

LiMux



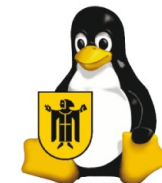
Landeshauptstadt
München

■ One Year with OpenOffice.org

Experiences in accessing ODF file contents directly

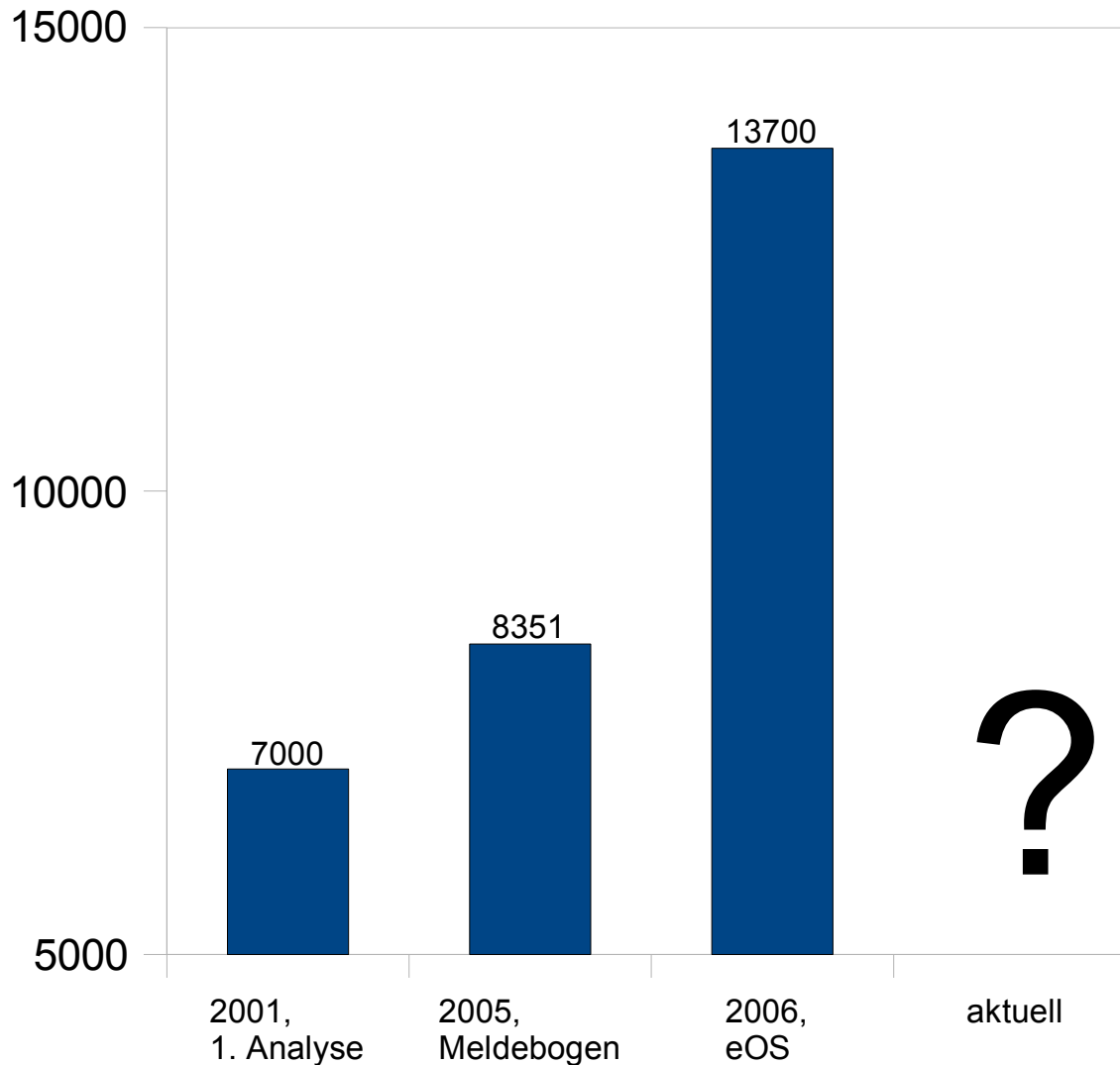
Florian Schießl

29th october 2007



LiMux
Die IT-Evolution

The requirements of the migration to OpenOffice.org



- About 14,000 objects are to be migrated
 - Some software solution redundancies - different solutions for equal problems
- **Migration is the opportunity to consolidate**

How LiMux benefits from ODF

Implementing ODF is a concomitant feature of our migration

- A feature to better cope with day-to-day document processing, not only a necessity
- Solving problems and tasks directly in a transparent and effective way using a scripting framework

From a Munich-view: Better than before

ODF Scripting Framework

Helps to work with ODF content directly

Should be able to handle ZIP archives and utf8 based XML files

Experience with proprietary python and bash based framework

Currently testing CPAN OpenOffice-OOoDoc
(<http://search.cpan.org/dist/OpenOffice-OOoDoc>)

Example 1: Handling of Macros

Task

- No longer required fragments of macros, typically in batch converted documents
- Extract, analyze and possibly remove embedded basic macros

Solution

- Access the macro content stored under `/Basic` by using an ODF scripting framework
- Remove the no longer required macros by deleting the folder in the ODF archive

Example 2: Customizing Text Modules

Task

- Migration of a text module system with thousands of text modules
- Changing the integrated wild-cards

Solution

- Replace the old wild-cards with new ones by searching and replacing within the `content.xml` of the document – very simple

Example 3: Presentation with large filesize due to embedded highres-pictures

Task

- Novices often use embedded high-resolution images in presentations, wasting resources (network, file server, ...)
- Images should automatically be shrunk to a reasonable size

Solution

- Access the images in `/Pictures` by using our ODF scripting framework
- Manipulate them to fit our requirements

Example 4: Inserting database values by scanning for special placeholders

Task

- Flexible and performant mechanism to fill template files with values, e.g. resulting from database queries
- Avoid OpenOffice.org as a server component

Solution

- Searching and replacing the wildcards for database fields within the `content.xml` of the document – very simple, very fast

Conclusion

We are the owner of our assets

- We now have the ability to really work with our documents
- No black boxes anymore
- Vendor independent and open standard

MS OOXML as an alternative?

- For scripting purposes: No, look at http://en.wikipedia.org/wiki/Ooxml#Technical_criticisms (e.g. inconsistent local conventions, non-standard language codes and color names etc.)
- In general: We don't need another standard. Competition of implementations, not standards!

Any questions?

Web: www.muenchen.de/limux

E-Mail: limux@muenchen.de

Devblog: planetlimux.org

Special thanks to Harald Kagerer from our extended office support center for his great help.

LiMux



Landeshauptstadt
München

- **One Year with OpenOffice.org**
Experiences in accessing ODF file contents directly

Thank you for your attention



LiMux
Die IT-Evolution