

David M. Krauss
2 South Brush Drive
Valley Stream, NY 11581
(217) 721-3164, cell
dkrauss@uiuc.edu

Objective

I aim to gain valuable real-world experience and broaden my technical knowledge with a summer internship in research and development. I would like to learn about hardware design and related workflows. It would be especially gratifying to apply my knack for creative metaprogramming* to a problem in a hardware design tool set.

**metaprogramming*: Writing code to direct tools, as opposed to writing a "product." More generally, writing orthogonally to the source language, especially when refactoring or extending a project which has "outgrown" its language and/or extending the language itself.

Skills

C++ "language lawyer." Experience in C, OCaml, PPCAsm (like IBM HLASM), AltiVec, Forth, and M68K assembly. Intermediate in VHDL and FPGA logic implementation. Experience with source management by metacompilation and application of logical or functional semantics without language support. Experience programming within many diverse environments (if not most) of Mac OS X and 9. Familiar with WebPack, CodeWarrior, Project Builder, MacsBug, Open Firmware, and MPW. Familiar with the fundamentals of UNIX. Can operate basic EE lab equipment. Can apply calculus.

Education

University of Illinois, Urbana-Champaign, 8/2002—present.

Bachelor of Science in Computer Engineering

GPA 3.4/4.0 from three semesters (Engineering course GPA 3.7)

Expected graduation 5/2006

Experience

5/2004—8/2004: Open Firmware Intern, Apple Computer Inc., Cupertino, California. Paul Resch, manager.

I assisted the team responsible for boot and initial bring-up software for Macintoshes. Provided some filesystem bug support. Implemented a system to help integrate disparate hardware engineering teams, improving on infrastructure in software build and employee communications systems.

6/2003—12/2003: Classic Compatibility Intern, Apple Computer, Inc. Matt Caprile, manager.

I assisted engineers in various aspects of maintaining the Classic compatibility environment, which runs older applications in a virtual Mac OS 9 machine, on equal footing with native apps.

Reimplemented synchronization in the audio and disk insertion subsystems for better performance and tolerance. Researched those systems and documented them for future engineers. Wrote an application automation tool for QA. Helped in general bug triage and maintenance. Job entailed extensive research, as expertise in the myriad functions of the old OS was often unavailable. Reverse engineering was the norm, as the system's heterogeneous nature often obfuscated the mapping of object code back to source, and bugs in third-party products without source also needed diagnosis.

7/2001—present: Project lead, Balance Software, Inc., Portland, Oregon. Brian Doyle, manager.

I implemented a product which is intimately connected with the Mac boot process. Given a general functional genre, I specified, designed, and implemented most of the product. Required extensive research of Mac OS X, Open Firmware, and Mac OS 9. Several custom development tools (a linker, a cross-linker and loader for a proprietary binary format, and a Forth disk utility) were necessary to build and test it. Some of the most important aspects of the project rely on undocumented parts of the OSes; research therefore included reverse-engineering. Learned Forth and 68K asm on the job. Project also implemented in C++, C, and PPCAsm. A set of spin-offs required expertise in the HFS+ file system.

7/2002: OFSplit, an Open Firmware program which forks the current instance of Open Firmware into a new physical address space. The user can then switch among the multiple Open Firmware contexts in memory. A proof of concept for a Macintosh logical partitioning project.

5/2002: Ephemeroboot, a low-level system hack for the MacHack competition. Developed in a couple weeks after school days. An Open Firmware program rewrites the description of installed RAM. A Mac disk driver then boots from the reserved RAM as a disk. Because RAM is momentarily depowered at reboot, I derived a data correction algorithm to restore the information from basic principles. The API to access physical RAM had to be reverse-engineered from an Apple application. Because of school, I was unfortunately unable to enter the contest.

1/2000—present: Project Upper/Mute. In free time, I work on an innovative new extensible compiler framework and compiler. I developed coherent theories about compiler design, formalized them, and translated them into C++ (for implementation) and technical English (for entry into the Intel science competition). This project includes a custom C++ utility class library.

10/1999—3/2000: MacAque, a highly parallel neural-network simulator. I independently researched molecular neurobiology and neural-network theory, then applied ideas to create a highly-optimized, yet extensible, simulator. Its engine is coded entirely in AltiVec/PPCAsm assembly. Memory is carefully organized to maximize cache benefits. A few support apps in C++ were quickly developed.

5/1998—present: Miscellaneous projects. A suite of PPCAsm memory utilities that can compare two strings faster than a serialized multiply, a runtime library compatible with both Metrowerks and Apple C++ compilers, and various classes that make efficient C++ a breeze.

Awards

Two Intel Achievement in CS awards, and recognition by the Army, Navy, and Air Force, for neural-network project. Long Island Software Awards finalist and Shipley-Ronal Science Invitational winner in CS for compiler project.

Other Interests

I enjoy playing Jazz and classical music on the trumpet, as well as creative 3D graphic design. I'm also an avid hiker and cyclist.

References

Brian Doyle, brian (at) balancesoftware.com

Matt Caprile, mcaprile (at) apple.com

Paul Resch, pmr (at) apple.com

Roger Pantos, rpantos (at) apple.com (coworker engineer, Apple)