: Product 'G100-352654' existed with another object type - supplied object type not set
- Line=3: Product 'J100-267225' existed with another object type - supplied object type not set
- Line=4: Product 'D100-289827' existed with another object type - supplied object type not set

- Illegal value "Maybe" for attribute MFOB in product D100-289827 (Not in legal values list). Source: Line=4, Column=Column2.

The first three warnings say that the products already existed in STEP with the object type of 'Product'. When the file was imported, the Identify Destination screen was skipped, and the default object type that was set in that screen was different from Product. STEP reported that it did not change the object type of the products.

The final message is an error that says that the attribute MFOB used an LOV and the value in the input file was not valid. Further investigation showed that LOV had only two valid values: Yes and No. Additionally, the LOV was set up so that no modifications could be made to the LOV by loading in data. So the value "Maybe" was rejected.

Unknown Classification

The following tab-delimited input file had two columns mapped: the first column was mapped to the product ID (the products already existed in STEP), and the second was mapped as a classification ID.

```
G100-352654      Hand tools
J100-267225     Hand tools
D100-289827     Hand tools
```

The file was imported and the execution report showed the following three errors:

- Unknown Classification "Hand tools" in product "G100-352654" (Classification 'Hand tools' does not exist in the import workspace). Source: Line=6, Column=Column2.
- Unknown Classification "Hand tools" in product "J100-267225" (Classification 'Hand tools' does not exist in the import workspace). Source: Line=10, Column=Column2.
- Unknown Classification "Hand tools" in product "D100-289827" (Classification 'Hand tools' does not exist in the import workspace). Source: Line=14, Column=Column2.

In this case, the import file requested that the products be linked to a classification folder, but STEP could not find a classification with the specified ID 'Hand tools'.

Not Privileged to Create Object

A file was imported and the execution report showed the following error:

- Line=2: Not privileged to Create product

Two scenarios can cause this error message:

1. Importing new products when the import user does not have the privilege to create new products. The import user must have all access required for the action to be performed during the import.
2. Importing a product with a product-to-product Reference Type by mapping one column of data to the Product ID and another column to a Reference, while the target product of the reference does not already exist in STEP. A reference can only be imported when the target already exists.

Unknown Asset ID

A file was imported and the execution report showed the following error:

- Line=2: Unknown Asset ID: AC200-627

This error is reported when attempting to link a Product ID to an Asset ID, but the asset does not exist in STEP.

Note: The Asset ID (not the Asset Name) is required in an import file.

Array Operation Failed

A file was imported and the execution report showed the following error:

- Line= at line 8: : Array operation failed (1 times): Error during array operation: ORA-20042: Value rejected: Attribute Gap not valid for this usertype ORA-06512: at "STEPSYS.VALUEMAP_BEF_INS", line 41 ORA-04088: error during execution of trigger 'STEPSYS.VALUEMAP_BEF_INS' ; SQL return code=20042; SQL return code=0

This error is reported when attempting to load an attribute value (not shown in the error) to an attribute that is not legal for the product's object type. Upon further investigation, we found that the product's object type was 'Product'. The error message is saying that the attribute that has the ID of 'Gap' is not valid for products with the object type of 'Product'. Since the value is invalid, it is ignored.

Data Quality in the Import Manager

The Import Manager comes with a number of data quality features.

- Functionality that allows you to easily spot errors in the data that you want to import.
- Extended functionality for manipulating data prior to import.
- Functionality for automatically replacing words and values in import data.
- The ability to apply predefined sets of transformations on import data
- Improved handling of errors in import background processes.

Discovering Data Import Quality Issues

The Import Manager provides data quality functionality that enables you to locate errors in the data you want to import. The functionality applies to row and column based import formats such as Excel, CSV, and FixedWidth.

To Import a Data Profile

1. From the **File** menu, open the Import Manager and go through steps 1-3. For more information about these steps, see [Data Import Manager Wizard](#) in the Import Manager guide.
2. On Step 4, **Map Data**, click **Generate Profile**.

Note: To prevent out of memory exceptions, use the property `Import.Profile.Max.Products` to limit the number of profiled rows. The default value for the property is 10.000. If the number of rows in the import file exceeds the limit, the remaining rows are not profiled.

Import Manager

Steps

- Select Configuration
- Select Data Source
- Select Format
- Map Data**
- Identify Objects
- Identify Destination
- Select Business Rules
- Advanced Settings

Map Data

Source:

<ID>	<Name>	Banbredd	Benämning	BK04-kod	Bygghöjd	Bygghöjd tapp	Bärighet	Diar
780001	56540800	40 mm	Länkhjul 40 x 40 ...	08204	55 mm		30 kg	
780002	56580806	44 mm		06101	64 mm		45 kg	
780003	Transporthjul							
780005	116150707	22 mm	HJUL FAST 50X22...	08204	65 mm		30 kg	
780006	116150806	23 mm	HJUL FAST 65X23...	08204	84 mm		35 kg	
780008	116150103	14 mm	Plastlänkhjul Typ ...	08204	34 mm		10 kg	
780009	116150202	14 mm	Plastlänkhjul Typ ...	08204	37 mm		15 kg	
780010	116150301	17 mm	Plastlänkhjul Typ ...	08204	54 mm		20 kg	
780011	116150509	23 mm	Plastlänkhjul Typ ...	08204	84 mm		35 kg	
780012	116150608	23 mm	Plastlänkhjul Typ ...	08204	102 mm		50 kg	
780013	116150400	22 mm	Plastlänkhjul Typ ...	08204	65 mm		30 kg	
780015	116130105	13 mm	OTTOMANHJUL 2...	06101	28 mm		15 kg	
780016	116130402	16 mm	OTTOMANHJUL 4...	06101	43.5 mm		30 kg	

Result:

Map to: Product

ID = <ID>	Name = <Name>	BBtool_Banbr...	BBtool_Descri...	BBtool_BK04 c...	BBtool_Byggh...	BBtool_Byggh...	BBtool_Barigh...	BBt
780001	56540800	40 mm	Länkhjul 40 x 40 ...	08204	55 mm		30 kg	
780002	56580806	44 mm		06101	64 mm		45 kg	
780003	Transporthjul							
780005	116150707	22 mm	HJUL FAST 50X22...	08204	65 mm		30 kg	
780006	116150806	23 mm	HJUL FAST 65X23...	08204	84 mm		35 kg	
780008	116150103	14 mm	Plastlänkhjul Typ ...	08204	34 mm		10 kg	
780009	116150202	14 mm	Plastlänkhjul Typ ...	08204	37 mm		15 kg	
780010	116150301	17 mm	Plastlänkhjul Typ ...	08204	54 mm		20 kg	
780011	116150509	23 mm	Plastlänkhjul Typ ...	08204	84 mm		35 kg	
780012	116150608	23 mm	Plastlänkhjul Typ ...	08204	102 mm		50 kg	

Buttons: Auto Map, Map, Constant, Remove, Transform, Generate Profile, Back, Next, Finish, Cancel

The Import Data profile consists of an Overview and a Details page.

Overview

The **Overview** is made up of three widgets.

- **Object Count:** displays the number of profiled objects
- **Most Complete:** displays the 10 most complete columns
- **Least Complete:** displays the 10 least complete columns

Import Data Profile

Overview | Details

Object Count

64
Objects

Most Complete Column

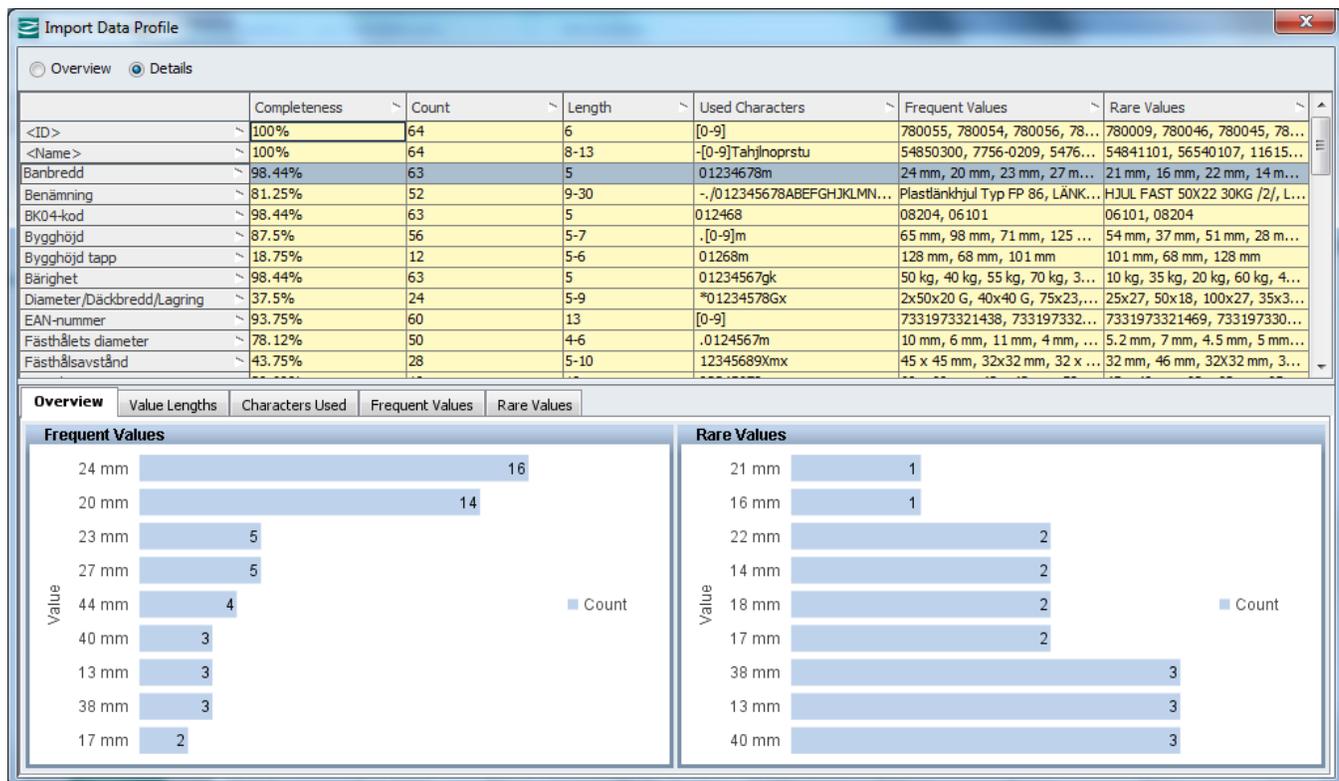
Column	Completeness
<ID>	100%
<Name>	100%
Banbredd	98.44%
BK04-kod	98.44%
Bärighet	98.44%
Förpackningen kan brytas	98.44%
Grundenhet	98.44%
Grundenhet innehåll	98.44%
Grundenheter per förpackning	98.44%
Hjuldiameter	98.44%

Least Complete Column

Column	Completeness
Hjul dimension	0%
Hjullagring	6.25%
Fästplattans yttermatt, länkhjul	14.06%
Bygghöjd tapp	18.75%
Fästplatta, L x B	29.69%
Tappens längd	31.25%
Tappens diameter	32.81%
Diameter/Däckbredd/Lagring	37.5%
Fästhålsavstånd	43.75%
Hjultyp	50%

Details

The Details page is divided into a master section at the top and a details section at the bottom. In the master section, every column of the import file is represented as a row.



The columns in the master section contain the following information.

Column	Description
Completeness	Shows the degree to which the attribute is populated in the category rounded to the nearest whole percent. Inherited values are included in the calculation.
Count	Lists the number of products below the category where the attribute has a value (including inherited values) and the number of products below the category where the attribute could possibly have a value.
Used Characters	List of the characters used for values in the column
Length	Displays value lengths, for example, 3-99 or 10.
Frequent	Displays a comma-separated list of the most frequently appearing values. The list is sorted

Column	Description
Values	ascending on frequency.
Rare Values	Comma-separated list of the least frequently appearing values. Sorted descending on frequency.
Frequent Patterns	This tab lists the most frequent patterns of the values of a given attribute together with the number values that match the pattern. The pattern describes the structure of a value.
Rare Patterns	This tab lists the most frequent patterns of the values of a given attribute together with the number values that match the pattern. The pattern describes the structure of a value.

When you select a row in the master section, the details section at the bottom is populated. The five tabs present a different view of the information that is displayed in the master section.

- **Overview** - Displays widgets for length distribution, frequent values, and rare values bar charts. If all values are of the same length, the length distribution bar chart is not displayed. Likewise, if only a single unique value exists in the import column, the frequent and rare values widgets are not be displayed.
- **Value Lengths** - Displays the number of occurrences of each value length.
- **Characters Used** - Displays the number of occurrences of each character.
- **Frequent Values** - Lists the most frequently appearing values.
- **Rare Values** - Lists the least frequently appearing values.

On the Fly Data Validation

When you map columns in the import file to data in STEP with the Data Quality functionality, the mapped data is validated on the fly. The data validation status is displayed with an icon on the header of each column in the Result area of the **Map Data** step. Cells with validation errors are highlighted with red background color.

Result: Map to: Product

...	BBtool_EAN n...	BBtool_Fastha...	BBtool_Fastha...	BBtool_Fastpl...	BBtool_Fastpl...	BBtool_Packin...	BBtool_Base u...	BBtool_Base u...
7331973321230	5 mm	32 x 32 mm		42 x 42 mm	1	ST	0	
7331973321278	5 mm	32 x 32 mm		42 x 42 mm	1	F	0	
7331973320899	6 mm	46 x 35 mm	58 x 47 mm		1	F	0	
7331973320905	7 mm	51 x 38 mm	Unit "x 35 mm" not valid for attribute "Fästhålsavstånd"			F	0	
7331973320837	4 mm	29 x 23 mm	37 x 34 mm		1	F	0	
7331973320844	4 mm	29 x 23 mm	37 x 34 mm		1	F	0	
7331973320851	5 mm	35 x 28 mm	45 x 40 mm		1	F	0	
7331973320875	7 mm	51 x 38 mm	65 x 54 mm		1	F	0	
7331973320882	6 mm	46 x 46 mm	60 x 60 mm		1	F	0	

If you hover your mouse over a cell with validation errors, the hover text provides information about the problem. If you apply transformations to a result column, the validation feature is also available in the Transformation Preview. Data is re-validated with every change.

Important: Only data for the first 200 rows in the import file is validated and displayed in the result panel.

It is possible to start an import process even if validation errors have been discovered.

Using Transformations to Improve Data Quality

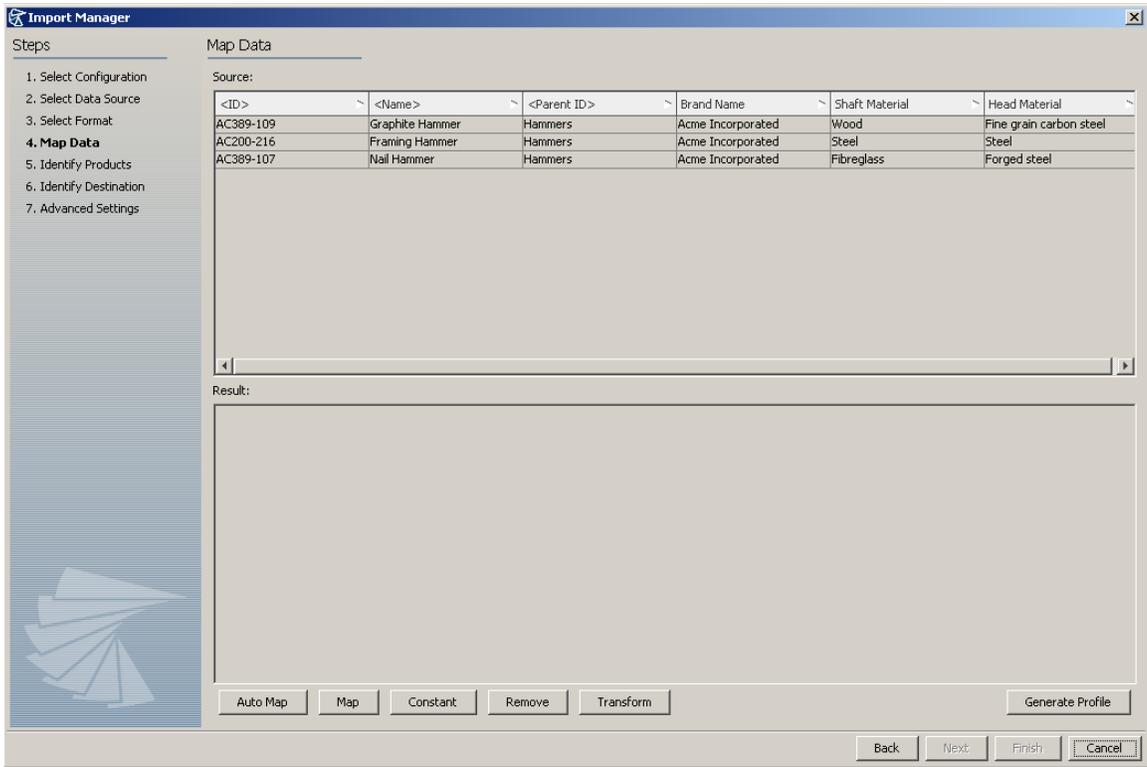
It is possible to use transformations to improve the quality the data you want to import. For general information about transformations, see the **Transformations Overview** section of the **Importing Data and Assets** documentation.

Concatenating Data Columns

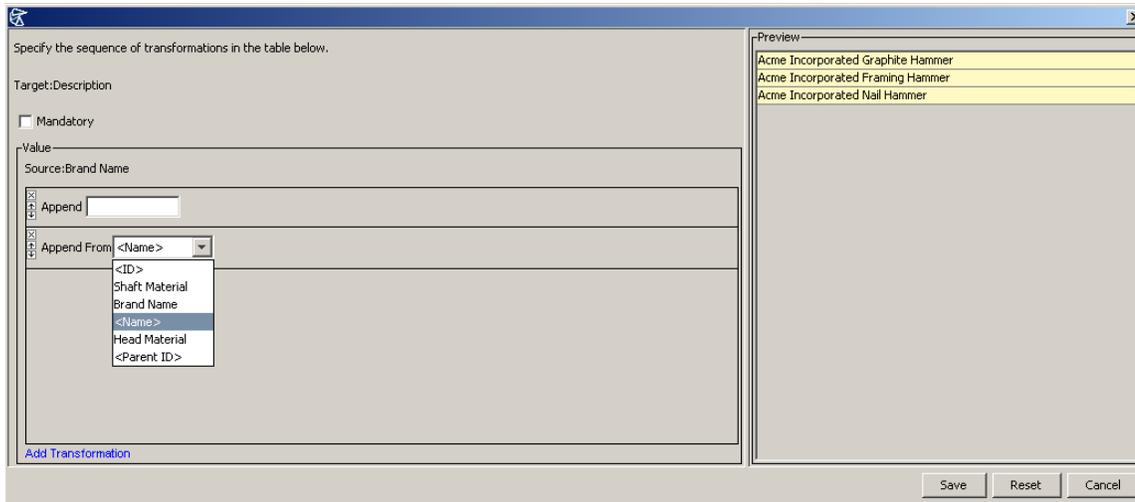
In step 4 of the Import Manager wizard, Map Data, you can concatenate data from different columns and map the result to a single entity in STEP. You can do this for row and column based import formats such as Excel, CSV, and FixedWidth.

To Merge Data into Single-Valued STEP Data

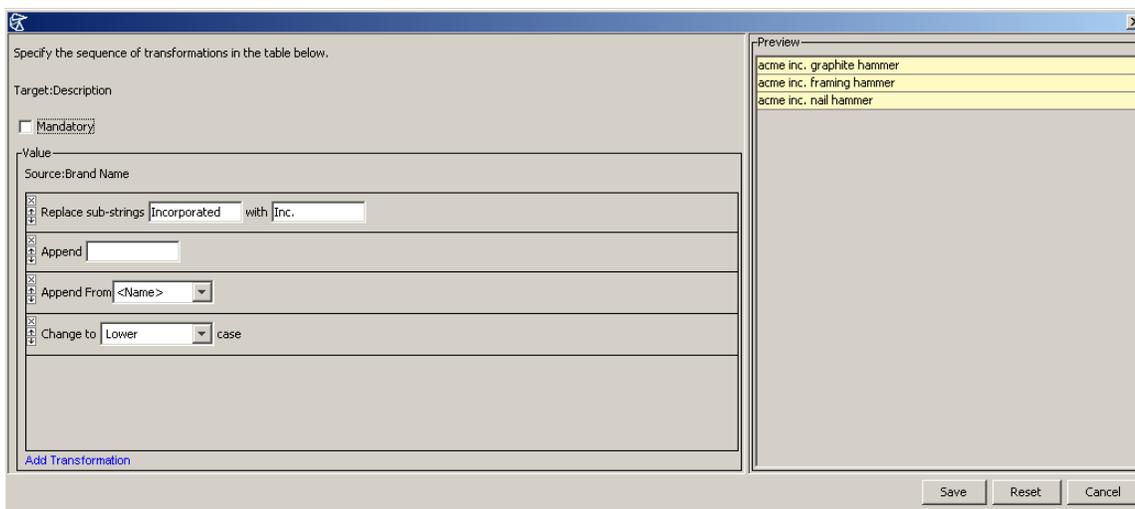
In this example, you have an import file where you want to concatenate the data in the "Brand Name" column with the data in the "<Name>" column and map it into an attribute named Description.



1. Select the "Brand Name" column in the **Source** area and map it to the "Description" attribute.
2. In the **Result** area select the "Description = Brand Name" column and click **Transform**.
3. In the **Transformations** dialog , add the transformation **Append from source** and select the "<Name>" column.



In this example, a space has been added to the "Brand Name" value before the **Append from source** transformation to separate the concatenated strings. You can add any number of transformations before the **Append from source** transformation. Also, you can add transformations that are applied to the entire string.



Note: If you need to prepend values from a source or intermediate variable column, a **Prepend from source** transformation is available.

To Merge Data into Multi-Valued Data

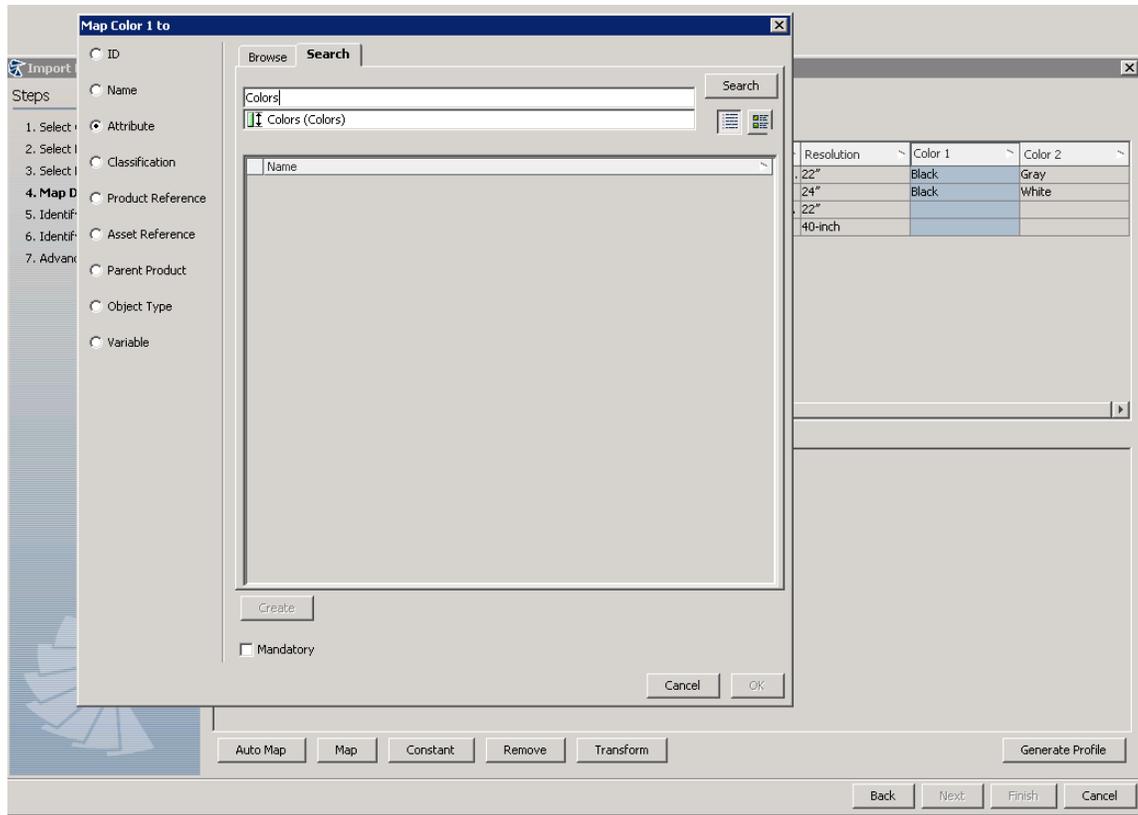
Multiple values for a single multivalued attribute and multiple targets for a reference or link type can be imported from a single field in the import file. In this case when using a CSV, the values / Target IDs must be separated by a

delimiter that is different from the delimiter used for separating columns (alternately the value can be enclosed in quotes).

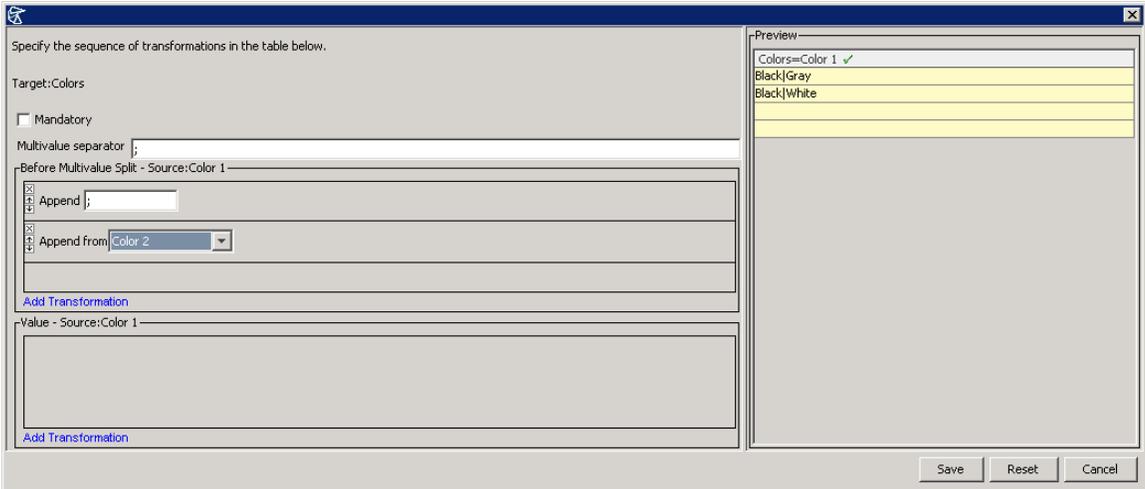
If STEP does not automatically recognize the import data as a 'multi value' after the data has been mapped to either a multivalued attribute or a reference / link type that allows multiple targets, the result column can be transformed and the delimiter / separator specified as shown below.

In this example, you are going to merge the values of the two columns "Color 1" and a "Color 2" into a single multi-valued attribute named "Colors".

1. Map "Colors 1" to "Colors".

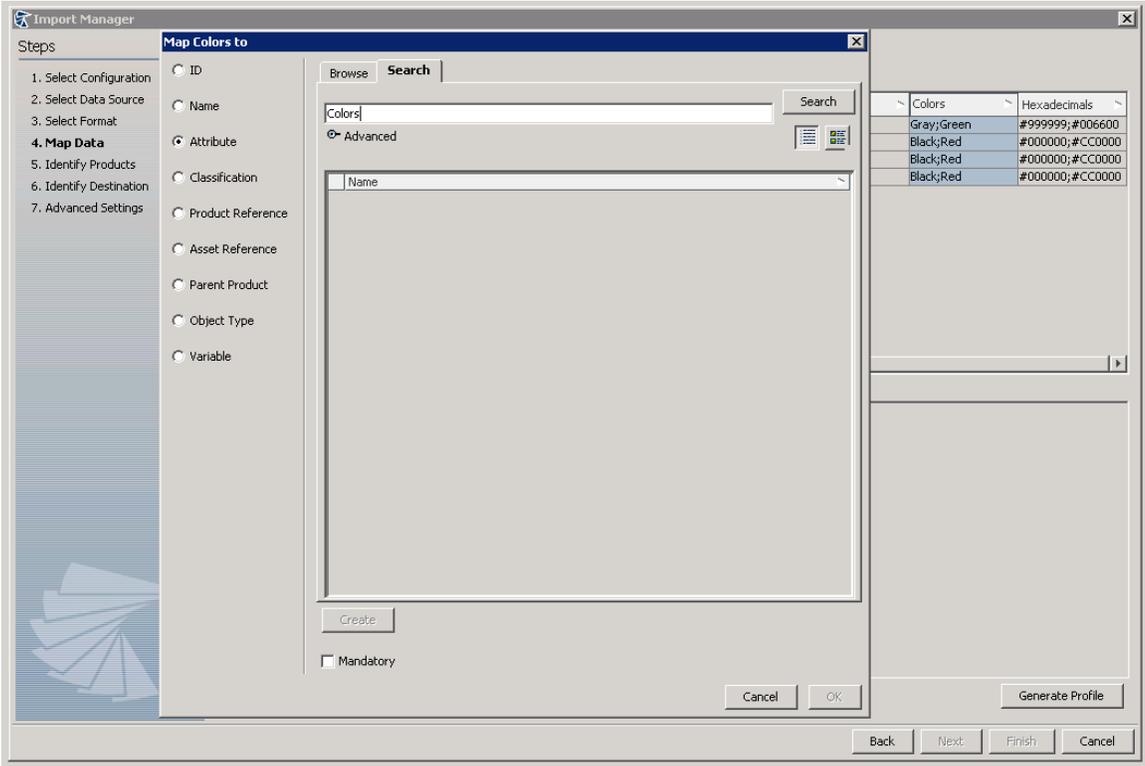


2. Transform the mapped data. The **Append text** and **Append from source** transformations are added in the **Before Multivalue split** section.

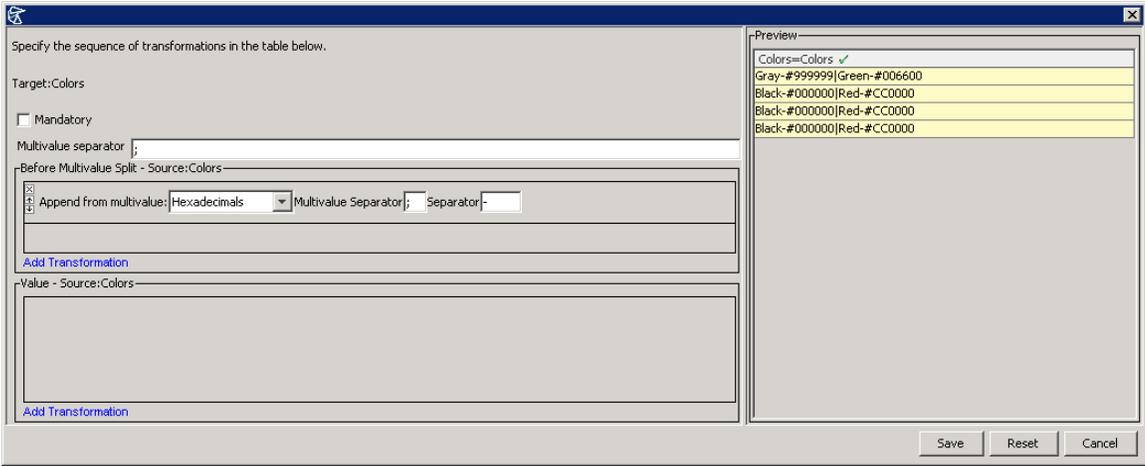


In the following example, you have a list of colors in one column, "Colors" and the corresponding hexadecimals in another, "Hexadecimals". You want to merge those into a multi-valued attribute, "Colors", so that each entry in the multi-value is a color-hexadecimal pair.

3. Map "Colors" to "Colors".



4. Transform the mapped data using the **Append from multivalue source** transformation like shown below. The value separator, "-" in the example, is optional.

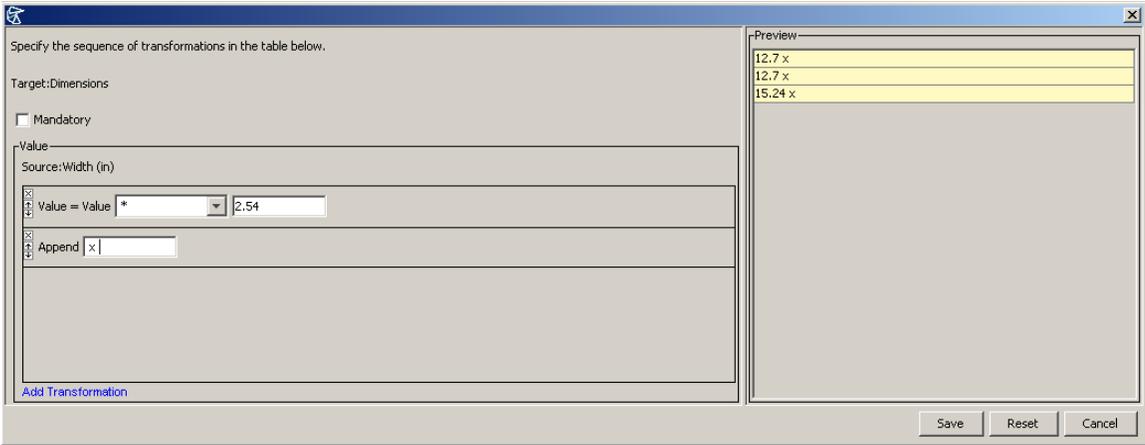


To Concatenate Data Using Variables

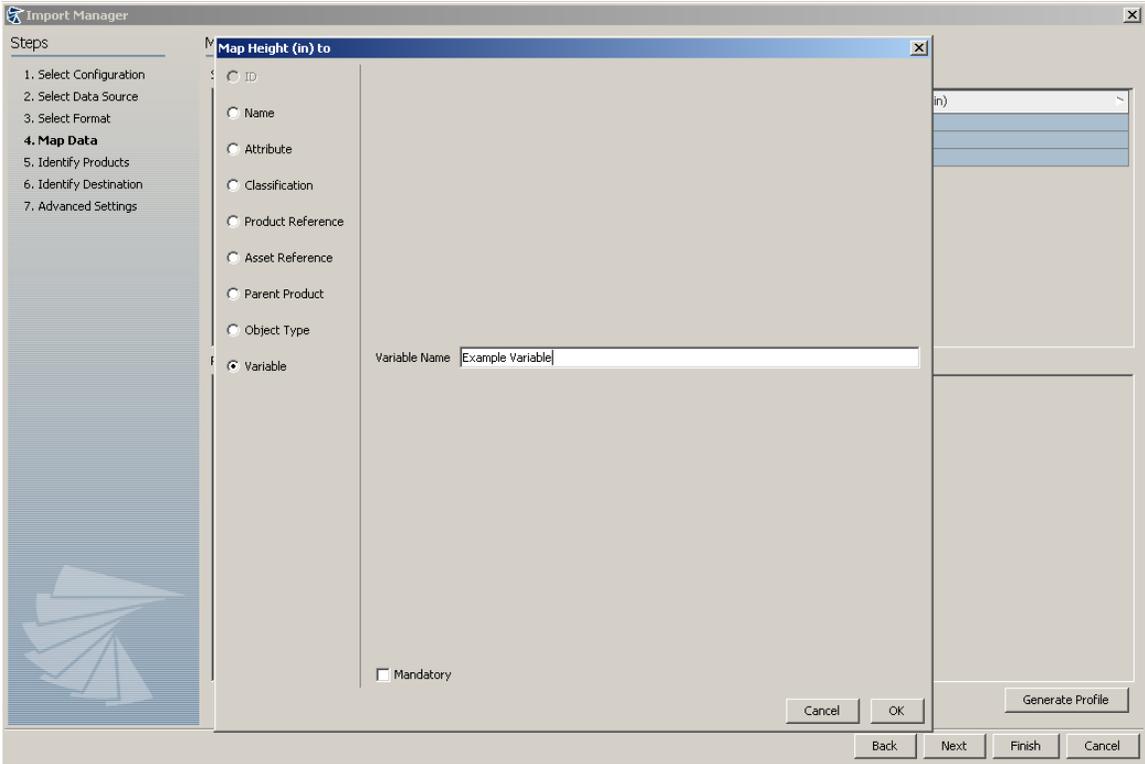
In special cases, you might need to transform both pieces of data that are to be concatenated before the concatenation.

In this example, you have an import file with a "Width (in)" and a "Height (in)" column. Both hold numerical values in inches, and you want to concatenate the values and map them to a "Dimensions" attribute in the format "[Width in cm] x [Height in cm]".

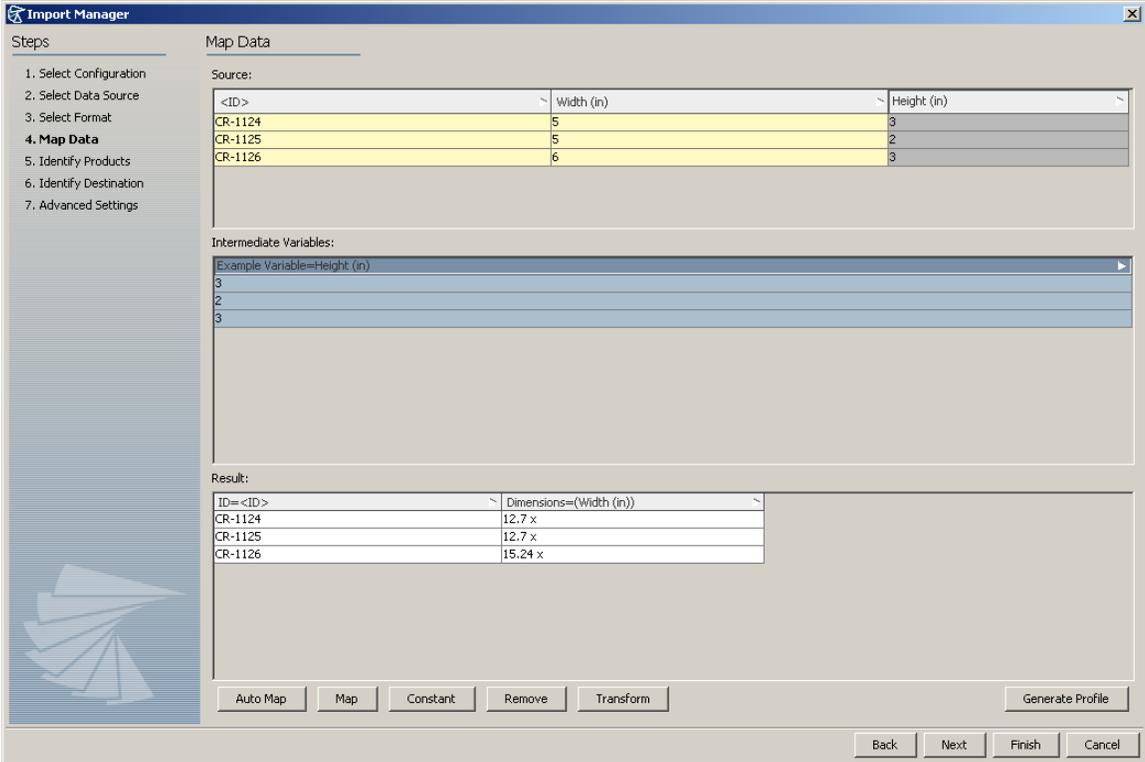
1. Map "Width (in)" to the "Dimensions" attribute and apply transformations.



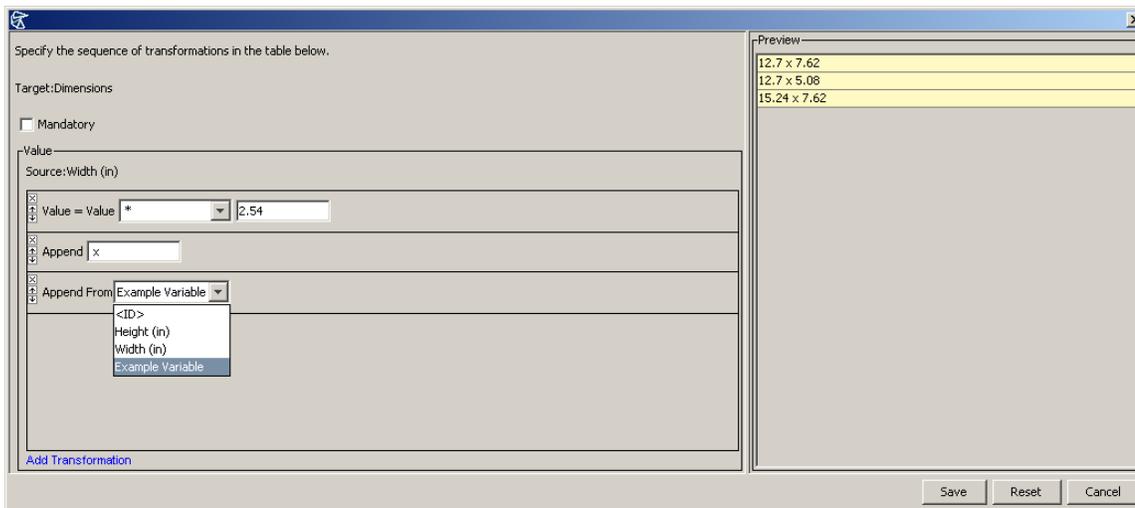
2. Manipulate the data in the "Height (in)" column before it is appended by saving the transformations made in the first mapping, selecting the "Height (in)" column in the **Source** area, and clicking the **Map** button. In the **Map <Column Name to** dialog, select **Variable** and enter a name.



3. Click **OK**. A new intermediate level is displayed between the **Source** and **Result** panels.



- You can now transform this data before it is appended to the data already mapped to the "Dimensions" attribute. Select the variable column, click **Transform**, and then add the transformation that change inches to centimeters.
- In the **Result** area, select the "Dimensions=(Width (in))" column, click **Transform** and add the **Append from source** transformation where the **Intermediate Variable** is now available.



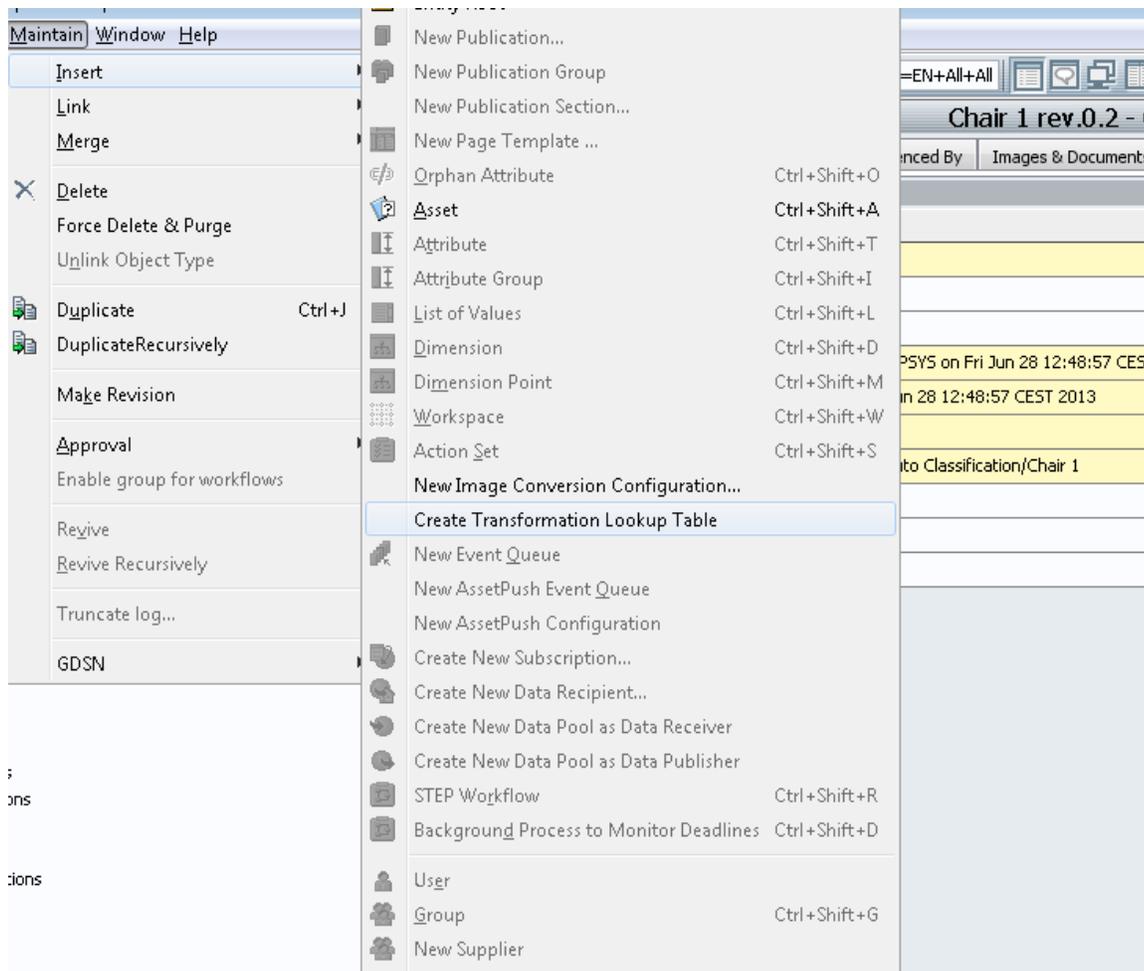
- Click **Save** and the variable displays the transformed value, which can be mapped in the same manner as source columns are mapped. If you save the import configuration, variables and transformations are saved as well as the mappings.

Transformation Lookup Tables

Use transformation lookup tables when you want to transform data by replacing either specific words in values or entire values. The transformations **Replace whole value using Lookup Table** and **Replace words using Lookup Table** are available in the list of available transformations in the Import Manager wizard, when working with tables and on attribute transformations.

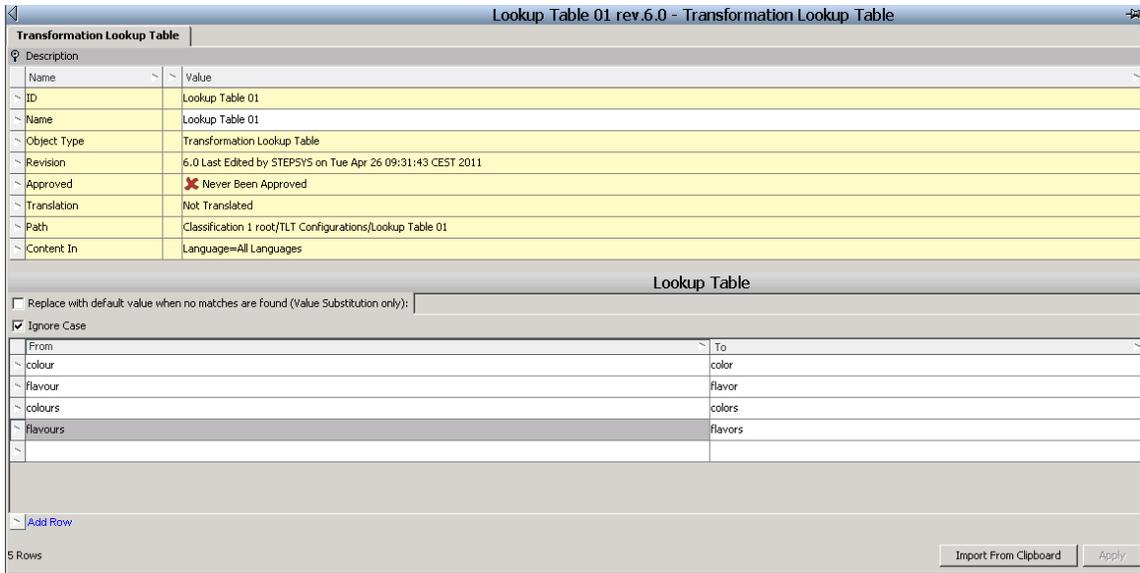
To Create Transformation Lookup Table Objects

- In the **Tree**, expand the **Classification Hierarchy**, and then select the preferred classification.
- From the **Maintain** menu, point to **Insert**, and then choose **Create Transformation Lookup Table**.



3. In the **Create Transformation Lookup Table** dialog enter an ID and a name, and then click **OK**.
4. In the **Transformation Lookup Table** editor, define the from-to replacement pairs, and then click **Apply**.

Note: If you already have substitution pairs defined in an Excel sheet, you can use the **Import From Clipboard** button to paste the data into the table.



Above the substitution table there are two options:

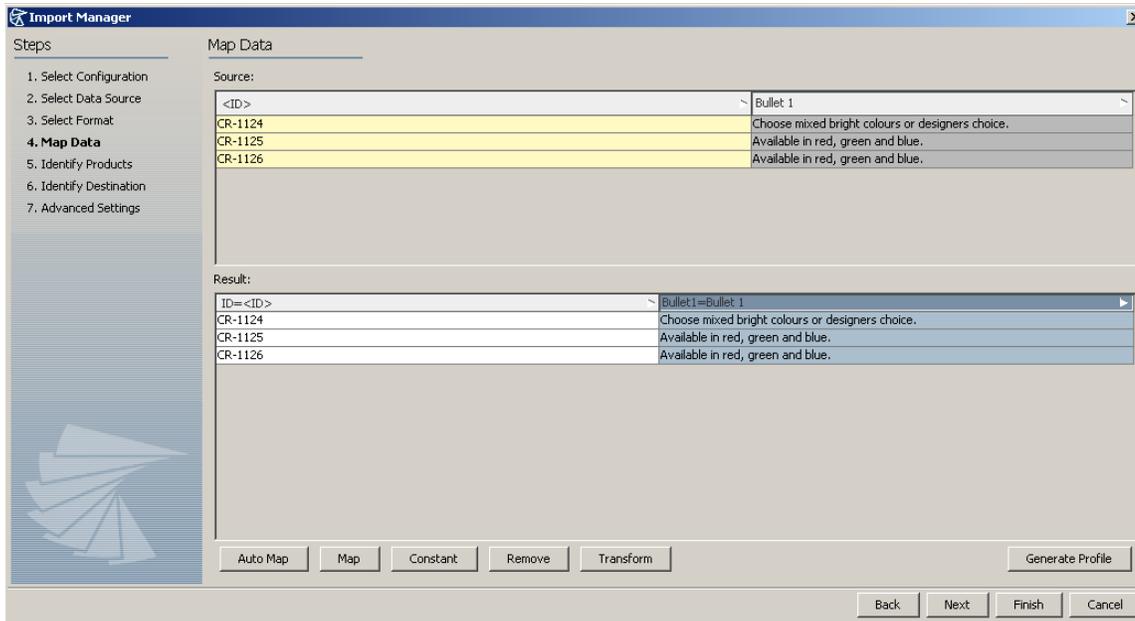
- **Replace with default value when no matches are found (Value Substitution only)** - Only select this option when you use the Transformation Lookup Table to replace entire values. When selected, you can specify a default replacement value that will be used if the values you are transforming do not match any of the entries in the **From** column. For example, if you are importing values into a List of Values and want to make sure all values match entries already in the LOV.
- **Ignore Case** - If selected, case is ignored when values or words in the data to be transformed are matched with entries in the From column. In this example, "Colour", "COLOUR" and "colour" are all replaced with "color".

Note: Transformation Lookup Tables are asset objects. If asset content is configured to be dimension dependent on the system, this also applies to these objects. You can therefore have different replacement pairs defined on the same object in different contexts.

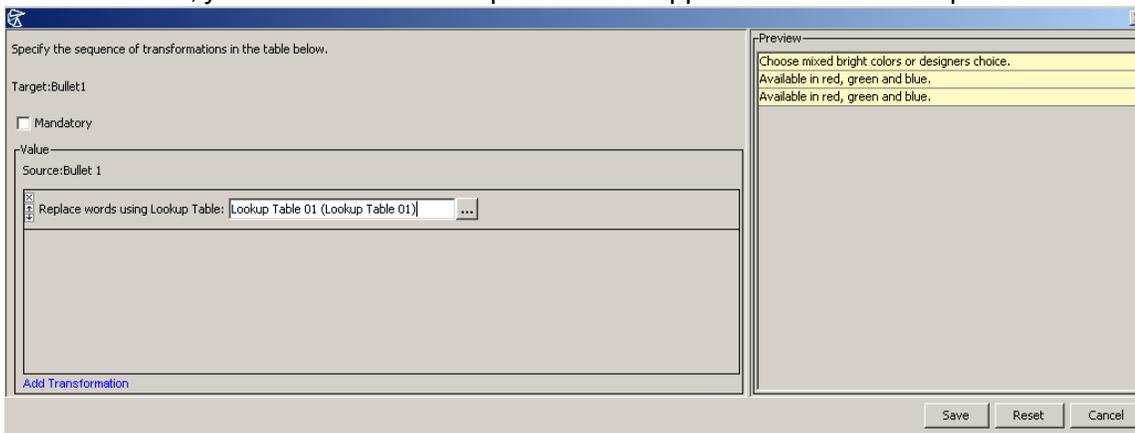
Using Transformation Lookup Tables When Importing Data

Transformation Lookup Tables can be used to transform words and entire values when importing row-column based data.

For example, it is possible to replace all occurrences of a word with another word before data is imported into an attribute. Using a Transformation Lookup Table, you can add a "Replace words using Lookup Table" transformation on the result column, and select the relevant Lookup Table.



In the **Preview**, you can see how the replacement is applied when the Lookup Table is selected.



Transformations that use Lookup Tables can be saved in Import Manager Configurations. To save a Lookup Table the referenced Transformation Lookup Table must not be deleted and must have content in the context of the import.

Note: When whole values are replaced with multi-valued attribute data, the replacement works on the separate entries in the multi-value when added in the Value section. When added in the Before Multivalue Split section it works on the entire string from the import file including the multi-value separators.

Transform Using Attribute Transformation

Using the **[Transform Using Attribute Transformation]** transformation, it is possible to apply a set of transformations specified in an Attribute Transformation to a value. The transformation takes a predefined attribute transformation as argument. Attribute transformations are defined in System Setup.

The If Equals, Source Transformation

Using the **If Equals, Source** transformation, you can make specific columns in a row or column based import file dependent on the content of a mapped column.

For example, a column in your import file specifies whether a product is for print or web publication, and another column contains color information in RGB Hex and CMYK. You now want to populate an attribute with CMYK information if the product is for print and RGB if it is for web.

First, map the column that contains the web and print information to the Color attribute in STEP. Next, apply the If Equals, Source transformation. The transformation specifies to get data from the RGB Hex column if the value equals web and otherwise to get data from the CMYK column.

The screenshot shows the 'Import Manager' interface. The 'Map Data' window displays a table with source data and a 'Result' section. The 'If Equals, Source' dialog is open, showing the configuration for the transformation.

Source	Output	CMYK	RGB Hex
<ID>	Output	CMYK	RGB Hex
3526728	Web	C:0M:31Y:37K:0	#FFB0A1
79205703	Print	C:0M:52Y:100K:64	#5C2C00

Result	Color
ID= <ID> ✓	Color=Output ✓
3526728	Web
79205703	Print

Specify the sequence of transformations in the table below.

Target: Color

Mandatory

-Value - Source: Output -

If value equals	Web
Source	RGB Hex
Else source	CMYK

Preview:

Color=Output ✓
#FFB0A1
C:0M:52Y:100K:64

Buttons: Save, Reset, Cancel

Handling Import Errors

If the Excel, CSV or XML import process reports one or more errors, the status of the background process is "completed with errors". The process is moved to the **Completed with Errors** process group, and an error file containing the defect data is available on the process tab.

You can download the error file from the bottom of the Background Process tab.

Importing New Configuration - Background Process

Background Process Queue Info

Properties

Property	Value
Started by	DOC
Id	BGP_64963
Description	Importing New Configuration
Progress	100%
Status	completed with errors
Created	Mon May 30 12:57:16 CEST 2011
Started	Mon May 30 12:57:20 CEST 2011
Finished	Mon May 30 12:57:21 CEST 2011
Processing Time	0 m 1 s
Time in Queue	0 m 4 s
# of warnings	0
# of errors	1

Execution Report

Error Warning Info

- 1 Retrieval started (Mon May 30 12:57:20 CEST 2011)
- 2 Retrieved 31232 bytes (Mon May 30 12:57:20 CEST 2011)
- 3 Conversion started (Mon May 30 12:57:20 CEST 2011)
- 4 Converted 90 products (Mon May 30 12:57:20 CEST 2011)
- 5 Logged on
- 6 Mapping started (Mon May 30 12:57:20 CEST 2011)
- 7 Mapping completed (Mon May 30 12:57:20 CEST 2011)
- 8 Import Started (Mon May 30 12:57:20 CEST 2011)
- 9 Logged On
- 10 Using import mode "domain"
- 11 Row 6, Column : The parent product with ID '45630' for product with ID '8526723' couldn't be found
- 12 Imported 0 new products
- 13 Processed 90 existing products.
- 14 Skipped 0 products
- 15 Found 0 warnings
- 16 Found 1 errors
- 17 Import completed (Mon May 30 12:57:21 CEST 2011)
- 18 Error file generation started (Mon May 30 12:57:21 CEST 2011)
- 19 Error file with 1 object(s) generated (Mon May 30 12:57:21 CEST 2011)

1-19 of 19

Result

Error file: errorfile.xls

Click buttons to download error file or run the import on the error file

For Excel and CSV imports, the error file contains the rows from the original file with the defect data. For XML files, the error file contains the objects with errors.

If you cannot correct the data errors with transformations, edit the error file offline, save it and then re-import the by selecting the existing import file on Step 1, Select Configuration, of the Import Manager.

When you re-import a file, the status of the background process changes from "completed with errors" to "succeeded" and the process is moved to the **Ended Processes** group. If you do not want to re-import the file, you can make the state transition by clicking the "Mark errors as handled" button as shown below.

Value
DOC
BGP_64963
Importing New Configuration
100%
completed with errors
Mon May 30 12:57:16 CEST 2011
Mon May 30 12:57:20 CEST 2011
Mon May 30 12:57:21 CEST 2011
0 m 1 s
0 m 4 s
0
1

Mark errors as handled