

Domaines d'intervention

- Architecture de systèmes à fortes contraintes (puissance, mémoire, **temps réel**, ...).
- Configuration et personnalisation du **noyau Linux** (intégration temps réel, réduction du temps de boot, couche réseau spécifique, réduction de la consommation électrique, ...).
- Ecriture de drivers. Debug matériel. Développement de firmwares.
- Gestion de la qualité sur systèmes embarqués ou temps réels. **Optimisation, instrumentation et profiling** de systèmes.
- **Portage** de noyaux ou d'applications. Développement de **BSP**.
- Support technique des équipes de développement, transfert de connaissances, gestion de projet.

Expériences

Depuis 2009 **Fondateur de Sysmic, société spécialisée sur Linux embarqué.** Quelques références:

- *Eolane/Areva*: Architecture et développement d'un système de supervision d'un réacteur nucléaire (ARM Cortex)
- *Odyssee Systèmes*: **Portage du noyau Linux** sur un SoC de Kontron (PowerPC 8541)
- *Nagra* : Mise en œuvre d'un système par simulation des environnements embarqués (Carte à puce)
- *Enensys*: Audit sur une régression de performance d'une nouvelle plateforme matérielle. Optimisation des timings de la DDR.
- *Eolane*: Audit sur un bug sur le DMA du contrôleur d'image de l'iMX6. Isolation des accès disque sur un CPU afin d'améliorer les temps de réponses des autres CPU.
- *Bouygues Telecom*: Audit des obligations légales liées à la réutilisation de projets open source dans les produits fixes de Bouygues Telecom
- *Tokhatec*: Expertise technique pour l'intégration des drivers Poulso sur Linux (majoritairement Atom). Expertise technique pour l'amélioration de la consommation électrique des chipsets Atom (*P-states* et *C-states* non-fonctionnels).
- *Thales*: Résolution de bugs noyau (régression sur le l'appel système « reboot »)
- *IER*: Conseil en ingénierie logicielle. Développement RFID (ARM)
- *Key Consulting*: Expertise pour l'architecture logicielle d'un satellite (Sparc)
- *Substantiel*: Gestion de problèmes d'**intégration de matériel** Conseil pour la mise en place d'une **QA**. Déploiement du noyau et des différents pilotes sur le parc de 10000 machines distantes (x86)
- *Sysmic*: Création d'un environnement de développement pour Linux embarqué à base de Buildroot, Eclipse, cmake, Qt et git
- *Ingénico*: Debug, conseil en architecture logicielle
- *Buildroot*: Ajout de paquets, ajout de la possibilité de créer automatiquement un dépôt de paquets.

Décembre 2013 — **Merck Millipore**. Optimisation des performances de rendu HTML5 sur iMX53. Optimisation du moteur Webkit de Qt5. Création d'un BSP optimisé pour les besoins de Merck.

Septembre 2013 — **Canon Research**. Etude de faisabilité : Modification d'un firmware de chipset Wifi afin d'implémenter un protocole Wifi expérimental :

- Etude des firmwares et des drivers des chipsets Broadcom et Atheros
- Etude du protocole d'expérimentation

Avril 2013 — **Canon Research**. Optimisation du codec H265 (aka HEVC) pour les vidéos de très haute définition (7840x4320). Tout l'enjeu de cette optimisation se situe sur la bonne parallélisation des traitements (certains problèmes de cohérence de cache entre CPU peuvent ralentir les calculs):

- Instrumentation du code à l'aide de perf et de vtune : nombre de cycles CPU, cache misses, NMI, etc...
- Utilisation des shaders pour certains traitements vidéo
- Utilisation des instructions SSE4 et AVX2
- Conception nouvelle architecture de parallélisation des calculs
- Optimisation de compilation : *arc profiling*, *link time optimization*, OpenMP,...

Mars 2013 — **Enensys**. Développement d'un driver pour une interface d'acquisition vidéo MPEG :

- Intégration du DMA avec l'interface Unix
- Utilisation de la norme OpenFirmware
- Mise en place des Fast IRQ (FIQ) (Mécanisme spécifique aux processeurs ARM. Permet des temps de réponse de l'ordre de 12 cycles d'horloge. L'ISR ne peut alors utiliser que certains registres et aucune autre exception ne peut se déclencher durant son exécution. L'ISR doit alors être développée en assembleur et ne pas faire appel à d'autres fonctions du noyau)

Juin 2012 — **Areva** souhaitait le développement d'une nouvelle version d'un de ces systèmes de supervision de réacteur

nucléaire. Le projet à été confié à **Eolane**. Eolane, spécialisé dans le développement électronique, souhaitait profiter de ce projet pour acquérir la maîtrise technologique d'un OS de haut niveau comme Linux. Je suis intervenu chez Eolane dans ce contexte. La mission s'est déroulée autour d'un Freescale iMX27.

- Développement système de mise à jour du firmware, drivers, intégration noyau
- Adaptation du BSP (fourni par Aramdeus, à base de Buildroot) à la porteuse du client
- Architecture
- Debug hardware
- Transfert de compétences
- Conseil en qualité logicielle

2010 — 2012 **Bouygues Telecom**. Responsable technique de la *Quality Assurance* des produits sous Linux embarqué (STi70xx, Mips, arm, x86):

- Mise en œuvre de la chaîne d'intégration continue. Déploiement automatique des cibles (NFS, TFTP, JTAG, RS232. ...).
- Mise en œuvre des outils de profiling des cibles (ftrace, valgrind). Couverture de code (gcov), mesure de l'empreinte mémoire et des consommation CPU (/proc)
- Mise en œuvre d'outils de vérification de la robustesse des firmware (ld_preload, C)
- Mise en œuvre de systèmes de scripting des tests (perl, shell, serveurs de flux vidéo, DVB, MPEG, H264, cartes à puces, SmartVision, CAS, redrat, infrarouge, RS232, etc...)

Depuis 2009 **Formateur Linux embarqué et développement noyau Linux** auprès des entreprises (STMicroElectronics, Thales, Davidson, ...). Création de BSP et de drivers, debug application et noyau, utilisation de sondes Jtags, mise en œuvre de Linux temps réel

Intervenant extérieur à l'Insa Rennes, Polytech'Paris (membre du conseil pédagogique) et Polytech'Tours.
Cours de temps réel et systèmes d'exploitations.

2006 — 2009 Ingénieur **systèmes** embarqués chez Hybrid MC, spécialisé en réalité virtuelle pour la TV. Développement des automates pour le déplacement des caméras.(Assembleur, C++, Lua, ...) ...

Hybrid MC est une PME de 7 personnes vendant des solutions de virtualisation de décors pour l'audiovisuel. La virtualisation consiste à instrumenter les caméras afin de connaître très précisément leur position (dans l'espace et dans le temps). Hybrid MC utilise des caméras robotisés commandés à partir d'un (ou plusieurs) pupitre distant. J'étais responsable du développement matériel et logiciel bas niveau de ces systèmes.

- Contraintes temps réelles
- Mise en place d'un puissant système de scripting en temps réel à base de **Lua**. Utilisation de **Swig** pour générer les *bindings* Lua. Gestion de plusieurs contexte d'exécution Lua. Partage des objets entre les instance de Lua et le programme natif. Adaptation du *garbage collector* pour satisfaire le partage d'objet et les problématiques temps réel.
- Recherche et développement des algorithmes de déplacement des automates (cinématique inversée, calculs matriciels, génération de modèles numériques des lentilles, gestion du scripting de scénario, etc...)
- Formation des équipes techniques, principalement à l'étranger
- Travail en collaboration avec divers corps de métier: spécialistes de la 3D, électroniciens, mécanicien, électro-mécaniciens

2006 Ingénieur développement système chez *Easter-Eggs*. SS2L spécialisée en développement et administration de systèmes open sources. Principale mission : développement d'un OCR sur ARM ayant pour but l'aide à la lecture pour les aveugles :

- Compilation et utilisation de la toolchain Gnu avec une sonde Jtag
- Initialisation du processeur : initialisation des horloges, gestion du cache d'instructions, mise en service du MMU,...
- Développement des algorithmes de **traitement d'image** et de reconnaissance de texte
- *Profiling* et optimisation de ceux-ci (15 images en 320x240 traitées par seconde).
- Mise en place d'une *Quality Assurance* sur le projet. Compilation automatique, à l'aide des Autotools, et tests automatiques sur les architectures i386 et ARM (Shell et Python).
- Travail en collaboration avec un électronicien

2004 Projet de fin d'étude : Recherche automatique de similarité de code. Utilisation de différents algorithmes de comparaisons à divers niveaux de la chaîne de compilation. Gestion de l'explosion combinatoire de ce genre de technique à l'aide d'utilisation massive de cache.

2004 **Assistant pédagogique** en C/UNIX pour les élèves de 1ère année d'ingénierie d'EPITA (~250 élèves). **Formateur** Apache, Samba, Shell, Perl et technologies UNIX dans le cadre de la junior entreprise.

2003 **Dassault Aviation** section Défense

Formation

Juillet 2005 Diplôme d'ingénieur en informatique à l'**EPITA** spécialité **sciences cognitives et intelligence artificielle.**
