

# Expressions – Using XPath

## Overview

In this tutorial we'll cover using XPath expressions in configuration item values to allow for runtime evaluation of programmatic instructions. This tutorial expands on concepts covered in "Expressions – Using OGNL," so users are expected to be familiar with that content.

## Steps

In the previous tutorial we walked through the use of OGNL in configuration items, which allowed us to externalize the values. There are other expression types available to eiConsole users, one of which is "xpath".

XPath expressions evaluate against the XML data stream. Within the eiConsole, XPath expressions assume that transaction data stream is validly formed XML.

You can find a reference guide for XPath, as well as more information, at the W3C site at: [http://www.w3schools.com/xsl/xpath\\_intro.asp](http://www.w3schools.com/xsl/xpath_intro.asp)

Basic XPath syntax is shown below:

```
{xpath:/aggregate/child}
Advanced XPath with Namespaces is shown below:
{xpath(ns1=something,ns2=else):/ns1:aggregate}
```

For this tutorial we will be configuring the 'Target file name' based on the Excel Sheet Name. We will use the 'Sample Route' configured in earlier tutorials and the Excel-to-XML Transformer. The basic structure of a Spreadsheet once converted to XML can be seen below.

```
<?xml version="1.0" encoding="utf-8"?>
<XCSExcelBook sheetCount="1">
  <XCSExcelSheet name ="People" rowCount="98">
    ...
  </XCSExcelSheet>
</XCSExcelBook>
```

We will use the following expression to extract the Sheet Name:

```
{xpath://XCSExcelSheet/@name}
```

The configuration of your Transport should look something like this:

The screenshot shows the 'Transport Configuration' dialog box. The 'Transport Configuration' tab is selected, and the 'Basic' sub-tab is active. The configuration details are as follows:

- Transport Name:** Simple Route Outbound
- Transport Type:** Directory
- Use Common Transport:** ☐ No Module Selected
- Transport Description:** Stores data as files in a specified directory.
- Basic Tab Fields:**
  - Target directory:** `$$target_directory=C:\code\substra`
  - Target file name:** `{xpath://XCSExcelsheet/@name}`
  - Target file extension:** xml
  - Append to existing file:** ☐ Append to File ☒ Create New File
  - Maximum File Size:** -1

Run your tests again by dropping the supplied people.xls file into the 'source' folder. Then validate that the 'output' folder contains a file called People.xml.