

LibreOffice Weekly News #2

William Gathoye

June 20, 2014

LibreOffice Weekly News consists of a weekly issue, published at the end of each week, gathering interesting development reported in discussions (mailing lists or IRC), Q/A and git commits made over the past week. This summary will emphase on the developer point of view. This week in LibreOffice...

Contents

1. Releases.	2
2. GUI.	2
2.1. UI glade dialogs.	2
2.2. Non printable characters color.	2
3. License and FOSS.	3
4. Code base evolution.	3
4.1. JDK.	3
4.2. Copy-paste code detector.	3
4.3. Access2Base.	3
4.4. LibreOffice UNO interface.	3
4.5. Plugin NPAPI drop.	4
4.6. Comment translations.	4
5. Build environment.	4
5.1. Cross-compilation.	4
5.2. Build on Windows.	4
5.3. MinGW port.	5
5.4. Reduce compilation time on Windows.	5
6. Other.	5
6.1. DEVONThink plugin.	5
6.2. Assertions and invariance.	5
7. QA - Quality Assurance.	6
7.1. Bug hunting.	6
7.2. Profiling.	6

8. GSoC.	6
8.1. Improvements to the Template manager (Efe Gürkan YALAMAN)	6
8.2. Dialog Widget Conversion (Mihály Palenik and Szymon Kłos).	6
8.3. Improved Color Selection (Krisztian Pinter).	6
8.4. Calc / Impress tiled rendering support (Andrzej Hunt).	7
8.5. Refactor god objects (Valentin).	7
8.6. Improve Usability of Personas (Rachit Gupta).	7
8.7. Enhancing text frames in Draw (Matteo Campanelli).	7
8.8. Connection to SharePoint and Microsoft OneDrive (Mihai Varga).	8
9. Hackfests and conferences.	8
9.1. Paris Hackfest.	8
9.2. Toulouse Hackfest.	8
9.3. LibO Conf.	8
9.4. Linux.conf.au.	8
10.UX / Design team.	8
11.Users.	9

1. Releases.

This week, we released:

- LibreOffice 4.2.5, fixing over 150 bugs and regressions over our previous release[2];
- The first release candidate of LibreOffice 4.3[3];

Aside these software releases, we have also seen a great one from a team most of us tend to forget: the Documentation team. Indeed, the Documentation for the LibreOffice 4.2 branch has been fully completed and has been released[4]. Great news for people who intent to evaluate the possibility to migrate from proprietary Office suites to LibreOffice and see if alternative features are available and how to use them.

2. GUI.

2.1. UI glade dialogs.

The conversion towards the .ui glade dialogs is still continuing (thanks Mihály Palenik!) [12, 13, 14, 15].

Do not forget we have two GSoC students (Mihály Palenik and Szymon Kłos) in charge of the dialog widget conversion too[18]. LibreOffice will use even more .ui dialog in the future.

2.2. Non printable characters color.

It has been discussed to add the ability to select the color of non printable characters, since the light blue color, which is currently used, is too difficult to be read when using a white page (the most common configuration)[6]. Since LibreOffice 4.3 UI is frozen, it should be added to 4.4, and definitely considered as an easy[7].

3. License and FOSS.

LibreOffice has been mentioned in Fortune[19], the business magazine published by Time Inc, editor of the famous TIME magazine. A LibreOffice contributor reacted on the mailing list[20] to clarify the situation: patent and copyrights are 2 different concepts. Even if LibreOffice is a FOSS like many similar projects, it is still copyrighted, but is not patented.

4. Code base evolution.

4.1. JDK.

Fix the JDK dependency on OS X for Rhino[9][10], a FOSS implementation of Javascript written in Java[8] (thanks Robert Antoni Buj i Gelonch). JDK > 1.6 and JDK 1.8 can now be used on OS X.

If you wonder where such a Javascript implementation would be needed: for UNO bridges. See this article to know more about UNO bridges[11].

4.2. Copy-paste code detector.

Last week, we spoke about a copy-paste code detector. Now we are beginning to have some great results from that tool. Duplicated code has been found and will be merged[21].

4.3. Access2Base.

The discussion about Access2Base has made no progress this week, because the discussion on the mailing list has turned into a bunch of strong arguments about this question “should we let Access2Base be upgraded ‘outside-of-cycle’ by the administrator, but in that case LibreOffice internals have to be overridden by the user (or administrators), this could make the install instable or even worse bypass password protected files, yes but we can anyway override any part by overriding the LD_PRELOAD environment variable,... blablabla”. The main discussion is basically now “should we let the user screw up this computer or not”. Nothing new then regarding last week, the decision is still the same “Access2Base will be bundled with LibreOffice as an extension that could be overridden. This override will stay the only exception.”

We should revert the exception, Access2Base doesn't need to be released quick. (Julien Nabet)

4.4. LibreOffice UNO interface.

According to the question asked by Lionel Elie Mamane, the LibreOffice UNO interface[54] is really badly typed, the pointers always need to be casted to be dereferenced. With his work, the code will be easier readable, by removing all this dynamic cast, NULL pointer check, and avoid “generic” pointers to be exception safe[45].

4.5. Plugin NPAPI drop.

The NPAPI LibreOffice plugins is used by less than 5% of our current user base, we need to have more accurate statistics about this subject. In the future, this number will still be reduced: all current browsers tend to block them by default, unless the user click on the content to play the plugin.

Currently, there is no alternative to the NPAPI plugin. "We have confirmed that PPAPI will not work because while Chrome supports it they don't provide any way for third parties to install PPAPI plugins. The only method of doing so involves a command-line argument, which is unsuitable for most purposes[55]."

Last week, we announced the NPAPI support has been dropped from LibreOffice. Currently, it will still be present in 4.3, but not in 4.4.

LibreOffice requires the NPAPI for 2 features:

- to be able to read LibreOffice documents into a web browser itself;
- read embedded Flash content into Impress presentations.

For this last feature, LibreOffice might not drop the NPAPI after all. This has not been decided yet and the decision is still open.

As a workaround to still provide the first feature, if the NPAPI would come to really disappear for 4.4, the idea to reuse the LibreOffice Online project has been brought to the table, but nothing more.

4.6. Comment translations.

About the translation of German comments to English, we saw some great progress here too[16, 17]. Thanks Philipp Weissenbacher!

5. Build environment.

5.1. Cross-compilation.

You might wonder how LibreOffice is cross-compiled for all majors OS? I asked this question on IRC, since the Wiki is not enough complete on this regard.

```
(2014-06-11 17:33:39) cloph: wget: windows builds for 4.2 and below are done on Windows Server
2008R2 with Visual Studio 2010, for 4.3 and later it is Windows Server 2012 with Visual
Studio 2012. Mac OSX release builds are done on Mac OSX 10.6, with xcode 3.something, and
Linux builds are done on CentOS 5.10 (with gcc 4.4)
```

5.2. Build on Windows.

According to the reaction we had on the mailing list, building LibreOffice with Visual Studio for contributors who are unfamiliar with Linux isn't as straightforward as we can think[32]. After a look at the wiki, we can understand this contributor's disarray: the "Build on Windows" article[33] is split in several parts and doesn't take the newcomer by the hand. We should rewrite it. Help is welcome on this topic.

5.3. MinGW port.

The MinGW port of LibreOffice is now dead[37], the only way to generate LibreOffice builds for Windows is to compile it on Windows itself using the Visual C++ compiler from Visual Studio (at least 2010). Yet another reason why upgrading the Windows build instruction on the wiki is so important.5.2.

5.4. Reduce compilation time on Windows.

Last week, we talked about the possibility to reduce compilation time on Windows. Several solutions have been proposed MinGW, GnuWin32, busybox, keep Cygwin and use a more native gmake utility provided by GnuWin32, or use MSYS.

This week, Michael Stahl made some MSYS tests. His first feelings about it are quite off-putting: this is quite different from Cygwin and he has some huge problems with paths. These paths were making the first step of the compilation, the `autogen.sh` file, hang[40].

6. Other.

6.1. DEVONThink plugin.

An user has reported that LibreOffice couldn't work with DEVONThink, a FOSS document organizer and previewer for OS X, while other old branches of LibreOffice (OO.o and NeoOffice) were able to do so. It seems like the reason comes from the removal of a quicklook plugin into LibreOffice. The seldom bugs found Bugzilla did not come in handy this time[5].

6.2. Assertions and invariance.

Florian Reisinger wrote a trivial document explaining why assertions (which could be used in unit tests) are so much important to avoid and detect bugs. He gave the example of a linked list. To manipulate this list, we have methods like `addElement`.

Three conditions are playing a role in a method: the precondition (the input), the postcondition (output) and the invariance. Invariance is typically what should be valid at the beginning and at the end of the method. Invariance is playing a huge role in this bug hunting context. These 2 locations (begin and end of method) will be used to put an `assert` and detect if an error occurred.

In this example, if the linked list had 9 elements, after we called the method, the list should have still 9 elements (if the argument was invalid for example; in Object-Oriented Programming, object's methods must check inputs to protect the object integrity, see encapsulation) or more element, otherwise, something goes wrong[23].

7. QA - Quality Assurance.

7.1. Bug hunting.

A first round of bug hunting has been organized before the beta of LibreOffice 4.3. As the latter is now out, a new bug hunting session will be held before the first release candidate gets released. To get involved, just play with a pre-release builds you can install from here[24]. “If you detects a bug, just report it on the bug tracker”[25][26].

You can also join our webchat, 15 contributors are able to help you report bugs, please join us NOW![64]

7.2. Profiling.

Matus published a spreadsheet of the evolution of the time it takes for LibreOffice to import/export some test documents over the time of development. As we interpret the graph, we see that we gained some time over the past few months then lost it again. Let’s fix that for 4.3.[44]

8. GSoC.

For all Google Summer of Code projects that have been accepted by Google, please read the Google Melange website[27].

The mid-term evaluation will be held on next week. (Cedric Bosdonnat)

(If we had to evaluate mentors too, Cedric was definitely ahead of schedule ;-) [36])

8.1. Improvements to the Template manager (Efe Gürkan YALAMAN)

Basically the template manager is moved to the start center. For more technical information, please read there [22].

8.2. Dialog Widget Conversion (Mihály Palenik and Szymon Kłos).

This project is basically converting the remaining .src files using fixed sizes and positions into the new .ui Glade format. Last week, some progress has been made on this subject: `RID_SANE_DIALOG`, `GRID_DIALOG`, `RID_DLG_MAPPING`, `RID_FORMULADLG_FORMULA` and `RID_SVXDLG_NEWHYPERLINK` are now converted[28].

8.3. Improved Color Selection (Krisztian Pinter).

The work on this topic has just started a week ago. The first step consists of having the colors of the currently opened document available as a color palette.

Unlike what was written in the task description, the methods from `SvxColorDockingWindow`[30] are not called when clicking the color toolbar. Instead the `SvxColorToolBoxControl`[31] and `SvxLineColorToolBoxControl` methods are called when the color toolbar is used in Writer[29].

The code of the color floating window `SvxColorWindow_Impl` can be found in the `tbcontrl.cxx` file. It sets the palette in the ctor, and also receives palette changes via UNO using `.uno:ColorTableState/SID_COLOR_TABLE`. Most of the sidebar color pickers (except the ones from the “Character” panel) are actually using a different code that can be found at `svx/source/sidebar/tools/ColorControl.cxx`.

8.4. Calc / Impress tiled rendering support (Andrzej Hunt).

We talked about this subject on the last LOWN edition. If you still don’t know what tiled rendering is about, please read LOWN 1. The tiled rendering tool is called `LibreOfficeKit`. This is a GTK widget which call `liblibreoffice` to parse documents. This widget could be used to preview all LibreOffice supported files in other applications like `Evince` or `Okular`. The latter aims to become an all-in-one previewer like `Aperçu` on OS X. `Okular` could drop its “not so well working” code and rely on a well-tried code for the future: `LibreOfficeKit`[34][39].

8.5. Refactor god objects (Valentin).

Weekly report → refactored `IDocumentTimerAccess` and `IDocumentLinksAdministrator` out of `SwDoc` and pushed to master.

8.6. Improve Usability of Personas (Rachit Gupta).

What has been achieved this week[38]:

- The complete theme is downloaded in a separate folder, allowing multiple themes to be stored in the system.
- As promised last week, the header and footer files of the selected theme are now downloaded in a separate thread. The UI is now no more blocked.
- Fixed many bugs introduced by the code.
- Added error messages in case anything goes wrong (no Internet or no results)

What will be on focus for the coming weeks will be:

- Terminating the transfer if the user clicks on Cancel while the theme is downloading.
- Improving the performance so the search results are shown quickly.

8.7. Enhancing text frames in Draw (Matteo Campanelli).

“Last week, the focus has been on exploring the problems related with finding the source of the issue for the text background color not working in non-edit mode. Currently the problem has been identified in the `drawinglayer` module[41].

The main task of this week is closing the subgoals in the thread cited above, thus getting solid text background working also in non-edit mode.”[42]

8.8. Connection to SharePoint and Microsoft OneDrive (Mihai Varga).

Last week, Michai has worked on OneDrive folder and file representation. He is now working on the implementation.[43]

9. Hackfests and conferences.

9.1. Paris Hackfest.

The Paris hackfest (Montreuil) will be hold on June 27-28th (thanks again Charles-H Schultz and Sophie Gautier), please add yourself to the wiki if you want to participate[57].

9.2. Toulouse Hackfest.

As Toulouse has been confirmed to hold a hackfest on November 15-16th (thanks again Arnaud Versini), as for all hackfests, attenders are asked to add themselves on the wiki[65].

9.3. LibO Conf.

“The call for the next LibreOffice Conference location for 2015 will be closed by the end of the month, don’t miss it, send your proposal now!” (Sophie Gautier) [35]

9.4. Linux.conf.au.

In 2015 linux.conf.au will be held 12-16 January in Auckland, New Zealand, at the University of Auckland Business School.

Last week, we said LibreOffice will not have a dedicated booth during the conference itself, but only one on the open day[53]. If you can attend and want to, by your relations and influence, make LibreOffice presence shine for this event, please show yourself now. The call for Proposals opened on 9 June 2014 and will close on 13 July 2014[52].

10. UX / Design team.

The UX chat have been quite unproductive lately, seeing the number of contributors decreasing over time: weekly design meetings became 14 days meeting, and now monthly. According to the current few contributors, the lack of motivation might have a huge factor and is the main reason why contributors are leaving out the design ship. This lack of interest has led to the need of a massive reorganisation of the design team.

For a common contributor, the purpose of a design team should be: be at the service of the whole community when the latter needs a new design for the website and online developers instances (gerrit, wiki,...), event banners for the website and social networks, and be be the leader on the UX side. UI implementations made by developers should be approved by the UX team first. All the aforementioned tasks are the usual work of a design team, at least in a corporate environment.

In the LibreOffice community and, a fortiori, in the Document Foundation, the design team was not in charge of the current website redesign, nor the developer web tool instances. The website is currently managed by 2 contracted companies: a French one for the website specification and an Italian for the design (maybe we do not have enough web skilled profiles into the design team, this should explain this decision). The web instances are managed by the infrastructure team. The UX is not playing a leading role[61], the UI will be designed most of the time according to what the developer coding the UI at that time has in head. When a mockup is designed, it is, broadly speaking, quite well respected[58]. Currently the design team is mostly focusing on user interface tests and design.

When the design team has a simple wish to implement, most of the time, she got no positive answer[59] or no answer at all from the developer team. Such wishes are often let open or marked as WONTFIX until someone fix it by himself. Some bugs are considered as easy hacks[60]. Easy hacks are quite easy starting points for junior developers joining the project.

In order for the design team to have a more important role, a reorganisation is vital. Mirek, one of the yet active design contributor, proposed the following organisation in a dictatorial way, the committee solution does not work: "integration of software cannot be achieved by committee, where everyone has to put in their own additions (featuritis again). It must be controlled by dictatorial artists with full say on the final cut."

The proposed solution is having the following leaders:

- UX lead
- Visual design lead
- User testing lead

For more description about each task, read the main thread[63]. If you are interested, please apply!

The design team could also be divided into 2 main parts: one for the UX and another for the marketing/banner/website part[62].

11. Users.

This week we saw a rather quite large discussion on the user mailing list regarding the ability to launch a presentation as an executable. The user asking this question wanted to make a presentation on the school computers which do not allow any software installation.

The solution we had was to use a portable LibreOffice version[49] which does not require any installation at all, and which could be easily brought along on a USB thumb drive.

During that discussion which digressed to ODF viewers, since ODF fat-client[46] viewers[47] are quite outdated (2006!)[48], Florian Reisinger came with an idea of a web based viewer (with editor capabilities) : WebODF. Using such online and self-hosted solution, since it is a tiny Javascript file, would be great in this use case: no installation required, no need to handle administration policies, you just need web browser.

Even more interesting in comparison to fat-clients, if your presentation has videos in it, just use a container like Webm or MP4 with supported codec for your videos. These are supported nowadays by all recent browsers. No need to check if your platform will be able to play your video, depending of the backend used (do you remember that bug with Gstreamer on Linux?[50, 51]).

References

- [1] <https://github.com/wget/lown>
- [2] <http://blog.documentfoundation.org/2014/06/20/libreoffice-4-2-5-hits-the-marketplace/>
- [3] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061919.html>
- [4] <http://listarchives.libreoffice.org/global/documentation/msg17353.html>
- [5] <http://listarchives.libreoffice.org/global/users/msg39426.html>
- [6] <http://listarchives.libreoffice.org/global/design/msg06643.html>
- [7] <http://listarchives.libreoffice.org/global/design/msg06646.html>
- [8] <https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Rhino>
- [9] <https://gerrit.libreoffice.org/#/c/9784/>
- [10] <https://gerrit.libreoffice.org/#/c/9785/>
- [11] https://wiki.openoffice.org/wiki/Uno/Article/About_Bridges
- [12] <https://gerrit.libreoffice.org/#/c/9696/>
- [13] <https://gerrit.libreoffice.org/#/c/9696/>
- [14] <https://gerrit.libreoffice.org/#/c/9779/>
- [15] <https://gerrit.libreoffice.org/#/c/9834/>
- [16] <https://gerrit.libreoffice.org/#/c/9831/>
- [17] <https://gerrit.libreoffice.org/#/c/9832/>
- [18] <https://www.google-melange.com/gsoc/project/details/google/gsoc2014/sk94/5685265389584384>
- [19] <http://fortune.com/2014/06/13/the-one-asterisk-on-teslas-patent-giveaway/>
- [20] <http://listarchives.libreoffice.org/global/marketing/msg16993.html>
- [21] https://bugs.freedesktop.org/show_bug.cgi?id=39593#c18
- [22] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061809.html>
- [23] <http://flosmind.wordpress.com/2014/06/15/assertion-errors-not-only-for-devs/>
- [24] <http://www.libreoffice.org/download/pre-releases/>
- [25] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061857.html>
- [26] <http://listarchives.documentfoundation.org/www/announce/msg00192.html>
- [27] <https://www.google-melange.com/gsoc/org2/google/gsoc2014/libreoffice>
- [28] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061840.html>
- [29] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061841.html>

[30] <http://opengrok.libreoffice.org/xref/core/svx/source/tbxctrls/colrctrl.cxx>

[31] <http://opengrok.libreoffice.org/xref/core/svx/source/tbxctrls/tbcontrl.cxx#2315>

[32] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061877.html>

[33] <https://wiki.documentfoundation.org/Development/BuildingOnWindows>

[34] <http://www.ahunt.org/2014/06/libreofficekit-gtk-viewer-widget/>

[35] <http://listarchives.libreoffice.org/global/marketing/msg17001.html>

[36] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061875.html>

[37] https://bugs.freedesktop.org/show_bug.cgi?id=41883#c17

[38] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061854.html>

[39] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061853.html>

[40] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061852.html>

[41] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061806.html>

[42] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061849.html>

[43] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061848.html>

[44] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061890.html>

[45] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061898.html>

[46] https://en.wikipedia.org/wiki/Fat_client

[47] <http://opendocumentfellowship.com/odfviewer>

[48] <https://code.google.com/p/odfviewer/downloads/list>

[49] <https://www.libreoffice.org/download/portable-versions/>

[50] <http://lists.freedesktop.org/archives/libreoffice/2014-June/061593.html>

[51] <http://listarchives.libreoffice.org/global/users/msg39480.html>

[52] <http://listarchives.libreoffice.org/global/marketing/msg16979.html>

[53] <http://listarchives.libreoffice.org/global/marketing/msg16979.html>

[54] https://en.wikipedia.org/wiki/Universal_Network_Objects

[55] <http://www.firebreath.org/display/documentation/Browser+Plugins+in+a+post-NPAPI+world>

[56] <https://wiki.documentfoundation.org/Hackfest/Paris2014>

[57] <https://wiki.documentfoundation.org/Hackfest/Toulouse2014>

[58] https://wiki.documentfoundation.org/Design/Whiteboards/Start_Center#Mockups

[59] https://bugs.freedesktop.org/show_bug.cgi?id=80054

- [60] https://wiki.documentfoundation.org/Development/Easy_Hacks#Welcome
- [61] <http://nabble.documentfoundation.org/Welcome-graphics-for-the-Startcenter-in-4-2-td4077497i20.html#a4079836>
- [62] <http://listarchives.libreoffice.org/global/design/msg06660.html>
- [63] <http://listarchives.libreoffice.org/global/design/msg06657.html>
- [64] <http://webchat.freenode.net/?channels=libreoffice-qa>
- [65] <https://wiki.documentfoundation.org/Hackfest/Toulouse2014>