

Introduction to the GPE Palmtop Environment and the OpenEmbedded Project

Phil Blundell <pb@handhelds.org>

Hello GPE!



Application suite for mobile devices

“GNOME for Handhelds”

Typical handheld computer

iPAQ h2210

400MHz CPU

64MB RAM

32MB Flash

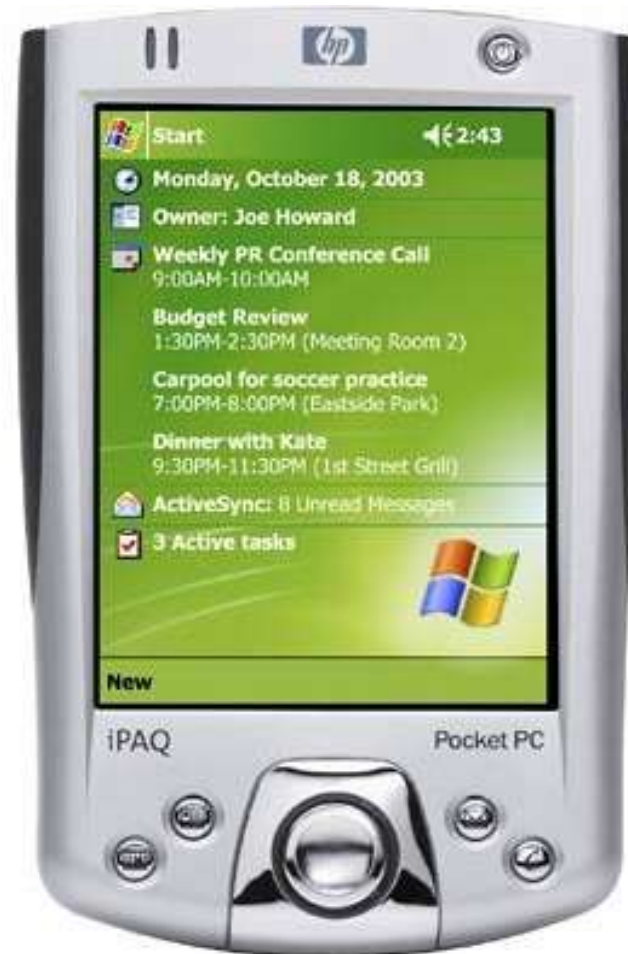
320x240 pixels

About €350



But...

What is wrong with this picture?



GPE Vision

Handheld environment based on open standards.

No reliance on “host” system.

Composed entirely of free software.

But vendors can create proprietary applications for specific vertical markets.



It's just a computer

Run standard application software as far as possible.

Interoperate with other computing devices as equals.



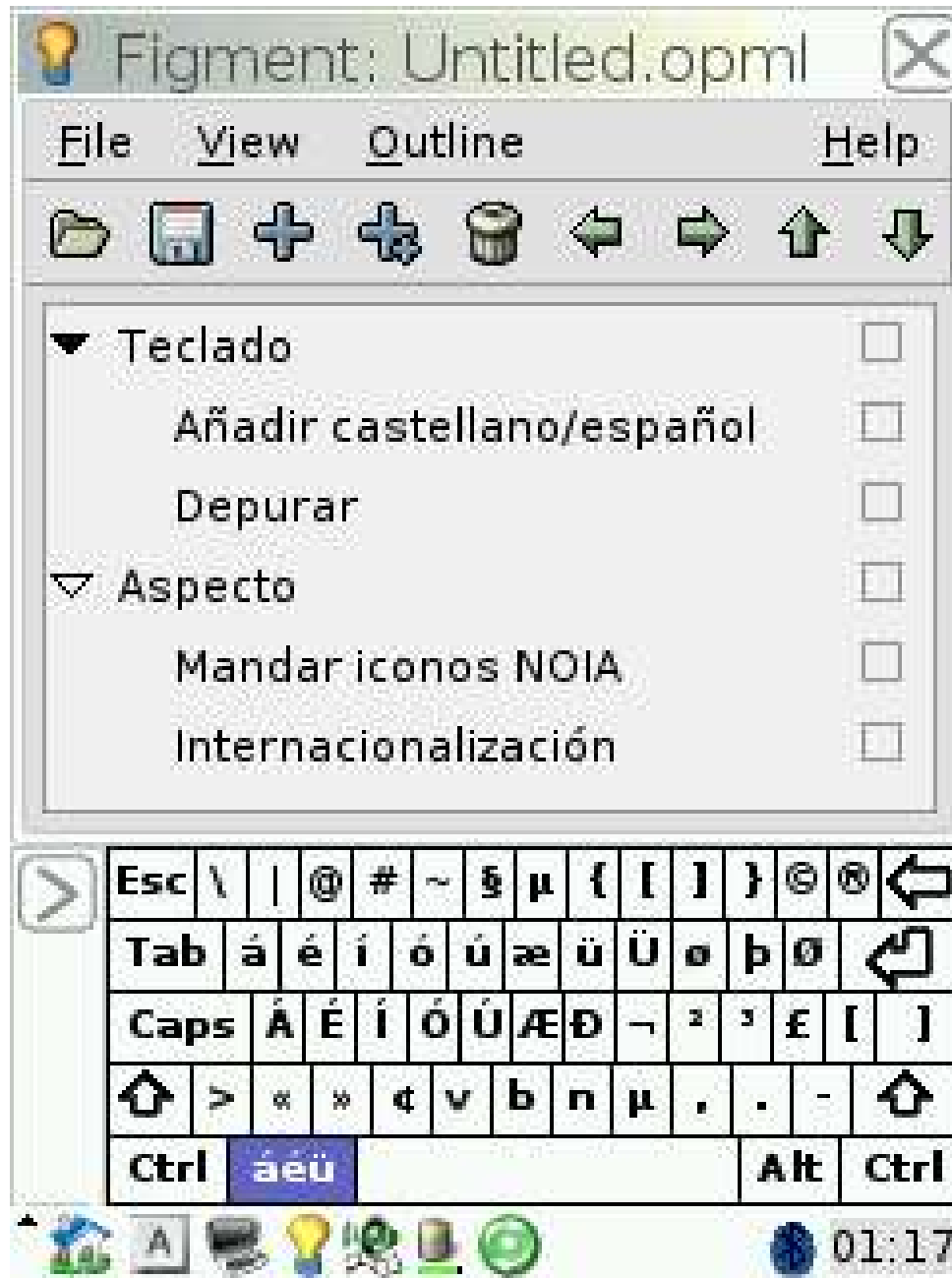
GPE Overview

Based on the X Window System
and the GTK toolkit

Existing GTK applications will run
with few/no changes

Core applications licensed under GPL, libraries
under LGPL





GPE Applications

Standard PIM suite

- calendar, addressbook, to-do list
- SQL database
- vCard/iCalendar import and export

Control panel applets

- System configuration, wireless LAN setup, etc



More applications

Linphone

- SIP voice-over-IP client
- As used on desktop, with UI tweaked for small screens

Minimo

- Mozilla browser variant for small devices
- Same rendering engine as Firefox, etc.
- Disk footprint below 20MB;
usable with 64MB RAM.



More applications

Audio player

- GStreamer framework with custom UI
- Integer-only decoders for MP3 and Vorbis

File manager

- Based on GnomeVFS



Home



Documents

Projects

Games

Settings

- To-do list
- Summary
- Time Tracker
- Contacts
- Web browser
- Figment
- Calendar
- Sketchbook

- Utilities
- Games
- Settings
- Other

- Desktop
- Log out



Calendar



Figment



Time Tracker



Summary



01:27



UI Challenges

Small screen

- Glade XML files useful for UI tweaking
- Custom GTK themes

Keyboardless text input

- Handwriting recognition
- On-screen virtual keyboard

Display migration

Running applications can be moved from one display to another

GPE Teleport

- Utilises multi-display support in GTK 2.2+
- Need to take care of security!



UI Challenges

Touchpanel input

- Only one “button”
- Can't move pointer without clicking

Mouse-over tooltips not usable

- Modal help system

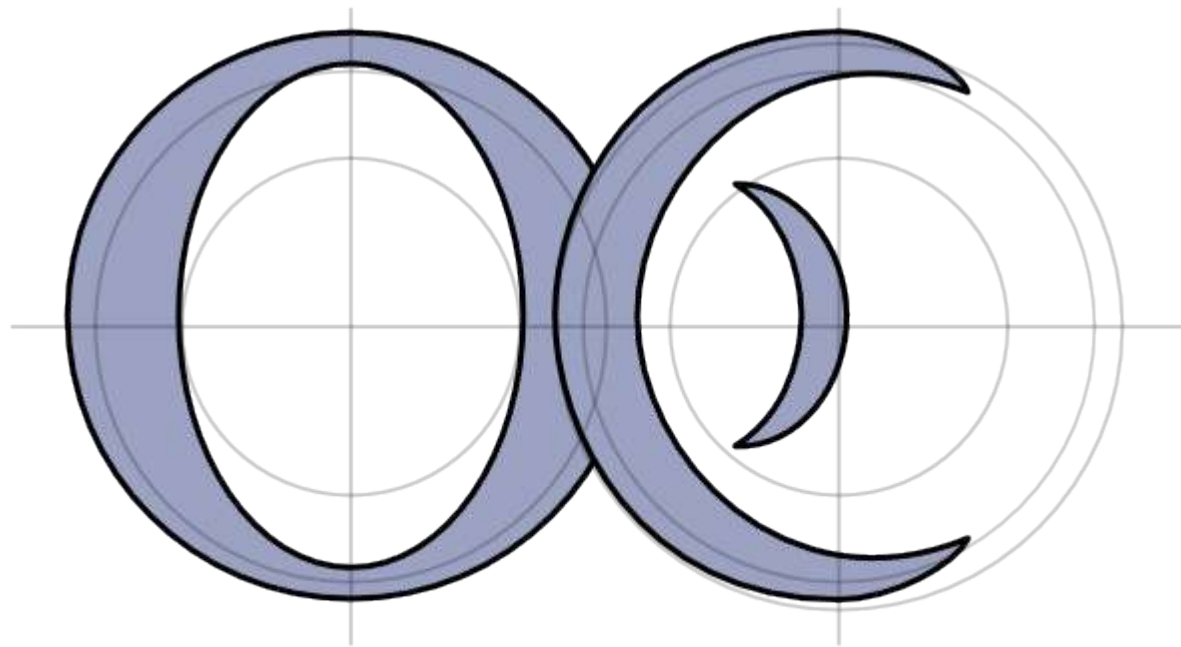


Development environment

Native compilation on target system not really practical.

Cross-compilation environments can be cumbersome to set up.





openembedded

Self-contained build environment

Includes compiler and runtime libraries

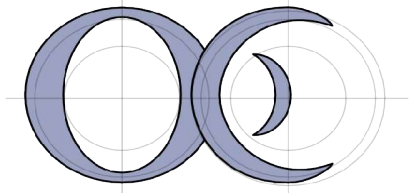
OE Overview

Grew out of OpenZaurus project

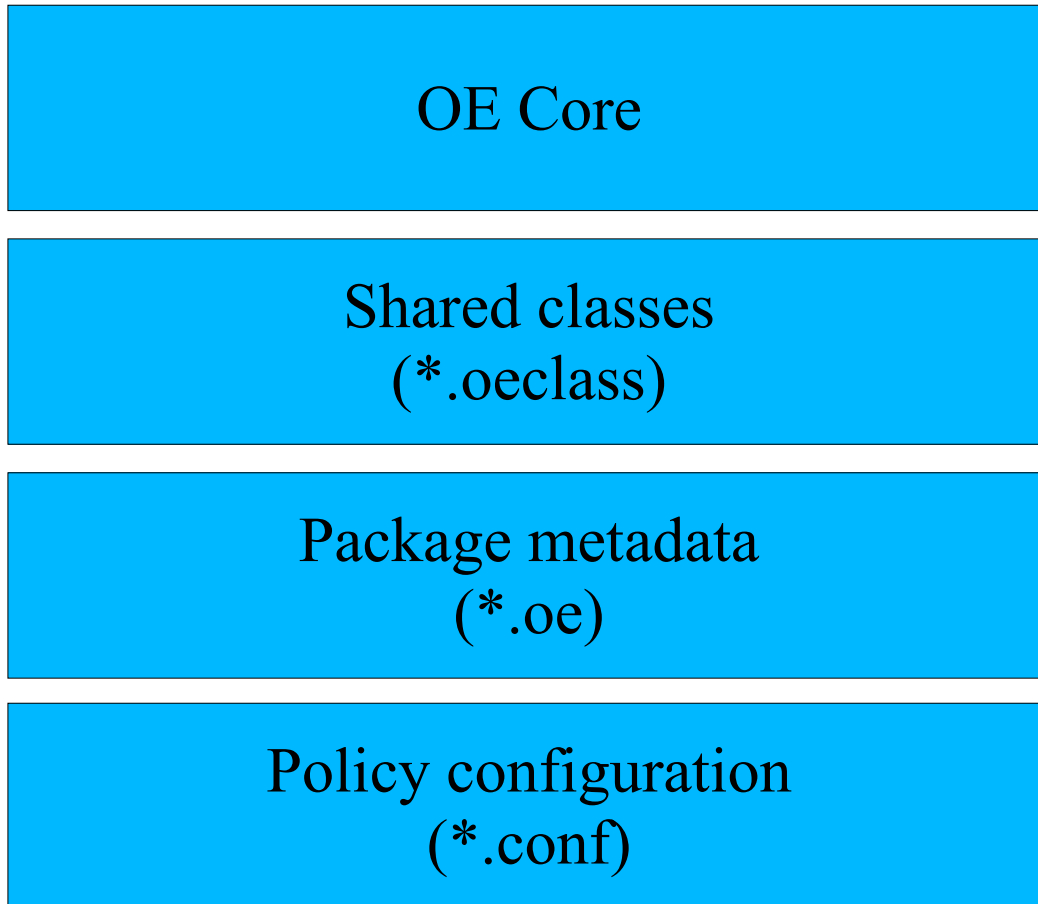
Inspired by Gentoo “portage”

Not tied to any target, project or set of build tools

Simple declarative syntax for package metadata



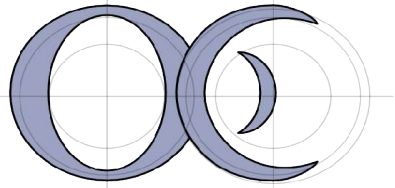
OE Structure



Independent of target
and project



Target and
project dependent



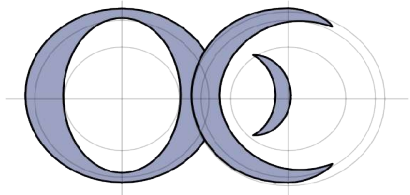
Writing a package description

```
SRC_URI = "http://ftp.gnu.org/gnu/hello/hello-${PV}.tar.gz"
```

```
DESCRIPTION = "Traditional one-line greeting"
```

```
LICENSE = "GPL"
```

```
inherit autotools
```



Building with OE

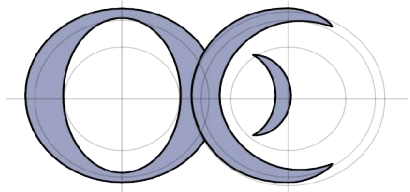
1. Configure important variables in local.conf

2. Set up environment variables

```
export OEFILES = $HOME/oe/packages/*/*.oe
export OEPATH = $HOME/oe/packages:...
```

3. Start the build!

```
oemake gpe-image
```



Standard variables

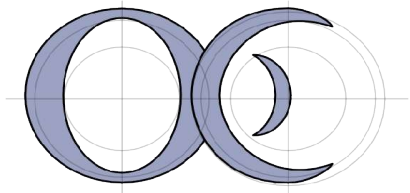
Package name and version

PN, PV, PR

Relationships to other packages

DEPENDS, PROVIDES

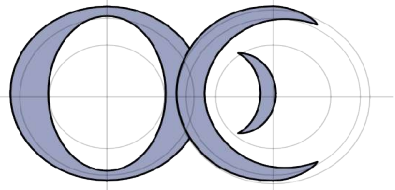
RDEPENDS, RPROVIDES, RRECOMMENDS



Custom methods

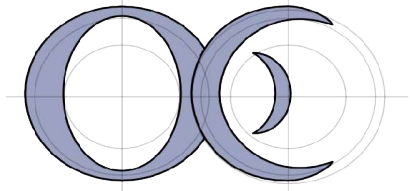
```
do_compile () {  
    oe_runmake wermitt \  
    CFLAGS="${CFLAGS} -DLINUX -DCK_POSIX_SIG \\  
    -DNOTCPOPTS -DLINUXFSSTND DNOWHATAMI "  
}
```

```
do_install () {  
    install -d ${D}/${BINDIR} ${D}/${MANDIR}  
    ${D}/${INFODIR}  
    oe_runmake 'DESTDIR=${D}' 'MANDIR=${D}/  
    ${MANDIR}' install  
}
```



Python interaction

```
def qte_arch(d):  
    import oe, re  
    arch = oe.data.getVar('TARGET_ARCH', d, 1)  
    if re.match("^i.86$", arch):  
        arch = "x86"  
    elif arch == "x86_64":  
        arch = "x86"  
    elif arch == "mipsel":  
        arch = "mips"  
    return arch  
  
QTE_ARCH := "${@qte_arch(d)}"
```



Getting involved

Both GPE and OpenEmbedded need more developers!

Visit us on the web:

<http://gpe.handhelds.org/>

<http://openembedded.org/>