

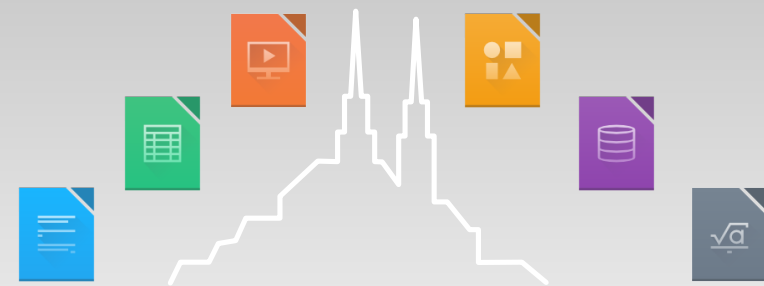
Evaluation of LibreOffice – MS Office interoperability and automated bisection of interoperability regressions

- ▼ Miloš Šrámek



*Society for Open Information Technologies
(Slovakia)*

Earlier Work

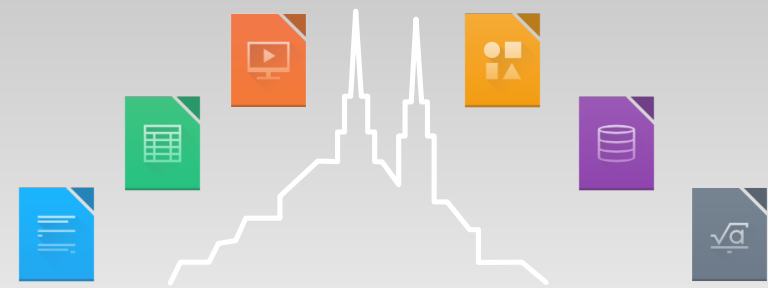


- ▼ Presented at Plugfest 2011 in Berlin:
 - ▼ ODF document overlays for a detailed visual inspection
- ▼ Presented at LibreOffice Conference 2013 in Milano:
 - ▼ added 4 error measures, 4 views
- ▼ Both: Testing applications “each with each”

File	#	0	1	2	3	8	4	5	6	7	9	10
AbiWord_2_8_6_Linux_	0	100	81,92	81,92	24,54	24,55	24,54	24,6	24,6	24,6	24,3	24,3
AbiWord_2_9_0_Linux_	1	81,92	100	100	24,51	22,89	22,89	22,95	22,95	22,95	22,71	22,71
AbiWord_2_9_1_Linux_	2	81,92	100	100	24,51	22,89	22,89	22,95	22,95	22,95	22,71	22,71
KOffice_2_3_3_Linux_	3	24,54	24,51	24,51	100	34,44	34,53	34,3	34,3	34,3	33,02	33,02
StarOffice_9_2_Linux_	8	24,55	22,89	22,89	34,44	100	99,51	79,01	79,01	79,01	56,29	56,29
LibreOffice_3_3_2_Linu	4	24,54	22,89	22,89	34,53	99,51	100	79,32	79,32	79,32	56,33	56,33
LibreOffice_3_3_3_Linu	5	24,6	22,95	22,95	34,3	79,01	79,32	100	100	100	60,8	60,8
LibreOffice_3_4_1_Linu	6	24,6	22,95	22,95	34,3	79,01	79,32	100	100	100	60,8	60,8
OpenOffice_org_3_4Be	7	24,6	22,95	22,95	34,3	79,01	79,32	100	100	100	60,8	60,8
Microsoft_Office_2007_	9	24,3	22,71	22,71	33,02	56,29	56,33	60,8	60,8	60,8	100	100
Microsoft_Office_2010_	10	24,3	22,71	22,71	33,02	56,29	56,33	60,8	60,8	60,8	100	100

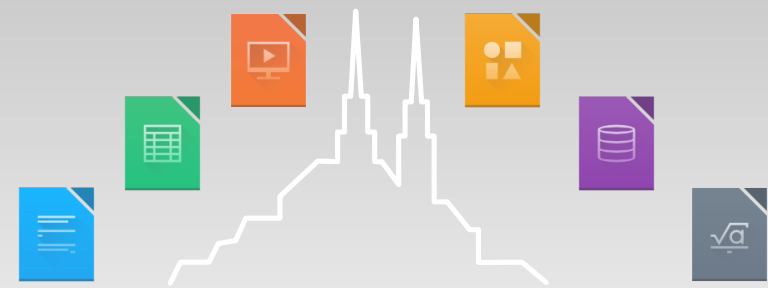


Talk overview



- ▼ Automated difference grading and visual inspection
- ▼ Roundtrip and print tests
- ▼ Document relevance
- ▼ Results of batch testing
- ▼ Automated bisection of interoperability errors

Difference grading



▼ Four numeric error measures used:

▼ Per page:

- 1) Line Number Difference (LND): missing lines, bad object position
- 2) Text Height Error (THE): errors in line spacing, table or object shape etc

▼ Per line:

- 1) Horizontal Line Position Error (HLPE): horizontal shift of dominating line segments
- 2) Feature Distance Error (FDE): maximum distance between features of aligned lines

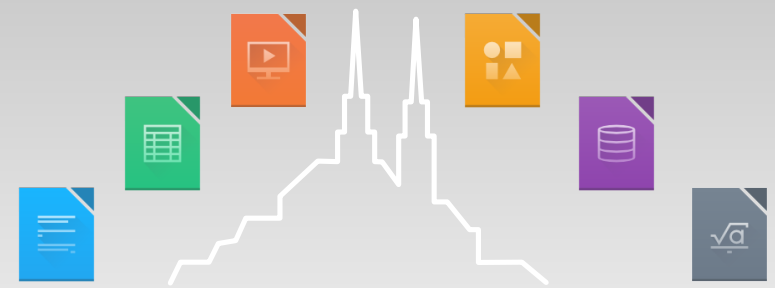
▼ Grading:

- ▼ 0 ... 5 (pixel identical ... very bad), 6 (empty document), 7 (failed to open)

▼ Textual report:

- ▼ automatically generated Calc spreadsheet

Feature Distance Error

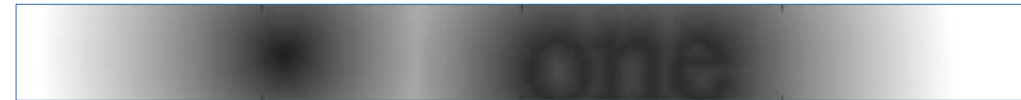
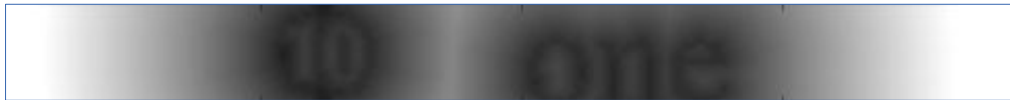
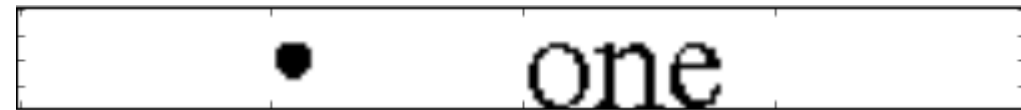


- Observation: Maximum value of $\text{abs}(\text{DF}(\text{L1}) - \text{DF}(\text{L2}))$ is equal to a distance between different local features
- Characterizes difference on character level



Document 1 line

Document 2 line

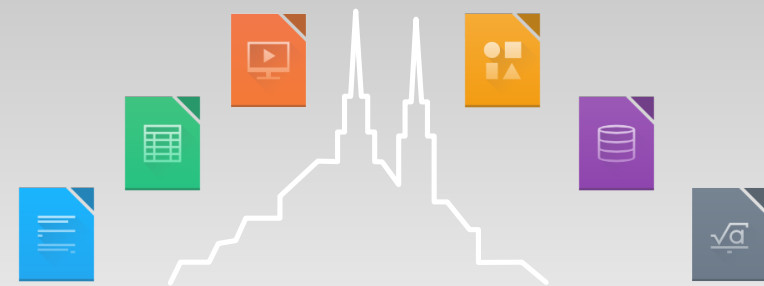


DF(L1)

DF(L2)

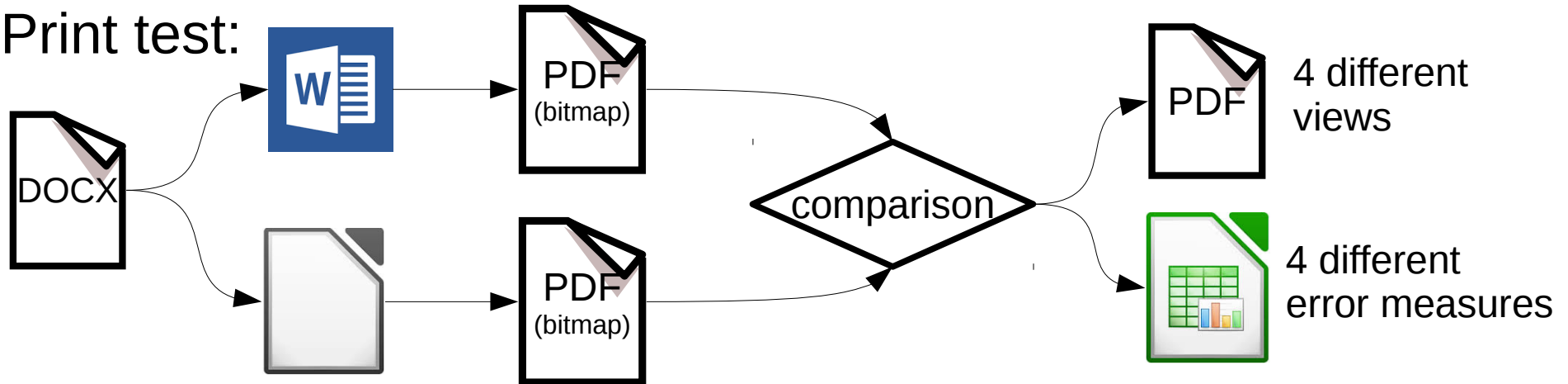


Roundtrip and print tests

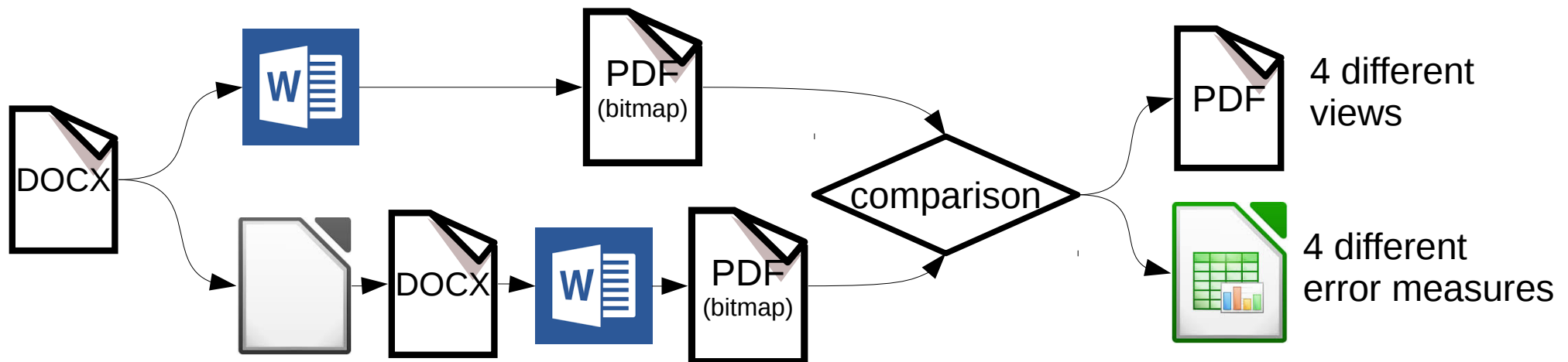


▼ Motivated by Adam Fyne's talk in Milano

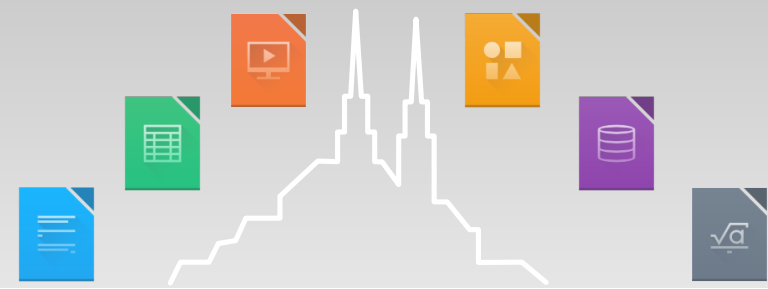
▼ Print test:



▼ Roundtrip test:

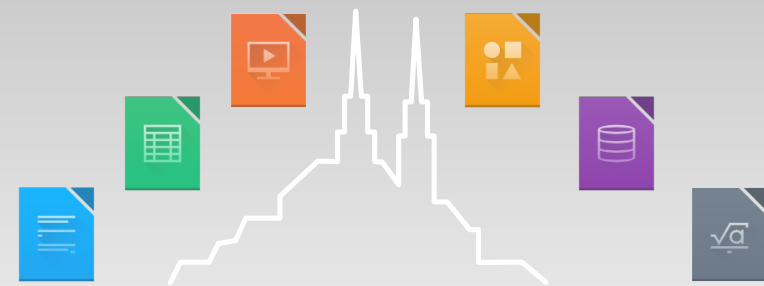


Document Comparison Views



- 1) Documents printed to PDF and converted to bitmap
 - ▼ View 1, side-by-side
 - ▼ View 2, *“Page overlay, no alignment”*
- 2) Segmentation in lines
- 3) Vertical line alignment
 - ▼ Heuristics used in number of line does not match
 - ▼ View 3, *“Page overlay, vertically aligned lines”*
- 4) Horizontal line alignment
 - ▼ View 4, *“Page overlay, vertically and horizontally aligned lines”*

Side-by-side view



- ▼ Shows large differences
- ▼ Useful for complex layouts
- ▼ **Example:** Test File_672.docx, LO43:
 - ▼ Line Number Difference (LND): 0 (graded 0)
 - ▼ Text Height Error (THE): 4.57 mm (graded 3)
 - ▼ Feature Distance Error (FDE): 11.4 mm (graded 3)
 - ▼ Line Position Error (LPE): 11.43 mm (graded 5)



Source: Synthetic_Files/./File_672.pdf

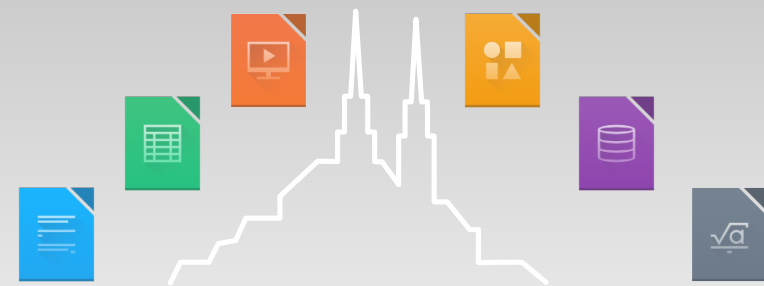
PagePixelOverlayIndex[%]: 22.6 : FeatureDistanceError[mm]: 11.4 : HorizLinePositionError[mm]: 11.43 : TextHeightError[mm]: 4.57 : LineNumDifference: 0

- a) Right Alignment
- b) Right Alignment
- c) Right Alignment
- d) Right Alignment
- e) Right Alignment

Target: BB43AL/./File_672.BB43AL.pdf

- a) Right Alignment
- b) Right Alignment
- c) Right Alignment
- d) Right Alignment
- e) Right Alignment

“Page overlay, no alignment” view



- ▼ A more detailed display of differences
- ▼ Different line spacing hides other differences
- ▼ **Example:** Test File_672.docx, LO43:

```
Page overlay, no alignment  
cyan: source Synthetic_Files/./File_672.pdf  
red: target BB43AL/./File_672.BB43AL.pdf  
PagePixelOverlayIndex[%]: 22.6 : : TextHeightError[mm]: 4.57
```

a) Right Alignment

b) **Right Alignment**

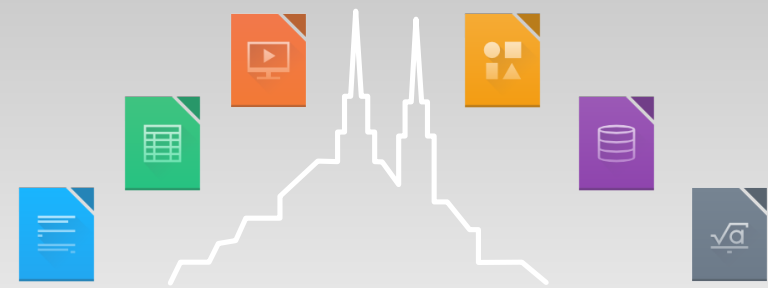
c) Right Alignment

d) **Right Alignment**

e) Right Alignment

e) Right Alignment

“Page overlay, vertically aligned lines” view

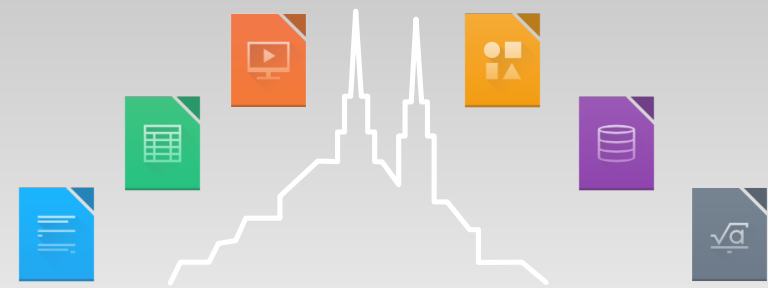


- ▼ Shows lines with differences
- ▼ **Example:** Test File_672.docx, LO43:

Page overlay, vertically aligned lines
cyan: source Synthetic_Files/./File_672.pdf
red: target BB43AL/./File_672.BB43AL.pdf
: HorizLinePositionError[mm]: 11.43

- a) Right Alignment
- b) Right Alignment
- c) Right Alignment
- d) Right Alignment
- e) Right Alignment

“Page overlay, vertically and horizontally aligned lines” view

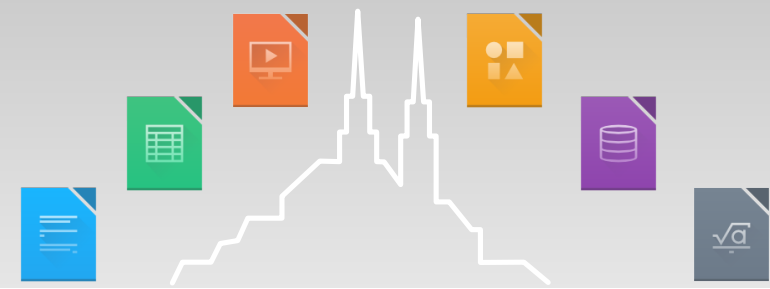


- ▼ Shows differences in lines
- ▼ **Example:** Test File_672.docx, LO43:

Page overlay, vertically and horizontally aligned lines
cyan: source Synthetic_Files/./File_672.pdf
red: target BB43AL/./File_672.BB43AL.pdf
FeatureDistanceError[mm]: 11.4

- a) Right Alignment
- b) **Right Alignment**
- c) Right Alignment
- d) **d) Right Alignment**
- e) **e) Right Alignment**

Improvement since LO44

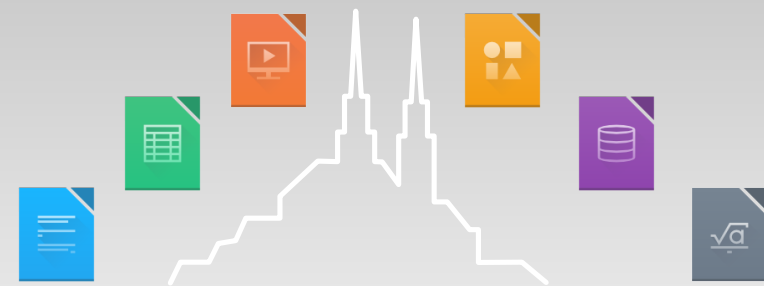


- ▼ Shows lines with differences
- ▼ **Example:** Test File_672.docx, LO44
- ▼ Horizontal position improved :-)

Page overlay, vertically and horizontally aligned lines
cyan: source Synthetic_Files/./File_672.pdf
red: target BB44DL/./File_672.BB44DL.pdf
FeatureDistanceError[mm]: 0.5

- a) Right Alignment
- b) **Right Alignment**
- c) Right Alignment
- d) **Right Alignment**
- e) Right Alignment

Improvement since LO44



- ▼ Shows lines with differences
- ▼ **Example:** Test File_672.docx, LO44
- ▼ Vertical position is the same :-)

```
Page overlay, no alignment  
cyan: source Synthetic_Files/./File_672.pdf  
red: target BB44DL/./File_672.BB44DL.pdf  
PagePixelOverlayIndex[%]: 27.5 :: TextHeightError[mm]: 4.70
```

a) Right Alignment

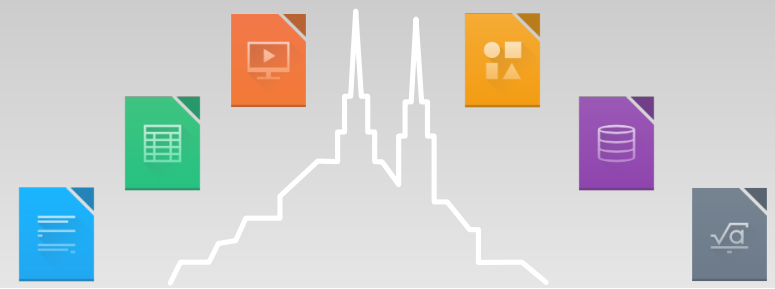
b) **Right Alignment**

c) Right Alignment

d) **Right Alignment**

e) Right Alignment

e) **Right Alignment**

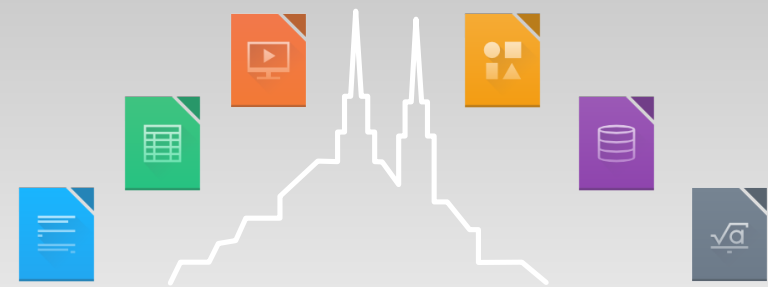


Demo: spreadsheet report

<https://is.gd/idudac>

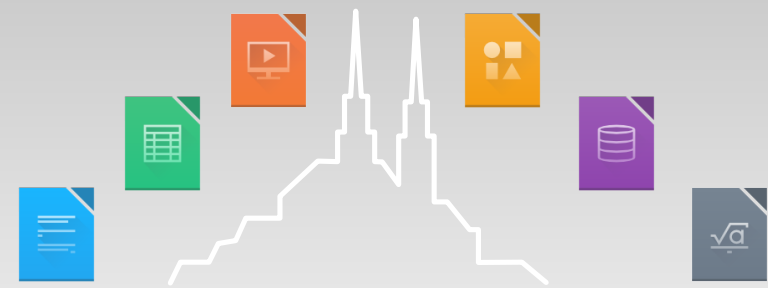


Document relevance

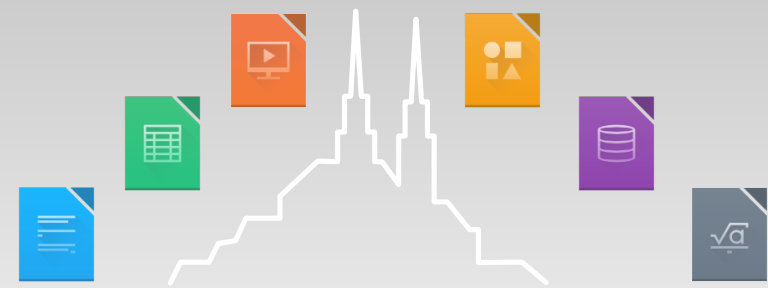


- ▼ An idea: sort documents according to popularity of their features
- ▼ **Document relevance:** relative frequency of the least frequently used tag in a file, normalized to range (0,1)
- ▼ How to get:
 - ▼ Download numerous random documents from the internet
 - ▼ Extract tags
 - ▼ Simplify, if necessary (ignore numeric values etc)
 - ▼ Count, sort and normalize.

Results



- ▼ Tested 1600+ docx files
- ▼ Reference Word program: MSOffice 2010 running in Wine
- ▼ LO versions:
 - ▼ 10 versions available from the **lo-linux-dbgutil-daily...** bibisect git repositories
- ▼ Tests:
 - ▼ Roundtrip
 - ▼ Print
- ▼ Statistics:
 - ▼ 32000 pair comparisons ($1600 \cdot 10 \cdot 2 = 32000$)
 - ▼ two – three days computation



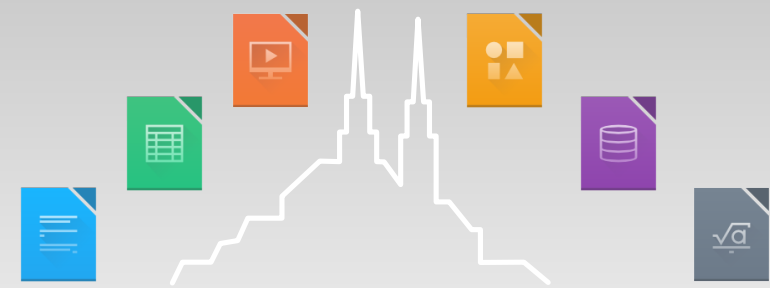
Demo: report for 1600+ tested documents

<https://is.gd/ehukoj>

(a large spreadsheet, opens only in LO52 and later)



Observations

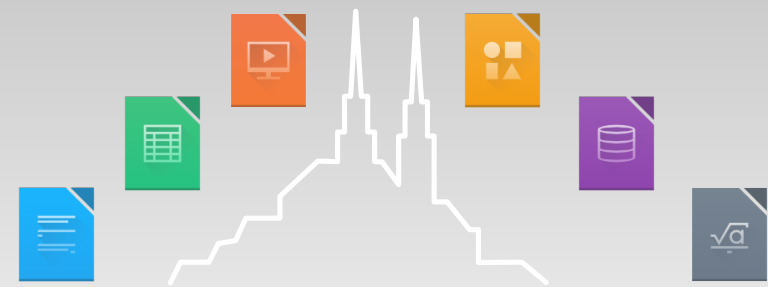


▼ Overview, 1600 documents:

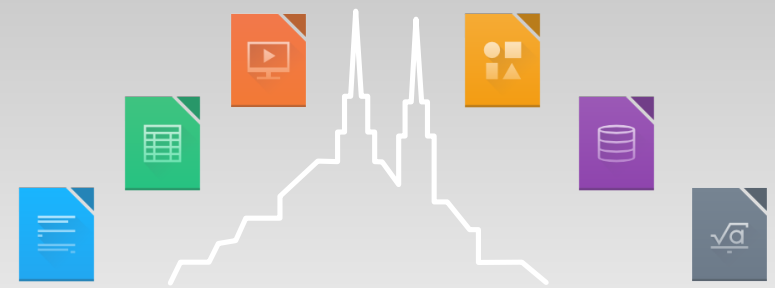
	regressions	progressions	good (grade < 3)
roundtrip	361	932	1078
print	731	676	583

- ▼ Problems even in simple documents, most visible ones:
 - ▼ Kerning handled incorrectly (hides other errors)
 - ▼ Incorrect line spacing (40 % of all documents)
- ▼ Shortcomings of evaluation
 - ▼ Feature distance error sometimes overestimated
 - ▼ absolute now, relative would perhaps be better
 - ▼ Binarized images compared, threshold 250 currently used
 - ▼ May cause false grading (charts and images)
 - ▼ Maybe not all problems covered

Automated Bibisection



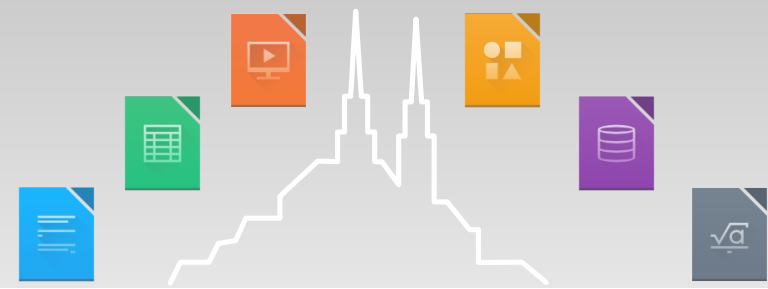
- ▼ Git provides tools for automated bisection
- ▼ A proper script needed returning good/bad status
- ▼ The procedure
 - ▼ Get a buggy document
 - ▼ Find the right bisection repository (see the previous example)
 - ▼ Run the bisection script
 - ▼ ...
 - ▼ Submit a patch :)



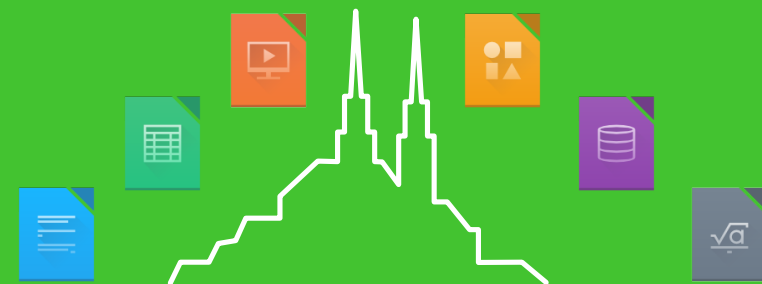
Demo, local
Report with regressions: <https://is.gd/gecuro>



Summary



- ▼ We ban test:
 - ▼ Any office application with command line interface
 - ▼ Any office format:
 - ▼ odt, doc, docx, rtf (tested)
 - ▼ odp, doc, docx (not tested, should work)
 - ▼ ods, xls, xlsx (not tested, should work)
- ▼ To be done:
 - ▼ Classify test document according to their features
- ▼ Future: add into the LO QA framework?



Thank you for watching!

- ▶ Code with example data is available at [github](#)
- ▶ Questions, comments, suggestions: [milos.sramek\(at\)soit.sk](mailto:milos.sramek@soit.sk)



All text and image content in this document is licensed under the [Creative Commons Attribution-Share Alike 3.0 License](#) (unless otherwise specified). "LibreOffice" and "The Document Foundation" are registered trademarks. Their respective logos and icons are subject to international copyright laws. The use of these therefore is subject to the [trademark policy](#).

