Jed Davis

<jld@xlerb.net>
2000 Walnut St. #215, Boulder, CO 80302; 415-845-4647

· Work History

Mozilla Corporation

Software Engineer, February 2013 - present

- Designed and implemented system-level security sandboxing technology for a Web renderer
- Refactored legacy code for asynchrony and process separation
- Developed tools to investigate system-level performance on a novel mobile OS

VMware, vSphere (ESXi) Kernel Core Team

Member of Technical Staff, May 2010 - January 2013

- Maintained subsystems ranging from memory allocation to filesystem implementation to error reporting
- Improved filesystem and memory management infrastructure owned by other groups
- Debugged and fixed a variety of code, kernel and userlevel, as part of rotating bug triage duties
- Upgraded the kernel build to a newer C compiler, modifying the compiler to make this possible
- Implemented a system for kernel-specialized low-overhead detection of a class of memory usage errors

Previously:

- Researched potential applications of NAND flash memory for transparent persistence in garbage-collected managed runtime environments.
- Improved crash recovery efficiency of NetBSD's software RAID driver
- Gained primary responsibility for a small ISP's email system; significantly improved its performance
- Analyzed and corrected defects in the NetBSD kernel, including Xen paravirtual drivers

· Education

Northeastern University, Boston, MA

- **M.S.**, Computer Science, 2010

Oberlin College, Oberlin, OH

- **B.A.**, Computer Science (high honors) and Mathematics, 2003
- Honors project: composable object-orientation for agent-based modelling

· Skills

- **Programming Languages:** C, C++, Coq, Erlang, Forth, Haskell, JavaScript, `m4`, OCaml, Perl, Python, Rust, Scheme, `sh`/`sed`/AWK; TEX/LATEX, x86 assembly (i386, x86_64)
- **Operating Systems:** Linux (Debian), FreeBSD, NetBSD, VMware ESXi
- **Revision Control:** Git, Mercurial, Subversion, Perforce, CVS

· Miscellaneous

- Judges' Prize and 3rd place, ICFP Programming Competition, 2007 (as team of one)
- Implemented the Rust language's optional pointer optimization
- Lost the 20th International Obfuscated C Code Competition

References available upon request