Lightning talks

TIRANA | Sept. 28th, 2018
Speakers

- Lázsló: LibreOffice and eTwinning
- Caolan: oss-fuzz, crash testing & coverity
- Muhammet: Customize dialog news
- Thorsten: Why is doing dev metrics hard
- Moggi: UI testing in LibreOffice
- Guilhem: the status.documentfoundation.org website
- Cathy: documentation & LibreOffice
- Gabor: a brief look at chart bugfixing
- Armin: system-dependent data buffering
- Gabriele: TBD
- Franklin: What "Designing with LibreOffice" taught me
- Mark: Vertical Ruby Support in LibreOffice
- Miklos: vcl layout speedup
Coverity, Crashtesting and Fuzzing

• Coverity Stats
• Crashtesting Stats
• OSS-Fuzz Stats
Coverity Stats 2017

Oct 05, 2017
Last Analyzed

6,309,197
Lines of Code Analyzed

6,275,320
Lines of Code in Selected Components

0.00
Defect Density

Defect changes since previous build dated Sep 30, 2017

8
Newly detected

6
Eliminated

Defects by status for current build

25,046
Total defects

12
Outstanding

338
Dismissed

24,696
Fixed
Coverity Stats 2018

- **Sep 22, 2018**
  - Last Analyzed
- **6,444,225**
  - Lines of Code Analyzed
- **6,424,737**
  - Lines of Code in Selected Components
- **0.00**
  - Defect Density

Defect changes since previous build dated Sep 21, 2018

- **0**
  - Newly detected
- **2**
  - Eliminated

Defects by status for current build

- **25,430**
  - Total defects
- **5**
  - Outstanding
- **349**
  - Dismissed
- **25,076**
  - Fixed
Coverity Stats 2017

C++11 Exceptions
CrashTesting

- 108480 (up from 98531) documents
CrashTesting Export

![Graph showing failures with builds]

Pesky block, Webdav?
Google OSS-Fuzz
Our current status

• libfuzzer+asan, libfuzzer+ubsan, and afl+asan
  • Just libfuzzer+asan last year
• 45 fuzzer (up from 34 fuzzers)
A year of oss-fuzz

719 bugs, of which 149 serious
• Goals
• Current State
• Since Last Year
• What’s Left
Goals

- Modernize to...
  - ...make it more usable
  - ...make it fully accessible
  - ...make it intuitive
  - ...eliminate bugs
Current State

• Finished tabs:
  - Menus
  - Toolbars
  - Context Menus
• Search feature in the keyboard tab
• Some annoying bugs fixed
Customize Dialog Box

- Menus Tab
  - Search: Type to search
  - Category: All commands
  - Function: 100%, 150%, 200%, 3D Color, 50%, 75%
  - Description: Displays the document at its actual size.

- Scope Tab
  - LibreOfficeDev Writer

- Target Tab
  - File

- Function Tab
  - New
  - Open...
  - Open Remote...
  - Recent Documents
  - Close
  - Wizards
  - Templates
  - Reload
  - Versions...
  - Save
  - Save Remote

- Customize Button
  - Insert
  - Modify
  - Defaults

- Buttons
  - OK
  - Cancel
  - Reset
Since Last Year

• Some fixed bugs & added features
  - tdf#108916: uno:LanguageStatus shouldn't appear in the customize dialog
  - tdf#112207: Allow assigning styles to ui elements
  - tdf#112207: Allow assigning macros to ui elements
  - tdf#114260: Add gear button to Customize dialog
  - tdf#114260: Allow moving top-level menus
  - tdf#115511: Make the description field keyboard-accessible
  - tdf#116494: Don't ask parent if no help text
  - tdf#114260: Allow setting toolbar style in the gear menu
  - tdf#112323: Sort the category list alphabetically in the Customize dialog
  - Add UI test for the Customize dialog
- Revamp the remaining tabs
  - Keyboard
  - Events
- Fix existing bugs on the meta bug
- Tdf#103238 – Customize Dialog Meta Bug
Thank you!
WHY IS DOING SOFTWARE DEVELOPMENT METRICS HARD?

THORSTEN BEHRENS
LIBREOFFICE CONFERENCE TIRANA
When you can measure what you are speaking about, and express it in numbers, you know something about it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely, in your thoughts advanced to the stage of science.

Lord Kelvin
In truth, a good case could be made that if your knowledge is meagre and unsatisfactory, the last thing in the world you should do is to make measurements. The chance is negligible that you will measure the right thing accidentally.

George Miller
Some advice (from others)

• Track meaningful work
• Track trends, not numbers
• Use metrics that lead to positive change
  – Avoid code churn
  – MTBF, MTTR
  – Award good behaviour: bugfixing, mentoring, reviews
Markus Mohrhard
UI testing in LibreOffice
UI testing in LibreOffice

Markus Mohrhard

2018-09-28
• make uicheck

• Can execute single test manually

• https://cgit.freedesktop.org/libreoffice/contrib/dev-tools/tree/uitest/execute.sh

• https://wiki.documentfoundation.org/Development/UITests
Logging and replaying

- Set `LO_COLLECT_UIINFO=test.log`

- Log file generated under `instdir/uitest/test.log`

- `uitest/loginterpreter.py instdir/uitest/test.log simple_test.py`

- Work by Saurav Chirania as part of GSoC
Thank you …

…it for listening!
Guilhem Moulin:

https://status.documentfoundation.org
Cathy Crumbley
Documentation & LibreOffice
LibreOffice bug fixing

in ooxml import/export of charts

Ádám Kovács and Balázs Varga

Junior developers

NISZ Inc.
tdf#108064: BEFORE the bugfix have done

the original file

after saving the file with LO to docx, xlsx
tdf#108064: AFTER the bugfix have done

the original file after saving the file with LO to docx, xlsx
The point was copy-pasting the `getLineDash` function:

```cpp
...from the file ChartLinePanel.cxx to drawingml.cxx
```
Calling `mAny.get<drawing::LineDash>();` was not successful in the case of charts, so if the result has 0 values, we call the „new” `getLineDash(...)` function:

```cpp
switch (aLineStyle)
{
    case drawing::LineStyle_NONE:
        bNoFill = true;
        break;
    case drawing::LineStyle_DASH:
        if (GetProperty(rXPropSet, "LineDash"))
        {
            aLineDash = mAny.get<drawing::LineDash>();
            if (aLineDash.Dots == 0 && aLineDash.DotLen == 0 && aLineDash.Dashes == 0
                && aLineDash.DashLen == 0 && aLineDash.Distance == 0)
            {
                OUString aLineDashName;
                GET(aLineDashName, LineDashName);
                if (!aLineDashName.isEmpty())
                {
                    if (xModel) {
                        css::uno::Any aAny;
                        aAny = getLineDash(xModel, aLineDashName);
                        aLineDash = aAny.get<drawing::LineDash>();
                    }
                }
            }
        }
}
```

And its result overwrites the former value of `aLineDash` variable.
Of course, in order to make this working, we need to modify some header files and the parameters of the function:

drawingml.cxx, WriteOutLine function (where we call the „new” getLineDash function):

```cpp
void DrawingML::WriteOutline( const Reference<XPropertySet>& rXPropSet, Reference< frame::XModel > const & xModel ) {
```

drawingml.hxx

```cpp
namespace uno {
    class XInterface;
}
namespace frame {
    class XModel;
}
```

chartexport.cxx (where we call WriteOutLine function):

```cpp
void ChartExport::exportShapeProps( const Reference< XPropertySet >& xPropSet )
{
    FSHelperPtr pFS = GetFS();
    pFS->startElement( FSNS( XML_c, XML_spPr ), FSEND );
    exportFill( xPropSet );
    WriteOutline( xPropSet );
    pFS->endElement( FSNS( XML_c, XML_spPr ) );
}
```
Of course, this fix is not perfect. We still have to work about the LO preset linestyles:

before ooxml export in LO

after ooxml export in LO
tdf#108028 Fix line breaking of chart title (before the fix)

- LibreOffice previously distorted the text and squeezed the chart, because the textbox of chart legend could not automatically wrap the text..
tdf#108028 Fix line breaking of chart title (after the fix)

- With the fix LibreOffice automatically wrap the text in the chart title textbox
• If the color or other property of a datapoint in the chart deviates from the dataseries property, a separate dPt tag will be written into the XML and fixing the lost properties during OOXML export.
Export the Overlap value with 100% for stacked charts, because the default overlap value of the Bar/Column chart is 0% and LibreOffice does nothing with the overlap value in Stacked Chart case, unlike the MS Office.
OOXML Export Chart shapes fill properties and some other bugs in one chart before a few fixes

- Just to represent how a Column charts looked like in LibreOffice 6.0.4 around July.
OOXML Export Chart shapes fill properties and some other bugs in one chart after a few fixes

Just to represent how a Column charts looks like in LibreOffice 6.2 these days
Thanks for the guidance from László Németh, and the attention from our viewers.
LIGHTNING TALK
SYSTEM-DEPENDENT DATA BUFFERING

CIB SOFTWARE GMBH
TIRANA, FRIDAY 27. SEPTEMBER 2018

ARMIN LE GRAND
SYSTEM-DEPENDENT DATA BUFFERING

- Transformation-invariant Instance of Data needed (translate/scale)
- OutputDevice to SAL-calls extended to use B2DHomMatrix as ObjectToViewTransform (old transformed to DeviceCoordinates)
- SystemDependentData (only known in sys-dep files/paths/Validity)
  - SystemDependentData_CairoPath (Cairo/Line)
  - SystemDependentData_GraphicsPath (Win/Line/Fill)
  - SystemDependentData_GdiPlusBitmap
  - SystemDependentData_Triangulation (X11 Line/Direct Triangulation, no Trapezoids anymore)
- SystemDependentDataManager (Global Instance/Mutex, std::shared_ptr)
  - ImplISVData::SystemDependentDataBuffer
  - size/time aspect buffering
- SystemDependentDataHolder (std::weak_ptr)
  - ImplBufferedData → B2DPolygon, B2DPolyPolygon
  - WinSalBitmap
- Uses typeid(T).hash_code() RTTI to identify
What Designing with LibreOffice Taught Me

Franklin Weng
TIRANA | 28 Sept. 2018
Has anyone ever deeply read this book?
Main Translator of Traditional Chinese version:

Cheng-Chia Tseng, Jeff Huang, Franklin Weng

*Also thanks to: Mark Hung, Bo-Ren Lin and Wei-Lun Chao*
What Designing With LibreOffice Taught Me
1. Use **style** instead of manually adjusting formats

Is that a common sense?
Apparently not.
春天感受春意

與烏來有約

烏來：位於臺北縣最南端，是臺北盆地周圍地勢較高之處，為臺北縣面積最大的鄉鎮。烏來境內有臺灣最大的林區『福山林區』，林木蓊鬱，全鄉土地 80 % 為森林所覆，涵養大台北地區的水資源，而植物方面有本區特有的烏來杜鵑、臺灣三角梅、烏來薰衣草等，因稀有及原始性，極具有學術及教育價值，福山的哈盆自然資源更是豐富，有『臺灣亞馬遜』之稱。

烏來：由於森林繁茂，溪流交錯，有利各種鳥類及哺乳類動物孳生，較具特色的有帝雉、藍腹鶴、臺灣藍鵲、竹雞、臺灣黑熊、臺灣野豬、臺灣獼猴、白鼻心等，但經長期獵捕，已所剩無多。烏來南勢溪、桶後溪、哈盆溪皆是臺灣富盛名之溪釣場所，常見魚類有臺灣魚斑、福山魚、香魚、虹鱒、臺灣缽口鰿....等。

獨特優雅的櫻花道
2. Create your own **templates** with styles you frequently use.

   Is that a common sense? Never.
3. Making images fixed in Writer is quite upsetting.

It's true when I merged every chapter of this book together...
4. Good and correct cross reference is really interesting and amazing.
Before merging...
After merging...

I don't need to do any additional setting!
5. Don't generate TOC like this:

<table>
<thead>
<tr>
<th>目錄</th>
</tr>
</thead>
<tbody>
<tr>
<td>壹、專案概述 .................................................................................................................. 1</td>
</tr>
<tr>
<td>一、專案名稱及時程 .................................................................................................. 1</td>
</tr>
<tr>
<td>二、專案執行目標 .................................................................................................. 1</td>
</tr>
<tr>
<td>三、專案範圍 ........................................................................................................... 1</td>
</tr>
<tr>
<td>貳、專案工作項目及執行策略 .................................................................................. 3</td>
</tr>
<tr>
<td>一、記者會 ................................................................................................................. 3</td>
</tr>
<tr>
<td>二、政策說明會 ......................................................................................................... 3</td>
</tr>
<tr>
<td>三、教育訓練 ............................................................................................................ 3</td>
</tr>
<tr>
<td>四、平面媒體報導 .................................................................................................... 5</td>
</tr>
<tr>
<td>五、製作 ODF 懶人包 ............................................................................................... 5</td>
</tr>
</tbody>
</table>

The result is ugly. It is also clumsy.

– Bruce Byfield,

*Designing With LibreOffice, 2016*

Unfortunately...
6. Don't use slides in speeches.

Hey, what am I doing?
If you really want to use slides

- Don't generate your slides this way.
- Summary slides can be useful only under one condition:
  - When there is no audience.
Anyway, we made it.

After translating and merging every chapter together, I'm feeling like an expert of typography.
Highly recommend you to deeply read it!

*And learn Chinese at the same time
Vertical Ruby Support in LibreOffice

Mark Hung
marklh9@gmail.com
TIRANA | 28 Sept. 2018
“Ruby” aka Asian Phonetic Guide

- Is a way to mark CJK ideographs so that people know how to read them.
- For children in Taiwan.
- Essential for Japanese names.

鬼門の方角を凝視する。
Vertical Ruby

- LibreOffice supports ruby over or below the base text since beginning.
- Marking on the right vertically is used in Taiwan. Top or bottom is rare.
Challenge: SwMultiPortion

- Fear of breaking BIDI support
• Text layout.
• New option for ruby position in the ruby dialog.
• ODF import / export with loext:ruby-position.
• DOCX import / export ready.
Thanks for Listening
Text layout performance

By Miklos Vajna
Software Engineer at Collabora Productivity
2018-09-28
Motivation

Quoting the TDF wiki:

- All over the code base assumes shaping text is cheap
- We can do it over and over again
- Want to measure the text? Shape it and discard the output afterwards. Want to measure part of the same text? Shape again. Want to find line breaks? Shape again. Want to finally draw it? Shape again.
- In contrast, other toolkits typically calculate the layout explicitly:
  - e.g. GtkLabel has an explicit PangoLayout, etc.
Work done: shared code

vcl

- StatusBar (thanks to Tamás Bunth)
- ListBox
- Menu

svtools

- Ruler

Not a cache: just lazy-calculate the layout
Work done: Writer

Good news

• Everything goes through SwFntObj
• Can cache GetTextArray(), DrawTextArray(), GetTextWidth(), GetTextBreak() calls on OutputDevice

Bad news

• Need a cache: SwFntObj typically doesn’t have a good idea about the lifecycle of the created text layout
• For now: SwTextGlyphsKey stores which subset of which text was laid out on which output device
• Storing the full output device state is too expensive
  • Can’t handle everything transparently in VCL
Results

Metric: # of layout calls

- GenericSalLayout::LayoutText()

Measured: one Writer scenario (equivalent of UITests in sw)

- 55,793 layout calls initially
- 18,126 layout calls now
- Only 32% of the original work is performed
Thanks indeed!