

The Document Foundation Planet - Latest News

- [Michael Meeks: 2025-12-05 Friday](#) (2025/12/05 13:58)

Mail chew, worked through a growing admin backlog. Great TTT from Kendy on Collabora Office on Mac. Published the next strip on trying to please everyone:

- [Official TDF Blog: LibreOffice Conference 2026 and 2027](#) (2025/12/05 11:20)

LibreOffice Conference 2026 in Pordenone, Italy, and LibreOffice Conference 2027 in Gothenburg, Sweden LibreOffice Conference 2026 will be organised by PNLug (Pordenone Linux User Group) and the University of Pordenone in the city of Pordenone, on the university campus. The conference will begin on Thursday, 10 September and will end on the morning of Saturday, 12 September with the usual closing session. In the days leading up to the conference, there will be a series of internal meetings and the usual community meeting on the afternoon of Wednesday 9 September. Of course, all the details of the conference, the call for papers, and the final programme with all the side events will be announced during 2026 on this blog and on the conference website. LibreOffice Conference 2027 will be organised in Gothenburg, Sweden, by the local community led by Leif-Jöran Olsson, who will be involved in the organisation of the 2026 conference to familiarise himself with the process. Again, all details will be announced on this blog and on the conference website starting in the last quarter of 2026.

- [LibreOffice Dev Blog: Handling CI build failures](#) (2025/12/04 23:10)

After submitting a patch to LibreOffice Gerrit, one has to wait for the continuous integration (CI) to build and test the changed source code to make sure that the build is OK and the tests pass successfully. Here we discuss the situation when one or more CI builds fail, and how to handle that. Why Build and Test on CI? After you submit code to LibreOffice Gerrit, reviewers have to make sure that it builds, and the tests pass with the new source code. But, it is not possible for the reviewers to test the code on each and every platform that LibreOffice supports. Therefore, Jenkins CI does that job of building and testing LibreOffice on various platforms. This can take a while, usually 1 hour or so, but sometimes can take longer than that. If everything is OK, then your submission will get `Verified +1`. CI Platforms for LibreOffice Currently, these are the platforms used in CI: Linux / GCC: `gerrit_linux_gcc_release` Linux / Clang: `gerrit_linux_clang_dbgutil` Android Viewer: `gerrit_android_x86_64` and `gerrit_android_arm` Windows: `gerrit_windows_wsl` macOS: `gerrit_mac` Some of the tests are more extensive, for example Linux / Clang also performs additional code quality checks with clang compiler plugins. Also, UITests are not run on each and every platform. LibreOffice CI uses Jenkins Why Failures Happen and How to Fix? There can be multiple reasons for why a CI build fails, and give your submission `Verified -1`. These are some of the reasons, and depending on the reason, solution can be different. 1. Your code's syntax is wrong and compile fails In this case, you should fix your code, and then submit a new patch set. You have to wait again for a new CI build. 2. The code's syntax is OK, but it is not properly formatted You should refer to the below TDF Wiki article and use clang-format tool to format your code properly. TDF Wiki - clang-format 3. Your code's syntax is OK, but it logically not OK and fails some tests. In this case, you should try fixing your code logic, and run the tests that fail and make sure they pass. After that, you may send a new patch set and wait for a new CI build. 4. Your code's syntax and logic is OK, but some machine fails for other reasons like their disk being full or other software/hardware failures or hiccups In this case, usually resuming the build can be a good option. You may ask on `#libreoffice-dev` or `#tdf-infra` IRC rooms for such a resume, or request access, if you submit many

patches. Resume build in LibreOffice CI 5. Your code's syntax and logic is OK, but there are issues from other patches. In this case, intervention from other LibreOffice developers is needed. Informing people on #libreoffice-dev can help, and then you have to re-base your submission in case new patches fix the build issue. Final Notes The best way to know the reason of the build failure is to look into the CI log files. Sometimes it needs more detailed look to understand the issue, but sometimes the reason is easily provided on Gerrit as a comment. But, in the end your submission should have `Verified +1` before it is suitable for merge in the LibreOffice code. This +1 as verified, does not guarantee that your patch will work as expected, but it is an important requirement.

- [Michael Meeks: 2025-12-04 Thursday](#) (2025/12/04 21:00)

Mail chew, tech. planning call, new Collabora Office launch retrospective call. Lunch. Sync with Lily, Laser, partner slide sync, drove to Cambridge, good to catch-up with people, finance team call. Christmas dinner with lots of the wider Collabora team, good to meet Tina.

- [Marius Popa Adrian: PHP Firebird changes in 2025 : 5.0.2 up through 6.1.1-RC.2.](#) (2025/12/04 15:06)

I fetched the release notes for FirebirdSQL/php-firebird and made a concise summary of the user-visible changes and upgrade impact for versions from PHP Firebird 5.0.2 up through 6.1.1-RC.2. I retrieved the release entries for 5.0.2, 6.1.1-RC.0, 6.1.1-RC.1 and 6.1.1-RC.2 and distilled the highlights and upgrade impact into a short, actionable summary below. Summary of changes (5.0.2 → 6.1.1-RC.2)-

- [Official TDF Blog: Welcome Dan Williams, new LibreOffice Developer focusing on UI/UX](#) (2025/12/04 13:40)

The Document Foundation is the small non-profit entity behind LibreOffice. It oversees the project and community, and is now expanding with new developer roles. So let's say hello to Dan Williams, who joins the team to work on design and user interface (UI) improvements, with an initial focus on macOS: Tell us a bit about yourself! I'm from the USA, have lived on both US coasts at various times, and now live back in the "midwest" where I grew up. I was previously a software engineer, team lead, and manager at Red Hat for more than 20 years. In that time I've worked on a large variety of projects, from local networking to cloud networking to desktop software. I spent two years helping build the One Laptop Per Child software stack which was an eye-opening experience from a UI and design perspective. I believe passionately in free and open-source software; all the code I've written so far in my career is open-source. Oddly enough, I'm not new to the LibreOffice community; I was an OpenOffice contributor and co-founded the NeoOffice port to Mac OS X (now called macOS). That led to being hired by Red Hat to package and improve OpenOffice for Fedora and Red Hat Enterprise Linux, where I helped start the 64-bit port and realize the Native Widget Framework that's still used in LibreOffice today. I eventually handed my Red Hat OpenOffice duties over to Caolán McNamara (now at Collabora) and moved on to networking. I still recognize quite a few of our community members who I worked with before! Outside of office software I'm the maintainer of the ModemManager WWAN software stack and I seem to have developed a hobby of collecting LTE and 5G base stations. I enjoy playing with hardware (especially if it runs Linux), baking and cooking, building large structures out of wood, and occasionally brewing beer in my basement. What's your new role at TDF, and what will you be working on? A significant part of my new role will echo my early OpenOffice contributions: Mac! I'll convert the last bits of Carbon API into modern Cocoa ones. This old code involves some pretty core functionality like popup menus and key modifier detection so it's going to be tricky but we need to do it. Nobody knows how much longer Apple's going to support HIToolbox but I'd like to be prepared. I'm also going to improve general Mac usability and experience; I've been using LibreOffice on macOS since the beginning so I'm familiar with its rough edges. But the Mac isn't the only platform so I'll be driving user interface improvements and fixing annoying bugs everywhere else too, regardless of platform or visual backend. I also look forward to working with the LibreOffice Design community to try out new ideas and see what sticks. I enjoy the challenge of working throughout the entire codebase, from the depths of VCL/SAL up to the UI logic

and layout in the applications at the top. I'm also a big believer in automated testing and continuous integration (CI) and I hope to improve our reliability and reduce regressions. It's often a pain to write tests for a small bug fix or any other change, but I think they're necessary for a healthy project. How can all users of LibreOffice help out? What are the most annoying interface and interaction bugs, for the Mac or otherwise? Make sure they're in Bugzilla! I need help prioritizing issues – so here's your chance to get your favorite bug looked at. If you're interested in LibreOffice's user interface and visual identity, join the Design team. Do you have Cocoa experience on the Mac and a bit of extra time? Help me out with bugs! I'd be happy to get you started.

- [Marius Popa Adrian: The Firebird Book — A Gift for Everyone](#) (2025/12/04 11:46)

The Firebird Book, Second Edition is made publicly available via the IBPhoenix digital store. Everyone can download the complete edition free of charge — no strings attached. If you'd like to support Helen's legacy and the Firebird community, there is an optional pay-what-you-like contribution. Funds will support the organization of the Helen Borrie Memorial Award, recognizing individuals with

- [Michael Meeks: 2025-12-03 Wednesday](#) (2025/12/03 21:00)

Early partner call, sync with Dave, then Ben, then Patrick & Kevin. E's maths progress review call. Interview, weekly sales call, sync with Philippe. All Saints band practice in the evening.

- [Official TDF Blog: The Document Foundation announces the approval of the Open Document Format \(ODF\) v1.4 standard by OASIS Open](#) (2025/12/03 15:25)

This new version of the native LibreOffice document format standard marks the 20th anniversary of the only open document format for office applications Berlin, December 3, 2025 – The Document Foundation announces that OASIS Open (www.oasis-open.org), the global open-source and standards organisation, has approved the Open Document Format (ODF) for office applications v1.4 as an OASIS standard, which is the organisation's highest level of ratification. The release of ODF v1.4 coincides with the 20th anniversary of ODF's adoption as an OASIS Standard. Since 2005, ODF has served users as a vendor-neutral, royalty-free format for office documents, ensuring that files remain readable, editable and interoperable across platforms. Several governments and international organisations, including NATO, the European Commission and countries across multiple continents, have mandated ODF for their operations worldwide. ODF v1.4 maintains full backward compatibility and improves developer documentation, adds better support for assistive technologies for accessibility, improves professional document formatting and visual design capabilities, and expands features for data analysis and technical documentation. These updates reinforce the Open Document Format's position as a comprehensive solution for office productivity and document creation. "ODF provides a vendor-neutral foundation for office productivity and collaboration. With v1.4, the standard continues to evolve, supporting cloud collaboration, richer multimedia, and standardised security," said Svante Schubert, Open Document Format's TC Co-Chair. "Looking ahead, ODF is moving beyond document exchange towards standardised, semantic, change-based collaboration, enabling the meaningful sharing of interoperable changes across platforms." ODF v1.4 safeguards digital sovereignty by removing a single vendor's control over documents and returning it to the community, to allow individuals and enterprises to independently decide how and with whom to share content, preventing it from being analysed for commercial purposes or potentially shared without the legitimate owner's knowledge. Like all other versions of the standard format, ODF 1.4 is based on an XML schema that complies with simplicity and readability guidelines, making files much more robust and secure than those commonly found on the market. Overall, this is another significant step towards transparency, openness and digital sovereignty, thanks to the collaborative efforts of open-source software developers, advocates and users. The finalized four-part specification of ODF v1.4 can be found in the OASIS library by clicking here:

docs.oasis-open.org/office/OpenDocument/v1.4/os/. The announcement mentions OASIS sponsors who have not contributed to the development of ODF 1.4, whereas The Document Foundation, which funded the development of the standard alongside companies such as Microsoft and has always advocated for it, is not mentioned at all. Unfortunately, business is business, even when it comes to open standards. ODF 1.4 New Features General The writing direction specification has been expanded and clarified. Complex backgrounds, such as colour gradients or hatching, can now be applied to a wider range of objects. Objects can be marked as “decorative” to support accessibility technologies. Shapes can now contain not only simple text and lists, but also tables. The method for specifying the handles of shapes has been improved. A new, more flexible method has been added for specifying the format of number labels in multi-level lists. Text Documents It is now possible to position objects relative to the page margins. A binding area (also known as a gutter) can be set in the page format. The overlay behaviour of objects can now be specified more flexibly. Spreadsheets A new function, `EASTERSUNDAY`, can be used to calculate the date of Easter and its associated dates. Text colour and cell background colour can be used as filter criteria. Charts Scales can be labelled more flexibly. For logarithmic scales, the underlying base can be specified. Formula Typesetting All versions of the MathML formula language are now permitted.

- [LibreOffice QA Blog: LibreOffice 26.2 Alpha1 is available for testing](#) (2025/12/03 12:58)

LibreOffice 26.2 will be released as final at the beginning of February, 2026 (Check the Release Plan) being LibreOffice 26.2 Alpha1 the first pre-release since the development of version 26.2 started at the beginning of June, 2025. Since then, 4651 commits have been submitted to the code repository and 676 bugs were set to FIXED in Bugzilla. Check the release notes to find the new features included in this version of LibreOffice. LibreOffice 26.2 Alpha1 can be downloaded for Linux, macOS and Windows, and it can be installed alongside the standard version. In case you find any problem in this pre-release, please report it in Bugzilla (You just need a legit email account in order to create a new account). For help, you can contact the QA Team directly in the QA IRC channel or via Matrix. LibreOffice is a volunteer-driven community project, so please help us to test “ we appreciate it! Happy testing!! ...

- [Miklos Vajna: Markdown import in Writer: the new template option](#) (2025/12/03 08:35)

Writer recently got a new markdown import option to take styles from a template, leading to much prettier output when converting markdown to PDF, DOCX or ODT. This work is primarily for Collabora Online, but the templating feature is available in desktop Writer as well. Motivation¶ A previous post mentioned recent improvements to the markdown import/export in Writer. But if you convert some markdown to e.g. PDF, all the headings just have the default look, wouldn't it be nice to take your organization template and add colors and other formatting there, automatically? Also, wouldn't it be nice if you could paste as markdown in COOL or copy the current selection as markdown? Which would enable all sorts of interesting use-cases, like using an external API to turn the selection into a summary or translating it to a different language. Results so far¶ Here is a sample input markdown: # heading 1 body text Here is how it looks like if you template it using the core.git sw/qa/filter/md/data/template.docx sample: PDF result: templated curl invocation for this: `curl -k -F "data=@/path/to/test.md" -F "template=@/path/to/template.docx" -F "format=pdf" -o out.pdf https://localhost:9980/cool/convert-to` Or example desktop command-line: `soffice --infilter='Markdown:{"TemplateURL":{"type":"string","value":"./template.ott"}}' test.md` While it would look like this by default: PDF result: default The other part is the PostMessage API of COOL, if you want to copy and paste as markdown. What's newly possible: Copy the current selection: set MessageId to Action_Copy and the value to `{"Mimetype": "text/markdown;charset=utf-8"}` Paste at the current cursor position: set MessageId to Action_Paste and the value to something like `{"Mimetype": "text/markdown;charset=utf-8", "Data": "foo_bar_baz"}` You can read more about the PostMessage API in the COOL SDK. How is this implemented?¶ If you would like to know a bit more about how this works,

continue reading... :-) As usual, the high-level problem was addressed by a series of small changes. Core side: tdf#169316 sw markdown import: add a TemplateURL parameter tdf#169316 sw markdown import, template: handle non-ODF formats as well cool#13468 sw markdown paste: add UNO command parameter to skip the detection Related: tdf#169251 sw markdown export: fix crash on OLE with no graphic Online side: cool#13419 convert-to template option: handle multiple streams in ConvertToPartHandler cool#13419 convert-to template option: pass it to doc broker cool#13419 convert-to template option: pass it to the kit process cool#13419 convert-to template option: more strict param name, generalize filenames cool#13419 convert-to template option: add testcase cool#13468 PostMessage API: allow copying the current text selection Want to start using this?¶ You can get a development edition of Collabora Online 25.04 and try it out yourself right now: try the development edition. Collabora intends to continue supporting and contributing to LibreOffice, the code is merged so we expect the core of this work will be available in TDF's next release too (26.2).

- [Michael Meeks: 2025-12-02 Tuesday](#) (2025/12/02 21:00)

Up early, new Collabora Office team call - nice. Planning call, sync with Laser, lunch. Call with Jim, sync with Anna, chart team call, sync with Laser.

- [Michael Meeks: 2025-12-01 Monday](#) (2025/12/01 21:00)

Still feeling groggy; early meeting on a11y testing. Sync with Miklos, Thorsten, lunch, signed documents, marketing content call. Sync with Naomi, Pedro & Eloy. Dinner, games with E&H&J, music with H. bed.

- [Official TDF Blog: Code of Ethics and Fiduciary Duties for TDF's Board of Directors](#) (2025/12/01 17:52)

The Document Foundation (TDF) is the non-profit entity behind the LibreOffice project. It collects donations from users, and employs a small team to support and coordinate the worldwide community that makes the software. In TDF there are various bodies including the Board of Directors, Membership Committee, and the Board of Trustees: These foundation bodies are guided by a set of policies, and now the Board of Directors has voted on a Code of Ethics and Fiduciary Duties. The code ensures that members of the Board of Directors “strive to speak and act with the mission and effectiveness and the best interest, and only the interest, of the foundation in mind.” See here for the results of the vote. Click here for the full Code of Ethics and Fiduciary Duties

- [Official TDF Blog: The Role of ODF in Digital Identity and Authentication](#) (2025/11/28 10:23)

Reliable data flows, verifiable signatures and predictable structures are essential for digital identity systems, which touch every aspect of modern digital life. They authorise transactions, confirm requests and guarantee security policies. In this context, the Open Document Format (ODF) offers a transparent, computer-readable foundation for verifying the authenticity of documents and ensuring their long-term integrity. Each ODF file is a structured ZIP container with a consistent internal layout. It contains a set of XML files that are always located in the same position. These files include meta.xml for metadata, manifest.xml for the list of files and relationships, content.xml for document data and styles.xml for presentation rules. The files are either ODT (text), ODS (spreadsheets), ODP (presentations) or ODG (drawings). Because everything is in XML format and in the same location, identity systems can analyse the content without searching for it as they would with OOXML files, which vary greatly depending on the application used to create them. Identity systems can therefore focus on specific parts of a file rather than scanning raw binary blocks, which are present in OOXML files. This is important for signing, integrity validation, metadata extraction and policy enforcement. When documents move from one identity platform to another, APIs can map ODF elements in a stable manner, reducing ambiguity and improving verification speed. Document Signing ODF supports the XML Signature and XML Encryption standards via the META-

INF/documentsignatures.xml file. This file can contain multiple independent signatures, each relating to specific parts of the document. The signature refers to an explicit path within the ZIP container, making automatic verification easier and avoiding confusion caused by false errors resulting from layout changes. Each document can contain user signatures, organisational seals, timestamps, and workflow attestations. Each signature can also contain its own certificate chain, revocation information, and policies. ODF is compatible with standard X.509 certificates, enabling the use of national eIDAS identification systems and corporate PKI systems. Verification pipelines can apply the same trust rules used for signed emails or encrypted communications. Interoperability and Identity Federation Digital identity works best when it is portable. ODF's openness supports this by avoiding vendor-specific binary constructs. Any identity framework can be integrated with ODF because its schema is public and stable, its structure is predictable, and there are no proprietary validators. In federated identity ecosystems, such as cross-border government services or multi-cloud enterprise configurations, ODF reduces friction and ensures that documents remain compatible, even when authentication systems differ. Long-Term Signature Validation and Archival Use In some cases, identity systems must verify a document signed many years earlier, which requires long-term validation. ODF supports long-term authentication because its XML structure is future-proof: it can store timestamps, revocation data and certificate chains, and it avoids vendor-specific cryptographic formats. In legal, regulatory and archival contexts, this aspect is more important than speed. Formats that rely on proprietary rendering engines risk becoming unreadable over time, whereas ODF remains readable, even many years later. ODF in Zero Trust Workflows In Zero Trust architectures, every resource must be verified at every stage. The structure of ODF fits perfectly into this model. Automated systems can verify the following: certificate validity, signature integrity, metadata trust levels, and the consistency of internal component hashes As ODF exposes everything via XML, identity engines can apply consistent rules without performing custom analysis, thereby reducing attack surfaces and simplifying compliance. The Evolution of ODF into the Future Technical opportunities include using more powerful predefined hash algorithms, adopting JSON-based metadata levels, providing native support for verifiable credentials, and creating standardised profiles for government identity systems. Given the growth of digital identity frameworks, ODF is the optimal format for documents requiring both authentication and additional security features.

- [LibreOffice QA Blog: QA/Dev Report: October 2025](#) (2025/11/06 12:00)

General Activities LibreOffice 25.8.2 was announced on October 9 LibreOffice 25.2.7 was announced on October 30 Olivier Hallot (TDF) added help pages for R1C1 Calc formula syntax and DOI citation recognition and improved and updated help on dimension lines, form properties, master documents, command line operations, online update, text boundaries and VBA constants. He also adapted the helper script used for patch submission to a version that works with Help Gábor Kelemen (Collabora) improved the script for finding unneeded includes in the code and did many code cleanups Tomaž Vajngerl (Collabora) continued working on sheet view functionality in Calc Pranam Lashkari, Dennis Francis, Szymon Kłos, Jaume Pujantell and Gülşah Köse (Collabora) worked on LOKit/jsdialog used by Collabora Online Rashesh Padia (Collabora) made the revamped Impress transition list more robust Michael Meeks (Collabora) did code cleanups and optimisations in PPTX export code Miklós Vajna (Collabora) improved image handling in Markdown import and export, continued improving the handling of tracked changes that depend on each other and fixed issues with handling of bulleted lists in PPTX files Xisco Faulí (TDF) added sqlite3, dbm and pythonw.exe to the internal Python, fixed an Impress printing crash, added a few new automated tests and upgraded many dependencies Michael Stahl (Collabora) implemented per-line paragraph properties for Word Compatibility Mode, fixed a PDF export issue involving variable fields and hidden text and fixed automatic captioning of images in Writer in the case of a single pasted image Mike Kaganski (Collabora) improved help for inserting page numbers and accessing remote files, made Google Drive two-factor authentication work on Windows, made date and time arithmetic more

robust, made the loading of macro class modules happen in the correct order, preventing name conflicts, made it so the automatic updater does not run in headless mode, made the display of empty hidden paragraphs between tables match that of Microsoft Word, improved text property handling in PPTX files, reduced console noise in debug builds by marking menu items without icons, fixed an issue with Writer column separator colour sometimes not being saved, made shape identification in ODF export more robust and improved the behaviour of conditional hiding of sections. He also did many code cleanups and optimisations Caolán McNamara (Collabora) added Adobe Font Development Kit for OpenType (AFDKO) library for converting and merging Type 1 fonts to OTF when importing PDFs via pdfium and fixed an (unreleased) issue with scrolling the Calc formula input box. He also fixed crashes and many issues found by static analysers and did code cleanups and optimisations Stephan Bergmann (Collabora) did build fixes, adapted the code to compiler changes and did code cleanups Noel Grandin (Collabora) made it faster to reject tracked changes in Calc, export EPUB files and render SVGs with pattern fills. He also did many code cleanups and optimisations, especially in the area of transparency handling Justin Luth (Collabora) made it so undoing an autocorrection triggered by a newline does not undo the insertion of the newline, fixed document and application colour configurations not being immediately applied, fixed changed font colour reverting to original when clicking Apply, made table AutoFormat styles use automatic font colour, fixed issues when printing in landscape orientation and fixed many spellchecking issues Michael Weghorn (TDF) fixed an issue with displaying extended tooltips for certain elements and fixed accessibility issues in the Additions, Extension Manager and Check for Updates dialogs. He also worked on using native widgets in Qt UIs Balázs Varga (Collabora) made the code for checking file permissions on Windows more robust Christian Lohmaier (TDF) raised the Linux runtime baseline to Almalinux 9, fixed an issue with gallery themes not using translations, made it possible to build qtz/keyID pseudo-language for the UI also with release builds, made media control tooltips in the Navigator translatable and switched the CI to use the WSL-based Windows build setup Jonathan Clark (TDF) implemented start and end paragraph alignment, fixed unwanted spacing between Hangul and non-CJK scripts, made it so RTL/CTL is automatically enabled based on locale, made it so Complex/Asian tab choice in the Character dialog is remembered across sessions, made RTL/CTL and CJK editing features always be shown, made increasing or decreasing font size affect all script types and thus avoiding unexpected discrepancies in situations like editing bulleted lists, fixed unwanted page number repetition in footnotes with continuation notices that end in non-Latin text and fixed string corruption after dismissing Input Method Editor (IME) Julien Nabet did some code cleanups Bayram Çiçek (Collabora) fixed a crash when opening certain XLSX files Heiko Tietze (TDF) resurrected a patch adding better quality icons for bullets, adapted application icons for new styles introduced in macOS 26 Tahoe, fixed shadows in Writer document canvas being sometimes rendered incorrectly, added an option to suppress dragging and dropping of text selection in Writer and continued working on vertical tabs in dialogs Regina Henschel continued working on Natural Sort in Calc data ranges and fixed an issue reading and writing a Data Provider transformation Ujjawal Kumar worked on the Markdown import GSoC project Jean-Pierre Ledure worked on the ScriptForge library Chris Sherlock did code cleanups and refactoring in VCL toolkit René Engelhard (Debian) fixed a pdfium build issue Neil Roberts fixed floating toolbar stealing focus when clicking buttons, fixed incorrect display of digits in default number formatting of a field in Base when language isn't English and fixed a crash when closing the table edit window in Base in certain situations Andras Timar (Collabora) fixed Python-related build issues Henry Castro (Collabora) implemented accessible HTML export from Impress and Draw Laurent Balland made it so changing mode to Matrix in Calc is treated as a modification Jim Raykowski made it so switching between footnotes and endnotes also changes the style, fixed inability to change the navigation order of objects in a single-page Draw document, fixed a Draw Navigator deleting issue affecting GTK3 UI, fixed unwanted expansion of Navigator items in Draw after deleting an object and fixed a couple of crashes related to Writer and Draw Navigator actions Devansh Varshney

worked on the BASIC IDE auto-completion GSoC project Áron Budea (Collabora) fixed special characters in comments not getting escaped in PPTX export, fixed incomplete DOCX export of empty charts and made it so user fields with empty names will not be exported to DOCX Lodev (OSSII) made Notebookbar UIs respect the application context set for an extension Casey Harkins made API-related connection code more robust David Hashe fixed a Writer undo-redo issue related to pasting and headers and footers Bojidar Marinov fixed Find and Replace with formatting sometimes changing the incorrect item Sahil Gautam (Collabora) reworked the presentation of OpenCL and threading options Arnaud Versini did some code cleanups Tamás Zolnai fixed the accuracy of snapping to grid in Writer Pierre improved the script for generating a list of dispatch commands for a wiki article Chenxiong Qi did code cleanups in Python Đoàn Trần Công Danh fixed a memory leak in poppler 25.10.0 Armin Le Grand (Collabora) reduced memory usage when rendering bitmaps using Cairo Karthik Godha (Collabora) fixed import of DOCX content controls with empty content and fixed saving backgrounds, font sizes and slide links from PPT to PPTX files David Gilbert implemented restarting of numbering in lists in PPTX files Dan Kingsley made it so changing the system colour theme on Linux refreshes the style of LibreOffice Qt-based UIs immediately Kudos to Ilmari Lauhakangas for helping to elaborate this list. Reported Bugs 536 bugs, 95 of which are enhancements, have been reported by 320 people. Top 10 Reporters Danat (47) Eyal Rozenberg (30) Gabor Kelemen (Collabora) (20) Aron Budea (19) Telesto (18) Regina Henschel (16) postix (6) BDF (6) Olivier Hallot (5) Mike Kaganski (5) Triaged Bugs 478 bugs have been triaged by 72 people. Top 10 Triagers Buovjaga (117) m_a_riosv (47) Heiko Tietze (36) raal (34) BogdanB (25) fpy (21) Aron Budea (19) V Stuart Foote (15) Mike Kaganski (15) Julien Nabet (11) Resolution of resolved bugs 417 bugs have been set to RESOLVED. Check the following sections for more information about bugs resolved as FIXED, WORKSFORME and DUPLICATE. Fixed Bugs 170 bugs have been fixed by 38 people. Top 10 Fixers Mike Kaganski (20) Olivier Hallot (12) Justin Luth (11) Jonathan Clark (9) Heiko Tietze (7) Jim Raykowski (7) Miklos Vajna (7) Noel Grandin (6) Xisco Fauli (4) Caolán McNamara (4) List of high severity bugs fixed tdf#157365 Options > LibreOffice Writer changes page color scheme from Dark to Light (steps in comment 2) (Thanks to Justin Luth) tdf#167605 (When “Automatic spell checking” is set,) text may overflow from the cell (Thanks to Armin Le Grand (collabora)) tdf#168500 Infinite loop of dialog spawning when opening Spelling in Impress (Thanks to Michael Weghorn) tdf#168870 Calc Formula Bar (inputwin) edit shell text is not fully connected to its scroll bar (Thanks to Caolán McNamara) tdf#169015 Impress crashes when printing notes if notes contain certain content (Thanks to Xisco Fauli) tdf#37128 Writer saves text alignment of RTL paragraph not according to the ODF specification (Thanks to Jonathan Clark) List of crashes fixed tdf#167405 crash collapse a heading (Thanks to Jim Raykowski) tdf#167753 Crash if I set the Slide Transition Modify Transition Variant to Top to Bottom (Thanks to Rashesh Padia) tdf#168627 PDF export dialog crashes when horizontal tabs are enabled (Thanks to Heiko Tietze) tdf#168723 Crash in: rtl_uString_release (Thanks to Mike Kaganski) tdf#168951 Draw crashes when dragging an unnamed object from Navigator and dropping it in the Pages pane (Thanks to Jim Raykowski) tdf#168979 It crashes when I try to open a file (Thanks to Caolán McNamara) tdf#169015 Impress crashes when printing notes if notes contain certain content (Thanks to Xisco Fauli) tdf#169115 Fileopen XLSX crashes debug Calc (Thanks to Bayram Çiçek) tdf#169147 Macro: Crash if operator Like has bad Pattern (Thanks to Mike Kaganski) List of performance issues fixed tdf#133976 Export to EPUB takes a long time (Thanks to Noel Grandin) List of old bugs (more than 4 years old) fixed tdf#100998 Add an option in Writer Configuration to disable selected text drag’n’drop when editing (Thanks to Heiko Tietze) tdf#104888 “Ctrl + Shift + Enter” hotkey after “F2” on the cell that I carelessly forgot to hold “Ctrl + Shift” when I entered matrix related formula should be treated as a modification to the formula on the cell. (Thanks to Laurent Balland) tdf#117020 EDITING Spellchecker should not be recovered together with the file (Thanks to Justin Luth) tdf#118350 Implement fo:text-align start and end paragraph alignment (Thanks to Jonathan Clark) tdf#125600 Calc spellcheck does not

recognize correct spelling of KWh (Thanks to Justin Luth) tdf#126493 EDITING: Objects don't snap to grid properly (Thanks to Tamás Zolnai) tdf#130196 Changing nav-order in navigator does not work on single page (Thanks to Jim Raykowski) tdf#132074 python shelve don't work - missing dbm module (Thanks to Xisco Fauli) tdf#133976 Export to EPUB takes a long time (Thanks to Noel Grandin) tdf#135435 ACCESSIBILITY: No information from screenreader in "check for updates" window (Thanks to Michael Weghorn) tdf#136663 Disable "Apply spacing between Asian and non-Asian text" option by default for Korean users (Thanks to Jonathan Clark) tdf#138249 PARAGRAPH STYLE DIALOG: Font color style change breaks when reverting to automatic color (Thanks to Justin Luth) tdf#139076 UI: Bulleted and numbering dialog image tab uses low resolution gif images (Thanks to Heiko Tietze) tdf#142118 Application color ...

- [Miklos Vajna: Interdependent tracked changes improvements in Writer, part 4: direct accept/reject](#) (2025/11/04 07:11)

Writer has some support for interdependent (or hierarchical) tracked changes: e.g. the case when you have a delete on top of an insert. See the third post for background. This work is primarily for Collabora Online, but the feature is available in desktop Writer as well. Motivation¶ Interdependent changes mean that the UI shows one type of change on top of another change, e.g. formatting on top of insert. Writer knows the priority of each type, so in case you have an insert or delete change and on top of that you have a formatting, then the UI will look "through" the formatting and work on the underlying insert or delete when you navigate with your cursor to a position with multiple changes and you press Accept on the Review tab of the notebookbar. Usually this is what you mean, but what if you want to work on the formatting at the top, directly? You can now open the Manage Changes dialog using the Manage button on the Review tab of the notebookbar and if you go to the formatting change row of the dialog, then pressing Accept there will accept the formatting change, not the insert or delete change. This is possible, because the dialog gives you a way to precisely select which tracked change you want to work with, even if a specific cursor position has multiple tracked changes. Results so far¶ Here is a sample ins-then-format.docx document from the core.git testcases, the baseline has an insertion, and part of that is covered by an additional formatting change on top: Interdependent tracked change: baseline If you just go in the middle of the document and press Accept, that will work with the more important insert change, so the result looks like this: Interdependent tracked change: default accept result But now you can also open the Manage Changes dialog, to be more specific by directly selecting the formatting change: Interdependent tracked change: direct accept via the dialog And when you accept the formatting change directly, the result will be just the insert change: Interdependent tracked change: direct accept result You can save and load the results in both DOCX and ODT, as usual. How is this implemented?¶ If you would like to know a bit more about how this works, continue reading... :-) As usual, the high-level problem was addressed by a series of small changes. Core side: tdf#166319 sw interdependent redlines: allow accept/reject for fmt on ins/del tdf#166319 sw interdependent redlines: fix redo of accept for fmt on ins/del tdf#166319 sw interdependent redlines: fix redo of direct reject for format Want to start using this?¶ You can get a development edition of Collabora Online 25.04 and try it out yourself right now: try the development edition. Collabora intends to continue supporting and contributing to LibreOffice, the code is merged so we expect all of this work will be available in TDF's next release too (26.2).

- [Marius Popa Adrian: A modern C++ wrapper for the Firebird database API](#). (2025/10/29 15:15)

A modern C++ wrapper for the Firebird database API. Documentation | Repositoryfb-cpp provides a clean, modern C++ interface to the Firebird database engine. It wraps the Firebird C++ API with RAII principles, smart pointers, and modern C++ features. Features Modern C++: Uses C++20 features for type safety and performanceRAII: Automatic resource management with smart pointersType Safety:

- [LibreOffice Dev Blog: enumarray for better data arrays - EasyHack](#) (2025/10/22 19:30)

In LibreOffice C++ code, there are many cases where developers have string literals that should be used in the code. If these are messages in the graphical user interface (GUI), they should be added to the translatable messages. But, there are many cases where the string literals has nothing to do with other languages, and they should be used as they are. In this case, enumarray is helpful. Although they are not limited to string literals, and can be used for any data. Using Symbolic Constants In old C code, using `#define` was the preferred way one could give a name to a string literal or other kinds of data. For example, consider this code: `const char[] FRAME_PROPNAME_ASCII_DISPATCHRECORDERSUPPLIER = "DispatchRecorderSupplier"; const char[] FRAME_PROPNAME_ASCII_ISHIDDEN = "IsHidden"; inline constexpr OUString FRAME_PROPNAME_ASCII_LAYOUTMANAGER = "LayoutManager"; const char[] FRAME_PROPNAME_ASCII_TITLE = "Title"_ustr; const char[] FRAME_PROPNAME_ASCII_INDICATORINTERCEPTION = "IndicatorInterception"; const char[] FRAME_PROPNAME_ASCII_URL = "URL";` And also, the relevant states: `#define FRAME_PROPHANDLE_DISPATCHRECORDERSUPPLIER 0 #define FRAME_PROPHANDLE_ISHIDDEN 1 #define FRAME_PROPHANDLE_LAYOUTMANAGER 2 #define FRAME_PROPHANDLE_TITLE 3 #define FRAME_PROPHANDLE_INDICATORINTERCEPTION 4 #define FRAME_PROPHANDLE_URL 5` Although this C code still works in C++, it is not the desired approach in modern C++. Using enumarrays In modern C++ code, you can use `enumarray`, which is defined in `o3tl` in LibreOffice. The above code becomes: `enum class FramePropNameASCII { DispatcherRecorderSupplier, IsHidden, LayoutManager, Title, IndicatorInterception, URL, LAST=URL };` And also, the string literal definitions: `constexpr o3tl::enumarray<FramePropNameASCII, OUString> FramePropName = { u"DispatchRecorderSupplier"_ustr, u"IsHidden"_ustr, u"LayoutManager"_ustr, u"Title"_ustr, u"IndicatorInterception"_ustr, u"URL"_ustr };` Why an `enumarray`? The names are much more readable, as they do not have to be ALL_CAPPS, as per convention for symbolic constants in C. Their usage is also quite easy. For example, one can use `[]` to access the relevant string literal: `- xPropSet->getPropertyValue(FRAME_PROPNAME_ASCII_LAYOUTMANAGER) >>= xLayoutManager; + xPropSet->getPropertyValue(FramePropName[FramePropNameASCII::LayoutManager]) >>= xLayoutManager;` Final Notes In LibreOffice, `enumarrays` are not limited to string literals, and they can be used with other data. This task is filed as [tdf#169155](#), and if you like, you may try finding some instances in the code and modernize it using `enumarrays`. To learn more about LibreOffice development, you can refer to TDF Wiki. You may follow this blog to read about EasyHacks, tutorials and announcements related to LibreOffice development.

- [LibreOffice QA Blog: QA/Dev Report: September 2025](#) (2025/10/21 11:20)

General Activities LibreOffice 25.2.6 was announced on September 8 Olivier Hallot (TDF) improved the help for Select Function in Calc's formula bar, expanded help for the selection of chart data sources, added AutoFilter and Pivot Table/Chart to the help page on sheet protection, added information about summary above/below to the help of Calc's SUBTOTAL() function, documented sensitivity in the help page for Solver and added the description meta element to the Help page templates Gábor Kelemen (Collabora) improved the script for finding unneeded includes in the code and did many code cleanups Tomaž Vajngerl (Collabora) implemented a sheet view functionality in Calc allowing to manipulate AutoFilters in one view without affecting other views Pranam Lashkari, Maya Stephens, Gökyay Şatır, Rashesh Padia and Mohit Marathe (Collabora) worked on LOKit used by Collabora Online. Maya also fixed an issue with Ctrl+clicking hyperlinked objects Miklós Vajna (Collabora) expanded Writer Markdown export support, worked on change tracking of formatting, continued improving the handling of tracked changes that depend on each other and fixed bullet list style going missing with PPTX export Xisco Faulí (TDF) implemented right-to-left text direction and bidirectional text support in SVG import, added venv to the internal Python, fixed crashes, added several new automated tests, upgraded many dependencies and did many code cleanups and optimisations Michael Stahl (Collabora) improved the DOCX compatibility of floating tables, helped Miklós with interdependent change tracking in Writer and fixed an issue with document compression Mike Kaganski (Collabora) fixed Help for BASIC's Str

function, adapted Writer's change tracking to DOCX's inability to track the deletion of comments, made date handling code more robust, improved the Visual Studio IDE integration, made the handling of double variables in BASIC more robust, replaced dtoa library with fast_float for faster string-to-float parsing and fixed an RTF import issue with some paragraphs appearing outside of their frames. He also did many code cleanups and optimisations Caolán McNamara (Collabora) fixed many issues found by static analysers and did code cleanups and optimisations Stephan Bergmann (Collabora) worked on the WASM build. He also adapted the code to compiler changes and did code cleanups Noel Grandin (Collabora) fixed an (unreleased) issue with image fills disappearing at lower zoom levels, improved the performance of spreadsheets with thousands of shapes, improved the performance of 3D charts, fixed a performance issue with importing SVGs with small scaling, improved the performance of Calc's Remove Duplicates, implemented Skia native rendering of bitmap tiling to improve the performance of filled polygons and fixed an issue with Insert Special Character dialog not being available with mergedlibs builds (Windows releases). He also did many code cleanups and optimisations, especially in the area of transparency handling Justin Luth (Collabora) fixed an issue with the alignment setting in Page Number Wizard, fixed localised accelerator shortcut conflicts, fixed many issues with page styles, page breaks and spell checking, improved the Bullets and Numbering dialog by showing the currently applied style as selected, improved the layout of the Tabbed UI and made the "restart numbering" option in lists deactivate upon joining of lists. He also fixed some crashes Michael Weghorn (TDF) continued cleaning up and reorganising accessibility-related code, improved the input method support of Qt-based UIs and improved the accessibility of Calc cell editing and the Check for Updates dialog. He also worked on using native widgets in Qt UIs Balázs Varga (Collabora) continued polishing PPTX placeholder text handling and fixed Calc comment text or entire comment elements sometimes getting removed when saving Patrick Luby fixed a macOS build issue, made it so LibreOffice no longer intercepts shortcuts using both Command and Option on macOS, fixed tab item drawing issues on macOS Tahoe and fixed a macOS crash Christian Lohmaier (TDF) improved the winget-based Windows build setup Jonathan Clark (TDF) made handling of rich text pastes into Edit Engine input boxes (Calc, Impress) more robust Andreas Heinisch enabled pasting unformatted text in Impress's outline view, made Calc's random number fill UI more robust and fixed a Calc crash Julien Nabet fixed DXF files with comments no longer being recognised as drawings and fixed (unreleased) crashes Bayram Çiçek (Collabora) continued polishing support for xmlMaps in XLSX files and improved MSO compatibility with line numbering in footnotes Heiko Tietze (TDF) continued working on vertical tabs in dialogs, fixed repetitive presets for bullets, improved the visual style of the AutoFilter button and improved the contrast between active and inactive tabs in Tabbed UI Regina Henschel continued working on Natural Sort in Calc data ranges Jim Raykowski made it so page ranges are shown for heading tooltips in the Navigator, fixed a text selection issue in the Quick Find Sidebar deck and fixed some crashes Ujjawal Kumar worked on the Markdown import GSoC project Pierre Vacher made it so new tables in Base are saved under unique names and made row insertion in Base more robust Kohei Yoshida (Collabora) made the data format detection with orcus library more robust Jean-Pierre Ledure worked on the ScriptForge library and Access2Base Adam Seskunas worked on the C++ Report Builder GSoC project Chris Sherlock did code cleanups and refactoring in VCL toolkit László Németh implemented features for minimum letter spacing and glyph scaling in paragraph justification René Engelhard (Debian) did some build fixes Amin Irgaliev added connector support for Writer Neil Roberts removed unexpected and confusing shortcut key behaviour on macOS, fixed inability to toggle focus modes with F2 in Calc, made it so movement accelerators are the expected ones on macOS, fixed issues with removing assigned shortcut keys and added a unit test for accelerator configurations Ilmari Lauhakangas (TDF) made it so Enclose with characters in Writer is no longer active by default and worked on vertical tabs in dialogs Thorsten Behrens (Collabora) fixed a build issue after Boost upgrade Kurt Nordback (Collabora) implemented support for OOXML chart styles Jaap Aarts fixed building on FreeBSD Andras Timar

(Collabora) made it so EXIF Orientation tag is always respected when importing a JPEG image Hossein Nourikhah (TDF) made it so .NET SDK examples and tests are no longer bound to a specific version Mohamed Ali worked on the Rust UNO bindings GSoC project Ashod Nakashian (Collabora) made loading empty documents more robust Aung Khant added saving window size and position data for each document Henry Castro (Collabora) fixed using custom text shapes in the Notes view of Impress/Draw and fixed a Writer view update issue with database records Laurent Balland fixed a French Autotext issue and restored a gradient gone missing in Impress's Candy template Dan Williams did cleanups and fixes in macOS address book code Daniel Richard G. improved apparmor profiles Khaled Hosny (Alif Type) made it so ligatures get disabled in Impress/Draw with character spacing Wendi Gan fixed selection and resizing issues with captioned images in Writer Devansh Varshney worked on the BASIC IDE auto-completion GSoC project Juraj Šarinay fixed memory leaks in digital signing code Áron Budea (Collabora) fixed unwanted shifting in certain DOCX tables Kudos to Ilmari Lauhakangas for helping to elaborate this list. Reported Bugs 403 bugs, 44 of which are enhancements, have been reported by 270 people. Top 10 Reporters Eyal Rozenberg (22) Mike Kaganski (10) SATYA SRINIVAS K (9) Telesto (7) Aron Budea (6) Heiko Tietze (6) nobu (6) Regina Henschel (6) Dragan (5) Miklos Vajna (5) Triaged Bugs 330 bugs have been triaged by 64 people. Top 10 Triagers m_a_riosv (53) Xisco Faulí (29) raal (28) Heiko Tietze (21) BogdanB (18) V Stuart Foote (12) Mike Kaganski (12) Miklos Vajna (10) nobu (9) Buovjaga (8) Resolution of resolved bugs 381 bugs have been set to RESOLVED. Check the following sections for more information about bugs resolved as FIXED, WORKSFORME and DUPLICATE. Fixed Bugs 184 bugs have been fixed by 34 people. Top 10 Fixers Justin Luth (24) Mike Kaganski (14) Noel Grandin (14) Heiko Tietze (12) Miklos Vajna (9) László Németh (5) Xisco Fauli (5) Balazs Varga (4) Olivier Hallot (4) Jim Raykowski (4) List of high severity bugs fixed tdf#167042 Calc cell comments are deleted on Save (Thanks to Balazs Varga) tdf#167897 LibreOffice changes the default folder to "saves as" and "export to pdf" many times for a same opened document, if it's saved during use (Thanks to Samuel Mehrbrodt) tdf#168306 When a previously saved .odg file is opened in LibreOffice Draw, any line is selected, then under "properties" the line "thickness" is selected, the previously saved line thickness automatically changes to 0.5. (Thanks to Caolán McNamara) tdf#168344 When i use the highlighter tool, the rest of the line after the highlighted content goes all white. (Thanks to Noel Grandin) tdf#168377 crash: use sidebar manage changes to reject all when UNDO is involved (Thanks to Justin Luth) tdf#168497 [26.2] Crash LibreOffice Calc when opening the Data Provider (Thanks to Julien Nabet) tdf#168548 CRASH: launching hyperlink dialog with horizontal tabs (gen) (Thanks to Heiko Tietze) tdf#41777 Window size reopening a document not like size when saved (Thanks to Aung) tdf#62032 Cannot specify a Paragraph Style to use a numbering level (over 1) for a chosen Numbering (List) style (See comment 35) (Thanks to Justin Luth) List of crashes fixed tdf#166713 Crash when editing paragraph after toggling outline folding (Thanks to Jim Raykowski) tdf#168159 CRASH: deleting columns and undoing (Thanks to Noel Grandin) tdf#168363 CRASH: pasting to Master Hangout (Thanks to Xisco Fauli) tdf#168377 crash: use sidebar manage changes to reject all when UNDO is involved (Thanks to Justin Luth) tdf#168478 Edit under filter and then reset the filter, Calc crashed (Thanks to Andreas Heinisch) tdf#168497 [26.2] Crash LibreOffice Calc when opening the Data Provider (Thanks to Julien Nabet) tdf#168548 CRASH: launching hyperlink dialog with horizontal tabs (gen) (Thanks to Heiko Tietze) tdf#168572 Export to markdown: crash when Writer document contains a formula object (Thanks to Julien Nabet) tdf#168599 Draw Crashes on new document when pressing CTRL+pgUp/pgDown (Thanks to Mike Kaganski) tdf#168609 LibreOffice writer 25.8.1 crashes when trying to insert a frame for a table (Thanks to Patrick Luby) List of performance issues fixed tdf#132494 Trying to save SVG image from presentation/drawing causes LibO to use 100% cpu and lock up (Thanks to Noel Grandin) tdf#149592 Importing SVG with too small scaling causes bad performance (Thanks to Noel Grandin) tdf#154913 Poor performance for ODS with frozen rows and columns (Thanks to Noel Grandin) List of old bugs (more than 4 years old) fixed

tdf#100287 [UI] Header and Footer controls (same content; left/right, first/other) don't update and are not independent (Thanks to Justin Luth)
 tdf#104833 Scroll panes in Check for Updates dialog do not expose names to assistive technologies (Thanks to Michael Weghorn) tdf#108791
 Comments with track changes deleted and crossed in Writer are wrongfully seen in MSO Word as if not deleted (Thanks to Mike Kaganski)
 tdf#111927 PPTX: Placeholder title text becomes small after clicking in&out (Thanks to Balazs Varga) tdf#113213 When "Continue previous numbering" is applied, it should also toggle off "restart numbering" for any item(s) that were selected for "Continue previous numbering" (Thanks to Justin Luth)
 tdf#114343 Adding a new Title Page via Format > Title Page is impossible with a ToC at the beginning of a document (Thanks to Justin Luth)
 tdf#116975 Laggy behavior when clicking between table cells in impress (Thanks to Noel Grandin) tdf#121253 Database wizard opens when press cmd+A inside the extension manager window (Thanks to Neil Roberts) tdf#126530...

- [LibreOffice Dev Blog: enum class instead of unscoped enum - EasyHack](#) (2025/10/16 14:05)

Since C++11 when enum class (also named scoped enum) is introduced, it is preferred to plain enum which is inherited from C programming languages. The task here is to convert the old enum instances to enum class. Rationale enum class has many benefits when compared to plain enum, as it provides better type safety among other things. Implicit conversion to integers, lack of ability to define the underlying data type and compatibility issues were some of the problems with plain enum that enum class solved in C++11. Although since then enum has improved and one can specify underlying type in the scoped enumerations. Plain enums pollute namespace, and you have to pick names that are too long, and have to carry the context inside their names. For example: INETMSG_RFC822_BEGIN inside enum _ImplNetRFC822MessageHeaderState. With an enum class, it is simply written as HeaderState::BEGIN. When placed inside a file/class/namespace that makes it relevant, it is much easier to use: it is more readable, and causes no issues for other identifiers with possible similar names., See this change: 593f08303 convert enum _ImplNetRFC822MessageHeaderState to enum class You can read more about that in: Why is enum class considered safer to use than plain enum? Finding Instances You may find some of the instances with: `$ git grep -w enum *.cxx *.hxx|grep -v "enum class"` When you count it with `wc -l`, it shows something more than 2k instances. Examples Commits You can see some of the previous conversions here, which is around 1k changes: `$ git log --oneline -i -E --grep="convert enum|scoped enum"` This is a good, but lengthy example of such a conversion: 9072c5c855 convert SbxFlagsBits to scoped enum Implementation First of all, please choose good names for the new enum class and values. For example, you may convert APPLICATION_WINDOW_TITLE into Application::WindowTitle. Therefore, do not use the old names as they were. Converting enum to enum class is not always straightforward. You should try to understand the code using the enum, and then try to replace it with enum class. You may need to add extra state/values for situations where 0 or -1 or some default value was used. There are cases where a numerical value is used for different conflicting purposes, and then you have to do some sort of conflict resolution to separate those cases. You may end up modifying more and more files, and a few static_casts where they are absolutely necessary because you are interpreting some integer value read from input. These are the places where you should check the values yourself in the code. You have to make sure that the numerical value is appropriate before casting it to the enum class. If you want to do bitwise operations, you should use o3tl::typed_flags, for example: enum class FileViewFlags { None = 0x00, MultiSelection = 0x02, ShowType = 0x04, ShowNone = 0x20, }; template<> struct o3tl::typed_flags : o3tl::is_typed_flags<FileViewFlags, 0x26> {} Then, you may use it like this: `if (nFlags & FileViewFlags::MULTISELECTION) mxTreeView->set_selection_mode(SelectionMode::Multiple);` Please note that 0x26 is the mask, and is calculated by applying OR over all possible values. All the values must be non-negative. Final Notes This is a simple development task for LibreOffice also known as EasyHack, which is filed in Bugzilla as tdf#168771. These small tasks are defined to help newcomers to LibreOffice development community to improve their skills with

LibreOffice coding. You may find other instances related to C++ here: [TDF Wiki: List of EasyHacks](#)

- [Miklos Vajna: Markdown import/export in Writer](#) (2025/10/07 06:13)

Writer recently got a Markdown import & export filter and there were a number of improvements to that. This work is primarily for Collabora Online, but the feature is available in desktop Writer as well. Motivation¶ Ujjawal Kumar contributed a markdown import to Writer, as part of Google Summer of Code (GSoC) this summer. Mike Kaganski of Collabora also created a minimal markdown export in Writer. I looked at the feature differences between the two, and filled in various gaps in the markdown export. I also added a few general markdown import/export improvements relevant for normal Writer documents, like embedded image support. Results so far¶ Here is a sample case of a document using inline code spans: Code span: baseline Exporting this to markdown & loading back to Writer, the code span was lost: Code span: old result And now it's preserved: Code span: new result This also works with code blocks. Second, here is a document with lists: Lists: baseline Exporting this to markdown & loading back to Writer, the lists were lost: Lists: old result And now they are preserved: Lists: new result This also works with nested lists. Third, here is a document with an image: Image: baseline Exporting this to markdown & loading back to Writer, the image was lost: Image: old result And now it's preserved: Image: new result This also works with embedded and anchored images. Fourth, here is a document with a table: Table: baseline Exporting this to markdown & loading back to Writer, the table was lost: Table: old result And now it's preserved: Table: new result This also works with table alignments and nested tables (to the extent the markdown markup allows that). Fifth, here is a document with a quote block: Quote: baseline Exporting this to markdown & loading back to Writer, the quote's paragraph indentation was lost: Quote: old result And now it's preserved: Quote: new result How is this implemented?¶ If you would like to know a bit more about how this works, continue reading... :-)

As usual, the high-level problem was addressed by a series of small changes. Core side: desktop lok, doc save: register .md for Markdown sw markdown export: handle code tdf#168152 sw markdown export: handle lists tdf#168172 sw markdown export: handle images tdf#167564 sw markdown export: handle tables sw markdown export: handle block quote tdf#168317 sw markdown export: handle code block sw markdown export: handle table cell adjustment tdf#168341 sw markdown filter: handle links on images sw markdown export: handle line breaks tdf#168446 sw markdown export: improve image name/description/title handling tdf#167564 sw markdown export: handle multi-para table cells tdf#167564 sw markdown export: handle nested table cells tdf#168617 sw markdown filter: map tasks to checkbox content controls and back sw markdown filter: import images with 'data:' URLs sw markdown filter: export non-linked inline images tdf#168662 sw markdown export: extract inline image export functions tdf#168662 sw markdown export: handle anchored images Want to start using this?¶ You can get a development edition of Collabora Online 25.04 and try it out yourself right now: try the development edition. Collabora intends to continue supporting and contributing to LibreOffice, the code is merged so we expect all of this work will be available in TDF's next release too (26.2).

- [LibreOffice Dev Blog: Debugging tips for LibreOffice](#) (2025/09/18 14:11)

If you are working with LibreOffice code, trying to understand the code, fix bugs, or implement new features, you will need to debug the code at some point. Here are some general tips for a good debugging experience. Let's start from the platform Choose the Right Debug Platform Choosing a platform to debug usually depends on the nature of the problem. If the problem is Windows-only, you need a Windows environment to build and debug the problem. But, if the problems can be reproduced everywhere, then you can choose the platform of your choice with the debugging tools that you prefer to debug the problem. On Linux, it matters if you are running X11 or Wayland. Also, as there are multiple graphical back-ends available for LibreOffice, it matters if you are using X11, GTK3/4, or Qt5/6 back-end for your debugging. Some bugs are specific to GTK, then you should use GTK3 UI for testing. In 2025, GTK4 UI of LibreOffice is still experimental, so it is better to work with GTK3. For

making the debugging easier, many developers work on X11 (gen) UI for debugging. Debugging Tools Various debugging tools can be used to debug the soffice.bin/soffice.exe LibreOffice binary that you have built. For the common debuggers, you can use GDB on Linux, lldb on macOS, and WinDbg or Visual Studio on Windows. For using the above debuggers, you can use the IDE or front-end that support them. Various IDEs are usable with LibreOffice code. For a detailed explanation, refer to this Wiki article: TDF Wiki: Development/IDE Make sure that you can build and debug a simple program before trying to build and debug LibreOffice. Environment Variables To have a better debugging experience, or to avoid problems you may have to customize the debugging session with environment variables. A complete article of the TDF Wiki is dedicated to discuss the environment variables that can be used with LibreOffice: TDF Wiki: Development/Environment variables Here is some of the most important ones: 1) Using the X11 user interface: If you want to use the X11 back-end that is simpler, and usually easier to work with on debug sessions, you have to set SAL_USE_VCLPLUGIN environment variable: export SAL_USE_VCLPLUGIN=gen That is specially useful when you are debugging graphical problems. But in some cases, you may need to avoid it or at least customize it. For example, while debugging mouse-related problems you may need to tell LibreOffice to avoid mouse grabbing this way: export SAL_NO_MOUSEGRABS=1 2) Using GTK user interface If you are using GTK user interface, then you may use GTK inspector to interactively debug LibreOffice GUI. You can use it this way: export GTK_DEBUG=interactive Pretty Printers In solenv/gdb/ inside LibreOffice source code, you may find pretty printers for GDB. This is helpful when debugging LibreOffice with GDB, to be able to see data in a more readable way. Dumping Data Sometimes when you debug a LibreOffice application, it is easier to dump data into file for easier diagnosis. There are key combinations that are enabled in debug mode for this purpose. To use them, you need to build LibreOffice with the configuration option --enable-dbgutil. These are some of them related to text boxes, rendered with editeng module: Ctrl+Alt+F11 - toggles dumping the edit engine state to the editenginedump.log on Draw Ctrl+Alt+F12 - dumps the current edit engine state to editenginedump.log There are other key combinations for using in Writer and Draw: SW_DEBUG=1 enables F12 for dumping layout.xml, and Shift+F12 for nodes.xml which are Writer document dumps SD_DEBUG=1 enable F12 for model.xml which contains Draw graphic object dump These key combinations can be used with Calc: Ctrl+Shift+F12: Dump the column width of the first 20 columns Ctrl+Shift+F11: Dump the graphic objects and their position and size in pixels Ctrl+Shift+F6: Dump the cell properties' of the current selection as dump.xml These key combinations help to debug some useful details about the application for diagnosis. They are only active in debug mode, as described. Further Information We have a complete TDF Wiki article dedicated to debugging. So, make sure that you take a look at the relevant parts: TDF Wiki: How to debug Debugging needs patience, but can be the best way to find the root cause of some bugs.

- [LibreOffice QA Blog: QA/Dev Report: August 2025](#) (2025/09/16 09:42)

General Activities LibreOffice 25.8.0 and LibreOffice 25.8.1 were announced on August 20 and August 29 respectively Olivier Hallot (TDF) updated help for the option to load printer settings with document, sorting blocks of cells in Calc, hyphenation, statistical functions, number of lines in charts, exponentiation operator in Calc, remote files, Edit menu in Calc, object rotation, Math options and MATCH function in Calc Celia Palacios added help for the new Intersect() method in ScriptForge Gábor Kelemen (Collabora) did many code cleanups Tomaž Vajngerl (Collabora) did many code cleanups and added OOXML test documents for text fitting / scaling Pranam Lashkari and Marco Cecchetti (Collabora) worked on LOKit used by Collabora Online. Marco also made it so hovering with the mouse over Chart data range colour palette entries in the Sidebar shows a live preview in the active chart Miklós Vajna (Collabora) added list and inline code block support for Markdown export and continued improving the handling of tracked changes that depend on each other Xisco Faulí (TDF) fixed crashes, added over a dozen new automated tests, upgraded many dependencies and did many code cleanups and optimisations Michael Stahl (Collabora) made it so pasted anchored objects are no longer

selected by default while adding an expert configuration option for the behaviour, added overline support to XHTML export and worked around a dbus bug affecting the build process on some Linux systems Mike Kaganski (Collabora) fixed an issue with embedded fonts getting dropped from opened files in certain scenarios on Windows, made it so the user can choose to either discard license-restricted embedded fonts in an opened document or switch to read-only mode, improved PPTX compatibility with trailing empty lines in automatically shrinking text boxes, fixed long links getting truncated when exporting to XLSX, fixed issues with inserting hyperlinks in Calc via the API, made Calc text insertion API methods more robust, fixed inserting PDFs into spreadsheets, fixed a string handling issue in Basic's Format function, fixed a VBA macro issue with dates and fixed processing of escaped backslashes in RTF files. He also did many code cleanups and optimisations Caolán McNamara (Collabora) fixed many issues found by static analysers and did code cleanups and optimisations Stephan Bergmann (Collabora) worked on the WASM build. He also adapted the code to compiler changes and did code cleanups Noel Grandin (Collabora) improved the scrolling speed in Writer documents with lots of comments. He also did many code cleanups and optimisations, especially in the area of transparency handling Justin Luth (Collabora) improved DOCX compatibility with margins of aligned floating objects, fixed right/left only page breaks going missing with DOC/DOCX export, fixed a DOCX indentation issue, fixed column breaks going missing in certain DOCX files, fixed an issue with numbered lists created by AutoCorrect, made it so justified text with section breaks in saved DOC files no longer triggers an MS Word bug and fixed numbering or bullets getting lost when applying a paragraph style Michael Weghorn (TDF) continued cleaning up and reorganising accessibility-related code, improved the accessibility of Spelling dialog, fixed an (unreleased) issue with Borders tab in formatting dialogs, improved the handling of programmatic accessibility selections and made the GTK4 Sidebar more robust. He also worked on using native widgets in Qt UIs Balázs Varga (Collabora) continued polishing support for Microsoft Media Foundation APIs on Windows, fixed an issue with the display of customised placeholder text set in the master slides of PPTX documents, fixed an issue with PPTX placeholder text becoming small after focusing into and out of it, fixed unwanted copying of elements when inserting rows before pivot table in row 1 in Calc and made it so empty Calc cells are never rotated Patrick Luby fixed not being able to access or create an ODB file based on the macOS system addressbook and made the addressbook handling code more robust. He also adjusted UI widget code after changes in macOS Tahoe Christian Lohmaier (TDF) fixed the MSI installer for Windows Arm64 and improved the winget-based Windows build setup Jonathan Clark (TDF) improved the performance of certain poorly structured DOC files, added handling of vertical text in DOCX files adhering to OOXML strict format and improved Microsoft Word compatibility regarding the laying out of text using CJK fonts Andreas Heinisch fixed some issues with tooltips and the display of URLs Julien Nabet did some code cleanups Bayram Çiçek (Collabora) implemented support for xmlMaps in XLSX files Heiko Tietze (TDF) applied vertical tabs to many dialogs and fixed an issue with column headers overlapping in Calc when the columns are very narrow Sahil Gautam (Collabora) added a UNO command to select a comment in the Navigator and did some code cleanups Armin Le Grand (Collabora) worked on EditEngine rendering revamp and other rendering-related reworks Arnaud Versini did some code cleanups Regina Henschel fixed unwanted duplication of "mouse as pen" objects in slideshows and implemented API support for Natural Sort in Calc data ranges David Gilbert implemented support for importing PDFs with transparency groups Ujjawal Kumar worked on the Markdown import GSoC project Pierre Vacher fixed an issue with creating tables in Base using the wizard, made the Base ResultSet code more robust and fixed a Base issue with redisplaying the authentication dialog after a failed connection Kohei Yoshida (Collabora) implemented automatic mapping of generic JSON and XML documents to Calc using the orcus library Markus Mohrhard fixed column labels not updating after inserting rows below title in Calc, fixed cell comments not getting exported to FODS in some cases, fixed range names getting dropped in XLSB export, fixed incorrect sizing of charts in XLS import and fixed a crash during colorscale calculation Jean-Pierre Ledure worked on

the ScriptForge library Adam Seskunas worked on the C++ Report Builder GSoC project Chris Sherlock did code cleanups and refactoring in VCL toolkit Rico Tzschichholz (Ubuntu) fixed a test failure on armhf platform László Németh added DTP feature maximum letter spacing to Writer René Engelhard (Debian) improved the build system options Bartosz Kosiorek improved EMF graphics support Darshan Upadhyay added UNO commands for setting page orientation and size and sheet margins in Calc Amin Irgaliev added connector support for Calc Neil Roberts made it so comments are not scrolled into view after a keypress that does not do anything Kudos to Ilmari Lauhakangas for helping to elaborate this list.

Reported Bugs 459 bugs, 48 of which are enhancements, have been reported by 295 people. Top 10 Reporters Mike Kaganski (14) Regina Henschel (14) Eyal Rozenberg (13) Gabor Kelemen (allotropia) (12) prrvchr (12) Telesto (9) nobu (8) Heiko Tietze (8) Xisco Faulí (8) Gerald Pfeifer (7)

Triaged Bugs 376 bugs have been triaged by 63 people. Top 10 Triagers m_a_riosv (59) Xisco Faulí (39) BogdanB (31) Mike Kaganski (27) Heiko Tietze (22) raal (17) Olivier Hallot (15) Saburo (13) nobu (12) V Stuart Foote (11)

Resolution of resolved bugs 302 bugs have been set to RESOLVED. Check the following sections for more information about bugs resolved as FIXED, WORKSFORME and DUPLICATE.

Fixed Bugs 122 bugs have been fixed by 28 people. Top 10 Fixers Mike Kaganski (15) Noel Grandin (12) Heiko Tietze (8) Justin Luth (7) Olivier Hallot (6) Michael Weghorn (5) Markus Mohrhard (4) Caolán McNamara (4) Miklos Vajna (4) Xisco Fauli (4)

List of critical bugs fixed tdf#167859 Creating or editing secondary x-axis in a chart crashes program. (Thanks to Xisco Fauli) tdf#167875 Writer crashes if you right click in latest table in the big document (Thanks to Xisco Fauli) tdf#167956 CRASH: opening a second document with tabbed notebookbar (WORKAROUND: Change UI to Standard Toolbar from View – User Interface) (Thanks to Noel Grandin) tdf#167988 Deleting an existing “Mean value line” or a “Trend line” in a chart crashes program. (Thanks to Mike Kaganski) tdf#168017 CRASH: importing certain documents (Thanks to Caolán McNamara)

List of high severity bugs fixed tdf#167812 Dark mode: font dropdown shows white rectangles (GTK3) (Thanks to Noel Grandin) tdf#167893 Presentation mode adds “random” visual artefacts to box shapes/groups (Thanks to Noel Grandin)

List of crashes fixed tdf#167859 Creating or editing secondary x-axis in a chart crashes program. (Thanks to Xisco Fauli) tdf#167875 Writer crashes if you right click in latest table in the big document (Thanks to Xisco Fauli) tdf#167901 In 2nd sheet, copy (and close), Crash (Thanks to Xisco Fauli) tdf#167936 LibreOffice crash (segmentation fault) when opening document from UNO pipe (Thanks to Noel Grandin) tdf#167956 CRASH: opening a second document with tabbed notebookbar (WORKAROUND: Change UI to Standard Toolbar from View – User Interface) (Thanks to Noel Grandin) tdf#167988 Deleting an existing “Mean value line” or a “Trend line” in a chart crashes program. (Thanks to Mike Kaganski) tdf#168017 CRASH: importing certain documents (Thanks to Caolán McNamara) tdf#168139 LibreOffice crash asserts with Skia/Vulkan on X11 (gen) VCL backend (Thanks to Hossein)

List of performance issues fixed tdf#156297 In Calc, scrolling becomes very slow when hiding (many) columns (Thanks to Caolán McNamara)

List of old bugs (more than 4 years old) fixed tdf#104827 FILESAVE DOC: Justified Text with Section Breaks Incorrectly Exported (Thanks to Justin Luth) tdf#108244 Cell Comments lost if not displayed before Saving in FODS Format (steps in comment 22) (Thanks to Markus Mohrhard) tdf#111634 Multiple memory leaks in libmacabdrv1 (Thanks to Patrick Luby) tdf#129808 FILEOPEN DOC/X: Line spacing narrower than Word due to special handling of font code page range bits (Thanks to Jonathan Clark) tdf#140027 EDITING inserting rows before pivot table in row 1 copies format and images from headings (Thanks to Balazs Varga) tdf#141033 Usability : Base Form Control Property dialog window width is too narrow by default (Thanks to Caolán McNamara) tdf#143182 Format Basic function converts string to number irrespective of format string (Thanks to Mike Kaganski) tdf#44729 PDF import from Inkscape PDF export with opacity shows black background for each opaque object (Thanks to Dr. David Alan Gilbert) tdf#67797 FORMATTING: Creating Manual Numbered List Fails, parses C. as Roman numeral “C” (Comment 11) (Thanks to Justin Luth) tdf#81003 BASIC runtime error. ‘380’ Incorrect property value (

Thanks to Mike Kaganski) tdf#84949 Placing mouse Cursor over Left tab, Center tab, Right tab, Decimal tab icon doesn't show a tooltip (Thanks to Andreas Heinisch) tdf#87058 column label functions do not update after inserting rows below title - needs save and reload including a hard recalc (Thanks to Markus Mohrhard) tdf#94259 FILEOPEN XLS Charts are vertically squashed (Thanks to Markus Mohrhard) tdf#94627 FILEOPEN: CALC do not recognise cell names on .xlsb files (Thanks to Markus Mohrhard) WORKSFORME bugs 42 bugs have been retested by 22 people. Top 10 testers Buovjaga (8) BogdanB (7) Michael (4) m_a_riosv (4) Dieter (2) xordevoreaux (2) Heiko Tietze (2) V Stuart Foote (2) Mauro Calzavara (1) Regina Henschel (1) DUPLICATED bugs 98 bugs have been duplicated by 29 people. Top 10 testers Xisco Faulí (18) BogdanB (11) nobu (10) Mike Kaganski (8) Buovjaga (6) m_a_riosv (6) Alex Thurgood (5) Saburo (5) Jonathan Clark (3) Heiko Tietze (2) Verified bug fixes 13 bugs have been verified by 8 people. Top 10 Verifiers BogdanB (3) Gerald Pfeifer (3 ...

- [Mike Kaganski: A fairy tale about poor UX enforcing vendor lock-in](#) (2025/09/15 08:25)

Once upon a time, there was a girl, who used WhatsApp in her iPhone. She was rather active there, and collected quite some important data in the app over time. But some things in her iPhone were inconvenient; and the phone was slowly aging. So she wanted to change her phone some day. For her birthday, a fairy, who learned somehow about the girl's wish, presented her a new Android phone. That was a nice new phone, and the girl was so happy! She decided to move everything from the old phone to the new one immediately. She was worrying about how to move the precious data between the devices; but she felt a huge relief, when the phone spoke: "The fairy told me how important your data is to you; and I have magic powers to handle it all. Just connect the old phone to me with a cord". So she did. The new phone started its work; and the girl could see how the progress bar was gradually moving to completion; but suddenly it stopped; minutes passed, but the bar was motionless. The girl was impatient to start using her new shiny device, but she knew that she needs to wait. And she waited; and waited; but after an hour passed, she noticed something horrible: the old phone was sucking the life out of the new device through the cable! The scared girl could only hope that the process would resume, and finish before the new phone is out of power. She searched and learned, that iPhones are known for their insatiable hunger, and whenever they are connected to anything with energy, they start sucking it. She couldn't even ask the new phone to shine less brightly to save the energy - because it wasn't ready for such things yet. She used her wireless charger, but its powers were fewer than the hunger of iPhone, combined with the hard work done by Android. The energy level still decreased too fast. In the end, when the hope almost vanished, the progress resumed moving! But immediately, the new phone said: "When I collected your data from your old phone, something bad happened, and I failed to collect something. I will continue, but please check later, what's missing!". Only a couple of energy drops were remaining in the new phone, when it finished its task, and could be disconnected from the vampire. But the girl was terrified, when she opened WhatsApp, connected to it (using a magic SMS confirmation), only to see that all her data is lost! She tried to open WhatsApp on the old phone to check if something is still there, and saw that the app had disconnected her. So she used the SMS magic again, and - to her great relief - everything was there! She asks WhatsApp, how to move the data; and it answered, that if she moved from iPhone to iPhone, or from Android to Android, she could use a backup; but from iPhone to Android, only the Transfer Wizard was supported. So she decided to try again. Long story short, but this time, everything repeated exactly the same. The energy was sucked from the new phone; the wireless charger couldn't fully compensate that; the progress stopped, and then a failure happened; the data wasn't there. This time, when she spelled the SMS magic, she needed to wait some minutes before it worked. It was because the wise powers out there were caring and guarded her from possible villains trying to steal her data, so demanded a delay. The girl was desperate. She was almost ready to throw the new phone away. But after some time, she decided to talk to WhatsApp again. She asked it, what to do, and got the same advice. She explained her problem, but the app was adamant.

And only after a long persuasion, and even some threats, the app told her a secret, that there are third-party paid apps, that can also move her data from phone to phone! Poor girl had no choice, and bought one such app. She launched it, and asked to transfer her data. And the helper app said: "Connect your phone to your old iPhone with a cable!" You can imagine how sad was the girl hearing that. But she did what the app asked; and as she feared, the iPhone started to do what it always did. The progress was painfully slow, as you already guessed. Everything was almost exactly as before. But something changed this time: there was no error! The task took even longer; and when it finished, the new phone almost died; but it finished!!! The heart of the girl was full of happiness. She wanted to open WhatsApp immediately, to know if everything is there! But first, she had to do the SMS magic. She casted the spell ... and the powers replied her, that she has to wait eight hours! I lack the ability to describe her anger, when she heard that. She came through pains, she lost her money, lots of time and nerves – and now she couldn't do the last step just now. The time lasted incredibly slow ... but eventually, she overcame that last obstacle, and was glad to learn, that this time, everything was there. But I hear the demonic laughter of someone, who designed a process, where one insanity was piled upon another: where you can't move the data using normal means; where you use a vampire cables; where error messages don't allow you to fix anything by telling where the problem is; where you have to pay to have your data back (oh no, WhatsApp is not like that ransomware, just the end result is the same); where the security measures aggravate the grief, because they don't account for problems of their own software; and overall, where the app makes its transfer so complicated, that people would rather stay with old vendor, just to not experience that again.

- [LibreOffice Dev Blog: Using C++ STL functions instead of loops – EasyHack](#) (2025/09/11 14:24)

C++ Standard library, which resides in `std::` namespace provides common classes and functions which can be used by developers. Among them, Standard Template Library (STL) provides classes and functions to better manage data through data structures named containers. Here I discuss how to use STL functions for better processing of data, and avoid loops. Checking Conditions To iterate over a container to see if some specific condition is valid for all, any, or none of the elements in that container, C/C++ developers traditionally used loops. On the other hand, since C++11, there are functions that can handle such cases: `all_of`, `any_of` and `none_of`. These functions process STL containers, and can replace loops. If you want to know if a function returns true for all, any, or none of the items of the container, then you can simply use these functions. This is the EasyHack dedicated to such a change: 153109 – Use `all_of`, `any_of` or `none_of` from STD instead of looping to check a condition over a range Here is an example patch which uses `any_of` instead of a loop: - `bool bFound = false; // convert ASCII apostrophe to the typographic one`
`const OUString aText(rOrig.indexOf('\'') > -1 ? rOrig.replace('\'', u'â€™') : rOrig); - size_t nCnt = aVec.size(); - for (size_t i = 0; !bFound && i < nCnt; ++i) - { - if (aVec[i] == aText) - bFound = true; - } + const bool bFound = std::any_of(aVec.begin(), aVec.end(), + [&aText](const OUString& n){ return n == aText; });` As you can see, the new code is more concise, and avoids using loops. Conditional Copying, Removing and Finding If you want to copy, remove or simply find a value in a container which conforms to a specific functions, you may use `copy_if`, `remove_if` or `find_if`. 153294 – Use `copy_if`, `find_if` or `remove_if` to simplify loops Again, this is an example patch: - `for (size_t i = 0; i < SAL_N_ELEMENTS(arrOEMCP); ++i) - if (arrOEMCP[i] == codepage) - return true; - - return false; + return std::find(std::begin(arrOEMCP), std::end(arrOEMCP), codepage) != std::end(arrOEMCP);` Final Words Refactoring code is a good way to improve knowledge on LibreOffice development. The above EasyHacks are among EasyHacks that everyone can try. More information about EasyHacks, and how to start working on them can be found on TDF Wiki: [TDF Wiki – EasyHacks](#)

- [Miklos Vajna: Interdependent tracked changes improvements in Writer, part 3](#) (2025/09/09 06:23)

Writer has some support for interdependent (or hierarchical) tracked changes: e.g. the case when you have a delete on top of an insert. See the

second post for background. This work is primarily for Collabora Online, but the feature is available in desktop Writer as well. Motivation¶ With the already mentioned improvements in place, the area of format redlines with character style or direct formatting changes were still lacking: Writer's original model here was just marking a text range as "formatted" and then either accept the format redline as-is, or reject reverting back to the paragraph style (default formatting), losing the old character style or old direct formatting. Results so far¶ Here is a sample case of a document where the old character style is Strong (~bold) and the font size is 24pt, while the new character style is Quote (~italic) and the font size is 36pt. The rest of the document uses no specific character styles and has the font size of 12pt: Interdependent tracked change: improved format, after document load Rejecting that format redline resulted in just the defaults, i.e. no character style and 12pt font size: Interdependent tracked change: old reject, lost character style / direct format But now we track the old character style & direct format: Interdependent tracked change: new reject, handled character style / direct format This required changes in the DOCX import, ODF import and ODF export, too. How is this implemented?¶ If you would like to know a bit more about how this works, continue reading... :-)

As usual, the high-level problem was addressed by a series of small changes. Core side: sw: document DocumentRedlineManager sw: document IDocumentRedlineAccess sw doc model xml dump: show the item set of a format redline tdf#166319 sw interdependent redlines: fix redo of accept of ins-then-fmt's fmt tdf#166319 sw interdependent redlines: handle deleting a self ins-then-fmt tdf#166319 sw interdependent redlines: fix redo of reject of del-then-fmt's fmt tdf#167194 sw redline reinstate: fix handling of self-inserts tdf#167761 sw format redline: register the item set in the autostyle pool tdf#167761 sw format redline: implement ODF export tdf#167761 sw format redline, char props: implement ODF import tdf#167761 sw format redline, char style: implement DOCX import tdf#167761 sw format redline, char style: implement ODF import tdf#167761 sw format redline, char style: fix missing encode in ODF filter tdf#167761 sw format redline, char style+direct: add ODT export tdf#167761 sw format redline, char style+direct: add ODT import Want to start using this?¶ You can get a development edition of Collabora Online 25.04 and try it out yourself right now: try the development edition. Collabora intends to continue supporting and contributing to LibreOffice, the code is merged so we expect all of this work will be available in TDF's next release too (26.2).

- [Miklos Vajna: Multi-page floating tables in Writer: keep together or not](#) (2025/08/15 06:14)

This post is part of a series to describe how Writer now gets a feature to handle tables that are both floating and span over multiple pages. This work is primarily for Collabora Online, but is useful on the desktop as well. See the 11th post for the previous part. Motivation¶ Previous posts described the hardest part of multi-page floating tables: making sure that text can wrap around them and they can split across pages. In this part, we'll look at a conflicting requirement. On one hand, headings want their text to not split across pages (and shapes anchored into paragraphs are considered part of the paragraph, too). On the other hand, it should be OK to have a floating table at the bottom of a page and the following heading to go to the next page. It turns out, Writer gave "keep together" a priority, while Word gave "floating tables are OK to split to a previous page" a priority. Note that if you have a shape (e.g. a triangle) and not a floating table, then both Word and Writer prevents the move of that shape to a previous page (if the shape is anchored in a heading); this difference was there just for floating tables. Results so far¶ Here is how the tdf#167222 bugdoc looks like now in Writer: Floating table, followed by heading: new Writer render And here is how it used to look like: Floating table, followed by heading: old Writer render And here is the reference rendering: Floating table, followed by heading: reference render This means that we leave layout for shapes unchanged in general: shapes anchored in headings are still considered to be part of headings and don't split. But for floating tables, we now allow them to split and use space at a previous page if they fit there. How is this implemented?¶ If you would like to know a bit more about how this works, continue reading... :-)

As usual, the high-level problem was addressed

by a series of small changes: tdf#167222 sw floatable: allow split of fly and its keep-together anchor text Related: tdf#167222 sw floatable: fix split of fly and its 'keep' anchor text Related: tdf#167222 sw floatable: fix split of fly and its heading text w/ ftn sw: fix assertion failure in SwFrameShell::Execute() Related: tdf#167222 sw floatable: fix split of fly and heading text fit check Want to start using this?¶ You can get a development edition of Collabora Online 25.04 and try it out yourself right now: try the development edition. Collabora intends to continue supporting and contributing to LibreOffice, the code is merged so we expect all of this work will be available in TDF's next release too (26.2).

- [LibreOffice QA Blog: QA/Dev Report: July 2025](#) (2025/08/11 17:04)

General Activities LibreOffice 25.2.5 was announced on July 17 Olivier Hallot (TDF) updated help for CSV import, explained Property Mapping in help for Charts and improved help for Calc's FILTERXML function and AutoFilter Gábor Kelemen (Collabora) did many code cleanups Tomaž Vajngerl (Collabora) made internal hyperlinks in a table of contents accessible when exported to PDF/UA Pranam Lashkari, Szymon Kłos and Hubert Figuière (Collabora) worked on LOKit used by Collabora Online Parth Raiyani (Collabora) did reorganisations in some dialogs Miklós Vajna (Collabora) polished the support for floating tables in Writer, fixed some crashes and continued improving the handling of tracked changes that depend on each other Xisco Faulí (TDF) added Albanian and Moldovan locale, fixed short weekdays in Romanian locale, improved the translation checker script, added some new automated tests, upgraded many dependencies and did many code cleanups and optimisations Michael Stahl (Collabora) fixed an issue with expansion of list level numbering formats with repeated levels and fixed a column width issue in RTF tables Mike Kaganski (Collabora) implemented Markdown export, fixed not being able to apply colour to Chart walls via Sidebar, fixed an issue with paragraph numbering in RTF files, helped Miklós with floating table polishing, fixed an issue with date conversion in Base, made URL handling more robust in Extension updating code, fixed an issue with spacing in lists in RTF files, fixed RTF export issues causing loss of bullet fonts and "No character border" explicit formatting and fixed some crashes. He also did many code cleanups and optimisations Caolán McNamara (Collabora) fixed many issues found by static analysers and did code cleanups and optimisations Stephan Bergmann (Collabora) worked on the WASM build. He also adapted the code to compiler changes and did code cleanups Noel Grandin (Collabora) made Skia rendering backend mandatory on Windows and greatly improved the import time of CSV data with trailing newline characters. He also did many code cleanups and optimisations, especially in the area of transparency handling Justin Luth (Collabora) made it so failed command line operations return exit status 1, allowing for automated bisecting of command line issues among other things, fixed an issue with spellchecking and the option "Check uppercase words" and fixed a style continuity issue with page breaks in DOCX files Michael Weghorn (TDF) continued cleaning up and reorganising accessibility-related code, made the orientation radio buttons in Envelope dialog accessible, fixed an issue with unwanted focus accessibility events being fired in Borders tab page of Writer's Paragraph dialog, made the border preset selection be clearly indicated when focused, implemented support for native colour pickers in GTK and Qt UIs and did cleanups and reorganisations in Android, vcl and report design code. He also worked on using native widgets in Qt UIs Balázs Varga (Collabora) implemented support for Microsoft Media Foundation APIs on Windows for playback of common codecs, fixed Calc's MATCH function returning an incorrect result with inline arrays and fixed an issue with missing graphic bullets upon copying & pasting Patrick Luby reduced memory consumption related to font handling on macOS, fixed an issue preventing moving animated GIFs with arrow keys on macOS and fixed unwanted slowdown of GIF animation Christian Lohmaier (TDF) did some build fixes Jonathan Clark (TDF) improved font rendering on Windows, reorganised icon themes to support "right-to-left, top-to-bottom" writing direction variant, greatly improved the opening time of certain long RTL DOC files and added support for DOC/DOCX script type hinting Andreas Heinisch made it so expanded text is included in the tooltip of index fields Julien Nabet did some code cleanups and fixed issues found by static

analysers Bayram Çiçek (Collabora) made field shadings adjust according to line spacing in order to prevent letters from being cut off Heiko Tietze (TDF) improved dark mode support and added an option for using horizontal tabs instead of vertical ones Kurt Nordback (Collabora) implemented exporting of chart style index to OOXML documents and continued working on support for recent MSO chart types Sahil Gautam (Collabora) continued polishing the Libreoffice Theme rework Armin Le Grand (Collabora) worked on EditEngine rendering revamp Arnaud Versini did some code cleanups and optimisations Khaled Hosny (Alif Type) optimised the performance of calculating text bounds Regina Henschel fixed an issue with Calc's LOOKUP function when using an inline array as result vector David Gilbert fixed PDF import issues with stroke opacity and clipping and opacity during axial fills Ujjawal Kumar worked on the Markdown import GSoC project Ilmari Lauhakangas (TDF) added an "unpublished" decoration to the SDK documentation to help API users avoid methods marked as such and improved the placement of other decorations Gülşah Köse (Collabora) fixed styles going missing in change-tracked DOCX files Marc Mondesir did improvements in PDF export automated tests Pierre Vacher improved the robustness of Base and its table creation wizard Kohei Yoshida (Collabora) added an alternative CSV filter based on Orcus library Manish Mahapatra fixed indentation of headings when pasted as unformatted text Markus Mohrhard fixed issues with charts, Calc's AutoFilter and made handling of OOXML files with invalid cell references more robust Hossein Nourikhah (TDF) fix the display of JDK version number and path in RTL UI Jean-Pierre Ledure worked on the ScriptForge library Bojidar Marinov unified the options for switching to form design mode Floris Bos added an option to disable external links in PDF export Samuel Mehrbrodt (Collabora) made it so Insert Hyperlink is shown in context menu when text is selected Buo-ren Lin (OSSII) fixed source fontwork object not being removed after being cut Adam Seskunas worked on the C++ Report Builder GSoC project Áron Budea (Collabora) made it possible to hide tracked changes in PDF export from the command line Chris Sherlock did code cleanups and refactoring in VCL toolkit Kudos to Ilmari Lauhakangas for helping to elaborate this list. Reported Bugs 429 bugs, 56 of which are enhancements, have been reported by 287 people. Top 10 Reporters Mike Kaganski (15) Gabor Kelemen (allotropia) (15) Telesto (14) nobu (12) Eyal Rozenberg (12) Justin L (9) Michael Weghorn (7) Timur (7) wodsfort (5) Pierre Fortin (4) Triaged Bugs 314 bugs have been triaged by 70 people. Top 10 Triagers m_a_riosv (48) Buovjaga (24) V Stuart Foote (24) Mike Kaganski (22) Heiko Tietze (20) nobu (13) raal (13) Saburo (13) Michael Weghorn (10) Xisco Fauli (9) Resolution of resolved bugs 268 bugs have been set to RESOLVED. Check the following sections for more information about bugs resolved as FIXED, WORKSFORME and DUPLICATE. Fixed Bugs 142 bugs have been fixed by 35 people. Top 10 Fixers Mike Kaganski (18) Noel Grandin (9) Jonathan Clark (8) Markus Mohrhard (7) Xisco Fauli (6) Balazs Varga (5) Heiko Tietze (4) Olivier Hallot (3) Justin Luth (3) Patrick Luby (3) List of high severity bugs fixed tdf#148117 FILESAVE: PPTX: date in chart changes in MSO (Thanks to Markus Mohrhard) tdf#159529 Excessive memory consumption in v24.2 (Thanks to Patrick Luby) tdf#166813 installation crashing when using georgian text (Thanks to Xisco Fauli) tdf#167466 Entering anything into the search box under Tools / Options causes LibreOffice 25.8.0.1 to instantly crash. (Thanks to Xisco Fauli) List of crashes fixed tdf#166813 installation crashing when using georgian text (Thanks to Xisco Fauli) tdf#167466 Entering anything into the search box under Tools / Options causes LibreOffice 25.8.0.1 to instantly crash. (Thanks to Xisco Fauli) tdf#167599 Crash in Object Inspector when switching from Properties to Services for ContentControls (Thanks to Mike Kaganski) tdf#167633 LibreOffice debug build crashed on launch when a Java installation with a non-ASCII path is set during configure(rtl_uString_newFromAscii(rtl_uString**, const char*): Assertion `static_cast(*pCharStr) < 0x80' failed.) (Thanks to Buo-ren Lin (OSSII)) List of performance issues fixed tdf#117636 Multi-page block of Chinese glyphs slows down file opening and navigation (Thanks to Jonathan Clark) tdf#64991 Opening a long RTL DOC file is extremely slow, while ok if resaved as DOCX in MS-Word (Thanks to Jonathan Clark) List of old bugs (more than 4 years old) fixed tdf#117636 Multi-page block of Chinese glyphs slows down file opening and navigation (Thanks to Jonathan

Clark) tdf#118619 WEBSERVICE(URI) / FILTERXML Xpath not well explained (Thanks to Olivier Hallot) tdf#118666 Headings are indented when pasting as unformatted text (Thanks to Manish) tdf#118668 FILEOPEN: LibreCalc doesn't show all sheets in XLSX modified with exceljs (Thanks to Markus Mohrhard) tdf#122336 Probably generated XLSX without DocProps folder opens empty (If resaved in MS Office, opens fine in LO) (Thanks to Markus Mohrhard) tdf#136753 Use MDL (Moldovan Lei) along with other currencies (Thanks to Xisco Fauli) tdf#136754 FILEOPEN: CHART: incorrect background (Thanks to Markus Mohrhard) tdf#139834 [UI] String breaks translation gender (Thanks to Olivier Hallot) tdf#143157 Alphabetical index field: on mouse-over include keys in tooltips (Thanks to Andreas Heinisch) tdf#58511 EDITING: Cut fontwork object does not work (Thanks to Buo-ren Lin (OSSII)) tdf#62408 MP4 videos aren't handled in Windows (without additional codecs) (Thanks to Balazs Varga) tdf#64991 Opening a long RTL DOC file is extremely slow, while ok if resaved as DOCX in MS-Word (Thanks to Jonathan Clark) tdf#70102 RTL: Flipped toolbar button icons dont always look suitable (Thanks to Jonathan Clark) tdf#80662 RTL: The update number of the java version number in Options dialog is on the wrong side (Thanks to Hossein) tdf#96151 Please add Albania's Currency, Locale Setting, and Albanian Language (Thanks to Xisco Fauli) WORKFORME bugs 32 bugs have been retested by 21 people. Top 10 testers Buovjaga (5) Regina Henschel (3) Dieter (2) BogdanB (2) alecrkerr (2) Julien Nabet (2) V Stuart Foote (2) Justin L (2) Wolfgang Jäger (1) xordevoreaux (1) DUPLICATED bugs 40 bugs have been duplicated by 19 people. Top 10 testers nobu (7) Buovjaga (5) m_a_riosv (4) V Stuart Foote (4) Mike Kaganski (3) Xisco Faulí (3) Heiko Tietze (2) Robert Großkopf (2) Markus Mohrhard (1) Timur (1) Verified bug fixes 14 bugs have been verified by 10 people. Top 10 Verifiers Eyal Rozenberg (2) Telesto (2) m_a_riosv (2) Michael Weghorn (2) BogdanB (1) lol (1) Patrick (volunteer) (1) Hossein (1) Tekstryder (1) 林博仁 Buo-ren Lin (1) Categorized Bugs 193 bugs have been categorized with a metabug by 24 people. Top 10 Categorizers BogdanB (31) Roman Kuznetsov (30) Eyal Rozenberg (21) V Stuart Foote (19) Michael Weghorn (11) Heiko Tietze (11) Jeff Fortin Tam (10) Telesto (10) Timur (7) Saburo (7) Regression Bugs 43 bugs have been set as regressions by 18 people. Top 10 Buovjaga (6) nobu (6) Saburo (4) Michael Weghorn (4) raal (3) Xisco Faulí (3) Telesto (3) Robert Großkopf (2) m_a_riosv (2) Timur (2) Bisected Bugs 52 bugs have been bisected by 10 people. Top 10 Bisectors Saburo (26) Xisco Faulí (6) Michael Weghorn (4) Buovjaga (3) raal (3) Gabor Kelemen (allotropia) (2) Telesto (2) Aron Budea (2) jussuom (2) Timur (...

- [Mike Kaganski: Microsoft, anybody home?](#) (2025/07/25 06:39)

You know what: Microsoft became miserably incompetent in IT. I develop open-source code. But that never made me one of the "I hate proprietary software or IT giant corporations" types. I always saw the nice things that Microsoft offered to its users; I saw not only downsides in its products. And I also used (and continue to use) things created by it: Windows to start with (and I develop there, being able to debug and address issues specific to the platform that most of our users use); but also its email service for personal mail. This Monday, I decided to send something to LibreOffice dev mailing list. Something I do from time to time, you know. Not too fascinating, right? Well, this time, it turned out, Microsoft decided to teach me to fear them. Thunderbird shown me a message, that the mail couldn't be sent (well, not a problem: will re-try again...), but then I found myself logged off, with "Your account has been blocked" message. They decided, that I violated their service agreement! FTR: here is the mail. I was able to send it using another tech giant's mail service. You may see that it's full of links. Yes, that's true; I prefer to provide references to my words. But tell me where was it violating anything in MS agreement? OK, they have a stupid AI that is worse than good old filters. OK, they made it react immediately, as an undoubted authority. But that's not a big problem, right? They provide a way to appeal! Let me do that. And of course, they ask for the phone, and I provide it, just to get a nice reply: And guess what: there is no other method! OK! Let's ask their support. (I am approaching to the point that fascinated me most.) I found a link to "Contact Microsoft Support" on the

“Troubleshooting verification code issues” page; and after some automatic answers there, which didn’t answer my problem, I finally got a button telling me ... tada ... Yes, you got it right. “Here is a page where we discuss problems signing in. You attempted our FAQ suggestions? You still can’t sign in? No problem! Contact our Support team, and we will solve your problem in a minute! But first, please sign in to continue.” Heh. I used my wife’s account to contact support. And then I was given a very secret link to an appeal form, where I could file a support ticket. And the next morning, I got a message! Yay! It told me to do something! Let me try! What is that they tell me to do? Reading... hmm... go to sign-in page, and when they tell me that my account is blocked, provide a phone number? Wasn’t it exactly the thing I attempted and failed, and told them about that? But hey, they obviously fixed that problem overnight, they couldn’t just send me the useless instructions, right? Or could they? They could. They just ignored my description. They repeated the useless instructions, without taking care to check what the problem was. And they closed the ticket automatically. It has been resolved, you know. I still am in the process. I filed another ticket, which they didn’t care to answer. I am still hopeful. But this, once an IT tech leader, became utterly incompetent in IT. And for me, it’s a pity.

- [allotropia: Collabora and allotropia merge](#) (2025/05/28 10:20)

This deal unites the largest team of corporate Office engineers to deliver on Collabora Productivity’s mission to restore Digital Sovereignty to its users, while making Open Source Office Rock. It supercharges Collabora’s Online Office products and services portfolio with rich German language capability, deeper experience of vertical applications, new Web Assembly skills, and a wider unified partner ecosystem. Through improved product richness this sharpens the competitive edge of FLOSS Office productivity against mass-market proprietary alternatives.

CAMBRIDGE, UK – May 28th 12:00 CEST – 2025 Collabora Productivity, the world’s leading provider of collaborative Open Source Office editors have completed a merger with allotropia. Collabora has invested heavily in building Collabora Online (COOL) – a market leading, on-premise, secure, interoperable, open-source solution for document editing and collaboration deployed to any modern browser. This is complemented by desktop and mobile apps across Linux, Windows, Mac, Android, iOS and Chrome-OS. Collabora provides support subscriptions to enterprise customers worldwide via a network of hundreds of trusted partners. This is now augmented by allotropia’s partner and customer base. Together with our partners we deliver document and productivity excellence integrated with our partners product and service offerings. allotropia’s expertise around Web Assembly combined with Collabora Online will we expect, in time, enable customer use-cases such as well as office-as-component embedding scenarios in vertical applications as well as off-line and end-to-end encrypted editing, and. This work builds on some visionary prototype funding from the Bundesministerium des Inneren (BMI) for a collaboration between the companies to enable the use of Collabora Online off-line in the browser. Further details of product investment, and direction will be announced and decided in workshops with our key customers and partners at our annual COOL Days conference in Budapest next week where staff, community and our customer and partner-ecosystem meet, swap ideas, and hear about the latest work in our upcoming major release featuring improved performance, usability, interoperability and much more. “Collabora is excited to welcome each member of the allotropia team today!” said Michael Meeks, CEO, Collabora Productivity, “We are excited to work together to accelerate our product development, enjoy our first COOL Days together, and plan the next features and possibilities to delight our customers.” Collabora has invested in building a network of hundreds of partners and is approaching one hundred million docker image downloads of its document editing server software, with millions of paying users of its products, all of whom will start to benefit from this merger from today. We expect to bring the experience that allotropia has from its relationship with CIB around vertical desktop applications (Fachverfahren) to help partners and customers migrate their Windows & Microsoft Office based business process to easy to deploy multi-platform web applications. “With our awesome team of engineers, and our WebAssembly know how, we can add

significantly to Collabora's powerhouse of Office engineering prowess & their product offerings", says Thorsten Behrens, CEO of allotropia, "we've worked with them as partners for many years, and align perfectly in our goals to make Open Source office rock!" allotropia's skills in supporting and contributing to the LibreOffice code-base in Germany strengthens and unifies popular shared partner products such CIB Office and Nextcloud Office. A larger team will accelerate development and improvement of Collabora Office based products, while providing an even deeper pool of support resources to rapidly respond to customers' needs. Together we want to pay tribute to the vast legacy of those who have worked so hard to preserve and improve the source code that we depend on from Sun Microsystems, Oracle, SUSE, RedHat, IBM, TDF, Canonical, and many more, as well as the innumerable volunteer community contributors who make the Collabora Online and LibreOffice ecosystem so rich and interesting: thank you allowing us the privilege of working alongside you as we revolutionize the office productivity world together. All of our code is open source and available to the public on GitHub. Join the Collabora Online Community, take part in easy hacks and discussions in the forum. Please also see our new parent company's mirror announcement!

- [Mike Kaganski: How could QA catch this in advance?](#) (2025/05/26 09:59)

Yesterday I merged a fix for Writer's tdf#165094. Not that it was something exceptional; something that often happens when we change the huge code: a regression. Something that we try to do for them: a fix. Why mention it here? It happens to show something, that people underestimate. The complexity of what they call "proper testing" – you know, that "I found a bug! Do you even try to test your software???" rant you often see in discussions. Let's look at this case. The problem was, that in some specific document, where there was a manually inserted page break, that page break, defined in a hidden paragraph, disappeared after an upgrade. Sounds easy? Should be caught immediately in the release testing? But other page breaks weren't lost. Debugging showed, that the bug would only occur when all of the following happened: The page break was defined in a hidden paragraph (something already known from the reporter – thank you Gabor!), and There were at least 26 paragraphs before that hidden paragraph, all on the same page, and The page break defined a paragraph style, and That page break defined a page number, and That assigned new page number happened to be the same "oddity" as the current one (i.e., either the number of that page with 26+ paragraphs was odd, and the new page number was odd; or the number of that page with 26+ paragraphs was even, and the new page number was even), and After the hidden paragraph (which defined the page break), a table immediately followed. I suppose, that's a combination of factors, that any QA engineer would naturally test first, don't you agree? (Disclaimer: no I don't think so.) Note that the complexity of this constellation of causing factors is, again, not uncommon in our codebase. In fact, it only needed less than ten features to take their specific forms, from thousands of features and options that the suite offers. But it is completely unsurprising, that the bug, that requires such a constellation of factors, actually appeared in our bug tracker. Given the tens of millions of users, who work with who knows how many documents, every low-probability event will happen, sooner or later. This is good; and we are thankful to everyone who files bugs. And let me say, that we at Collabora Productivity are glad to do many good things to make the office suite better for everyone.

- [Marius Popa Adrian: FirebirdSQL introduces support for Windows ARM64 builds](#) (2025/04/30 11:36)

This FirebirdSQL pull request introduces support for Windows ARM64 builds to the Firebird project. The changes cover updates to build scripts, configuration files, and Visual Studio solution/project files to accommodate ARM64 architecture, ensuring compatibility and enabling compilation and functionality on Windows ARM64 platforms.

- [Marius Popa Adrian: SQL-compliant aliases GREATEST and LEAST for the existing MAXVALUE and MINVALUE functions.](#) (2025/04/30 07:24)

This FirebirdSQL pull request introduces SQL-compliant aliases GREATEST and LEAST for the existing MAXVALUE and MINVALUE functions. These

aliases align with the SQL:2023 standard and provide a more intuitive and widely recognized syntax. The changes include updates to documentation, keywords, parser tokens, and system function definitions to support these new aliases.

- [Ravi Dwivedi: Libreoffice Conference 2024 in Luxembourg](#) (2025/03/14 16:18)

Last year, I attended the annual LibreOffice Conference in Luxembourg with the help of a generous travel grant by The Document Foundation (TDF). It was a three-day event from the 10th to the 12th of October 2024, with an additional day for community meetup on the 9th. Luxembourg is a small country in Western Europe. It is insanely wealthy with high living standards. After going through an arduous visa process, I got to the country on the 8th of October. Upon arriving in Luxembourg, I took a bus to the city center, where my hotel — Park Inn — was located. I deboarded the bus at the Luxembourg Central station. Before walking towards my hotel, I stopped to click a few pictures of the beautiful station. All the public transport in Luxembourg was free of cost. The experience of being in Luxembourg was as if I had stepped in another world. The roads had separate tracks for cycling and separate lanes for buses, along with wide footpaths. In addition, the streets were pretty neat and clean. Luxembourg's Findel Airport. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. Separate cycling tracks in Luxembourg. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. A random road in Luxembourg with separate lane for buses. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. The conference venue was in Belval, while I stayed in the city center. Even though my stay was 20 km from the conference venue, the commute was convenient thanks to free of cost train connections. The train rides were comfortable, smooth, and scenic, covering the distance in half an hour. Moreover, I never found the trains to be very crowded, which enabled me to always get a seat. This is what trains look like in Luxembourg. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. The train ride from my hotel to the conference venue had some scenic views like this one on the way. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. A tram in Luxembourg with Luxembourg Central station in the background. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. My breakfast was included in the hotel booking. The breakfast had many options. It had coffee and fruit juices, along with diverse food options. Some of the items I remember were croissant, pain au chocolat, brie (a type of cheese), scrambled eggs, boiled eggs, and various types of meat dishes. Other than this, there were fruits such as pears. That circular pie in the center of the image is brie - a type of cheese - which I found delicious. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. Pre-conference, a day was reserved for the community meetup on the 9th of October. On that day, the community members introduced themselves and their contributions to the LibreOffice project. It acted as a brainstorming session. All the attendees got a lovely conference bag, which contained a T-Shirt, a pen and a few stickers. I also met my long time collaborators Mike, Sophie and Italo from the TDF, whom I had interacted only remotely till then. Likewise, I also met TDF's sysadmin Guilhem, who I interacted before regarding setting up my LibreOffice mirror. Lovely swag bag. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. The conference started on the 10th. There were 5 attendees from India, including me, while most of the attendees were from Europe. The talks were in English. One of the talks that stood out for me was about Luxchat — a chat service run by the Luxembourg government based on the Matrix protocol for the citizens of Luxembourg. I also liked Italo's talk on why document formats must be freedom-respecting. On the first night, the conference took us to a nice dinner in a restaurant. It offered one more way to socialize with other attendees and explore food at the same time. A slide from Italo's talk on document freedom. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. Picture of the hall in which talks were held. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. On the 11th of October, I went for a walk in the morning with Biswadeep for some sightseeing around our hotel area. As a consequence, I missed the group photo of the conference, which I wanted to be in. Anyway, we enjoyed roaming around the picturesque Luxembourg city. We also sampled a tram ride to return to our hotel. We encountered such scenic views during our walk. Photo by Ravi Dwivedi. Released under the

CC-BY-SA 4.0. Another view of Luxembourg city area. Photo by Ravi Dwivedi. Released under the CC-BY-SA 4.0. The conference ended on the 12th with a couple of talks. This conference gave me an opportunity to meet the global LibreOffice community, connect and share ideas. It also gave me a peek into the country of Luxembourg and its people, where I had good experience. English was widely known, and I had no issues getting by. Thanks to all the organizers and sponsors of the conference!

- [allotropia: ZetaJS: Combining Writer & Calc](#) (2025/03/06 10:30)

Weâ€™ve added a great new Vue.js-3 ZetaJS demo (source)! It showcases word processing and spreadsheets inside a single web app. Calc is being used as a data source for an HTML app, filling letter templates in Writer. You can even upload custom data spreadsheets or document templates! And have you seen the nice Writer toolbar, all done with Vue.js? Weâ€™ve also updated the existing demos, showcasing Chrome PWA support with the Ping Monitor demo – just click the little install button at the top-right of the address bar, to get the Ping Monitor â€œinstalledâ€” on your desktop! Talks Meanwhile, our team was giving some great talks about our work for ZetaOffice and LibreOffice. Why not check out the recordings during your lunch break? ZetaJS & ZetaOffice Moritz Duge – OSXP Paris: LibreOffice as Web Component – moving customized office workflows into the browser Moritz Duge – 38C3 CCC congress: LibreOffice WASM & JS – Blending a C++ FOSS into a web app Stephan Bergmann: LOWA, In Need Of a VCL Plug (description) Thorsten Behrens – Distributed real-time collaboration for Writer – a first prototype (description) FOSDEM LibreOffice DevRoom talks Balazs Varga – Introducing Glow Effect for texts in shapes (description) Gabor Kelemen – Testing the QA instructions (description) Sarper Akdemir – LibreOfficeâ€™s Python API: Working around limitations of the Pythonic approach (description) Gabor Kelemen – Exploring the deprecated parts of LibreOffice API (description) News clippings Look, we made some headlines! TheRegister was following up some earlier coverage about the WebAssembly port, after Thorsten gave Liam a demo during FOSDEM. Read up the full article here. Next up In case youâ€™re around, meet us in two weeks at the FOSSAsia Summit in Bangkok, where Sarper Akdemir will give an update over our work. Dates are March 13-15. If youâ€™re based in Europe, you might instead enjoy Thorstenâ€™s talk at the Chemnitz Linux Days (Germany) from March 22-23. Looking forward to meet you there! Feedback appreciated! Please subscribe to our Newsletter or on Mastodon and let us know how you liked ZetaJS and the demos! If youâ€™re playing with the code leave a star at the ZetaJS repo or if you hit any issues please file a report on GitHub. Or just leave a comment and let us know directly – thanks for reading!

- [LibreOffice Design Blog: New Templates For You – Your Feedback Matters!](#) (2025/03/03 13:33)

By Ndidi Folasade Ogboi For the past two months, Iâ€™ve been working on adding more templates to LibreOffice Writer as part of my Outreachy project. My goal has been to create functional templates that users need the most. I created these templates based on what you told us in our survey and your response was incredible!...

- [LibreOffice Design Blog: Results from a survey about Writer templates](#) (2025/01/13 10:06)

By Ndidi Folasade Ogboi LibreOffice Writer has long been a trusted tool for users worldwide, offering an open-source solution for documents. But what happens when we take a step back and look at the user experience? How do templates fit into the workflows of users, what makes a great template and where do users want LibreOffice writer to improve?...

- [LibreOffice Design Blog: LibreOffice Themes will replace the color customization](#) (2024/12/20 12:55)

Since the first implementation of a dark color theme we continuously improved the customization of LibreOffice. In a GSoC projects this year, Sahil Gautam made it possible to not only change the application colors but also what is defined by the operating system respectively the desktop environment....

- [Chris Sherlock: The mess that is the VCL](#) (2024/11/29 22:58)

Let me count the ways, in no particular order and in no way exhaustive: `OutputDevice` is the base class for printing, windowing and PDFs. It doesn't just do output. `OutputDevice` has `GetOutDevType()` because the base class needs to know what child class is using it. Ugh. `OutputDevice` drawing primitives not only draw, but they record a metafile. There are literally functions that turn off drawing and just let it record the metafile. I made an attempt at separating the concerns, but it got nowhere. VCL relies on `DrawingLayer` and `DrawingLayer` relies on the VCL. There is a concept of a `VirtualDevice`, which is derived from `OutputDevice`. `VirtualDevice` does a bunch of things, but one of which is alpha-handling. In `OutputDevice`, there is a member which is a `VirtualDevice`. Each drawing function in `OutputDevice` calls upon the correlated drawing function in this member `VirtualDevice`. Bitmaps don't get modified via the `Bitmap` class. Instead, you have to use `BitmapInfoAccess`, `BitmapReadAccess` and `BitmapWriteAccess`. I'm still puzzling out why these are separate classes. Bitmaps are transformed in `SalGraphics` indirectly via `OutputDevice`. Except when they aren't, in which case it fails, whereby `OutputDevice` tries an alternative way via `SalGraphics`. Otherwise, it tries its own poor man approach at drawing the bitmap. Consequently, often times you bypass the platform optimized ways of doing things, because its not been implemented. Fonts are lazy loaded from `OutputDevice`. There is no central font manager. To get the fonts, you have to go through `SalGraphics`. To get a `SalGraphics`, you need to initialize a lot of stuff not related to fonts. Font caching is done from `OutputDevice`. Lazily. Font data is updated for all frames. Frames are a concept needed for Windows. Frames are not a concept needed by Printers and `VirtualDevices`, or even PDFs. Note that Printers, `VirtualDevices` and PDFs all inherit from `OutputDevice`. `OutputDevice` converts between "logical" units and display units. It's a nightmare to know what each function needs what sort of units. For the mapping between units, I refer you to `vcl/source/gdi/mapmod.cxx` and `vcl/source/outdev/map.cxx`. There is `tools` and `basegfx`. They do the same thing, though `basegfx` is considerably better written. You have `Size` and `B2DSize`, `Point` and `B2DPoint`, `Polygon` and `B2DPolygon`, `PolyPolygon` and `B2DPolyPolygon`. `OutputDevice` must handle it all. Gradient handling is sort of half baked in `OutputDevice`, much of gradient handling is done in other modules. Font substitution is truly, truly weird. `PhysicalFontSelect::FindFontFamilyByAttributes()` has clearly got a bug in it - (e.g. `ImplFontAttrs::None == ((nSearchType ^ nMatchType) & ImplFontAttrs::Rounded)` an XOR?) and it is a truly strange weighting scheme. Yes, I did try to untangle that beast with proper unit tests, but gave up after being told I was being unreasonable. There is VCL, canvas, `cppcanvas` and `drawinglayer`. `drawinglayer` is way better than VCL, but we are stuck with VCL for everything. Consider the following Window hierarchy: `WorkWindow` inherits from `SystemWindow`, which inherits from `Window`. `Window` holds an `OutputDevice` to do stuff. `WindowOutputDevice` derives from `OutputDevice`. This is needed because `OutputDevice` often needs to know if it is doing Window operations, via `WindowOutputDevice`. Try untangling this in your head. Text layout is its own beast, and has its own set of classes. A lot of text layout is worked out in `OutputDevice`. Text layout is done via `OutputDevice::ImplLayout()`. I present to you the `ImplLayout` function signature: `std::unique_ptr<SalLayout> ImplLayout(const OUString&, sal_Int32 nIndex, sal_Int32 nLen, const Point& rLogicPos = Point(0, 0), tools::Long nLogicWidth = 0, KernArraySpan aKernArray = KernArraySpan(), std::span<const sal_Bool> pKashidaArray = {}, SalLayoutFlags flags = SalLayoutFlags::NONE, vcl::text::TextLayoutCache const* = nullptr, const SalLayoutGlyphs* pGlyphs = nullptr, std::optional<sal_Int32> nDrawOriginCluster = std::nullopt, std::optional<sal_Int32> nDrawMinCharPos = std::nullopt, std::optional<sal_Int32> nDrawEndCharPos = std::nullopt) const;`

From:
<https://wiki.tromjaro.alexio.tf/> - **TROMjaro wiki**

Permanent link:
<https://wiki.tromjaro.alexio.tf/doku.php?id=news:planet:documentfoundation&rev=1583651701>

Last update: **2021/10/30 11:38**

