

# Content FW in M minutes

**Ivan Frade**

**Tracker maintainer in Meego**

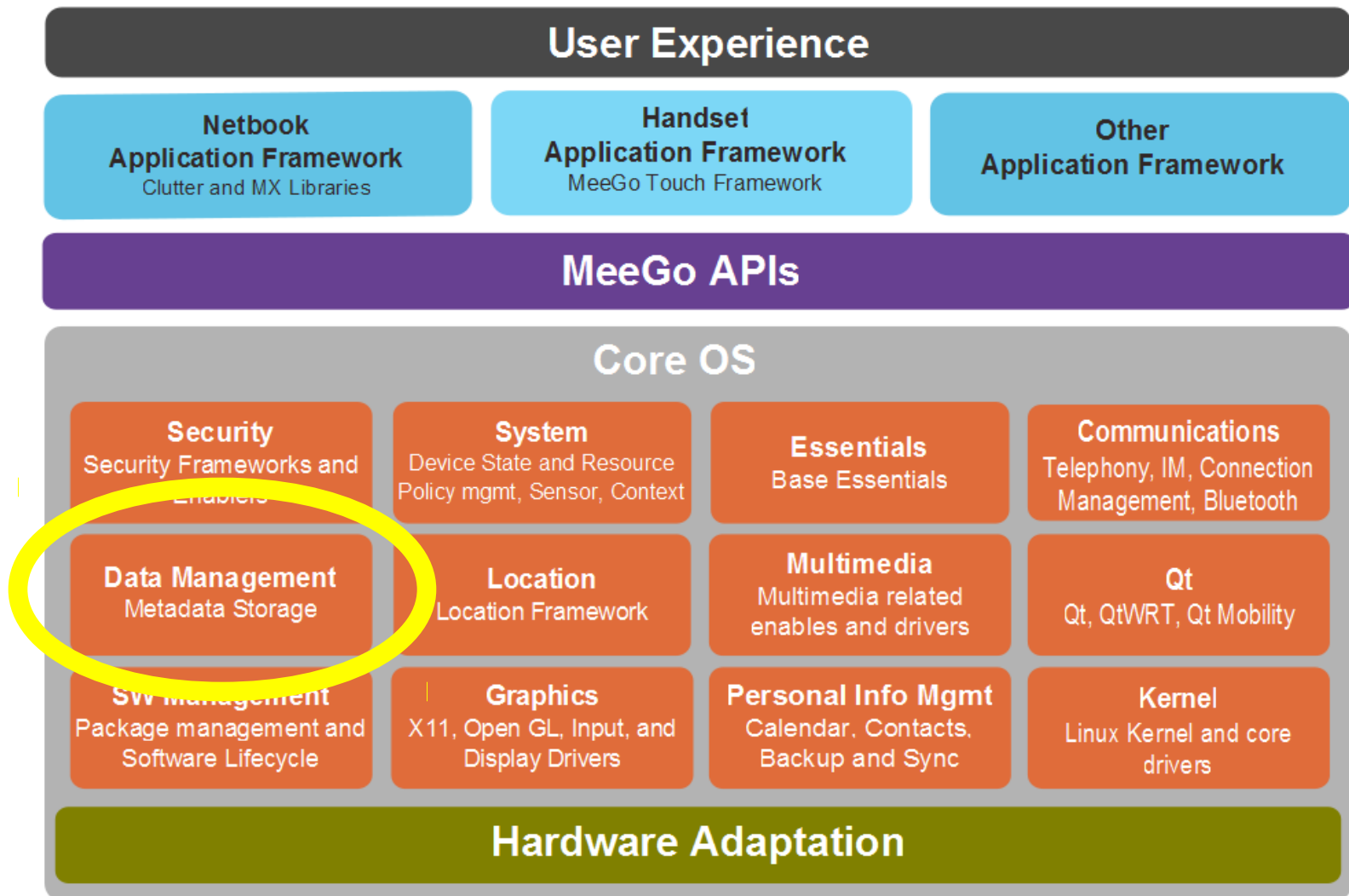
**Nokia**

**[Ivan.frade@nokia.com](mailto:Ivan.frade@nokia.com)**

MeeGo



# Content... what?



# The original problem

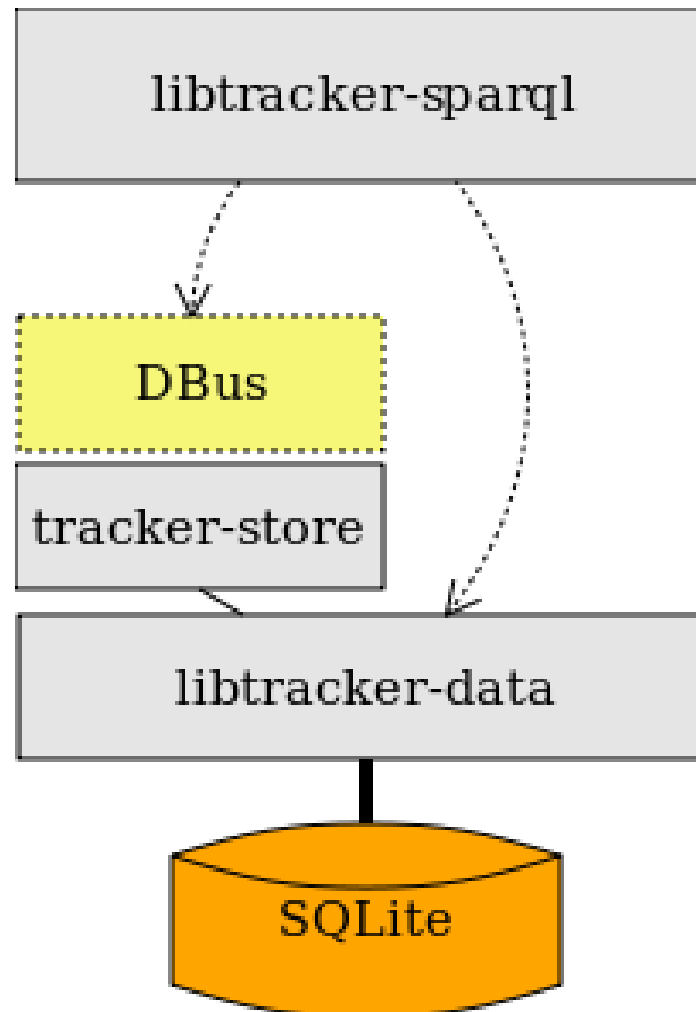
- **Each application has its own DB**
  - Eds, feeds, media, messaging...
- **Lots of problems**
  - Data 'owned' by applications
  - Accessible (or not) by design
  - API (in)consistency
  - Duplicate efforts (e.g. crawling)
  - Mash-ups are impossible

## **Central repository of user information**

- SHARE between apps
- LINK everything

- **One central place**
- **Well-defined shared schema**
- **A graph model**
- **A query language for graphs**

# Let there be Tracker!



- **Miners**

- Specific processes monitoring sources
- E.G. filesystem (in meego), flickr, feeds...
- Extractors for filesystem

- **Services**

- Telepathy

- **Applications**

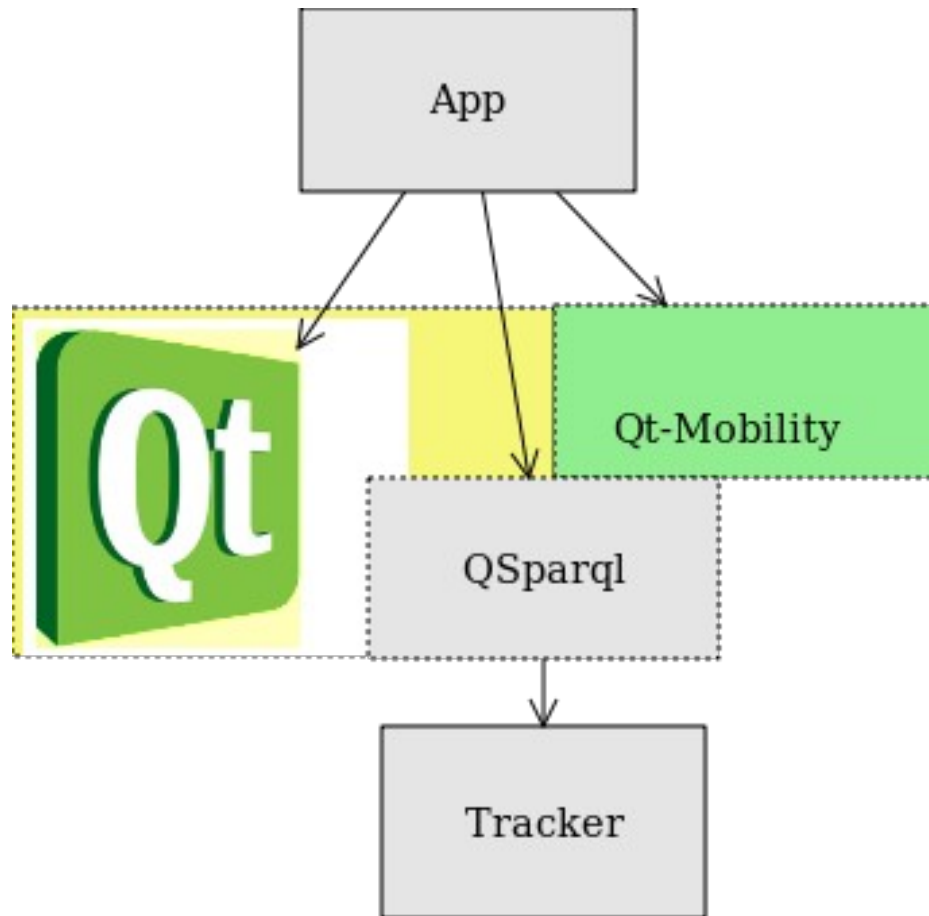
- Media player (playcount), Feeds (readed),...

# But... what is actually there?

- **Contacts & IM**
- **Images**
- **Videos**
- **Music**
  - Playlists
- **Feeds**
- **Support for geolocation**
- **Tagging**



# New friends: Qt & QML



- **QSparql**
  - Async API
  - Abstract backends
  - QML friendly
  - Simple
- **Eventually QT**
  - Backend for high level APIS
  - Pro access

**Great! But not enough...**

## **A consistent / complete Data Management API**



QSPARQL



libContentAction



libThumbnailer

- **Libthumbnailer**
  - Qt-ish API
  - Thin wrapper over DBus
- **Tumbler**
  - Daemon in DBus
  - Upstream project
  - Pluggable in/out process
  - Priority queues

- **What can i do with an item?**
  - It depends...
  - Mimetype ^2
  - Extensible in a similar fashion

# So, Content FW for app developers is...

- **QSparql: to get content**
- **Libthumbnailer: to get thumbs of the content**
- **Libcontentaction: to guess what to do with it**

- **Extension points in the Framework:**
  - Miners to bring new content
  - Extractors to understand new file types
  - Plugins to thumbnail new types

- **All Tracker work in GNOME git**
  - Upstream release → meego release
- **Tumbler is upstream project**
- **Everything else in gitorious**
- **Active in IRC, bugzilla, mailing list**
  - #tracker in GIMP net as entry point

# Content FW is the work of...

- **Tracker core developers:** Jürg Billeter, Ivan Frade, Aleksander Morgado, Mikael Ottela, Martyn Russell, Philip Van Hoof
- **Content FW APIs devs:** Richard Dale, Mark Doffman, Marja Hassinen, Iridian Kiskinen, Sami Rosendal
- **Community contributors:** Michael Biebl, Adrian Bustany, Tshepang Lekhonkhobe, Rob Taylor, ...
- **MeeGo integrators:** Jean-Luc Lamadon, Maitrey Mishra



**Ivan Frade**

**Software engineer/architect  
Tracker maintainer in Meego  
Ivan.frade@nokia.com**