AFULTab Contest

WeTab Freedom Report

24/04/2012

DUEKIN CONSULTING Gubener Straße 40 10243 Berlin Germany	**AFUL** 23 rue Greneta 75002 Paris France

Project: WeTab Freedom Report

by Pierre Pronchery

Copyright © 2012 **DUEKIN** ᴄᴏɴꜱᴜʟᴛɪɴɢ

Any error in this document is not intentional, please consult section 4, Errata for a list of known issues. More information on reporting issues, or obtaining an updated list of known errors can be found in the user manual in section 3.6, Finding and reporting issues.

Revision history

Version: 0.1	April 28, 2012	Reviewed by: PP
Initial version delivered		
Version: 0.2	April 30, 2012	Reviewed by: PP
Minor formatting fixes		
Version: 0.3	May 1, 2012	Reviewed by: PP
Minor wording fixes		

Table of Contents

List of tasks

Title	Reference	Status
Defining the objectives	2.4.1	Completed
Advocating the project	2.4.2	Completed
Becoming a NetBSD developer	2.4.3	Completed
Collecting the initial status	2.4.4	Completed
Unifying and debugging the existing interface	2.4.5	Ongoing
Implementing the touchscreen driver	2.4.6	Ongoing
Integrating the DeforaOS distributed framework	2.4.7	Cancelled
Preparing for the release	2.4.8	Completed

List of software installed

This list is not exhaustive for practical reasons.

Name	Purpose	Version	License
From the NetBSD Project, http://www.netbsd.org/			
NetBSD	Kernel, base system	6.0_BETA	BSD or FOSS
From pkgsrc (the NetBSD Packages Collection), http://pkgsrc.org/ and http://pkgsrc-wip.sourceforge.net/			
bash	Shell	4.2	GPL version 3
CUPS	Printing service	1.5.0	GPL version 2
dhcpcd-gtk	Networking	0.5.2	BSD
dia	Diagram editor	0.97.1	GPL version 2
Evince	PDF viewer	2.32.0	GPL version 2
GDM	Display manager	2.20.11	GPL version 2
GQview	Image viewer	2.0.4	GPL version 2
Liferea	RSS reader	1.4.28	GPL version 2
Matchbox-WM	Window Manager	1.1	GPL version 2
Mozilla Firefox	Web browser	8.0	MPL 1.1
Mozilla Thunderbird	Mail client	8.0	MPL 1.1
MPlayer	Media player	SVN-r32219-snapshot-4.5.3	GPL version 2
OpenOffice.org	Productivity suite	3.1.1	GPL version 3
Pidgin	Instant Messaging	2.10.0	GPL version 2
Sofia-SIP	VoIP library (SIP)	1.12.11	LGPL version 2.1
Transmission	Bittorrent client	2.42	GPL version 2, MIT
TSClient	Remote desktop	0.132	GPL version 2
xchat	IRC client	2.8.8	GPL version 2
XTerm	Terminal	259	MIT
From the DeforaOS Project, http://www.defora.org/			
Browser	File manager	0.4.5	GPL version 3
Editor	Text editor	0.2.0	GPL version 3
Keyboard	Virtual keyboard	0.1.2	GPL version 3
Locker	Screensaver	0.1.3	GPL version 3
Mailer	Mail client	0.1.4	GPL version 3
Mixer	Volume mixer	0.1.0	GPL version 3
Panel	Desktop panel	0.2.12	GPL version 3

Name	Purpose	Version	License
PDFViewer	PDF viewer	0.0.0	GPL version 3
Phone	Telephony suite	0.3.9	GPL version 3
Player	Media player	0.1.3	GPL version 3
Surfer	Web browser	0.2.5	GPL version 3
Todo	Task management	0.1.2	GPL version 3

List of known issues

This table summarizes the issues listed in section 4, Errata.

Platform	Component	Section	Title
WeTab	Telephony	4.1.1	The 3G modem is not supported
	Hardware support	4.1.2	Additional sensors are not supported
	Boot	4.1.3	The USB chipset may fail to initialize
IdeaPad S10-3t	Telephony	4.2.1	The 3G modem is not supported
	Hardware support	4.2.2	General system instability
DeforaOS	Desktop	4.3.1	Mailer/#27: SMTP support
	Desktop	4.3.2	Browser/#42: Copy & paste is not supported across browser windows
	Desktop	4.3.3	Browser/#47: Drag & Drop support is not optimal
	Desktop	4.3.4	Keyboard/#51: Support additional keyboard layouts
	Desktop	4.3.5	Keyboard: Modifier keys are not properly tracked
NetBSD	Hardware support	4.4.1	kern/45853: Support for touchscreens in xf86-input-mouse
	Hardware support	4.4.2	kern/45872: Issues and fixes for tpcalib (touchscreen calibration)
	Polishing	4.4.3	kern/46189: wsfb/genfb red adn blue colours are reversed on bit depths 15 16 24 and 32
	Hardware support	4.4.4	Intel GMA graphics cards are not fully supported
	Boot	4.4.5	port-amd64/45197: failing to load secondary boot
Other issues	Desktop	4.5.1	Finger-scrolling is not supported
	Desktop	4.5.2	Single-click activation is not supported
	Desktop	4.5.3	Third-button emulation is not supported
	Desktop	4.5.4	Additional languages are not supported

1. Project summary

1.1 Overview

The AFUL is a non-profit association based in France, whose aim is to promote general use of Free and Open Source Software (FOSS) as well as Open Standards. It has an international reach, particularly among French-speaking countries around the world. As part of this mission statement, the AFUL has organized an international competition called "AFULTab", calling for Open Source developers to compete and offer a fully functional Tablet device in a finite amount of time. It is intended for both professional and private users, by the exclusive means of software and applications under a free software license. This competition was announced in November 2011, with a deadline for the final submission on May 1st, 2012.

DUEKIN CONSULTING is the freelance activity of Pierre Pronchery, a Freelance IT-Security Consultant and Open Source developer based in Berlin, Germany. **DUEKIN CONSULTING** entered this competition, choosing the WeTab tablet from WeTab GmbH as target hardware. An additional tablet hardware was also used while implementing this project, the IdeaPad S10-3t from Lenovo. Besides fulfilling the requirements of the competition, by installing and documenting a FOSS Operating System on this hardware, the initial aim was to develop a proof-of-concept (PoC) deployment of the most advanced technologies part of the DeforaOS Project.

The aim of the DeforaOS project is to allow remote and secure access to user applications and data. This approach is based on a cross-platform distributed computing framework, which can installed and used on current Operating Systems. Even though existing applications can benefit directly from the framework, the project supplies its own desktop environment and user interface. It also supports embedded, touch-based platforms, with a finger-friendly version of the interface. In this competition, the tablet was developed on top of the NetBSD Operating System.

The NetBSD project is similar to a GNU/Linux distribution, however based on the historical 4.4BSD family of UNIX systems from the University of California in Berkeley, USA. Its license terms, also known as the BSD license, are compatible with those of the newer FOSS licenses such as the GPL, and therefore suitable for this project.

1.2 Involved parties

For AFUL:

> **Laurent Séguin**
> President
> e-mail: laurent.seguin@aful.org
> Web: http://aful.org/

For DUEKIN CONSULTING:

> **Pierre Pronchery**
> Freelance Open Source Developer
> e-mail: pierre.pronchery@duekin.com
> Mobile: +49 (0)177 472 7481
> Web: http://www.duekin.com/

1.3 Project outcome

Most importantly, the deployment of a functional end-user environment on the tablet, based on the DeforaOS User Interface and NetBSD Operating System was successful. A number of achievements were completed during the course of this project, and believed to be beneficial to both the NetBSD and DeforaOS projects. A number of new components, fixes, or general improvements as either software or documentation were submitted to both projects, with most already integrated while writing this report.

It was also a very positive experience on a personal level, with the developer of this project, Pierre Pronchery, being invited and accepted as an official NetBSD developer before the final submission. This was both the result and additional motivation for contributions and learning more about end-user and system development in Open Source Operating Systems.

Unfortunately, some parts of the initial proposal could not be fully completed, are known to have problems, or are still completely missing from the current deployment. They are listed in section 4, Errata in this document.

1.3.1 Contributions to the DeforaOS Project

The DeforaOS has already benefitted to a major extent from this project: the desktop environment could be further tested, fixed and improved, with a number of new software releases hitting the website. It has certainly also gained in visibility and definitely in its documentation, which should be merged online with the contents of this document.

A complete rewrite of the website was also initiated during this project, although not detailed in this report; while not being ready at this time, it should also help the project be more attractive.

(The DeforaOS logo is under Copyright to The DeforaOS Project)

1.3.2 Contributions to the NetBSD Project

Likewise, it is also believed that the NetBSD project has gained from this initiative. While not being easy to measure, it has apparently welcome this effort by inviting and accepting its developer into the NetBSD Foundation, the parent organization to the project. This certainly means that the contributions were valuable, and a trust relation built while submitting issues and patches.

(The NetBSD logo is under Copyright to The NetBSD Foundation)

1.4 Perspectives

While still far from being perfect, the environment is already fully functional. Once the remaining hardware issues solved, it can be further leveraged for everyday use in terms of usability and power management in particular.

More interestingly, it proved an excellent and practical test platform for the DeforaOS desktop environment. As such, it will be very helpful when actually implementing the distributed framework as part of the user interface.

2. Project execution

2.1 Technical orientation

The hardware platform chosen for this project is the WeTab from WeTab GmbH. Although based on commodity hardware (Intel-compatible) and not really proprietary, this platform is not fully standard (dedicated firmware). The target Operating System for the project was NetBSD, which was not known to have support for tablet devices thus far (with the exception of the Sharp Zaurus). The user interface chosen was based on a smaller project, DeforaOS, also mainly developed by the author of this project and report. The target audience was intended to be for both casual and business use, through the integration of DeforaOS' advanced distributed framework within the user interface.

The WeTab tablet (from "Serv" via Wikipedia, http://en.wikipedia.org/wiki/File:WeTab.png)

Therefore, the major achievement in this project was expected to be on the software side, both for the NetBSD Operating System (hardware support, desktop integration) and for the DeforaOS desktop environment (tablet interface, polishing, extending).

2.2 Foreseeable challenges

The first uncertainty was to be able to install and boot NetBSD on the target hardware. Although based on a common Intel architecture and chipset (found in many netbooks), support for this chipset was already known to be incomplete within NetBSD (for the graphics subsystem in particular). Additionally, the target hardware was known to have a specific firmware when booting, which could mean modifications to the NetBSD bootloader.

Then, hardware parts specific to tablets were not believed to be well supported so far within NetBSD, of which the touchscreen in particular. A Google Summer of Code (GSoC) project called "Expansion for wstablet in NetBSD " was completed in 2008, but without particular support for user interaction on tablets included so far (see http://netbsd-soc.sourceforge.net/projects/wscons/).

Finally, the third challenge was to adapt and raise the quality of the existing DeforaOS applications to a fully functional environment, when installed and configured to run on a tablet device. Hopefully, the additional innovative features planned could be developed and included as well, but only as far as the timeframe would allow it.

2.3 Development environment

Hardware components

Most of the development effort was conducted on two systems alternatively, a Sun Ultra 24 workstation and a Lenovo ThinkPad T60 (2007) laptop respectively. They were both running NetBSD for Intel-compatible 64-bits computers (amd64), and upgraded to the 6.0_BETA version as soon as it was available.

While waiting for the final tablet hardware to be available and delivered, tests were conducted on a Lenovo IdeaPad S10-3t netbook. It offered very similar hardware to the WeTab, including a tablet mode (rotating screen and touchscreen). More details about this tablet hardware can be found in 3.2.2, Lenovo IdeaPad S10-3t.

Testing on the final tablet hardware, the WeTab, could only begin in early March 2012. More details about this tablet can be found in 3.2.1, WeTab GmbH Wetab.

Software platforms used

As mentioned above, NetBSD was used for development, along with packages from pkgsrc's pkgsrc-2011Q4 release (custom built). The desktop environment used was from DeforaOS, like that used on the tablet (except not in embedded mode). All of the tools required were part of either of these two platforms, and distributed under a FOSS-compliant software license.

Specific tools

A few tools were useful in particular, of which:

- GCC, the GNU Compiler Collection, http://gcc.gnu.org/
- DeforaOS configure, a Makefile generator, http://www.defora.org/os/project/16/configure
- make, as provided byNetBSD, http://www.netbsd.org/
- vim, a vi clone improved, http://www.vim.org/
- CVS, Concurrent Versions System, http://savannah.nongnu.org/projects/cvs
- Xephyr, a KDrive-based X server, http://www.freedesktop.org/wiki/Software/Xephyr
- Apache OpenOffice, the Free and Open productivity suite, http://www.openoffice.org/

Additional resources

DeforaOS is hosted on both personal infrastructure and a virtualized hosting server, provided by **DUEKIN** CONSULTING.

2.4 Tasks performed

2.4.1 Defining the objectives

Description

Before anything, a list of functionalities essential to regular operation of the device was compiled.

The list was defined as follows, in decreasing priority:

1. File management (drag & drop, MIME interaction)

2. Web surfing (tab browsing, finger-based interaction...)

3. Proper handling of e-mail (IMAP, SSL...)

4. Media player (codec support...)

5. Rest of the desktop environment (panel, screensaver, office applications...)

6. Bootup time

7. Documentation covering all essential aspects (installation, user manual)

8. Connectivity to wireless networks

9. Hardware support in general

10. Integration of the DeforaOS experimental distributed framework

Challenges

This list is long and ambitious for a single developer in spare time. In practice, the development effort was focused on whichever area felt the most lacking at a given time.

Outcome

The tablet was delivered in a functional state, although most areas could not be completed as good as desired. Integration of the DeforaOS distributed framework in particular could not be performed at all.

2.4.2 Advocating the project

Description

Communicating about the project was done regularly. The primary reason was to look for direct support (testers, developers), although raising awareness about the technologies used is also welcome and important.

Means used for this task were social networking websites like Tweeter (through the @DeforaOS account, with hashes #AFULTab, #DeforaOS and #NetBSD), or freecode for software releases in particular (formerly known as freshmeat). Both DeforaOS and NetBSD projects can also be found on Ohloh (http://www.ohloh.net/) and CIA (http://cia.vc/), two popular software repository directories.

Additionally, a talk was held at the FOSDEM Open Source conference in Brussels, on February 5th. It took place at the BSD Devroom.

Challenges

NetBSD is not the most popular Open Source Operating System around, and not the most popular BSD-based one either. As such, it does not have as much visibility as GNU/Linux for instance. Likewise, DeforaOS is the effort of a single developer for the most part, and its current popularity reflects this status. There was little

feedback so far on the ideas and concepts proposed as part of the project.

Outcome

Reception of the talk given at FOSDEM was very positive. It eventually created a new but welcome task, being invited and then joining the NetBSD project as an official developer. This task is detailed below as 2.4.3, Becoming a NetBSD developer.

2.4.3 Becoming a NetBSD developer

Description

This task was not planned initially, but was a most welcome consequence of the project advocacy from task 2.4.2, Advocating the project. It allowed contributions to the NetBSD Project to be integrated faster, thanks to a direct access to the source tree, to the issue tracker as well as to internal discussions about specific issues.

Moreover, prior submission of a number of issues (often including fixes) as well as regular presence at conferences and development events involving members of the NetBSD Foundation before this project certainly also helped for this task to happen in the first place.

Challenges

Although not as thorough and heavy process as some other popular Open Source Operating System projects, the integration process does not happen overnight, and requires meeting at least one active NetBSD developer in person. This was difficult given the traveling obligations of Pierre Pronchery in the second half of the project, but could nonetheless be planned a few weeks after submitting the application form.

Another minor glitch was with the request for the last administrative details, which e-mail was delayed by another couple weeks before reaching its destination.

Outcome

As of April 16th 2012, Pierre Pronchery was officially accepted as a NetBSD developer. He could append his name to the source tree and main website himself the following day. He could also submit his first contributions as a developer after a week, with more awaiting integration. Most should be available in the upcoming release, NetBSD 6.0.

2.4.4 Collecting the initial status

Description

Once the objectives defined, a test system with the current state of the kernel, base system and desktop applications was installed and configured. This was a more practical approach to the project planning and management, and very likely to be more realistic and accurate for the development of a usable platform.

At this point, an initial list of known issues could be compiled, published and worked incrementally upon. This was to be done according to their respective priorities.

Challenges

At this point, the final hardware was not available for tests. It was therefore chosen to use another tablet hardware, but more suitable for development. The Lenovo IdeaPad S10-3t was chosen for its apparent similarity with the WeTab. This system is therefore also described in this document, as it had an important role in the development process.

As was quite expected, the list of places to improve, work on or even fully implement was certainly longer than time would allow.

Outcome

The initial priorities had to be re-evaluated, with a list of the most important ones maintained online at http://www.defora.org/os/wiki/3622/AFULTab-contest. Where applicable, bug reports were issued upstream, and their status also tracked on this page.

The most important change in priorities was with hardware support, which became the highest priority: support for the touchscreen was required before anything else. Then, the main focus was on bugfixing and improving the DeforaOS desktop, while advocating the project as opportunities occurred.

2.4.5 Unifying and debugging the existing interface

Description

The development sources of the DeforaOS desktop were generally more advanced than the code released thus far. They included a number of new features, often required or desirable for proper tablet-based interaction, but unfortunately incomplete or not fully functional in some cases. These changes required further testing, completion and proper releases and packaging.

Challenges

There was no particular challenge associated with this task, as this operation benefitted directly from the experience gathered when working on the DeforaOS smartphone interface for the hackable:1 project (see http://trac.hackable1.org/trac/wiki/DeforaOSSmartphone). It was actually easier, as testing could occur within a native environment, without required cross-compilation or hardware emulation. Xephyr was particularly useful while performing this task.

Outcome

Most projects part of the DeforaOS desktop environment were released at least once while performing this task.

2.4.6 Implementing the touchscreen driver

Description

Two existing drivers for touchscreen were found to be available within the NetBSD project: uep(4) and zts(4). Gathering information about the touchscreen panel found on the Lenovo IdeaPad S10-3t, it was determined that it is a USB HID device, not unlike most modern multi-touch panels. Very useful information could be obtained

from Microsoft, as linked here: http://msdn.microsoft.com/en-us/windows/hardware/gg487437 .

Although useful as a reference implementation, neither of the existing touchscreen drivers in NetBSD supported this kind of touchscreen. However, the ums(4) driver (for USB mice) was very useful when interacting with the USB HID bus on NetBSD.

Challenges

Writing a kernel driver is rarely a trivial task. It can be a time-consuming and frustrating experience, as proper testing often requires complete reboots with little more information than "it does not work". In this case, it also required transferring the test kernel over a USB memory stick, or alternatively setting up a remote boot infrastructure.

The availability of a hardware platform supporting the same driver for the touchscreen chipset was of great help (the Lenovo IdeaPad S10-3t). With its built-in keyboard and touchpad, it was much easier to successfully debug and integrate the driver within the existing infrastructure.

Getting the driver to work on the Lenovo netbook was not sufficient though. In practice, the chipset found in the WeTab required additional parameters in order to work correctly. Because of their particular form-factor, Microsoft recommended touchscreen chipset makers to pre-configure their firmwares in "mouse mode" for tablets, where they would allow Operating Systems to function without dedicated support for instance. The resulting behavior was wrongly interpreted as incorrect screen calibration, whereas the real solution was to force the chipset to "multi-input" mode.

Outcome

The uts(4) driver was submitted for inclusion on January 17th, and imported to the NetBSD tree shortly after. It is already part of the initial NetBSD 6.0_BETA release. The steps providing additional support for Xorg were submitted as separate Problem Reports, and available in the base system as well as within pkgsrc.

Its addition seems to have been well received, and has certainly helped the process of becoming an official developer.

2.4.7 Integrating the DeforaOS distributed framework

Description

The DeforaOS Project is not only about providing a desktop environment, but primarily about enabling applications to access user data remotely yet securely. This is even possible when the data is only found on a workstation behind a firewall, and accessed from a tablet computer on a separate network. A working prototype is already available, where a video file can be played remotely and efficiently over a TCP connection (optionally secured with SSL), without requiring any change to the media player.

Challenges

The current codebase already required a major rewrite when beginning the project. This is for a number of reasons:

- Proof-of-Concept (PoC) code only used for prototypes thus far

- as a result, hard-coded implementation of a single underlying protocol

- most importantly, no easy possibility of integration with the event-driven programming library underlying Gtk+ (the graphical toolkit used in the DeforaOS desktop environment)

Unfortunately these issues could not be tackled early enough to be integrated into the project.

Outcome

This task had to be cancelled.

2.4.8 Preparing for the release

Description

The deadline for this project was on May 1st 2012. This applied for the hardware and the documentation accompanying it, which both had to be delivered in Paris before this date.

Challenges

Polishing is a neverending task, where invariably new bugs or problems are discovered. A tradeoff had to be made between a satisfying, functional result and delivering in time. This was considered to be fine, as long as the remaining issues were documented accordingly.

Outcome

Software releases were issued for the relevant projects, with their respective packages updated in the pkgsrc-wip repository.

The elaboration of this report was another direct outcome of this task, with the tablet scheduled to be delivered on time on April 30th.

3. Platform documentation

3.1 How to read this documentation

3.1.1 Disclaimer

This section was originally published as part of the final submission for the AFULTab project, with the intention to be redistributable as a separate document. It summarizes the steps required for the preparation, installation, configuration and final use of a tablet device such as the WeTab, together with a user environment based on the NetBSD Operating System and DeforaOS desktop environment. The reader of these instructions is assumed to have a working knowledge of UNIX systems, and to be familiar with the manual installation procedures for such systems.

The instructions and commands mentioned in this documentation are given to the best of the writers' knowledge, under time constraints. They should be carefully reviewed before used, and adapted as necessary. They may be subject to changes, as either of the underlying platforms may be evolving as well. Therefore, no damage may be claimed as a result of applying these instructions.

3.1.2 Conventions used

The commands listed in this document are typically preceded by the "#" or the "$" sign, representing the shell prompt for an administrator and a regular user respectively. They should therefore be entered with the relevant privileges, which availability should be ensured beforehand. For clarity, a standard procedure has been included in some cases; it may be replaced with a different procedure if so desired or necessary.

Commands, system components or drivers for which a manual page is available on a NetBSD system are followed with the corresponding manual section enclosed within parenthesis. For instance, "ls(1)" means that the command "ls" is documented by a manual page, in section 1. This page can be obtained as follows:

```
$ man 1 ls
```

In this example, it is therefore meant to enter "man 1 ls" directly within a shell, and that this command should work as an unprivileged user.

3.2 Target hardware

3.2.1 WeTab GmbH Wetab

Hardware specifications

The WeTab is an Intel-based 64-bits tablet computer, using an Atom N450 processor clocked at 1.66 GHz. The processor is completed by 1 GB of RAM, along with 32 GB flash memory (for local storage). This capacity can be extended through the SD card slot (up to another 32 GB memory with SDHC cards). The screen is a 11.6" widescreen (16:9) with a resolution of 1366x768 pixels, and supports multi-touch input. Battery life is estimated at 6 hours (according to the vendor). Besides the headphones output, the tablet offers 2 standard USB 2.0 ports, and a mini-HDMI video output. Finally, wireless connectivity includes 802.11bgn, Bluetooth 2.1 and 3G as

well.

The complete specifications can be found in 5.2, Hardware specifications for the WeTab GmbH WeTab.

Software support

The WeTab is delivered pre-installed with a custom Linux distribution called WeTab OS, based on the former Meego platform. Replacing this system with a fully FOSS software platform is detailed in 3.3, Installation procedure.

Please refer to the Errata section 4.1, Known issues on the WeTab GmbH WeTab for a list of known issues.

3.2.2 Lenovo IdeaPad S10-3t

Hardware specifications

The IdeaPad S10-3t from Lenovo is a netbook, equipped with a rotating screen with multi-touch support. Like the WeTab, it is based around an Atom N450 CPU clocked at 1.66 GHz, but more extensible with up to 2 GB of RAM. The screen size differs from the WeTab, with a 10" screen offering a resolution of 1024x600 pixels, also supporting multi-touch input. The rest of the specifications can be compared with the WeTab, including 802.11bgn, Bluetooth and 3G networking, MicroSD slot, USB and VGA ports, although with a hard disk drive (up to 320 GB) and a full laptop keyboard of course.

The complete specifications can be found in 5.1, Hardware specifications for the Lenovo IdeaPad S10-3t.

Software support

The IdeaPad is typically delivered with a Windows 7 system. Just like any regular Personal Computer, it is easily replaced with a FOSS Operating System. The installation procedure of the WeTab can however be followed, from which a few steps can simply be omitted.

A list of known issues on this system was elaborated as well, and is found in 4.2, Known issues on the Lenovo IdeaPad S10-3t in this report.

3.3 Installation procedure

3.3.1 Requirements

This guide assumes the availability of a 6.0_BETA NetBSD system (or newer 6.0 release). An Intel-compatible 64-bits platform is recommended and referred to here; this port of NetBSD is called "amd64". Documentation to obtain, install and configure this system can be found in the NetBSD Guide, which is available online at http://netbsd.org/docs/guide/en/. Be sure to install the "comp" set A complete repartitioning and formatting of the main hard disk drive may be required, therefore its initial contents should be backupped before anything, as they may be lost or damaged during this operation.
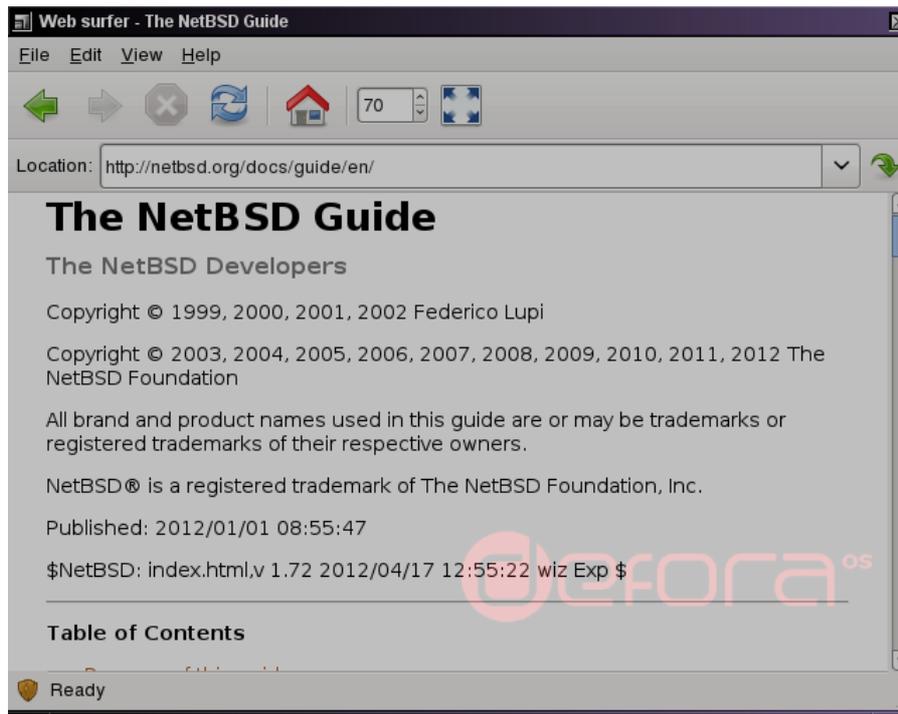
The NetBSD Guide as found online

Once a NetBSD system installed, it is also recommended to have at least 20 GB of free storage space before beginning this procedure. A single USB memory stick will be used to transfer and install the new system on the tablet, which should however offer at least 2 GB of storage. Likewise, a complete repartitioning and formatting of the USB stick may be required; make sure to backup its contents if necessary before proceeding.

Additionally, a USB keyboard will be required while installing and configuring the tablet, before it can be fully used with finger-based interaction.

3.3.2 Obtaining the sources

Although also likely with the binary sets and packages available online, the installation will be performed from the source files. They can be obtained as follows, assuming the regular user is in the "wsrc" group:

```
$ su -
Password:
# mkdir -p -m 775 /usr/src /usr/pkgsrc
# chgrp wsrc /usr/src /usr/pkgsrc
# logout
$ cd /usr
$ cvs -d:ext:anoncvs@anoncvs.netbsd.org:/cvsroot co -r netbsd-6 src
[...]
$ cvs -d:ext:anoncvs@anoncvs.netbsd.org:/cvsroot co -r pkgsrc-2011Q4 pkgsrc
[...]
$ cd /usr/pkgsrc
$ touch ~/.cvspass
$ cvs -d:pserver:anonymous@pkgsrc-wip.cvs.sourceforge.net:/cvsroot/pkgsrc-wip co
wip
[...]
```

At this point, the sources for the entire NetBSD base system (6.0 branch), the pkgsrc Packages Collection

(pkgsrc-2011Q4 release) and its staging repository should be present on the local system.

3.3.3 Building the base system

First, the system and installation kernels and sets have to be compiled. It can be done as follows:

```
$ su -
Password:
# mkdir -p -m 775 /usr/obj /usr/tools
# chgrp wsrc /usr/obj /usr/tools
# logout
$ cd /usr/src
$ ./build.sh -O /usr/obj -T /usr/tools releasekernel=GENERIC \
 releasekernel=INSTALL distribution sets
[...]
```

The resulting binary kernels and sets should then be available in the `/usr/obj/releasedir/amd64/binary/kernel` and `/usr/obj/releasedir/amd64/binary/sets` folders, respectively.

At this point though, the `netbsd-INSTALL.gz` kernel cannot be used to install the system, as it does not contain the installation ramdisk; to create it, proceed as follows:

```
# cd /usr/src/distrib/amd64/instkernel
# /usr/tools/bin/nbmake-amd64
```

The resulting kernel should then embed the ramdisk, and be found in `/usr/obj/distrib/amd64/instkernel/netbsd-INSTALL.gz`.

3.3.4 Building packages

A number of different tools are available to build and install packages, such as pkgin(1), which resembles Debian's apt subsystem. However, here, the use of pkg_chk(8) was prefered, as it allows compiling and deploying the packages required directly. This allowed more control over the compilation options and eventual patches required. The `/etc/mk.conf` file contains these options, of which a copy can be found in 5.3.1, /etc/mk.conf. As for pkg_chk(8), it uses the file `/usr/pkgsrc/pkgchk.conf` by default, of which a copy is also found in 5.3.2, /usr/pkgsrc/pkgchk.conf.

This assumes that no package is currently installed on the system; to reset the system to this state, simply enter the following command as the root user:

```
# pkg_delete '*'
```

First, install the pkg_chk package, as follows:

```
# cd /usr/pkgsrc/pkgtools/pkg_chk
# make install clean
```

With this done, build and create all the packages required through pkg_chk(8) this way:

```
# pkg_chk -askD wetab
```

The resulting packages should be available in the `/usr/pkgsrc/packages` folder.

3.3.5 Preparing the USB memory stick for installation

For simplicity, the filesystem used on the USB stick will be a 32-bits MS-DOS FAT. The first thing to identify, is how the stick is partitioned and detected by the system. One way to identify it is to read the output of the dmesg(8) command after inserting the stick; the corresponding device node is usually `sd0e`, which will be assumed here. Be however very careful to be using the proper device.

Re-partitioning can be done with the fdisk(8) command, which features an interactive mode. Proper detection by the system can be verified (or updated) with the disklabel(8) command. A FAT filesystem can then be created on a given partition as follows:

```
# newfs_msdos -F 32 sd0e
```

Mounting and accessing this filesystem can then be performed as described here:

```
# mkdir -p /mnt/usb
# mount /dev/sd0e /mnt/usb
```

To prepare the stick for installation, while it is mounted:

```
# cp /usr/mdec/boot /mnt/usb/boot
# installboot /dev/rsd0e /usr/mdec/bootxx_msdos
# cat > /mnt/usb/boot.cfg << EOF
banner=Welcome to the DeforaOS installer
banner==================================
banner=
banner=Please choose an option from the following menu:
menu=NetBSD/amd64 install:boot release/amd64/binary/kernel/netbsd-INSTALL.gz
menu=Drop to boot prompt:prompt
default=1
timeout=10
EOF
# mkdir -p /mnt/usb/release
# cp /usr/obj/distrib/amd64/instkernel/netbsd-INSTALL.gz /mnt/usb/release
# cp -R /usr/obj/releasedir/amd64 /mnt/usb/release
# mkdir -p /mnt/usb/packages/amd64
# cp -R /usr/pkgsrc/packages/amd64/All /mnt/usb/packages/amd64
# cp /usr/pkgsrc/pkgchk.conf /mnt/usb/packages
```

Before the last step, be sure to unmount the stick before removal:

```
# umount /mnt/usb
```

This last operation is required by the firmware on the WeTab, to actually recognize the USB stick as being an authorized boot device:

```
# printf '\x9d\x2a\x44\x7b' | dd of=/dev/sd0d oseek=440 bs=1 count=4
```

3.3.6 Installing the new system on the WeTab

Before trying to boot the USB stick on the WeTab, it is recommended that its battery is fully charged, and the tablet connected to the power supply. With the device turned off, insert the USB stick into either of the USB ports, and press the power button briefly. Then, as soon as the blue LED turns on, press again briefly on the power button while holding the tablet over the "soft button". The "soft button" is a sensor located at the top left of the tablet, left from the LED indicators. The system will then boot the installer after a few seconds.

At this point, the generic installation instructions can be followed:

- it is easier to use the whole disk, and leave the disk geometry as detected;

- it is recommended to create a single "a" partition for the root filesystem (entering "+" for its size), with 1 GB of flash and a 256 MB memory filesystem "mfs" for /tmp;

- a "Custom" installation should be chosen, and the "kern-GENERIC.tgz" set disabled from the list;

- the sets are to be found from the USB stick, which is an "Unmounted filesystem", on the "sd0e" device, with an "msdos" filesystem (the other fields can be left unchanged)

At the end of the installation process, it is likely that the sanity check for the new installation will fail: it is missing a kernel to boot from. It has be installed from the utility menu, from the command line interface:

```
# mkdir -p /target /usb
# mount /dev/sd0e /usb
# mount /dev/wd0a /target
# cp /usb/release/amd64/binary/kernel/netbsd-GENERIC.gz /usb/netbsd
# umount /target /usb
```

At this point, the tablet should be in a bootable state. It can be rebooted, and will be easier to configure after the first boot.

3.4 Initial configuration

3.4.1 Creating the default user

As recommended by the system after the first login as the "root" superuser, a regular user account has to be created. It is recommended to create it this way, assuming "afultab" is the username chosen:

```
# mkdir -p /home/afultab
# groupadd -g 1000 afultab
# useradd -g afultab -u 1000 -G wheel,operator -d /home/afultab \
  -s /bin/ksh afultab
# chmod 700 /home/afultab
# chown afultab:afultab /home/afultab
```

The default shell for the user could be later changed for bash, which some might consider more practical. Once its package installed, the following command would select it by default:

```
# usermod -s /usr/pkg/bin/bash afultab
```

3.4.2 Installing the packages required

For this step the USB memory stick used during the installation procedure has to be mounted again. A default mountpoint for the memory stick can be created as follows:

```
# mkdir -p /mnt/usb
# cat >> /etc/fstab << EOF
/dev/sd0e /mnt/usb msdos rw,noauto,nosync,-u=1000,-g=1000,-M=0700,-m=0600
EOF
```

The packages required can then be installed automatically as follows:

```
# mount /mnt/usb
# cd /mnt/usb/packages
# pkg_add pkg_chk-2.0nb1.tgz
```

```
# pkg_chk -abP "$PWD" -D wetab
[...]
# cp /usr/pkg/share/examples/rc.d/* /etc/rc.d/
```

At this point all of the packages required should be installed. Their initialization scripts should also be available to the base system, and will require proper configuration for the startup sequence.

3.4.3 Configuring the startup sequence

On NetBSD systems a single file controls most of the bootup sequence: /etc/rc.conf. This file should typically be extended with the following contents:

```
hostname=wetab.afultab
dhcpcd=YES
mixerctl=YES mixerctl_mixers="mixer0"
wpa_supplicant=YES wpa_supplicant_flags="-B -iath0 -c/etc/wpa_supplicant.conf"

avahidaemon=NO
cupsd=NO
dbus=NO
estd=YES
famd=NO
hal=NO
gdm=YES
slpd=NO
```

Of course the hostname can be set to another value as desired.

Then, it may make sense to configure the X and gdm display servers already. It can be performed as follows:

```
# cd /root
# X -configure
# mkdir -p /usr/pkg/etc/X11
# mv /root/xorg.conf.new /usr/pkg/etc/X11/xorg.conf
# /etc/rc.d/gdm start
```

Note that it may be required to set the default depth for Xorg to 24, and to force the mouse device to "/dev/wsmouse2" instead of the default "/dev/wsmouse".

At this point it is recommended to enable automatic login for the "afultab" user via the configuration interface from gdm. The keyboard sequence Ctrl+Alt+F1 to F5 can be used to switch back to virtual consoles; the fifth is where X should be found by default.

The user session has to be defined next; it can be done as the "afultab" user:

```
# su - afultab
$ cat > .Xclients << EOF
#!/bin/sh

/usr/local/sbin/wscalib
desktop &
dhcpcd-gtk &
locker &
panel &
phone &
matchbox-window-manager -use_titlebar no -use_cursor no -use_desktop_mode plain
EOF
$ chmod 700 .Xclients
```

With this done, the tablet should already boot to a usable interface.

Finally, the bootloader can be re-configured to display a splash picture while booting. It is recommended to

enable this feature only once the system is known to boot properly. A splash picture also has to be copied on the filesystem; JPEG files are supported, although without progressive encoding. Assuming the `/deforaos.jpeg` file is to be used as the splash screen, the contents of the `/boot.cfg` file can be modified as follows:

```
banner=Welcome to NetBSD
banner=================
banner=
banner=Please choose an option from the following menu:
menu=Boot normally:vesa 800x600x16;splash /deforaos.jpeg;boot /netbsd.gz -z
menu=Boot normally (text mode):boot /netbsd.gz
menu=Boot single user:boot /netbsd.gz -s
menu=Disable ACPI:boot /netbsd.gz -2
menu=Disable ACPI and SMP:boot /netbsd.gz -12
menu=Drop to boot prompt:prompt
default=1
timeout=5
```

Also ensure that the `/netbsd.gz` kernel is installed as expected. As keyboards are not supported during the boot sequence, this menu is hardly required for proper operation, and the timeout can even be reduced to 0 for instant boot.

3.4.4 Configuring the touchscreen

The screen is to be calibrated at every startup by the end-user:

```
# mkdir -p /usr/local/sbin
# cat > /usr/local/sbin/wscalib << EOF
#!/bin/sh

/sbin/wsconsctl -f /dev/wsmouse2 -w
calibration.samples=0,0,0,0:32767,32767,1023,767:32,0,1,0:0,43,0,1
EOF
# chmod 770 /dev/wsmouse2
```

While its driver is incomplete, the touchscreen also has to be switched to multi-input mode at every startup:

```
# cat >> /etc/rc.local << EOF
/usr/bin/usbhidctl -f /dev/uhid1 -w 0x000e.Finger.Input_Mode=2
EOF
```

This latter command may require an updated version of the `/usr/share/misc/usb_hid_usages` file, with the "Input Mode" HID usage defined accordingly within the "Digitizer" section.

3.4.5 Extra configuration

Some additional steps for proper integration are included here. They are by no means complete, and improvements are always likely, although often subjective at some point.

For integration between the WPA sub-system and the desktop:

```
# cat > /etc/wpa_supplicant.conf << EOF
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=wheel
update_config=1
EOF
```

For system-wide telephony support:

```
# chmod 770 /dev/ttyU0
```

```
# mkdir -p /etc/ppp/peers
# touch /etc/ppp/options
# cat > /etc/ppp/ip-up << EOF
#!/bin/sh

cp /etc/ppp/resolv.conf /etc/resolv.conf
EOF
# chmod 700 /etc/ppp/ip-up
# cat > /etc/ppp/peers/phone << EOF
notty
defaultroute
local
noauth
usepeerdns
mru 296
mtu 296
EOF
```

For the desktop environment, a number of sample files to be placed in the user's home directory have been included in the Appendix of this document:

- ~/.browser

- ~/.desktoprc

- ~/.gtkrc-2.0

- ~/.locker

- ~/.mime

- ~/.panel

- ~/.phone

Most of the desktop applications can otherwise be configured via their respective user interfaces.

3.5 User manual

3.5.1 User interface

The user interface is organized as follows by default:
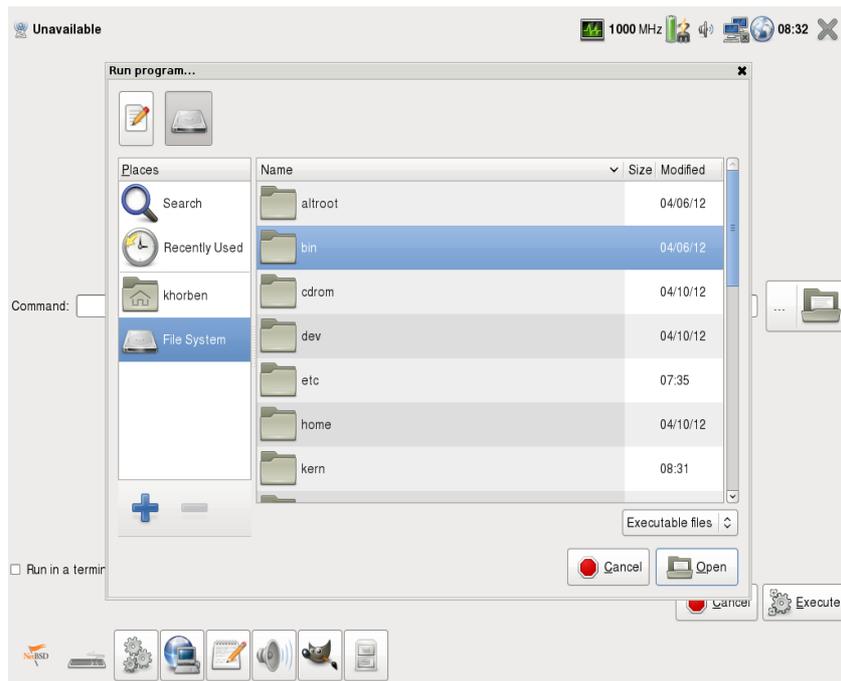
1. The top left indicator tracks the current mobile operator

2. The top right indicator track the CPU frequency, battery and volume information, followed by the notification area (systray), clock and a close button

3. The middle of the screen is the main area, where the current application is running (in fullscreen mode by default)

4. The bottom left button is the main menu, followed by the virtual keyboard switch and the task list (sorted by most recently used)

Here is an example of text input, from within the text editor, with the virtual keyboard activated:
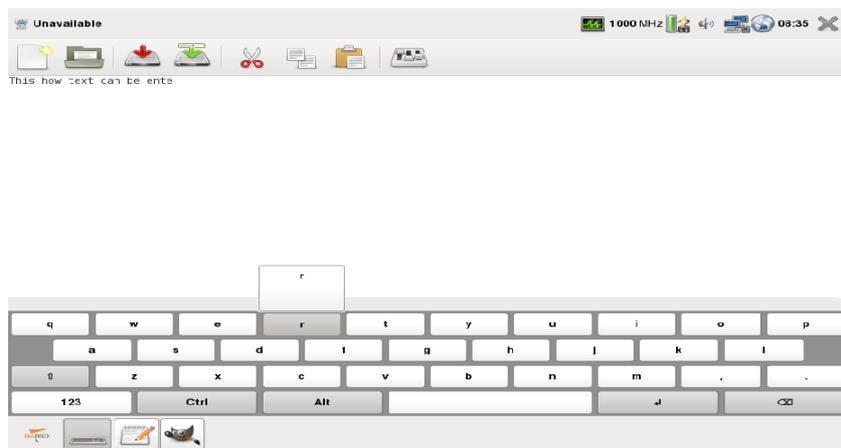
3.5.2 Connecting to a Wireless network

Connecting to a wireless network can be performed via the notification area, with a single click on the network icon; the different networks available are then listed, with a prompt for the WPA passphrase if required:
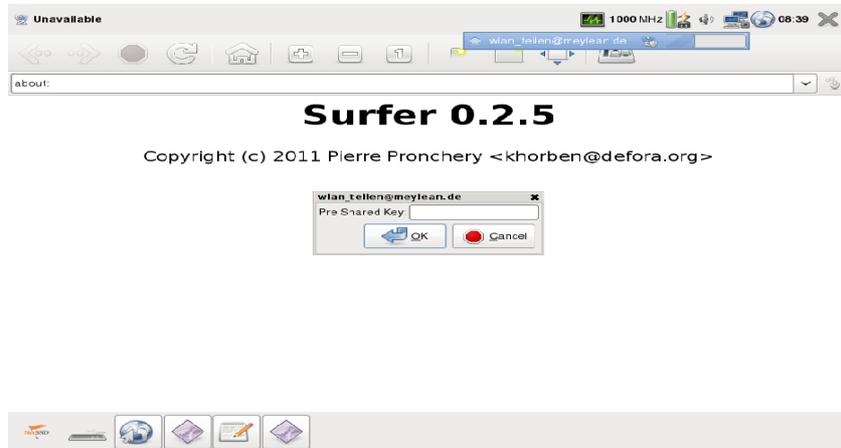
3.5.3 Connecting to the 3G network

Connecting to the Internet over the mobile network is performed similarly, using the globe icon instead. The connection window contains two tabs (settings and connection status). A sample configuration screen is as follows:
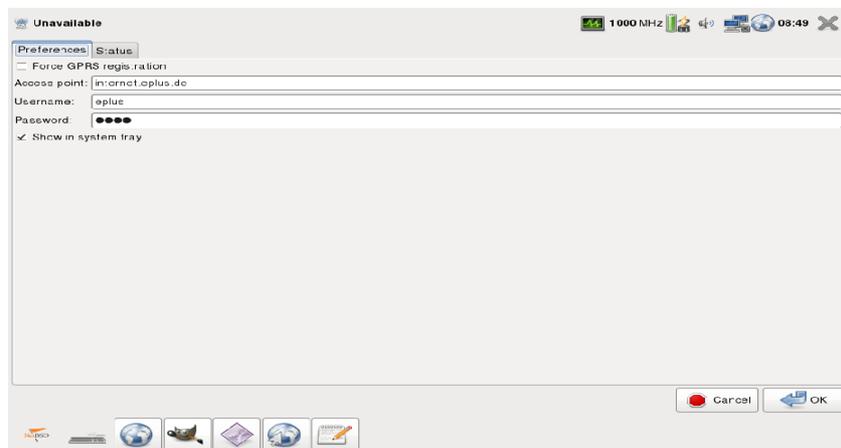
The status screen can then be used to connect and disconnect to and from the network, as well as to monitor connectivity.

3.6 Finding and reporting issues

3.6.1 How to report issues

The first thing to do when finding an issue, is to try to identify in which component it belongs:

- the DeforaOS desktop environment, or any end-user applications part of it

- the NetBSD Operating System, including the kernel or the base system

- a third-party application, provided by a package from the pkgsrc Packages Collection

The latter two are handled together by the NetBSD Project, for which instructions are found below in 3.6.3, For the NetBSD Project. If in doubt, the issue tracking system from the DeforaOS Project can be used, as detailed in 3.6.2, For the DeforaOS Project.

A list of known issues at the time of this report can be found in section 4, Errata, and is also summarized in a table at the beginning of this document. For practical reasons, this list is not exhaustive and only contains the most significant issues.

3.6.2 For the DeforaOS Project

The issue tracker for the DeforaOS Project can be found online at the following address: https://www.defora.org/os/project/bug_list (also works without SSL). Issues are tied to the particular sub-project they belong to, or to the global "DeforaOS" project in any other case. New issues can be filed at https://www.defora.org/os/project/bug_new; a user account is currently necessary. It is possible to register at https://www.defora.org/os/user/register.

Even though a complete rewrite of the website is currently in progress, the reporting procedure is expected to remain the same.
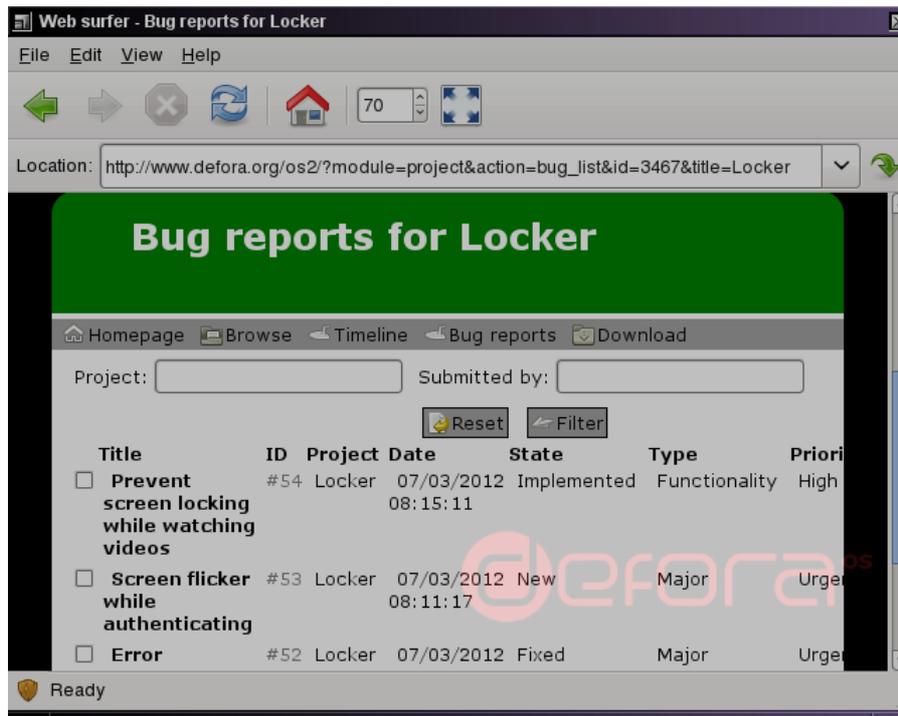
The newer interface on defora.org to list and report bugs

3.6.3 For the NetBSD Project

Likewise, NetBSD offers a public issue tracker online, where existing issues can be listed through a search engine at http://www.netbsd.org/support/query-pr.html. There are mainly two ways to report new issues:

- with the "send-pr" utility on any NetBSD system (requires a functional local mailer)

- by filling in a form online, at http://www.netbsd.org/cgi-bin/sendpr.cgi?gndb=netbsd

Issues for the kernel, base system and packages from pkgsrc are all listed in the same place, with issues bound to the relevant category: it can be "kern" for the kernel, "pkg" for packages for instance.
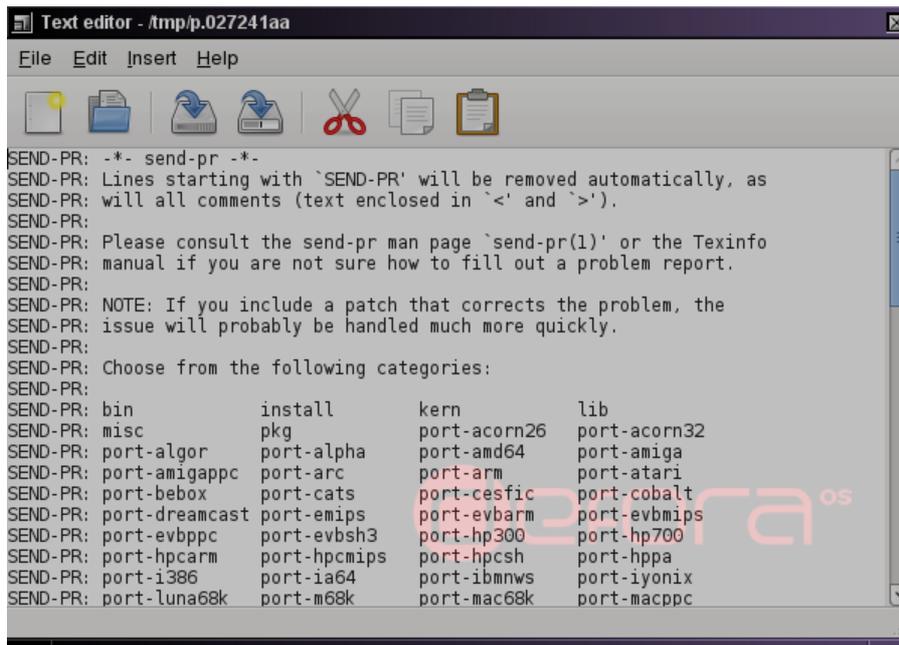
```
Text editor - /tmp/p.027241aa

File  Edit  Insert  Help

SEND-PR: -*- send-pr -*-
SEND-PR: Lines starting with `SEND-PR' will be removed automatically, as
SEND-PR: will all comments (text enclosed in `<' and `>').
SEND-PR:
SEND-PR: Please consult the send-pr man page `send-pr(1)' or the Texinfo
SEND-PR: manual if you are not sure how to fill out a problem report.
SEND-PR:
SEND-PR: NOTE: If you include a patch that corrects the problem, the
SEND-PR: issue will probably be handled much more quickly.
SEND-PR:
SEND-PR: Choose from the following categories:
SEND-PR:
SEND-PR: bin            install          kern            lib
SEND-PR: misc           pkg              port-acorn26    port-acorn32
SEND-PR: port-algor     port-alpha       port-amd64      port-amiga
SEND-PR: port-amigappc  port-arc         port-arm        port-atari
SEND-PR: port-bebox     port-cats        port-cesfic     port-cobalt
SEND-PR: port-dreamcast port-emips       port-evbarm     port-evbmips
SEND-PR: port-evbppc    port-evbsh3      port-hp300      port-hp700
SEND-PR: port-hpcarm    port-hpcmips     port-hpcsh      port-hppa
SEND-PR: port-i386      port-ia64        port-ibmnws     port-iyonix
SEND-PR: port-luna68k   port-m68k        port-mac68k     port-macppc
```

An example submitting a bug report for the NetBSD Operating System

4. Errata

4.1 Known issues on the WeTab GmbH WeTab

4.1.1 The 3G modem is not supported

Affected component

Telephony and Internet connectivity over mobile networks

Description

The built-in 3G modem in the WeTab is recognized on the USB bus as "Huawei Mobile WCDMA EM770W", but no driver is claiming it. There is an existing, Open Source driver for the Linux kernel, but it could not be ported or re-implemented before the end of this project.

Workaround

Most external USB 3G modems are supported. They should be automatically detected by the telephony application, DeforaOS Phone once inserted.

Resolution

This issue could not be solved during this project.

4.1.2 Additional sensors are not supported

Affected component

The GPS, ambient light and accelerometer sensors on the WeTab

Description

Hardware support on the WeTab is incomplete in the NetBSD kernel, and at least the following components are not supported:

- GPS built-in device
- Ambient light sensor
- Accelerometer

Besides these components, the video camera is working but not directly supported by the desktop applications at the time of this report.

Workaround

The accelerometer can be emulated with the screen rotation button from the panel; this is however unsupported

by the vesa video driver.

The mplayer media player can be configured to capture and display the camera output.

Resolution

This issue could not be solved during this project.

4.1.3 The USB chipset may fail to initialize

Affected component

The boot sequence on the WeTab tablet.

Reference

None officially.

Description

The system firmware of the WeTab tablet was found to be pre-initializing the USB chipset of the WeTab tablet in ways preventing the NetBSD kernel to boot properly. While the NetBSD kernel would be probably to blame and require a fix, a workaround sequence was researched and found in the meantime.

Workaround

Cold-booting the tablet was found to be necessary in place of regular reboots.

Resolution

Pending.

4.2 Known issues on the Lenovo IdeaPad S10-3t

4.2.1 The 3G modem is not supported

Affected component

Telephony and Internet connectivity over mobile networks

Description

The built-in 3G modem in the Lenovo IdeaPad S10-3t could not be identified. This issue was not investigated any further, although an Open Source driver for Linux is believed to already exist.

Workaround

Most external USB 3G modems are supported. They should be automatically detected by the telephony

application, DeforaOS Phone once inserted.

Resolution

This issue could not be solved during this project.

4.2.2 General system instability

Affected component

To be determined, likely the ACPI support from the kernel

Description

Besides not being able to power off properly, the system regularly freezed while performing some operations in particular:

- accessing internal sensors on the ACPI bus
- accessing the USB video camera

No particular cause could be identified. These problems have not been reported so far.

Workaround

None known.

Resolution

Pending.

4.3 Known issues in DeforaOS

4.3.1 Mailer/#27: SMTP support

Affected component

The mail client for the DeforaOS desktop

Reference

http://www.defora.org/os/project/bug_display/274/SMTP-support?bug_id=27

Description

Mailer can send mails using the sendmail utility, but the possibility to use arbitrary SMTP servers is necessary.

Workaround

Configure a local mailer as desired.

Resolution

Planned.

4.3.2 Browser/#42: Copy & paste is not supported across browser windows

Affected component

The file manager for the DeforaOS desktop

Reference

http://www.defora.org/os/project/bug_display/3638/Copy---paste-is-not-supported-across-browser-windows?
bug_id=42

Description

Copying files via copy & paste (either the menu entries or keyboard shortcut) does not work from a browser window to another.

Workaround

None known.

Resolution

Planned.

4.3.3 Browser/#47: Drag & Drop support is not optimal

Affected component

The file manager for the DeforaOS desktop

Reference

http://www.defora.org/os/project/bug_display/3643/Make-sure-drag---drop-works?bug_id=47

Description

Dragging and dropping files to and from either browser windows or the desktop should be supported. This should be done according to the XDND standard.

Workaround

The current implementation is partial and seems to have problems.

Resolution

Planned.

4.3.4 Keyboard/#51: Support additional keyboard layouts

Affected component

The virtual keyboard for the DeforaOS desktop

Reference

http://www.defora.org/os/project/bug_display/3647/Support-additional-keyboard-layouts?bug_id=51

Description

The ability to change the keyboard layout would be very welcome, including:

- text-based support (or otherwise automatically generated)

- switching while the application is running.

Workaround

None known.

Resolution

Planned.

4.3.5 Keyboard: Modifier keys are not properly tracked

Affected component

The virtual keyboard for the DeforaOS desktop

Reference

None officially.

Description

Modifier keys such as Shift, Alt and Ctrl are typically implemented using toggle buttons. Their status is however not properly tracked while switching layout pages:

- the key preview is wrong

- modifier keys are not toggled back on to reflect their current status

Workaround

Activate the relevant toggle buttons once to twice to reset them to a known status.

Resolution

Planned.

4.4 Known issues in NetBSD

4.4.1 kern/45853: Support for touchscreens in xf86-input-mouse

Affected component

The wscons framework in the NetBSD kernel.

Reference

http://gnats.netbsd.org/45853

Description

This was a follow-up to PR/45850, "USB multi-touch panels are not supported", in which the newly-introduced uts(4) driver reports absolute coordinates (on the X and Y axis notably). This was however unsupported by the xf86-input-driver for modular-xorg in pkgsrc.

Workaround

A patch was supplied in this problem report.

Resolution

Support for the xf86-input-mouse driver in the base system from NetBSD is believed to have been included, but is probably awaiting integration upstream of within pkgsrc (for modular Xorg support).

4.4.2 kern/45872: Issues and fixes for tpcalib (touchscreen calibration)

Affected component

The tpcalib framework in the NetBSD kernel.

Reference

http://gnats.netbsd.org/45872

Description

While working on the uts(4) driver, the touchscreen device had to be calibrated for proper mapping of screen coordinates. The driver itself used the tpcalib helper from the kernel, as also used in the ztp(4) and uep(4) drivers, or on the hpcarm, hpcmips and hpcsh architectures apparently. There were some problems with tpcalib_ioctl() for WSMOUSEIO_SCALIBCOORDS, when setting calibration values through sample collection:

1. An invalid array access

2. Division-by-zero

3. An integer overflow (and invalid calibration)

Workaround

A partial patch was provided, as well as a procedure to avoid the integer overflow.

Resolution

The full resolution of this issue is still pending.

4.4.3   kern/46189: wsfb/genfb red adn blue colours are reversed on bit depths 15 16 24 and 32

Affected component

The wscons framework in the NetBSD kernel

Reference

http://gnats.netbsd.org/46189

Description

Generic splash screen support is performed on x86 platforms via the genfb(4) driver, when properly initialized by the bootloader. Unfortunately, there was a mismatch for the bit order between the driver and the VESA palette in the 15, 16, 24 and 32-bits depth video modes. As a result, the colors were wrong for every color except green (BGR versus RGB).

Workaround

The patch supplied in the problem report fixes the issue on x86 platforms specifically. Additional issues have been observed with ATI/AMD Radeon graphics cards, but do not affect the WeTab or the IdeaPad S10-3t.

Resolution

In progress.

4.4.4 Intel GMA graphics cards are not fully supported

Affected components

The NetBSD kernel and xf86-video-intel drivers from the base system and pkgsrc.

Reference

None officially.

Description

The graphics card found on both the WeTab and the IdeaPad S10-3t are not supported directly by the xf86-video-intel driver, and need Kernel Mode Settings (KMS) support from the system. This was not supported by the NetBSD kernel during this project. Additional video output connectors may or may not work, depending on the BIOS settings (when available) or the connection to a screen during the bootup sequence.

A NetBSD developer, Grégoire Sutre (gsutre@), is currently working on this issue. His progress can be tracked online on his github account, found at https://github.com/gsutre/netbsd-drmgem.

Workaround

Grégoire Sutre has provided test binaries online, in his message to the current-users mailing-list: http://mail-index.netbsd.org/current-users/2012/03/11/msg019462.html.

Resolution

In progress.

4.4.5 port-amd64/45197: failing to load secondary boot

Affected component

The boot process for the NetBSD Operating System

Reference

http://gnats.netbsd.org/45197

Description

Recent builds of the boot components were found to fail to function properly, at least while preparing USB memory sticks for the installation process. The system was looping at boot, with the following error message:

```
NetBSD/x86 msdos Primary Bootstrap
Boot failed (errno 5): Can't open /boot
```

The bug report referenced here is believed to be the same issue.

Workaround

The boot binaries from the official 5.1.2 release could be used successfully instead. They did not support setting up the splash screen though.

Resolution

Planned.

4.5 Other known issues

4.5.1 Finger-scrolling is not supported

Affected components

Applications using the Gtk+ toolkit (hence the complete DeforaOS desktop environment)

Reference

None.

Description

Finger-scrolling support inside graphical applications is often expected by tablet users to be supported. This was however not the case in the system delivered.

Workaround

Most applications provide a scrollbar on the side whenever scrolling is possible. In some cases, finger selection can be abused to emulate the behavior of finger scrolling, although possibly not in the direction expected from users.

Resolution

Pending.

4.5.2 Single-click activation is not supported

Affected components

Applications using the Gtk+ toolkit (hence the complete DeforaOS desktop environment)

Reference

None.

Description

Single-click activation is supported by most GNOME applications, but was never ported or integrated to the underlying Gtk+ libraries.

Workaround

This issue could be compensated in some desktop applications by the selection signals; this was however not applied universally, for lack of consistency, possibility of ambiguous overlap of features, or simply lack of time.

Resolution

Pending.

4.5.3 Third-button emulation is not supported

Affected components

Applications using the Gtk+ toolkit (hence the complete DeforaOS desktop environment)

Reference

None.

Description

Right-click events from the mouse are often emulated by a longer press of a finger on the screen. This feature was not available in the final tablet setup delivered.

Workaround

There are apparently at least two possibilities to implement this feature:

- via the "libgtkstylus" Gtk+ extension (as done within the hackable:1 smartphone environment)
- via the Xorg server directly (therefore potentially applying to the whole environment)

These workarounds were not applied, due to lack of time for proper testing and integration.

Resolution

In progress.

4.5.4 Additional languages are not supported

Affected component

End-user applications

Reference

None.

Description

English was the default language configured on the tablet. Although this was not the only language supported on the system, the user was never offered to change it to the other languages supported.

The virtual keyboard application in particular suffered from the same problem, as additional layouts were not available. This issue was tracked through a separate bug report though.

Workaround

The default language could be changed through the environment, with the addition of the following line to the ~/.bashrc file for instance:

```
export LANG=fr_FR.UTF-8
```

Resolution

Planned.

5. Appendix

5.1 Hardware specifications for the Lenovo IdeaPad S10-3t

Processor	Intel Atom N450 at 1.66 GHz
Memory	Up to 2 GB
Storage	Up to 320 GB hard disk drive
External memory	MicroSD card slot
Networking	• Gigabit Ethernet • Wireless 802.11bgn adapter (Broadcom) • 3G adapter • Bluetooth
Display	10.1" at 1024 ×600
Touchscreen	Multi-touch
Camera	On-screen, 1.3 MP
Sound	Built-in Stereo speakers
Battery	Up to 4 hours autonomy (8 hours with an 8-cell battery)
Connectivity	• 3 USB ports • 1 VGA video output • headphones, microphone
Additional sensors	
Dimensions	280×15~20.2×174 mm
Weight	1.25 Kg

5.2 Hardware specifications for the WeTab GmbH WeTab

Processor	Intel Atom N450 at 1.66 GHz
Memory	1 GB
Storage	32 GB flash
External memory	MicroSD card slot, up to 32 GB (SDHC)
Networking	• Wireless 802.11bgn adapter (Atheros) • 3G adapter (Huawei) • Bluetooth 2.1 + EDR
Display	11.6" at 1366 ×768
Touchscreen	Multi-touch
Camera	Front, 1.3 MP

Sound	Built-in speakers
Battery	Up to 6 hours autonomy
Connectivity	• 2 USB ports • 1 mini-HDMI video output • headphones • docking connector
Additional sensors	• Accelerometer • Ambient light • GPS
Dimensions	294×194×14 mm
Weight	1 Kg

5.3 Configuration files

5.3.1 /etc/mk.conf

```
#licenses
ACCEPTABLE_LICENSES+=gnu-gpl-v2
ACCEPTABLE_LICENSES+=postfix-license
ACCEPTABLE_LICENSES+=vim-license

#environment
FAILOVER_FETCH=yes
FETCH_BEFORE_ARGS=-4
FETCH_USING=fetch
MASTER_SORT=.fr .org
MASTER_SORT_REGEX=paris
MOTIF_TYPE=lesstif
PASSIVE_FETCH=yes
USE_DESTDIR=yes
USE_SSP=yes
X11_TYPE=modular

#packages
PKG_DEFAULT_OPTIONS+=-arts
PKG_DEFAULT_OPTIONS+=cups
PKG_DEFAULT_OPTIONS+=embedded
PKG_DEFAULT_OPTIONS+=-esound
PKG_DEFAULT_OPTIONS+=ipv6
PKG_DEFAULT_OPTIONS+=lang-de
PKG_DEFAULT_OPTIONS+=lang-en-GB
PKG_DEFAULT_OPTIONS+=lang-en-US
PKG_DEFAULT_OPTIONS+=lang-fr
PKG_DEFAULT_OPTIONS+=ssl

#evince
PKG_OPTIONS.evince+=dvi

#modular-xorg-server
PKG_OPTIONS.modular-xorg-server+=dri

#mplayer
PKG_OPTIONS.mplayer+=dts
PKG_OPTIONS.mplayer+=faad

#postfix
PKG_OPTIONS.postfix+=tls

#thunderbird
PKG_OPTIONS.thunderbird+=mozilla-enigmail
PKG_OPTIONS.thunderbird+=mozilla-lightning
```

```
#xchat
PKG_OPTIONS.xchat+=gtkspell
PKG_OPTIONS.xchat+=mitshm

#xterm
PKG_OPTIONS.xterm+=freetype
```

5.3.2 /usr/pkgsrc/pkgchk.conf

```
amd64   = bulk wetab
console = bulk wetab
deforaos= bulk wetab
desktop = bulk wetab
embedded= bulk wetab
fancy   = bulk wetab
media   = bulk wetab
meta    = bulk
xorg    = bulk wetab

#essential
pkgtools/pkg_chk

#all
editors/vim                     *
security/sudo                   *
shells/bash                     *

#platform-specific
comms/obexapp                   amd64
comms/obexftp                   amd64
sysutils/acpica-utils           amd64
sysutils/acpidump               amd64
sysutils/dmidecode              amd64
sysutils/e2fsprogs              amd64

#host-specific
mail/postfix                    bulk wetab
misc/openoffice3                bulk wetab
sysutils/estd                   bulk wetab

#bulk
meta-pkgs/desktop-gnome         bulk
meta-pkgs/modular-xorg-apps     bulk
meta-pkgs/modular-xorg-drivers  bulk
meta-pkgs/windowmaker-desktop   bulk
meta-pkgs/xfce4                 bulk
meta-pkgs/xfce4-extras          bulk
net/samba                       bulk
pkgtools/pkg_comp               bulk

#console
chat/irssi                      console
mail/mutt                       console
misc/figlet                     console
misc/screen                     console
net/lftp                        console
net/rsync                       console
net/wget                        console
news/slrn                       console
pkgtools/pkgin                  console
security/gnupg                  console
security/mozilla-rootcerts      console
sysutils/smartmontools          console
www/lynx                        console

#deforaos
wip/deforaos-asm                deforaos
wip/deforaos-browser            deforaos
wip/deforaos-configure          deforaos
wip/deforaos-editor             deforaos
wip/deforaos-framer             deforaos
wip/deforaos-init               deforaos
wip/deforaos-keyboard           deforaos
wip/deforaos-locker             deforaos
wip/deforaos-mixer              deforaos
```

```
wip/deforaos-panel              deforaos
wip/deforaos-pdfviewer          deforaos
wip/deforaos-phone              deforaos
wip/deforaos-player             deforaos
wip/deforaos-probe              deforaos
wip/deforaos-surfer             deforaos
wip/deforaos-todo               deforaos


#desktop
audio/bmp                       desktop
audio/bmp-flac                  desktop
audio/id3ed                     desktop
audio/pulseaudio                desktop
audio/vorbis-tools              desktop
chat/pidgin                     desktop
chat/pidgin-otr                 desktop
chat/xchat                      desktop
databases/shared-mime-info      desktop
fonts/dbz-ttf                   desktop
fonts/dejavu-ttf                desktop
fonts/freefont-ttf              desktop
fonts/freefonts                 desktop
fonts/liberation-ttf            desktop
fonts/ms-ttf                    desktop
fonts/vera-ttf                  desktop
graphics/dcraw                  desktop
graphics/gimp                   desktop
graphics/gnome-icon-theme       desktop
graphics/gqview                 desktop
graphics/hicolor-icon-theme     desktop
graphics/librsvg                desktop
graphics/tango-icon-theme       desktop
graphics/xfce4-icon-theme       desktop
mail/thunderbird                desktop
multimedia/libdvdcss            desktop
multimedia/mplayer              desktop
net/synergy                     desktop
net/Transmission                desktop
news/pan                        desktop
print/apsfilter                 desktop
print/cups                      desktop
print/evince                    desktop
print/foomatic-filters-cups     desktop
print/foomatic-ppds-cups        desktop
textproc/hunspell-de            desktop
textproc/hunspell-fr_FR         desktop
wm/metacity                     desktop
www/firefox                     desktop
www/liferea                     desktop
x11/xscreensaver                desktop
x11/xscreensaver-demo           desktop


#embedded
meta-pkgs/matchbox              embedded
net/dhcpcd-gtk                  embedded


#fancy
graphics/dia                    fancy
graphics/inkscape               fancy
net/tsclient                    fancy
x11/gdm                         fancy
x11/gnome-themes                fancy
x11/gtk2-engines                fancy


#wmaker
audio/wmsmixer                  wmaker
net/wmnd                        wmaker
sysutils/wmSMPmon               wmaker
wip/wmitime                     wmaker
wm/windowmaker                  wmaker
wm/wmakerconf                   wmaker
x11/docker                      wmaker


#xorg
meta-pkgs/modular-xorg-drivers  xorg
meta-pkgs/modular-xorg-fonts    xorg
x11/glx-utils                   xorg
```

```
x11/modular-xorg-server        xorg
x11/sessreg                    xorg
x11/setxkbmap                  xorg
x11/xauth                      xorg
x11/xbacklight                 xorg
x11/xconsole                   xorg
x11/xdm                        xorg
x11/xdpyinfo                   xorg
x11/xev                        xorg
x11/xfontsel                   xorg
x11/xhost                      xorg
x11/xkill                      xorg
x11/xmodmap                    xorg
x11/xprop                      xorg
x11/xrandr                     xorg
x11/xrdb                       xorg
x11/xset                       xorg
x11/xsetroot                   xorg
x11/xterm                      xorg
x11/xvinfo                     xorg
x11/xwininfo                   xorg
```

5.3.3 ~/.browser

```
plugins=dirtree,preview,properties,volumes
alternate_rows=1
sort_folders_first=1
show_hidden_files=0
```

5.3.4 ~/.desktoprc

```
[background]
color=#000000000000
extend=0
how=centered
wallpaper=/deforaos.jpeg

[icons]
background=#000000000000
foreground=#ffffffffffff
font=Sans Bold 8
layout=none
```

5.3.5 ~/.gtkrc-2.0

```
gtk-font-name = "Sans 7"
gtk-icon-sizes = "gtk-button=32,32:gtk-large-toolbar=48,48:gtk-menu=48,48:gtk-
small-toolbar=24,24:panel-smaller=24,24:panel-small=32,32:panel-large=48,48"
gtk-icon-theme-name = "gnome"
gtk-key-theme-name = "Emacs"
gtk-theme-name = "Clearlooks"
gtk-toolbar-icon-size = GTK_ICON_SIZE_LARGE_TOOLBAR
gtk-toolbar-style = GTK_TOOLBAR_ICONS
gtk-touchscreen-mode = 1
```

5.3.6 ~/.locker

```
auth=slider
```

5.3.7   ~/.mime

```
[application/msword]
open=swriter

[application/postscript]
open=evince

[application/pdf]
open=pdfviewer

[application/rtf]
open=swriter

[application/vnd.ms-excel]
open=scalc

[application/vnd.ms-powerpoint]
open=simpress

[application/vnd.oasis.opendocument.presentation]
open=simpress

[application/vnd.oasis.opendocument.spreadsheet]
open=scalc

[application/vnd.oasis.opendocument.text]
open=swriter

[application/vnd.openxmlformats-officedocument.spreadsheetml.sheet]
open=scalc

[application/vnd.openxmlformats-officedocument.wordprocessingml.document]
open=swriter

[application/x-bittorrent]
open=transmission

[application/x-planner]
open=planner

[application/x-shellscript]
open=view
edit=editor

[image/svg+xml]
open=/usr/pkg/bin/view
edit=inkscape

[image/*]
open=/usr/pkg/bin/view
edit=gimp

[text/html]
open=surfer
edit=editor
view=view

[text/*]
open=view
edit=editor

[video/*]
open=player

[application/mbox]
open=mailer
```

5.3.8 ~/.panel

```
top=phone,spacer,gsm,gps,bluetooth,battery,volume,systray,clock,close
top_size=panel-small
bottom=main,keyboard,tasks
bottom_size=panel-large

[lock]
command=xset s activate

[keyboard]
command=keyboard -x
width=540
height=180
```

5.3.9 ~/.phone

```
plugins=gprs,oss,panel,profiles,ussd

[modem_hayes]
baudrate=460800
hwflow=1
device=/dev/ttyU0

[plugin_gprs]
attach=0
systray=1

[plugin_profiles]
default=General
```