

**LibreOffice**  
The Document Foundation

**LibreOffice Technology**  
the unique FOSS platform  
for personal productivity

Italo Vignoli



# LibreOffice Technology

- LibreOffice Technology is the result of 10 years of development in the right direction
  - Same processing engine common to all modules
  - Based on a cleaned and refactored source code
  - With a focus on code quality and consistency
  - Supported by easy and extensive APIs
- The best open source platform for personal productivity



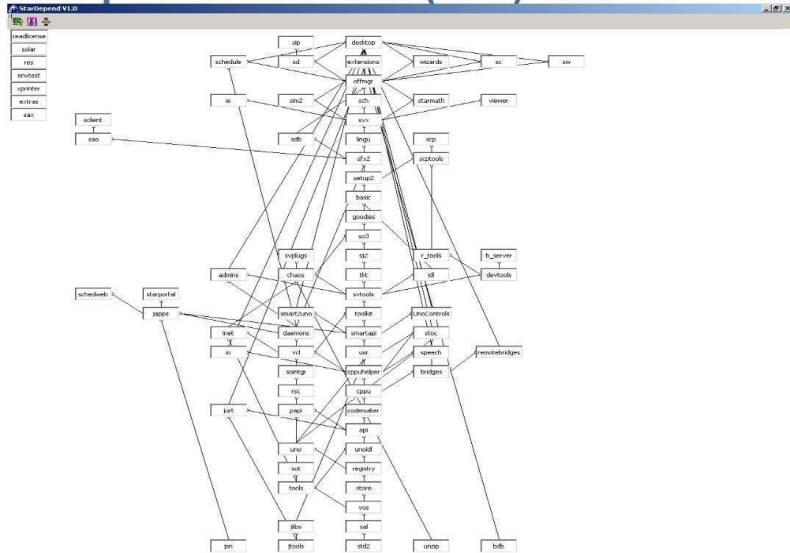
# Timeline

- 1985: Marco Borries releases StarWriter
- 1990: StarOffice, with StarCalc & StarBase
- 2000: announcement of OpenOffice.org
- 2006: ODF (Open Document Format) 1.0
- 2010: announcement of LibreOffice
- 2014: ODF (Open Document Format) 1.2
- 2020: LibreOffice 7.0 with ODF 1.3 Draft
- 2021: LibreOffice 7.1 Community

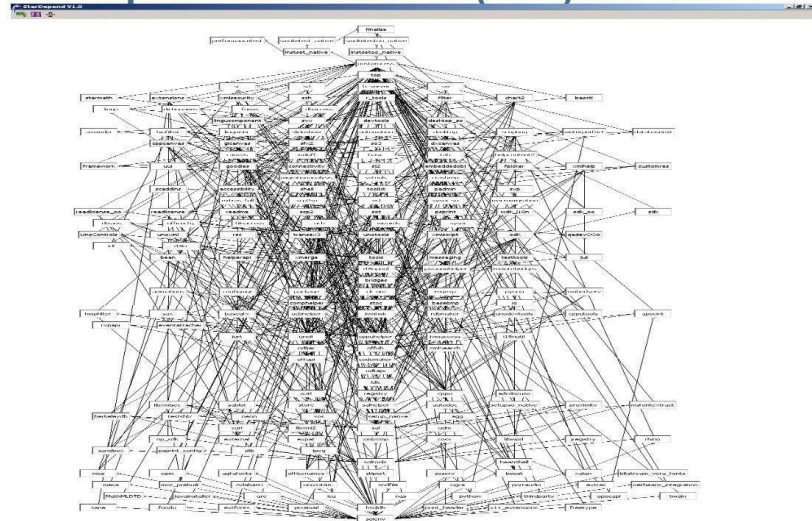


# Measure of OOo Technical Debt

Build dependencies SO 5.2 (2000)



Build dependencies OOo 2.0 (2005)



# Main Issues

- Code too complex for the average developer
- Build environment too difficult to create and manage
- Slow (and sometime obfuscated) handling of patches from volunteer contributors
- Mentoring of new volunteer developers to reduce the learning curve and speed up the process
- Lack of transparency for project sites, wiki and blog



# Relaunch the Innovation

## *OpenOffice to LibreOffice*

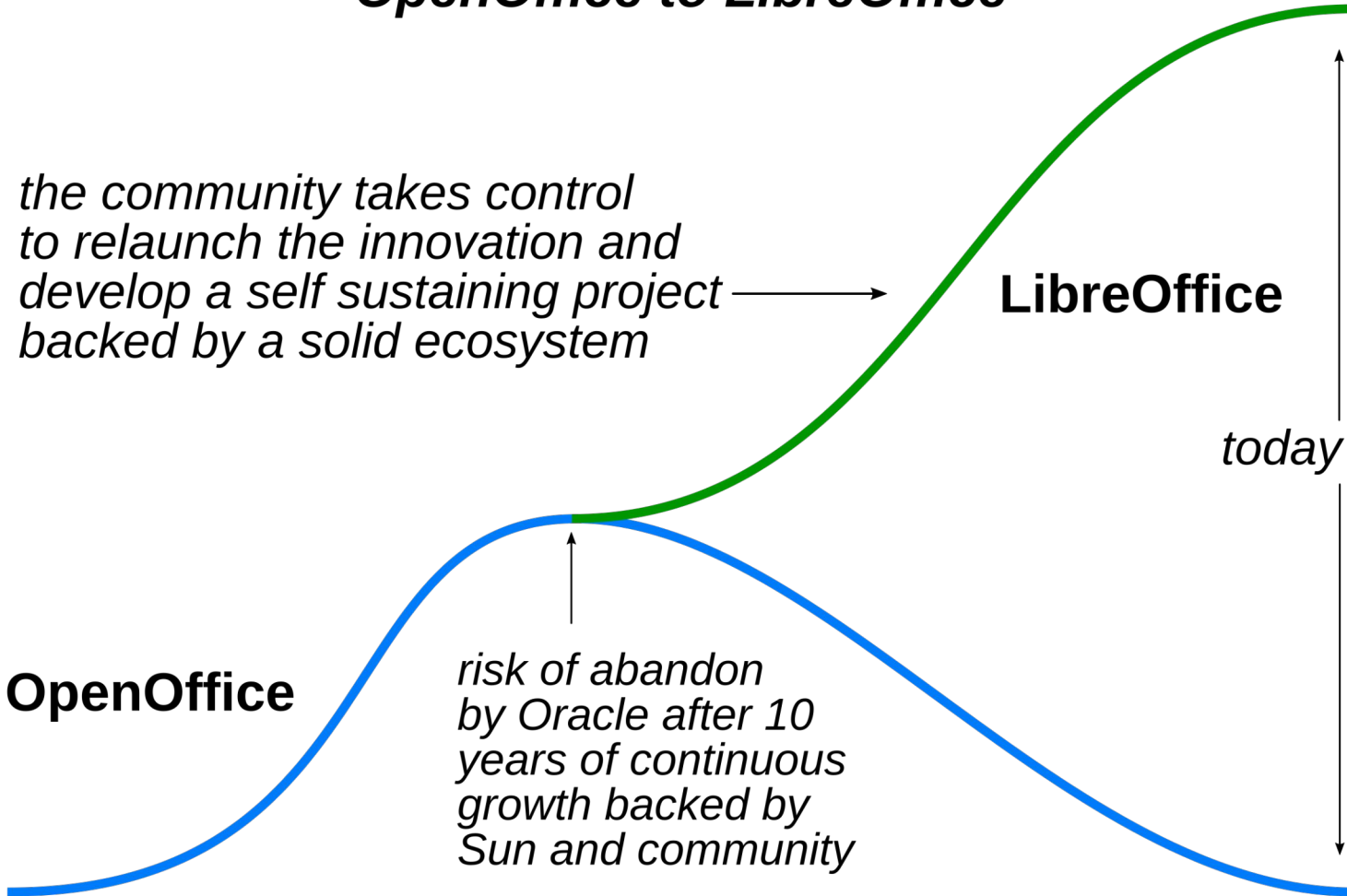
*the community takes control  
to relaunch the innovation and  
develop a self sustaining project  
backed by a solid ecosystem*

**LibreOffice**

*today*

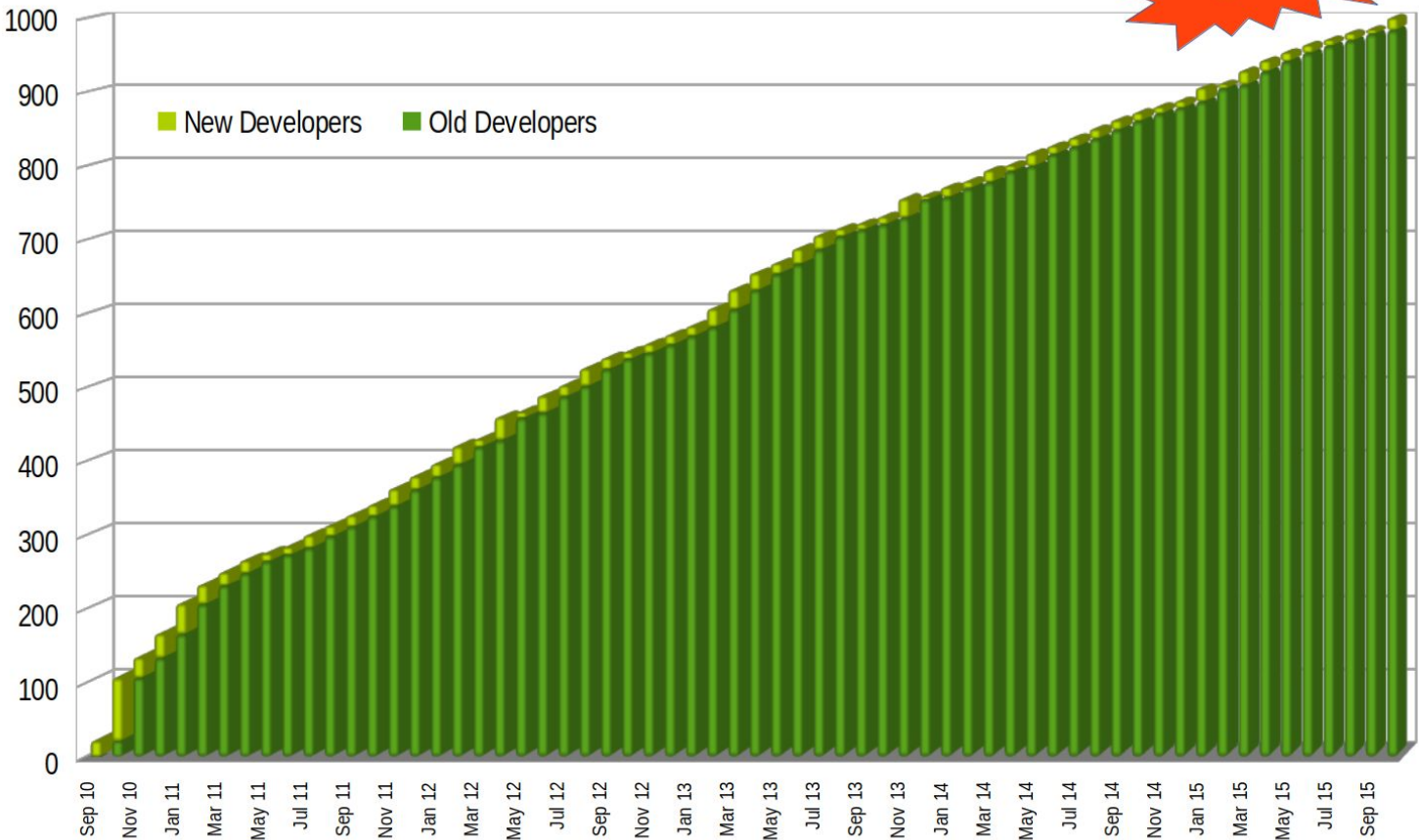
**OpenOffice**

*risk of abandon  
by Oracle after 10  
years of continuous  
growth backed by  
Sun and community*



# Serial Growth of LibreOffice Developers from September 2010 to October 2015

1,000



# Summary

- Balanced the mix of contributions
- Attracted many new code contributors
- Increased the number of core developers
- Created a strong base of new developers thanks to a huge mentoring effort
- Attracted several new "star" contributors performing challenging cleaning tasks





# Objectives

- Reduce software footprint
- Undertake long awaited code renovation
- Remove tens of thousands lines of dead code
- Remove deprecated libraries
- Translate majority of German comments to English
- Many other code renovation actions left and right
- Use state of the art C++ constructs

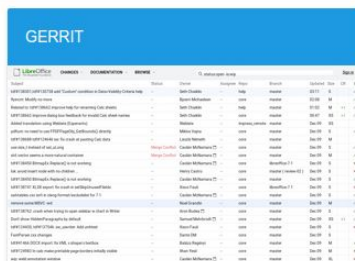


# Engineering Steering Committee ...

- **Talking & ensure everyone is heard**
- **Many large FLOSS projects fight tooth & nail**
  - Module "maintainers" rampant waste & re-writing
  - Bickering, and passive aggression
- **Whole team are peers ... no-one "owns" modules**
  - Respect input from local experts: of course !
- **Calls live transcribed to reduce English listening needs**
  - Avoids the "confused IRC wait-fest" problem



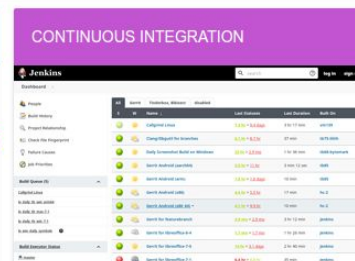
# Websites related to LibreOffice Development



Gerrit is the tool LibreOffice developers use for code review. This is where developers submit and review patches.



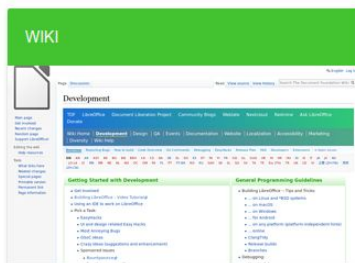
TinderBox present the result of the many, many tinderboxes that build Libreoffice all day and all night.



LibreOffice has a Continuous Integration system, running gerrit verification jobs, tinderboxes and all sort of other task.



Bugzilla is the tool for LibreOffice developers, QA and users to submit, triage, handles bug reports.



LibreOffice has a Wiki where you can find a trove of information. It is a wiki, so help organize and keep the content up-to-date.



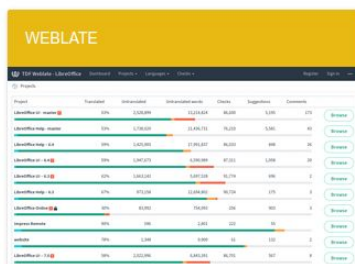
OpenGrok is a source browser, that allows you to do search on the very large LibreOffice code base.



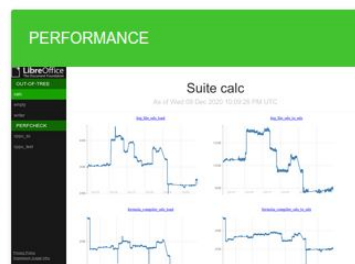
The Online help for LibreOffice contains the end user documentation for Writer, Calc, Draw, Base, Math, and Basic.



You will find the documentation and examples of use of the API for Universal Network Objects (UNO).



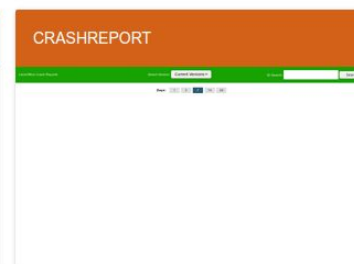
LibreOffice operate a Weblate server with which we manage the translation of the product in many languages.



We run a set of Callgrind Performance tests. the result of which are presented graphically on there.

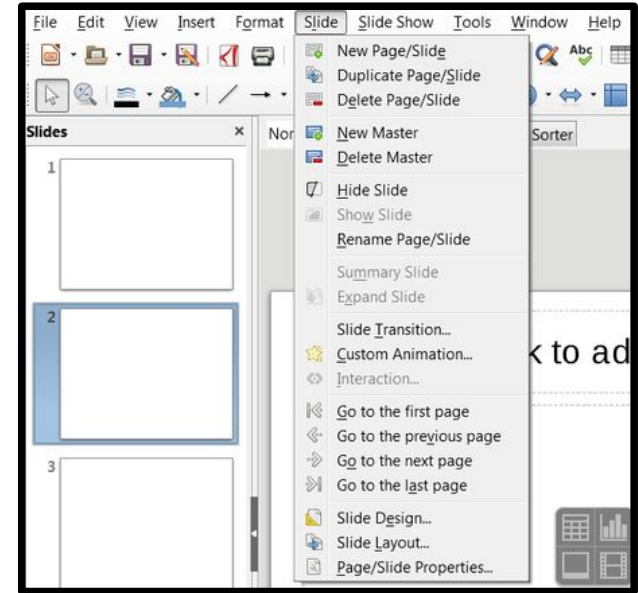
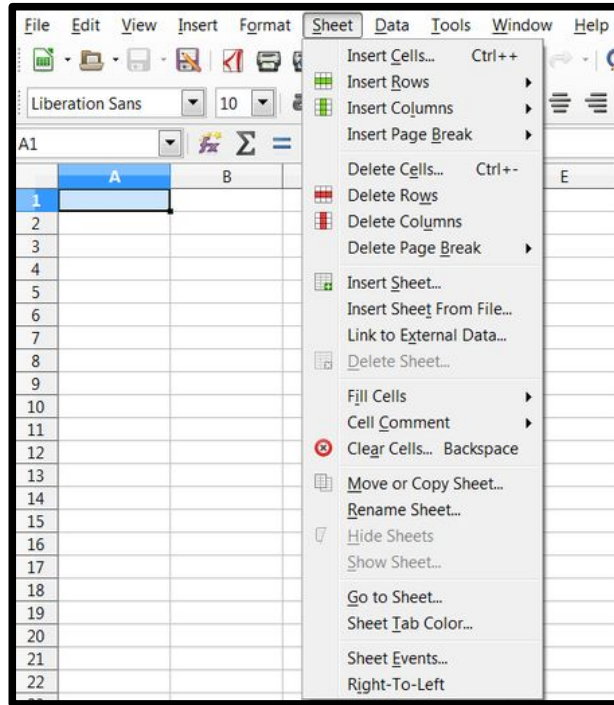
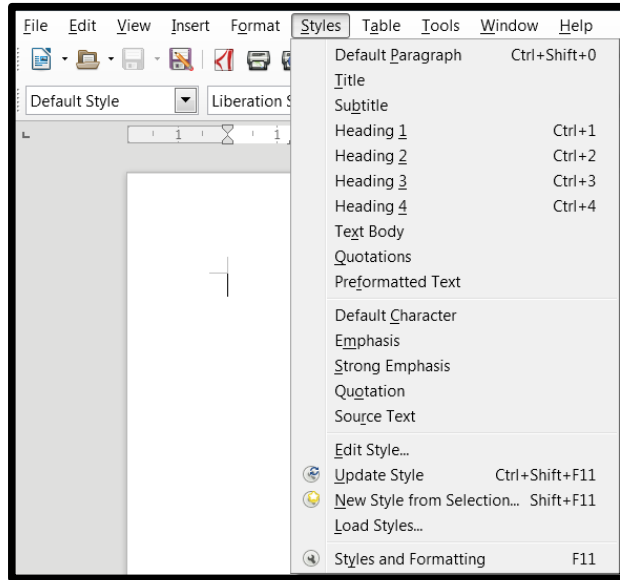


We run a cppcheck regularly.



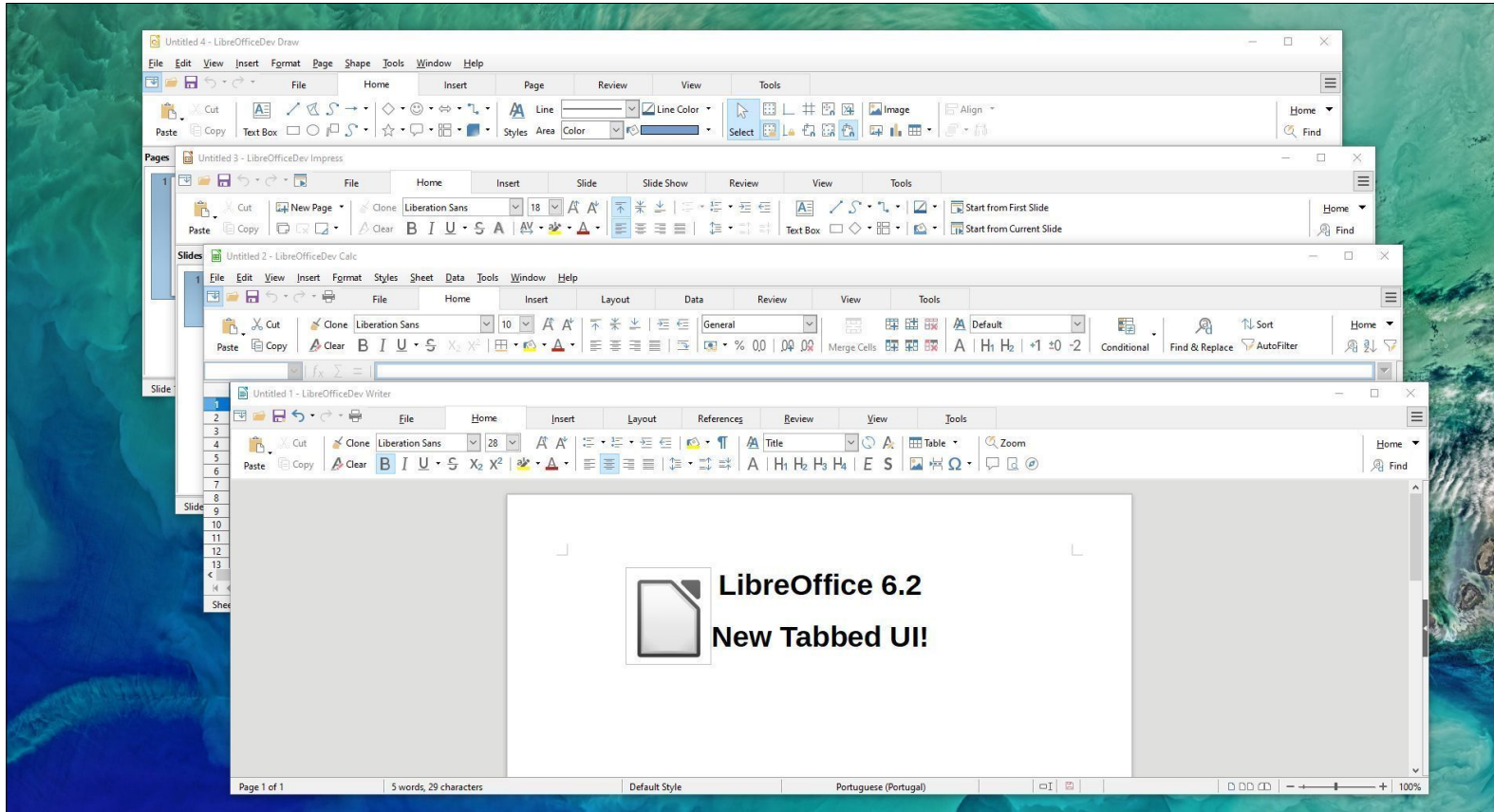
We collect crash reports.

# Reorganization of Menus

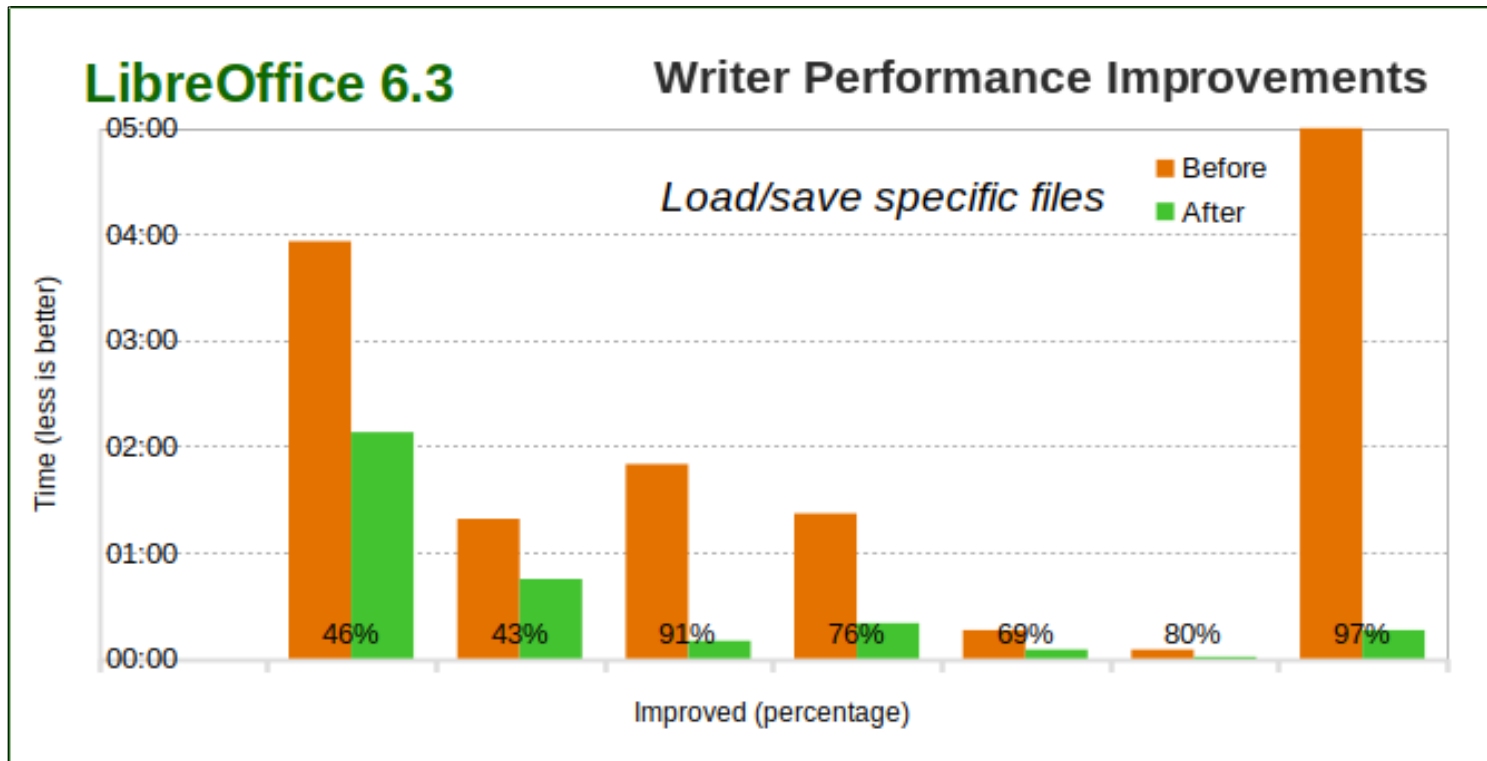


Reorganization of Writer, Calc and Impress menus, including the addition of a Styles menu (Writer), a Sheet menu (Calc) and a Slide menu (Impress)

# NotebookBar



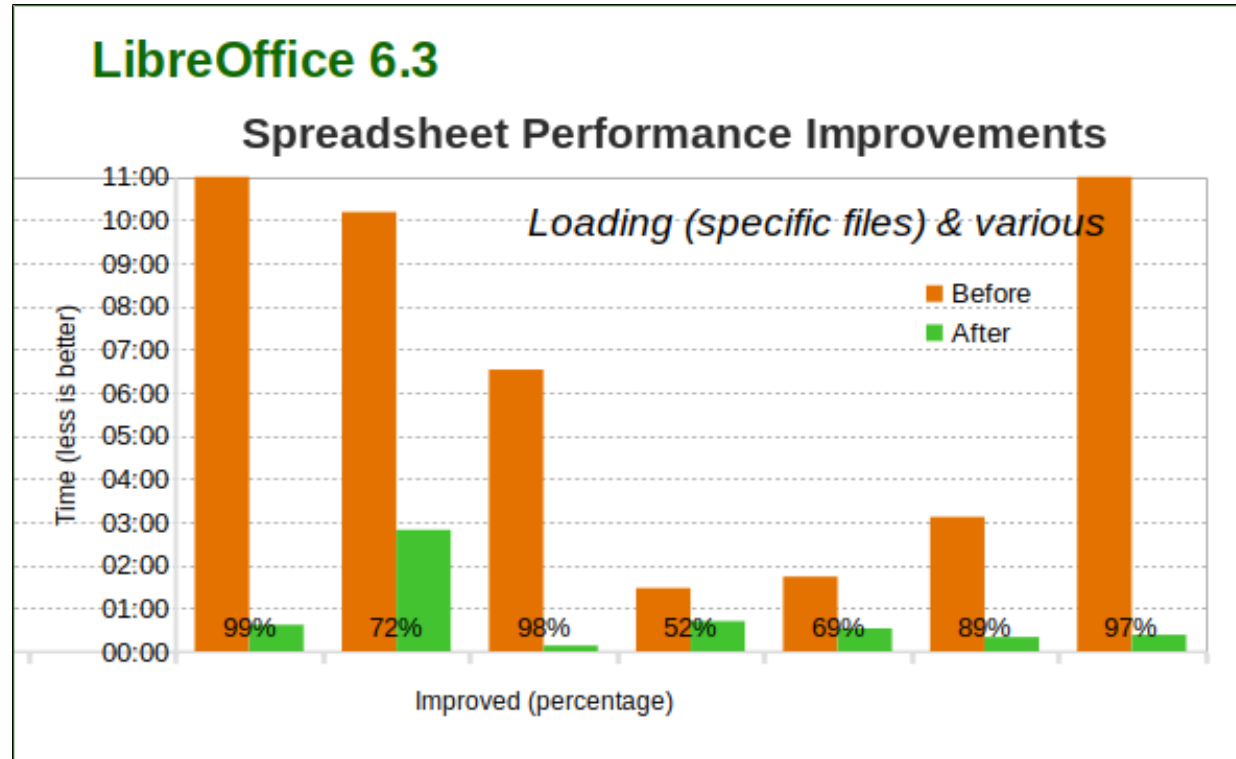
# Writer 6.3



Significant Open/Save performance improvements



# Calc 6.3



Significant Open/Save performance improvements



# Static Code Analysis

- Tool: *Coverity Scan*
  - Free for open source projects
- Detects dead code, uninitialized variables, uncaught exceptions...
- Defect density reduced from 1.1 to ~0,00x
  - Density measured in defects every 1.000 lines
  - Average density for similar sized projects: 0.71





# Coverity Scan Score

Dec 06, 2020

Last Analyzed

6,143,004

Lines of Code Analyzed

0.00

Defect Density

**Defect changes since previous build dated Dec 05, 2020**

0

Newly detected

1

Eliminated

**Defects by status for current build**

26,283

Total defects

0

Outstanding

356

Dismissed

25,927

Fixed

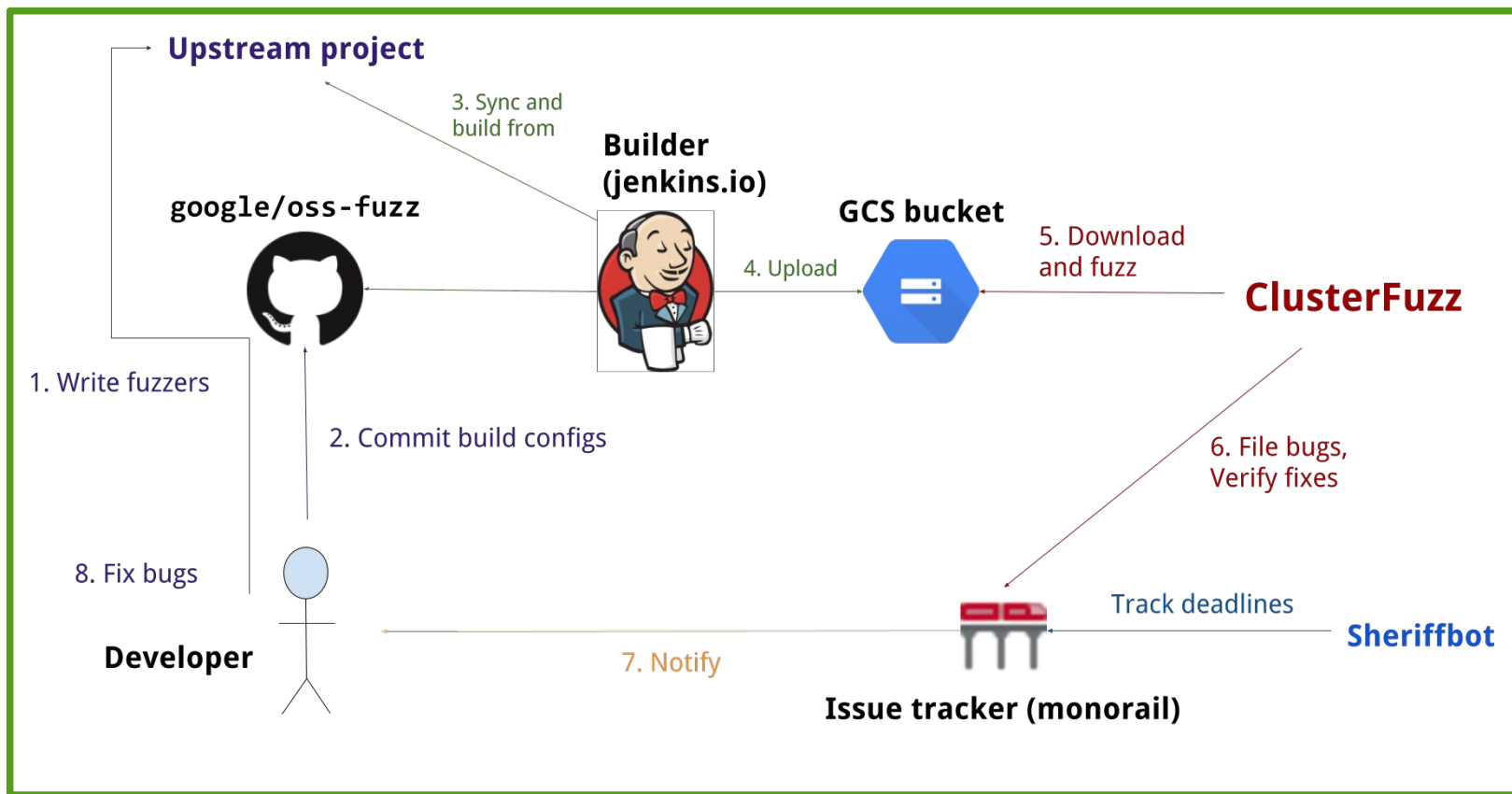


# Fuzzing

- Feeding a program random data in order to induce faults
- Black box fuzzing assumes nothing about the expectations of the program
- White box fuzzing knows about the underlying formats and protocols



# Google's OSS-Fuzz





# Open Document Format

the true document standard  
which offers freedom of choice

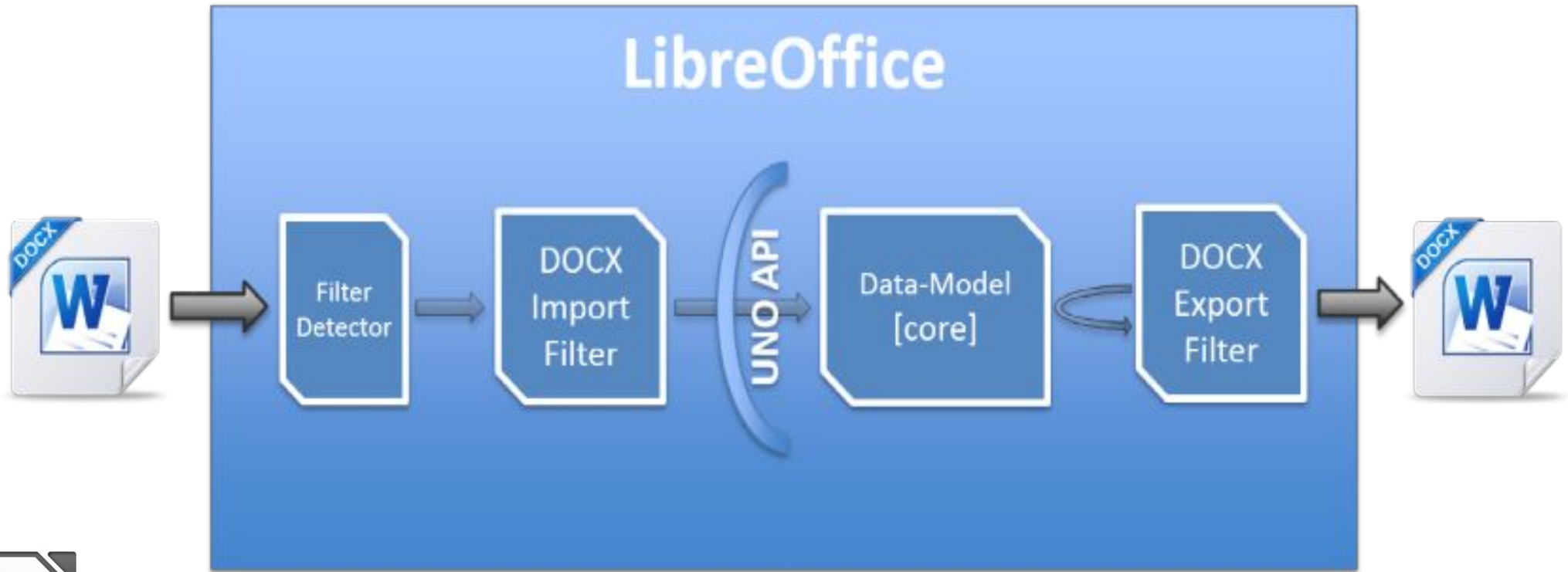


# ODF Advantages

- ODF is solid and robust
- ODF is consistent across OS
- ODF is truly interoperable
- ODF is predictable
- **ODF is the best standard file format for users of personal productivity SW**



# Roundtrip Test



# LibreOfficeKit

- LibreOfficeKit can be used for accessing LibreOffice functionality through C/C++, without any need to use UNO
- The essence of the idea is to provide a thin API for LibreOffice, that allows people to write small, external apps without having to compile or link to LibreOffice, and learn UNO or other complex topics
- Allows fast rendering of documents for any application



# LibreOffice: Product to Platform

- 2010: LibreOffice only for the Desktop
- 2020: LibreOffice for the Desktop, LibreOffice LTS optimized for Enterprises, LibreOffice Online for the Cloud, LibreOffice Mobile for Android & iOS, and LibreOffice for Chrome OS
- All these products, from different organizations, share the same engine, which is common to all modules





# Who Pays LO Development?

- 68% of development is paid by customers of ecosystem companies selling a LibreOffice Enterprise product
  - The majority of this activity is focused on new features and improvements to interoperability
- 28% of development is provided by volunteers, and paid by their willingness to contribute to the project
  - The majority of this activity is focused on the UX (user interface and usability), with notable exceptions



# Donations & Contributions

- 90% of donations to The Document Foundation are from individuals and 10% from SMBs
- Based on estimates, less than 5% of all LibreOffice enterprise users – including governments – contribute in any way to the project, including buying any kind of product/service from ecosystem companies
- This represents an issue for the sustainability of the project's model



# Educate Free Riders

## *LibreOffice to LibreOffice Rolling*

*with a growing number of enterprises using the product without providing anything back and not paying for LTS versions the sustainability is at stake and we have to educate free riders about contributing to FOSS*

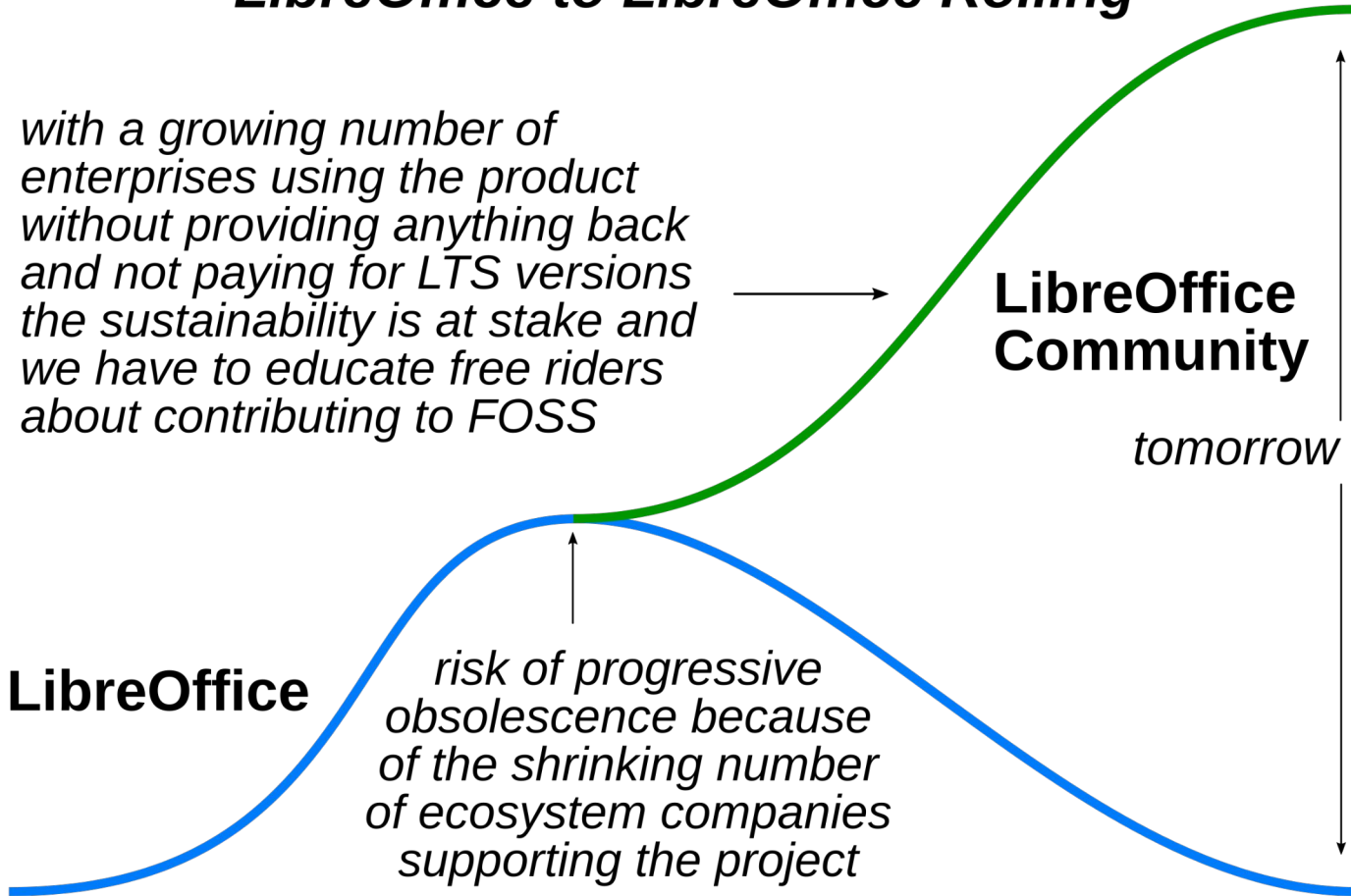


**LibreOffice  
Community**

*tomorrow*

**LibreOffice**

*risk of progressive obsolescence because of the shrinking number of ecosystem companies supporting the project*



# Thanks

Italo Vignoli  
The Document Foundation  
[italo@libreoffice.org](mailto:italo@libreoffice.org)  
[italo@vignoli.org](mailto:italo@vignoli.org)

