

Nicholas Shectman

75 Lexington Ave
Somerville, MA 02144

Mobile Phone: +1.617.543.9265

Email: nicholas.shectman@gmail.com

Objective

A full-time design engineering or management position, in the Boston area, which will showcase my creativity, problem-solving ability, and process expertise.

Selected Publications and Patents

- N. Shectman and N. Shi. *Practical experiences with open-source software testing tools*. Poster session presented at the New England Database Summit. January 2010.
- *Method and System for Performing Frame Recovery in a Network*. US Patent 6,804,316. October 2004.
- *Large-Capacity Content Addressable Memory with Sorted Insertion*. US Patent 6,252,790. June 2001.
- J. C. R. Bennett, C. Partridge, and N. Shectman. "Packet Reordering Is Not Pathological Network Behavior". *IEEE/ACM Transactions on Networking*, vol. 7 no. 6. December 1999.

Employment History

Software Developer VoltDB May 2009–April 2011

Developed and debugged database and management feature, release, and test infrastructure code for a high speed, highly scalable, in memory database, in C++ and Java. Refactored code for improved testability and reliability, and encouraged others to do the same. Resident expert on general programming best practices and C++ coding issues, especially obscure language and compiler features.

- Instituted a machine reservation process for allocating servers of different configurations and operating systems to developers or automated tests as appropriate, generating builds and test code appropriately, monitoring and repairing server health and readiness, and checking that manual test processes are cleaned up before reallocating machines.
- Incrementally developed a parameterizable multiple node test automation, regression, and performance test infrastructure for use with Hudson, allowing tests using a subset of machines or single machines to run in parallel, followed serially by tests using larger numbers of machines.
- Developed, streamlined, and automated a software release process, including regression tests of the release process itself, improving repeatability and instituting safety checks.
- Integrated a smart power distribution unit into the test infrastructure, allowing machines to be power cycled remotely for test or maintenance purposes.
- Provided immediate and historical visual code quality feedback using new analytics and existing tests.

Member, Technical Staff Vanu, Inc August 2007–February 2009

Developed new features and bug fixes for a real-time cellular base station system in C++ on Debian Linux. Guided the GSM product team from reactive firefighting to agile process, delivering improved reliability, efficient, on-time feature delivery, and considered, prioritized defect resolution. Trained staff and developed tools to reduce escalation rate from customer support to development. Managed patch releases and provided release engineering support. Drove project scheduling to avoid resource starvation.

- Drove estimation, selected staff, and introduced new testing, documentation, consultant relationship, and maintenance separation practices, resulting in on-time delivery of a 4-task-year project.
- Developed and documented a system test and release validation process, improving system reliability without sacrificing feature agility.
- Introduced new static analysis techniques and supervised dynamic analysis of race conditions and threading invariant violations.
- Coordinated bug triage, diagnosis, and resolution, resulting in a 20-fold system reliability improvement in six weeks.
- Promoted to a team leadership role in April 2008 and to the lead of a new team in September 2008.

Hacker *ITA Software* January 2007–August 2007

Diagnosed and fixed bugs in a Java scheduling program. Analyzed emergent behavior of complex systems. Fixed bugs and participated in new feature design as part of a team implementing an airline reservation system using XML, Oracle, and Lisp.

- Presented explanations of difficult issues to a government customer, regaining trust for the company.

Senior Software Engineer *Permabit* January 2001–December 2006

As part of a small team, wrote a content-addressed storage software product in C++ on a Debian Linux platform, from zero lines of code through completion. Participated in the design and implementation of a secure client/server protocol, a failsafe versioning filesystem, performant NFS fileserver software, and an extensive distributed multiplatform test infrastructure.

- Helped drive adoption of Extreme Programming methodology, resulting in an 8-week feature-request-to-ship-date cycle and five on-time releases.
- Eliminated unnecessary complexity, resulting in improved maintainability, testability, and reliability.
- Brought outsourced subsections in house, lowering schedule risk.
- Achieved high availability on commodity hardware, resulting in industry-best pricing.
- Implemented data security and regulatory compliance filesystem features, including releasing SnapLock compatibility before SnapLock itself was released.
- Designed and implemented a nonlinear aggregated-file representation to enhance block reuse.

Network Scientist *BBN Technologies* October 1997–May 2000

Participated in the design and implementation of several different high-speed, multiprotocol routers, for internal, spinoff, and contract projects. Contracts and spinoffs included Quarry Technologies, Crescent Networks, and Avici Systems. Projects included a high-speed routing engine combining custom and embedded-processor hardware; a routing security architecture; network testbeds and demonstrations; and a multicast extension of a custom routing engine.

- Designed, implemented, redesigned and debugged FPGA programs.
- Designed, implemented, and debugged routing, measurement, testbed, and concept test software.
- Produced system, process, tutorial, contract proposal, and patent documentation.
- Supervised timing documentation and analysis.

Awards

- **Finalist**, MIT 50K Business Plan Competition, 2002
Formed a team to commercialize my Content Addressable Memory patent. Placed in the top seven of approximately 150 entrants, and attracted several offers of angel funding.

Skills

Languages: C/C++, Java, SQL, Python, Perl, Expect, TCL/tk, Lisp variants, sh/bash/csh, PHP, awk.

Tools: Eclipse, Hudson/Jenkins, ant, emma, testability explorer, lcov, gcc/g++, gdb, gprof, valgrind, junit, cppunit, Doxygen, Subversion, Perforce, CVS/RCS, Jira, RT, Bugzilla, MySQL, ethereal/wireshark.

Techniques: pthreads, smart pointers, futures, mix-ins, template remapping, exception registration, function call caching, design for test, fault insertion, health monitoring, FSM equivalence modeling, mixed-platform development and testing.

Processes: Extreme Programming, Scrum.

Education

BA in Mathematics *Harvard University* 1988–1992

Coursework included: advanced work in probability, combinatorics, and graph theory.