GNUstep on embedded devices

Dr. H. Nikolaus Schaller
Golden Delicious Computers GmbH&Co. KG
Oberhaching, Germany
hns@goldelico.com
We have a dream.
OSX compatible devices in our pockets.
Objective-C +
a powerful platform.
Easy application development for real world.
Open Source.
We continue to dream.
No!
Open Source makes dreams come true.
DIY: GNUstep/mySTEP.
Enough Marketing.
Let’s become technical...
Why OSX in your pocket?

From an OS X Application developer’s perspective OpenSTEP/Cocoa is a really great toolkit:

- write fancy applications in 10 minutes (ease of application development using Objective-C, Xcode, Interface Builder, MVC, Cocoa Frameworks, CoreData)
- fully supports MVC, graphical interface design, unicode, localization, networking, entity-relationship storage, ...
- flexible system architecture due to plugins and bundles
- It’s free for any Macintosh owner but not open

===> How can we get that for our pocket sized devices?
Objective-C in a Nutshell

- Originally inspired by Smalltalk
- Very pragmatic Object Extension of C
- Developed by Brad Cox and Tom Love in the early 80ies at StepStone and from 1988 on by NeXT, now Apple (1996)
- Supported in GNU Compiler Collection for a long time
- Deployed in approx. 25 Mio Apple MacOS X systems
- Now also used in iPhone and iPod Touch

Code Sample:

```objective-c
@interface MyClass : NSObject
{id delegate;
}
-(float) getFloatValue;
-(void) setValue:(float) val;
@end

@implementation MyClass
-(float) getFloatValue
{
  return [delegate getFloatValue:self];
}
@end
```
So many Steps...
A Clarification

- **NeXTStep** - OpenStep implementation by NeXT (1989ff)
- **StepStone** - Company by Brad Cox that developed Obj-C
- **OpenStep** - API Specification by NeXT and SUN
- **Cocoa** - Apple's current OpenStep implementation
- **GNUstep** - FSF project to implement LGPLed OpenStep/Cocoa
- **GAP** - GNUstep Application Project (Desktop)
- **mySTEP** - special project to adapt all this to embedded devices
- **QuantumSTEP** - a project to add an Application Suite for Handhelds (not OSS due to some license issues)

*“quantum steps“ are minimal steps, but continuous progress, and tunnelling through barriers
The Characteristics of GNUstep

• GNUstep started in 1992 as FSF project (www.gnustep.org)

• FOSS (LGPL) implementation of the basic Frameworks (Base=Foundation & GUI=AppKit) plus some extensions

• GNUstep OS support: Linux, *BSD, Windows, MacOS X, Solaris, ...

• GNUstep Backend support: plain X11, libart, Cairo, ...

• Aims at being API compatible to MacOS X Cocoa

• GORM (Graphical Object Relationship Modeller) is the analogon to Interface Builder

• GNUstep Application Project: desktop applications
mySTEP was born 2003

Targets

• open source platform for Mobile Devices
• write once run everywhere
  • Cocoa compatible
  • develop Applications on Xcode / IB plus gcc-cross-compiler, debug and test on Mac - or GNUstep + GORM
  • no need for a simulator
• optimize for handhelds with touch-pen (maybe QWERTY, maybe finger operation) and small screen
  • but don’t change any bit of the API
• reuse from and contribute to GNUstep
• easy installation
Why mySTEP and not full GNUstep?

• different GUI visuals
• reduce library dependencies
• address efficiency problems on a FPU-less device
• we do need only X11 backend and no window manager
• development within Xcode plus Xtoolchain (not using autoconf/make)
• historical (mySTEP started independently and is now mixed with GNUstep)
Why consider another GUI Toolkit?

• *STEP is rich and mature
• choice
• People love Objective-C if they master the first steps
• (L)GPLed components are here to stay forever
A walk through mySTEP
Linux Distro

Linux Kernel

Device Drivers

libc, libm, libX11, libpng, ...

sh, ps, X11fb, hcitool, ...

Applications (Adresses, Mail, Notes, Viewer, ...)

QuantumSTEP

mySTEP

Foundation*, AppKit*, Addresses, Calendar, MenuExtra, PrefPanes, Install, Messaging, Small WebKit*, IOBluetooth, SysStatus, Sync, PDFKit, CoreData*

*source shared 100% with GNUstep
## Supported Devices and Kernels
(currently, more to come)

<table>
<thead>
<tr>
<th>Device</th>
<th>Distro</th>
<th>Linux</th>
<th>FPU</th>
<th>ABI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sharp Zaurus</td>
<td>Sharp ROM</td>
<td>2.4</td>
<td>Hard</td>
<td>OABI</td>
</tr>
<tr>
<td>Acer n30</td>
<td>Letux</td>
<td>2.6</td>
<td>Hard/Soft</td>
<td>OABI</td>
</tr>
<tr>
<td>Neo1973</td>
<td>OpenMoko</td>
<td>2.6</td>
<td>Soft</td>
<td>EABI (OABI*)</td>
</tr>
<tr>
<td>Mac</td>
<td>OSX</td>
<td>Darwin</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

* needs a special kernel and rootfs
The least common denominator determines the ABI

- Linux 2.4 (or 2.6)
- X11R6
- OABI
- Hardfloat FPU commands
- Hard/Softfloat libm
- gcc 2.95.3 with Obj-C
Installation

- Installer packages like on the Mac (.pkg and .mpkg bundles)
- overwrites/removes parts of the host OS that are not needed
- builtin installer (doesn’t help for first install)
- first install: through Shareware* “ZMacSync“ (installs the .mpkg through ssh on the remote device)

* installation function does not require a license key but donations are welcome
/System/Libraries/Frameworks

root@fic-gta01:/usr/share/QuantumSTEP/System/Library/Frameworks$ ls -l
drwxr-xr-x 3 501 root 0 Apr 7 10:48 AddressBook.framework
drwxr-xr-x 3 500 daemon 0 Apr 12 09:48 AppKit.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 Cocoa.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 CoreData.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 Foundation.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 IOBluetooth.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 IOBluetoothUI.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 Message.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 PreferencePanes.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 Security.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 SystemStatus.framework
drwxr-xr-x 3 501 root 0 Apr 7 10:49 SystemUIPlugin.framework
drwxr-xr-x 3 500 daemon 0 Apr 10 08:35 WebKit.framework
root@fic-gta01:/usr/share/QuantumSTEP/System/Library/Frameworks$
/System/Libraries/Frameworks
(File sizes - unstripped)

root@fic-gta01:/usr/share/QuantumSTEP/System/Library/Frameworks$ du -hd 1 .
15.0k ./SystemUIPlugin.framework
107.5k ./SystemStatus.framework
5.0M ./Foundation.framework
195.5k ./CoreData.framework
19.0k ./PreferencePanes.framework
2.9M ./WebKit.framework
13.5k ./Message.framework
12.0k ./Security.framework
88.0k ./AddressBook.framework
8.7M ./AppKit.framework
8.5k ./Cocoa.framework
12.5k ./IOBluetoothUI.framework
55.5k ./IOBluetooth.framework
17.2M .
root@fic-gta01:/usr/share/QuantumSTEP/System/Library/Frameworks$
Anatomy of a Framework

root@fic-gta01:/usr/share/QuantumSTEP/System/Library/Frameworks$ ls -lR AddressBook.framework/
AddressBook.framework/:  
  lrwxrwxrwx    1 501      root           28 Apr  7 10:48 AddressBook -> Versions/Current/AddressBook
  lrwxrwxrwx    1 501      root           24 Apr  7 10:48 Headers -> Versions/Current-Headers
  lrwxrwxrwx    1 501      root           26 Apr  7 10:48 Resources -> Versions/Current/Resources
  drwxr-xr-x    3 501      root           0 Apr  7 10:48 Versions

AddressBook.framework/Versions:
  drwxr-xr-x    5 501      root            0 Feb 15  2008 A
  lrwxrwxrwx    1 501      root            1 Apr  7 10:48 Current -> A

AddressBook.framework/Versions/A:
  drwxr-xr-x    2 501      root            0 Jan 15  2008 Headers
  drwxr-xr-x    2 501      root            0 Apr  7 10:48 Resources
  drwxr-xr-x    3 501      root            0 Apr  7 10:48 arm-quantumstep-linux-gnu

AddressBook.framework/Versions/A/Resources:
  -rw-r--r--    1 501      root         2906 Jan 15  2008 Info.plist

AddressBook.framework/Versions/A/arm-quantumstep-linux-gnu:
  lrwxrwxrwx    1 501      root           17 Apr  7 10:48 AddressBook -> libAddressBook.so
  drwxr-xr-x    2 501      root           0 Apr  7 10:48 Headers
  -rwxr-xr-x    1 501      root          83952 Feb 15  2008 libAddressBook.so

AddressBook.framework/Versions/A/arm-quantumstep-linux-gnu/Headers:
  lrwxrwxrwx    1 501      root           13 Apr  7 10:48 AddressBook -> ../..//Headers
root@fic-gta01:/usr/share/QuantumSTEP/System/Library/Frameworks$
Foundation.framework
What API provides

- Unicode strings, containers, calendar dates, time zones, index sets, timers, tasks, threads, notifications, interprocess communication, procedure calls, files, communication & connections, ZeroConf, XML, ...
Foundation.framework
Special issues on embedded systems

- ARM Hard/Softfloat mixture
- mixed endianness for double (when reading binary data)
- NSInvocation without libffcall
- printing logs is quite slow (date calculations...)
- memory efficiency
AppKit.framework
What the API provides

- Primitive objects: Fonts, Colors, Bezier Paths, Images, Sound, Events, Layout manager, ...
- Rendering: fill, stroke, draw, blit, shadow, glyphs
- MVC GUI Basics: Application, Controllers, Documents, Windows, Views, Cells
- Widgets: Buttons, Tables, Text, Scrollers, ...
- Other UI: Speech Synthesis & Recognition, Spellchecker, ...
AppKit on Embedded
Squaring the Circle

- Device limitations
  - QVGA...VGA resolution in portrait mode
  - just one pointer, usually no keyboard
  - no FPU but float based coordinate transforms

- Create fancy effects while being 100% compatible to given API

- Compatible to Cocoa API (which assumes a Desktop)

- Reasonable Operation on a Handheld even without keyboard

- Pen operation vs. Finger operation

iPhone uses UIKit which is NOT compatible to AppKit!
No write once run everywhere.
AppKit
It is possible...

• Use intelligent caching to reduce float and drawing operations
• Resolution independent graphics
• Autosizing of all windows to visible screen (if resizable)
• long-press to open context menus (without a right mouse button)
• -orderFrontKeyboardPalette: open input daemon
• Further ideas:
  • smooth scrolling (using triple buffering)
  • ordering of windows (slide in/out like iPhone)
  • speech/optical character recognition
AppKit
Input Methods

Write your own plugin...
Small WebKit.framework
Targets

- Obj-C only (no need for Obj-C++ on destination machine)
- (X)HTML 4.0, DOM-Tree, ECMAScript, CSS, no Editing
- 100% API compatible to WebKit.framework
- uses NSTextView and NSAttributedString as the rendering engine

Examples later
Small WebKit.framework

Status

• (X)HTML works quite well
• ECMAScript is loaded and mostly parsed - no execution
• CSS is loaded but not processed
• works on GNUstep, mySTEP, and Cocoa
• mySTEP ARM binary: ~1.6 MB (unstripped libWebKit.so)
## Comparison of *step Platforms

<table>
<thead>
<tr>
<th></th>
<th>OSX</th>
<th>GNUstep</th>
<th>mySTEP</th>
<th>iPhone (unjailed)</th>
<th>iPhone SDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foundation</td>
<td>Base</td>
<td>yes</td>
<td></td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>AppKit</td>
<td>GUI</td>
<td>yes</td>
<td></td>
<td>UIKit</td>
<td></td>
</tr>
<tr>
<td>CoreData</td>
<td>GSCoreData</td>
<td>no (?)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IOBluetooth</td>
<td>-</td>
<td>yes</td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>WebKit</td>
<td>Simple WebKit</td>
<td>yes</td>
<td></td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>PDFKit</td>
<td></td>
<td>yes</td>
<td></td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>SyncServices</td>
<td>-</td>
<td>(partially)</td>
<td>?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AddressBook</td>
<td>yes</td>
<td>yes</td>
<td></td>
<td>SQL (?)</td>
<td></td>
</tr>
<tr>
<td>CalendarStore</td>
<td>-</td>
<td>yes</td>
<td></td>
<td>SQL (?)</td>
<td></td>
</tr>
<tr>
<td>ImageKit</td>
<td>-</td>
<td>-</td>
<td></td>
<td>?</td>
<td></td>
</tr>
</tbody>
</table>

*we may know more by Tuesday...*
Some Application Examples built on mySTEP
Bluetooth Pairing

This example also shows some bugs that are still to be fixed:

- Battery level shows -1%
- „Show All“ button clipping box
- Image rendering does color errors for icons with alpha
- Menu is too wide - should be scrollable or resized
GNUchess frontend
~1000 LOC

MacOS X version

36 kByte ARM binary
Sudoku generator
just 200 LOC

MacOS X version

16 kByte ARM binary
Web Browser using Small WebKit

<html>
<head>
<title>Formatting test</title>
</head>
<body bgcolor=#884422>
<h1>H1 header</h1>
<h2>H2 header</h2>
<h3>H3 header</h3>
Plain text
<b>Bold text</b>
<i>Italics text</i>
<u>Underlined text</u>
<strike>Strikethrough text</strike>
<tt>TT text</tt>
<em>Emphasized text</em>
<big>Big</big>
<small>Small</small>
<sub>Subscript</sub>
<sup>Superscript</sup>
<strong>Strong</strong>
...
</body>
</html>
More Examples

Navigator Demonstrator
just 200 LOC
20 kByte ARM binary

Phone Dialer
just 450 LOC
37 kByte ARM binary
Time for a Demo?

- Download **Polygons** sample code from Apple Developer pages
- Adapt to MacOS X 10.4 (Xcode format, Upgrade, > 10.2 NIB)
- Add a „Run Script Build Phase“ to Cross Compile:

```bash
export SOURCES="main.m NSBezierPath_Polygons.m NSViewExtensions.m PolygonAppController.m StringArtController.m StringArtView.m"  # all source codes
export LIBS=  # add any additional
export FRAMEWORKS=  # add any additional Frameworks (e.g. AddressBook) etc. (adds -l and -L)
export INSTALL_PATH=/Applications/Test  # override INSTALL_PATH for MacOS X for the Zaurus

#export STRIP_Framework=true  # set to true if you want to strip headers etc.
#export STRIP_MacOS=true  # set to true if you want a Linux-ARM only bundle
#export INSTALL=true  # true (or empty) will install locally to $ROOT/$INSTALL_PATH
#export SEND2ZAURUS=true  # true (or empty) will try to install on the Zaurus at /$INSTALL_PATH (using ssh)
#export RUN=true  # true (or empty) will finally try to run on the Zaurus (using X11 on host)
export ROOT=$HOME/Documents/Projects/QuantumSTEP  # project root
/usr/bin/make -f $ROOT/System/Sources/Frameworks/mySTEP.make $ACTION
```

- Build & Run on Linux/X11 Handheld
What is coming next

- Speed Optimization
- Transparent Graphics (Xrender)
- Resolving some Memory Leaks
- Look needs more polishing
- Improved input methods for Text
- Improve useability (e.g. size of window controls)
- 10.5 compatibility
Invitation

We need:

• users (private, research and commercial) who give constant feedback
• developers

What is needed to participate?

• Enthusiasm for Objective C and Cocoa
• A Mac and a compatible device would be nice
Questions?
Further Information

GNUstep and mySTEP / QuantumSTEP

http://www.gnustep.org
http://www.quantum-step.com

Sponsor: Golden Delicious Computers GmbH&Co. KG
(consulting and The Handheld Linux Shop)

http://www.goldelico.com
Thank you!