

# PIM - It's all about the contacts

Patrick Ohly, Intel GmbH  
October 25<sup>th</sup>, 2013



# Content

- Use cases
- Open source PIM solutions:
  - Akonadi
  - QtContacts/Tracker
  - Tizen contacts service
- Tizen IVI:
  - Evolution Data Server
  - SyncEvolution
  - libphonenumber
  - folks
- First steps and links



CC BY 2.0, Jim D. Woodward



# Head unit as a better UI for multiple phones

- Fast caller ID lookup for incoming call (not the same as text phone number search!), local photo data.
- Look up a contact via searching or browsing and initiate a phone call.
- Find address and start navigation.
- Manage data from more than one phone: driver and passenger.
- Unified address book: no duplicates, merge data from all available sources. May include dynamic information (online presence).



# Existing open source PIM solutions

- **Akonadi**
  - Unified API for read/write access to item collections, with client/server architecture.
  - Storage not included, external storage hidden behind Akonadi API.
  - No search API, searching via indexing with Nepomuk.
  - Fairly complex, hard to tune for specific searching.
- **QtContacts/Tracker**
  - Provides read/write/search API.
  - Write via server, read in client.
  - Several translation layers: QtContacts <-> RDF/SPARQL <-> SQL
  - Same drawback as for Akonadi.
- **Tizen contacts service**
  - Client/server architecture specifically designed for contacts in Tizen Mobile.
  - All address books in one sqlite file.
  - No read access to database file in client.
  - Has a concept of merging contacts, but not very configurable.
  - Developed by Samsung, not enough outside expertise.



# The GNOME PIM Stack for Tizen IVI: Overview

- **Evolution Data Server:** store contacts in sqlite.
- **libphonenumber:** parsing and normalization of phone numbers.
- **folks:** unified address book in memory.
- **SyncEvolution:**
  - Phone and cloud syncing.
  - Hosts the unified address book.
  - “IVI PIM Manager” D-Bus API.
- **Bluez obexd:** Phone Book Access Protocol



# Overview, cnt.

- LGPL 2.1 or more liberal.
- Minimal additional dependencies (no GTK):  
ICU, glib, sqlite, vala, libgee, libsecret, libgcr, protobuf, gtest;  
optionally also libsoup + neon
- Locale aware:
  - Phone number parsing.
  - Sorting, special case Pinyin (transliterate, then mix with Western names).
  - Support systemd locale for dynamic change of locale.





CC BY 2.0, Elsie esq.

## IVI Features in the GNOME-based Stack





# Evolution Data Server: “Per-device access”

## One address book per phone

### Traditional EDS:

- Abstract API, framework, storage provided by specific backends.
- File backend: Berkley DB + sqlite index.

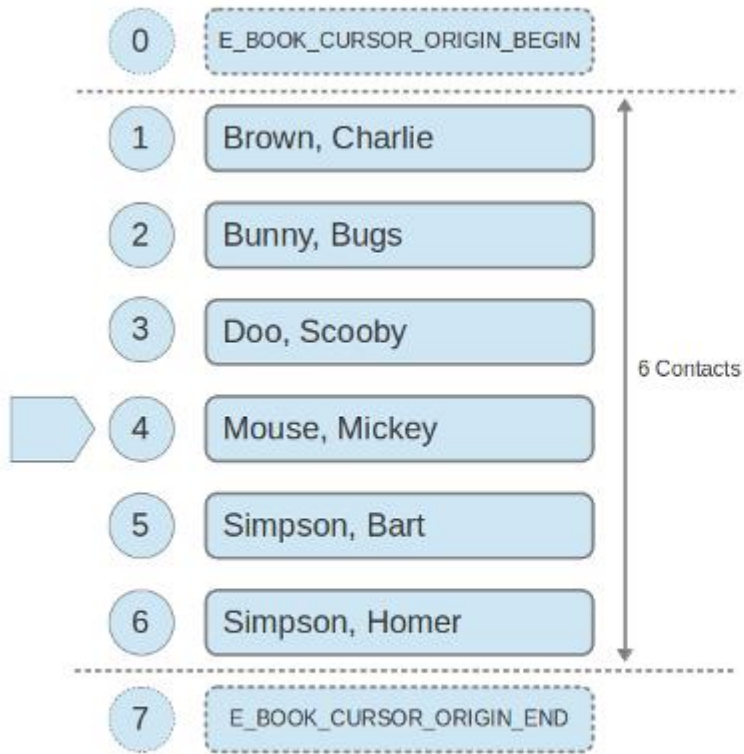
### Enhanced EDS (3.6 and later plus Openismus patches):

- All data in sqlite, configurable indices to reflect searches made by UI.
- Normalize phone numbers with libphonenumber:  
find “089-1234” when looking for “+49891234”.
- Regular expression support for fuzzy phone number search and keypad search.
- Writing in EDS daemon, reading in clients:  
same API, just different open method.
- Efficient browsing through all or some contacts:  
sorted results, cursor marks current position, reading only returns next  $n$  contacts.
- Locale-aware alphabetic index.

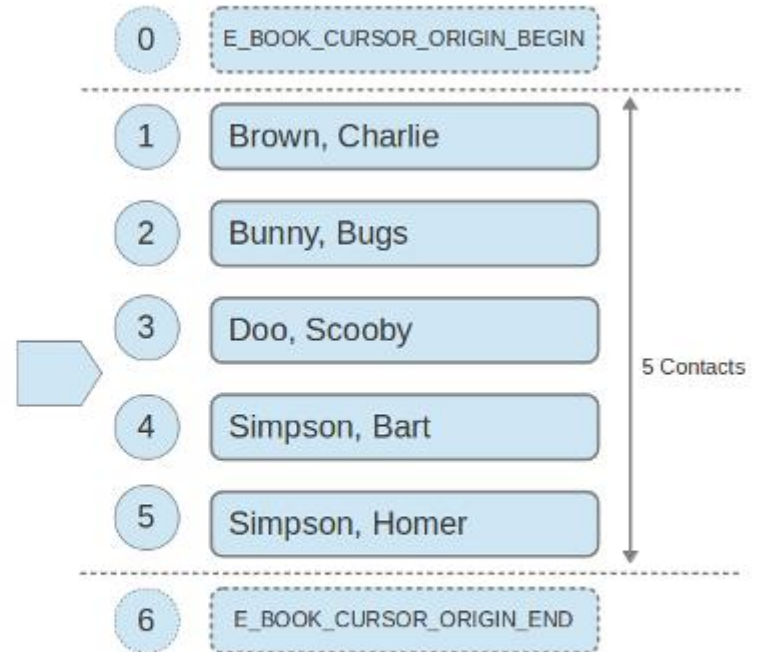




# Evolution Data Server: Cursor



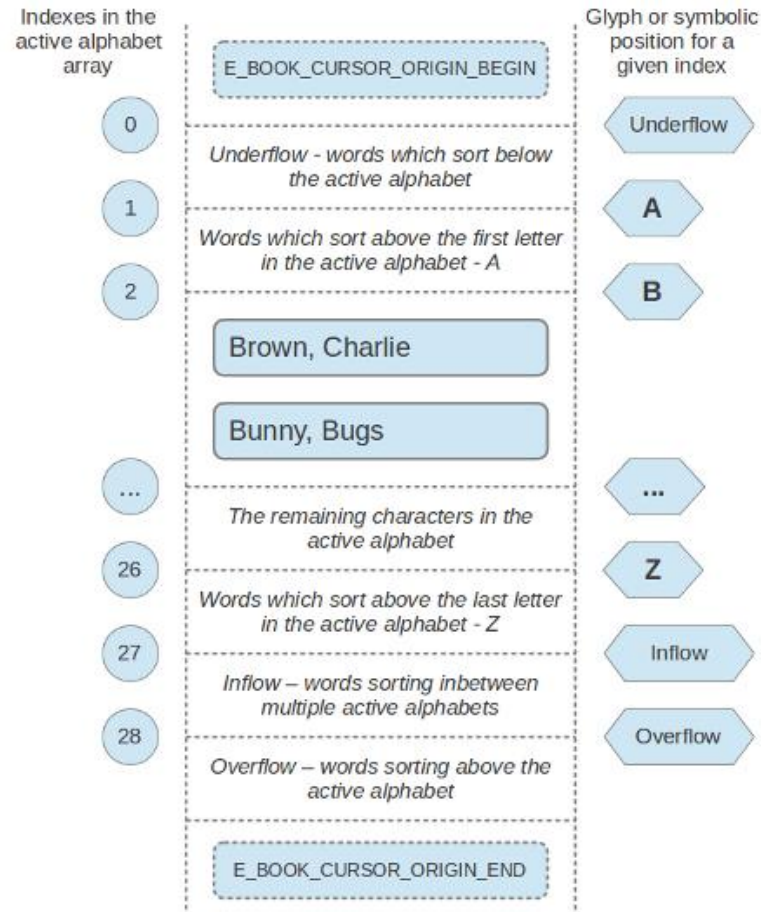
*Cursor points to Mickey Mouse, the cursor position is now 4 out of a total of 6 contacts.*



*After deleting Mickey Mouse, the cursor still points to Mickey Mouse but now the calculated position and total have changed.*



# Evolution Data Server: Alphabetic Index



# SyncEvolution + folks: “Unified address book”

- Configurable set of enabled address books.
- No disk writes when reconfiguring.
- Might include transient information (presence status).
  - Kept in memory.
- Sorting, searching, fast caller ID lookup.
  - Model/view/controller principle for results.

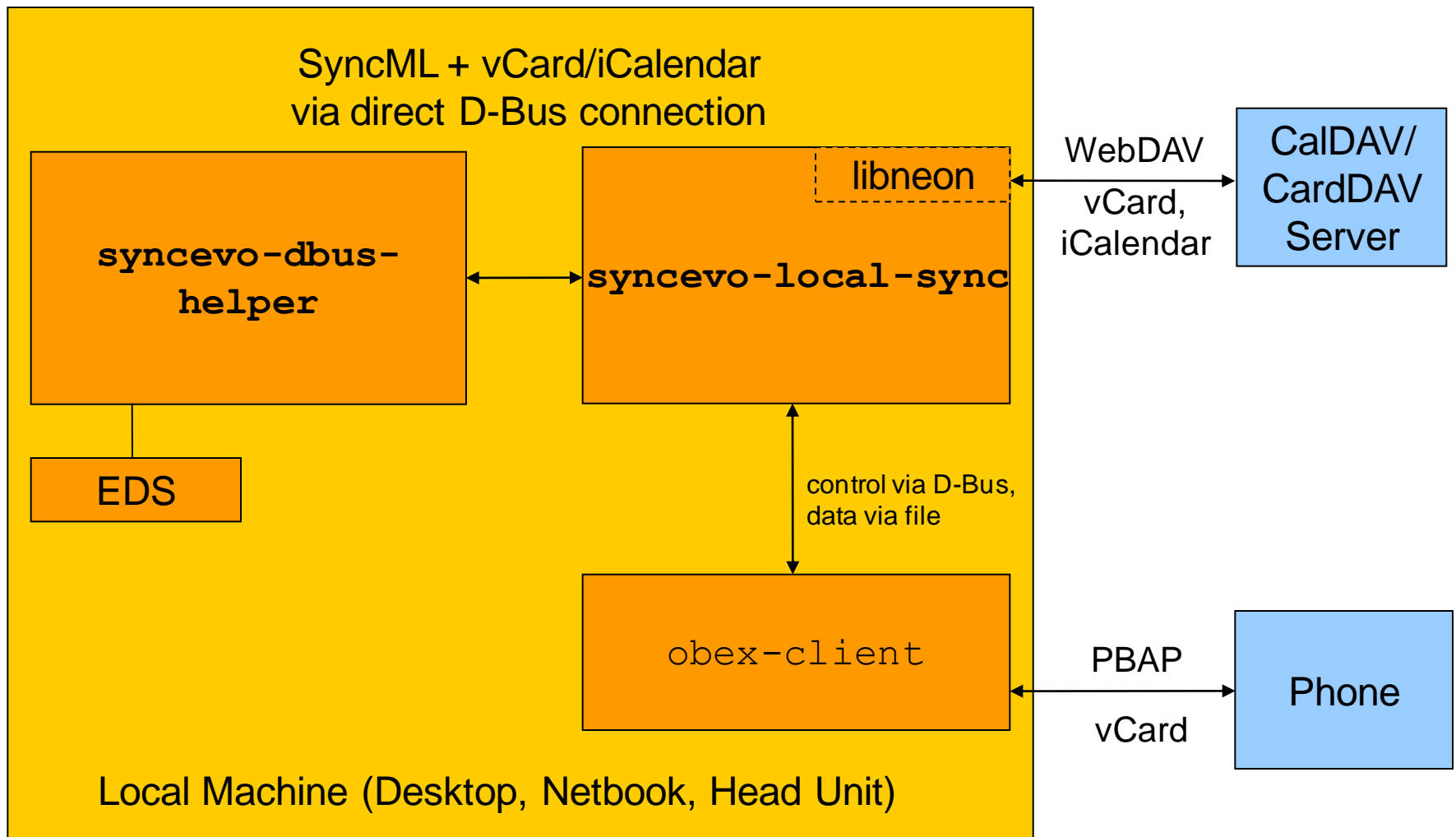


# SyncEvolution + obexd: PBAP syncing

- Take full dump of phone address book, detect changes and apply them to local cache.
- No writes (logs, DB, sync meta data) if nothing changed on phone.
- Incremental syncing:
  - First (or only) text: must not modify local photos.
  - Then everything: must add/update/remove local photos.
- Overlap download and processing.
- Goal is to finish sync shortly after finishing PBAP download (“processing at wire(less) speed”).



# PBAP + CalDAV/CardDAV sync





**Getting started**

# Deployment

- Compile EDS with libphonenumber support.
- Compile SyncEvolution with PIM Manager, PBAP and ebook enabled.
- XDG\_CONFIG\_HOME: location of config files.
- XDG\_DATA\_HOME: EDS databases.
- SYNCEVOLUTION\_PBAP\_SYNC=incremental/text/all
- LC\_\*/LANG: current language, also used to determine home country.
- Logging via PIM Manager peer config:
  - “logdir”: location of log files (default: ~/.cache/syncevolution).
  - “maxsessions”: number of session directories/logs (default: 10).
  - Soon: GENIVI Diagnostic Log and Trace (DLT) support.





# Examples and Tests

- `src/dbus/server/pim/testpim.py`
- `src/dbus/server/pim/examples`
- Use “file” peer instead of real phone.
- Test data from Openismus EDS benchmark.



# First Steps

- **PBAP:**  
`syncevolution --export -- \`  
`backend=pbap database=bt-obex://<bdaddr>`
- **EDS:**  
`syncevolution [--print-items|--export -|--import <file>] \`  
`backend=evolution-contacts [database=<DB name>]`
- **Syncing after configuring a peer with the PIM Manager**  
(not quite the same way as via PIM Manager – writes sync meta data),  
debug log files in `~/.cache/syncevolution` and output to `stderr`:

```
SYNCEVOLUTION_DEBUG=1 \  
syncevolution --daemon=no \  
              --run \  
              loglevel=4 \  
              <peer config name>
```





CC BY 2.0, Dell



## Getting involved

# Next Steps

- In development:
  - PIM Manager
  - PBAP backend
  - Google CalDAV/CardDAV
  - Tizen Device Contacts Web API based on EDS
- Ideas for IVI:
  - Calendar support
  - Transparent access to contacts without caching
- Needs community help:
  - KDE
  - GTK UI



## SyncEvolution:

<https://syncevolution.org>

<http://cgit.freedesktop.org/SyncEvolution/syncevolution/tree/src/dbus/server/pim/pim-manager-api.txt>

<http://cgit.freedesktop.org/SyncEvolution/syncevolution/tree/src/dbus/server/pim/README>

<http://cgit.freedesktop.org/SyncEvolution/syncevolution/tree/src/backends/pbap/README>

## Evolution Data Server:

<https://developer.gnome.org/libebook/stable/EBookClient.html>

<https://people.gnome.org/~tvb/libebook/EBookClientCursor.html>

## Other projects:

<https://wiki.gnome.org/Folks/>

<http://code.google.com/p/libphonenumber/>

<http://code.google.com/p/googletest/>

<http://code.google.com/p/protobuf/>





**Backup**

