LibreOffice: the story of cleaning and re-factoring a giant code-base

Michael Meeks <michael.meeks@suse.com>

mmeeks,#libreoffice-dev, irc.freenode.net

"Stand at the crossroads and look; ask for the ancient paths, ask where the good way is, and walk in it, and you will find rest for your souls..." - Jeremiah 6:16

LibreOffice

openSUSE

How we did it – an overview ...

- Culture as a vital foundation
- Making it easy to contribute
  - Quality through sharp tooling
  - Making the code comprehensible
- The history of LibreOffice
- Cleanups past & future
- Quality measurements
- LibreOffice 4.0
- Getting involved

Culture – it's power and reach

indebted to "The wealth and Poverty of Nations" (Landes)

What is different about LibreOffice ?

Free Software is primarily about people:

And not about software.

Ethos / Reciprocity / Licensing / Friendship / Fun

The importance of culture

- Historical accident or cultural consequence ?
  - eg. Japan – the only Asian nation to Industrialize fast.
- Early / Ming China:
  - Invented ~everything … paper, printing, gunpowder,
  - Would send huge navies to other nations:
    - to show them how wonderful they were.
      - (today China has a different ethos of course.)
- The West:
  - Greedy / rapacious (of course)
  - Also greedy for new ideas, better ways of doing things
  - Syndicates saving to buy clocks in the UK: hungry to optimise their lives by measurement.

LibreOffice

A snapshot of a dead culture

- Change is dangerous
    - Rapid change might loose us control ...
- Past Orientation: things were better previously
    - Things should be like they were but more so.
- Mandatory process must be used to deter changes that might cause problems.
- Development must be done professionally

- Get permission first
    - Code owners should approve changes in their area
- External patches distract me from my more important work
- Over-design, under-implement

"A problem for every solution"

Of course, the big picture is complicated by a lot of outstandingly friendly & open individuals.

Attempting a cultural rupture ...

- Change is mandatory
  - Without it we will die.
- Future Orientation: things are going to get better
  - LibreOffice's best days are in the future …
  - Will you join us getting there ?
- Inclusion is more important than outright perfection
- under-design and iterate

- Breakage is a normal part of development, fix it if we find it.
- Unnecessary process annoys us too: lets fix that together !
- Hacking LibreOffice should be a fun, rewarding, relational experience.

"can-do attitude & if not: a really, really good reason"

LibreOffice
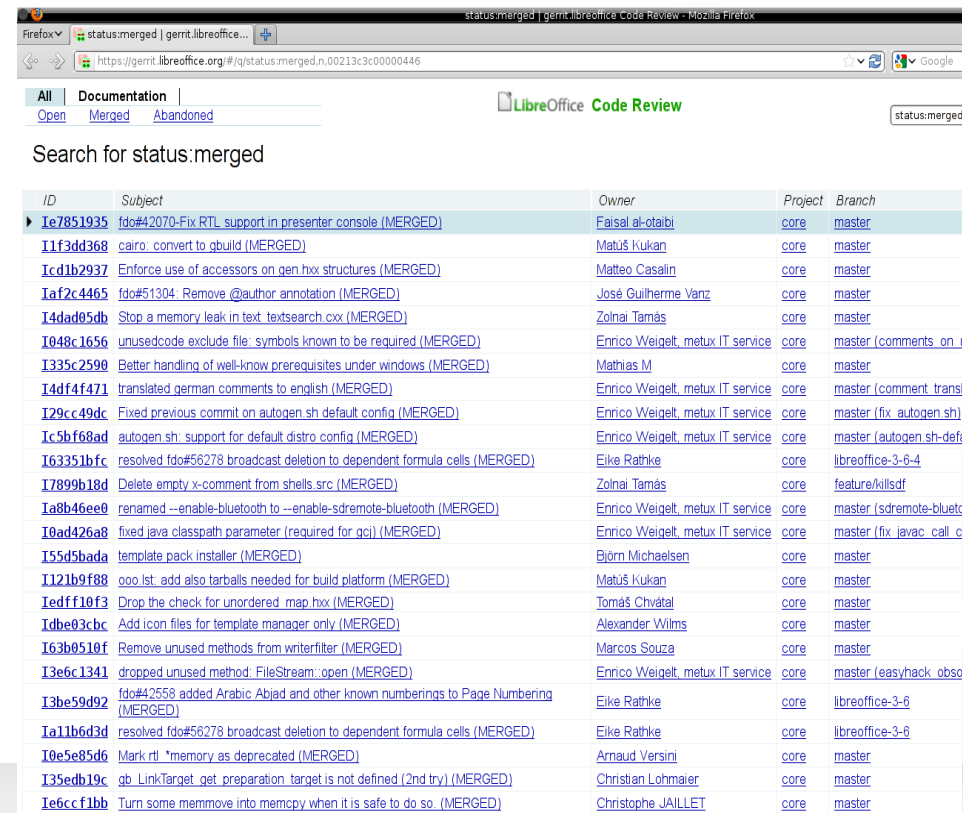
Making it easy and fun to contribute

Permission free, low friction on-ramp

- Easy hacks page
  - We want your first patch to be non-controversial, and easy, so you're up-to-speed and included outright:
    - http://wiki.documentfoundation.org/Development/Easy_Hacks
    - **changes are most welcome !**
- Open Mailing lists
  - No subscription required
  - No Reply-To: mangling – '*I get a reply not just the list*'
  - Mail your patch and you're done …
- Documentation http://docs.libreoffice.org/
  - 200+ READMEs files with overview in git modules ...
  - migration of comments to doxygen format

LibreOffice

Gerrit – permission free commits

- Gerrit - https://gerrit.libreoffice.org
  - Magic to turn an openID account (eg. Gmail)
  - Into **no-ask** git commit / push access to gerrit
  - Submit to patch queue backed by mailing-list
  - Code inclusion is: git fetch / cherry-pick FETCH_HEAD
  - Test build integration on it's way.
    - *Thanks to*
      Norbert Thiebaud
      Bjoern Michaelsen
      David Ostrovsky

Reliable out-of-master builds ...

Lots of big / fast / ccache enabled tinderbox slaves.

Android (x86, ARM), iOS, (many) Linux x86, x86_64, Windows, Mac

Building & up-loading binaries as well for QA.

Thanks to
Norbert Thiebaud
Bytemark & more

More powerful static checking ...

- Clang: `compilerplugins/`
- eg. Adding custom compile-time verification hooks

```
SAL_INFO( "bridges.ios", "info message to output..,");
```

  - Compile-time parse doxygen docs & verify **"bridges.ios"**

- eg. Or sensible whitespace paranoia:
  - Body of if/while/for not in {}

```
if( a != 0 )
        b = 2;
        c = 3; // tinderbox death here ...
```
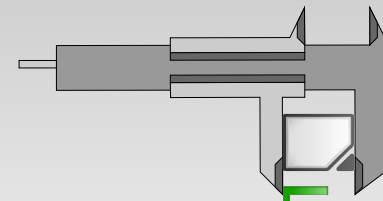
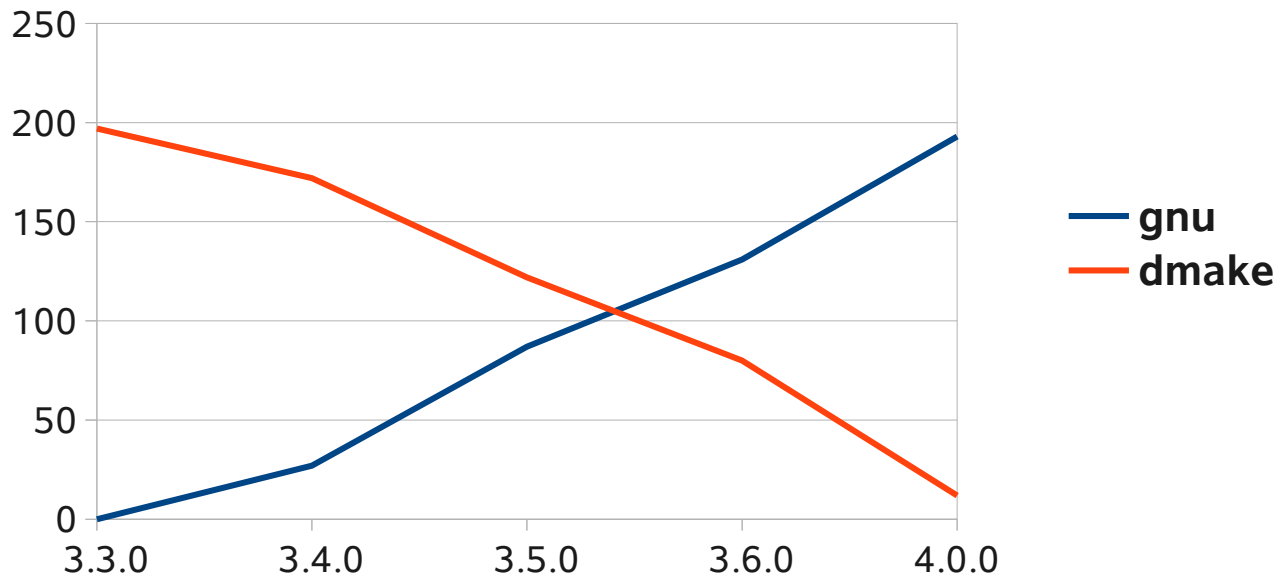- Helps reduce scope for manual mistakes …
  - Thanks to *Lubos Lunak !*

Making the code readable / hackable

Using a standard make tool (gnumake)
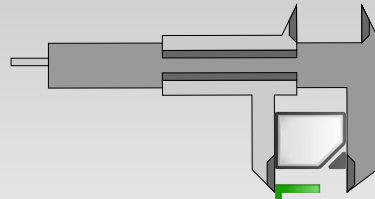
- Near-complete work to kill **'dmake'**
  - Faster, more standard & hack-able
  - Huge parallelism possible for builds on big-iron.
 - Enables library merging → one monster lib.

gnumake vs. dmake by module count

With thanks to
**David Tardon
(RedHat)
Peter Foley,
Matúš Kukan
David Ostrovsky
Pierre-Eric
Pelloux-Prayer
(Lanedo)
and more ...**

Translating German Comments

- Makes things **significantly** easier for non-Germans …
- **~20k lines** done - **~20k to go** ~40%...
- **Making the code more internationally accessible ...**
  - Lots of comment translators distracted by code hacking...

Detected lines of German comment

60,000	
50,000	
40,000	
30,000	
20,000	
10,000	
0	3.3 3.4 3.5 3.6 4.0
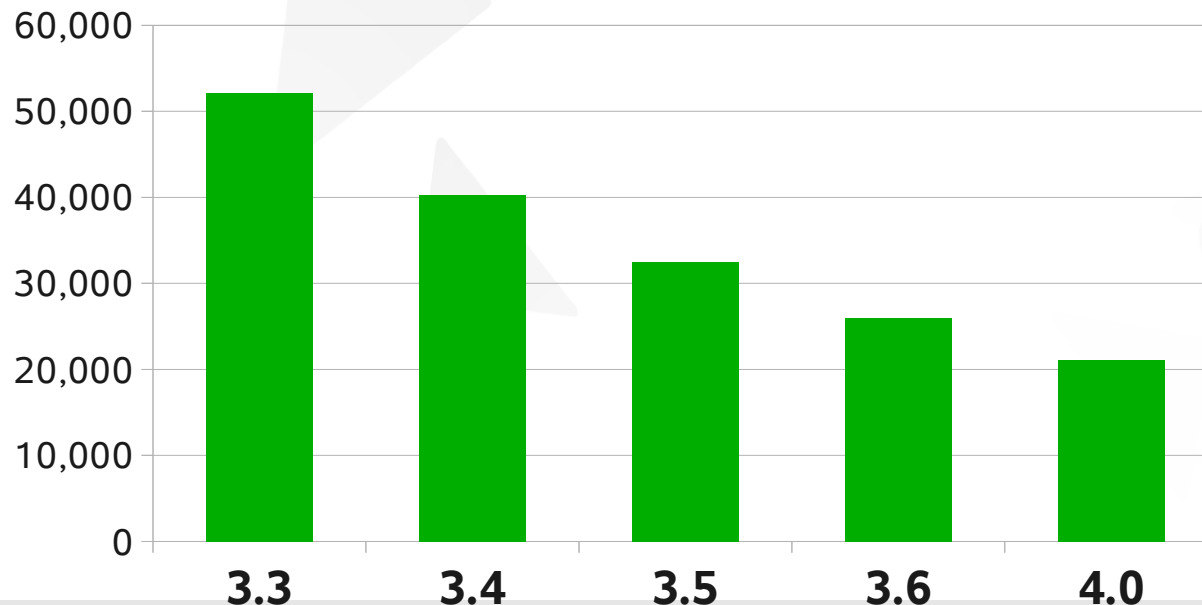
With thanks to (recent translators):
Philipp Weissenbacher
Philipp Riemer
Samuel Mehrbrodt
Enrico Weigelt
Lennard Wasserthal
Albert Thuswaldner
Oliver Günther
Markus Maier
Peter Baumgarten

and many more !

Four+ String classes: cleanup for readability ...

- Killed ByteString completely
- Tools' UniString well on it's way out …
  - many modules completely clean
- rtl:: prefix not required for OUString / OUStringBuffer etc.
- Kill horrible string macros & generate more efficient code:
  - Template driven goodness; readability finally ...

```
- if( aModuleId.equalsAsciiL( RTL_CONSTASCII_STRINGPARAM(
-                              "com.sun.star.text.TextDocument" ) ) ||
-     aModuleId.equalsAsciiL( RTL_CONSTASCII_STRINGPARAM(
-                              "com.sun.star.text.GlobalDocument" ) ) )
+ if( aModuleId == "com.sun.star.text.TextDocument" ||
+     aModuleId == "com.sun.star.text.GlobalDocument")
```

- Thanks to: *Caolan McNamara, Lubos Lunak, Jean-Noël Rouvignac, Ricardo Montania, Matteo Casalin, Christina Rossmanith, Noel Grandin, Marcos Paulo de Souza + many others …*

Other cleanups & general changes ...

- cppcheck - ~1000 patches
  - Thanks to *Julien Nabet, Radu Ioan, Christophe Jaillet & more …*
  - Also a number of cppcheck fixes …
- Dead code removal
  - ~all certainly-dead code (from callcatcher) removed …
  - unused virtual methods need hunting: (a clang plugin?)
  - Thanks to *Caolan McNamara, Julien Nabet, Marcos Paulo de Souza, Paula Mannes, Enrico Weigelt, Matúš Kukan* and others for finishing the job ...
- Adopting new UNO technologies
  - Actually deploy cool new things across the code-base.
  - Compile-time checked component names not fragile strings.
  - `- ...”com.sun.star.embed.EmbeddedObjectCreator”`
    `+ embed::EmbeddedObjectCreator::create()`

Completed STL porting ...

- Danger for readability / learnability's sake …
- Binning internal stlport copy: use boost + system STL
- Legacy C++ code-base pre-dates STL, and even templates …
- Tools classes: List, Container, Table, DynArray and derivatives
    - large scale code clean-up
- Benefits:
    - readability, performance (?), less generated bloat (?)
    - finding fragile code & fixing it
    - checking iterators / OOB issues
- Thanks to: *Noel Grandin, Michael Stahl, Caolán McNamara, Ivan Timofeev, Fridrich Strba, Nigel Hawkins*, and many others

More risky re-work / cleanup / re-factoring

- Binning obsolete libraries: 'libvos' *(Norbert Thiebaud)*
- Enforced whitespace cleanup: tabs → spaces
- Windows installer:
  - NSIS → clean .msi with .msp patching *(Andras Timar - SUSE)*
- Killed horrible SDF translation intermediate
  - Direct .po → .res / .xml etc. *(Zolnai Tamás)*
- LanguageTags extending country/lang
  - All language features: Serbian: Cyrillic / Latin etc. *(Eike Rathke – RedHat)*
- VCL re-factor: *(Michael Meeks - SUSE)*
  - Share more backend / generic code + partial gtk3 impl.
  - improved 'headless' backend for servers *(Ricardo Cruz)*
- Wizards Java → Python Migration *(Xisco Fauli)*

LibreOffice

Lightning History of the codebase

A quick potted history of LibreOffice' code ..

- Many decades ago,
  - A very talented programmer sat his shed and created a C++ Object Oriented toolkit …
    - Then some demo apps …
  - Fast forward to today, those demo apps are LibreOffice
    - Many key architectural considerations made without careful thought of knock-on consequences today ...
- In latter history:
  - Architect-led, cargo-cult, UNO component-model fetishism
    - Galloping inefficiency; UNO not focused on what it' could excel at ie. scripting bindings
    - Un-necessarily opaque, over-generic code: hard to extend.
    - Rampant duplication for the sake of 'UNO'-isation …
    - Scattered / incomplete UNO migrations ...

Example larger re-factorings

Completed Microsoft filter re-factoring ...

- Re-factored RTF to share domain-mapping logic for import + export
  - Much richer feature compatibility.
- Previously
  - Three duplicate import filters …
- One to go ...

Writer Core

UNO

Bulk of Import Filter: Domain-map / Insert

RTF

DOCX

Import

C++

Duplicate Domain-map DOC importer

Import

Cut + Paste

C++

Bulk of Export Filter: Collect / Domain-map

DOC

RTF

DOCX
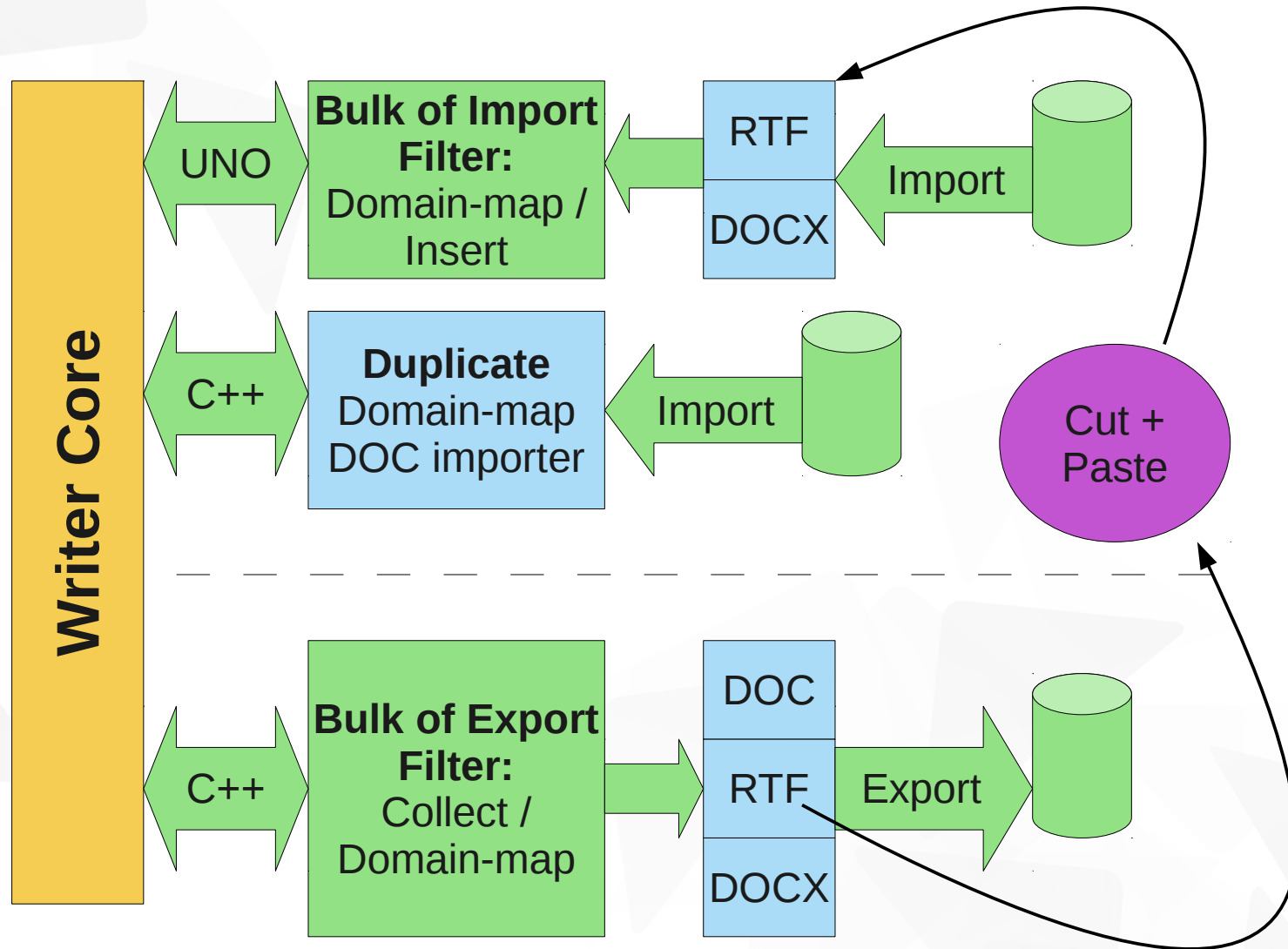
Export

Thanks to *Miklos Vjana*

LibreOffice

Calc: re-factoring issues out ...

- eg. Cell Storage / notes
  - Wasting 4-8 bytes per cell for a note that is ~never there
  - O(num-cells*num-notes) performance in export code
  - Re-worked note storage / copy/paste/undo/redo – big savings.
    - Thanks to *Markus Mohrhard*
- Targetted de-UNO-isation & re-work of XLSX filter
  - Substantial performance wins: thanks to *Daniel Bankstone*
- Multi-Dimensional Data Structures (MDDS)
  - Split-out, re-usable C++ template library for calc core structures
  - `mdds::flag_segment_tree`
    - column & row flags / widths, pivot table cache
      - big memory savings & scalability wins
  - multi-type containers coming for more wins ...

Ongoing heavy lifting ...

Catching up to 1995: layout
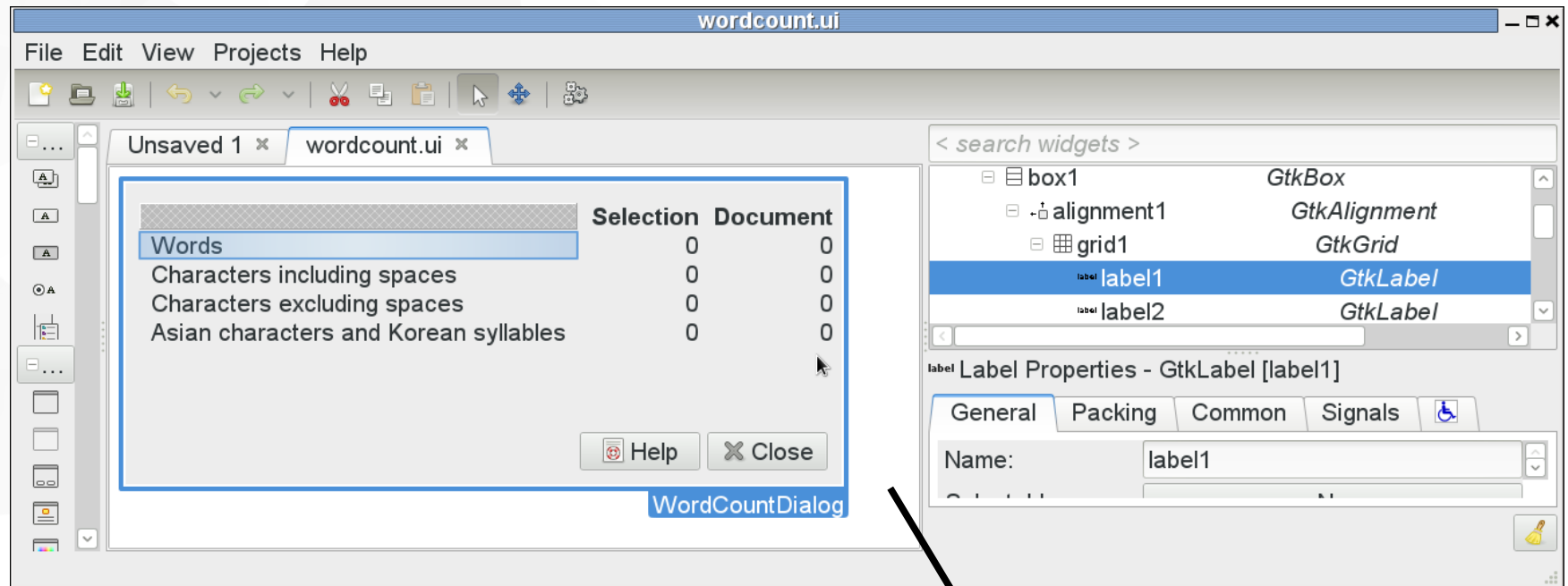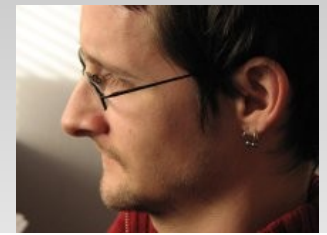
- Thanks to **Caolan McNamara, Gokul, Joren De Cuyper**

- Sizing, positioning, Designer tweaking, much improved l10n

- Start of a long process:

  - 100 (of 500) dialogs converted
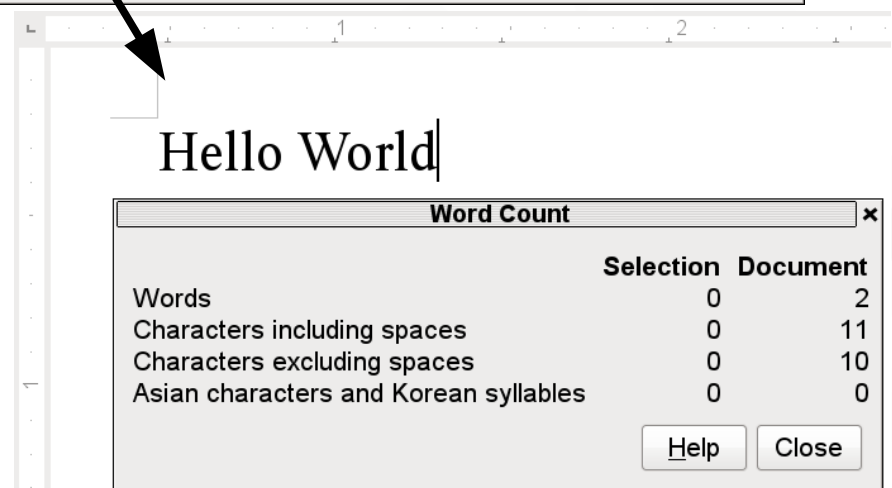
Other ongoing work ...

- Extreme VCL / toolkit lameness
  - no sensible lifecycle handling – Clang re-writing needed
  - Re-rendering should happen at idle
  - main-loop has no true 'idle' or priority concept.
- Writer:
  - BigPtrArray – a real pain, type at the start of a big word doc.
  - Try using red-lining in a large document
  - Dense B+ Tree replacement in progress *(feature/bplustree)*
    - O(log(N)) operations. Thanks to *Kendy*
- Code cleanliness re-working
  - com::sun::star:: namespace ~done – waiting for big merge window to open post 4.0.2 thanks to *Lubos Lunak.*
- There should be no "no-go areas" in the code-base ...

Clang code cleanups ...

- Auto-generated patching using compiler semantics
- Bulk-change merge window opens ~2 months out
  - after 4.0.2 and era of busy cherry-picking closes ...
- Wish-list / in-progress:
  - help appreciated
  - Misc. function renames
  - SvStream << operator re-writing to 'writeInt16' etc.
    - Protect binary file-formats from unexpected type changes.
  - Global fix of OUString("foo") to "foo" wherever possible
  - And many other scattered annoyances
- Tutorial: how to write plugins ...
  - http://wiki.documentfoundation.org/Clang_plugins
- Thanks to *Lubos Lunak (SUSE)* for mentoring.

LibreOffice
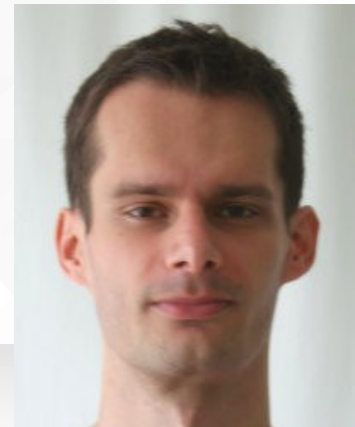
Retaining / Improving Quality
How can we do that ?

Unit testing for quality ...

- Improved quality through unit testing
  - Coverage & number of tests increasing rapidly
  - security regressions, document layout, formulae calcs etc.

Count of types of gnumake enabled unit tests
(each runs a battery of tests)
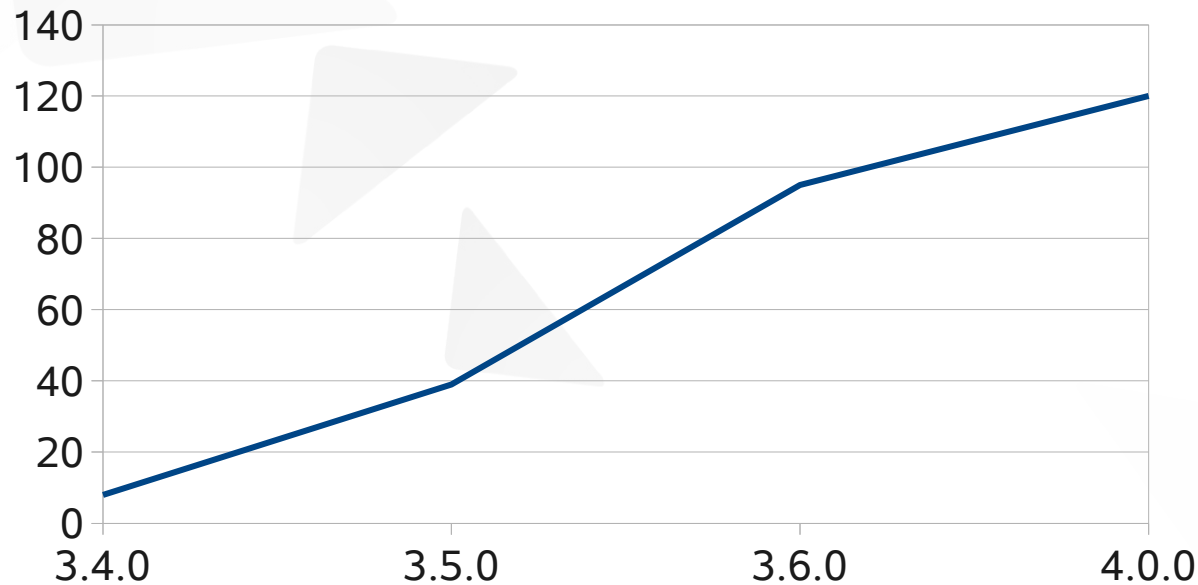
With many thanks to
  Markus Mohrhard
    (above)
  Miklos Vjana
  Kohei Yoshida
    (SUSE)
  Artur Dorda
  Daniel Bankstone
    (GSOC)
  Caolan McNamara
  Michael Stahl
    (Redhat)
  And many more
  Who got tired of fixing
  The same bugs ...

LibreOffice

Bugzilla Assistant on freedesktop

- **Connected to Help menu "Send Feedback"**
  - pre-populates version, component, etc. for you
- **Avoids ~unusable default bugzilla / file bug UI**
- **Thanks to: Loic Dachary & Rob Snelders**

Release often, and predictably on-time ...

- A **time based**, **six-monthly** release train …
  - **synchronized** with the Linux distributions cadence
    - ie. a normal Free Software project
  - rapid fire, (**monthly**) bug-fix releases: ~1x per 2 weeks.
- Any bugs only **partially fixed**: can be re-fixed in < a month.

LibreOffice Releases

By using a time-based release schedule and maintaining two parallel branches,
LibreOffice will always be available both for early adopters and conservative users,
with the most appropriate version for each group's specific needs and habits.

3.3 Family	3.3.2		3.3.3			3.3.4			
3.4 Family		3.4.0		3.4.1	3.4.2		3.4.3	3.4.4	
3.5 Family							3.5.0		3.5.1	3.5.2

Developers	daily snapshots of master / self-built versions					
Bleeding Edge	3.3.2	3.4.0	3.4.1		3.4.3	3.5.0	3.5.1
Stable		3.3.3		3.4.2		3.4.4	3.5.2
Very Stable	3.3.2	3.3.3			3.4.3	3.4.4	
Rock Solid			3.3.3	3.3.4			3.4.4

LibreOffice

QA / BiBisection – amazingly powerful

- Chasing regressions:
  - Highly coupled, fragile code in many places
  - Large scale changes going on all around …
  - How can we find what broke it ?
- Git bisection prohibitively slow:
  - 4+ hour build time each time
- Bi(nary)Bisect (thanks to *Bjoern Michaelsen - Canonical*)
  - A single git repository …
  - 1000's of (relocatable) binary builds of LibreOffice
  - just run "git bisect" …
    - An non-developer can point to the ~20 commits
    - Few skills required: 10 tests in <10 minutes.
  - The magic of git packing makes this reasonable small

LibreOffice

Sane debugging with gdb ...

- LibreOffice uses UTF16 to represent 90%+ ASCII strings
  - gdb can't cope with that – without some extra python:

```
...
#5  0xb7a4b220 in createTypeRegistry
(libraryDirectoryUri="file:///data/opt/OOInstall/ure/lib",
 uris="file:///data/opt/OOInstall/ure/lib/../share/misc/types.rdb ?
file:///etc/opt/ure/types.rdb ?file:///home/michael/.ure/types.rdb
<file:///data/opt/OOInstall/program/types>*    ") at
cppuhelper/source/defaultbootstrap.cxx:2132

#6  cppu::defaultBootstrap_InitialComponentContext
(iniUri="file:///data/opt/OOInstall/ure/lib/unorc") at
cppuhelper/source/defaultbootstrap.cxx:2181
```

- Massive improvement to debugging speed ...
- Uses Tom Tromey's awesome gdb / python work – checkout:
  - http://sourceware.org/gdb/wiki/PythonGdbTutorial
- Many thanks to *David Tardon* (RedHat)
  - This makes life incredibly sweeter … and quicker.

LibreOffice

Pure Hard-working Manual QA ...

- Amazing work done by the QA team …
  - Particular thanks to: Rainer Bielenfeld, Joel Madeo, Joren De Cuper, Petr Mladek, Urmas and many others ...
- Triaging incoming bugs …
  - Resolving duplicates
  - Correctly tagging / marking them 'most annoying'
  - Please get involved – see the BugTriage
- Running master builds …

But Does it work ?

Can you really do significant, gratuitous, esthetic code change and re-factoring, and have fun without dying of regressions ?

Metrics help – watching the stats...

- Always nice to have a longer series but ... looks good:
- Generate these numbers for the ESC meeting each week ...

Regression bugs over time

Open
Closed

Easier to tag regressions in the Bugzilla Assistant.

LibreOffice
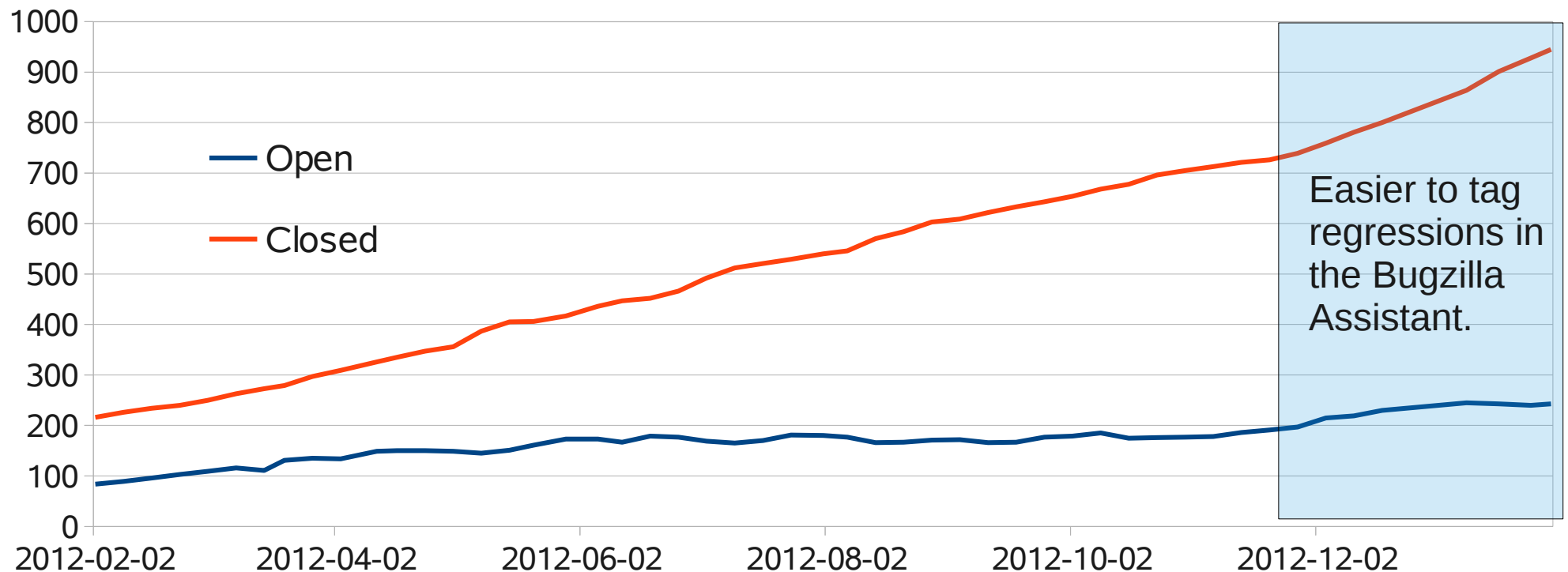
More prosaically in numbers ...

- We average ~1570 commits per month in the last 6 months
  - ~50 commits per day
- We've tracked ~450 regressions in the last 6 months
  - 2.5 per day
- **Iff** 1:1 commits:regressions
  - ~5% of commits cause a regression
  - Prolly fewer – some may cause several.
- Many regressions are fixed before we ship
  - More research appreciated on escaped regressions …
  - some are inevitable
- Even one regression is too many
  - But cost/benefit wise we seem to do well.

LibreOffice

Most Annoying bugs ...

- QA's prioritisation of the most serious bugs out there ...
  - Reflected in a set of tracker issues
- Shows a remarkable similarity to the regressions (why?)

Total Most Annoying Bugs across all versions
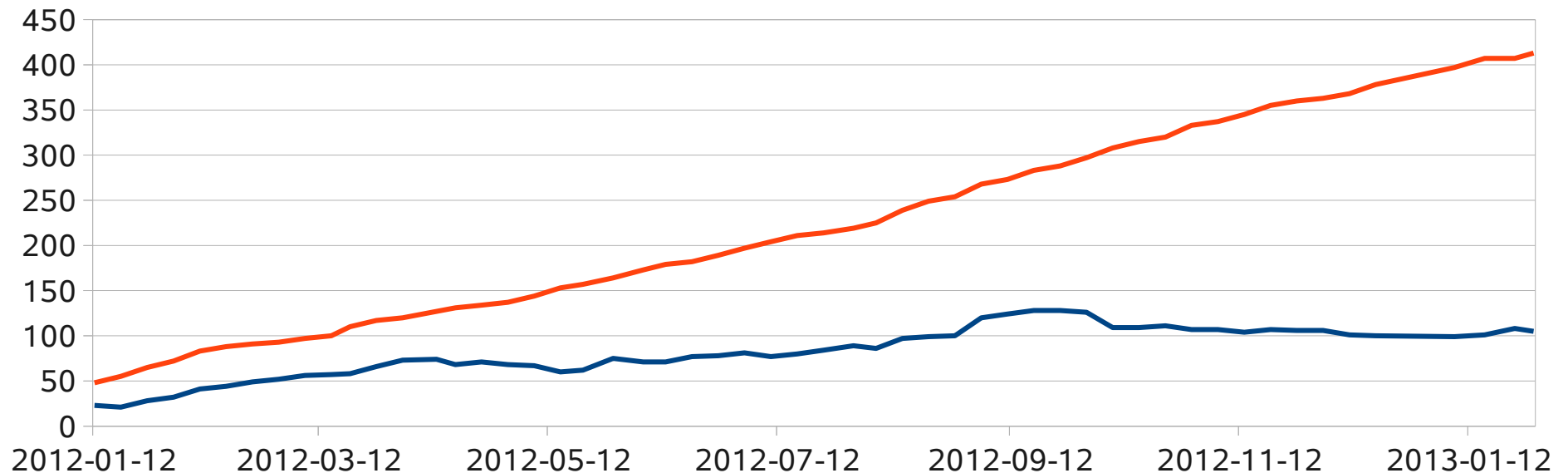
LibreOffice

A crazy picture of the space ...

- A hugely multi-variate problem space transected

Incremental Quality (vertical axis)
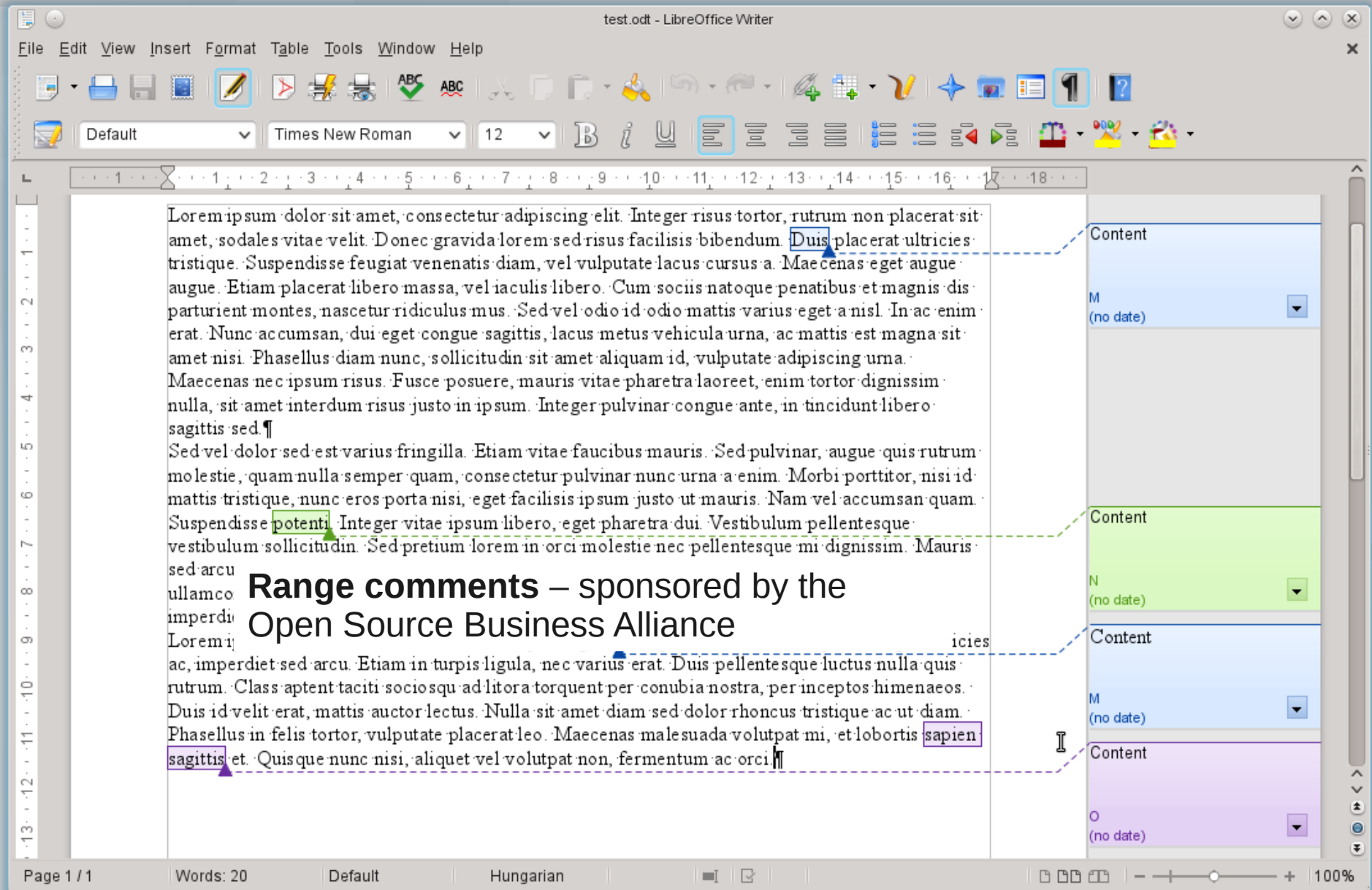
Bugs get fixed faster than you can create them.

Quality through Obsolesence: the 18month freeze ...

Enterprise release changes process … To increase quality ?

B

A

Death for communities

Worst of all worlds: slow release, lower process, higher change

Slow and careful change, Process, Infrequent releases

High Speed of change Low process Rapid release
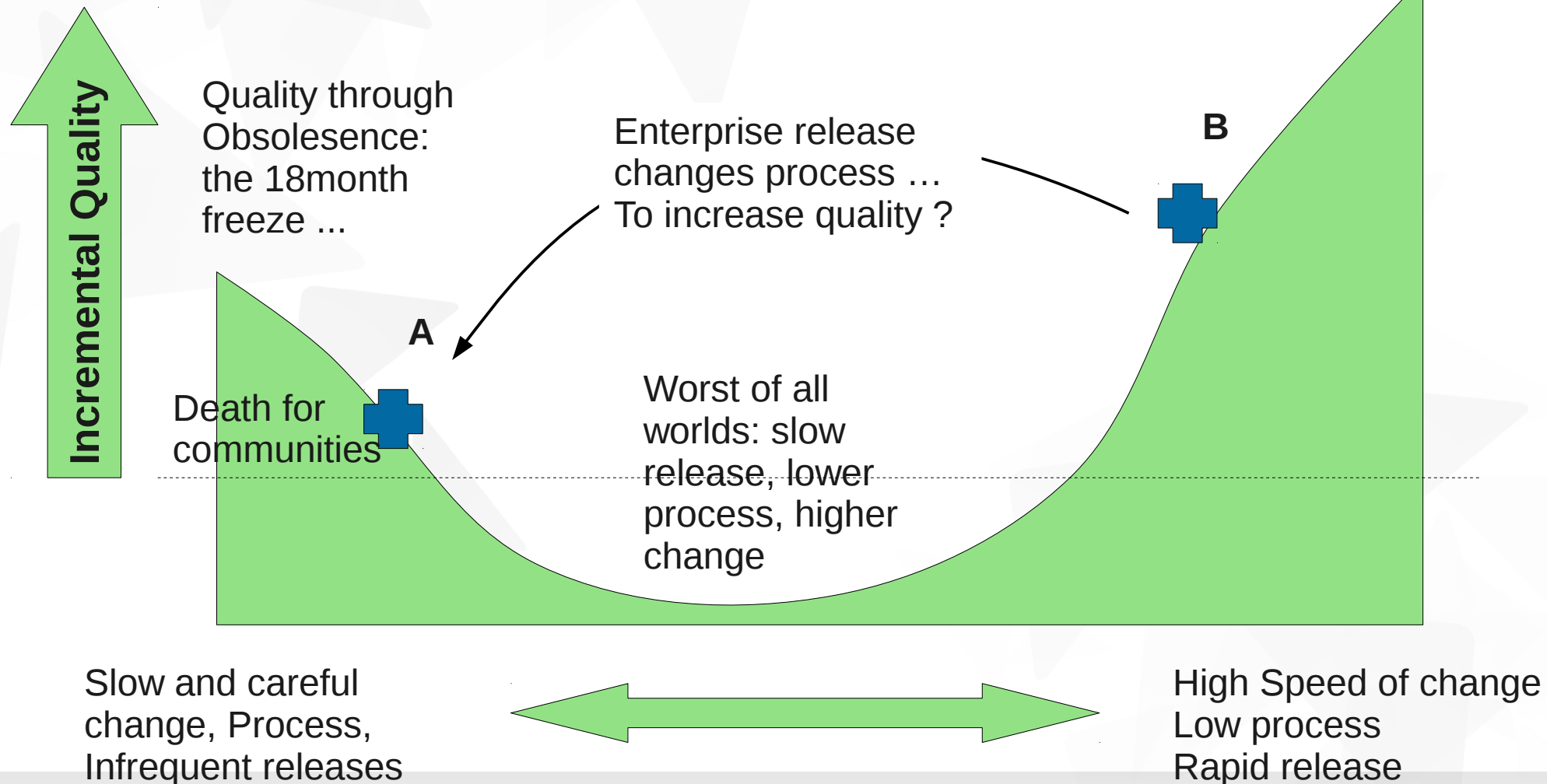
LibreOffice

Applying the fun to features: LibreOffice 4.0 – due next week.

a very quick & partial snapshot of ***some*** *of the new features*

Interoperability in 4.0

Core interoperability features

Range comments – sponsored by the Open Source Business Alliance

43
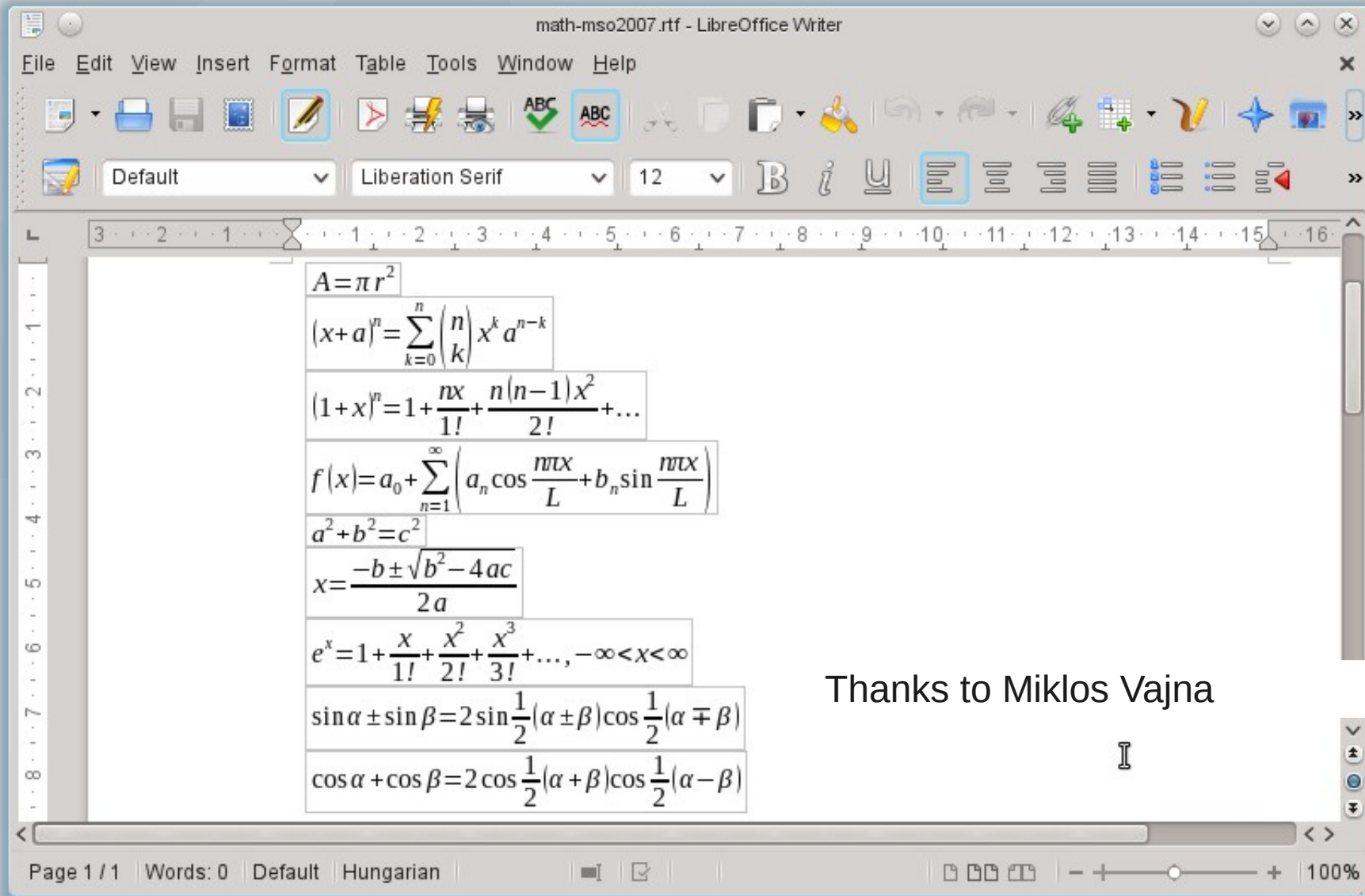
RTF: Drawing Object import

4.0

Same document in
LibreOffice 3.6

RTF improved eg. Formulae

Thanks to Miklos Vajna

DOCX – ink annotation import
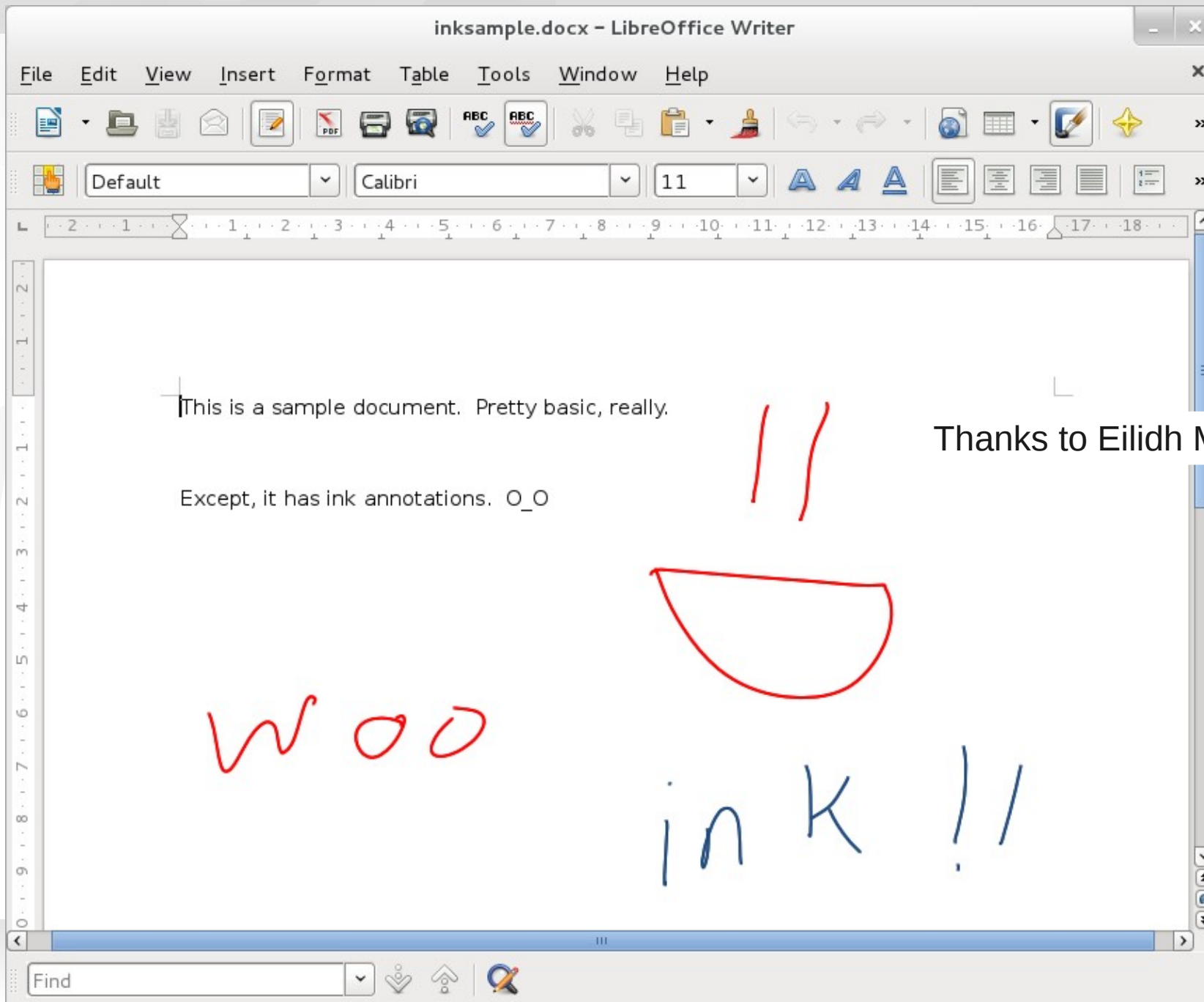
Thanks to Eilidh McAdam / Lanedo

CMIS: Sharepoint / Alfresco / Nuxeo ...

- Using the CMIS protocol – load / save / checkout to your favourite content / document management system:
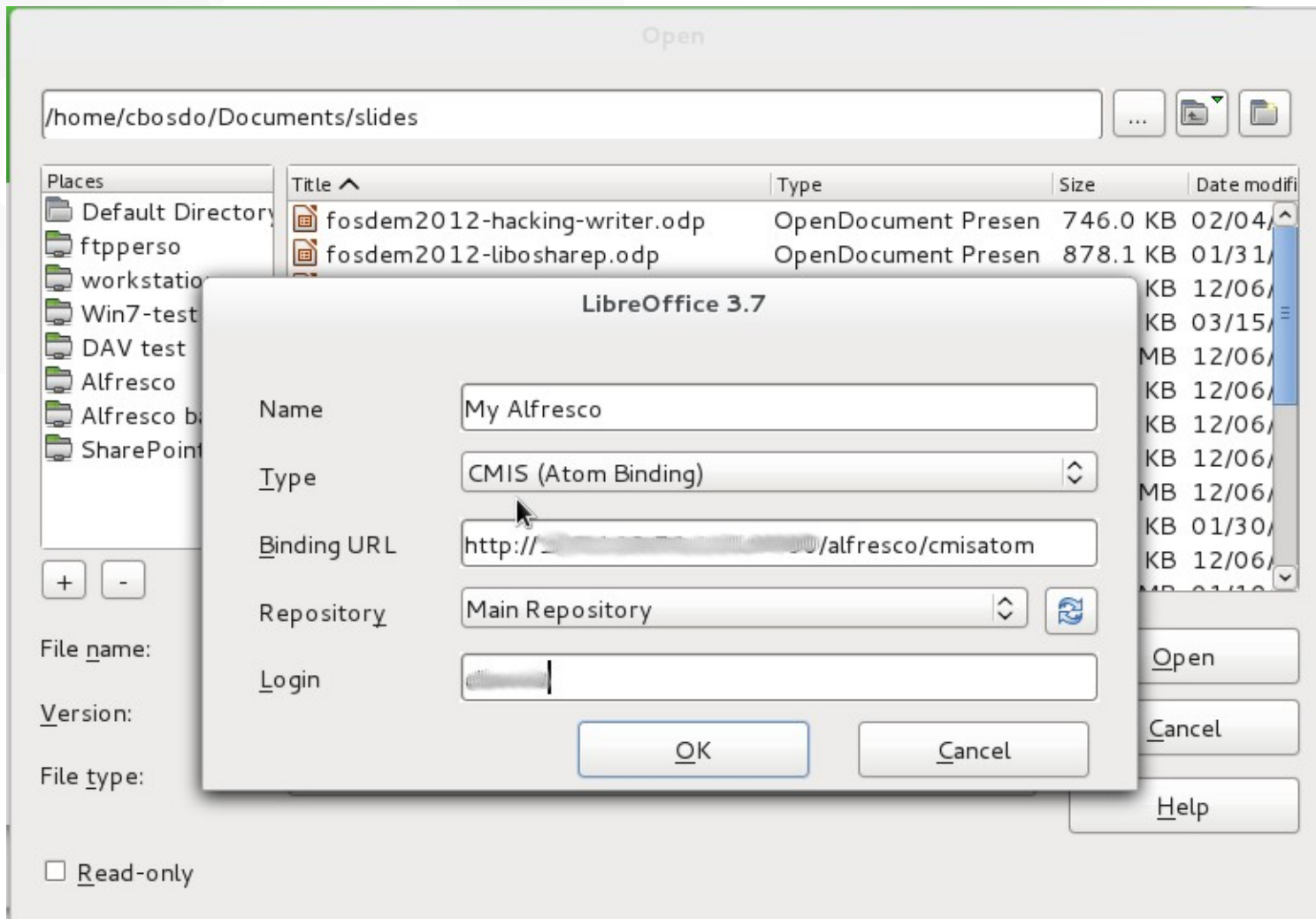
Proprietary → ODF continues
Wordperfect, Works, Visio, Corel Draw ...

Microsoft Publisher import thanks to
Brennan Vincent (GSOC)
Valek Filippov
Fridrich Strba

Visio: all file formats now imported

Thanks to
Fridrich Strba
Valek Filippov

Includes the **just-release** Visio 2013 format.
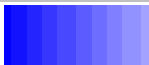
Calc improvements …
a few details on one component

Arbitrary XML → spreadsheet
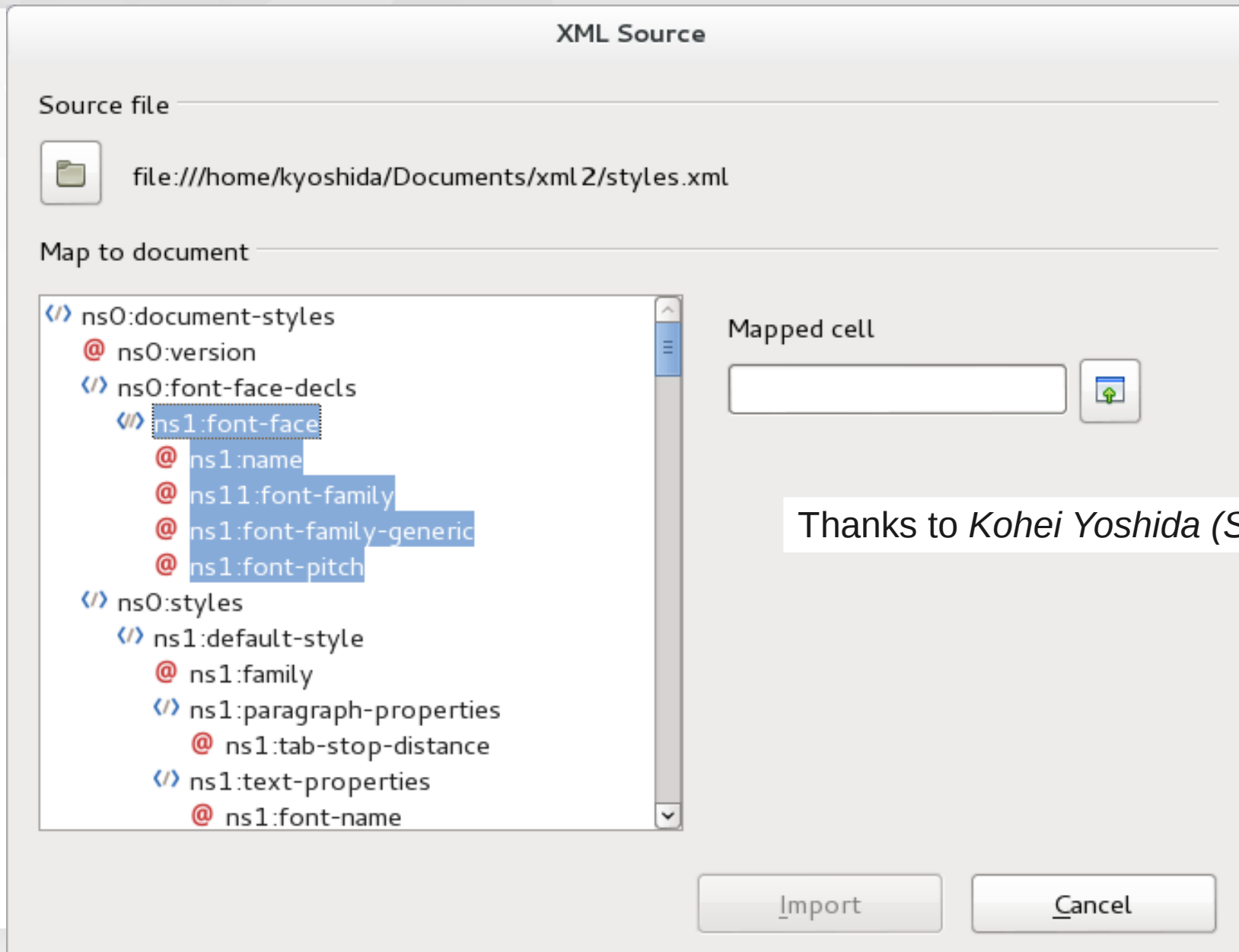
Thanks to *Kohei Yoshida (SUSE)*

LibreOffice

Conditional fmts: bars + icons

	A	B	C
1	**Name**	**Score A**	**Score B**
2	Mary	16.38%	🏳 65.98%
3	Bob	22.35%	🏳 32.36%
4	Abdul	22.61%	🚩 13.90%
5	Holly	49.00%	🏳 77.50%
6	Oak	17.57%	🏳 18.98%
7			

A1 = Name

Many thanks to:
Markus Mohrhard &
Stefan 'Astron' Knorr

Stock option pricing formulae ...

Fun improvements …
Those features that make life better

Android remote control ...

- **Thanks to** *Andrzej J. R. Hunt (and GSOC)*
- **Use your smart-phone as a powerful remote control**
  - see your notes
  - switch slides
  - up-load your slides ?

LibreOffice
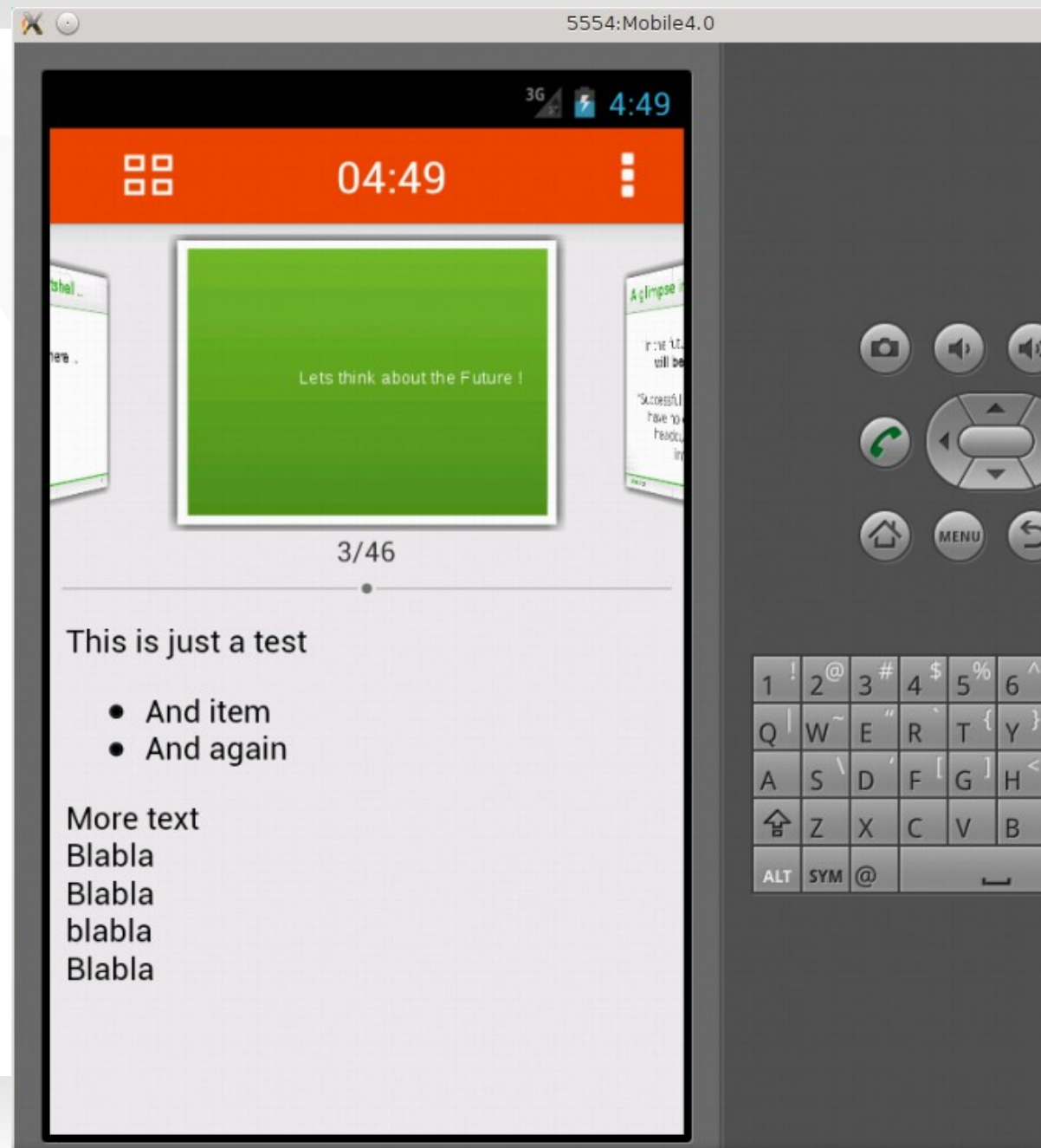
Android remote control ...

- Pretty slide sorter / selector
- Clock / count-down etc.
- Potential future work
  - accelerometer / laser pointer ?
  - fuse with viewer code to allow projection from a tablet ?
- Should ship in 3.7/4.0

Schools: LibreLogo integration ...

A tiny Python implementation

 If schools teach 'typing' instead of programming: ensure they have no excuse: add a localised, pretty training language in the office-suite !

Thanks to Laszlo Nemeth
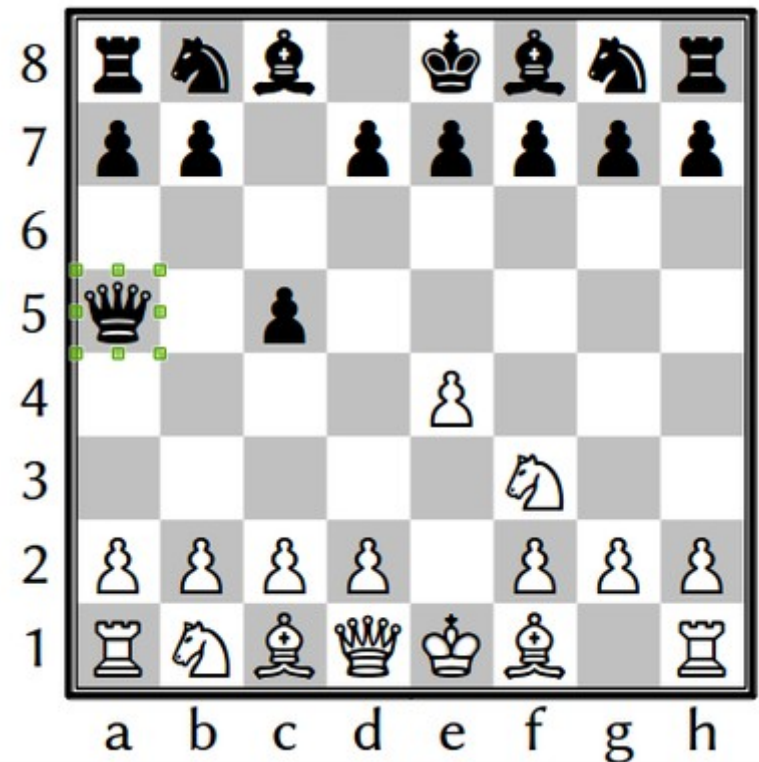
UI improvements …
or how we're looking less awful

Improved graphics scaling

OLD:
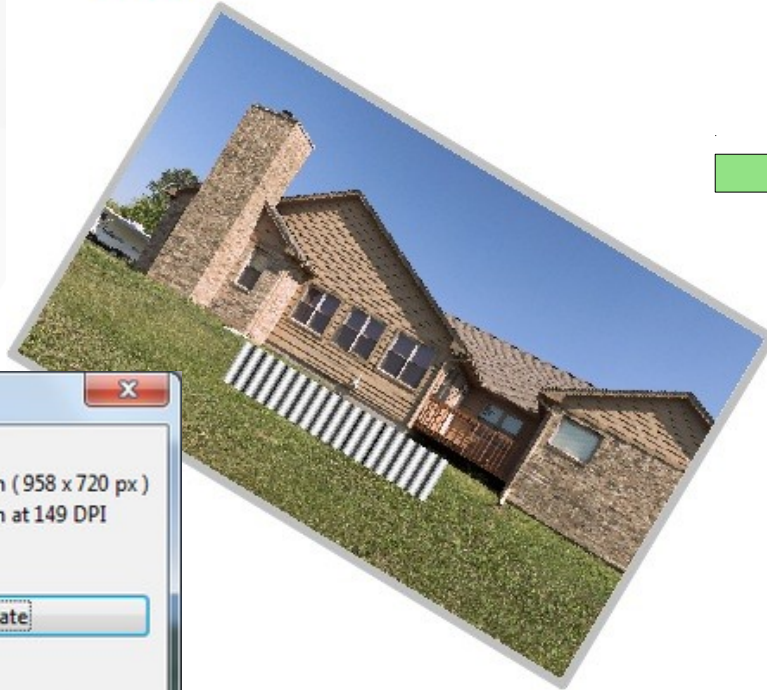
NEW:
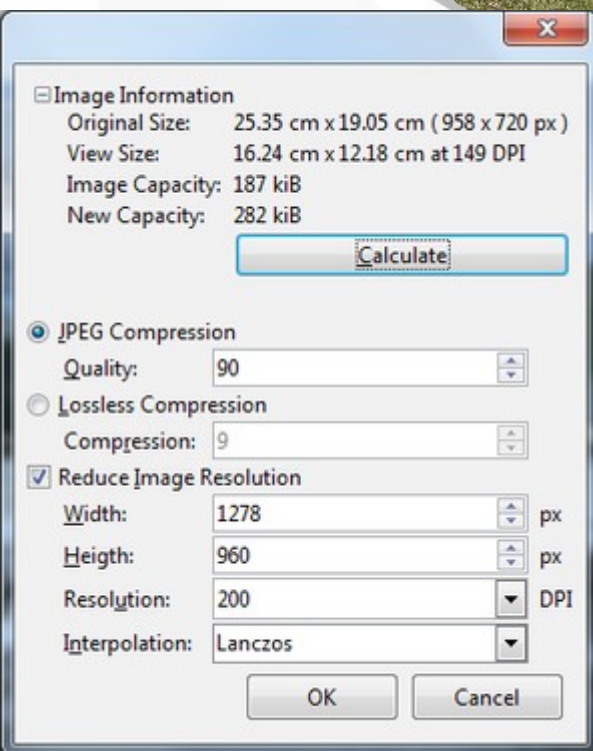
Higher quality image rendering

Improved image smoothing tool

Re-sizing, re-scaling, and adapting compression of embedded objects now possible during editing.

Thanks to Tomaž Vajngerl

LibreOffice

Snapshots of the great work from KACST : Motah
Getting Arabic & Right-To-Left right.

Funky OLE object dragging fixed.
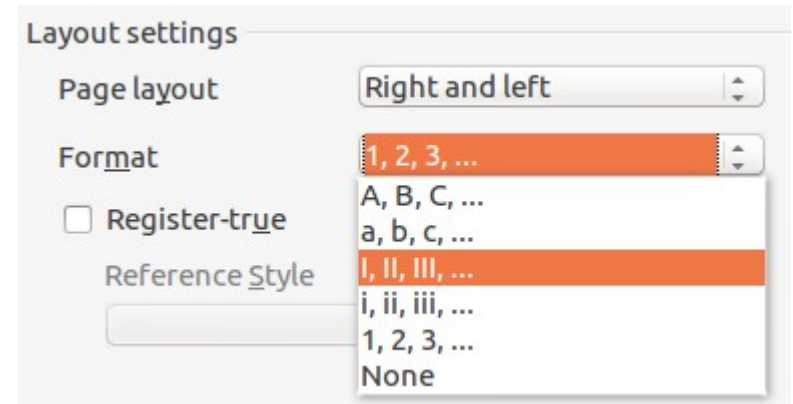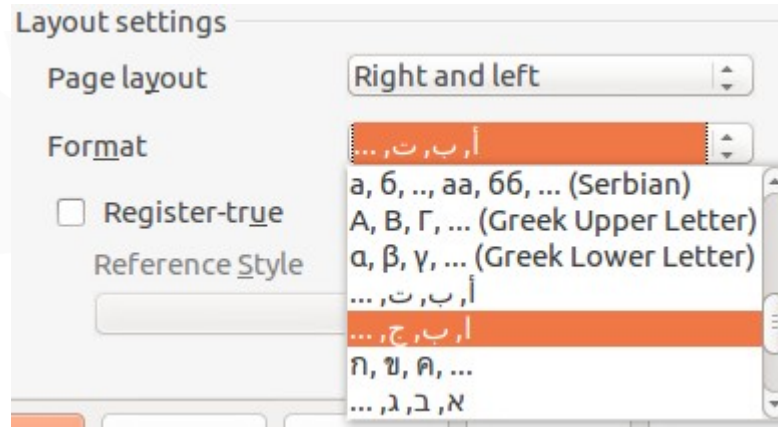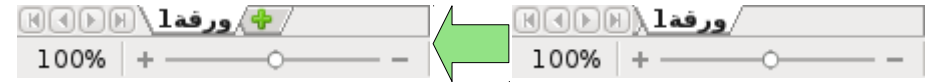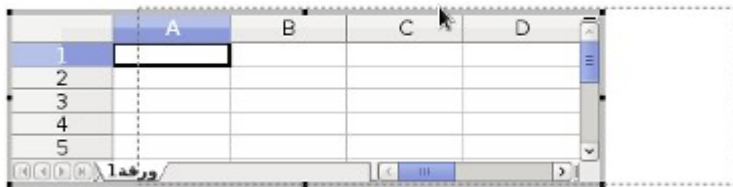
And much more; see above link.

New Template selection UI

- Thanks to

  Rafael Dominguez
  (GSOC)
  Cedric Bosdonnat
  & Design team

- Making template selection and search prettier & simpler.
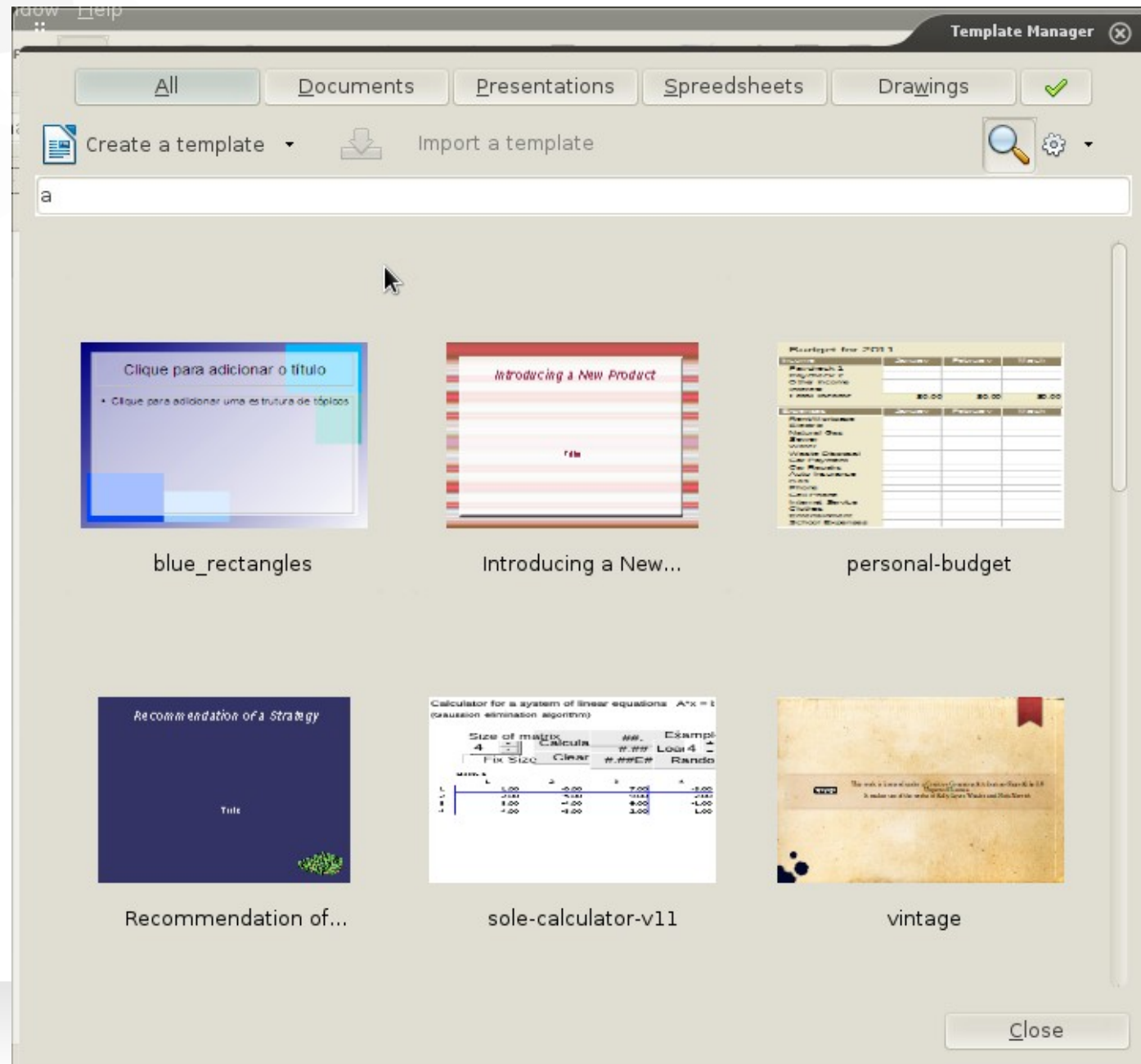
LibreOffice

Style previews in drop-down

- Thanks to *Jan Holesovsky*

- Making styles easier to use.

Unity menu integration
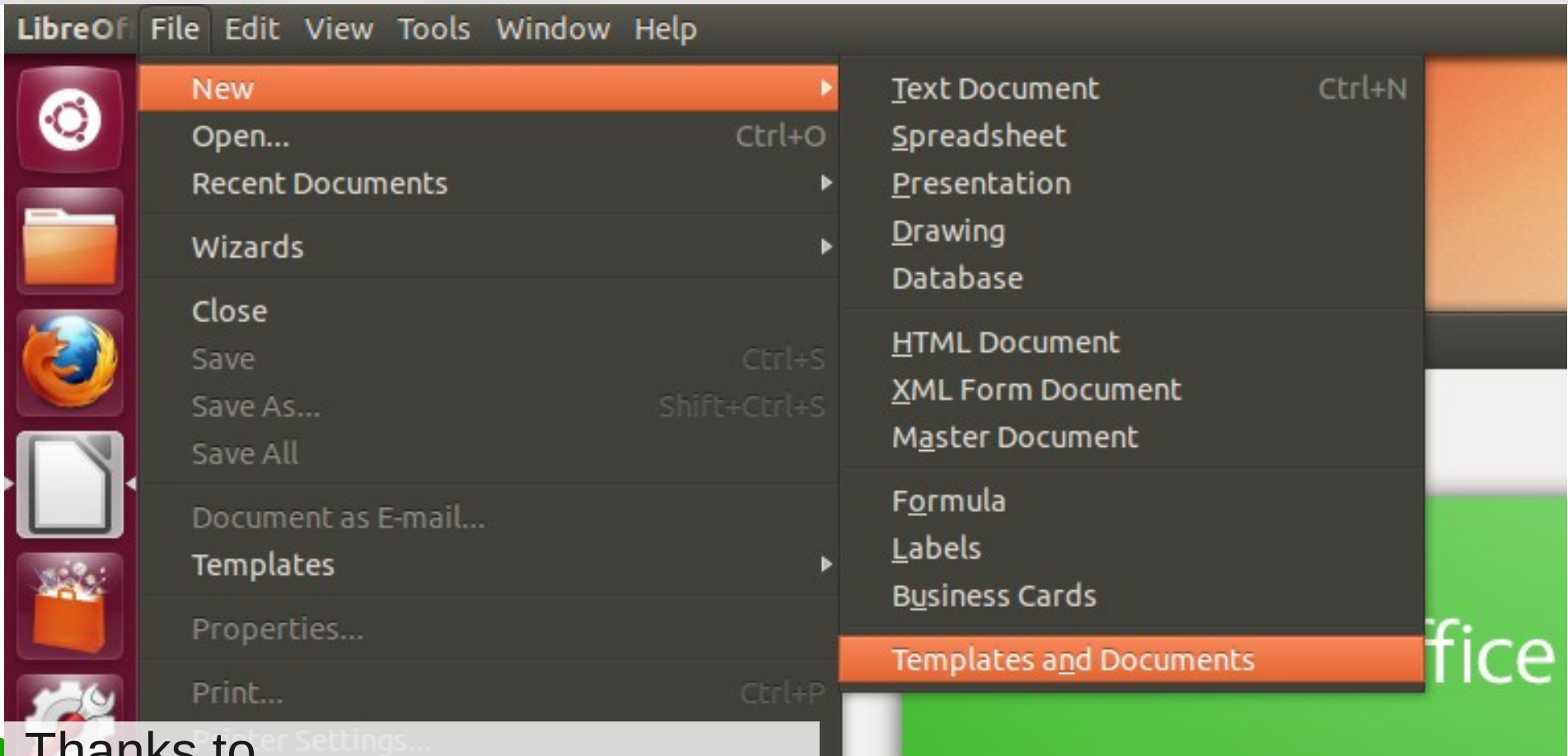
Thanks to
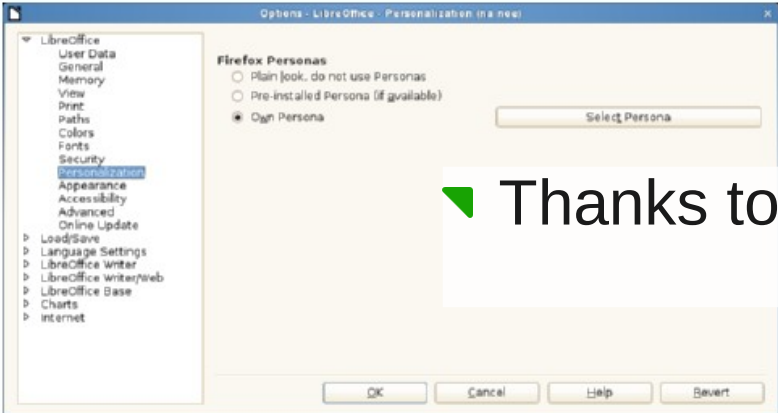(*Antonio Fernandez, Björn Michaelsen, Alberto Ruiz, Ryan Lortie, Ted Gould*)

Personas ... shared with Firefox

Thanks to (*Jan Holesovsky*)

Conclusions & Thanks …

- **LibreOffice continues to grow & execute**
  - paying down decades of technical debt
  - Adding features / function and re-factoring
- **Accelerating change using sharp tools & good QA**
  - **without** introducing **regression spikes**
- **We have fertile work for new contributors**
  - Want to make a real difference ? Apply here ...
- **Thank you for your support !**

Oh, that my words were recorded, that they were written on a scroll, that they were inscribed with an iron tool on lead, or engraved in rock for ever! I know that my Redeemer lives, and that in the end he will stand upon the earth. And though this body has been destroyed yet in my flesh I will see God, I myself will see him, with my own eyes - I and not another. How my heart yearns within me. - Job 19: 23-27