

## Nicholas Sheckman

75 Lexington Ave  
Somerville, MA 02144

Home Phone: +1.617.628.3595

Mobile Phone: +1.617.543.9265

Email: nicholas.sheckman@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

- *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. Sheckman. "Packet Reordering Is Not Pathological Network Behavior". *IEEE/ACM Transactions on Networking*, vol. 7 no. 6. December 1999.

## Employment History

### 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.
- Clearance information available on request.

## **Engineer** *EVI* September 1996–September 1997

- Designed and debugged circuits, FPGA programs, and hybrid multichip modules for digital radio.
- Designed and debugged an FSK encoder hybrid, a complex programmable logic hybrid, and a micropower digital transmitter control hybrid.
- Designed various revisions of and modifications to existing hybrids.
- Designed and debugged prototype digital circuit boards.
- Supervised schematic capture and layout of prototype and support circuitry.
- Produced system and contract proposal documentation.
- Top Secret clearance granted September 1997.

## **Research Scientist** *MIT AI Lab* November 1994–August 1996

Part of the design team for a humanoid robotic platform for artificial intelligence research.

- Redesigned and debugged a small scalable MIMD supercomputer, using simple shared memory and commercial microprocessor modules.
- Specified and designed a DSP module to be used in the supercomputer and in other lab robots.
- Supervised assembly and debugging of supercomputer nodes.

## **Community Organizer** *Arisia, Somerville Open Studios, et al.*

Volunteer roles for various community events. Led organizations ranging from 2 to 250 volunteers.

## **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++, Perl, Python, Java, SQL, Lisp variants, sh/bash/csh, awk, PHP.

*Tools:* gcc/g++, gdb, gprof, valgrind, junit, cppunit, Tinderbox, Doxygen, Perforce, Subversion, CVS/RCS, 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.