GNUstep on a PDA

The QuantumSTEP project

Dr. H. Nikolaus Schaller
Oberhaching, Germany
hns@computer.org

Why?

- Most PDAs and more and more phones are sold today with Windows CE
- The Mac-User needs optimal sync of data and applications
- *x*-step (GNUstep, OpenStep, NeXTSTEP) is one of the most powerful open software development frameworks for desktop systems

GNUSTEP

➡ why not transfer *x*-step to a PDA?

How?

- Products like OS X Lite, Embedded Cocoa are not available from Apple
- GNUstep is an open source package and largely compatible to OpenStep / Cocoa
- Make it compile for and run on an ARM CPU
- Adapt and extend to specific needs of a handheld

➡ mySTEP (now QuantumSTEP) was started in 2003, based on mGSTEP

Design issues to solve

1. Get a PDA with Linux and X-Server
2. Define Development System and Cross Compiler
3. Define File System Layout on the PDA
4. Memory constraints, missing FPU
5. NSInvocation
6. Missing Frameworks, Classes, Methods
7. UI-Principles
8. Designing Application's UI
9. Designing PDA-Applications (PIM + Communication)

PDA with Linux and X-Server

- The only largely available commercial model was until recently the Sharp Zaurus
  - SL5500G (cheap on eBay; a little old; slow)
  - latest models: SL-C1000 and C3100 (4GB HDD integrated)
  - Linux 2.4.x Kernel with Qtopia (Framebuffer based)
  - X/Qt-Server using Qtopia Framebuffer

- Alternatives
  - alternate ROMs for Zaurus: pdaXrom, Open Embedded, OpenBSD 3.8
  - Open Embedded/Familiar Linux for iPAQ, Dell, etc.
  - Nokia 770 (Maemo)
  - Open Embedded/Familiar Linux/GPE for iPAQ, Dell, Acer, etc.

- Ideas for the Future:
  - Port Darwin (Apple) to the ARM processor
  - Mobile Phones with Linux

Development System and Cross Compiler

Optimized solution:

- Develop on a Macintosh with MacOS X 10.4 and Xcode 2
  - ➡ direct access to the online manual
  - ➡ allows for "fat binary" applications that just need to be copied back and forth
- use X-Server on Mac
- use ssh to connect to the PDA
- use gcc with Objective-C extension (Zaurus still uses 2.95.3)
- recompiled libobjc as shared library
- added several missing libraries for ARM-Linux (jpeg, png, ...)

File System Layout on the PDA

- Decided to mimick MacOS X as good as possible and where reasonable
- introduced `.app` bundles with `.app/Contents/Linux-ARM` for the binary
- allows to build "Fat binaries" for PowerPC and ARM binaries but sharing Resources
- added `Launch Services` to `NSApplication` to locate applications by name or bundle identifier
- installation location (`/home/myPDA`) hidden in `NSObject's -fileSystemRepresentation`

Example

```
bash-2.05# ls -lR /home/myPDA/Applications/myMail.app

myMail.app:
drwxr-xr-x    4 zaurus   bin             0 Oct 10 09:40 Contents

myMail.app/Contents:
-rw-r--r--    1 zaurus   bin          1800 Oct  2 17:39 Info.plist
drwxr-xr-x    2 zaurus   bin             0 Oct 10 09:40 Linux-ARM
drwxr-xr-x    4 zaurus   bin             0 Oct 10 09:40 Resources

myMail.app/Contents/Linux-ARM:
-rwxr-xr-x    1 zaurus   bin         65750 Oct  2 17:39 myMail

myMail.app/Contents/Resources:
-rwxr-xr-x    1 zaurus   bin           261 Oct  2 17:39 E_Mail.png
drwxr-xr-x    2 zaurus   bin             0 Oct 10 09:40 English.lproj
drwxr-xr-x    3 zaurus   bin             0 Oct 10 09:40 Mail.prefPane
-rwxr-xr-x    1 zaurus   bin           324 Oct  2 17:39 Mail_send.png
-rw-r--r--    1 zaurus   bin           452 Oct  2 17:39 Mailsend_16.png
-rw-r--r--    1 zaurus   bin           654 Oct  2 17:39 add_user_16.png
-rwxr-xr-x    1 zaurus   bin           674 Oct  2 17:39 addbk_16.png

myMail.app/Contents/Resources/English.lproj:
-rw-r--r--    1 zaurus   bin         33546 Oct  2 17:39 MainMenu.mib

myMail.app/Contents/Resources/Mail.prefPane:
drwxr-xr-x    4 zaurus   bin             0 Oct 10 09:40 Contents

myMail.app/Contents/Resources/Mail.prefPane/Contents:
-rw-r--r--    1 zaurus   bin          1036 Oct  2 17:39 Info.plist
-rw-r--r--    1 zaurus   bin           195 Oct  2 17:39 sub_16.png
-rw-r--r--    1 zaurus   bin           476 Oct  2 17:39 InfoPlist.strings
-rw-r--r--    1 zaurus   bin         30117 Oct  2 17:39 MainMenu.mib
```

Memory constraints, missing FPU

- Work was based on mGSTEP - a fork of GNUstep
  - forked approx. 1999 from GNUstep
  - since then continuously developed by Felipe A. Rodriguez with embedded systems (memory footprint and speed) in mind
  - was in 2003 more complete and smaller footprint than mainstream
  - some parts from more recent GNUstep CVS have been integrated
- Additional classes and methods required for a Cocoa compatible PDA environment have been added
  - strict use of shared libraries (even libobjc)
  - implemented as much as possible in C/Obj-C instead of using external libraries (e.g. XML, Unicode, Timezones)
- missing FPU is not really an issue as the demos will show

NSInvocation

- required since we need Distributed Objects
- needs to pass arbitrary arguments to an indirect function call
- very processor architecture specific (stack frame & calling conventions)
- `avcall`, `libffcall` or `libffi` did not work for ARM
- hacked the code to directly use the gcc extensions `__builtin_*()`
- works mostly (has still a bug with returning `BOOL`)

Designing Application's UI

To GORM or not to GORM?

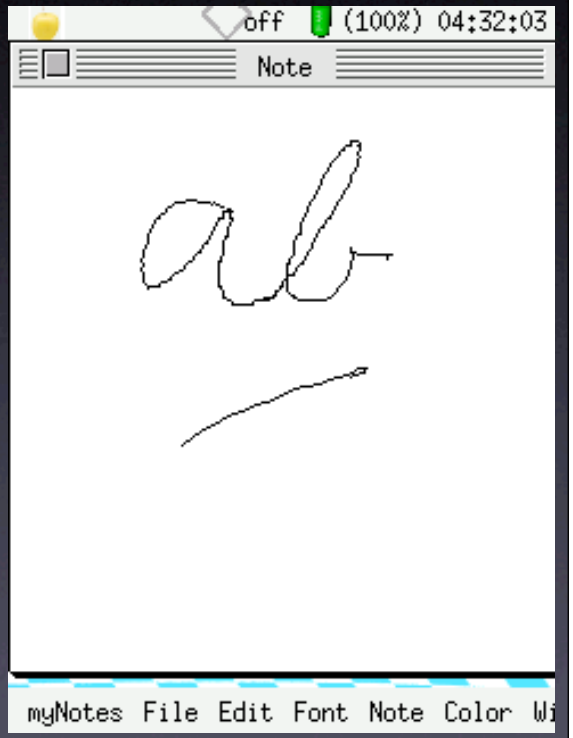
- Decided to use Interface Builder (best integration into Xcode) and translate NIBs (nib2gmodel approach)
- adapted to the `.mib` format used by mGstep
- added a proprietary compressed binary property list format to speed up loading (to be replaced by `NSKeyedArchiver` asap)
- result was a tool `nib2mib`, capable of translating most NIB files

User Interface Principles

- We can benefit from Apple's Single-Button-Mouse architecture: we have only one pen
- We can use handwriting and gestures for operation
- mapping gestures to `keyEquivalents` is straightforward
  - e.g. horizontal stroke => `Meta-N` => New File
- Some issues remain:
  - Do we need to implement Drag&Drop? If yes, how?
  - How to initiate Context-Menus
  - How to do Copy&Paste without a keyboard for Meta-C & Meta-V
  - Screen sizing and rotation
    - Automatically enlarge windows to full screen
    - Zaurus hinge rotation is unsolved

Handwriting

- Handwriting is not that difficult to integrate
  - Option 1: Window Manager / X application
  - Option 2: use `eventStartsInk:` and track mouse movements into an `NSBezierPath`
- Main challenge is a good pattern recognition algorithm
- A smaller issue is the lack of transparent windows - handwriting has to obscure the screen with a white sheet



Missing Frameworks

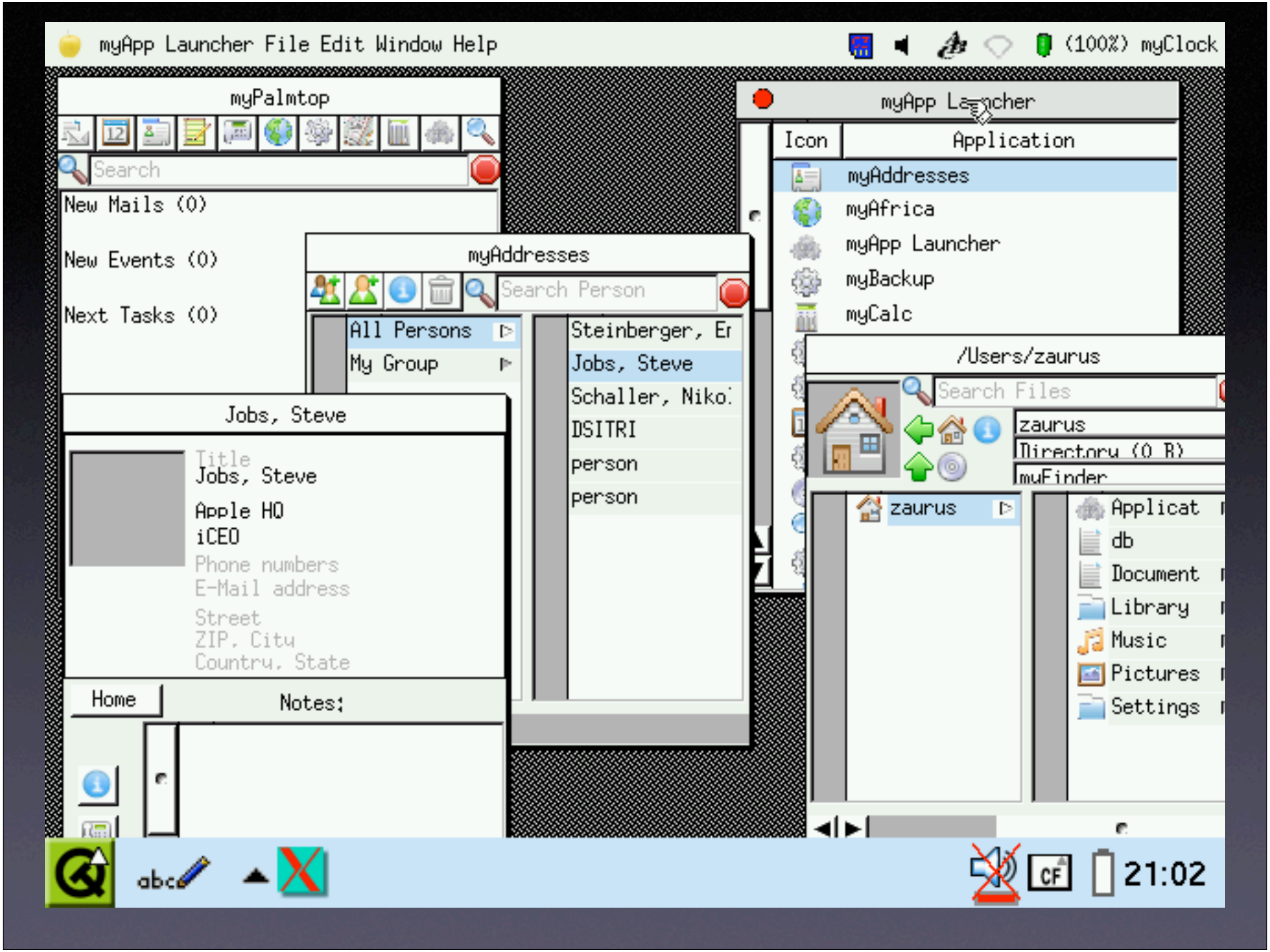
- GNUstep is not completely Cocoa compatible
  - didn't have NSPrefPane
  - NSMenuView, NSMenuItemCell with Apple-like menu bar
  - SystemUI (Menu Extras)
  - Calendar & Mail framework
  - PDFKit (Quartz compatible API)
  - SyncServices
  - ...
- QuantumSTEP adds these
  - written from scratch to keep control over memory footprint
  - Partially made LGPL open source

Designing PDA Applications

- QuantumSTEP has its own suite (partially running)
  - File manager
  - Address book (myAddresses)
  - Calendar (myDates - iCal)
  - Notes (myNotes - Stickies)
  - Mail (myMail)
  - System Settings
  - etc.
- What we would like to have
  - Chat
  - Web Browser (based on WebKit...)
- Bonus
  - JVM
  - GPS Navigator

Project Status

- Demo 1: Running QuantumSTEP on a Sharp Zaurus C860
- Demo 2: Creating and Cross-Compiling a simple project


myApp Launcher File Edit Window Help
(100%) myClock
myPalmtop
Search
New Mails (0)
New Events (0)
Next Tasks (0)
myApp Launcher
Icon
Application
myAddresses
myAfrica
myApp Launcher
myBackup
myCalc
myAddresses
Search Person
All Persons
My Group
Steinberger, Er
Jobs, Steve
Schaller, Niko
DSITRI
person
person
Jobs, Steve
Title
Jobs, Steve
Apple HQ
iCEO
Phone numbers
E-Mail address
Street
ZIP, City
Country, State
Home
Notes:
/Users/zaurus
Search Files
zaurus
Directory (0 B)
myFinder
zaurus
Applicat
db
Document
Library
Music
Pictures
Settings
abc
CF
21:02

Future

- Make QuantumSTEP more compatible with latest MacOS X (10.4)
  - add NSStream, NSPredicate, NSIndexSet, KVC, KVO, etc.
  - Review all header files
- Port to more Handheld Devices
- Contribute modifications back to mainstream GNUstep project
- Volunteers and participation is very welcome!

More Info

Many thanks to core team GNUstep developers!

http://www.quantum-step.com