

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

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

Lasernet

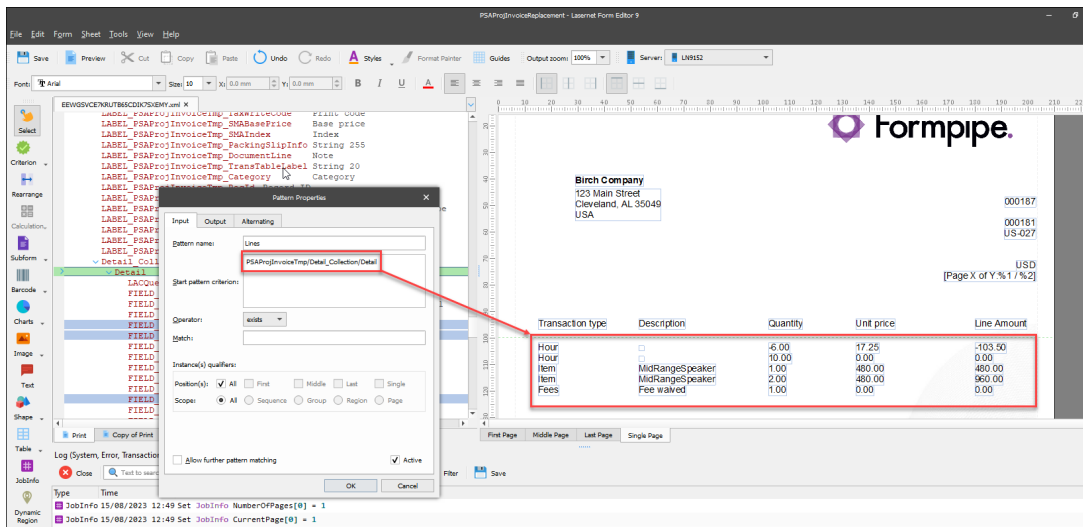
The Lasernet Form Editor allows for the use of Patterns to identify line items. Patterns can be defined based on the "Start pattern criterion." Start pattern criterion can be defined simply by using the XPath to each line item. However, expressions can be used within the Start pattern criterion to further 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 Lasetnet 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 shows the Lasetnet Form Editor interface. On the left, there is a tree view of the form structure. A dialog box titled "Pattern Properties" is open, showing the "Start pattern criterion" field with the XPath expression: `PSAPProjInvoiceTmp/Detail_Collector/Detail`. The "Operator" is set to "exists".

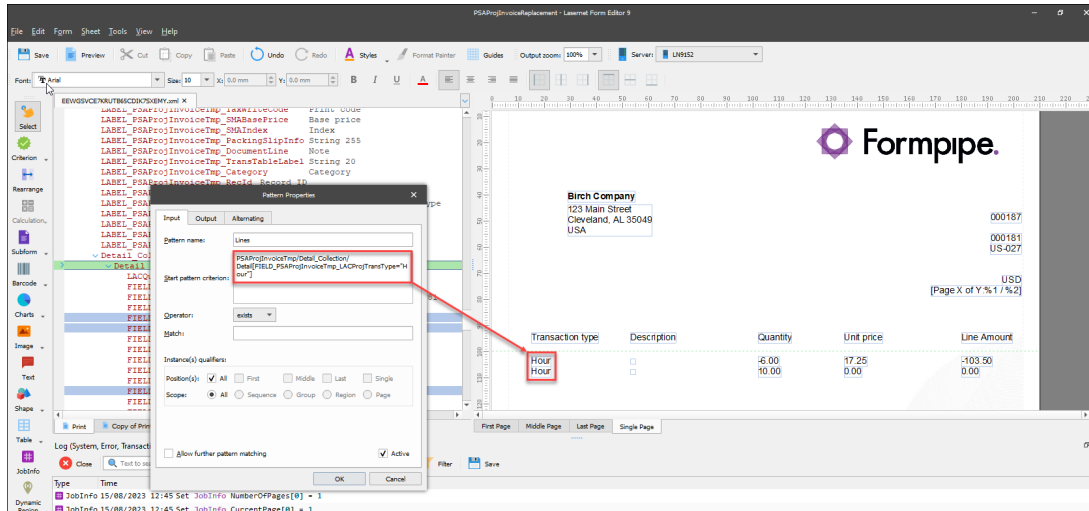
The main preview area shows a form for "Birch Company" with a table of invoice items. The table has the following data:

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

Validate a specific field

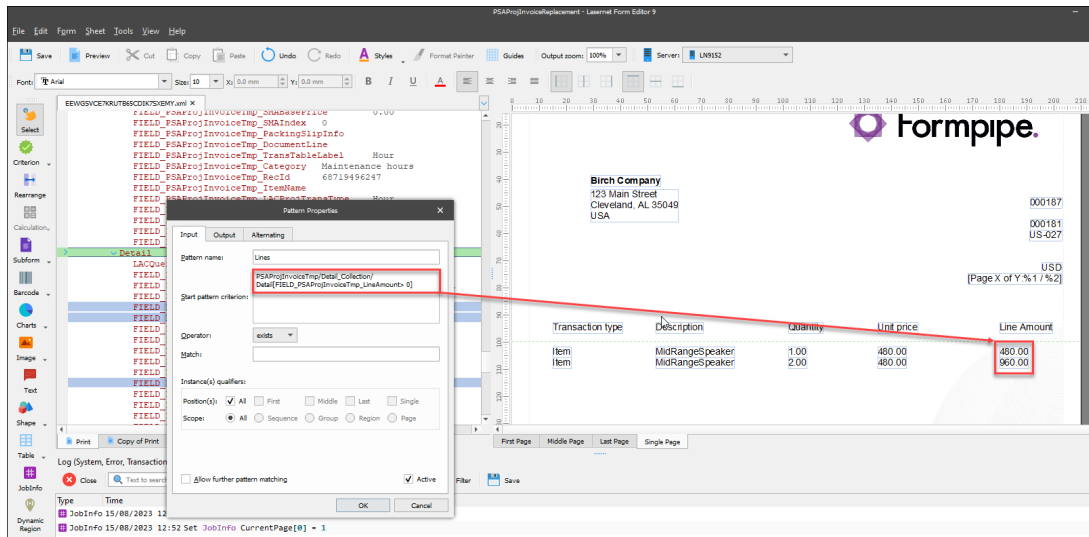
Add square brackets to the end of the XPath, followed by the complete field name, an operator, and your value to check for a specific field within the pattern. This pattern is using the '=' operator, but you can also use '<>' to check that the fields do NOT equal the value.

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 'Pattern Properties' dialog box on the left and an invoice preview on the right. The dialog box has the following settings:

- Pattern name: Lines
- Start pattern criterion: PSAProgInvoiceTmp_Detail_Collection[Detail[5]]
- Operator: exists
- Match: (empty)
- Instance(s) qualifiers: Position(s): All, First, Middle, Last, Single; Scope: All, Sequence, Group, Region, Page
- Show further pattern matching
- Active

The invoice preview on the right shows the following details:

- Company: Birch Company, 123 Main Street, Cleveland, AL 35049, USA
- Invoice numbers: 000187, 000181, US-027
- Currency: USD (1 / 1)
- Table with columns: Transaction type, Description, Quantity, Unit price, Line Amount
- Table 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 'Pattern Properties' dialog box on the left and an invoice preview on the right. The dialog box has the following settings:

- Pattern name: Lines
- Start pattern criterion: PSAProgInvoiceTmp_Detail_Collection[Detail[FIELD_PSAProjInvoiceTmp_LACHogTransType="Item"] [FIELD_PSAProjInvoiceTmp_Quantity > 1]]
- Operator: exists
- Match: (empty)
- Instance(s) qualifiers: Position(s): All, First, Middle, Last, Single; Scope: All, Sequence, Group, Region, Page
- Show further pattern matching
- Active

The invoice preview on the right shows the following details:

- Company: Birch Company, 123 Main Street, Cleveland, AL 35049, USA
- Invoice numbers: 000187, 000181, US-027
- Currency: USD (1 / 1)
- Table with columns: Transaction type, Description, Quantity, Unit price, Line Amount
- Table 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.

Pattern Properties

Input Output Alternating

Pattern name: Lines

Start pattern criterion: PSAProjInvoiceTmp_Detail_Collection[Detail[FIELD_PSAProjInvoiceTmp_LACProjTransType="Item" and FIELD_PSAProjInvoiceTmp_Quantity > 1]

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

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.

Pattern Properties

Input Output Alternating

Pattern name: Lines

Start pattern criterion: PSAProjInvoiceTmp_Detail_Collection[Detail[FIELD_PSAProjInvoiceTmp_LACProjTransType="Item" or FIELD_PSAProjInvoiceTmp_Quantity > 1]

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		10.00	0.00	0.00
Item	MidRangeSpeaker	1.00	480.00	480.00
Item	MidRangeSpeaker	2.00	480.00	960.00

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.

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"

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 \(64.02 KB\)](#)