

Jeffrey Schwab

(617) 407-2542

jeff@schwabcenter.com

<http://schwabcenter.com>

25 Crescent Street, Unit 238, Waltham, MA 02453

Objective

Software Architect seeks interesting projects, great code, and great people.

Accomplishments at General Electric / Morpho Detection, July 2009 - Present

Led the redesign and re-implementation of a scientific application for a hand-held device. Moved existing functionality from Windows CE to Linux. Drastically improved performance, capacity, and reliability. The changes have received extremely positive reviews from existing clients. Work was mostly done in C++, but with C device drivers and a hand-rolled, dynamic DSL. Most development was done without functioning hardware. Contracted through Adecco Engineering and Technical.

Accomplishments at CLK Design Automation, April 2008 - February, 2009

Improved the performance, functionality, and quality of a commercial EDA timing tool. Extended C and Tcl APIs, and added a C API to annotate semiconductor designs with SPEF-style parasitics. Fixed existing bugs. Increased the concurrency of code that was already heavily threaded. Served as C++ language expert.

Accomplishments at Silicon Navigator, September 2007 - March, 2008

Led new development of a commercial Schematic Editor. Enhanced support for visualization and manipulation of logical connectivity. Introduced support for multi-bit connectivity. Closed numerous existing bugs. Interfaced with the OpenAccess EDA database. Implemented graceful error-handling, adding descriptive visual markers for conditions that previously caused crashes. Collaborated in the design of a general-purpose event distribution mechanism to replace Qt signals, and in the refactoring of a graphical selection mechanism. Served as C++ language expert.

Accomplishments at IBM, June 2004 - September 2007

Ported FreeBSD device drivers to IBM blade servers. Led a small, dedicated team. Defined and automated a special-purpose build and release methodology. Wrote and integrated proprietary and Open Source C code. Added support for USB, mass storage devices, and TCP/IP over Ethernet.

Developed firmware for military avionics project. Wrote and debugged self-test software and extensions to Phoenix BIOS. Complied with International Traffic in Arms Regulations (ITAR). Merged C, MASM, and application-specific languages.

Worked with IBM's advanced research division to perform early chip design analysis. Generated custom VCD stimuli to stress designs in specific ways. Created, coordinated, and interpreted thousands of simultaneous simulations. Automated tool-chain feedback loop, enabling rapid, iterative approach to corner-case analysis.

Integrated tools into project-specific tool chains. Converted data formats using Perl, Python, and C++. Ported legacy infrastructure code from AIX to Linux. Provided ad hoc solutions for architecture, verification, and circuit design teams.

On-boarded successful new product delivery team. Coordinated team of twenty-five engineers. Managed VHDL designs using ClearCase and CVS. Established small server farm for batch simulations of RTL.

Accomplishments at Sun Microsystems, May 2000 - June 2004

Developed in-house software for use by microprocessor design teams. Rewrote an event-based functional coverage tool, for 100x improvements in performance and capacity. Wrote circuit analysis tools based on SPICE and MATLAB. Implemented a data mining program to track the release readiness of circuit designs. Provided documentation and support.

Demonstrated leadership ability on diverse, distributed teams. Achieved highest ratings on all performance reviews. Contributed to the tape-out of the UltraSPARC-V microprocessor. Helped define methodologies for design verification and I/O data management.

Technical Expertise

Languages: C++, Java, Perl, assembly, shell scripting, Tcl, Ruby, Python, Clojure (Lisp)
Tool chains: GNU (GCC and binutils), Microsoft, and Apple
Profilers: gprof and oprofile
Operating Systems: Linux, FreeBSD, Solaris/Unix, AIX, Microsoft Windows, and Apple OS X
Version Control: CVS, Subversion, ClearCase, and Git
Issue tracking: Bugzilla and Lighthouse
Libraries: Boost, POSIX, Qt4, Intel TBB, the STL, and others
Design styles: Object-oriented (OOP), Functional, Procedural, Generic
Web technology: (X)HTML, CSS, and limited JavaScript
Editors: Vim, XEmacs, and IDEs including Eclipse, XCode, and Visual Studio
Open Source Licensing: GPL, LGPL, and BSD/MIT
Other: C++ Template Metaprogramming, Design Patterns, Simulation, and Documentation

Formal Education

Graduate engineering course in Photonics
Boston University, Fall 2003

Baccalaureate Degree (BS) in Computer Systems Engineering
Boston University, May 2000
Dean's List

Academic Awards
National Merit Scholarship
Perfect score (1600) on GREs

Professional Education

Conferences and seminars, including Synopsys Users' Group (SNUG) and Cadence design tool classes
Sun Microsystems training on Solaris Administration and Six Sigma Methodology
IBM training on Intellectual Property law, with focus on patents
Wind River intensive courses on VxWorks
Delivered poster presentation on API usability at IBM Software Engineering for Tomorrow (SWEFT)

References available upon request.