

Stéphane Doyon
s.doyon@videotron.ca

Education

- **Master’s degree in computer science** at *Université Laval*, completed in 1999. Scholarship from *NSERC*, Dean’s Honor List (“Faculté des études supérieures”).
- **Bachelor’s degree in computer science** at *Université de Montréal*, completed in 1997. Awards include the Medal of the Governor General (97), Faculty of Arts and Sciences Dean’s Scholarship (96), a research scholarship from *NSERC* for undergraduate students (summer 95).

Work Experience

Google, since 2008.

Maximum Throughput, 2002 to 2007

- Team leader for the “Systems” group since February 2007.
- Optimization, tuning, feature integration and bug fixing in the *Linux* kernel inside a *video I/O* and *network attached storage* solution used in the film production industry.
- Developed and integrated experimental patches to enhance throughput, scalability or stability in the NFS server, pieces of filesystem and memory management code, the block elevator, volume management, SCSI drivers, network card drivers and bits of the networking layer. Experience with a multiprocessor platform.
- Porting of drivers or features across kernel versions.
- Developed a high-throughput video recorder application which receives images over *InfiniBand*.
- Wrote a multi-threaded high-throughput file transfer tool for data migration.
- Wrote tools to reproduce various I/O patterns, measure throughput and gather system statistics, as well as a multi-machine throughput test harness. Languages used include *python*, *C*, *bash* and *perl*.
- Developed build scripts for a custom distribution. Acquired experience working with RPM spec files while managing patches to various system utilities and drivers. System administration experience.

VIPSwitch (Software group and Architecture team), 2000 to 2002

Development of a high-capacity switch/router for metropolitan area networks, based on special-purpose hardware. Developed specifications defining the behavior of each hardware component for the forwarding of packets in accordance with several protocols (MPLS, IPv4, IPv6, ARP, *ethernet bridging*, IEEE 802.1Q). Defined protocols allowing to configure the hardware and to handle protocol exceptions in software. Implemented a communication link between the CPU module and the hardware, a virtual network device for the software, as well as the software mechanisms required to support the ARP protocol (which involved modifications to the *Linux* kernel core networking functions). Also some embedded development experience while working on a *wide area network* router product.

Gemplus Canada (“groupe carte”), 1999 to 2000

Design and development on a JavaCard smart card. I developed a cryptographic applet that stores private keys on the card and uses them to sign or decipher messages. Inter-operability with other cards; adherence to some cryptographic standards. Optimization of the space required by the applet. Work related to compilation of applets, communication with the card, and debugging of a new card. Work related to several smartcard standards. Perfected my knowledge of **cryptography** : **public key** cryptography and symmetric cryptography, electronic signatures, message digests, message authentication codes.

VisuAide (R&D group), 1997

Worked on a digital recorder and personal organizer designed for blind users. The only outputs available on this device were sound and pre-recorded speech. I added several new functions such as support for an extended (non-standard) diskette format and I made improvements to the user interface. System programming in a real-time embedded environment, with significant space constraints.

Loto-Québec (“Groupe architecture et recherche technologie”), summer 1996

I designed and tested an algorithm solving a feasibility issue for a certain function that was essential in implementing a new game (strong constraints on execution time). Simulations to foresee the number of winners for a variant of that game.

Projet Safari (intelligent tutoring systems, Université de Montréal), summer 1995

Implementation using CGI programs of an interactive web questionnaire (with user state tracking). Analyses related to tutoring strategies.

Centre for Research on Transportation (Université de Montréal), summer 1994

Operations research project : modeling queues of cars at a traffic light.

Projects and Achievements

- Master’s thesis on **Java security** : investigation of the security aspects of the *Java* architecture, in particular of the *bytecode verifier* which performs a static analysis on mobile code. Understanding of the workings of the *Java* virtual machine and experience with its *bytecode* instruction set. I published three articles on the subject : in *Computer Communications* (Elsevier Science), proceedings of the *2000 ACM Symposium on Applied Computing* and proceedings of *Parallel and Distributed Computing and Systems*.
- Developed a streaming digital audio player appliance based on a wireless router and the *OpenWRT* open firmware.
- Contributed patches to the *Rockbox* project, the open firmware for portable digital audio players : enhancements to spoken menus and dialogs.
- Developed a **vocal note-taking tool and audio book reader** on the iPaq personal digital assistant under *Linux*, using *python* and C. Experience cross-compiling kernels and applications, working with a flash filesystem, debugging via serial port, using an *nfsroot* work environment, porting applications to the *StrongARM* architecture (alignment, converting floating point algorithms to **fixed point**). Experience working with **digitized sound** : mixing, resampling, speed scaling, compression algorithms... Experience designing a **non-visual user interface**.
- Some experience with *Asterisk*, the open source PBX : implemented a game of tic-tac-toe played over the phone, as well as a call relay function. Some experience forwarding the SIP protocol through firewalls.
- Developed a system for digitizing and preparing talking books from audio cassettes, using *python* and C. It processes sound from four track cassettes played on an ordinary deck, detecting silence periods and audible tones for indexing purposes.
- Co-author of the BRLTTY software which drives Braille display terminals and offers screen review functions enabling blind people to use *Linux*. Among other contributions, I

co-authored the **first Linux driver for a *USB* display**.

- Co-author of *cicero*, a software **text-to-speech engine** for the French language, written in *python*. Based on context sensitive rules, it translates text into phonemes with prosodic attributes, which are then fed through the *MBROLA* voice synthesizer. The *Cicero* pronunciation rules have also been used to create the French voice for *eSpeak*, another Open Source TTS solution.
- As a project during my undergraduate studies, I took part in the implementation of a **machine learning** algorithm applied to **speech recognition**.
- During four semesters I was part of a team of students who **taught an introductory course to *UNIX*** and who offered technical support for new students of the computer science and mathematics departments at *Université de Montréal*.
- Developed a web-based scanner frontend using *quixote*, *XMLRPC*, *python* and *apache*.
- Developed a script that finds new references on a given subject as they appear on the web over time, using the ***Google API***.
- Needs assessment, purchase of equipment and **setup of a local area network** of Windows stations for a small financial services business.
- Wrote a program that plays the game of othello using the *minimax* search algorithm and the *alpha-beta* cut-off procedure.

Skills

English and french (spoken and written).

Programming :

- Strong knowledge and experience with **C**, good knowledge of ***python***, as well as *Bourne shell* scripting (and common *UNIX* utilities).
- Some experience with **C++**, *perl*, *Java* and a bit of **PHP**.
- Concurrent programming experience.
- Notions of secure programming and understanding of classic types of **security** vulnerabilities.
- ***UNIX* system programming** experience, including networking, threads, files, process management, serial communications...
- Additional *Linux* kernel development experience : *USB* driver, IDE *hot-swapping*.

Linux Operating system :

- Some experience configuring services : from mail and web to ***VPNs***, uninterruptible power supplies, modems and backups.
- **NETfilter** and advanced networking : configuring firewall rules, limiting connection rates, port knocking, **NAT** and port forwarding, tunneling.

Miscellaneous :

- Some knowledge of the **SQL** language and of relational database systems principles. Brief experience with **PostgreSQL**.
- Experience working with **L^AT_EX**.
- Familiarity with **CVS** and *subversion*. Experience managing patches with *quilt*. Basic knowledge of *git*.

Leisure activities : woodworking, canoe camping, music.