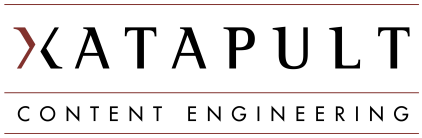
xtpxlib-xoffice

Conversions for Word and Excel files

Erik Siegel - Xatapult Content Engineering
2019-12-11

0 Table of Contents

0 Xatapult XML Library - Conversions for Word and Excel files



xtpxlib library - component **xtpxlib-xoffice** - **v0.9** (2019-12-11)
Xatapult Content Engineering - http://www.xatapult.com - +31 6 53260792
Erik Siegel - erik@xatapult.com

xtpxlib-xoffice is part of the **xtpxlib** library. **xtpxlib** contains software for processing XML, using languages like XSLT and XProc. It consists of several separate components, all named `xtpxlib-*`. Everything can be found on GitHub (https://github.com/xatapult).

This component contains pipelines for converting Microsoft Office Word (`.docx`) and Excel (`.xlsx`) files into some more manageable XML formats.

Installation and usage information can be found on **xtpxlib**'s main website https://www.xtpxlib.org.

Technical information:

Component documentation: https://xoffice.xtpxlib.org
License: GNU GENERAL PUBLIC LICENSE - Version 3, 29 June 2007
Git URI: `git@github.com:xatapult/xtpxlib-xoffice.git`
Git site: https://github.com/xatapult/xtpxlib-xoffice
This component depends on:

- xtpxlib-container (Support for XML containers (multiple files wrapped into one))
- xtpxlib-common (Common component: Shared libraries and IDE support)

1 Description

Microsoft Office files are actually zip files with a lot of XML and other stuff inside. It is remarkably difficult to get to the actual contents of them: What is in Excel cell A1B2 or what is written in this Word document. To help with this, the `xtpxlib-xoffice` component contains XProc (1.0) pipelines to extract contents from Excel (`.xlsx`) and Word (`.docx`) files:

The namespace prefix `xtlxo:` is bound to the namespace `http://www.xtpxlib.nl/ns/xoffice` (`xmlns:xtlxo="http://www.xtpxlib.nl/ns/xoffice"`).

1.1 Converting Excel (.xlsx)

The `xtlxo:extract-xlsx` pipleine takes an Excel `.xlsx` file and turns this into much more manageable XML. The schema for the resulting XML format is here.

Take for instance this simple Excel sheet:

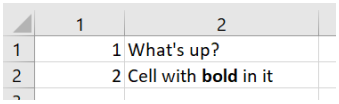


Figure 1-1 - Excel example sheet

Running this through the `xtlxo:extract-xlsx` pipeline returns something like:

```
<?xml version="1.0" encoding="UTF-8"?>
<workbook xmlns="http://www.xtpxlib.nl/ns/xoffice"
          href="file:///path/to/excel.xlsx"
          timestamp="2019-12-11T12:50:20.252+01:00">

   <properties>
      … Sheet properties …
   </properties>

   <worksheet name="Sheet1">
      <row index="1">
         <cell index="1" ref="A1">
            <value>1</value>
         </cell>
         <cell index="2" ref="B1">
            <value>What's up?</value>
         </cell>
      </row>
      <row index="2">
         <cell index="1" ref="A2">
            <value>2</value>
            <formula>A1+1</formula>
         </cell>
         <cell index="2" ref="B2">
            <value>Cell with <span class="b">bold</span> in it</value>
         </cell>
      </row>
   </worksheet>

</workbook>
```

1.2 Converting Word (.docx)

The `xtlxo:extract-docx` pipeline takes a Word (`.docx`) file and turns this into an understandable XML format. This format is more experimental than the format created by the Excel conversion and there isn't (yet) a schema for it.

As an example take this simple Word file:

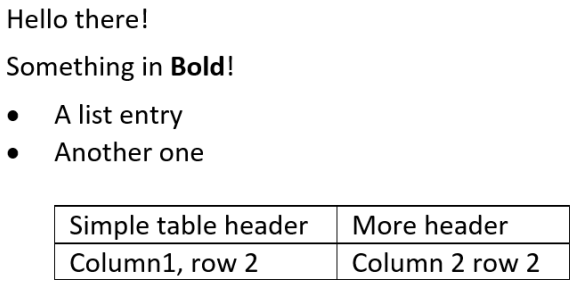


Figure 1-2 - Example Word document

Running this through the `xtlxo:extract-docx` pipeline returns something like:

```
<document xmlns="http://www.xtpxlib.nl/ns/xoffice"
          dref=""
          timestamp="2019-12-11T13:09:15.415+01:00">
   <properties>
      … document properties …
   </properties>

   <p xml:space="preserve">Hello there!</p>
   <p xml:space="preserve">Something in <span class="b">Bold</span>!</p>
   <p class="ListBullet" xml:space="preserve">A list entry</p>
   <p class="ListBullet" xml:space="preserve">Another one</p>
   <p class="ListBullet" indent-left="360" indent-level="0" xml:space="preserve"/>
   <table>
      <tr>
         <td>
            <p class="ListBullet" indent-level="0" xml:space="preserve">Simple table header</p>
         </td>
         <td>
            <p class="ListBullet" indent-level="0" xml:space="preserve">More header</p>
         </td>
      </tr>
      <tr>
         <td>
            <p class="ListBullet" indent-level="0" xml:space="preserve">Column1, row 2</p>
         </td>
         <td>
            <p class="ListBullet" indent-level="0" xml:space="preserve">Column 2 row 2</p>
         </td>
      </tr>
   </table>
   <p class="ListBullet" indent-left="360" indent-level="0" xml:space="preserve"/>

</document>
```

There is an experimental pipeline `xtlxo:create-docx` to create Word documents (using a template Word document for things like styles, margins, etc.). If you feed this the same kind of XML you get from `xtlxo:extract-docx`, the result *should* be a valid, useable Word document with the new text in it. Use at your own risk.

2 XProc Libraries

The xtpxlib-xoffice component contains the following XProc (1.0) library module:

Module	Description
`excel.mod.xpl`	Conversions for Excel (`.xlsx`) files.
`word.mod.xpl`	Conversions for Word (`.docx`) documents.

Table 2-1 - Module overview

2.1 XProc (1.0) library: excel.mod.xpl

File: `xplmod/excel.mod/excel.mod.xpl`
Conversions for Excel (`.xlsx`) files.

Prefix	Namespace URI
`xtlxo`	`http://www.xtpxlib.nl/ns/xoffice`

2.1.1 Step: xtlxo:extract-xlsx

Extracts the contents of an Excel (`.xlsx`) file in a more useable XML format.

Port	Type	Primary?	Description
`result`	out	yes	The resulting XML representation of the Excel file.

Option	Rq?	Default	Description
`xlsx-href`	yes		Document reference of the `.xlsx` file to process (must have `file://` in front).

2.2 XProc (1.0) library: word.mod.xpl

File: `xplmod/word.mod/word.mod.xpl`
Conversions for Word (`.docx`) documents.

Prefix	Namespace URI
`xtlxo`	`http://www.xtpxlib.nl/ns/xoffice`

Step	Description
`xtlxo:create-docx`	Turns Word XML (back) into a Word `.docx` file, using a template file.
`xtlxo:extract-docx`	Extracts the contents of a Word file in a more useable XML format.

2.2.1 Step: xtlxo:create-docx

Turns Word XML (back) into a Word `.docx` file, using a template file.
The input must be in the format the `xtlxo:extract-docx` pipeline creates.

Port	Type	Primary?	Description
`source`	in	yes	The Word XML that must be converted to `.docx` format.
`result`	out	yes	The document-container (see xtpxlib-container) as written to the final Word file.

Option	Rq?	Default	Description
`result-docx-href`	yes		Document reference where to write the resulting `.docx` file (must have `file://` in front).
`template-docx-href`	yes		Document reference of the template `.docx` file to use (must have `file://` in front).

2.2.2 Step: xtlxo:extract-docx

Extracts the contents of a Word file in a more useable XML format.

Port	Type	Primary?	Description
`result`	out	yes	The resulting XML representation of the Word file.

Option	Rq?	Default	Description
`docx-href`	yes		Document reference of the `.docx` file to process (must have `file://` in front).

3 XML Schemas

The xtpxlib-xoffice component contains the following XML Schemas:

Module	Description
`xlsx-extract.xsd`	Schema for the result of an xlsx file extraction to XML (by the extract-xlsx pipeline in the XProc module xplmod/excel.mod)

Table 3-1 - Module overview

3.1 XML Schema: xlsx-extract.xsd

File: `xsd/xlsx-extract.xsd`

Target namespace: `http://www.xtpxlib.nl/ns/xoffice`

Schema for the result of an xlsx file extraction to XML (by the extract-xlsx pipeline in the XProc module xplmod/excel.mod)

Element	Description
`workbook`	