

## Going Tubes – collaborative editing

- ▼ Eike Rathke (erAck)
- ▼ Senior Software Engineer at Red Hat, Inc.

# Agenda

- ▼ About the speaker
- ▼ Why collaborative editing
- ▼ Thoughts and decision making
- ▼ Getting dirty
- ▼ Seeing light
- ▼ Opening up
- ▼ The results
- ▼ Demo
- ▼ Q&A

# About the Speaker

- ▼ Eike Rathke, known on the net as erAck
- ▼ Based in Hamburg, Germany
- ▼ Worked on StarOffice from 1993 to 2000 for Star Division
- ▼ Worked on OpenOffice.org from 2000 to 2011 for Sun Microsystems and one other company
- ▼ Works on LibreOffice since 2011, employed by Red Hat, Inc.
- ▼ Areas of expertise:
  - ▼ Calc core, formula compiler and interpreter
  - ▼ number formatter/scanner
  - ▼ i18n framework, locale data
- ▼ Also mentor and knowledge spreader whenever possible

## Why collaborative editing?

In an office suite? Can't everyone use a web service or something in the cloud?

# Implement collaborative editing because ...

- ▼ Not everyone trusts an arbitrary service provider
- ▼ Enterprises do not want their data reside in the internet
- ▼ Individual persons neither
- ▼ Users do not want to register accounts with umpteen service providers just because each contact uses a different one
- ▼ Web services may vanish
- ▼ Web services do not offer the full feature set of an office application, and all are different
- ▼ Users want to be able to save in the preferred file format (which of course is ODF ;-)
- ▼ Users want control over own data

## How to start?

Not missing the forest for the trees.

# Decision making

- ▼ Use an already available framework for communication that
  - ▼ uses standard protocols
  - ▼ is not bound to a specific server implementation
  - ▼ can be used over internet and in intranet
- ▼ => *Telepathy*, the framework used by *Empathy* (GNOME) and *KDE Telepathy* that offers
  - ▼ standard Jabber/XMPP protocol
  - ▼ contact-to-contact networking with *Tubes* over XMPP
  - ▼ file transfer
- ▼ First LibreOffice application to use collaboration is Calc
  - ▼ spreadsheet data entry benefits most

# Thoughts about Calc

- ▼ Avoid complete implementation of a special collaboration mode having to rewrite all data entry and operations
- ▼ Original idea was to use change tracking and send/receive change sets, which was discarded because
  - ▼ change tracking itself is limited in its capabilities
  - ▼ too much overhead to manage the changes before they are applied on the sender side while the change set is created only after the actual change
  - ▼ receiver would have to construct undo actions of change sets



# Thoughts about Calc, continued

- ▼ Instead, send/receive actions and process them as usual on each side
  - ▼ benefit: Undo/Redo integration
  - ▼ send as clear text
    - ▼ easily extendable
    - ▼ avoids endianness and binary traps
    - ▼ easier to log and debug
    - ▼ numeric and formula input need to be parsed on the sender side for locale dependent characteristics
      - ▼ transmit as defined in ODF OpenFormula (ODFF)
  - ▼ requires stricter separation between view, model and controller than it is currently the case

In the beginning ...

... there was chaos.

Getting dirty.

# Telepathy

- ▼ libtelepathy-glib is complex
- ▼ Provides hundreds of functions, many with asynchronous callbacks to be implemented
- ▼ Diving into documentation is challenging
  - ▼ docs are sparse on what needs to be done from an application's point of view
- ▼ Half-way running framework after 4-5 weeks
  - ▼ picking existing (in Empathy) account from the account manager worked
  - ▼ connecting to D-Bus worked
  - ▼ starting a session, creating a channel to a buddy, offering a tube worked

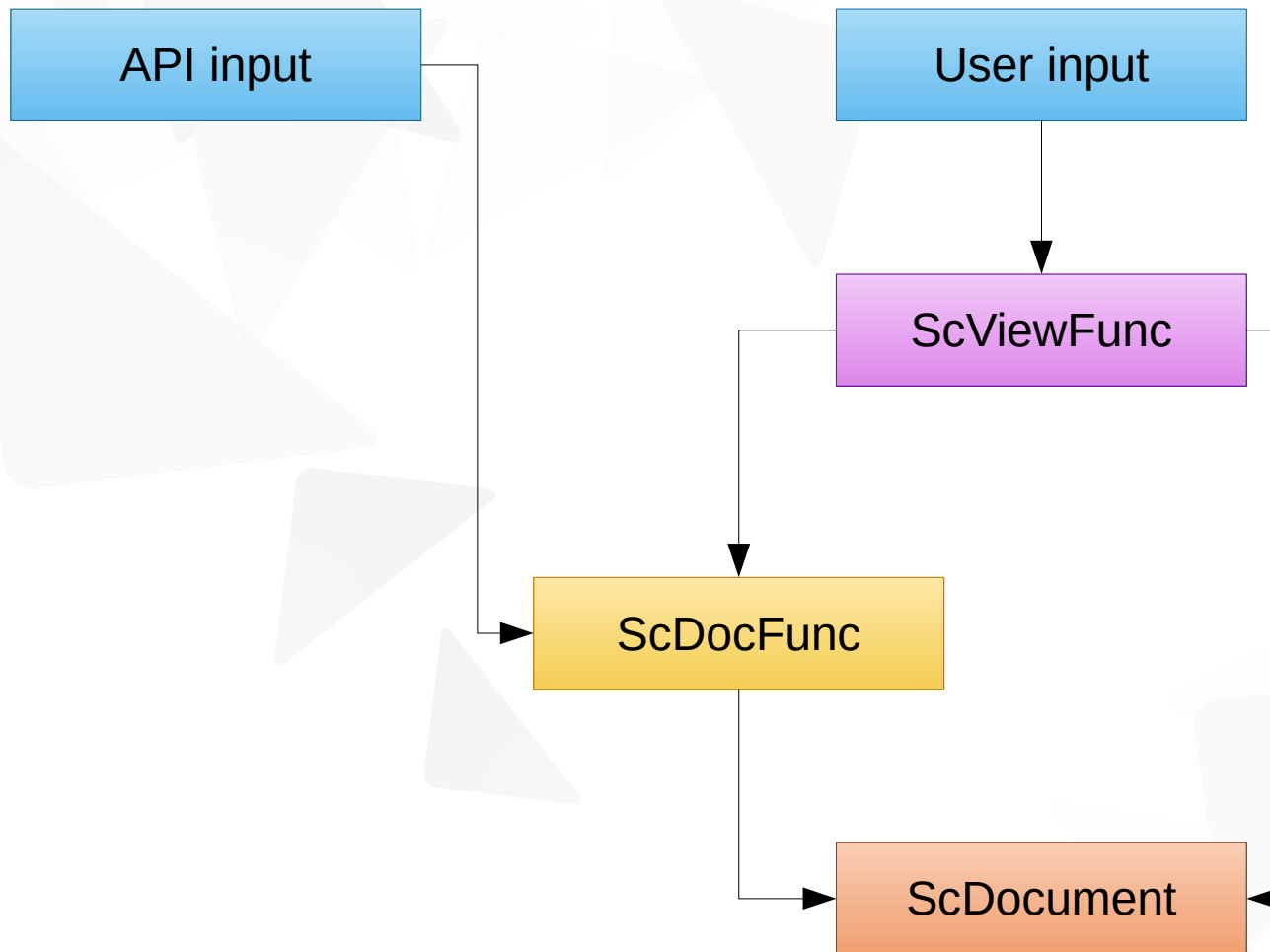
# Seeing light (at the end of a Tube)

In the middle of everywhere.

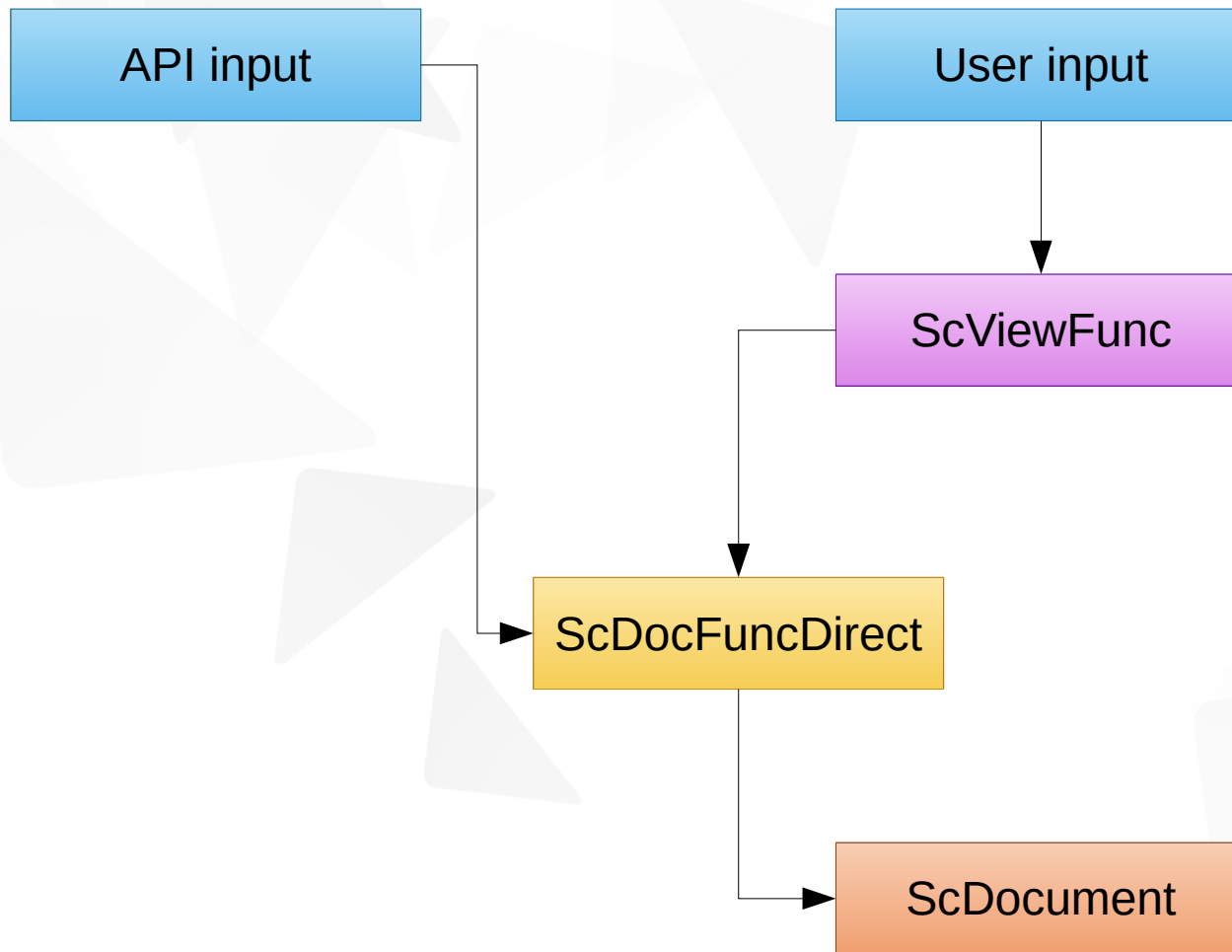
# Hackfest @ Collabora

- ▼ The Collabora guys (maintainers of Telepathy) chimed in
- ▼ Will Thompson, Michael Meeks and myself gathered at Cambridge and locked ourselves in a room for 3 days in March
- ▼ For my Telepathy/Tubes related problems Will immediately came up with solutions
- ▼ Michael hacked a transfer protocol prototype and split ScDocFunc
- ▼ I myself integrated the TeleManager (LibO implementation managing Telepathy and Tubes) to Calc
- ▼ Crude “user interface”, session was started and file was transferred after having typed “saveme” into a cell

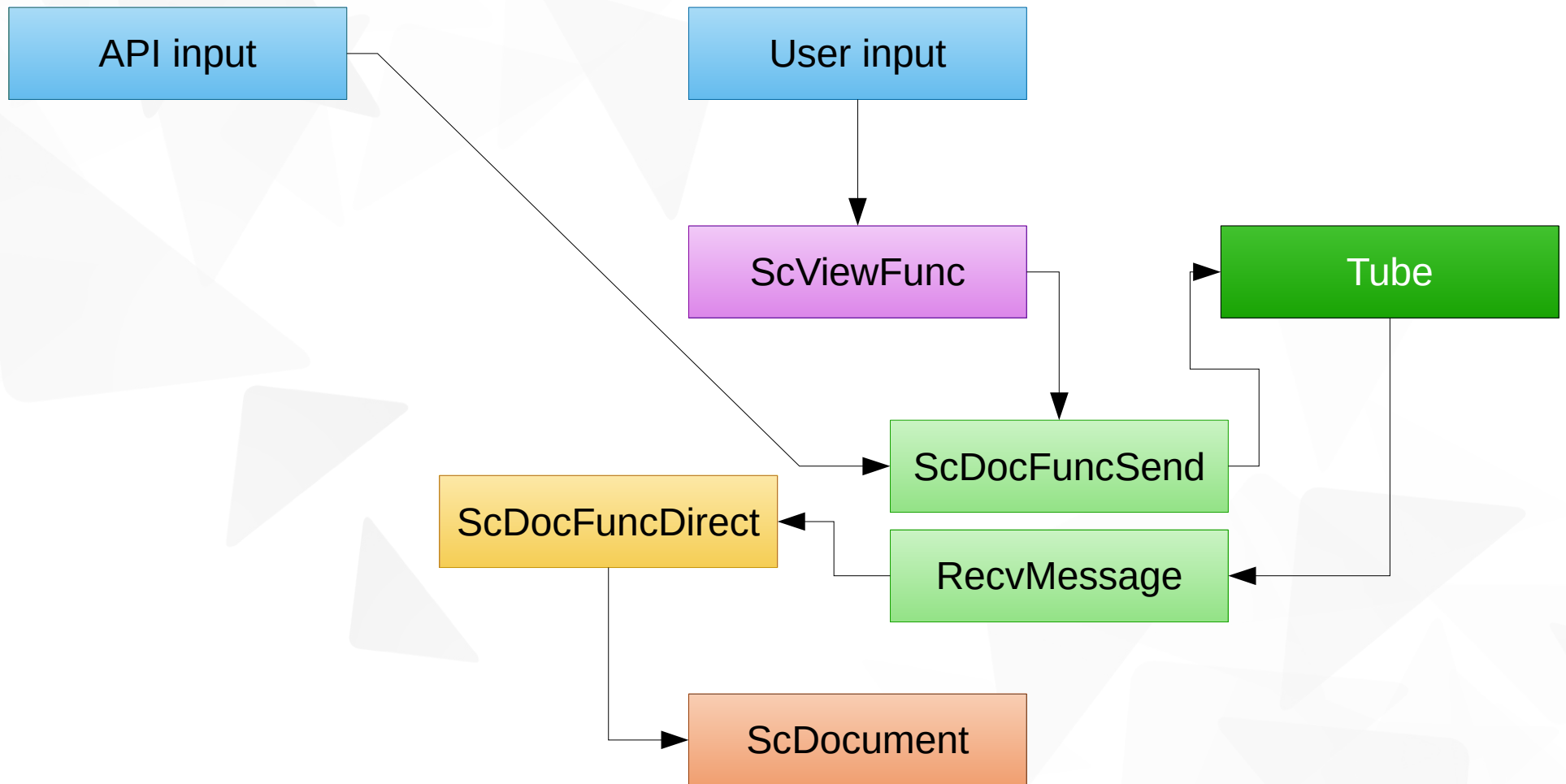
# Calc input flow



# Calc input flow, modified direct



# Calc input flow, modified tubed





# Opening up

Google Summer of Code 2012

# Google Summer of Code 2012

- ▼ Matúš Kukan
- ▼ Implemented menu entry in Calc as File → Collaboration
  - ▼ automatically added when build is configured with `--enable-telepathy`
- ▼ Changed the contacts dialog displayed from that menu
  - ▼ display avatars
  - ▼ select an alias/contact pair and start a session
  - ▼ demo session can be started from the same dialog without the need for Jabber accounts or server
  - ▼ updates when contact list changes
- ▼ Removed the hack of needing `LIBO_TUBES=slave / master` environment variables to distinguish clients

# Google Summer of Code 2012

- ▼ Made editing tied to a single document, previous prototype received changes in all open documents
- ▼ Automatically send the current file and associate the session when starting a 1-to-1 session
- ▼ Enabled MUC (Multi-User-Channel) for many-to-many sessions, inviting is possible
- ▼ When collaborating notify if contact leaves
- ▼ Send back edit events to ensure consistent strict ordering
- ▼ Restructured code and moved collaboration fragments from module *sc* to module *tubes* where they belong
- ▼ Code using *tubes* does not need to link to *libtelepathy-glib* anymore, only 2-3 headers are exported from module *tubes*
- ▼ Adapted unit tests to new code

# The results

Technical pointers.

# Where to find things

- ▼ All Telepathy/Tubes related code in module *tubes*
- ▼ Check `tubes/README`
- ▼ Build with `--enable-telepathy` configure switch
  - ▼ needs `libtelepathy-glib 0.18` at least
  - ▼ `#define ENABLE_TELEPATHY` is set for use in code for the rare case where it needs to distinguish
- ▼ Calc collaboration specific code
  - ▼ resides in `sc/source/ui/collab/`
  - ▼ portions (grep for `ENABLE_TELEPATHY`) in
    - ▼ `sc/source/ui/inc/docsh.hxx`
    - ▼ `sc/source/ui/docshell/docsh.cxx`
    - ▼ `sc/source/ui/view/cellsh3.cxx`

# Demo

Pause those boring slides.

Questions?

I hope to be able to answer.

# BERLIN 2012 CONFERENCE

17th-19th October

## Thank you ... for hacking LibreOffice!

- See you on IRC `#libreoffice-dev` at freenode
- or [libreoffice@lists.freedesktop.org](mailto:libreoffice@lists.freedesktop.org) mailing list



All text and image content in this document is licensed under the [Creative Commons Attribution-Share Alike 3.0 License](https://creativecommons.org/licenses/by-sa/3.0/) (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](#).