



The long road to new KDE in Debian

Why you need Java and MySQL for a proper desktop

Sune Vuorela

February 8th 2009



- 1 Build system
- 2 File tagging
- 3 Data caching
- 4 Old farts
- 5 Status
- 6 Code
- 7 End



- 1 Build system
- 2 File tagging
- 3 Data caching
- 4 Old farts
- 5 Status
- 6 Code
- 7 End



Build system

KDE - linking the world together



CMake good

- No libtool mess
- No autotools timestamp issues, no .in files to regenerate

CMake sucks

- `for i in /usr/lib/lib*.so ; do LINKLINE+=" -L$i" ; done ; echo $LINKLINE`
- **Recursive** linking

CMake good

- Fix it
- Big patch sets
- Push upstream



1 Build system

2 File tagging

3 Data caching

4 Old farts

5 Status

6 Code

7 End



Tagging files

Learning new things



- Flickr, blogs, debtags
- Sematic desktop is coming - tagging files

Resource Description Framework

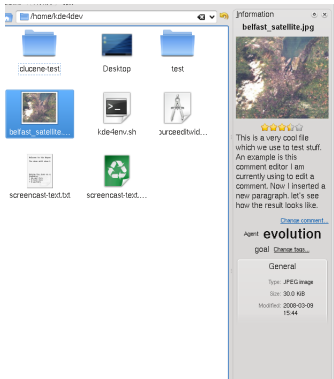
- Meta data model
- Stored in relational databases or in dedicated stores

Storage

- Soprano - storage layer with pluggable backend
 - Redland
 - Sesame2
 - More coming!



Tagging - actions in place





- 1 Build system
- 2 File tagging
- 3 Data caching**
- 4 Old farts
- 5 Status
- 6 Code
- 7 End



Data everywhere

From everywhere to everywhere



Issues

- All apps wants to access all data at the same time
- many apps reading contacts, many apps reading calendar
- many apps contacting imap server

Concurrency

Your data everywhere at the same time



Data everywhere - solution

Databases



Akonadi

- Data cache server framework
- Caches stuff in database for easy access
- Simple dbus api
- With "Agents" to serialize and store data in correct standard format

Database

- QtSQL - abstracts most things away
- MySQL - the current chosen thing
- SQLite - doesn't currently scale
- PostgreSQL - rumors says someones working on it



- 1 Build system
- 2 File tagging
- 3 Data caching
- 4 Old farts**
- 5 Status
- 6 Code
- 7 End



FancyCyness

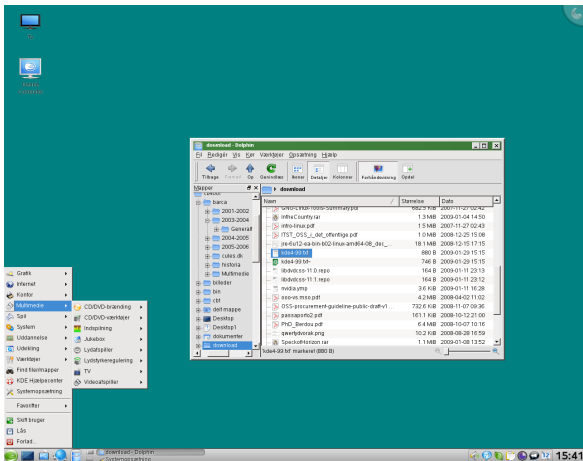
Too fancy?





Less bling

Windows 95





I want KDE3

Really old ...



JUST KEEP USING LENNY



1 Build system

2 File tagging

3 Data caching

4 Old farts

5 Status

6 Code

7 End



Casualties

Some old things must go



Let go of

- Third party plugins to KDE applications
- Window decorations to KDE3
- Some features in unported applications
- Non-c++ apps



Perl-qt might need to go and with that one of the debconf frontends. We need a Qt4 based one.



TODO

- Finalize a exact list of actions to be done before pushing new KDE to unstable
- Finalize writing up a packaging and policy document for third party application maintainers



- 1 Build system
- 2 File tagging
- 3 Data caching
- 4 Old farts
- 5 Status
- 6 Code**
- 7 End



Packaging for CDBS lovers

I sometimes fall in this group



Build-depend on pkg-kde-tools and cdb

Example

```
#!/usr/bin/make -f
include /usr/share/pkg-kde-tools/makefiles/1/cdb/kde.mk
include /usr/share/cdb/1/rules/debhelper.mk
```



Packaging for normal people

I sometimes fall in this group



Build-depend on pkg-kde-tools

Example

```
#!/usr/bin/make -f
include /usr/share/pkg-kde-tools/makefiles/1/variables.mk
builddir/Makefile:
mkdir -p builddir
cd builddir && cmake .. $(DEB_CMAKE_KDE4_FLAGS)
#continue with make -C builddir and so on
```



- 1 Build system
- 2 File tagging
- 3 Data caching
- 4 Old farts
- 5 Status
- 6 Code
- 7 End**



Help?
Yes please



What's needed?

- Go over your own bugs with new KDE and update test if they still apply
- If you have interest in the "niche" applications or cornercase setups, we might need you
- If you maintain a application using KDE libraries, please update to new, updated edition



Questions?

Info

Debian Qt/KDE: <http://pkg-kde.alioth.debian.org>

Me: Sune Vuorela <sune@vuorela.dk>