Gaël PORTAY

Free Software Consultant

Working experiences

Vocational

March 2016–Today **Free Software Consultant**, *Savoir-Faire Linux*, Montréal.

Savoir-Faire Linux is a company of consultants specialized in *Free-Software*. I am part of the Industrial Solution Team who is in charge of developing the *Linux Embedded Systems* that are embedded in the devices of our customers.

- I implemented a framework in C++ 11 to exchange data between different devices. I used *ZeroMQ* as the bus for communications and *Protobuf* as the format for data exchange. I developed a system of plug-in based on *ld*. The code is covered using *gcov*.
- I bumped the embedded software of an *IoT* device to the latest version of *Yocto*. I moved the init system to *systemd*. I fixed low-level and system related issues.
- I customized the interface written in *LuCI* of a network device based on *OpenWrt*. I compiled and setup the docker daemon to run *Docker* images on that device.

July 2010–September 2015 **Embedded Linux Engineer**, *Overkiz SAS, Somfy group*, Archamps.

Overkiz is specialized in *Cloud Computing* for *Home-Automation*. It develops a solution which connects objects from Home to *Internet* (*IoT*). It consists of a gateway that links home-automation devices to its servers. We can control objects from Home thanks to a smart-phone and Web-Services. I was part of the Embedded Team who is developing the Embedded Linux System of gateways.

- I was co-maintainer of our home-made Embedded Linux distribution. I did the integration of software from the Open-Source community.
- I was also responsible for the deployment of the embedded software updates.
- I set up the Yocto Build System that builds the embedded software. It builds from scratch the whole embedded software. Yocto allows to save hours to the embedded developers and minimizes errors when releases are built introduced by human operations.
- I developed home-made frameworks and applications to support new home-automation protocols into our gateways. I was also in charge of developing the application that creates the connection between the box and the server. Frameworks and applications are both developed in C++.
- I developed kernel modules and I did two boards bring-up and mainlined them inside into Linux kernel sources.

Training Periods

2009 **Master Trainee**, *LC Mobility*, Australia.

LC Mobility is specialized in receiving foreign Ph.D. students. I made a market search about the mobility of the Australian students in Australia.

2008 **3rd-year Engineer Trainee**, *Freescale Semiconductors*, Toulouse.

I was responsible for the development of a battery-charging driver on Freescale smartphone platform (ARM-11 based). This mobile-phone is power-supplied by a Li-ion battery. It runs Nokia S60 (Symbian OS). I developed the driver in C++ using Symbian mechanisms.

2007 **2nd-year Engineer Trainee**, *Sagem Monetel*, Valence.

I was in charge of memory optimization (dynamic and static) of a C application embedded in a banking terminal powered by two ARM processors. I used listing and map files generated by the free GNU/GCC compiler to locate heavy memory structures. I reduced the application size by more than 30%.

2004 **Technical Trainee**, *Sagem Monetel*, Valence.

I was in charge of changing the compiler of an embedded C application to the free GNU/GCC compiler. I also developed a benchmark to demonstrate the power of a payment terminal during a transaction.

Open-Source Contributions

Buildroot I added package QtWebEngine and config Raspberry Pi 3 (64-bits).

Barebox I fixed readline implementation to prevent from printing a non printable character and from looping to infinity. I fixed a NULL pointer dereference that causes an crash.

genimage I added a hdimage property to set the position of the extended partition recorded into the Master Boot Record.

Lighttpd I mainlined CRLs for client certificate verification and make client verification errors ignored is not enforced.

jack2 I fixed uninitialized members that cause invalid reads when run by valgrind.

Linux Kernel I added two Atmel SoC based device-trees.

Atmel at91bootstrap I brought support for UBI. The goal is to improve critical upgrades against unexpected power-cuts. Critical volumes, such as kernels or bootloaders, are duplicated and stored in UBI static volumes. The bootstrap simply checks the volume integrity using update flag from UBI headers.

CURL I upgraded libcurl to that it is compatible with the latest PolarSSL Library API. I also fixed a bug with the polling mechanism that causes a timeout while processing SSL handshake with the distant server.

OPKG I improved CURL integration to allow libcurl related settings into the configuration file. I also fixed unexpected behaviors.

Dropbear I removed compilation warnings from the entire project.

Miscellaneous

2000–2006 **Casual Jobs**, *Miscellaneous Employers*.

Education

2008–2009 **Master MAE**, *IAE*, Grenoble, *5-year University degree*.
Last year of Master in Management and Business of Administration.

2005–2008 **Master 3I**, *Polytech'Grenoble*, Grenoble, *5-year University degree*.
Engineering degree in Industrial Computing and Microelectronic.

2004–2005 **DEUG TIC**, *University of Savoy*, Chambéry, *2-year University degree*.
in Information Technology.

2002–2004 **DUT ISI**, *IUT of Valence*, Valence, *2-year Academic and Technical degree*.
in Industrial Computing.

2001–2002 **DEUG SV**, *University of Savoy*, Chambéry, *2-year University*.
1st-year of a 2-year University degree in Biology.

Projects

Personal

2018 **blkpg-part**, *Partition table and disk geometry handling utility*, C, GPLv2.
blkpg-part creates, resizes and deletes partitions on the fly without writing back the changes to the partition table. Thanks to blkpg-part, it is possible to export any consecutive blocks, that are not already part of a partition, as a temporary partitioned block device. A typically use case in embedded systems is to export hidden blobs that are stored in raw in block devices (i.e. blobs that are not stored into a file-system).

2018 **kmake**, *Maintain kernel dependencies by extending Kbuild*, Makefile, GPLv3.
kmake runs on top of make using a set of Makefiles to extend the Kbuild's features. It enhances the kernel build-system with the build of a tiny rootfs and an additional Qemu target to emulate the linux kernel alongside a userland. The userland is a tiny InitRAMFS cpio archive based on a static build of busybox.

2017-2018 **dosh**, *Run a user shell in a container with pwd bind mounted*, Bash, Docker, MIT.
dosh is a shell-like frontend written in *bash* for *docker-run*. It runs commands in a container; using the current user, with pwd bind mounted.

2017-2018 **tini**, *Simple init daemon that spawns processes and reaps zombies*, LGPLv2.1.
tini is a damn small process spawner and zombie reaper.

2015-2017 **mpkg**, *Managing packages from a shell script*, Shell, MIT.
mPKG is a lightweight package manager written in pure Shell. It uses standard utilities such as sh, grep, tar, wget and awk that are shipped in any POSIX system. This makes mPKG suitable for embedded devices that usually embed Busybox which provides everything it needs in a single binary.

2015-2018 **templates**, *Some templates source files*, C/C++, Shell, Makefile, MIT, BSD, GPL.
Those examples of code are mostly written in C/C++, Shell/Bash, and make/Makefile. Those languages are the foundations of low-level and system development.

School

2009 **LHOG Minatec**, *900MHz amplifier*, Grenoble.
Conception of a GSM-900 amplifier at LHOG Minatec (micro-nano technology laboratory). Design, simulation, layout, assembly, tests, and characterization.

2008 **LHOG Minatec**, *CMOS 6 bits accuracy ADC*, Grenoble.
Conception of a 6-bits Analog-to-Digital Converter in AMS CMOS 0,35µm technology at CIME Minatec (micro-nano technology laboratory). CMOS comparators had been designed using Cadense, and the Corrector and the Decoder developed in VHDL.

2006 **Polytech'Grenoble**, *ASM HC12 assembler*, Grenoble.
Conception of an assembler for the 68HC12 instruction set. This command line tool was developed in C on Linux.

2005 **University of Savoy**, *Windows Desktop Search*, Chambéry.
Conception of Desktop Search for Windows. I developed the application in Java using the Eclipse environment. The engine indexes all files from the computer and enables searching files by file name using regular expressions. The UI was built thanks to Swing framework.

2004 **IUT of Valence**, *2D Game Engine*, Valence.
Conception of a basic 2D game engine using Microsoft Direct Draw library. The player moves in a 2D map with collision handling. I developed the engine in C++.

Languages

French	Mother tongue	
English	Good knowledge	*I spent one year traveling around Australia and New-Zealand (from April 2009 to March 2010). I scored 765 at TOEIC in 2008.*

Computer skills

Programming languages	Shell/Bash scripting, Python, C/C++ (11, STL), Assemblers (68k, MIPS), Java, VHDL, LaTeX
Others	Linux Kernel, Git, Autotools, Cross-compilation, Yocto

Interests

BDE/BDS Polytech'Grenoble	Member of Student Association at Polytech'Grenoble. Organization of the Integration event in 2006 (350 students, 3 days). Organization of two Ski event in 2006 and 2007 (400 students, 3 days, 11 engineering school from France).
Mountain	Ski, hiking, snowshoes hiking.
Table tennis	Player, referee, and coach (9-year practicing).
Photography	Hobbyist, Canon EOS 500D

References

Overkiz, SAS.	Florent PELLARIN,Chief Operational Officer(f.pellarin@overkiz.com)
Savoir-Faire Linux Inc.	Jérôme OUFELLA,Vice President Technologies(jerome.oufella@savoirfairelinux.com)