



GPE Phone Edition

Nils Faerber <nils.faerber@kernelconcepts.de>

About G(PE)² ?

- Graphical User Interface (GUI) for Linux based mobile phones
 - Mobile phone specific applications
 - Service abstraction for application developers (Voice-Call, SMS/MMS/IMS, PIM, ...)
 - Split into Service-Enabler & -Provider
-
-

GPE and Relatives



- Based on GPE
<http://gpe.linuxtogo.org>
- Based on LiPS Standards
<http://www.lipsforum.org>
- Uses Gnome Mobile And Embedded (GMAE) framework
<http://www.gnome.org/mobile/>



There is more to it than "mobile phone"

- Mobile phone software is more than just a "dialer"
 - Reliability must be ultimate goal
Example: Red hangup-button **must** always end a call
 - Application have to properly interact and cooperate
Example: Incoming call during SMS input



There is more to it than "mobile phone"

- Applications have to cooperate
Example: Send SMS to contact from addressbook
 - Needs built-in security
Example: A misbehaving application must not be allowed to send hundreds of SMS
 - Problem: Liability
-
-

There is more to it than "mobile phone"

- Connection-Management
Example: VoIP using which channel?
GPRS/UMTS, WLAN, Bluetooth?
Registration & Authentication
 - For best acceptance also take into account provider/operator requirements
Example: Device Management,
Changing of device settings through operator (mobile) network
-
-

Implications for $G(PE)^2$

- Requirements are strong/high, if ever a GPE Phone Edition phone is expected to become available of the shelf
- *Almost* impossible for an OpenSource & spare time project to handle

LIPS & G(PE)^2



- Linux Phone Standards Forum
- Industry consortium with more than 30 members, e.g. Orange/FT, BT, Telecom Italia, Texas Instruments, NXP, Access/PalmSource, Mizi, PurpleLabs, VirtualLogix, Cellon, ZTE, etc.

LiPS & G(PE)^2

- LiPS goal: Developments of common specifications, especially APIs, for Linux based mobile phones
 - By definition: "Open Source friendly"
All resulting standards will be and are publicly available and free to use
-
-

LiPS & G(PE)^2

- Main Initiator: France Telecom / Orange, R&D Lab Beijing, China, Danny Huang
 - Orange/FT looked actively for collaboration with OpenSource community to create LiPS testbed implementation
 - GPE developers work together with Orange/FT team since 2005 on G(PE)^2!
-
-

LiPS & G(PE)²

- G(PE)² is the only non-commercial LiPS member
- This way also represents the OpenSource community in this industry driven context
- Community feedback and collaboration is welcome and goal of G(PE)²

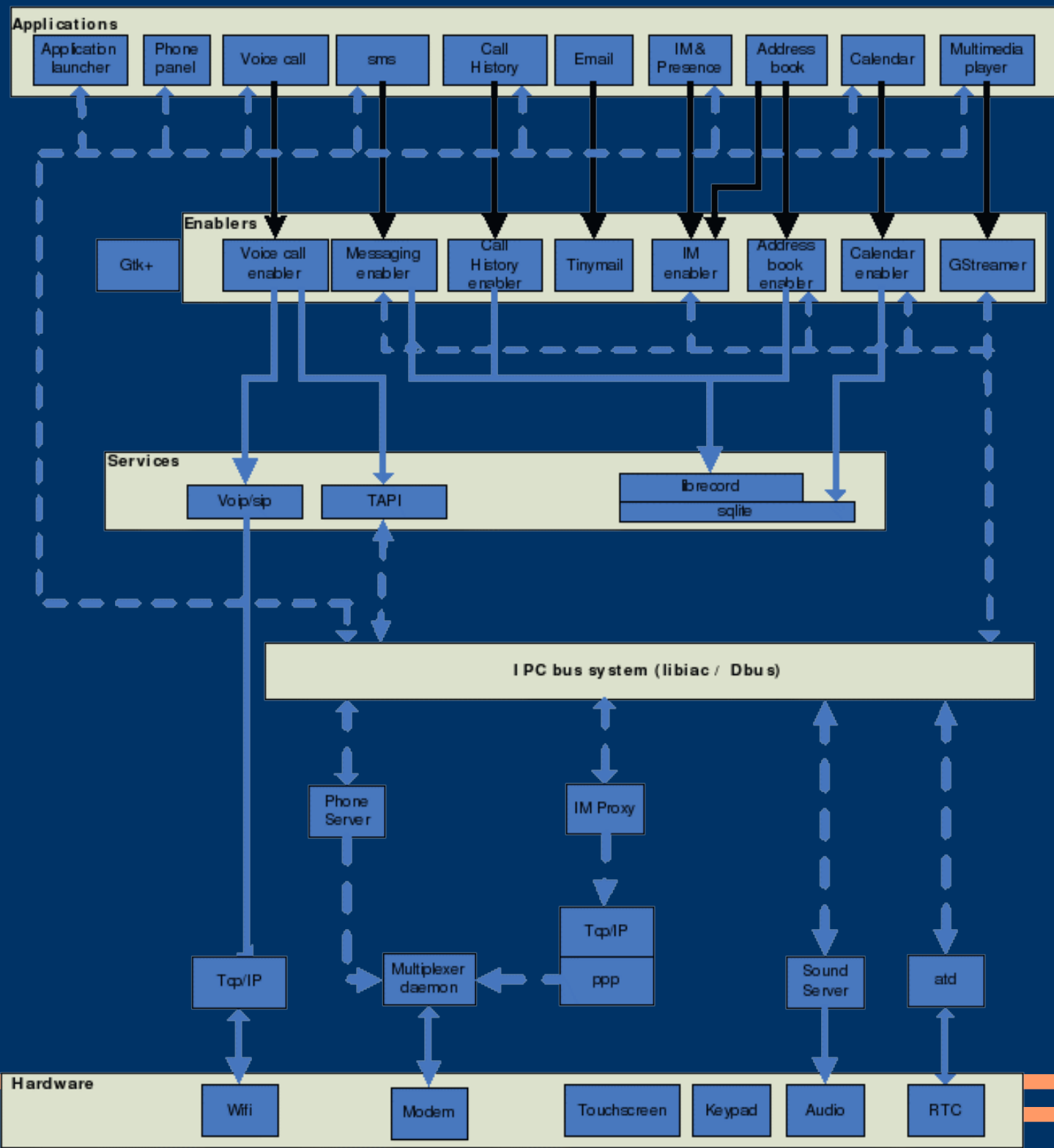
LiPS & G(PE)^2

- LiPS User Interface Workinggroup has specified GTK+ as GUI toolkit
- Several other technologies used by GPE and GMAE are in process to be specified as standard by/for LiPS

G(PE)^2 – More Technical

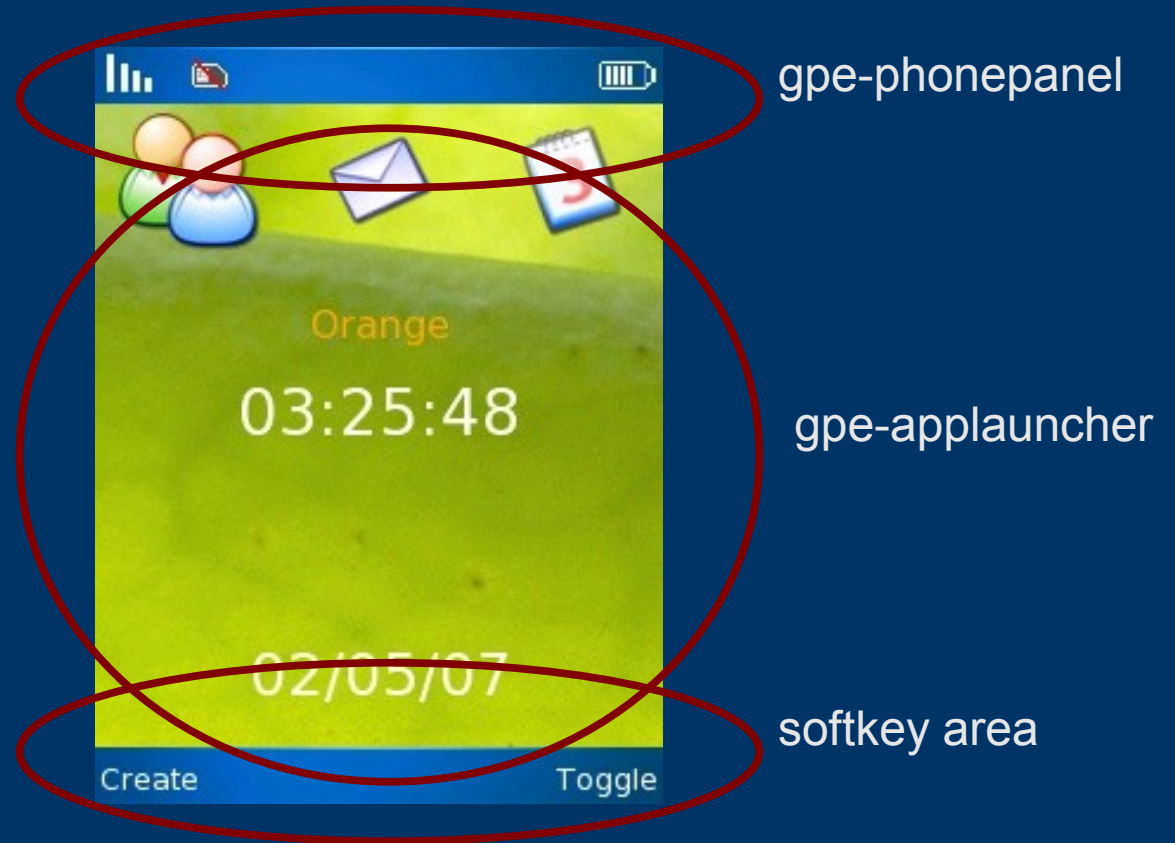
- G(PE)^2 uses, like GPE, X11, GTK+, Matchbox, DBus, etc.
 - GPE is a typical "Hacker Project" - consists of modules without a big "master-plan"
 - This approach is not sufficient anymore for a competitive mobile phone stack, a clear architecture is needed:
-
-

G(PE)² Architecture



G(PE)^2

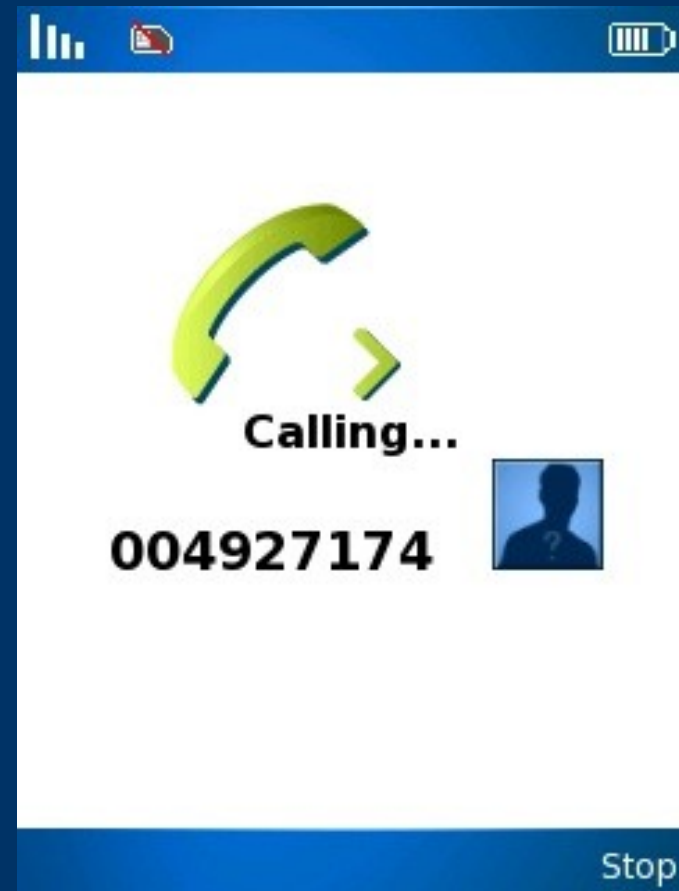
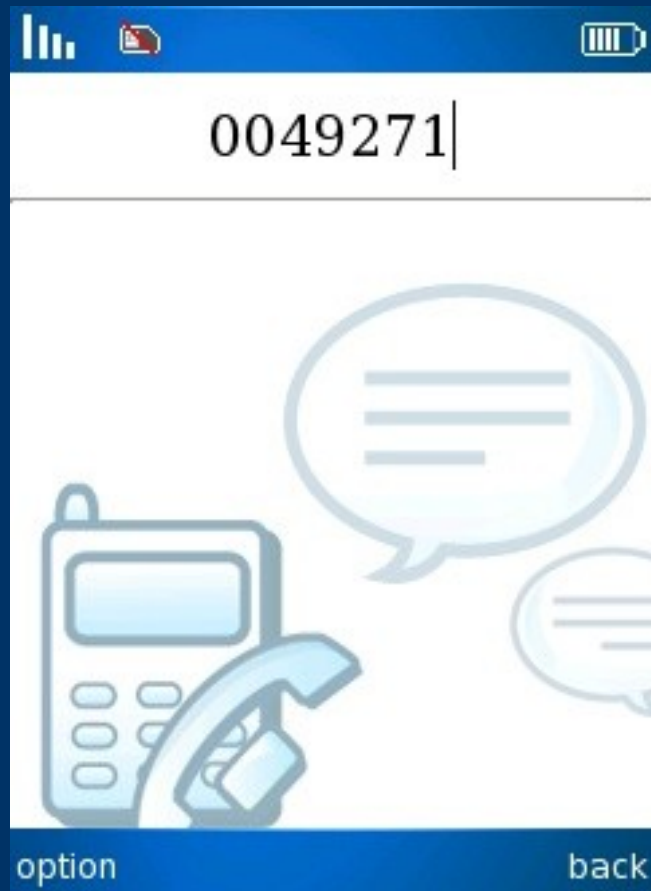
- Indicator Panel with plugins
- Application Launcher with Application-Quickstart
- Configurable application-independent softkeys



Look & Feel



Look & Feel



Look & Feel



Demo, VMware

- Download
<http://gpephone.linuxtogo.org>
 - Demo is very close to a real mobile phone system
 - It is an independent virtual machine with similarly limited resources
(RAM, HDD)
-
-

Demo, VMware?

- Exactly the same tools, rootfs and libraries can be used like they will be on a real device – it is host independent
- VMware Player available free of charge for many platforms - not only Linux

Currenty State

- Approximately 50MB sourcecode (incl. pixmaps)
 - Applications:
 - Contacts (also from SIM)
 - Calendar
 - SMS
 - eMail
 - Audio-/ Video-Player
 - Voicecall, Call-History
-
-

Current State

- GUI parts:
 - GPE-Applauncher (also Homescreen)
 - GPE-Phonepanel
 - libgpephone
 - libgpewidget

- Example GUI-Applikation
 - GPE-Example



Current State

- "Enablers"
 - libabenabler – Addressbook Enabler
 - libcalenabler – Calendaring Enabler
 - libchenabler – Call History Enabler
 - libiac – Inter Application Communication
 - libim – Instant Messaging
 - librecord – Storage Backend helper
 - libvocenabler – Voice Call enabler
-
-

Current State

- Servers
 - phoneserver – GSM Modem Handling
 - soundserver – Sound output
 - Videoserver – Video output (video player)
-
-

Current State

- Working on all devices:
 - Addressbook
 - Calendar
 - Application Launcher/Homescreen
 - Softkeys
 - Audio-/Videoplayer
 - Simple eMail

Current State

- Working on some devices,
depends on hardware support:
 - Voice Call
 - SIM Phonebook
 - SMS, read/write, send

Those functions mostly depend on support for the GSM modem.

Missing

- Current state is a good basis and proof of concept
 - Needs further stabilisation
 - More support for other hardwares
 - **We need more open mobile phones!**
-
-

G(PE)^2

Relation To Other Projects

- GMAE - Mobile Awareness
A lot of mobile projects today use Gnome parts (GTK+, Cairo, Pango, Dbus, etc.)
<http://www.gnome.org/mobile>
 - OpenMoko / NEO1973
Alternative mobile phone software stack and "open" hardware
<http://www.openmoko.org>
-
-

Many Thanks!

- Questions?

