

Improving Python Support in LibreOffice

Python support in LibreOffice and OpenOffice is quite old, and goes back to the early OpenOffice 1.x days with Python 2¹. Although this long history of Python support is something to be proud of, it was a project that was maintained by one person in his spare time since the start. With the extensive usage of Python in scripts and other applications, this approach is no longer an option.

Microsoft has recently announced Python support for Excel as a big feature, although the Python support was available in LibreOffice for many years. Also, there is now a separate effort to provide better Python support in LibreOffice:

LibrePythonista

LibrePythonista is an extension for LibreOffice Calc. The extension allows Interactive Python code to be run directly in a spreadsheet.

<https://extensions.libreoffice.org/en/extensions/show/99231>

The project is interesting in itself, but it is hard to install and use, and to provide a better experience for the users, it is important to provide such a feature in LibreOffice core.

This proposal is written to address the above needs. The goal is to hire 1 in-house developer to focus on the development of LibreOffice Python support.

TDF has successfully hired developers in the areas of a11y, RTL/CTL/CJK, which both lead to significant improvements in these areas that were neglected by other developers during several years. Building on this background, and the focus areas of the suggested hiring, it is expected that the staff can handle the hiring in short term, and help the hired developers to start in relatively short time.

Focus Areas

Various aspects of LibreOffice Python support need attention and improvement. In the next sections, a brief overview of these aspects is discussed. Fixing bugs is a priority, but as there are also some important features that need to be implemented.

The required programming skills for the developer working in this area will be C++ and Python.

¹ <https://www.openoffice.org/udk/python/python-bridge.html>

Security

While running Python scripts inside LibreOffice as a script, and connecting to LibreOffice via UNO bridge is possible, the whole process is brittle in case of errors and exceptions, and should be improved with the security in mind.

Interactive Python Support

This feature can provide interesting possibilities inside LibreOffice. Specifically, in LibreOffice Excel and Writer, interactive Python shell with the support of generating images and graphs can be helpful for data science and AI applications.

MS Excel Compatibility

MS Excel compatibility is among the important feature requests. As discussed, Microsoft has recently added support for interactive Python in LibreOffice². The goal is to provide compatibility with this feature from MS Excel to be able to import, process and export such feature in .xlsx files.

Fixing open bugs

There are various bugs related to Python, but unfortunately there is not a good categorization for such bugs. Searching for Python in the title of the issues lead to ~35 open bug reports.

Internal IDE

The internal LibreOffice IDE for scripting does not provide support for editing and debugging Python scripts. That is a limiting factor for the script developers to be able to use Python inside LibreOffice. The goal is to improve the internal IDE to support editing, running and debugging Python in LibreOffice.

While APSO provide such functionality, it is very limited, and lacks many required features.

APSO - Alternative Script Organizer for Python

<https://extensions.libreoffice.org/en/extensions/show/apso-alternative-script-organizer-for-python>

Code Completion

There is an ongoing effort to provide Python code completion for external IDEs via a GSoC project, but that feature will need improvements and maintenance in the future.

² <https://support.microsoft.com/en-us/office/get-started-with-python-in-excel-a33fbcbe-065b-41d3-82cf-23d05397f53d>

Conclusion

The goal is to hire 1 in house developers to improve LibreOffice Python support. The Executive Director, or other staff member appointed by the ED, shall direct management for the developers to ensure they effectively focus on their development areas, will follow the progress and address eventual issues.

The development work of the in-house developers will include fixing bugs and implementing features in full, fixing bugs that are blockers for community contributors and to help evaluating which complex tasks should be tackled by external specialists.

Regular weekly exchanges between the team and ESC will help the team to independently define the tasks to be performed so that the most relevant topics will be prioritized and overlaps will be avoided.

While the main areas of relevance are listed in this document, there may be situations where the in-house developers will work with the team to help fixing urgent issues, such as implementing security patches or app stores required changes, if skill sets and experience allow.

The in-house developers will have the same rules, rights and conditions as any other volunteer or corporate contributors. Overlaps or prioritization that find no clear conclusion within team will be evaluated by our Executive Director with the support of our mentors.

In-house developers will follow agile development methodologies, including short-term goals and delivery times.

In case the developers need to spend time mentoring community developers, the expectations may be adjusted by the Executive Director with the assistance of our current mentors.

In-house developers should be able to provide performance levels comparable to other developers working in the same area after an on-boarding and mentoring period of approximately 6 months.

In-house developers will greatly benefit from the experience accumulated by the team while also providing our current mentors a way to improve their mentoring techniques within the team to better serve other developers. The new members of the team may also provide mentoring services, once deemed ready by our current senior mentors, which will be scheduled in a way that will be compatible with their primary role as in-house developers.

In-house developers should be able to present their activities, discuss their achievements and possible obstacles on a regular basis in team and/or ESC meetings depending on the relevance of the topics to be discussed.

It is expected from the developers to set together with the team the planning of their activities and to participate to weekly team meetings.

TDF's staff will evaluate and report to our ED any attempt made by community members and/or third parties to propose tasks for our in-house developers that are distracting them from the work assigned to

them by the ED. If ED suspects the request is driven by a personal or commercial interest, the ED will consult the Board to decide about eventual further actions.