

**xtpxlib**

**An XML processing library**



## 0 Table of Contents

0 XML processing xtpplib library .....	2
1 Description .....	3
2 Usage .....	4
2.1 Installing xtpplib components .....	4
2.2 Running xtpplib components .....	4

## 0 XML processing xtpxlib library



Xatapult Content Engineering - <http://www.xatapult.com> - +31 6 53260792

Erik Siegel - [erik@xatapult.com](mailto:erik@xatapult.com)

**xtpxlib** is an open source library containing software for processing XML, using XSLT (2.0 and 3.0) and XProc (1.0 and 3.0). It consists of several separate components, all named `xtpxlib-*`. Everything can be found on GitHub (<https://github.com/xatapult>). See its [description](#) for more information. Installation and usage instructions can be found [here](#).

The Xatapult XML Library currently consists of the following components:

Component	Description
<a href="#">xtpxlib-common</a>	Common component: Shared libraries and IDE support
<a href="#">xtpxlib-container</a>	Support for XML containers (multiple files wrapped into one)
<a href="#">xtpxlib-xoffice</a>	Conversions between Word/Excel and XML
<a href="#">xtpxlib-xdoc</a>	DocBook publication toolchain

# 1 Description

The **xtpplib** library is an open source library containing software for processing XML, using languages like XSLT and XProc. It consists of several separate [components](#). I've tried to document everything on the accompanying GitHub pages.

Some highlights (the parts I use most often):

- The [xtpplib-common](#) component contains several [XSLT modules](#). For instance:
  - [general.mod.xsl](#) contains a lot of small utility functions and templates. For instance capitalize a string, safe conversions from strings to other data types, etc.
  - [href.mod.xsl](#) is a library I use very often. It contains all kinds of href (filename or web address) juggling functions. For instance get the name/extension/path portion of an href, get/replace/delete its protocol, canonicalize an href (replace the . and . . parts, compute relative href-s, etc.
  - [parameters.mod.xsl](#) contains code for the general handling of parameters and the substitution of parameter values in texts. There is a separate [parameters documentation section](#) about this.
- The <https://container.xtpplib.org/> component contains code for handling *XML containers*: Constructs that provide support for working with multiple related files by wrapping them into a single one. Binary files are referenced instead of included. Also a very useful technique if you need to create a lot of files (like when you're generating a website).
- The <https://xoffice.xtpplib.org/> component contains an [XProc pipeline](#) for turning Excel sheets into much more legible and processable XML. This allows Excel to be used as a user interface to data-driven XML applications.
- The [xtpplib-xdoc](#) component contains a [toolchain](#) for DocBook driven documentation. Among its functionality are [pipelines](#) with mechanisms for generating DocBook from other markup. It also contains custom PDF and HTML generating.

## 2 Usage

### 2.1 Installing xtpxlib components

To install an **xtpxlib** component, first go to its documentation home page to find details about where it is on GitHub and the Git URI you need for cloning it:

Component	Description
<a href="#">xtpxlib-common</a>	Common component: Shared libraries and IDE support
<a href="#">xtpxlib-container</a>	Support for XML containers (multiple files wrapped into one)
<a href="#">xtpxlib-xoffice</a>	Conversions between Word/Excel and XML
<a href="#">xtpxlib-xdoc</a>	DocBook publication toolchain

Then do one of the following:

- Clone its GitHub repository to some appropriate location on disk. The master branch will always contain the latest stable version.
- Or download the latest release zip from its git website (releases page) and unpack it somewhere.

That's all folks...

### 2.2 Running xtpxlib components

**xtpxlib** consists of XSLT (2.0 and 3.0) stylesheets and XProc (1.0 and 3.0) pipelines. To run these you'll need:

- **XSLT:**  
To run XSLT stylesheets, the [Saxon](#) processor is preferred. In most cases the (open source) HE (Home Edition) version will be sufficient.  
As an alternative, run the software from within the [oXygen XML](#) IDE.
- **XProc 1.0:**  
To run the XProc 1.0 pipelines, use the [XML Calabash](#) Xproc 1.0 processor. The xtpxlib library uses several of its non-standard extensions, another XProc processor is therefore not (directly) usable.  
As an alternative, run the software from within the [oXygen XML](#) IDE.  
*WARNING:* The XProc 1.0 pipelines are no longer actively maintained.
- **XProc 3.0:**  
To run the XProc 3.0 pipelines, use the [MorganaXProc-III](#) processor. For some of the more advanced pipelines (mainly the ones in xtpxlib-xdoc) you need the EE version, for which a license is required.