Akonadi - Overview and State

Volker Krause
Overview
Background

- KResource framework does not scale
  - A fully running Kontakt keeps your addressbook in memory 6 (!) times
- Applications and libraries not designed for asynchronous operations
Storage

- Filesystem-like storage of PIM items and collections
- PIM item:
  - Persistent, unique identifier
  - Currently: BLOB, storage in a SQL database
  - Future: Tree of BLOBs, only store structure + metadata in a database
- Storage knows nothing about data formats
- Caches PIM items from remote servers
- Change notifications
Control

• Starts and monitors the storage and resource processes
• Provides D-Bus API for managing resources
• Applications which synchronize data between the storage and an external data source (e.g. groupware server or file)
• One process per desktop instead one plugin per application
libakonadi

- KJob-based API for async retrieval and manipulation of collections and PIM items
- Change monitoring of collections/items
- Self-updating collection and item models
- Independent of PIM item types, type-specific libs (kabc, kcal) build on top of libakonadi
Current State
Current State

- Storage
  - Stores items and collections
  - All basic operations are implemented
- Control
  - Lifetime management for storage and resources
- Resources
  - Basic change recording and replay for offline mode
  - Simple iCalendar file resource
- libakonadi
  - API for basic operations
  - collection and item models
Development Tools

- akonadiconsole:
  - akonadi command line tool
  - KMail