

Let Tux work in your hands

Oliver Fels

project manager mobile solutions

InnoTek Systemberatung GmbH

oliver.fels@innotek.de

On most conferences important people would like to demonstrate their importance by carrying important digital tools in their pockets, hands or jackets. LinuxTag is probably no exception. However when it comes to sticking with everybody's beloved operating system, it seems to be not so easy always carrying a tux based laptop in the palm of your hand.

So the first question to answer is: "Can I" ? And the second question follows: "What can I do with it besides demonstrating importance ?"

First things first– Why another PDA OS ?

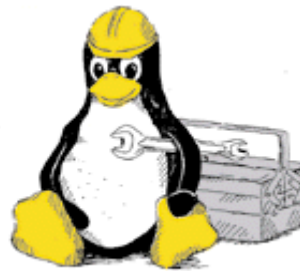
Currently, the market does not offer much choice– there is Microsoft's PocketPC which scares hard core Linux nerds for obvious reasons, and the Palm based devices from Palm Computing, Handspring or Sony. Both systems have their specific advantages– PPC is multi-colored, user friendly, multi colored, supported by a large developer community and, very important, it is even more multicolored.

The beauty of Palm devices is that they are cheap, more productive than PPCs and have a very large users base– plus a huge pool of applications. They interoperate with different operating systems and show a better standard conformity than PPC does.

Linux PDAs are able to combine the advantages of both worlds. A tux handheld benefits from the typical Linux factors– openness, standard conformance and a large applications and users base. If it can be made multicolored **and** productive by keeping it as easy to use and interoperable as possible there is a big chance that they not even make it into the hands of the Tux geeks but also into the hands of Joe User. And chances are very good that more and more industrial solutions can be based on Linux based PDAs, as Embedded Linux already has a good stand here. How Linux can support the new "Distributed Working" paradigm will be discussed at the end of this document.

Penguin construction kit

So what does it take to get a penguin running in your hands ?



First of all it needs a device, of course. Most handheld architectures are rather similar– a StrongARM 1100 CPU, flash memory and controllers, touch screen, sound device.

This architectural standardization also simplifies the porting of Linux to different devices.

However with new mobile CPUs at the front door this will change in the future.

The core of the embedded operating system is a standard kernel– preferably from the 2.4 tree. Current PDA kernels have caught up to the desktop: Familiar's 2.4.18 is the latest.

To adapt the kernel to the underlying architecture, some patches have to be applied to adjust the handling of storage, memory access, frame buffer etc.

If the hardware abstraction layer is adapted, most of the work is already done. Due to the modular concept of the Linux driver modules, most device drivers can be brought to operate on the new platform when cross compiled. However some might need modifications to integrate them into the interrupt and timing scheme and to adapt alignment issues due to the different CPU

architecture. Some drivers might not be able to work at all under the conditions offered by the architecture.

The positive aspect: If a Linux driver is available for a specific piece of hardware, chances are good that it will also run with your PDA— with WinCE and Palm vendor support is required.

/dev/handhelds/* – Distribution and Devices

Currently, the market for Tux-PDAs can be described as, well, limited. A few older devices are still around like the Agenda VR3 and the VTech Helio, however those have not been able to gain much market share. The new devices and distributions promise more to come so let us have a look at distributions and Linux versions. There are two approaches gone by the vendors.

First (and most) expensive is the development of an own distribution as described in the last section.

Second is licensing an existing platform from 3rd party vendors and third is using an existing open source distribution.

There are not many companies licensing their own distribution— Lineo with its Embeddix is the one most known, while Lisa is packaging the open source Familiar distribution.

Familiar is based on the work of the Compaq Cambridge Research Labs which are still sponsoring the project. It is therefore mainly targeted at Compaq iPAQ devices and fully supports the 31xx, 36xx and 37xx series with 38xx being under development. However there are projects underway to port Familiar to the HP Jornada series.

The Debian based distribution includes a boot loader, kernel, most common Linux utilities and a software installer which is network capable. Familiar completely replaces WinCE on the iPAQ, though it is even possible to dual boot both using a Compact Flash card. Familiar currently represents the most sophisticated Linux distribution available and appears as an excellent platform to work on.

For those who do not want to build up a cross

compiler themselves for application development, Familiar also offers an arm-gcc archive which is totally sufficient for most development tasks.

As older iPAQs are becoming cheaper, they appear to be the perfect platform for the Linux PDA adventure, however it should be noted, that the older ones are equipped with 16MB Flash RAM only, while newer devices are using 32 MB (**not** to be confused with Heap RAM which is always between 32 and 64 MB).

Besides the iPAQ other Linux based devices have shown up or have been announced, however the most promising seems to be Sharps Zaurus. Basically similar to the iPAQ, it has a hidden keyboard and is based on Embeddix while TrollTech's Qtopia delivers the graphical environment.

With the OpenZaurus project it is intended to replace Embeddix on the device with a free distribution.

As a last note and to round up the overview: The G.Mate Yopy and the Agenda VR3+ have been announced but have not been spotted yet.



/dev/fb0 – what is running inside the frame buffer ?

The headline says it all—a (mostly) non-keyboard device needs some form of user interaction. Naturally the pen comes into play and the only thing needed is a graphical user interface.

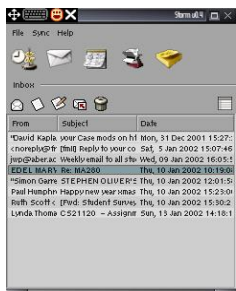
Currently, there are several approaches of which two will be presented. The basic display device is the frame buffer— the graphical user interface directly operates on the device.

As we are talking about Linux it is not surprising, that X based environments are available. Using X brings a pair of big advantages— the nerd-and-fun factor ("Hey, I got Mozilla running on my PDA") and most importantly the

availability of a large pool of free X based applications. As a result some rather interesting pieces of software have already been ported: Eg. the GTK libraries, the xine media player, Python or the X screen savers. However the limited display size of the smaller devices is a serious constraint and simply cross compiling the programs makes them unusable. Instead modifications to the user interfaces have to be made to adjust screen size and navigation. Usability is another factor– the X desktop is not an integrated one. Instead it is composed of several different applications running upon the the window manager thus bringing a different look and feel each. Interoperation of applications or an underlying communication architecture is missing. The fragmentation with differently behaving applica–tions and programs is of course not an issue which annoys enthusiasts and tekkies, however this will scare away the average user.



X remote Session running Gnome



Storm Mailer running with Python under X

Norway based TrollTech, famously known for their QT libraries, once started a project named QPE which now led into Qtopia: A graphical user interface for PDAs based on the QT/Embedded framework and designed for usability. Qtopia integrates all applications within a common platform on a messaging and document oriented framework. It comes closer to the multimedia aspect the Microsoft PPC platform offers as it has, besides the standard PIM applications, multimedia capabilities for playing mpeg movies or mp3 and wav sound files.

Though the source code has been released under the GPL very few restrictions apply– there is still some code owned by TrollTech on which some features depend– in short terms there are licensing issues. And there is little progress on the platform from a third party side– as the CVS

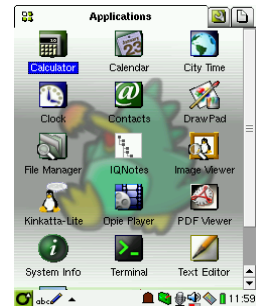
source archives are for obvious reasons strictly maintained by TrollTech employees. As a consequence contribution by free authors are rather limited– the source is GPLed but controlled by the vendor.

The OPIE project tries to resolve this conflict between commercial in–terests and free program–ming engagement. The OPIE platform is based on the GPLed Qtopia code base but is open for contribu–tions.

As a result many extensions and improvements have found their way into the software, most of them dealing with improving usability. The key point: By keeping binary compatibility between Qtopia and OPIE application interchange is possible. Software developed under one framework flawlessly runs under the other.



OPIE's Today application for advanced PPC like usability



OPIE screenshot–though Qtopia looks similar some minor differences exist at first glance

OPIE currently has more advanced features than Qtopia, however Qtopia seems to be more stable due to the strict quality assurance issues enforced by Troll–Tech's OEM licensing.

Distributed working– Linux support for the new working paradigm

Information management is vital for today's modern enterprises. No matter what sort of work has to be done information is needed to fulfill the tasks assigned.

However in lots of industrial and economic environments, people are not bound anymore to fixed desktop places. Managers are meeting their teams, workers are controlling machines and physicians are visiting patients. Providing important information in the shortest amount of time is a key factor in today's working environ–

ments, a factor perfectly suited for PDA usage. In most cases laptops are way too large and heavy to be carried around, especially in narrow areas. Think about an aircraft mechanic working in the inside of a plane, doing maintenance tasks.

So whenever there are ways to go while working or a desktop workstation is not available, the mobile information management shows its strengths– a scenario we call "distributed working".

Linux PDAs are perfectly supporting this paradigm and we will see why.

First there is the Tux bonus. Most embedded Linux software and their appropriate modules can be easily brought to the Linux PDA. Connecting the device to a production machine for maintenance purposes is as easy as using existing driver modules from embedded Linux environments.

Second is the availability and support of required hardware components – Wireless LAN and Bluetooth support can be used to communicate with common industry backend architectures, serial and IrDA connectors interface with standardized industry equipment.

The better: As current XML technology components are available for standard Linux systems those can also be used to easily integrate the PDA into the internet and intranets of the next generation. There is no need for a specialized Tux.NET– the technology is already present without the need for operating system upgrades as in WinCE.NET. This guarantees, that even today's devices can benefit from new visions.

To summarize this section: The common denominator "Linux" on workstations and PDA architectures does not only bring a common penguin feeling to the devices it also gives the ability to share reusable code and standards on all platforms thus supporting new and exciting applications in the mobile industrial and economical environment.

Tux+mobile+work= ? – Summarizing

Tux is ready for mobile operations support and to satisfy the requirements needed for the new mobile working environments.

Devices are available, distributions are stable and connectivity is present.

Pocket PCs and Palm devices are perfectly fitting the needs of different groups– the first is a power toy which also handles productive issues as a side effect, the second represents a small organizing device with productive organizing capabilities and fewer fun aspects.

Penguin PDAs are different– they fill the gap between both worlds. They represent the power toy for the Linux nerd as well as the productivity tool for daily use. Both categories can of course be met by PPC or PalmOS each as the Tux distributions are not perfect in neither of them. However there is more– there are the capabilities for deploying them as an industrial and economical information device supporting daily work operations. Especially if seen from an economical point of view– older iPAQ models for instance are much cheaper but still represent a perfect Linux device.

About the author:

Oliver Fels (oliver.fels@innotek.de) is a project manager and consultant for InnoTek Systemberatung GmbH in Eriskirch at Lake Constance.

The development of *mobile information management solution* is one of his working areas as well as *bridging the gap* between mobile devices and standard ecommerce architectures.

