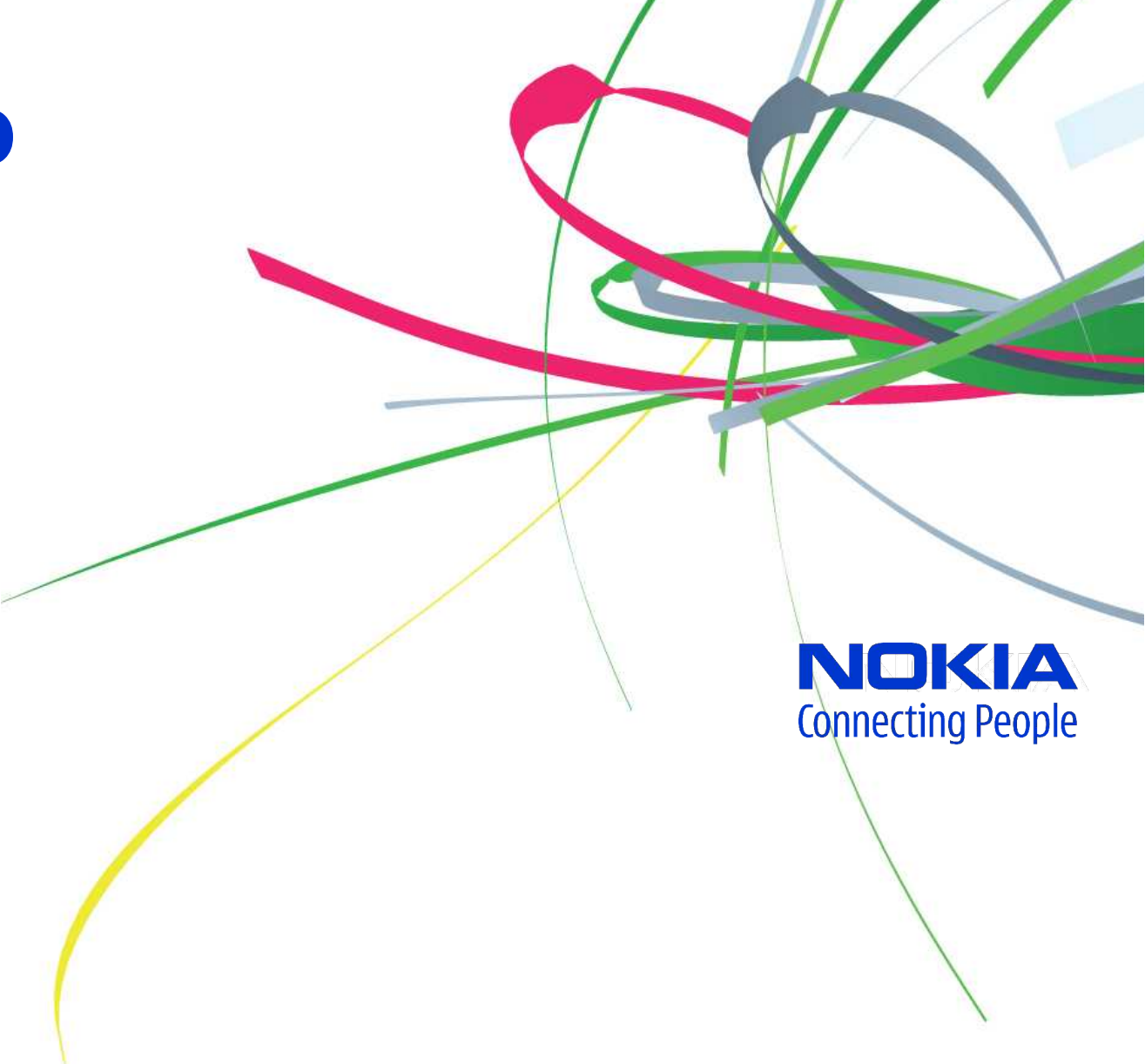


Qt on Maemo

Kate Alhola

Antonio Aloisio

An abstract graphic in the top right corner consisting of several overlapping, curved lines in green, yellow, and magenta, creating a sense of motion and connectivity.

NOKIA
Connecting People

Forum **NOKIA**

9.8.2008

Kate Alhola



- Maemo Chief Engineer in Forum Nokia
- Long term Open Source developer, first contributions 8-bit microprocessor in early 80's
- Linux kernel driver from early 1.x kernels
- Katix RTOS with IP stack for PC, 68K and PPC
- Before Nokia, long career embedded Linux and RTOS related development in small subcontractor companies
- Numerous embedded HW designs with 6809, 68xxx and PPC

Antonio Aloisio (GnuTon)



- Maemo Qt developer in Forum Nokia
- KDE4 Kblogger main developer
- He has contributed to the developing of KDEPim/KBlog

**Easy, out of the box Qt 4.4
will compile and run
in maemo internet tablets**

**Maemo is just
like any other Linux distro, is it ?**

- maemo is like any Linux distro,
 - based on same standard open source components and debian packaging
- You can compile most of applications to Maemo without any modifications
 - in many cases this just don't make sense !!!!
- Maemo is mobile optimized and applications should be mobile optimized as well
- Qt should be mobile optimized
- Maemo qt4 port in adding maemo/hildon mobile optimizations to Qt

Mobile optimized



- Maemo is optimized for mobile internet tablets
- Maemo is not for desktops
- Maemo is not small screen mobile phones
- No stuff that wont fit in your pocket
 - No hard disk
 - No full size keyboard
 - No mouse
 - No big heavy battery
 - No big screen
- No stuff that spoils your internet experience
 - No keyboard only navigation
 - No mini size screen
 - Not limited to one toolkit,

Internet tablet

- Beats any mobile phone or PC in mobile internet usage
- 4.1inch 800x480 lcd touch screen
- Flash memory
- Hours of continuous internet usage
- Full multimedia capabilities, speakers, microphone, camera, codecs, DSP
- Connectivity Wlan, bluetooth, Wimax
- Alphanumeric mini keyboard and GPS in N810

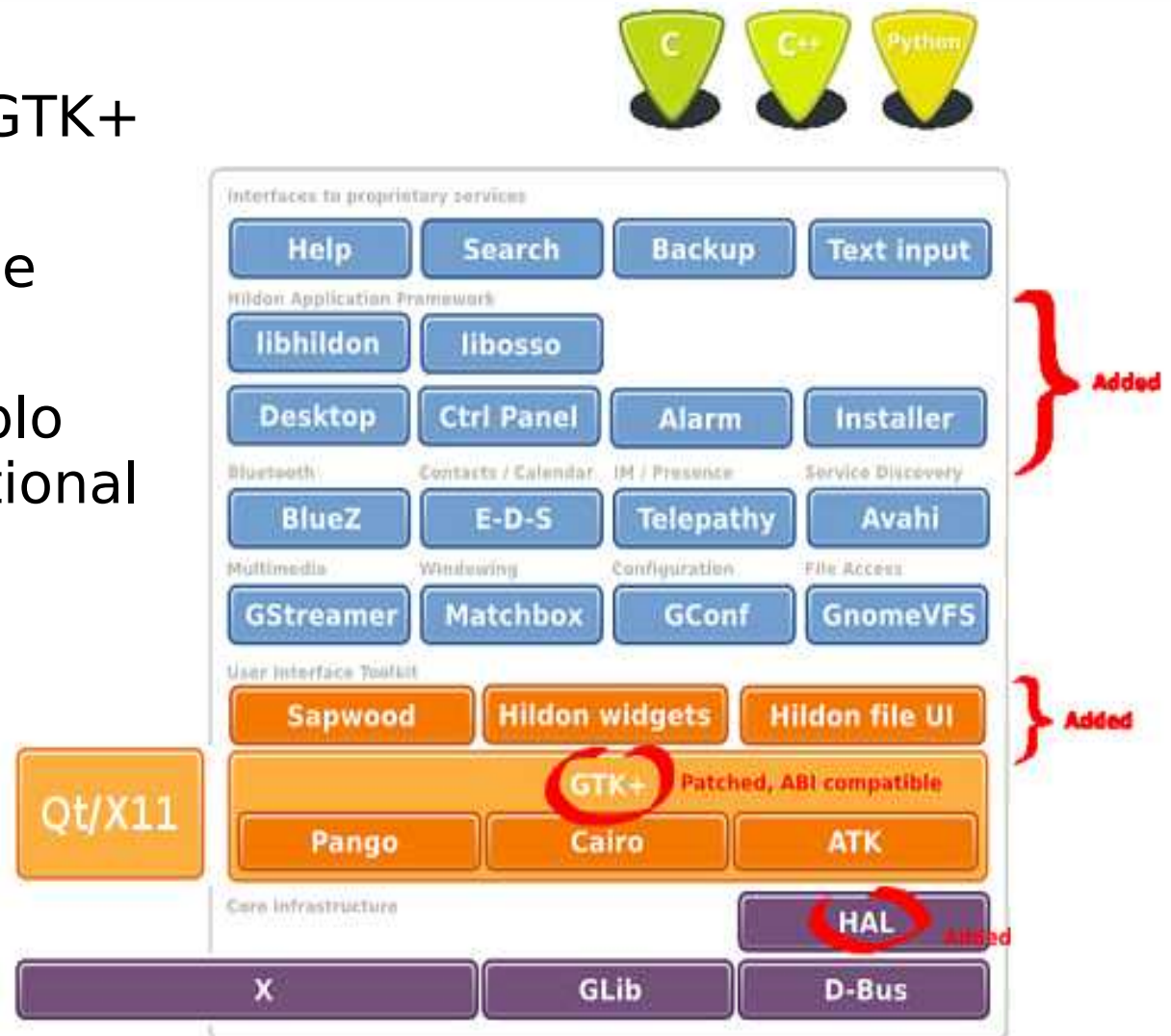


Maemo Linux for internet tablets

- Small form factor 800x480 touch screen 256M flash memory
- Takes advantage of device's multimedia capabilities including DSP all device connectivity features
- Strongly optimized power management allows long battery life and small device battery size
- On screen virtual keyboard or mobile optimized small size qwerty keyboard
- maemo versions up to 4.1 Diablo was based GTK+
- Open architecture with X11 that allows to run any GUI toolkit like Qt or Java

Maemo layers

- Original maemo was GTK+ based
- X11 makes easy to use other GUI toolkits
- For maemo up to Diablo release, Qt is an additional component



UI Framework progression



Fremantle
Hildon & GTK+ evolved



Harmattan
Qt integrated

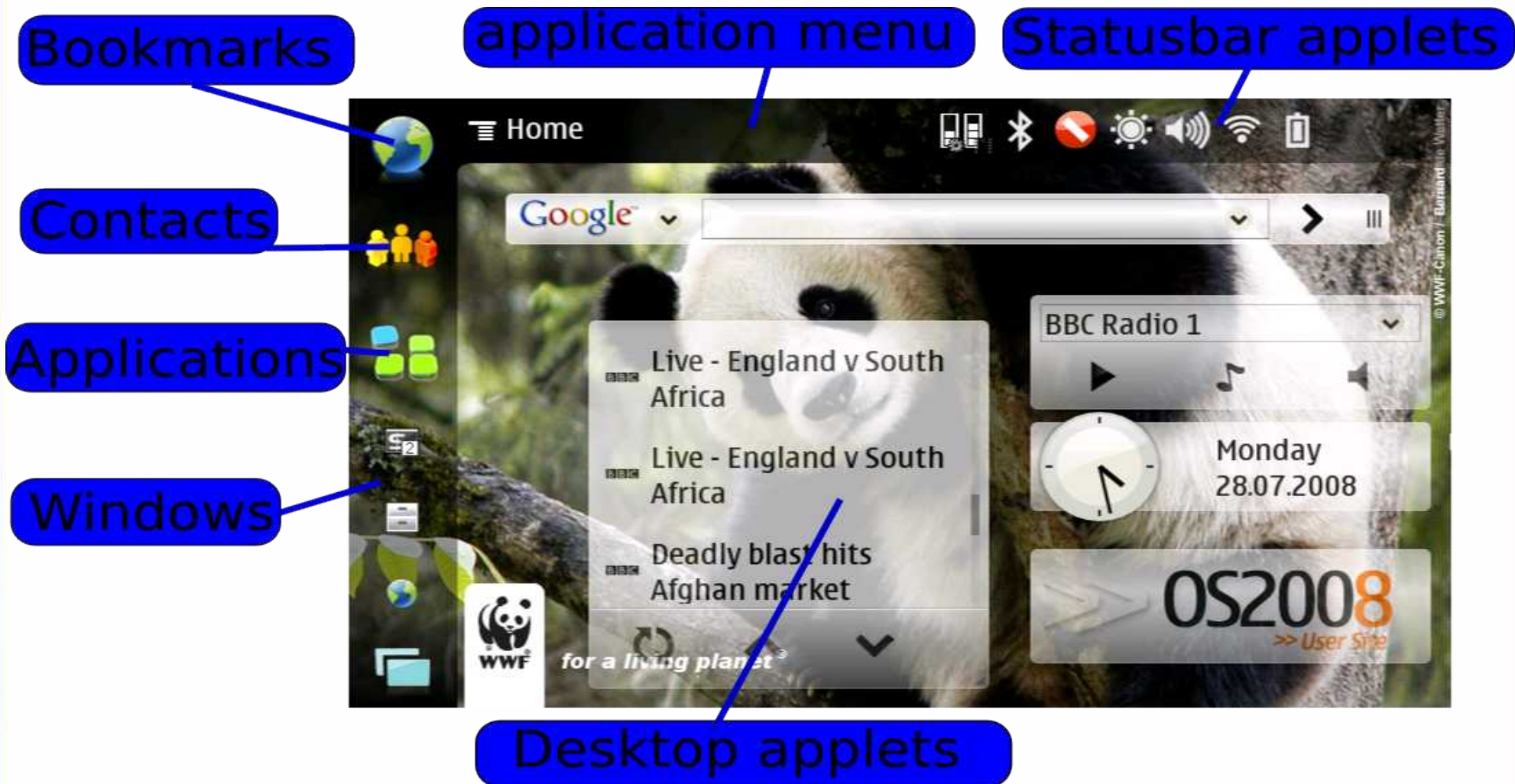


Consolidation of essential parts of the platform

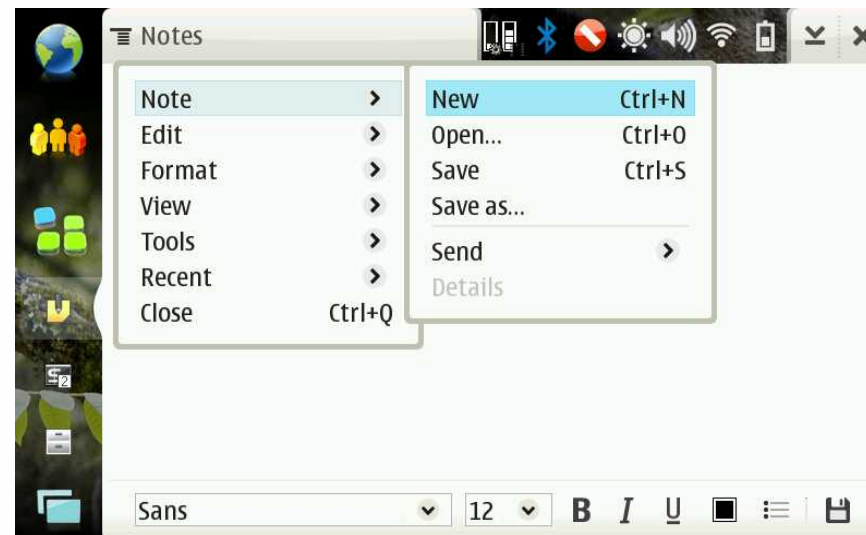
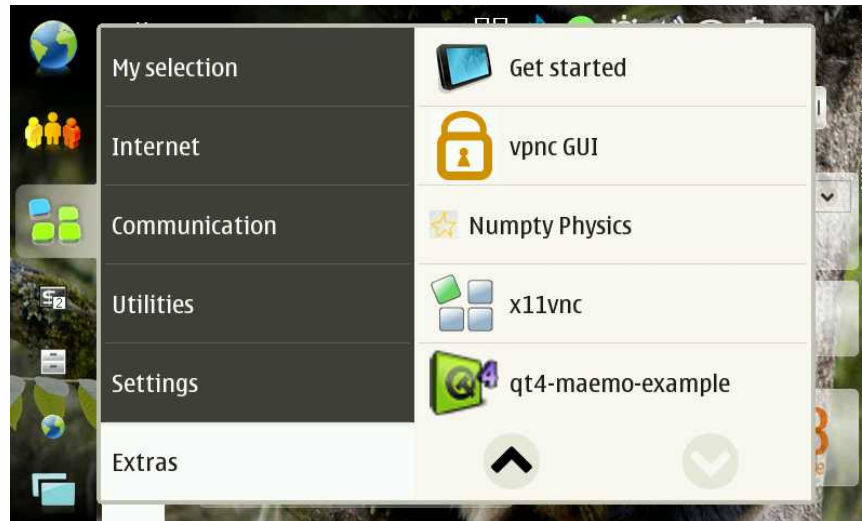
Image: *Industrial-Glade*, by John Brian Silverio. CC Attribution-Non-Commercial-No-Derivative-Works License

- Current maemo qt4.4 port is experimental community port
- There is discussions to integrate maemo-hildon support to qt4.5 release
- Maemo Harmattan release will have full qt support integrated

Screen layout

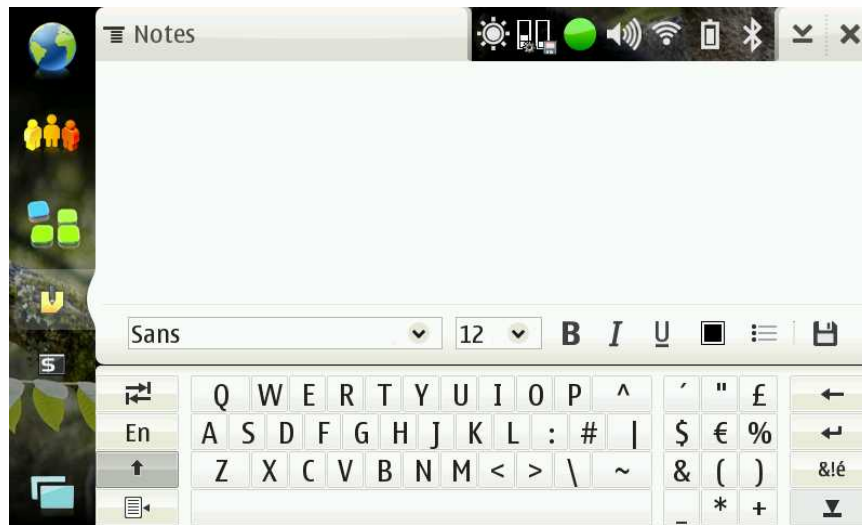


Mobile optimized UI

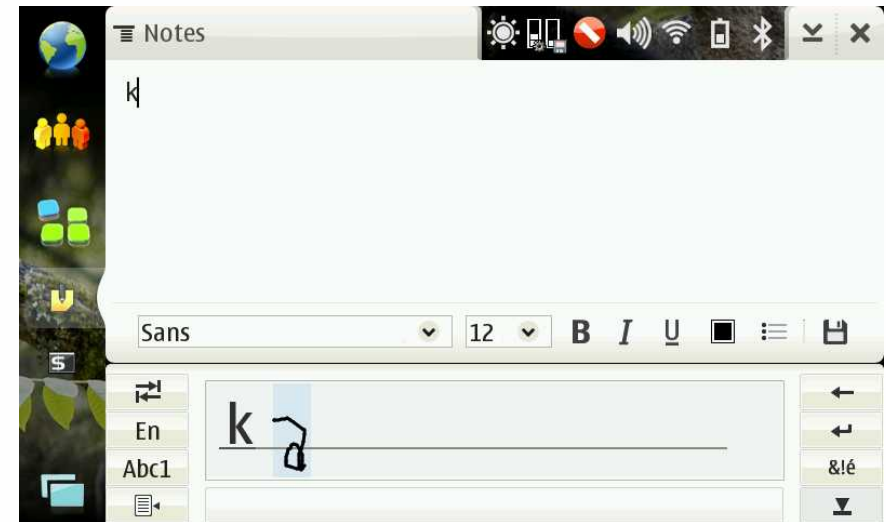


Multiple forms of IM

Virtual keyboard



Handwriting mode



Thumb mode



Qwerty

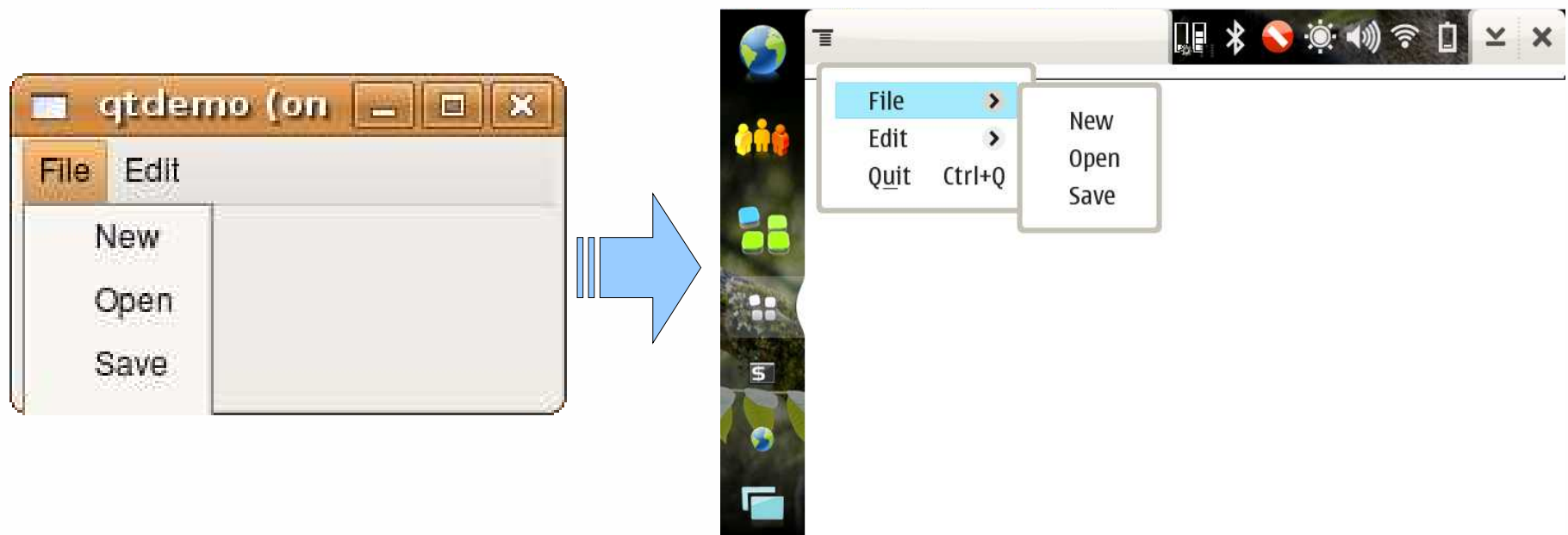


What is needed for maemo Qt

- Hildon menu
- Hildon input method
- maemo theming
- Debian packing for maemo
- Gnome VFS (or GVFS or KIO)
- Optimizing widgets for mobile screen
- OpenGL-ES support

Maemo menu

- Application main menu is not in application owned window, it is in window manager/desktop owned area
- Application receives X11 Grab transfer event to show menu
- In full screen mode, the menu bar is not visible
- Tablet has hardware menu button, keycode as F4
- No changes are required to existing applications



Input method

- tablet keyboard uses normal X key events
 - there are many mandatory features that need to be supported
- small keyboard has a different modifier configuration
- Modifiers have three modes, “**traditional**” pressed same time with the key, “**sticky**”, pressed before key or “**lock**” pressed twice and modifier is active until pressed third time.
- Fn modifier to access numbers and special characters
- Char modifier to access some characters not in keyboard
- Auto completion mode
- Input mode can be changed by application



Input Method

Let's look at an Application to understand how the HIM Works



Input Method

- The user gives the focus to a key entry widget
- A new window appears at the bottom of the screen



Input Method

- We have two different windows
 - The application (Qt)
 - The Main HIM User Interface (GTK+/Hildon)



Input Method

BASIC ARCHITECTURE

- The key entry widgets interact with the Main HIM UI using the QhildonInputContext that sends and receives XMessages.



How does it work?

- When the user wants to insert text:
 - The key widget receives the focus
 - The Context underneath the widget
 - Finds the window of the main HIM UI (XgetWindowProperty request)
 - Tells the HIM UI to popup
 - The HIM UI pops up at the bottom of the screen

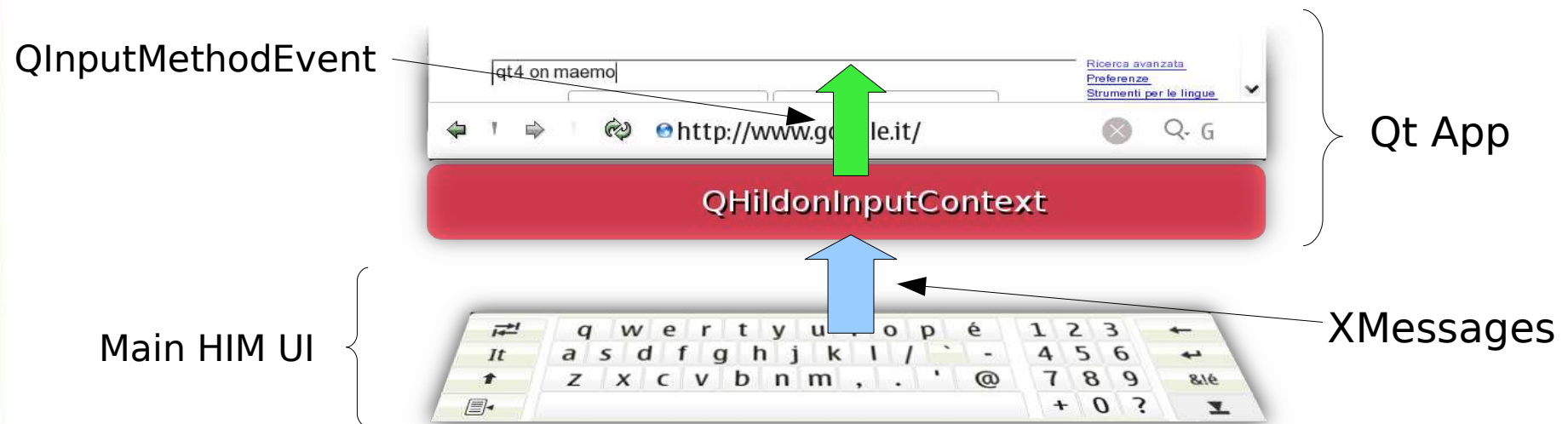
The main HIM UI popped up is different when the hardware keyboard is closed or opened.

What happens in these cases?

Input Method

When the keyboard is closed:

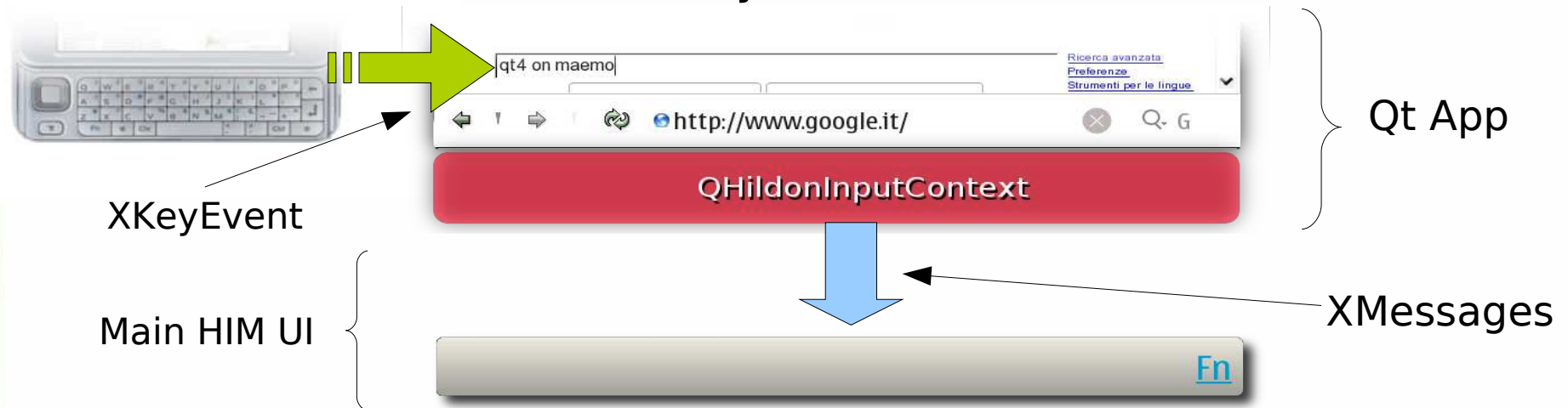
- The user uses the main HIM UI to add text.
- The main HIM UI sends these data via XMessages
- The Qt application:
 - Receives the XMessages
 - QApplication::x11ClientMessage filters them;
 - The messages that match a specified Atom type are dispatched to the QHildonInputContext.
- QHildonInputContext dispatches the string received/elaborated to the focus key entry widget via QInputMethodEvent



Input Method

When the keyboard is opened:

- The QHIC installs an event filter in the widget that receives the focus
- The user uses the hardware keyboard to add text.
- X11 dispatches the XKeyEvent events directly to the Application
- The key events are filtered:
 - The modifier keys set an internal bit mask.
 - The Dead keys are stored in the buffer memory and then combined with letters
 -
- The QHIC Sends elaborated keysim, state, etc.. to the main HIM Ui.



Input Method

- The Main Ui Receive the XMessages and it can:
 - Show the extended keyboard
 - Display the modifiers status
 -



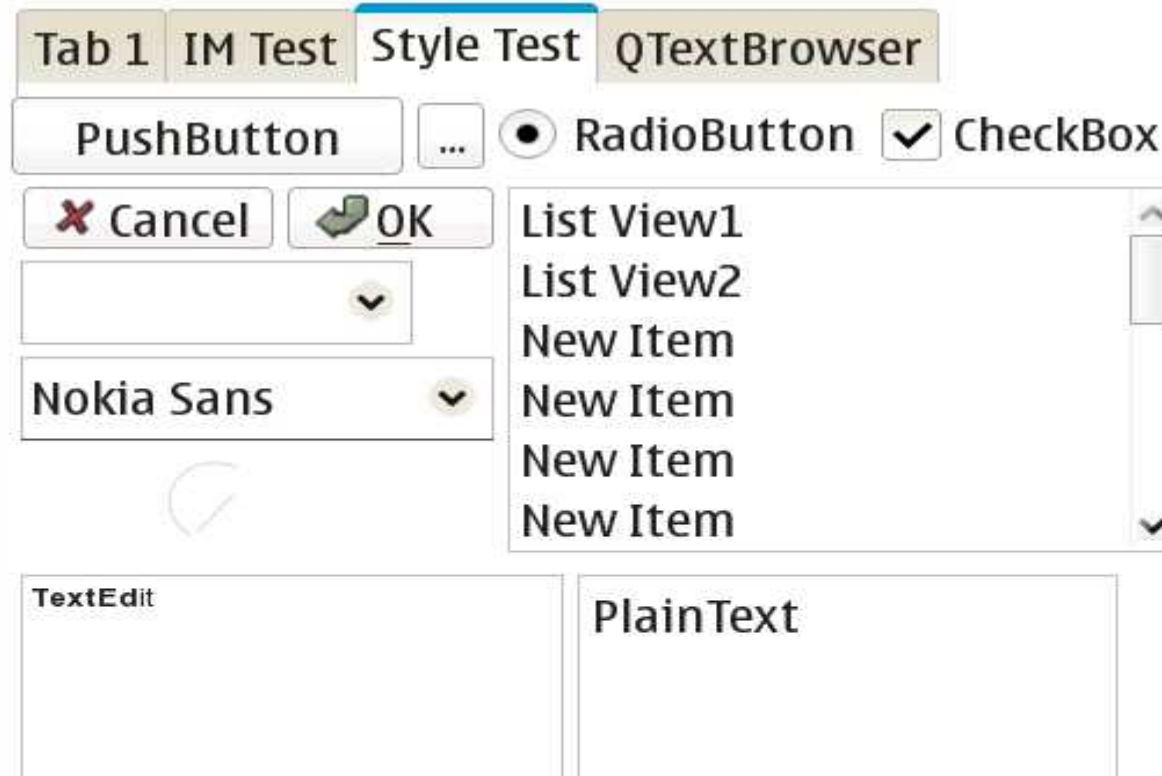
Virtual Filesystem



- Current maemo uses Gnome VFS for virtual file system view
- User view to local file system, System directories etc are hidden
- Transparent access to bluetooth resources, as example mobile phone file system via OBEX and shared media via UPNP
- Gnome VFS is deprecated
- Should we consider KIO or GIO ?
- Not implemented in current maemo Qt

Maemo styles

- Maemo uses optimized sapwood theming engine
- Maemo Qt uses sapwood themes
- Themes are optimized for small screen, larger sizes, larger fonts



- Qt application in Maemo uses QGTKStyle to change its Look & Feel
- QGTKStyle:
 - is a Qt style rendered using GTK
 - uses QCleanLookStyle as base style
- In Qt for Maemo, QGTKStyle:
 - has been integrated
 - is the default style
 - supports hildon specific theme element
(e.g. QSpinBox looks like a HildonNumberEditor).

How does It work?

AS any Qt style:

- uses the widget painters with the style options to draw:
 - Primitive element (e.g. FrameFocusRects)
 - Control element (e.g MenuItems)
 - Complex Control (e.g SpinBoxes)
- sets the relative positions of an element
- gives new hints to a widget (e.g. Scrolling in Qmenus)

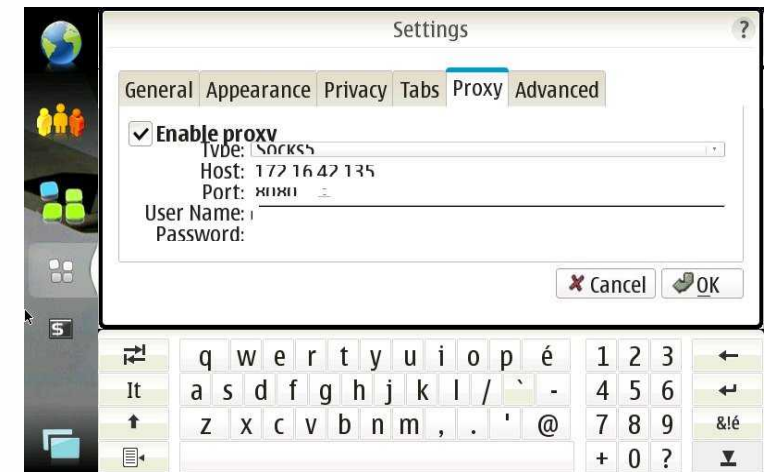
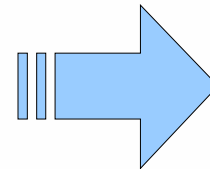
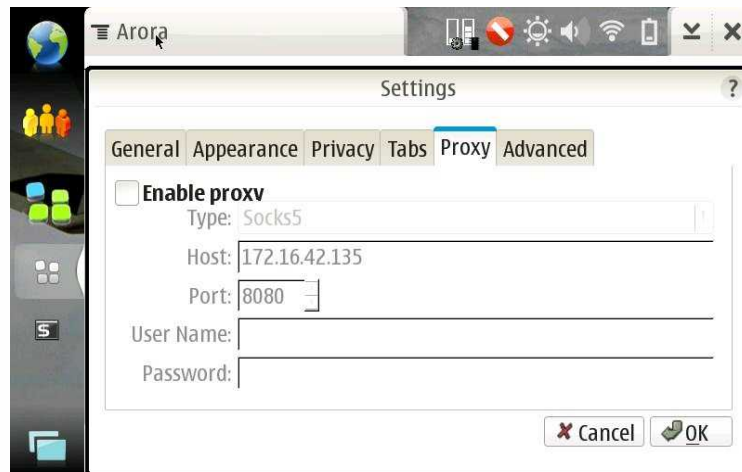
BUT it uses GTK to render, so it:

- Resolves the GTK+/Hildon symbols
- Creates and stores the GTK+/Hildon widgets in a Map
- Uses GTK+ functions to paint widget and to retrieve pixel metric values and style hints.

- Internet Tablet OMAP supports OpenGL-ES
- Nokia has not yet released OpenGL-ES drivers for N800 and N810 because current TI driver is not GPL compliant
- We are working to get developer version of OpenGL-ES released
- If we can resolve licensing issues and released driver the Qt should also have OpenGL-ES support

When porting application to maemo

- Remember that the screen is small
- Remember, that user is using finger or stylus, try to allow using finger always when possible
- Allow scrolling by finger from content pane, not from small scrollbar with stylus
- Don't use absolute layout, in maemo style, fonts, buttons etc are much larger than in desktop styles
- When on screen virtual keyboard is displayed, arrange your dialogs and inputs so that they will fit in same screen.



Useful Links:

- Maemo Qt4 project website:

<http://qt4.garage.maemo.org>

- Maemo Qt4 Developer ML:

<https://garage.maemo.org/mailman/listinfo/qt4-devel>