

Lasernet 10.

Lasernet XML Transformer 10

Adam McStravick
Revision 3
2021-11-30



Contents.

1 Introduction.....	3
1.1 Who Should Use This Guide?	3
2 Terms of Use.	4
3 Designing Forms.	5
3.1 Overview	5
3.2 Getting Started.....	7
3.2.1 Creating a Form.....	7
3.2.2 Editor Options.....	7
3.2.3 Selecting a Grab File.....	8
3.2.4 Defining Criteria.....	9
3.2.5 Saving the XML File	14
3.3 Actions	15
3.3.1 Groups.....	15
3.3.2 Actions.....	16
3.3.3 Managing Actions and Groups	35
3.4 XML Namespaces	37
3.5 Keyboard Shortcuts	42

1 Introduction.

1.1 Who Should Use This Guide?

This guide is intended to provide Lasernet users with an overview of the XML Transformer and how to create and edit forms using this feature

2 Terms of Use.

No part of this publication may be reproduced, transmitted, transcribed, or translated into any language in any form by any means without the prior written permission of Formpipe Software. The information in this manual is subject to change without notice. Any company names or data is fictive unless otherwise stated.

Formpipe Software shall not be liable for any loss or damage whatsoever arising from the use of this manual and the information contained therein (including errors or omissions).

Trademarks of other companies mentioned in this document appear for identification purposes only and are the property of their respective companies.

© 2021 Formpipe Software.

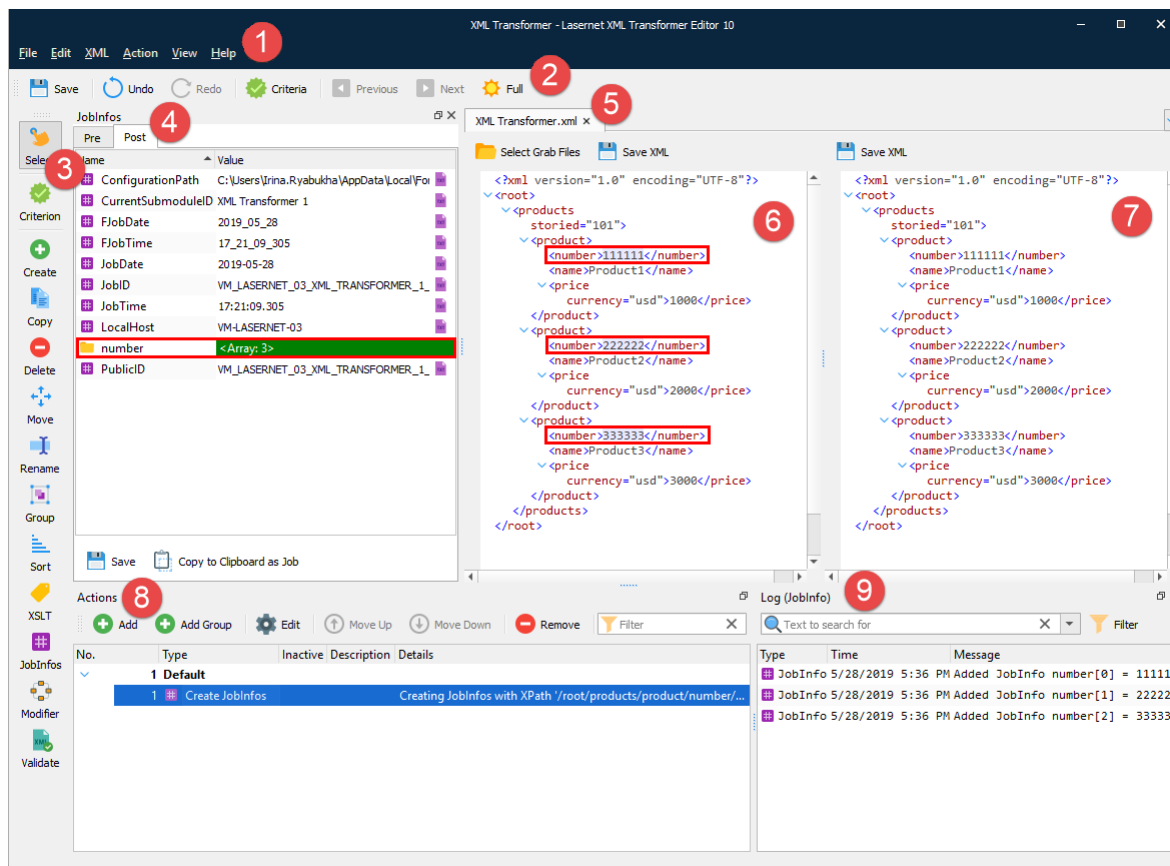
3 Designing Forms.

3.1 Overview

Lasernet XML Transformer Editor provides an easy to use environment for designing and maintaining XML documents. It is usually used as a “beautifier” for preparing documents for the Form Engine. It is expected that the user of this editor has relevant knowledge on XML standards and Xpath 1.0. For more advanced transformations it is also recommended that the user has knowledge of XSLT.

The features of the XML Transformer editor include (but are not limited to):

- Creating, copying, removing, moving, sorting and renaming all types of xml nodes using XPath-expressions.
- Using XSLT for the whole XML document or a specific XML node.
- Specifying different types of behaviours for each action.
- Creating a list of actions to be executed on each form.
- Specifying the order in which actions are executed.



The window where you edit your XML Transformer forms is split in two: the left part shows the input file data (the original data received from the administrative system), and the right part shows the actual form design.

The XML format is represented in a tree structure with nodes identifying the variables. These also contain the actual data. You can switch the XML input view between Full XML, XML without end tags and XML as a table via the **View** menu.

See the table below to learn more details about the user interface elements:

	Element	Description
1	Menu bar	Contains the following commands: <ul style="list-style-type: none"> • File • Edit • XML • Action • View • Help Use the Menu bar to access the main functions of the XML Transformer Editor.
2	Toolbar	Provides easy access to commands such as Save , Undo/Redo , Criteria , Previous , Next , and Full .
3	Developer toolbar	Allows you to easily define criteria and apply actions to the XML file. The Select tool allows you to turn off the activated action.
4	Pre / Post JobInfos	Names and values of JobInfos used before and after actions are executed.
5	Grab file tab(s)	Each input document is opened in a separate tab.
6	XML input view	Result of XML before action transformation
7	XML output view	Result of XML after action transformation
8	Action panel	Allows you to add new actions or manage the list of existing actions. Contains the following commands: <ul style="list-style-type: none"> • Add • Add Group • Edit • Inline • Toogle Inactive • Move Up • Move Down • Move to index... • Move to group • Copy • Paste • Remove • Expand All • Collapse All

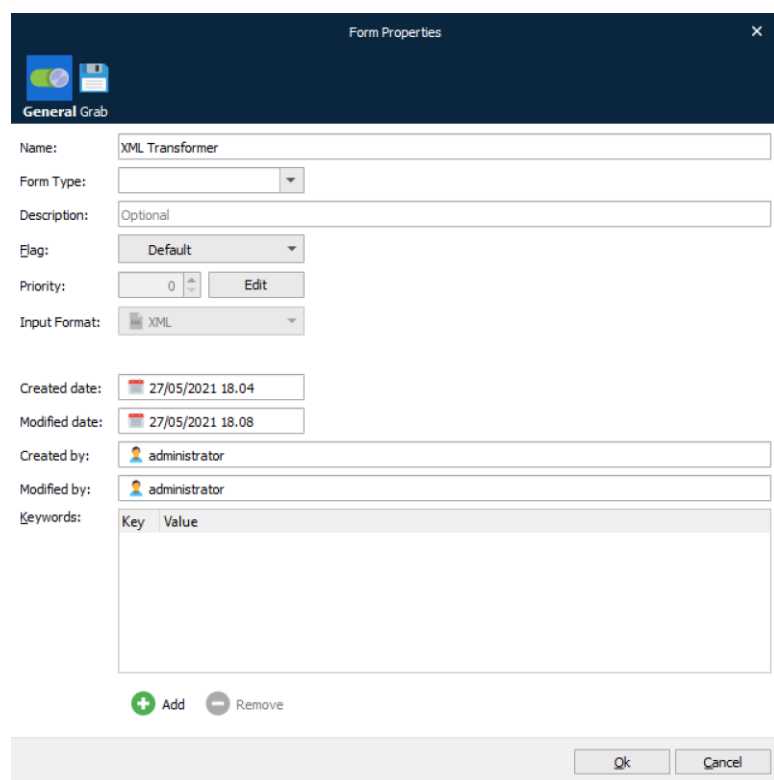
9	Log grid	Presents log records of different types (if any).
---	----------	---

3.2 Getting Started

3.2.1 Creating a Form

When you have added an XML Transformer Engine to your project, you are ready to start creating forms. To create a new XML Transformer form, proceed through the steps listed below:

1. In the left-hand corner of the Lasernet Developer main window, select the **Forms** developer tool.
2. On the **Engines** panel, select your XML Transformer Engine.
3. On the toolbar, click the **Add** button.
4. In the **Form Properties** dialog, provide parameters for your new form, and then click **OK**.



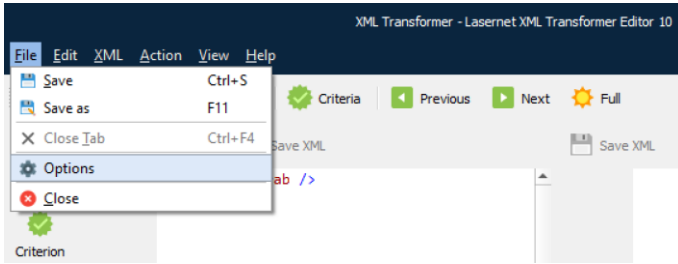
5. You can now start editing your new form by double-clicking it or by right-clicking it and then clicking **Edit** in the context menu.

When you add a new XML Transform form, a blank form automatically opens.

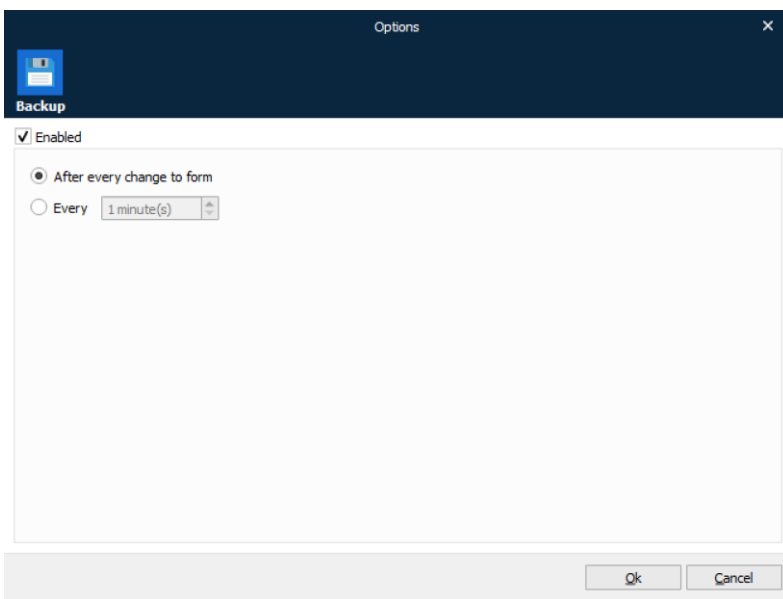
The form is initially empty, so you need to select a grab file.

3.2.2 Editor Options

Once you have created at least one form, ensure that the Editor options are set as needed. You can access the Editor options by clicking **Options** on the **File** menu. The options defined affect all the configuration forms. You can change these options any time whilst working with the application.



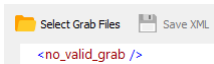
You can also define the **Backup** options. When managing the **Backup** options, you can enable or disable the continuous backup. If enabled, you can also define when the backup process occurs: after every change to a form or every specified point of time (e.g.: every 1-60 minutes).



3.2.3 Selecting a Grab File

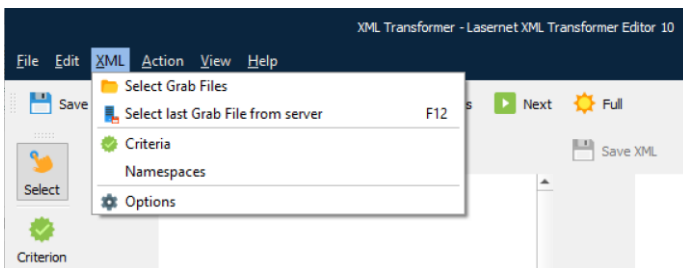
To select a grab file, proceed through the steps listed below:

1. In the top right-hand corner of the form, click the **Select Grab Files** button.



OR

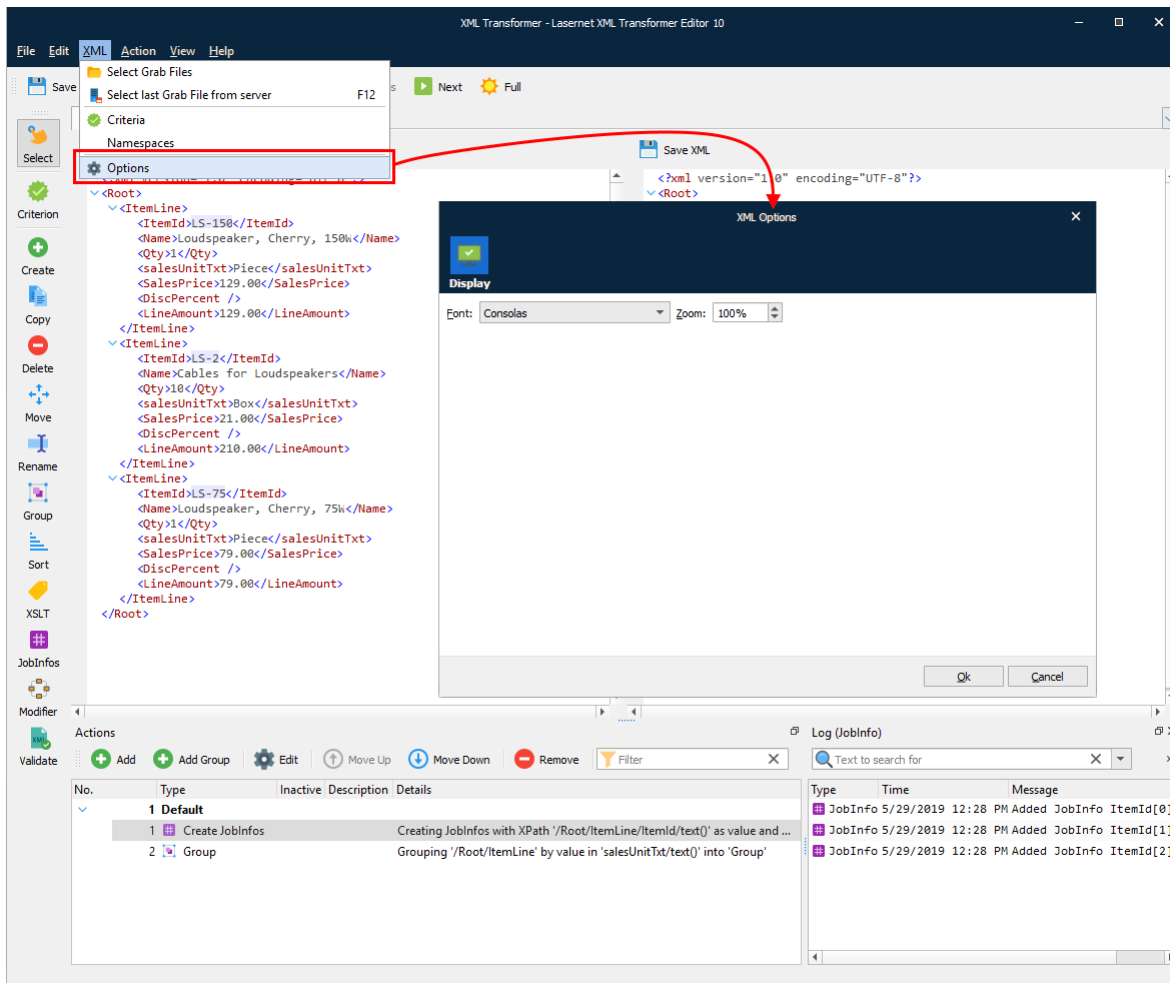
2. On the Menu bar, click **XML**, and then click **Select Grab Files**.



3. With the dialog displayed, select a grab file you wish to process and click **Open**.

Browse to the location on your computer where the relevant grab file is located. This is usually the Grab subdirectory of your configuration.

Once a grab file opens, you can change **Font** and **Zoom** for both input and output. To do this click '**Options**' on the **XML** menu. With the dialog box displayed, set the required values, and then click **Ok**.

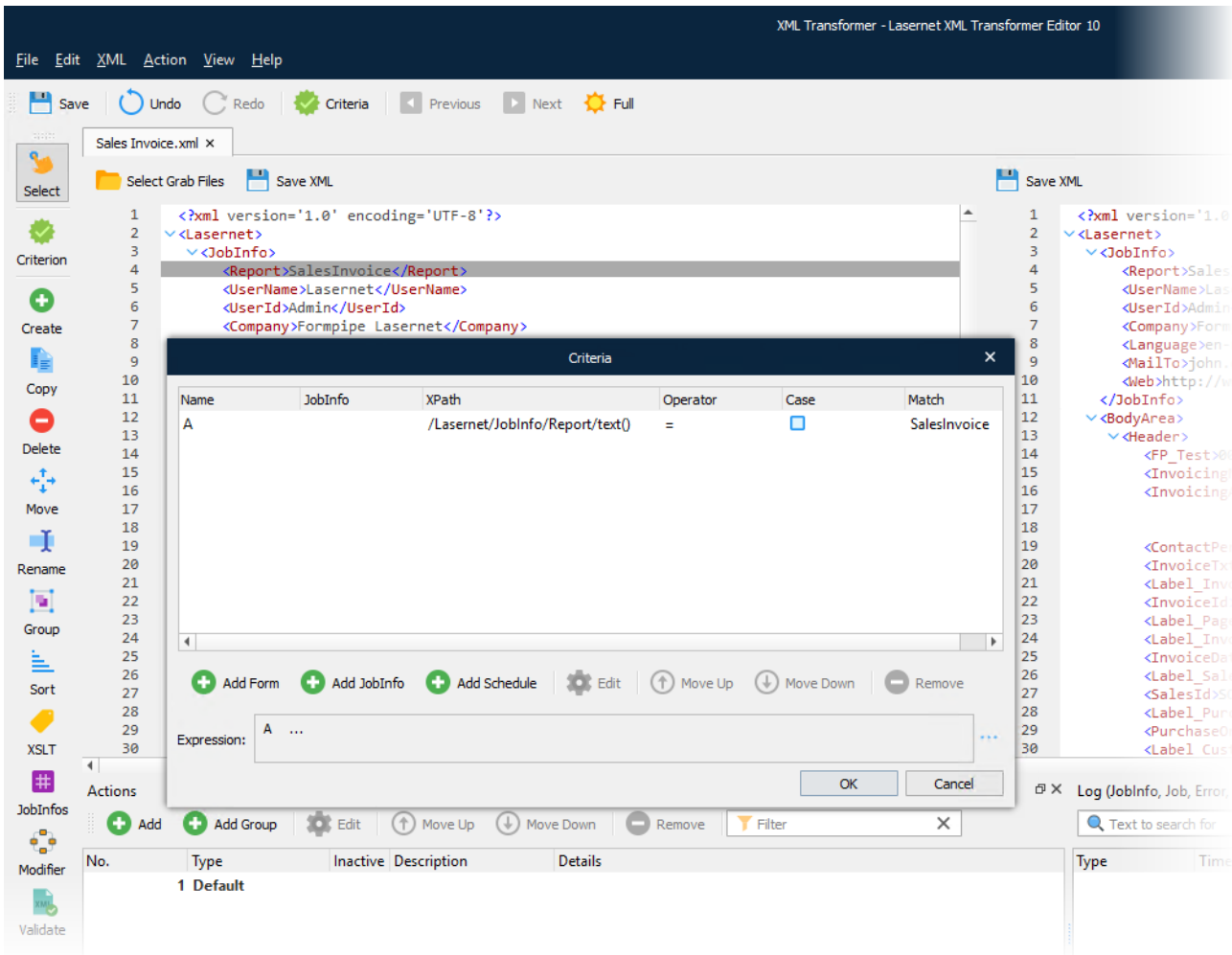


3.2.4 Defining Criteria

Using a grab file, you can define various criteria for your XML Transformer form to recognize input data files.

A criterion is used to look for key data in specific locations in a data input file to ensure that the right input file has been received. If no matches are found the file is not processed by the form.

To define a criterion, open a grab file. For more information on how to select a grab file, see [Selecting a Grab File](#).



Once you have the grab file open, activate the **Criterion** tool and follow the steps below:

1. On the developer toolbar, click the **Criterion** button.



2. Once the cursor turns into a plus sign, click the node that you want to create a criterion on.

3. With the **Edit Criterion** dialog displayed, indicate the following parameters:

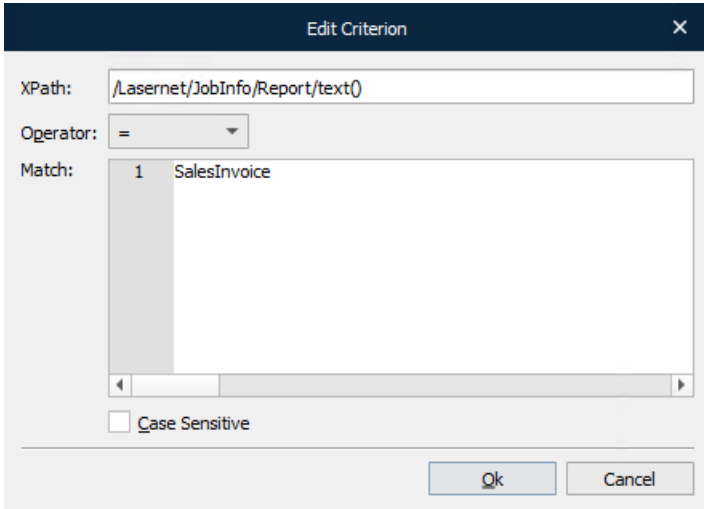
- XPath;
- Operator;
- Match string.

Note: the parameters are automatically filled in. Modify them manually, if required.

4. Select the **Case Sensitive** check box, if required.

5. Once you are done, click **OK**. You can see the string highlighted in grey.

To edit the criterion, double click the string on which it has been created.



The screenshot shows a dialog box titled "Edit Criterion". It has a close button (X) in the top right corner. The "XPath:" field contains the text "/Lasernet/JobInfo/Report/text()". Below it, the "Operator:" dropdown menu is set to "=". The "Match:" field is a list box containing one item: "1 SalesInvoice". At the bottom left, there is a checkbox labeled "Case Sensitive" which is currently unchecked. At the bottom right, there are two buttons: "Ok" and "Cancel".

XPath

The name of the XML node, including the top node.

Operator

The following operators are available:

contains

Returns true if the text data contains the match string.

not contains

Returns true if the text data does not contain the match string.

=

Returns true if the text data matches the match string.

<>

Returns true if the text data does not match the match string.

Operators < <= > >=

These operators try to convert the value of the text data and match string to numbers before comparing them. From here it is possible to make numerical comparisons. If at least one of the text strings is not converted to a number, a regular string comparison is performed.

like

Returns true if the text data matches the regular expression.

not like

Returns true if the text data does not match the regular expression.

exists / not exists

You can choose whether a tagname exists within the XML input file or not by using the operators “exists” and “not exists” and leaving the match string field empty.

Match

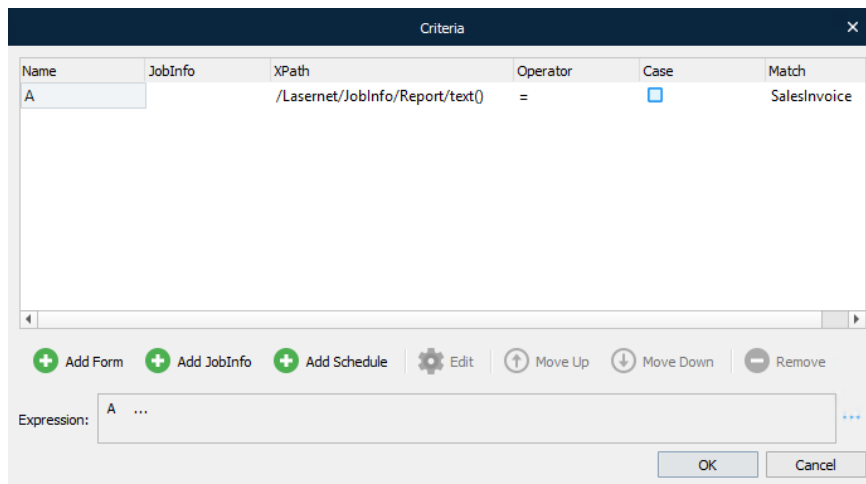
The value (criterion) of the XML node.

Case Sensitive

If checked, the string will be case sensitive.

In addition, you can manually add a criterion. For the purposes of this guide, proceed through the steps listed below:

1. On the toolbar, click the **Criteria** button.
2. With the **Criteria** dialog displayed, click the **Add** button.

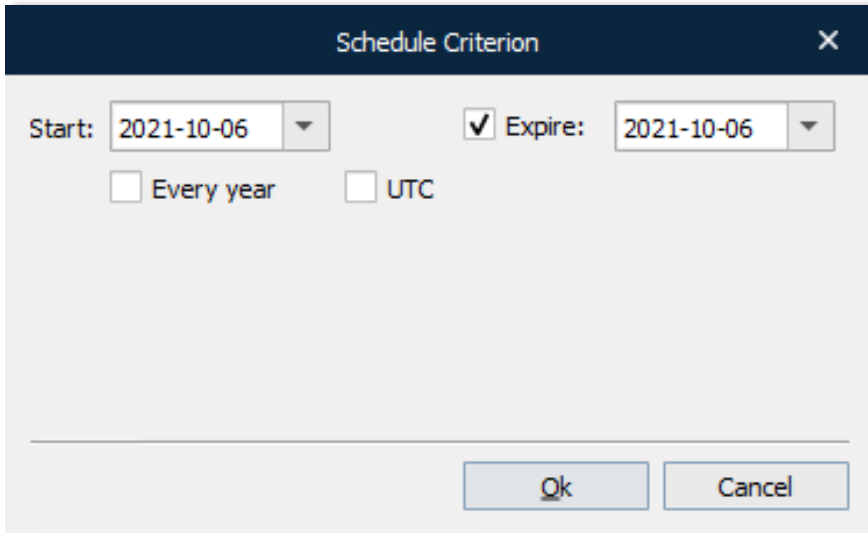


3. Repeat steps 3-5 of the previous instructions to specify parameters for the criteria.

Note: the parameters are not automatically filled in. All the fields of the **Edit Criterion** dialog are empty.

Schedule Criteria

You can include a Schedule in the list of criteria for a form. A Schedule criterion can only be added manually by clicking the **Add Schedule** button.



- Start** The date from which the criterion is true. Choose a date from the calendar.
- Expire** If checked, this is the final date on which the criterion will be true. Choose a date from the calendar.
- Every year** If checked, the criterion will be true between the specified Start and Expire dates (day and month) every year.
- UTC** If checked, UTC time will be used instead of the server time.

Alongside manually adding additional criteria, you can also add jobinfos, edit or remove existing criteria and change the order of the criteria using the **Criteria** dialog. Use the corresponding command buttons on the **Criteria** dialog to:

- **Add JobInfo** – click the button to add a JobInfo.
- **Edit** – click the button to edit the criterion selected in the grid.
- **Move Up** – click to move the selected criterion up in the grid.
- **Move Down** – click to move the selected criterion down in the grid.
- **Remove** – click to remove the selected criterion from the grid.

Form Expressions

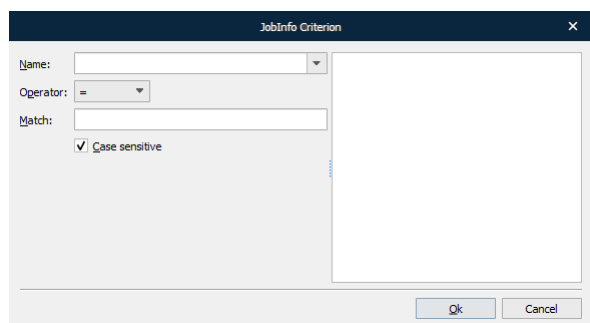
Within the **Criteria** dialog, you can define a form expression as a logical expression. Here are some examples of possible expressions:

- A and B and C and D
- (A or B) and (C or D)
- (A or B) and not (C or D)

If no form expression has been defined, all criteria is true by default. The form expression field allows a high degree of flexibility when defining rules to be used when recognizing data input files.

JobInfo Criteria

You can include JobInfos on the list of form criteria. You can only add a JobInfo criterion manually by clicking the **Add JobInfo** button.



With the **Edit JobInfo Criteria** dialog displayed, define the following values:

- JobInfo Name** From the dropdown list, select the JobInfo which you want to work with. You can also manually type the name of one of your own JobInfos instead.
- Operator** Define a type of operator to use.
- Match string** Type the value to look for in the specified JobInfo.
- Case Sensitive** Select the check box to activate the case-sensitivity rule.

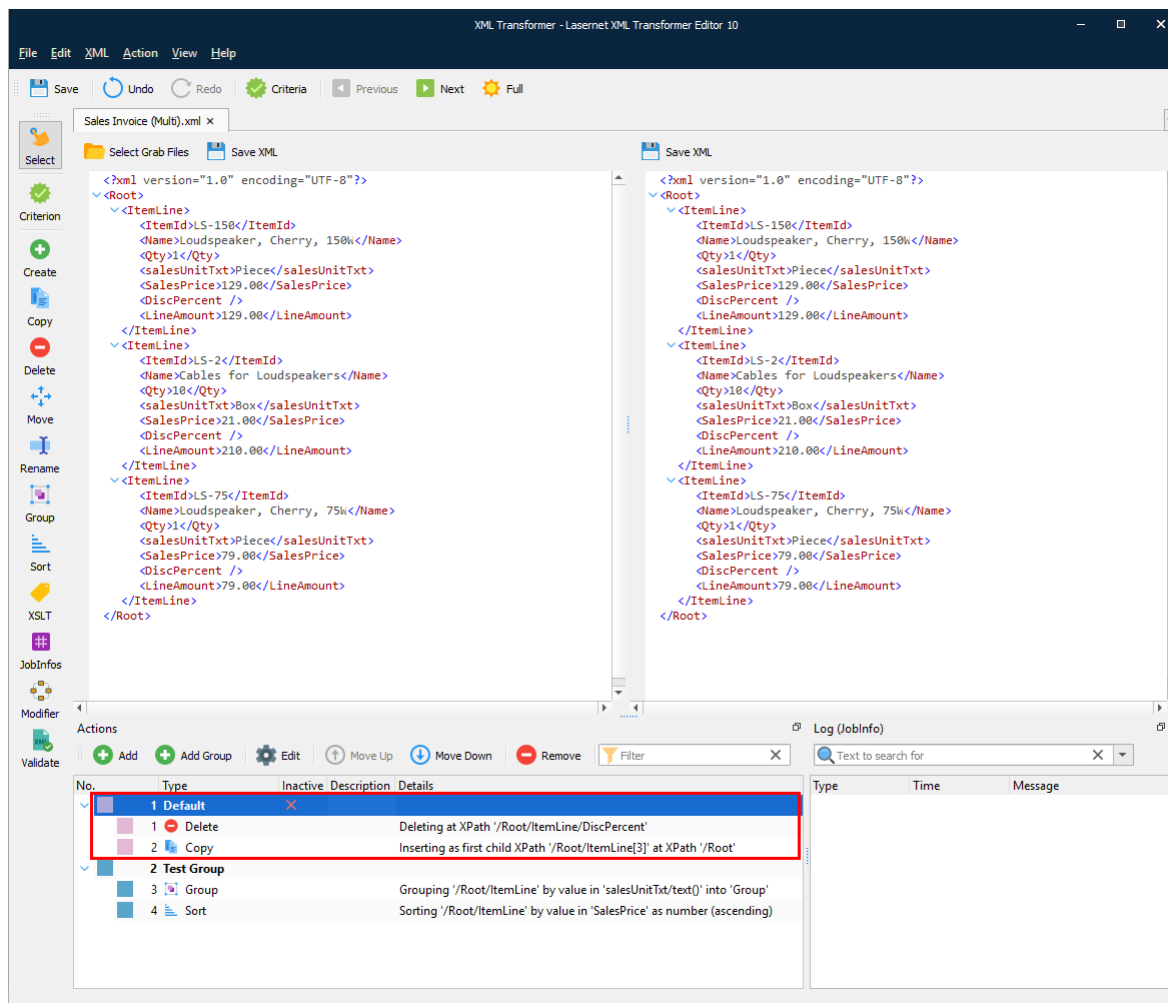
3.2.5 Saving the XML File

While working with an XML file, you can save a local copy (either output or input data, or both) at any time, to preserve its current state. Click the **Save XML** button to do this. With the dialog displayed, specify the location and name for the file to be saved.

3.3 Actions

3.3.1 Groups

You have a possibility to create groups within which you can arrange actions you have created. The tool can ease manipulating actions moved to a definite group by applying the required action to all the actions available under the group at a time. For example, you can toggle inactive all the actions available in a definite group by making this group inactive.

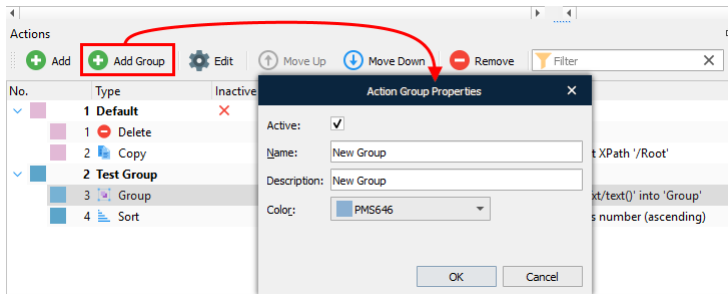


The screenshot displays the XML Transformer Editor interface. The main window shows two XML documents side-by-side, both representing a sales invoice with three item lines. The left document is the source XML, and the right document is the result after applying transformations. The Actions panel at the bottom is active, showing a list of actions:

No.	Type	Inactive	Description	Details
1	Default	<input checked="" type="checkbox"/>		
1	Delete	<input type="checkbox"/>	Deleting at XPath '/Root/ItemLine/DiscPercent'	
2	Copy	<input type="checkbox"/>	Inserting as first child XPath '/Root/ItemLine[3]' at XPath '/Root'	
2 Test Group				
3	Group	<input type="checkbox"/>	Grouping '/Root/ItemLine' by value in 'salesUnitTxt/text()' into 'Group'	
4	Sort	<input type="checkbox"/>	Sorting '/Root/ItemLine' by value in 'SalesPrice' as number (ascending)	

The **Default** group is created by default, and all the actions you create are added to this group until you move them to another one, if needed. If you have more groups, except for the **Default** one, every new action you create is added to the group which is currently selected in the **Actions** panel.

To create a group, click the **Add Group** button on the **Actions** panel. In the **Action Group Properties** dialog that appears, define the required properties (the **Name** text box is a required field), and then click **OK**.



3.3.2 Actions

3.3.2.1 Create

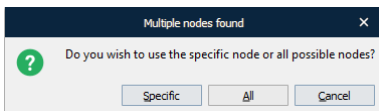
The create action is one of the simplest actions available. It takes an XPath and a given name and then creates a new node based on the type and behaviour chosen.

To apply the create action, follow the steps listed below:

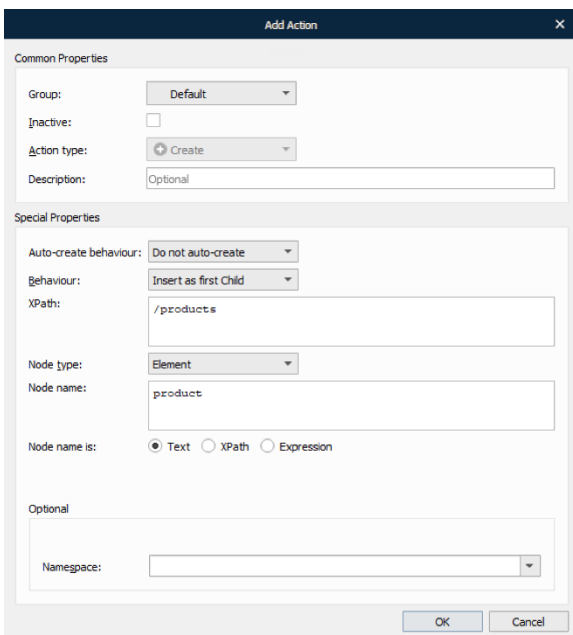
1. On the developer toolbar, click the **Create** button.



2. Once the cursor turns into a plus sign, click the node on which you want to base your new node. If more than one element of the same name is found, the following dialog appears. Make your choice by clicking the corresponding button.



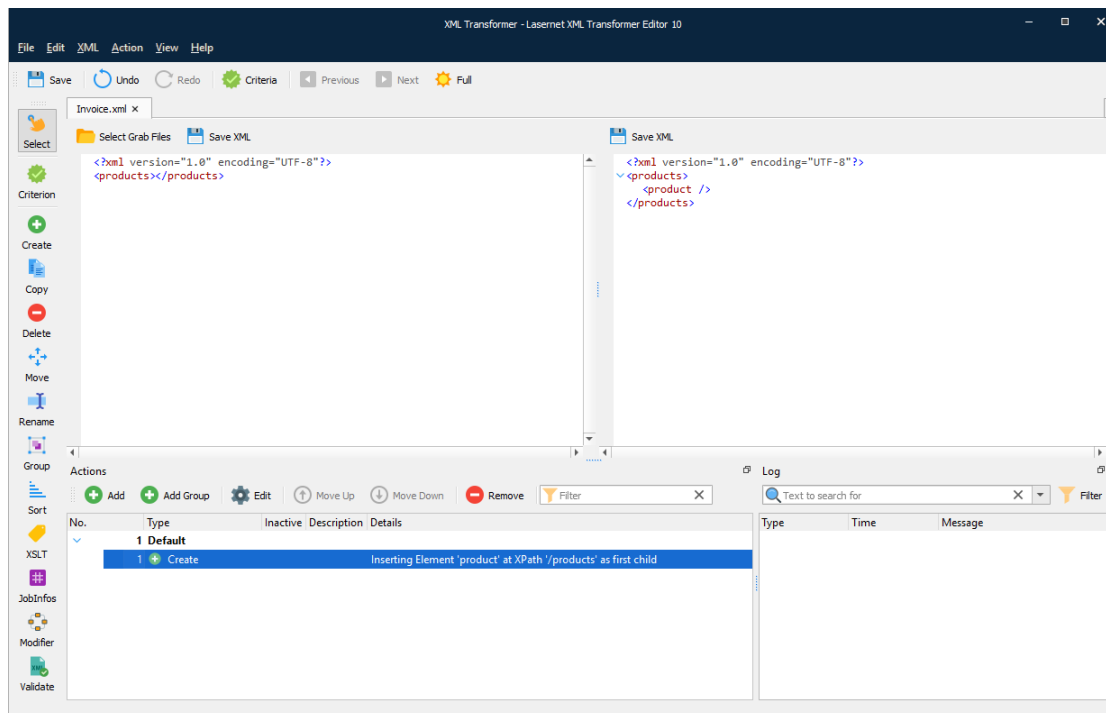
3. With the **Add Action** dialog displayed, fill in the fields, as required, and click **OK**.



See the following table to learn more about the action properties:

Property		Description
1	Auto-create behaviour	<p>This setting can auto-create a new element displayed in XPath if not already present in XML document. The place of the new element insertion:</p> <ul style="list-style-type: none"> • Do not auto-create (default value) • Insert as first sibling • Insert as last sibling • Insert as first child • Insert as last child
2	Behaviour	<p>Select a behaviour item to determine the place of the new node insertion:</p> <ul style="list-style-type: none"> • Insert as first sibling • Insert as last sibling • Insert as first child • Insert as last child • Insert before • Insert after • Replace
3	XPath	Displays the path expression to select a node or a node-set in the XML document.
4	Node type	<p>Select a type for a new node:</p> <ul style="list-style-type: none"> • Element • Attribute • Text • Comments • CDATA • Entity reference • Processing instruction • XML <p>If the type is set to Attribute, an additional property is added, (Node value) and must be defined as either Text, Xpath, Expression, or JobInfo.</p>
5	Node name	Type a node name.
6	Node name is	Specify the node name by selecting if it is Text , Xpath , Expression or JobInfo (XML only).
7	Namespace	<p>Select the required namespace from the available ones or type your own namespace to define the namespace declaration.</p> <p>To create a non-default namespace, type a node name in the Node name text box in the following format:</p> <pre><prefix>:<node name>.</pre>

The following image shows the result of the create action: inserting the Product element to the /products/ XPath as the first child.



3.3.2.2 Copy

Due to the number of possible variations that could be encountered when using the copy action, a complex logic is used to ensure that valid XML actions are always observed. The table below shows all of the combinations available using a source XPath, a target XPath, and a behaviour.

Source	Target	Insert as last sibling	Insert as first sibling	Insert as first child	Insert as last child	Insert before	Insert after	Replace
Attribute	Attribute	+	+					+
Attribute	Element	+	+	+	+	+	+	
Attribute	Text types*	+	+			+	+	+
Element	Attribute							
Element	Element	+	+	+	+	+	+	+
Element	Text types*	+	+			+	+	
Text types*	Attribute							+
Text types*	Element	+	+	+	+	+	+	+
Text types*	Text types*	+	+	+	+	+	+	+

* Text types covers: text-nodes, comments, cdata, and entity references.

Attribute → Attribute

Behaviour	Strategy
Insert as first sibling Insert as last sibling	Replaces the value of the target attribute with the value of the source attribute. The name of the attribute remains unchanged.
Insert as first child Insert as last child Insert before Insert after	N/A
Replace	Replaces the value of the target attribute with the value of the source attribute. The name of the attribute remains unchanged.

Attribute → Element

Behaviour	Strategy
Insert as first sibling Insert as last sibling Insert as first child Insert as last child Insert before Insert after	Adds the source attribute to the attributes of the target.
Replace	N/A

Attribute → Text types

Behaviour	Strategy
Insert as first sibling Insert as last sibling	Adds a new node of the same type as the target node (text, comment, cdata, or entity ref.) as the first child of the parent of the target node. Inserts the value of the source attribute as the value of the new node.
Insert as first child Insert as last child	N/A
Insert before	Inserts a new node of the same type as the target node (text, comment, cdata, or entity ref.) just before the target node. Inserts the value of the source attribute as the value of the new node.
Insert after	Inserts a new node of the same type as the target node (text, comment, cdata, or entity ref.) just after the target node. Inserts the value of the source attribute as the value of the new node.
Replace	Replaces the value of the target node with the value of the source attribute.

Element → Attribute

An element may not be copied to an attribute.

Element → Element

All copy methods for an element are recursive. This means that whenever a copy action is performed on an element, the descendants (the contents) of that element are always copied.

Behaviour	Strategy
Insert as first sibling	Adds the source element as the first child of the parent of the target element.
Insert as last sibling	Adds the source element as the last child of the parent of the target element.
Insert as first child	Adds the source element as the first child of the target element.
Insert as last child	Adds the source element as the last child of the target element.
Insert before	Inserts the source element just before the target element.
Insert after	Inserts the source element just after the target element.
Replace	Replaces the target element with the source element.

Element → Text types

Behaviour	Strategy
Insert as first sibling	Adds the source element as the first child of the parent of the target node.
Append as sibling	Adds the source element as the last child of the parent of the target node.
Insert as first child Insert as last child	N/A
Insert before	Inserts the source element just before the target node.
Insert after	Inserts the source element just after the target node.
Replace	N/A

Text types → Attribute

Behaviour	Strategy
Insert as first sibling Insert as last sibling Insert as first child Insert as last child Insert before Insert after	N/A
Replace	Replaces the value of the target attribute with the text value of the source node.

Text types → Element

Behaviour	Strategy
Insert as first sibling	Adds the source node as the first child of the parent of the target element.
Insert as last sibling	Adds the source node as the last child of the parent of the target element.
Insert as first child	Adds the source node as the first child of the target element.
Insert as last child	Adds the source node as the last child of the target element.
Insert before	Inserts the source node just before the target element.
Insert after	Inserts the source node just after the target element.

Replace	Replaces the target element with the source node following the steps: 1. Inserts after the target element. 2. Deletes the target element.
---------	---

Text types → Text types

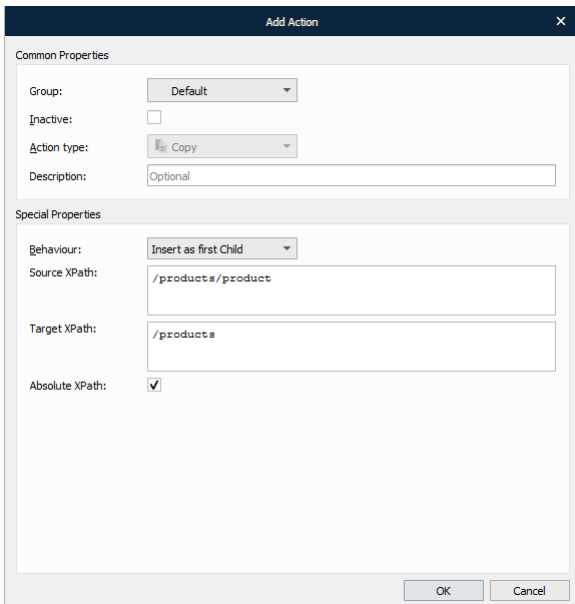
Behaviour	Strategy
Insert as first sibling	Adds the source node as the first child of the parent of the target node.
Insert as last sibling	Adds the source node as the last child of the parent of the target node.
Insert as first child Insert as last child	N/A
Insert before	Inserts the source node just before the target node.
Insert after	Inserts the source node just after the target node.
Replace	Replaces the value of the target node with the value of the source node.

To apply the copy action, proceed through the steps listed below:

1. On the developer toolbar, click the **Copy** button.



2. Once the cursor turns into a plus sign, click the node you want to copy. Then move the rectangle to the place where you want to paste the node.
3. With the **Add Action** dialog displayed, fill in the fields, as required, and then click **OK**.

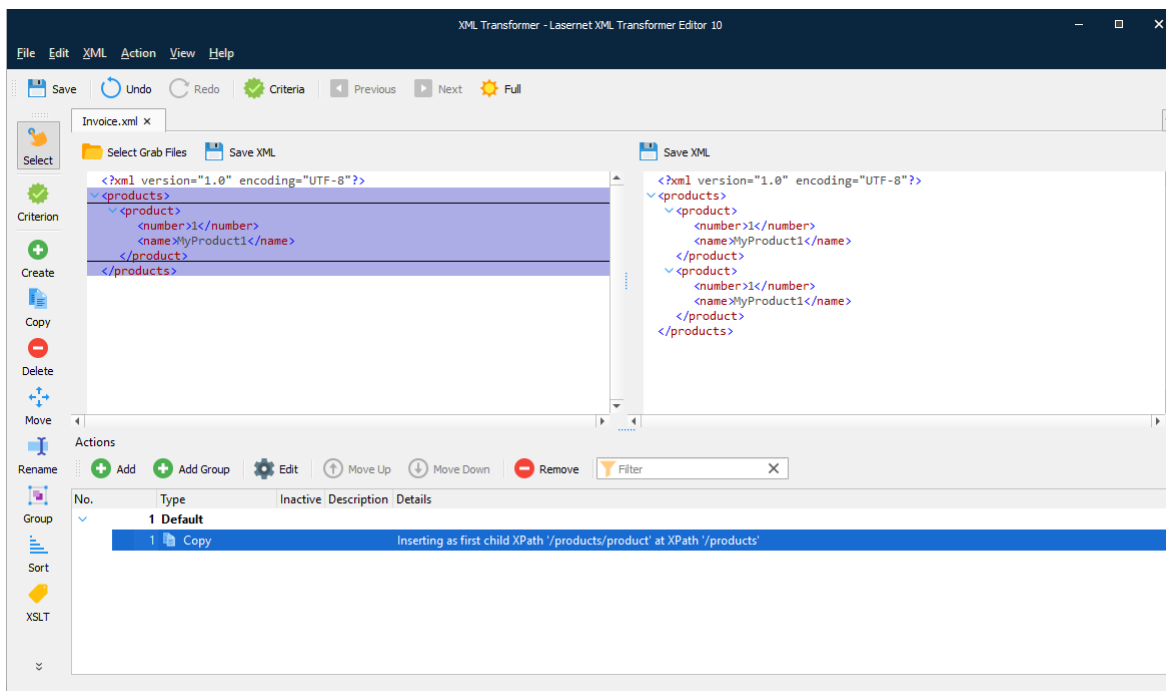


See the table below to learn more about the action properties:

Property	Description	
1	Behaviour	Select the behaviour of the item to determine its place:

		<ul style="list-style-type: none"> • Insert as first sibling • Insert as last sibling • Insert as first child • Insert as last child • Insert before • Insert after • Replace
2	Source XPath	Shows the source XPath.
3	Target XPath	Shows the target XPath.
4	Absolute XPath	Select the check box to indicate the full path in the Target XPath field. If you clear the check box, you must type a relative path in the Target XPath field.

Below is a picture illustrating the result of the copy action.



In addition, you can right-click the node you want to copy and then click **Copy node** on the context menu that appears. Afterwards, right-click the node where you want to paste it and then select the **Paste node** in the context menu. Once you paste the node, the **Copy** action is added to the **Actions** list.

3.3.2.3 Delete

The delete action takes an XPath which returns either a node or a node list to be deleted.

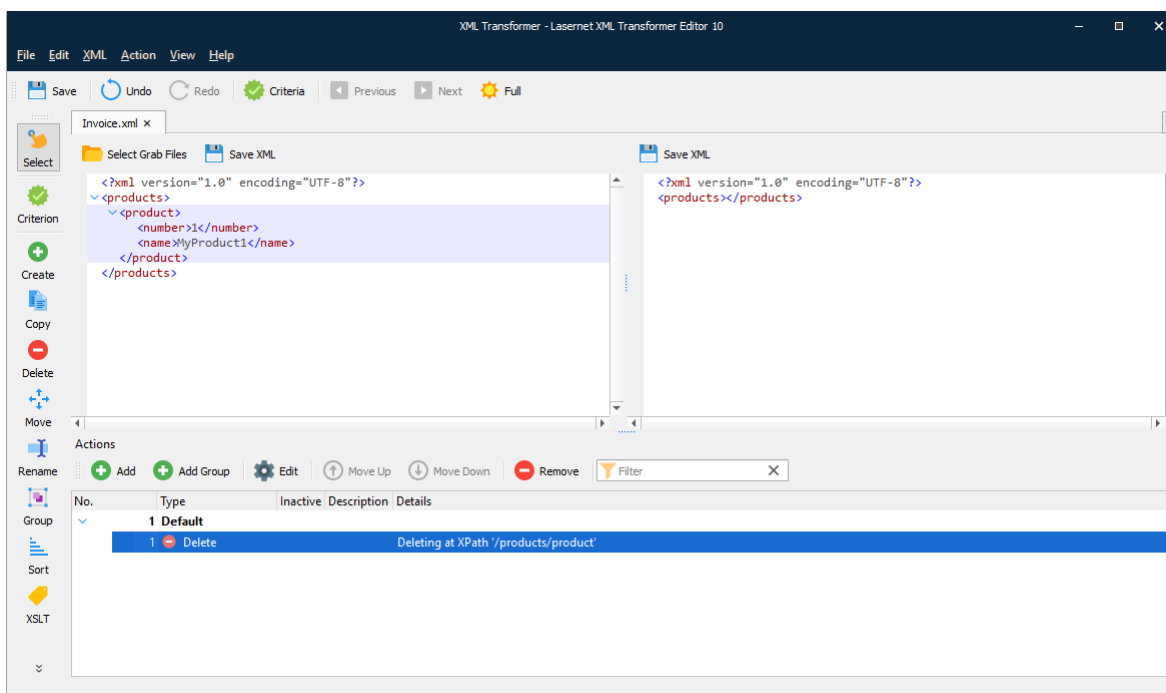
To apply the delete action, proceed through the steps listed below:

1. On the developer toolbar, click the **Delete** button.



2. Once the cursor turns into a plus sign, click the node you want to delete.

Below is a picture illustrating the result of the delete action.



3.3.2.4 Move

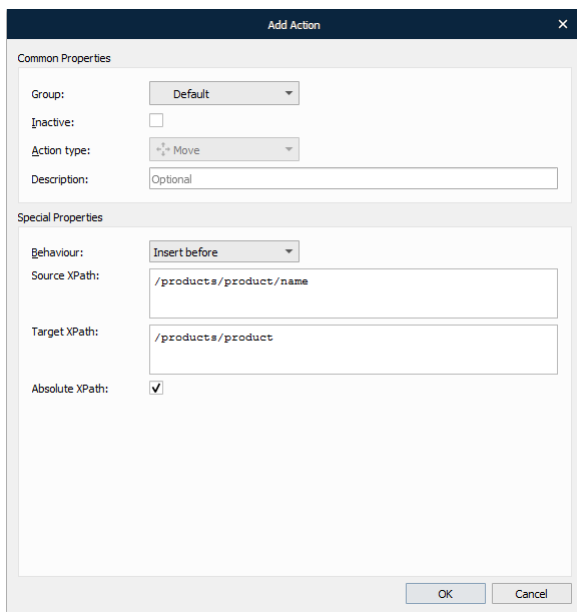
The move action is a combination of a copy action combined with a delete action on the source XPath.

To execute a move action, follow the steps listed below:

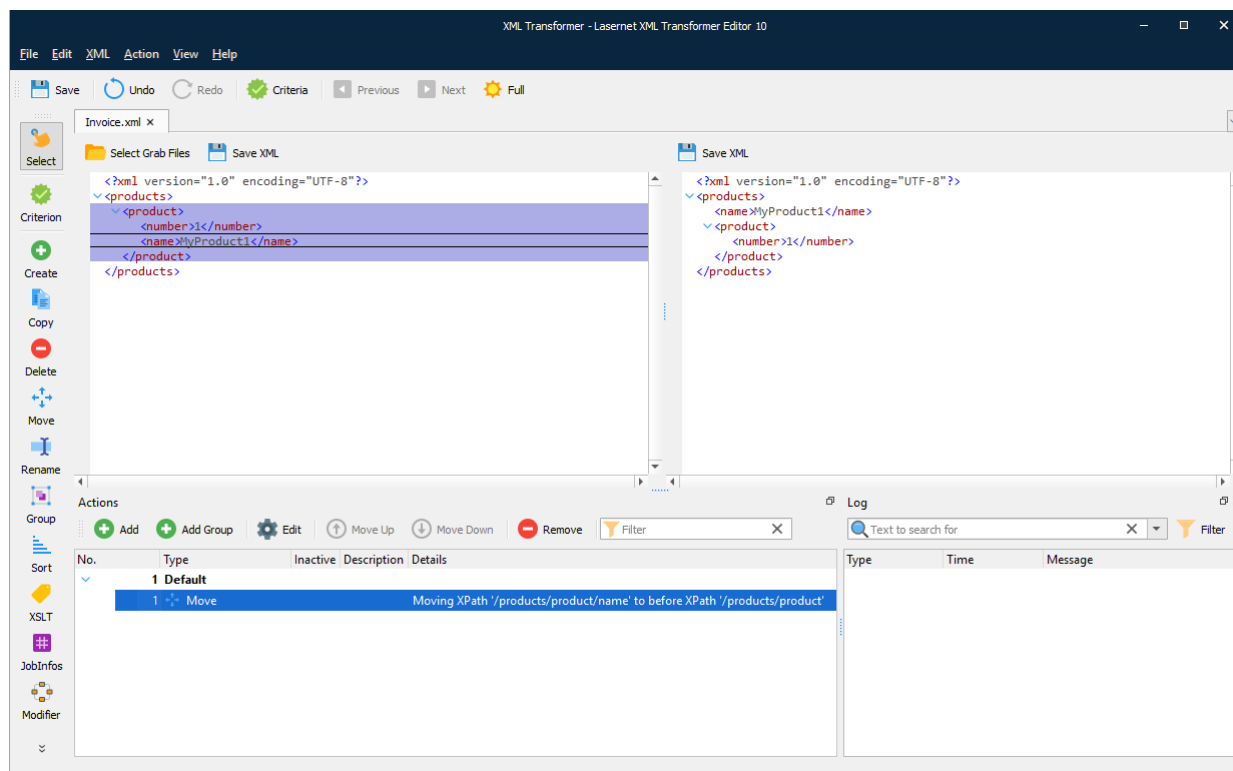
1. On the developer toolbar, click the **Move** button.



2. Once the cursor turns into a plus sign, click the node you want to move. Then move the rectangle to where you want to paste it.
3. With the **Add Action** dialog displayed, fill in the fields, as required, and then click **OK**.



Below is a picture illustrating the result of the move action.



In addition, a new **Move** action is added to the **Actions** list.

You can also move a node easily by using the right-click context menu. Right-click the node you want to cut and select **Cut node**, then right-click the node that you want to paste and select **Paste node**.

3.3.2.5 Rename

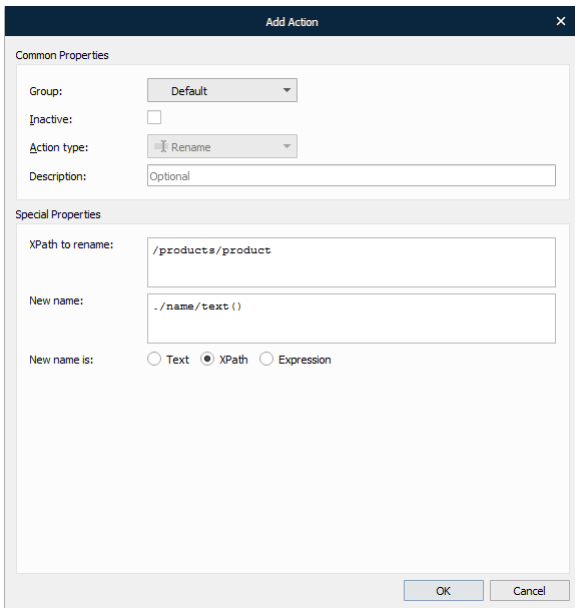
The rename action takes an XPath and a node name as parameters. The node or node-set returned by the XPath expression is then renamed to the value you specify.

To apply the rename action, follow the steps listed below:

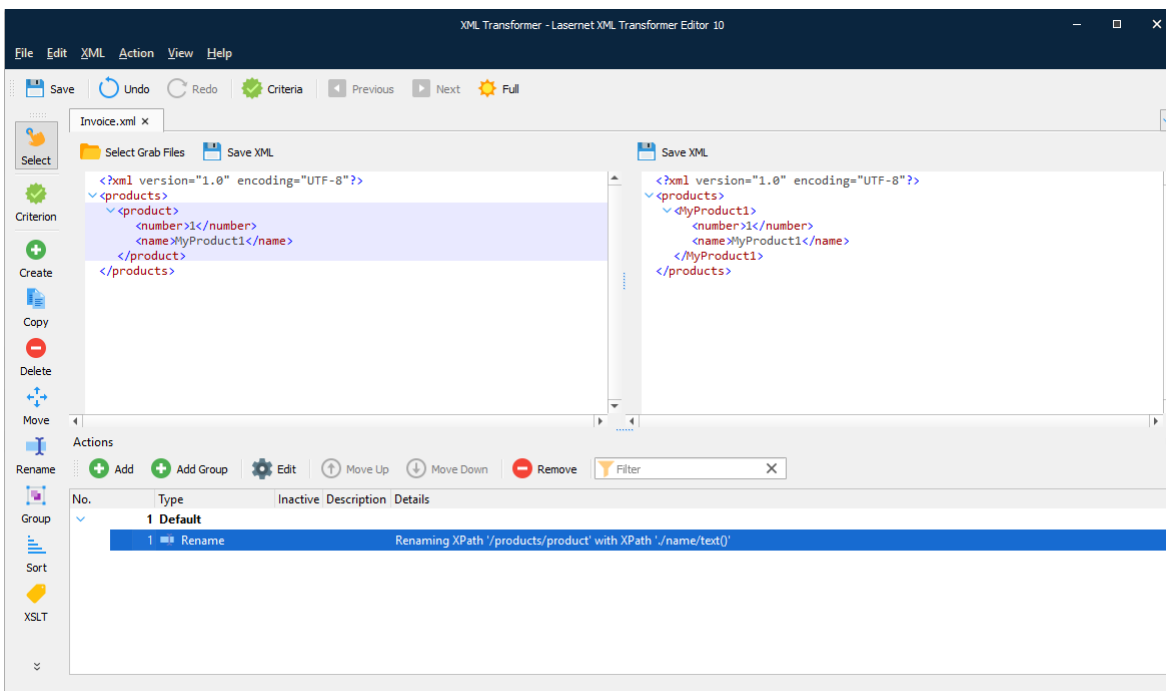
1. On the developer toolbar, click the **Rename** button.



2. Once the cursor turns into a plus sign, click the node you want to rename.
3. With the **Add Action** dialog displayed, type a new name in the corresponding field and then click **OK**.



Below is a picture illustrating the result of the rename action.



3.3.2.6 Group

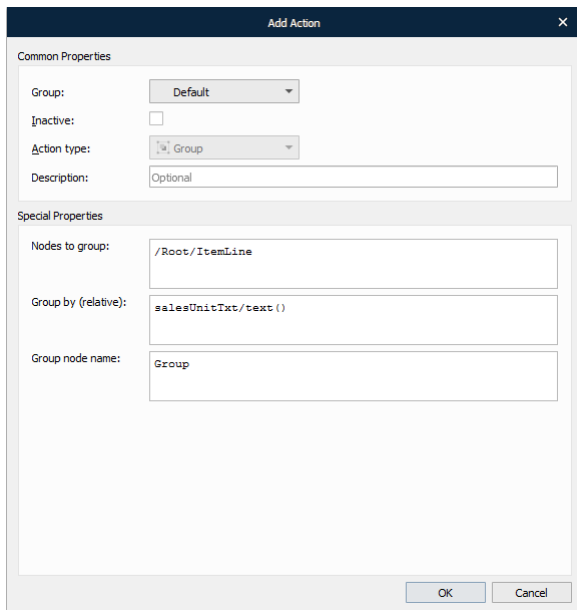
The group action is used for creating new groups in an XML structure according to the parameters you specify. This is convenient if groups are required to create complex calculations based on the unstructured XML data.

To apply the group action, follow the steps listed below:

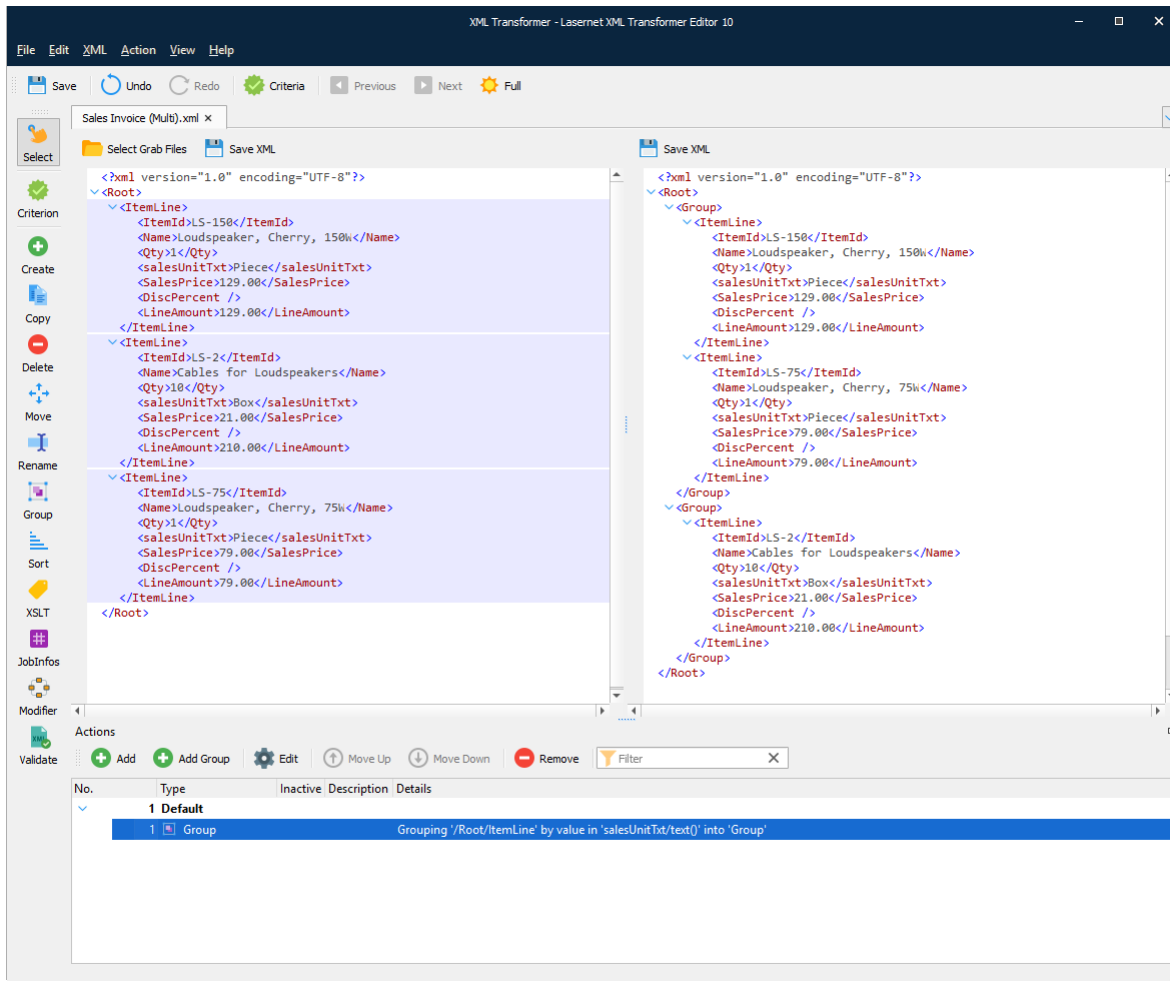
1. On the developer toolbar, click the **Group** button.



2. Once the cursor turns into a plus sign, click the node you want to group.
3. Next, click a node value by which you want to group the specified node(s).
4. With the **Add Action** dialog displayed, specify a name for the new node group. Once you are done, click **OK**.



Below is a picture illustrating the result of the group action.



3.3.2.7 Sort

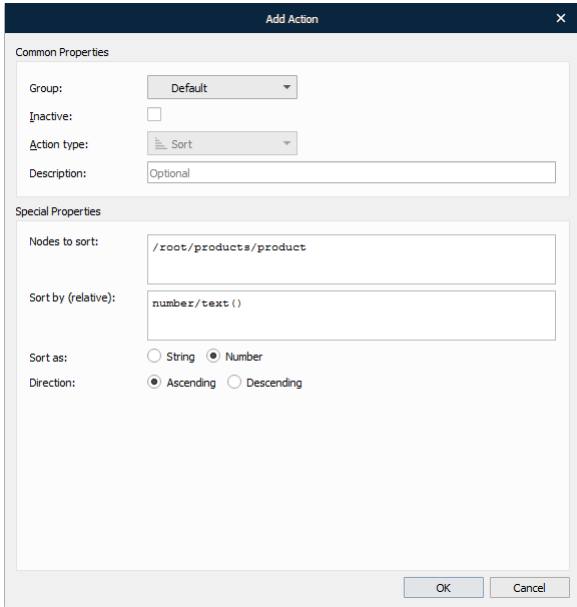
The sort action is used for sorting nodes in the node list according to the parameters you specify.

To apply the sort action, follow the steps listed below:

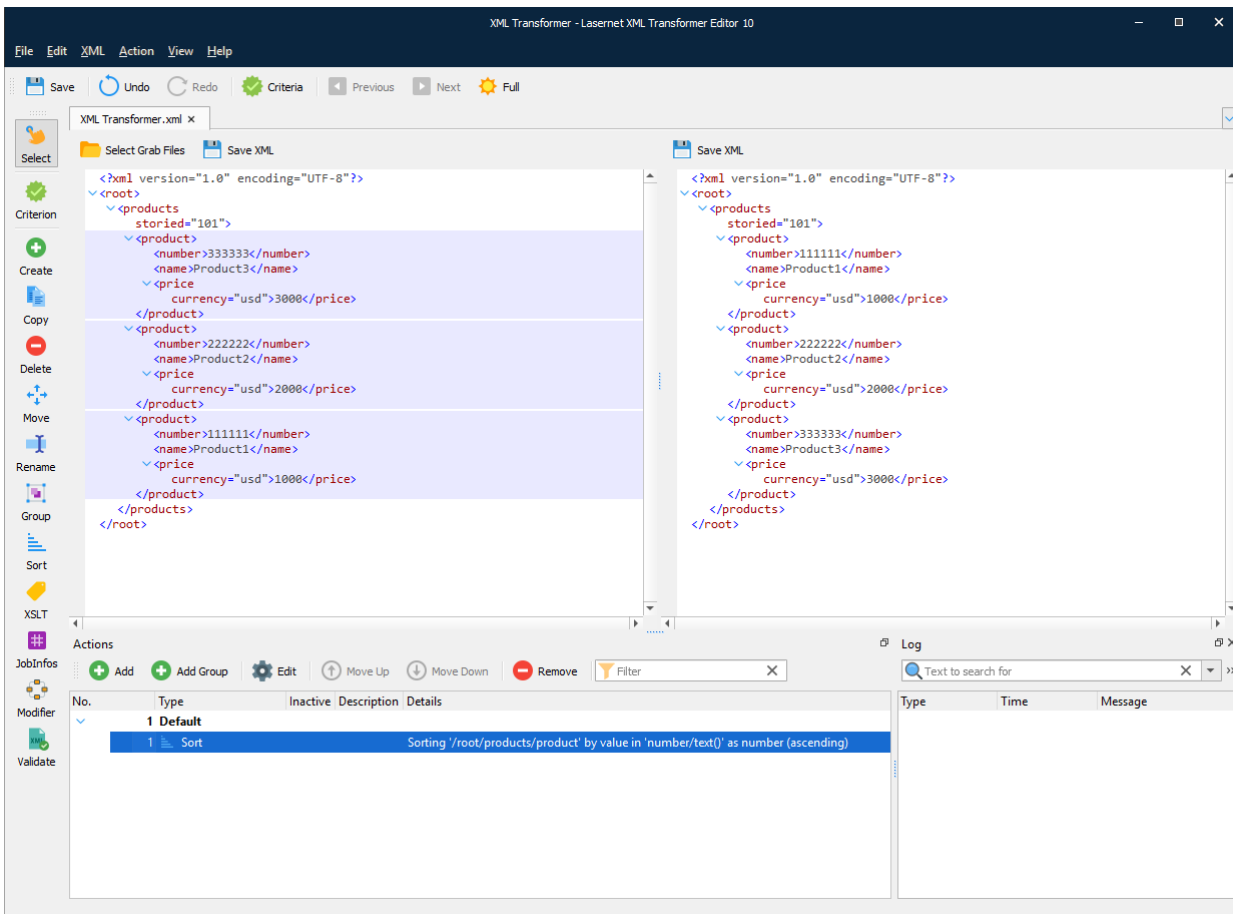
1. On the developer toolbar, click the **Sort** button.



2. Once the cursor turns into a plus sign, click the node you want to sort.
3. Next, click a node value to sort by.
4. With the **Add Action** dialog displayed, select the type to sort (as **String** or **Number**), as well as the direction (**Ascending** or **Descending**). Once you are done, click **OK**.



The following image shows the result of the sort action.



3.3.2.8 XSLT

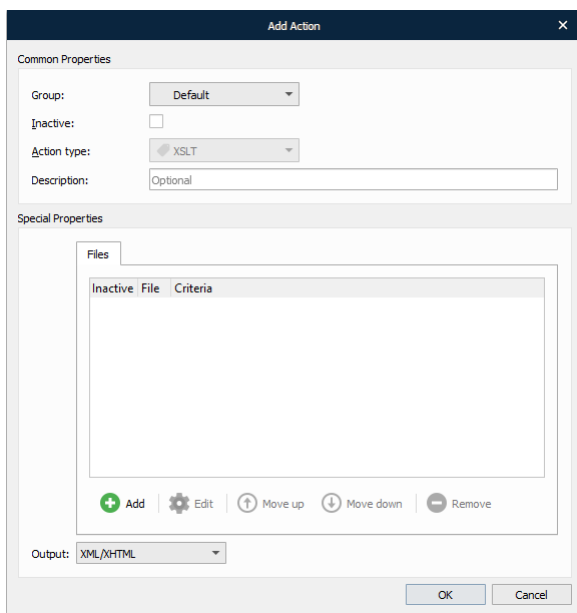
Using the XSLT action type, you can create advanced transformations. The XSLT action applies an XSLT file. The root node of an XML document normally undergoes the XSL transformation, but different behaviour can be defined in the XSL itself.

To apply the XSLT action, follow the steps listed below:

1. On the developer toolbar, click the **XSLT** button.



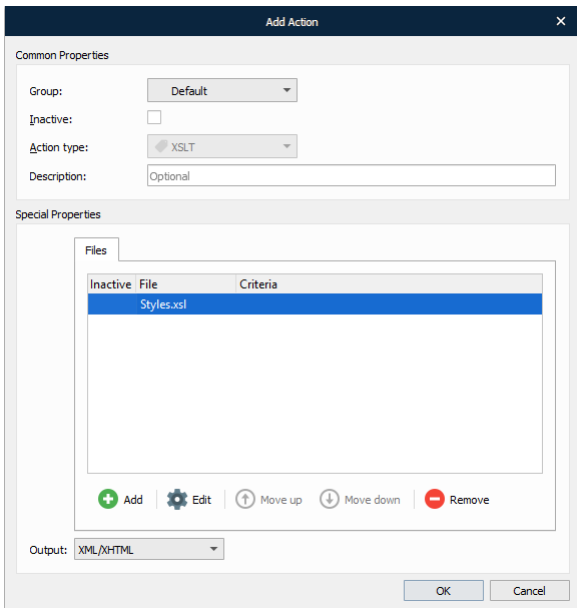
2. With the **Add Action** dialog displayed, click the **Add** button to add an XSLT file for transforming the XML input file.



3. With the **File** dialog box displayed, browse the location of the XSLT document you want to add.

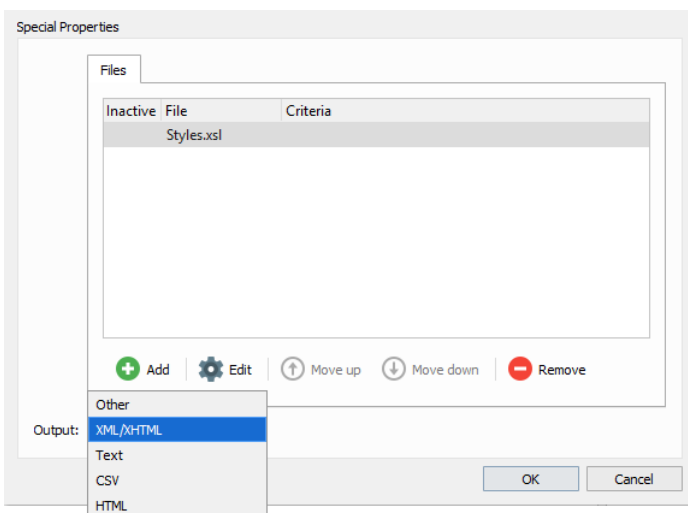
Use the **Criteria** tab to add JobInfos and specify criteria, if required. JobInfo criteria provide the user with additional methods for recognizing input data files.

Once you are done, click the **OK** button.



Output

You can create a wide range of output formats via XSLT. Select **Other** if your output format is not available in the list or **XML/XHTML** (default), **Text**, **CSV** or **HTML**. The selected output format has no influence on the final output, the application only prevents you from adding/running unsupported actions after XSLT is inserted/executed. Only the XML/XHTML formats are able to run additional actions after the stylesheet is processed.



3.3.2.9 Create JobInfos

The create JobInfos action allows to create your own JobInfos, i.e. variables.

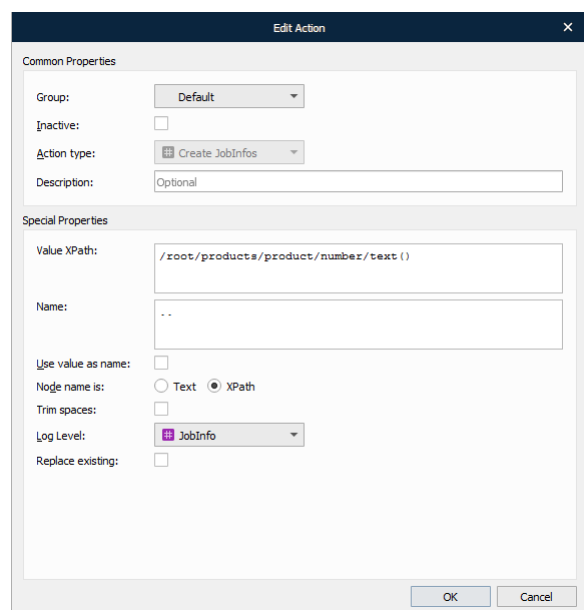
To create a JobInfo, proceed through the steps listed below:

1. On the developer toolbar, click the **JobInfos** button.



2. Once the cursor turns into a plus sign, click the node for which you want to create a JobInfo.

3. With the **Add Action** dialog displayed, fill in the field, as required, and click **OK**.



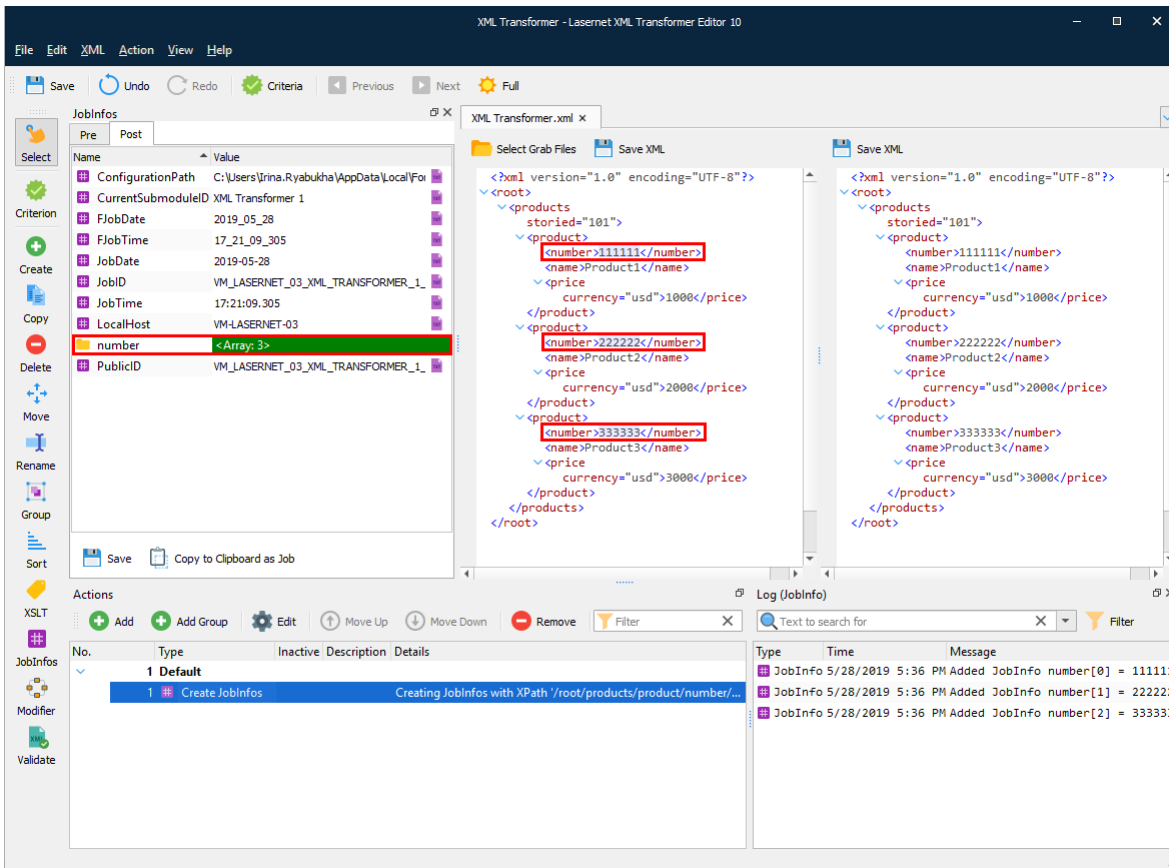
The table below provides a brief description of the action properties:

Property		Description
1	Value XPath	The XPath value specifies which nodes contain the values for JobInfos.
2	Name	Specify a JobInfo name.
3	Use value as name	Select the check box to apply the Value XPath as a JobInfo name.
4	Replace existing	Select the check box to replace any other JobInfo with the same name. Some JobInfos can contain a list of several values for example, an email distribution list.
5	Node name is	Text – select the radio button to interpret the value indicated in the Name field as text. XPath – select the radio button to interpret the value indicated in the Name field as an XPath.
6	Log level	To extract the JobInfos from the document and create them on the current job, you need to specify a category under which the events related to the JobInfo are logged. JobInfo The events related to the JobInfo are logged under the JobInfo category. Debug The events related to the JobInfo are logged under the Debug category. NoLog The events related to the JobInfo are not logged.

Properties defined in the **Add Action** dialog, shown above, take all underlying text-nodes in the JobInfos element, regardless of the position within the element. These are values of the JobInfos. The XPath name is

relative to the current XPath value. Since the element contains all the text-nodes and the **Use value as name** check box is cleared, the parent (specified as '..') takes the name of the element.

The following image shows the list of created JobInfos.



3.3.2.10 Validate

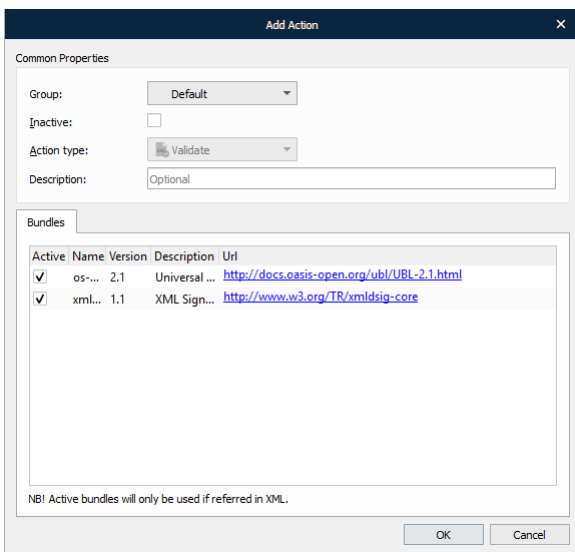
XML validation is a way to verify that data conforms to the associated schema prior to its delivery to another system. The XML Schema describes a valid structure for the XML document.

To apply the validate action, follow the steps listed below:

1. On the developer toolbar, click the **Validate** button.



2. With the **Add Action** dialog displayed, you can clear any **Active** checkbox(es) next to schema bundle(s) that you *do not want to use* during validation.

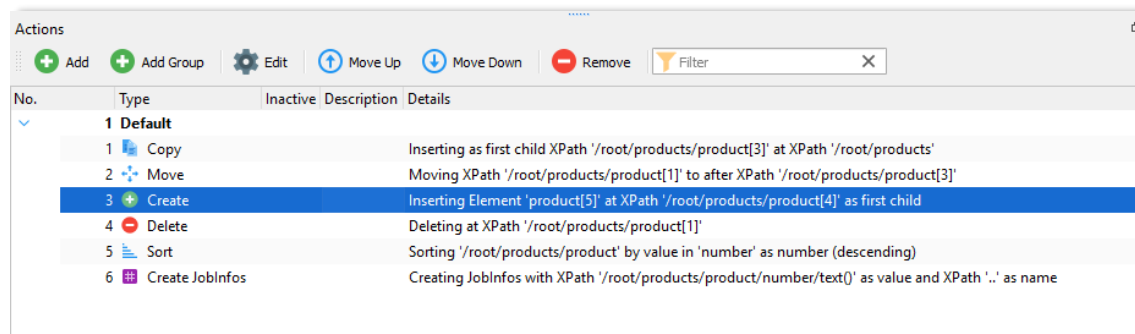


3. Once you are done, click the **OK** button.

For more details about validation, see *XML Validator* in Chapter *Developing with Lasernet*.

3.3.3 Managing Actions and Groups

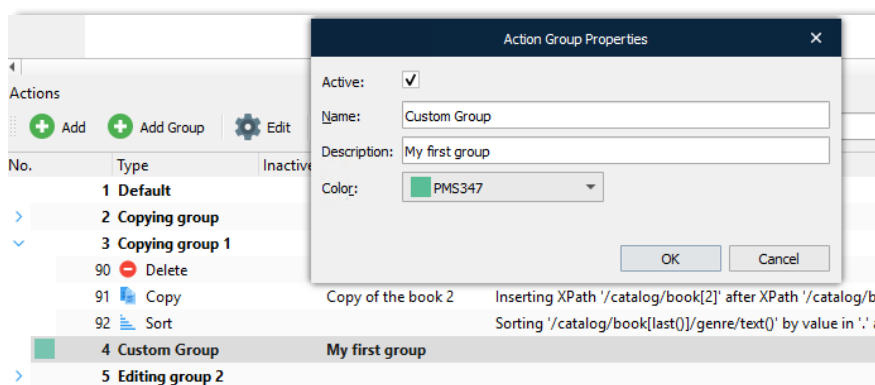
Once you have created an action or a group, it is shown in the **Actions** panel of the **XML Transformer Editor** form. For more details, see *Overview*.



Use the panel to add an additional action or a group, edit or remove existing actions/groups, change the order of actions/groups, etc. The following commands are available on the **Actions** panel:

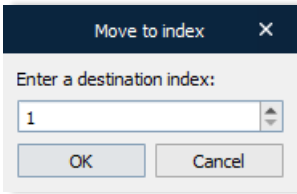
- **Add** – Displays the Add Action dialog. Manually fill in the required fields and click **OK**.
Note: This is an alternative to using the toolbox commands. The options in the dialog will change depending on the selected Action type.
- **Add Group** – Displays the Action Group Properties dialog, enabling you to Add a group. Fill in the fields as required and click **OK**.

Note: The new group will be inserted after the Group in which the selected Action is contained.



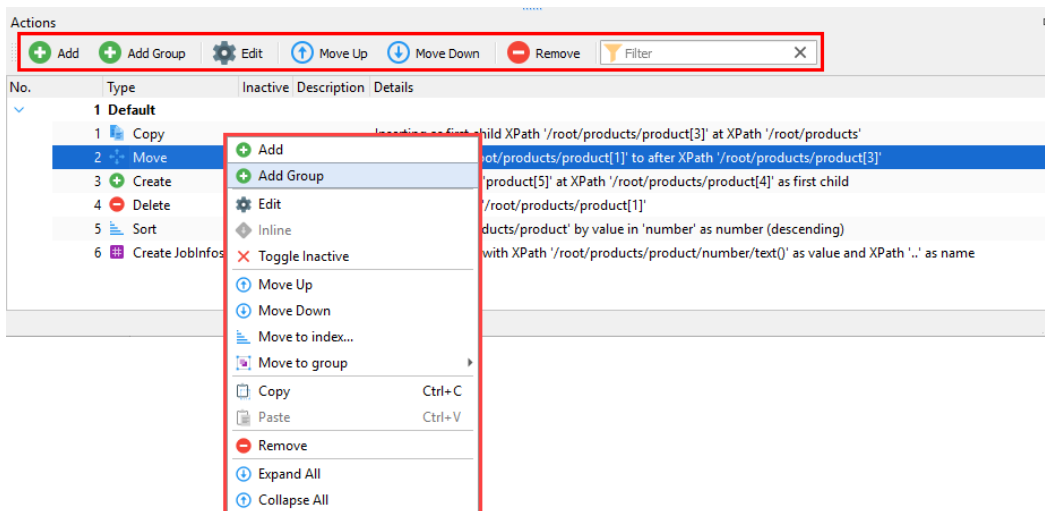
- **Edit** – Displays a dialog enabling you to edit the selected Action(s) / Group(s). Alternatively, double-click an Action(s) / Group(s) to edit it. Modify the settings as required and click **OK**.
- **Toggle Inactive** – Toggles the selected Action between active / inactive states.
- **Move Up** – Moves the selected Action(s) / Group(s) up.
- **Move Down** – Moves the selected Action(s) / Group(s) down.

- **Move to index...** – Displays the Move to index dialog, enabling you to move the selected Action(s) to a specified index.

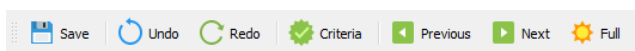


- **Move to group** – Displays a menu from which you select a group. The selected Action(s) are added to the end of the selected Group.
- **Copy** – Copies the selected Action(s) / Group(s) to the clipboard.
- **Paste** – Pastes the Action(s) to the selected Index.
Note: If you copy multiple groups, all Actions will be pasted into the same group.
- **Remove** – Removes the selected Action(s) / Group(s).
- **Expand All** – Expands all Groups.
- **Collapse All** – Collapses all Groups.

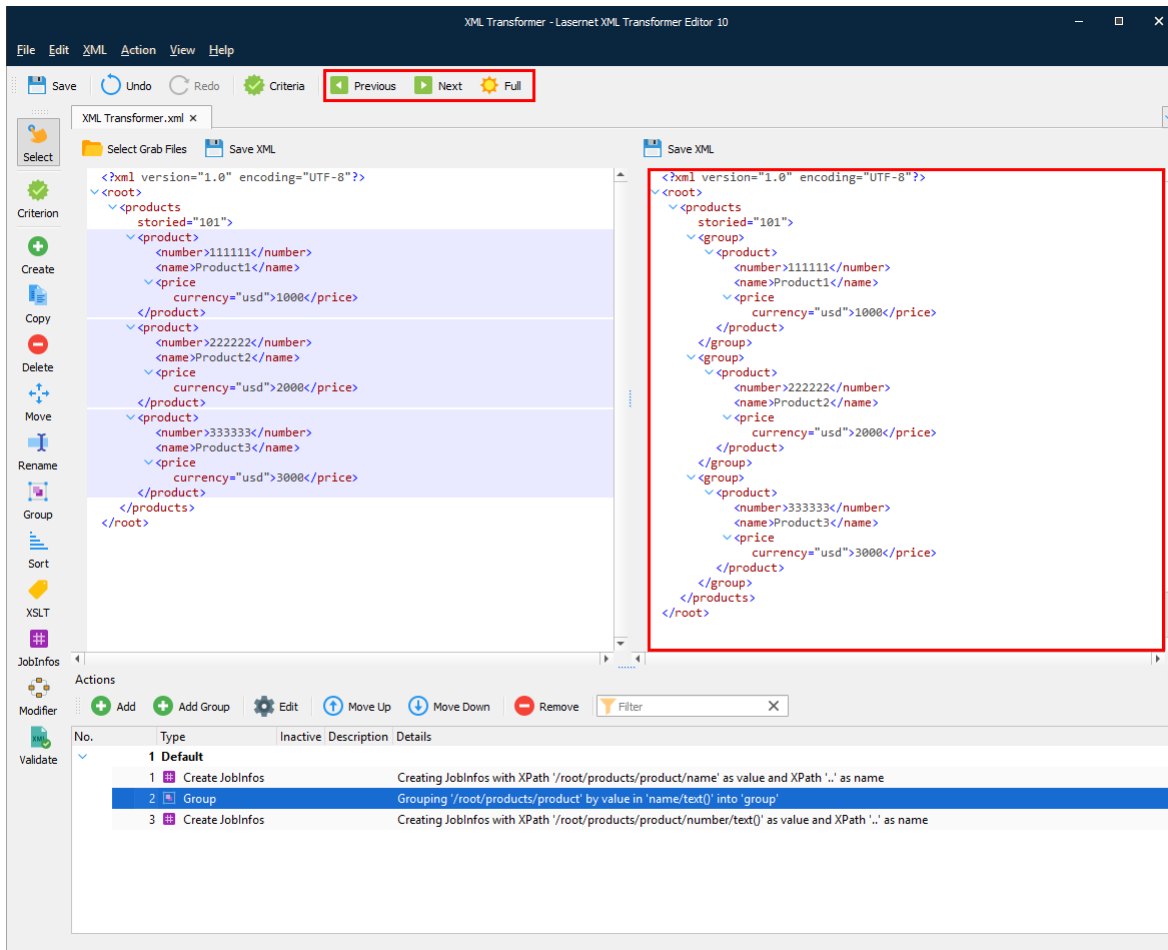
Alternatively, use the context menu to perform any of the actions mentioned above.



In addition, you can also manage actions using the toolbar.



To apply all the actions you have added, click the **Full** button on the toolbar. You can view the final result in the output XML file.



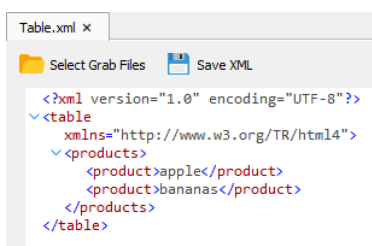
To view the results of the previously applied action, click the **Previous** button.

To view how the next action in the list affects the XML file, click the **Next** button.

3.4 XML Namespaces

XPath 1.0 is unable to find nodes in a default namespace unless the namespace is mapped to a known prefix.

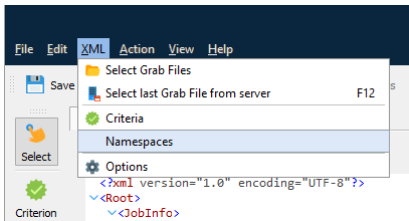
The picture below shows a default namespace. As can be seen in this example, no prefixes are used.



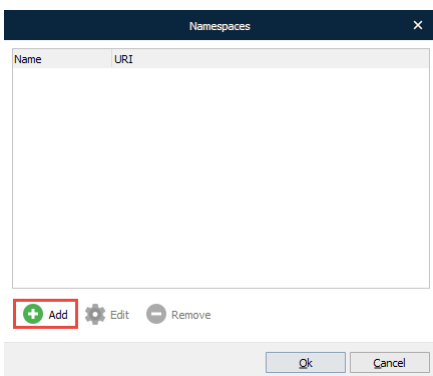
If you are to create an action (e.g. Delete) to any element of the output, it is not transformed. To correctly transform the file, you have to create a namespace.

To add a namespace, follow the steps listed below:

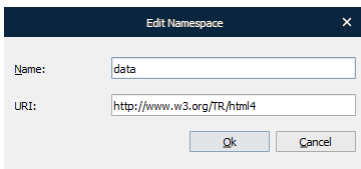
1. On the **XML** menu, click **Namespaces**.



2. With the **Namespaces** dialog displayed, click the **Add** button.

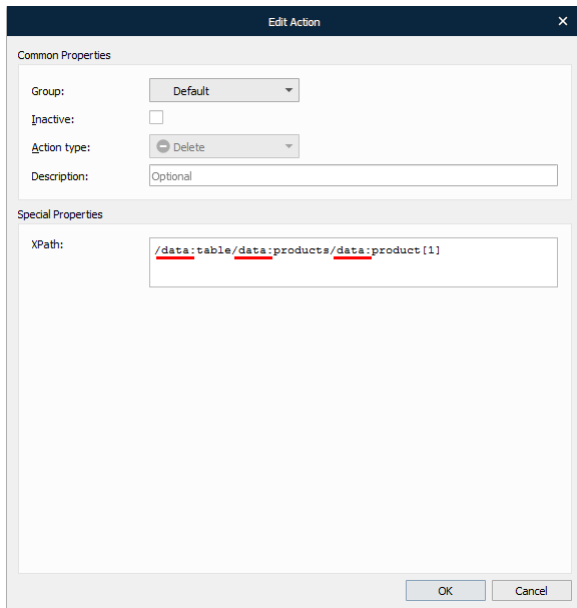


3. With the **Edit Namespace** dialog displayed, type a name for the namespace which is a prefix and URI.

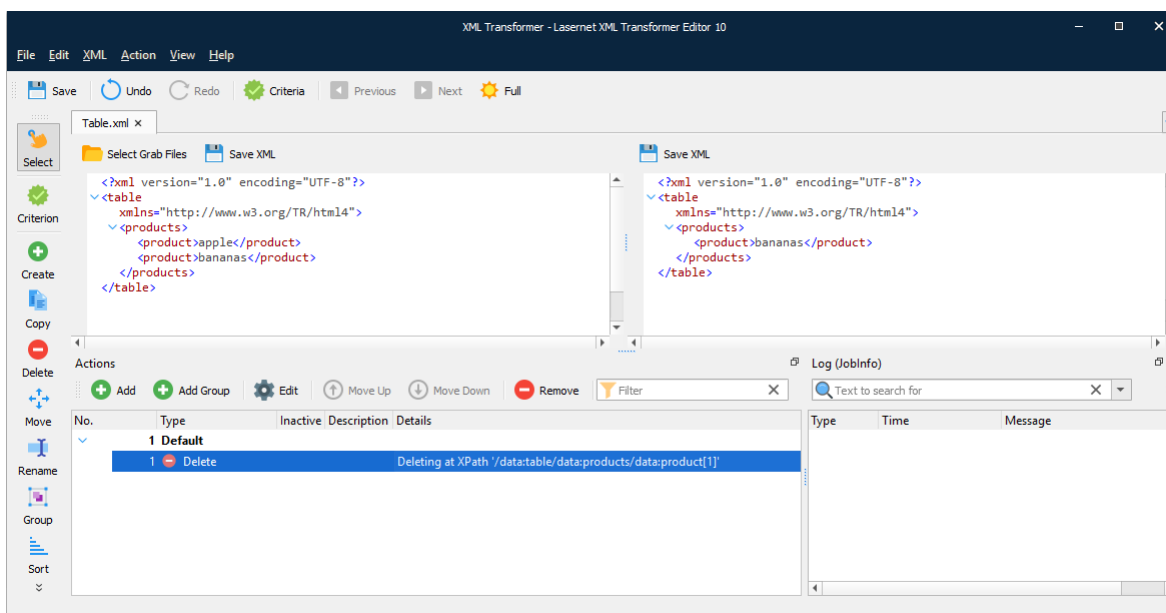


4. Once you are done, click **OK**.

Next, edit the action you have created (e.g. the [Delete](#) action). Within the **Add Action** dialog, type a prefix in the **XPath** field for the node, as shown in the picture below:



As a result, the action is applied:



In addition, the prefix defined in the **Namespaces** dialog allows you to override any prefix defined in the XML file. This is useful when an incoming XML file uses a default namespace or when the prefixes in the incoming XML file vary from one document to another.

The picture below shows two XML files which contain a common URI and different prefixes.

Table1.xml

```

1 <root>
2
3 <h:table xmlns:h="http://www.w3.org/Products/html4/">
4 <h:Products>
5 <h:product>Apples</h:product>
6 <h:product>Bananas</h:product>
7 </h:Products>
8 </h:table>
9
10 <f:table xmlns:f="http://www.w3schools.com/furniture">
11 <f:name>African Coffee Table</f:name>
12 <f:width>80</f:width>
13 <f:length>120</f:length>
14 </f:table>
15
16 </root>

```

Table2.xml

```

1 <root>
2
3 <d:table xmlns:d="http://www.w3.org/Products/html4/">
4 <d:Products>
5 <d:product>Apples</d:product>
6 <d:product>Bananas</d:product>
7 </d:Products>
8 </d:table>
9
10 <f:table xmlns:f="http://www.w3schools.com/furniture">
11 <f:name>African Coffee Table</f:name>
12 <f:width>80</f:width>
13 <f:length>120</f:length>
14 </f:table>
15
16 </root>

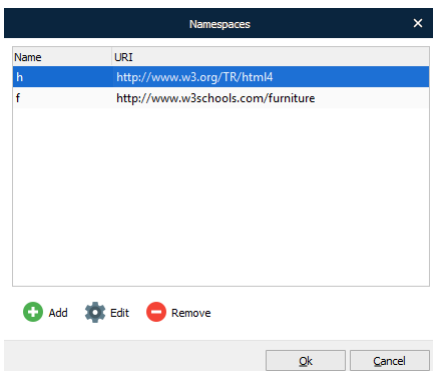
```

To override prefix defined in the XML file, follow the steps listed below:

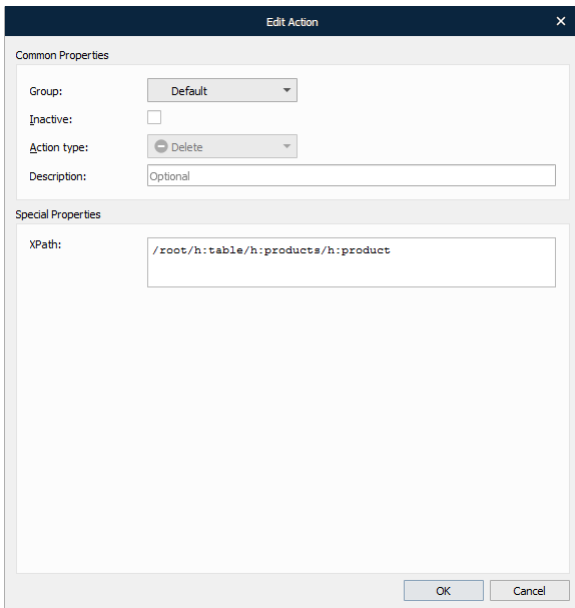
1. Open one of the available XML files by clicking the **Select Grab Files** button in the *XML Transformer Editor*.

Namespaces are automatically added.

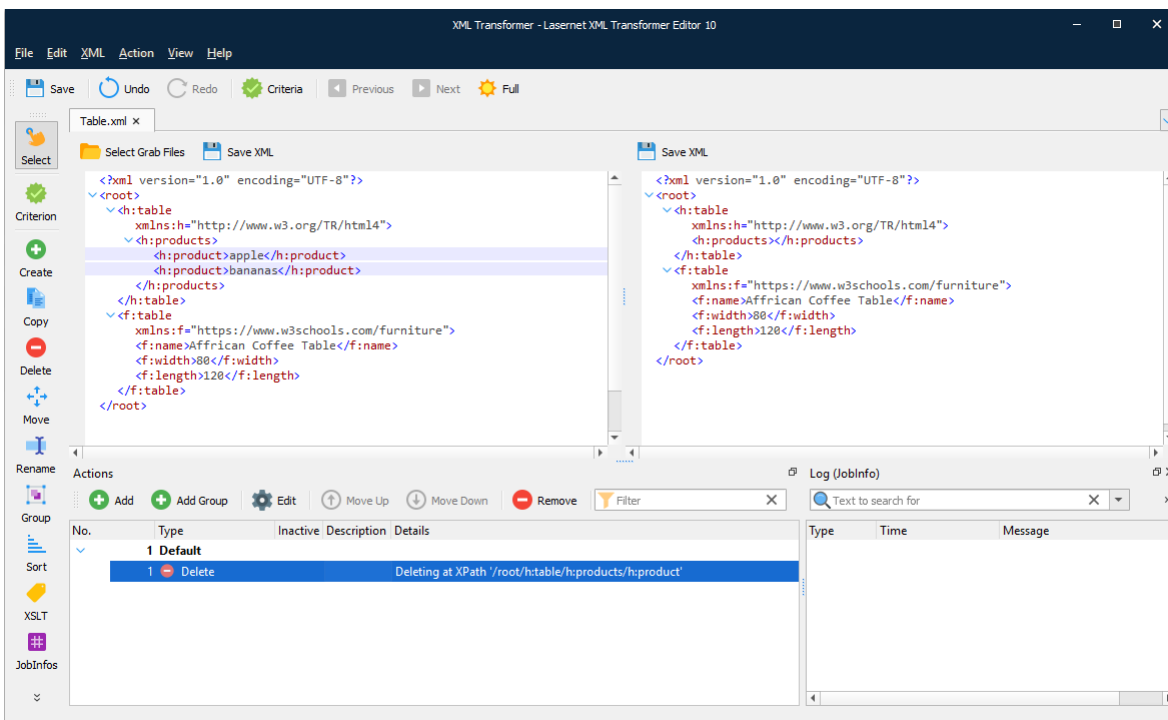
To access the list of namespaces which are automatically added, click **Namespaces** on the **XML** menu.



2. Create an action to apply it to any element of the output (for example, the [Delete](#) action), and the output is transformed accordingly.



3. Afterwards, the same action is applied to all documents regardless of the prefixes available. As the result, each file is transformed in the same way.



3.5 Keyboard Shortcuts

You can use keyboard shortcuts for different operations in Lasernet. A list of keyboard shortcuts is provided in the table below.

Action	Shortcut
Close grab file	CTRL + F4
Switch to right grab	CTRL + Tab
Switch to left grab	CTRL + SHIFT + Tab
Save	CTRL + S
Undo	CTRL + Z
Redo	CTRL + Y
Full	CTRL + A
Switch to Select Tool	CTRL + ALT + S
Add Criterion	CTRL + ALT + K
Add Create Action	CTRL + ALT + C
Add Copy Action	CTRL + ALT + T
Add Delete Action	CTRL + ALT + D
Add Move Action	CTRL + ALT + M
Add Rename Action	CTRL + ALT + R
Add Group Action	CTRL + ALT + G
Add Sort Action	CTRL + ALT + A
Add XSLT Action	CTRL + ALT + H
Add JobInfos Action	CTRL + ALT + P
Add Modifier Action	CTRL + ALT + L
Add Validate Action	CTRL + ALT + V
Find	CTRL + F
Find Next	F3
Find Previous	SHIFT + F3
Show Logger Window	CTRL + SHIFT + L
Grab JobInfos	CTRL + SHIFT + G
Recalculate output	F5