

## How to use X-Path expressions in XML Conditional Patterns?

- 2023-09-01 - Comments (0) - Lاسernet Developer FAQs

# Lasernet

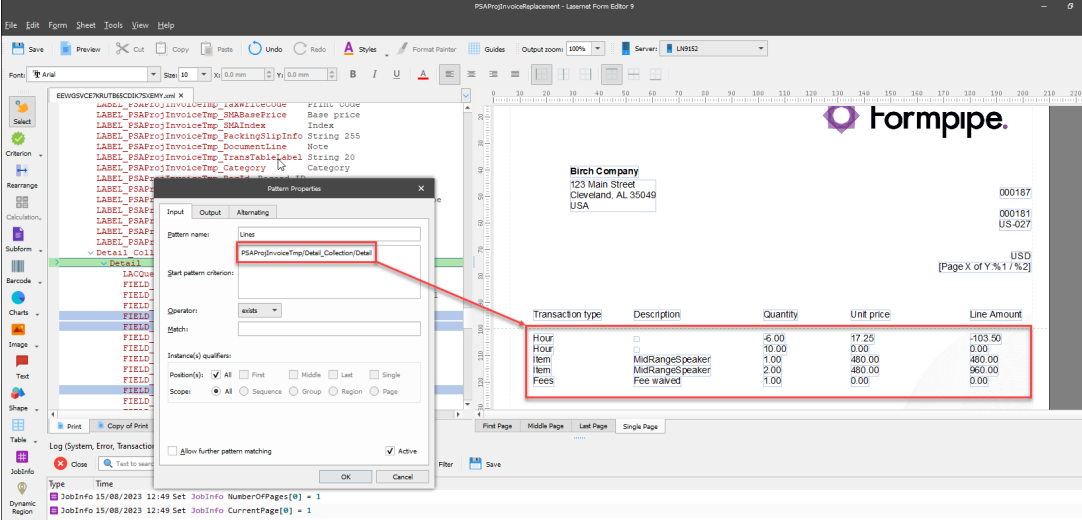
The Lasernet Form Editor allows for the use of Patterns to identify line items. Patterns can be defined based on the "Start pattern criterion." Start pattern criterion can be defined simply by using the XPath to each line item. However, expressions can be used within the Start pattern criterion to further specify which lines to pick up. This article provides a guide on how to build conditionals for Start pattern criteria by showing examples.

XPATH is a query language for selecting nodes from an XML document. XPATH was not invented or created by Formpipe, but it is used by Lاسernet Form Editor.

You will require a good knowledge of X-PATH rules, because certain operators you may want to use are not allowed in X-Path, for example Negative number comparisons.

### Examples

The example used in the article will be for a project invoice with five lines. This is a basic Start pattern criterion that validates the xpath existence so it can identify all the lines of this project invoice. As shown, this example has five lines, two of which are type Item, two of type Hour, and one of type Expense.



The screenshot displays the Lاسernet Form Editor interface. On the left, a list of XML elements is shown, including `EEWOSVCENRUTBSCONVSEMYand X`. A pattern definition is visible, with the pattern name `PSAProInvoiceTmp_Detail_Collection/Detail` and the start pattern criterion `exists`. The pattern properties dialog box is open, showing the pattern name and start pattern criterion. The main form area displays a project invoice for **Birch Company**, 123 Main Street, Cleveland, AL 35049, USA. The invoice includes a table with the following data:

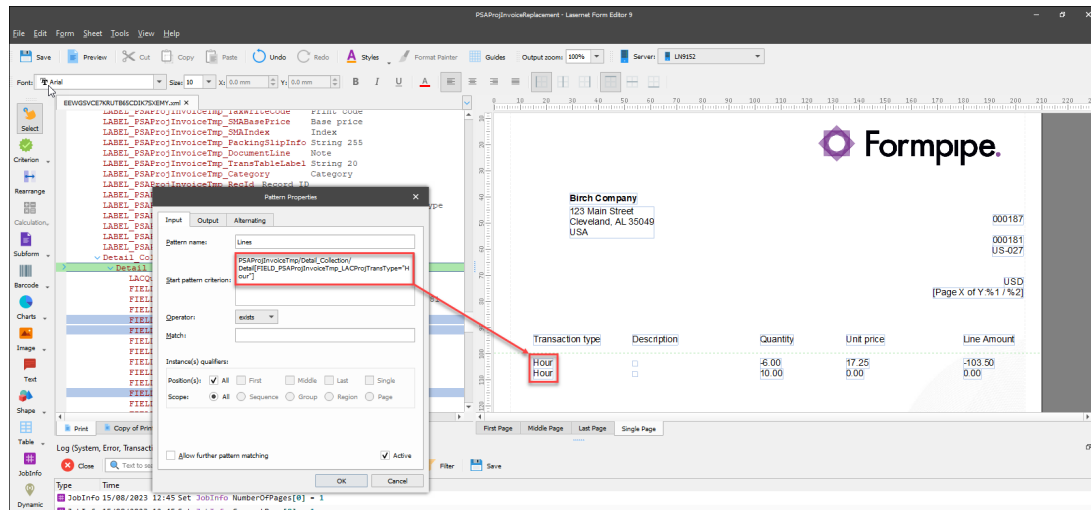
Transaction type	Description	Quantity	Unit price	Line Amount
Hour		6.00	17.25	-103.50
Hour		10.00	0.00	0.00
Item	MidRangeSpeaker	1.00	480.00	480.00
Item	MidRangeSpeaker	2.00	480.00	960.00
Fees	Fee waived	1.00	0.00	0.00

The invoice also shows a total of **USD 000187** and a page number of **Page X of Y: 1 / 2**.

## Validate a specific field

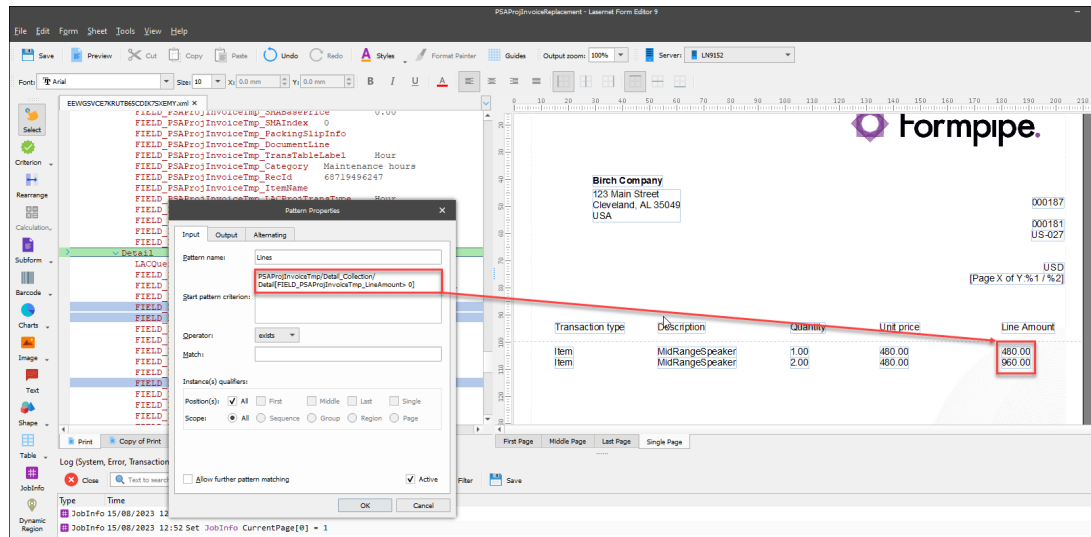
Add square brackets to the end of the XPath, followed by the complete field name, an operator, and your value to check for a specific field within the pattern. This pattern is using the '=' operator, but you can also use '<>' to check that the fields do NOT equal the value.

Example: Only show invoice lines of transaction type "Hour".



Here is another example of validating a specific field. In this case, the field is a number and so we can use operators like '>' or '<,'

Example: Only show invoice lines with a line amount greater than 0.



## Specify which line number to print

Add square brackets to the end of the XPath, followed by the line number you wish to display.

Example: Only show the 5th invoice line.

The screenshot shows the Formpipe interface. On the left, the 'Pattern Properties' dialog is open, showing the 'Input' tab. The 'Pattern name' is 'Lines'. The 'Start pattern criterion' is 'PSAProjInvoiceTmp\_Detail\_Collection[Detail[5]]'. The 'Operator' is 'exists'. The 'Match' field is empty. The 'Instance(s) qualifiers' section shows 'Position(s)' with 'All' selected, and 'Scope' with 'All' selected. The 'Active' checkbox is checked. On the right, a sample invoice is displayed. The invoice header shows 'Birch Company' with address '123 Main Street, Cleveland, AL 35049, USA'. The invoice number is '000187' and the date is '000181 US-027'. The currency is 'USD' and the unit is '1 / 1'. The invoice table has columns: 'Transaction type', 'Description', 'Quantity', 'Unit price', and 'Line Amount'. The table contains one row: 'Fees', 'Fee waived', '1.00', '0.00', '0.00'.

### Combine different conditions

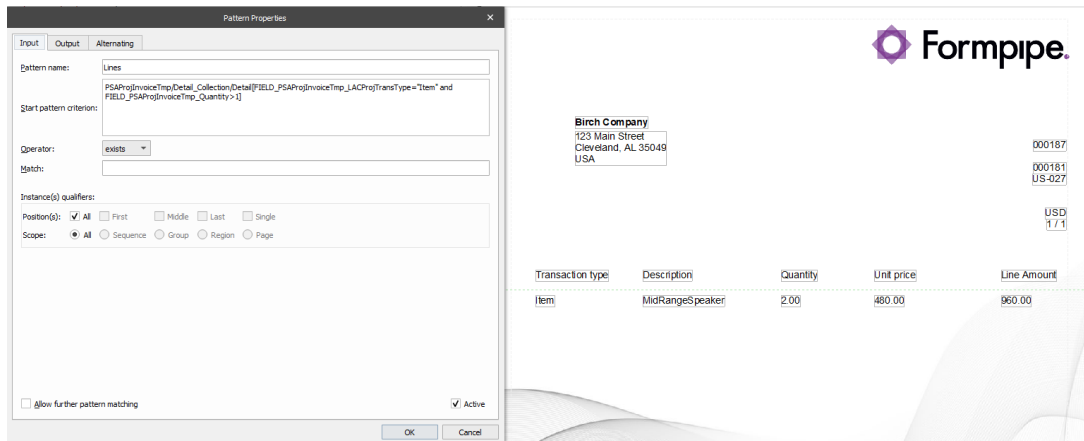
Add multiple conditions by defining each one within square brackets. In this case, you are not using operators but the behavior is similar to using an AND operator. These conditions must all be true for the line to pass the criterion.

Example: Only show invoice lines of transaction type "Item" with a quantity greater than 1.

The screenshot shows the Formpipe interface. On the left, the 'Pattern Properties' dialog is open, showing the 'Input' tab. The 'Pattern name' is 'Lines'. The 'Start pattern criterion' is 'PSAProjInvoiceTmp\_Detail\_Collection[Detail[FIELD\_PSAProjInvoiceTmp\_LACProjTransType="Item"] [FIELD\_PSAProjInvoiceTmp\_Quantity>1]]'. The 'Operator' is 'exists'. The 'Match' field is empty. The 'Instance(s) qualifiers' section shows 'Position(s)' with 'All' selected, and 'Scope' with 'All' selected. The 'Active' checkbox is checked. On the right, a sample invoice is displayed. The invoice header shows 'Birch Company' with address '123 Main Street, Cleveland, AL 35049, USA'. The invoice number is '000187' and the date is '000181 US-027'. The currency is 'USD' and the unit is '1 / 1'. The invoice table has columns: 'Transaction type', 'Description', 'Quantity', 'Unit price', and 'Line Amount'. The table contains one row: 'Item', 'MidRangeSpeaker', '2.00', '480.00', '960.00'.

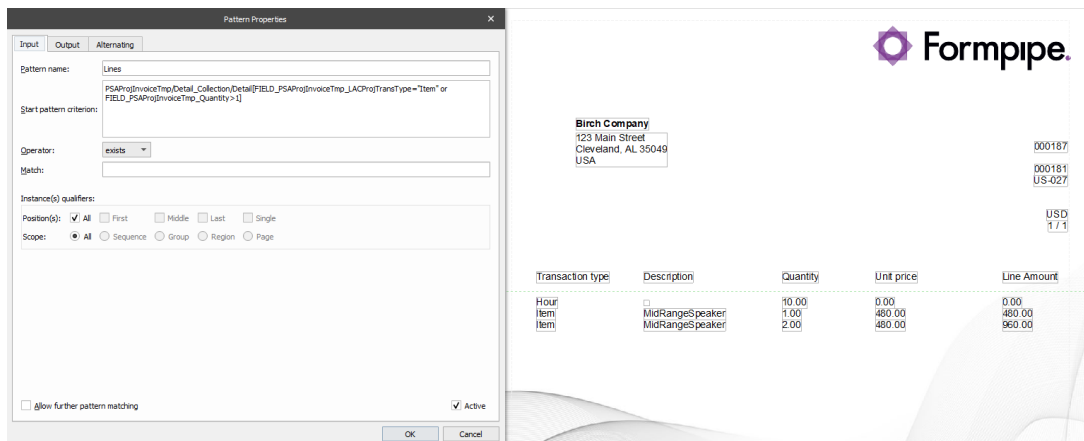
Add multiple conditions within the same square brackets and combine them using 'and'. All conditions before and after the 'and' must be true to pass the criterion.

Example: Only show invoice lines of transaction type "Item" with a quantity greater than 1.



Add multiple conditions within the same square brackets and combine them using 'or'. At least one of the conditions before or after the 'or' must be true to pass the criterion.

Example: Only show invoice lines that are either transaction type "Item" or have a quantity greater than 1.



The following are two examples of a start pattern criterion that combines 'and' and 'or.' Best practice when combining 'and' and 'or' is to use parenthesis to group conditions. Both start pattern criteria are the same, except for the difference in the placement of the parenthesis.

Example: Only show invoice lines that are transaction type "Hour" or

Show invoice lines that have a quantity greater than 1 and a description of "MidRangeSpeakers"

We get the hour lines displayed because they meet the condition before the 'or' operator. We also get the item line because, while it is not type Hour, it does have a quantity greater than 1 and a description of MidRangeSpeaker.

Pattern Properties

Input

Output

Alternating

Pattern name:

Lines

Start pattern criterion:

PSAProjInvoiceTmp\_Detail\_Collection[Detail[FIELD\_PSAProjInvoiceTmp\_LACProjTransType="Hour" or FIELD\_PSAProjInvoiceTmp\_Quantity > 1 and FIELD\_PSAProjInvoiceTmp\_TransTxt="MidRangeSpeaker"]]

Operator:

exists

Match:

Instance(s) qualifiers:

Position(s):

☒ All
☐ First
☐ Middle
☐ Last
☐ Single

Scope:

☒ All
☐ Sequence
☐ Group
☐ Region
☐ Page

☐ Allow further pattern matching
☒ Active

OK

Cancel

Formpipe.

Birch Company

123 Main Street

Cleveland, AL 35049

USA

000187

000181

US-027

USD

1 / 1

Transaction type	Description	Quantity	Unit price	Line Amount
Hour		6.00	17.25	103.50
Hour		10.00	0.00	0.00
Item	MidRangeSpeaker	2.00	480.00	960.00

Example: Only show invoice lines that are either transaction type "Hour" or that have a quantity greater than 1

These lines must also have a description of "MidRangeSpeakers"

We get 1 line that matches this criterion. The line has a quantity greater than 1 and a description of "MidRangeSpeakers"

Pattern Properties

Input

Output

Alternating

Pattern name:

Lines

Start pattern criterion:

PSAProjInvoiceTmp\_Detail\_Collection[Detail[FIELD\_PSAProjInvoiceTmp\_LACProjTransType="Hour" or FIELD\_PSAProjInvoiceTmp\_Quantity > 1] and FIELD\_PSAProjInvoiceTmp\_TransTxt="MidRangeSpeaker"]]

Operator:

exists

Match:

Instance(s) qualifiers:

Position(s):

☒ All
☐ First
☐ Middle
☐ Last
☐ Single

Scope:

☒ All
☐ Sequence
☐ Group
☐ Region
☐ Page

☐ Allow further pattern matching
☒ Active

OK

Cancel

Formpipe.

Birch Company

123 Main Street

Cleveland, AL 35049

USA

000187

000181

US-027

USD

1 / 1

Transaction type	Description	Quantity	Unit price	Line Amount
Item	MidRangeSpeaker	2.00	480.00	960.00

Attached is an object file and a grab file from the examples presented in this article to use for testing/ practice.

## Attachments

- [3C3B7MOYDPNU5LQLWEK5CIFV4U.xml \(23.36 KB\)](#)
- [StartPatternExample.Inobjectx \(64.02 KB\)](#)