# KMail 2

## The Road to Akonadi

Thomas McGuire

KDAB

July 5th 2009

# Outline

1. Why do we need Akonadi in KMail?
2. What is Akonadi?
3. How will KMail be ported?
4. When will KMail 2 be released?

# About Me

## About Me

- Student of computer science at the University of Siegen
- KDE developer since 2007
- KMail Maintainer since 2008
- KDAB employee

# Outline

1. Why do we need Akonadi in KMail?
2. What is Akonadi?
3. How will KMail be ported?
4. When will KMail 2 be released?

# About KMail

## About KMail

- Official KDE Mail Client, part of Kontact
- Now over 12 years old
- Geared towards power users
- Second highest bug count on bugs.kde.org

## High Bug Count?

- Many bugs caused by the aging storage layer
- Bug count also reflects popularity
- KMail works OK except for corner cases

# History of KMail

A look into the past

## The Early Days

- Mail handled by system programs like fetchmail and sendmail
- Filtering also handled by the system with procmail
- Mail fetching also with POP3 and sending with SMTP
- KMail only needed to read mail from local maildir or mbox folders

## Design

- Mails could be read fast and synchronously
- Clean internal design, based on that assumption

# The Advance of IMAP

## How the clean design of KMail was corroded

### IMAP

- **I**nternet **M**essage **A**ccess **P**rotocol
- Mails are kept on the server, not stored locally
- Many advantages over POP3

### Adding IMAP support to KMail

- Mails can no longer be read synchronously
- Folder listings are not synchronous either
- Basic KMail design assumptions broken
- Workarounds added to make IMAP work

# Groupware support

Yet more workarounds from the initial design

## KResources

- KResource framework provide support for addressbook and calendar backends
- Used by KOrganizer and KAddressbook
- Has some design issues as well

## Kolab & Scalix

- Calendar events and contacts stored as attachments to mails in IMAP folders
- Implementing IMAP in KResources too impractical
- Workaround: The Kolab and Scalix KResources talk to KMail over D-Bus

# Other pain points

## More trouble with the design of the storage layer

### Other problems

- Index files
- Easy to get invalid pointers to mails
- No Extendibility
- Code monolithic with many internal dependencies
- Uses outdated mimelib instead of KMime for mail parsing
- Filtering blocks the GUI (**most hated KDE bug!**)

# The need for a new architecture

Something needs to be done about the old storage layer

## The Problems

- Complicated and brittle storage layer
- IMAP and Groupware support an afterthought
- KResource framework also has design issues
- Experienced developers keep Kontact working, but maintenance is difficult

## The Solution

- **Akonadi!**
- Started in 2006
- Getting ready now

# Outline

1 Why do we need Akonadi in KMail?

2 What is Akonadi?

3 How will KMail be ported?

4 When will KMail 2 be released?

# Akonadi

## Overview

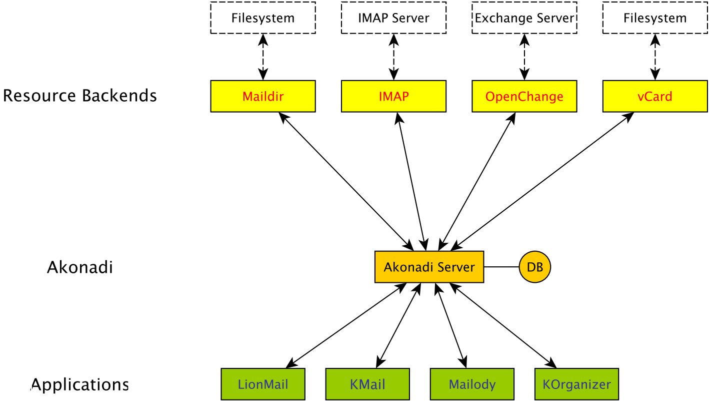
### Design

- Central point to access PIM data like mails, contacts and events
- Think of it as a proxy or a cache

  Caching makes things like disconnected IMAP possible and also replaces the old index files
- Applications talk to Akonadi server, which uses a DB to store data
- Akonadi server talks to *resources*, which are the real backends accessing e.g. maildir, IMAP or Exchange

# Time for a diagram!

## Because a picture says more than 1024 words



# Akonadi

## How it will solve KMail’s problems

### Advantages

- Clean design from the beginning

  No more problems with IMAP not fitting in
- Central place to access all PIM data

  KMail not started when accessing the addressbook
- Modular

  Easier to maintain and change
- Robust by using process separation
- Automatic Testing

  Lower chances of regressions

# Outline

1. Why do we need Akonadi in KMail?
2. What is Akonadi?
3. How will KMail be ported?
4. When will KMail 2 be released?

# Porting KMail

## Overview

### Porting

- Akonadi can be (and was) developed in parallel to KMail

  Modular design allows working on different aspects without dependencies
- Mail parts of Akonadi already used by Mailody and LionMail

  Akonadi has proved itself in the field
- Actual porting in SVN branch: *work/akonadi-ports*

  Your data will always be safe
- Porting will be done piece by piece

# Porting Piece by Piece

## What is already there

### Done

- Akonadi Infrastructure
  - Server
  - Libraries
- Some Resources
  - Maildir
  - MBox
  - OpenChange
  - vCard, iCal, ...
- Others
  - Akonadi-based mail sending

# Porting Piece by Piece

## What is being worked on

### In Progress

- IMAP Resource

  Will support IMAP IDLE (Push E-Mail)
- Kolab Resource
- POP3 Resource

  Will support deleting on server
- Filtering System
  - Will solve most hated KDE bug
  - GSoC project by Szymon
- Reader Widget

  Mostly mimelib -> KMime porting
- Migration

# Porting Piece by Piece

## What still needs to be done

### Todo

- Folder List

  Model and view already in the libraries, but need improvements
- Message List

  Abstract design should make it easy to port
- Composer
  - Again, mostly mimelib -> KMime porting
  - Part of a GSoC project by Constantin
- Searching

  Difficult topic
- Others, like KMCommand classes

# Outline

1 Why do we need Akonadi in KMail?

2 What is Akonadi?

3 How will KMail be ported?

4 When will KMail 2 be released?

# Release Plans

I want KMail 2 *now*!

## Plans

- Actual porting takes place in *akonadi-ports* branch
- A stable KMail will still be provided in trunk
- Release when it is ready
  - Release beta versions of the *akonadi-ports* branch
  - Only a finished version is released, to avoid problems like with KDE 4.0
  - When finished, move back to *trunk*
- Kontact **prototype** by the end of the year
  - KDAB has been contracted for this
  - The prototype is not supposed to be production ready
  - KDE 4.4 will not yet have an Akonadi-based KMail Hopefully in KDE 4.5!

# Summary of this Talk

Just in case you have forgotten what was said

## Summary

- Storage layer of KMail is aging and its design doesn’t fit modern requirements
- Akonadi as a framework solves many old KMail problems
- Porting KMail of Akonadi has begun
- Expect an Akonadi-based prototype of KMail by the end of the year
- Conservative release plan

# The End.

# Questions?

Thank you for attending this talk.
Thanks to Kevin Ottens for the LaTeX template.

Thomas McGuire
mcguire@kde.org