Development/Extension Development

Introduction

Extensions make it possible to extend the functionality of LibreOffice without modifying the source code of LibreOffice itself. LibreOffice has three kinds of extensions.

- If you know how to program, you can use several languages to develop LibreOffice extensions using the UNO component API . Supported programming languages are Basic, Python, JavaScript, Java, BeanShell and C++. The code and data for the extensions are bundled as `.oxt` files and can be shared on the LibreOffice Extensions repository .

  Extensions are the preferred way for sysadmins to modify default settings. A guide for developing configuration extensions is available in PDF or ODT format. A sample OXT configuration extension accompanies the guide.
- There are also special Calc extensions known as Add-Ins that add spreadsheet functions to Calc. As a UNO components you can provide new implementations of existing functionality or specialized components like Calc Add-Ins, Add-Ons, DataPilots, Chart Add-Ins, or linguistic components: Spellchecker, Hyphenator or Thesaurus (see Add-Ins documentation).
- You can also pack extensions for additional Templates, Gallery and Autotexts . See the examples in the "Examples" section below.

About templates

You may have noted that in some examples the templates are put in a folder named "templates", while in some other examples the templates are put in a folder named "template". Don't be confused by the names here. Both will work, provided that you specify the correct name you have used in the Paths.xcu file. Please note that in the Paths.xcu file, the `"%origin%"` as in `<node oor:name="%origin%/template" oor:op="fuse"/>` refers to the location of the extension after it has been installed. For example:

```
file:///<path-to-your-libreoffice-installation-or-config>/user/uno_packages/cache/uno_packages/lu168780135zflp.tmp_/<extenion_filename>.oxt
```

As such, the "template" as in `<node oor:name="%origin%/template" oor:op="fuse"/>` should be the path relative to the root of your un-zipped extension. This means that if you put the actual templates in a folder named "my_templates" when organizing the extension structure, then in Paths.xcu it should be: `<node oor:name="%origin%/my_templates" oor:op="fuse"/>`.

Also, note that the "Template" as in `<node oor:name="Template" oor:op="fuse" oor:mandatory="true">` shown in the Paths.xcu file indicates that this path setting is for the "Template" category, which is defined in unotools/source/config/pathoptions.cxx (together with Addin, AutoCorrect, AutoText, Dictionary, Gallery. etc.).

Resources

You can mostly refer to OpenOffice's documentation , but LibreOffice is moving forwards rapidly (see Compatibility notes below), so you should use our up-to-date documentation. They are still largely compatible.

For a full description of extensions and the `.oxt` file format, see the Developer's guide.

Current resources are currently being consolidated on the LibreOffice help pages .

Before you can start, you need to install the SDK .

Tutorials and examples

To get started, you can take a look at the tutorials or the source code of the example extensions below. The LibreOffice source code also has a collection of SDK examples . Another way to find examples is to unzip the `.oxt` file of an existing extension and look at the contents.

Basic

- Create a Hello World LibreOffice extension

Java

- Starter Extension Starter extension to be used as a base for own extensions
- Sidebar Extension Similiar to starter extension, but for creating an own Sidebar Deck
- NOA-libre is an object-oriented lightweight Java wrapper around LibreOffice UNO API, providing higher-level abstraction of many UNO interfaces.
- UNOHelper offers helper functions for working with the the UNO API. It wraps commonly used functions from the UNO API for easier use.
- LibreOffice Template System (LOTS) is a large Java Extension with enhanced template, form, and autotext functionality.

Eclipse Plugin

TDF offers the Eclipse Plugin **LOEclipse** which helps with developing LibreOffice extensions/components in Java. You can find it on the Eclipse Marketplace .

Python

- unodit automate some of the tedious tasks with dialogs in order to help you write your own extension for LibreOffice in Python
- OOO Development Tools wraps often used functions to make them simpler to use

- the LibreOffice Love Letter Writer

See also the pages on Python Extensions Development, Python Macro Guide and Designing & Developing Python Applications.

C++

- Libreoffice extension development with C++ - Part 1 - Getting to know UNO ;
- the LibreOffice Search extensions which adds a context menu entry in order to search online for the selected text]

Non-coding Examples

- Andreas Mantke's GitHub extensionbook repository contains a collection of non-coding examples explaining how to package a set of colors, templates, auto-text and galleries into an extension. The book is also available as PDF .
- Kevin Suo's GitHub lo_l10n_templates repository is an example of how to make a template-set as a LibreOffice extension and make it localized.

Compatibility notes

The LibreOffice developers are conservative when it comes to changes to the public (UNO) API and thus, most extensions written for OpenOffice.org or Apache OpenOffice will still work with LibreOffice. If changes happen, they are noted in the ReleaseNotes (API changes in 4.2, 4.1 and 4.0).

- The `OpenOffice.org-minimal-version` dependency states that an extension works in the specified OpenOffice.org versions as well as in LibreOffice versions that correspond to at least the specified OpenOffice.org reference version. All LibreOffice versions up to 4.2.3 report that they correspond to OpenOffice.org reference version 3.4, while LibreOffice versions since 4.2.4 report that they correspond to OpenOffice.org reference version 4.1. Using this dependency, the extension declares that it only uses features found in the specified OpenOffice.org reference version (and that it consequently works across all the products, not only for OpenOffice.org itself).
- If an extension makes use of features specific to LibreOffice, starting with LibreOffice 3.5 it can specify a new dependency `LibreOffice-minimal-version` with namespace `http://libreoffice.org/extensions/description/2011`. Its required `value` attribute (without namespace) specifies the minimally supported LibreOffice version; it should not contain a micro version number, and it should not be less than `3.5`.
  The generally required `name` attribute (with namespace `http://openoffice.org/extensions/description/2006`) should be "`LibreOffice` *`version`*" (where *`version`* is the given version value).
  Examples may be found in core/tree/desktop/test/deployment/dependencies : loversion35.oxt or loversion36.oxt.
- If an extension is intended for a specific version of LibreOffice, starting with LibreOffice 7.0 it can specify a new dependency `LibreOffice-maximal-version` with namespace `http://libreoffice.org/extensions/description/2011` in addition to the dependency `LibreOffice-minimal-version`. This should be needed very rarely.

Categories: Development | Extensions

This page was last edited 13:02, 29 May 2025 by Dennis Roczek. Based on work by Han V and Samuel Mehrbrodt and others.

Please note that all contributions to The Document Foundation Wiki are considered to be released under the Creative Commons Attribution-ShareAlike 3.0 Unported License, unless otherwise specified. This does not include the source code of LibreOffice, which is licensed under the GNU Lesser General Public License (LGPLv3). "LibreOffice" and "The Document Foundation" are registered trademarks of their corresponding

registered owners or are in actual use as trademarks in one or more countries. Their respective logos and icons are also subject to international copyright laws. Use thereof is explained in our trademark policy (see Project:Copyrights for details). LibreOffice was based on OpenOffice.org. If you do not want your writing to be edited mercilessly and redistributed at will, then do not submit it here.

Privacy policy Imprint