XML2XLS

Java Excel conversion tool for experiment outputs

Josep Llodrà Grimalt
Jlg.hrtc@gmail.com

XML2XLS

A tool to convert an XML experiment output file to an Excel spreadsheet.
It can handle both Petri Nets and Queueing Networks.

How to run

>java –jar XML2XLS.jar (-pn|-qn) <input_file.xml> <output_file.xls>

Classes
Main.class: Main class, to deal with command-line.
NetConverter.class: Superclass providing XML validation.
PeriNetConverter.class: Loads an XML file, validates it with the corresponding schema, walks through the DOM and writes the corresponding XLS file for a Petri Net experiment output.
QueueNetConverter.class: Loads an XML file, validates it with the corresponding schema, walks through the DOM and writes the corresponding XLS file for a Queue Net (pmif) experiment output.
XLSCreator.class: Class providing the necessary methods to create and write Excel (xls) files.

Basic files

- XML2XLS.jar – Java archive.
- PN-ExResults.xsd – Schema for Petri Nets experiment outputs. It's very important to have this file in the same folder you are running the program, or in the path.
- PMIF-ExResults.xsd – Schema for Queue Nets experiment outputs. It's very important to have this file in the same folder you are running the program, or in the path.
- Pn.xml – PN experiment output example.
- Qn.xml – QN experiemtn output example.
- Pn.xls – Excel example, resulting of Pn.xml
- Qn.xls – Excel example, resulting of Qn.xml

Extra libraries
The POI project consists of APIs for manipulating various file formats based upon Microsoft's OLE 2 Compound Document format, and Office OpenXML format, using pure Java. In short, you can read and write MS Excel files using Java with it.
This package (version 3.2) is included, so you don't need to download or install at all.

Notes

1) Take care of the encoding, xml version and namespace of your xml document. It must be congruent with the XML Schema Definition file in order to be valid.
Here you have an example of the very first lines:

pmif-ExResults.xsd file:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://www.uib.es/PMIF"
targetNamespace="http://www.uib.es/PMIF"
elementFormDefault="qualified">
```

A **pmif xml** file to be validated and coverted:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Output xmlns='http://www.uib.es/PMIF'>
```

2) SolutionID's names can be repeated. As you will notice, XML2XLS creates a nes sheet for each solutionID, adding a suffix to its name. (e.g. For a SolutionID named "Run1", which is found twice, a sheets named "Run1_1" and "Run1_2" are created).