



Gaël PORTAY

Consultant en Logiciels Libres

Expériences

Professionnelles

- Novembre **Développeur Logiciel Senior**, *Collabora*, Montréal.
- 2018–Juin *Collabora* est une entreprise internationale de consultants spécialisés dans le *Logiciel Libre*.
2022 Je fais parti de l'équipe Core de la branche ingénierie qui est en charge d'analyser les problèmes des clients et de concevoir des solutions en utilisant à la fois les technologies issues des *logiciels libres* et l'expertise technique de l'entreprise.
- Investigation et débogage du noyau Linux
 - J'ai analysé la diminution de performances de transferts sur le bus *SPI* avec *kernelshark*, suite à une mise à jour du noyau Linux et de son sous système *SPI*.
 - J'ai participé à la résolution de deux blocages systèmes du pilote *V4L2* sur la plateforme *i.MX6* ; lorsque le flux vidéo est stoppé puis redémarré immédiatement.
 - J'ai analysé un inter blocage dans l'allocation de mémoire continue *CMA* sur l'architecture *ARM* ; lorsqu'une tâche effectue une allocation de mémoire continue entraînant la migration de page sur d'autres tâches pouvant elles-mêmes nécessiter de nouvelles allocations de mémoire continue.
 - J'ai participé à la création d'une distribution dédiée au jeu vidéo basée sur *Arch Linux*
 - J'ai modifié et créé des paquets avec *makepkg* via les fichiers *PKGBUILD*, et j'ai mis en place le dépôt pour le gestionnaire de paquets *pacman* en développant des scripts *Shell* utilisant *repo-add* et *repo-remove*.
 - J'ai créé des images disques *UEFI* ainsi que ses artefacts de mise à jour *RAUC* et *casync*.
 - J'ai participé à la mise en place la configuration de l'installateur multiplateforme *calamares* et j'ai développé plusieurs plugins spécifiques en *Python*.
 - J'ai participé à la création de scripts *Shell* et des conteneurs *Docker* pour l'automatisation des tâches de constructions de paquets et d'images sur *Jenkins* puis *GitLab CI/CD*.
 - J'ai effectué la redistribution des contributions aux logiciels libres (*systemd*, *plymouth*, *grub-debian*, *debian-installer*, *dracut*, *calamares*, *RAUC*, *casync*...).
 - Création de plugins *fwupd/LVFS* pour la mise à jour de micro logiciels de périphériques
 - J'ai ajouté le support pour la mise à jour des hubs *USB* Genesys Logic et des moniteurs *USB-C* HP.
 - J'ai ajouté le support pour la mise à jour des souris « esport » Steelseries via la connections sans-fil *2.4G* et la connections filaire *USB*.

Mars 2016– **Consultant en Logiciel Libres, Savoir-faire Linux, Montréal.**

Aujourd'hui Savoir-faire Linux est une entreprise de consultants spécialisés dans le *Logiciel Libre*. Je fais parti de l'équipe Solution Industrielle qui est en charge de développer les *Systèmes Embarqués Linux* qui équipent les objets connectés de nos clients.

- J'ai développé un framework en C++ 11 d'échange de données entre différents objets connectés. J'ai utilisé *ZeroMQ* comme bus de communication et *Protobuf* comme format pour les données échangées. J'ai développé un système de plugin basé sur *ld*. Le code est couvert en utilisant *gcov*.
- J'ai mis à jour le logiciel embarqué d'un objet connecté vers la dernière version de *Yocto*. J'ai migré le système d'init vers *systemd*. J'ai résolu les problèmes bas niveau et système.
- J'ai personnalisé l'interface développée en *LuCI* d'un périphérique réseau basé sur *Open-WRT*. J'ai compilé et configuré le démon docker pour faire fonctionner des images *Docker* sur ce périphérique.

Juillet 2010– **Ingénieur Linux Embarqué, Overkiz SAS, groupe Somfy, Archamps.**

Septembre 2015 Overkiz est une entreprise spécialisée dans le « *Cloud Computing* » pour la domotique. Elle propose une solution qui connecte les objets de la maison à *Internet (IoT)*. Elle se compose d'une passerelle qui fait le lien entre les périphériques domotiques et ses serveurs. Il est possible de piloter ses objets de la maison grâce à son smartphone et des services web. J'ai fait partie de l'équipe qui développe le système Linux embarqué de la passerelle.

- J'étais coresponsable de la distribution Linux embarquée. J'intégrais des outils issus de la communauté du logiciel libre.
- J'étais responsable du déploiement des mises à jour du logiciel embarqué.
- J'ai mis en place le système de « Build automatisé » Yocto. Cet outil construit l'intégralité du système embarqué de la passerelle. Yocto permet de gagner de nombreuses heures à l'équipe tout en minimisant les erreurs liées à des interventions humaines lors de la construction de nouvelles versions.
- J'ai développé des frameworks internes et des applications ajoutant la prise en charge de nouveaux protocoles domotiques. J'ai été en charge du développement de l'application responsable du maintien de la connexion entre la passerelle et le serveur. Les frameworks et les applications sont développés en C++.
- J'ai développé des modules noyaux et ajouté le support de deux nouvelles cartes électroniques développées par notre équipe dans le noyau Linux.

Stages

2009 **Stage de Master, LC Mobility, Australie.**

LC Mobility est une entreprise spécialisée dans l'accueil de doctorants étrangers. J'ai effectué une étude de marché sur la mobilité des étudiants australiens, en Australie.

2008 **Stage Ingénieur de 3e année, Freescale Semiconductors, Toulouse.**

J'ai été responsable du développement du pilote de charge de batterie de la plateforme de téléphonie mobile de Freescale (ARM-11). Ce téléphone mobile est alimenté par une batterie Li-ion. Il fonctionne sous Nokia S60 (Symbian OS). J'ai développé le pilote en C++ en utilisant les mécanismes propres à Symbian.

2007 **Stage Ingénieur de 2e année, Sagem Monetel, Valence.**

J'ai été chargé d'optimiser l'emprunte mémoire (dynamique et statique) d'une application C embarquée dans un terminal de paiement bancaire. J'ai utilisé les fichiers listing et mapping générés par le compilateur libre GNU/GCC pour localiser les parties de code utilisant le plus de ressources mémoires. J'ai réduit la taille de l'application de plus de 30 %.

2004 **Stage Technicien de 2de année de DUT, Sagem Monetel, Valence.**

J'ai eut la charge de porter une application C embarquée vers le compilateur libre GNU/GCC. J'ai également développé un outil de test de performance afin de démontrer la puissance du terminal de paiement durant une transaction bancaire.

Contributions Open-Source

- Buildroot J'ai ajouté le paquet QtWebEngine et la configuration Raspberry Pi 3 (64-bits).
- Barebox J'ai modifié l'implémentation de readline pour prévenir l'affichage de caractères non imprimables et de boucler à l'infini. J'ai corrigé le déréférencement d'un pointeur NULL qui causait un crash.
- genimage J'ai ajouté une propriété à hdimage pour configurer la position de la partition étendue enregistrée dans le Master Boot Record.
- Lighttpd J'ai ajouté le support des CRLs pour la vérification du certificat client et ignoré les erreurs de vérification du certificat client si l'option n'est pas forcée.
- jack2 J'ai initialisé des variables membres non initialisées qui entraînaient des lectures invalides lorsqu'il était exécuté sous valgrind.
- Noyau Linux J'ai ajouté deux nouvelles plateformes basées sur des SoC d'Atmel (device-tree).
- Atmel J'ai ajouté le support d'UBI. Le but est d'améliorer les mises à jour critiques vis-à-vis d'éventuelles coupures de courant. Les volumes critiques, tels que les noyaux ou les bootloaders, sont dupliqués et stockés dans des volumes UBI statiques. Le bootstrap vérifie simplement l'intégrité du volume en utilisant le bit de mise à jour présent dans l'en-tête UBI.
- CURL J'ai mis à jour libcurl afin qu'il soit compatible avec la dernière version des API de la librairie PolarSSL. J'ai également corrigé un bogue avec le mécanisme de polling qui entraînait un timeout lors de la négociation SSL avec le serveur distant.
- OPKG J'ai amélioré l'intégration de CURL en autorisant certaines options liées à libcurl dans le fichier de configuration. J'ai également corrigé des comportements inattendus.
- Dropbear J'ai corrigé les warnings de compilation du projet.

Divers

2000–2006 **Emplois saisonniers**, *Plusieurs employeurs*.

Éducation

- 2008–2009 **Master MAE**, IAE, Grenoble, *Bac +5*.
Année spéciale de Master en Management des Administrations et des Entreprises.
- 2005–2008 **3I**, Polytech'Grenoble, Grenoble, *Bac +5*.
Diplôme d'ingénieur en Informatique Industrielle et Instrumentation.
- 2004–2005 **Licence TIC**, Université de Savoie, Chambéry, *Bac +3*.
2e année de Licence en Technologie de l'Information et de la Communication.
- 2002–2004 **DUT ISI**, IUT de Valence, Valence, *Bac +2*.
Diplôme universitaire de Technologie en Informatique et Systèmes Industriels.
- 2001–2002 **DEUG SV**, Université de Savoie, Chambéry, *Bac +2*.
1re année de Diplôme d'Etudes Universitaires Générales en Science de la Vie.

Projets

Personnels

- 2018 **blkpg-part**, *Utilitaire de table de partition et géométrie du disque*, C, GPLv2.
blkpg-part créait, redimensionne et supprime des partitions à la volée sans enregistrer les changements dans la table de partition. Grâce à blkpg-part, il est possible d'exporter des bloc consécutives, qui ne font pas partie d'une partition, comme un périphérique temporaire. Un cas typique d'utilisation dans les systèmes embarqués est d'exporter les binaires qui sont stockés dans des périphériques de type bloc (i.e. des binaires qui ne sont pas enregistrés dans un système de fichier).
- 2018 **kmake**, *Extension de Kbuild*, Makefile, GPLv3.
kmake fonctionne au-dessus de make en utilisant des Makefiles pour étendre les fonctionnalités de Kbuild. Il complète le système de construction du noyau avec la construction d'un rootfs minimal et d'une règle Qemu supplémentaire pour émuler le noyau linux au côté d'un espace utilisateur. L'espace utilisateur est une petite archive cpio en InitRAMFS basée sur une compilation statique de busybox.
- 2017-2018 **dosh**, *Execute un shell utilisateur dans un conteneur*, Bash, docker, MIT.
dosh est une interface à la shell écrit in Bash pour docker-run. Il exécute les commandes dans un conteneur ; utilisant les droits utilisateurs, avec le répertoire courant monté.
- 2017-2018 **tini**, *Simple démon d'initialisation*, C, LGPLv2.1.
tini est un petit démon d'initialisation qui démarre des processus et s'occupe des processus zombies.
- 2015-2017 **mpkg**, *Gestion des paquets depuis un script shell*, Shell, MIT.
mPKG est un gestionnaire de paquet léger et écrit en pure Shell. Il utilise des utilitaires comme sh, grep, tar, wget et awk fournit par tout système POSIX. mPKG est adapté aux systèmes embarqués utilisant Busybox.
- 2015-2018 **templates**, *Quelques exemples de code.*, C, Shell, Makefile, MIT, BSD, GPL.
Ces exemples de code sont principalement écrit en C/C++, Shell/Bash and make/Makefile. Ces langages sont les fondations du développement bas niveau et système.

Universitaires

- 2009 **LHOG Minatec**, *Amplificateur 900MHz*, Grenoble.
Conception d'un amplificateur GSM-900 au LHOG Minatec (laboratoire de micro nano technologie). Conception, simulation, routage, assemblage, tests et caractérisation.

- 2008 **LHOG Minatec**, *Convertisseur Analogique Numérique CMOS 6-bits*, Grenoble.
Conception d'un convertisseur analogique numérique 6-bits utilisant la technologie 0,35µm CIME Minatec (laboratoire de micro nano technologie). Les comparateurs CMOS ont été conçus via le logiciel Cadence, le correcteur et le décodeur ont été développés en VHDL.
- 2006 **Polytech'Grenoble**, *Assembleur HC12*, Grenoble.
Conception d'un assembleur en 2 passes pour le jeu d'instruction 68HC12. Cet outil en ligne de commande a été développé en C sur Linux.
- 2005 **Université de Savoie**, *Windows Desktop Search*, Chambéry.
Conception d'un moteur de recherche rapide pour Windows. J'ai développé cette application en Java sous l'environnement de développement Eclipse. Le moteur indexe tous les fichiers de l'ordinateur et permet à l'utilisateur de rechercher des fichiers via des expressions régulières. L'UI a été développée via le framework Swing.
- 2004 **IUT de Valence**, *Moteur de jeu 2D*, Valence.
Conception d'un moteur de jeu 2D basic en utilisant les bibliothèques Microsoft Direct Draw. Le joueur se déplace sur une carte 2D avec gestion des collisions. J'ai développé le moteur en C++.

Langues

- Français Langue maternelle
Anglais Bonnes connaissances

Compétences techniques

- Langages de programmation Shell, Makefile, C/C++
Autres Linux Kernel, Git, Cross-compilation, Yocto, Buildroot, Docker

Centres d'intérêts

- BDE/BDS Membre du bureau des étudiants de Polytech'Grenoble.
Polytech'Grenoble Organisation du week-end d'intégration en 2006 (350 étudiants, 3 jours).
Organisation de deux week-end ski en 2006 et 2007 (400 étudiants, 3 jours, 11 écoles d'ingénieurs du réseau Polytech).
Montagne Ski, raquettes et randonnées.
Tennis de table Joueur, arbitre et entraîneur (9 ans).

Références

- Overkiz Florent PELLARIN, Directeur des Opérations(f.pellarin@overkiz.com)
Savoir-Faire Jérôme OUFELLA, Vice Président Technologie(jerome.oufella@savoirfairelinux.com)
Linux
Collabora Dave BEVAN, Responsable du Personnel d'Ingénierie(dave.bevan@collabora.com)