

Toby Cabot

Email: toby@caboteria.org

URL: <http://www.caboteria.org/~tobyc/>

Who am I?

- I'm a technical architect, engineer, and erstwhile manager who has been responsible for software development and IT organizations that build and run large, high-volume transaction systems. I design and write software and have grown and managed organizations up to 35 people.
- I have been successful at clean-slate development efforts, and have also taken 'raw' applications and turned them into manageable products.
- I have written, deployed, and managed business-critical Ruby On Rails applications, both web-facing and internal.
- I have worked extensively with large-scale technology, both proprietary and Open Source.
- I am well-versed in the procedural aspects of software development such as configuration management and policy development, and have worked extensively with Agile, Scrum, Kanban, and User Stories. Back in the dark ages I was even involved in an ISO-9000 certification (twice), and in a CMM effort.
- I have managed IT organizations through periods of growth and technology transitions.

Employment History

Chief Software and Operations Architect

Affectiva, Waltham, MA

April 2013–Present

Affectiva is an MIT Media Lab spin-off that reads peoples' emotions from their facial expressions. We use Tensorflow, Ruby on Rails, Python, MySQL, and Presto/Hadoop. We run our infrastructure on AWS using AMI's and Docker and we use Terraform, Chef and ad-hoc Python scripts to manage it.

I'm the company's engineering technical leader. This involves applying the core technology to specific applications such as market research, cloud APIs, and mobile apps. I'm the lead developer on our "Emotion as a Service" product which is a cloud app for processing video and is the keystone of the infrastructure that we use to manage our multi-billion-frame video database. I'm also the lead engineer on our Presto implementation that lets us quickly find specific frames from that data set.

For the "operations" part of my job I provide DevOps leadership for our cloud-based applications. I've been a change agent while respecting what works. I've introduced new technology (Rails, Docker, Terraform) to Affectiva and have improved existing processes (e.g., cutting our AWS monthly bill from \$20k to \$7k).

Principal Software Systems Engineer (AC5)

The MITRE Corporation, Bedford, MA

June 2010–April 2013

MITRE is a non-profit consulting company that works with the US Federal Government. I worked in the "Open Services" group improving healthcare systems, often using open-source technology.

Projects:

- **Prescription Drug Monitoring** Prescription drug misuse and abuse is an epidemic problem in the United States that causes more than 20,000 fatalities annually. Our project connected electronic health record systems in doctors' offices, pain clinics, and hospital emergency departments with a multi-state prescription drug monitoring network to give medical professionals critical information about their patients' controlled substance prescription history.
- **Health Program Delivery for Senior Citizens** The Federal Administration on Aging provides services for senior citizens to help them live independently. We built a registration system in Ruby on Rails to track use of these services by senior citizens.
- **Population Health Reporting** Population Health Reporting helps physicians deliver high-quality care by providing data about care delivery across their patient base. I converted PopHealth, our open source population health reporting tool, from a relational database to a document-oriented NoSQL database.

Senior Principal Engineer

Sermo, Cambridge, MA

May 2008–December 2009

Sermo runs the largest online physician community in the US. They started with a Java implementation and migrated to a hybrid that included both Java and Ruby on Rails components working cooperatively. I provided technical leadership on the Java implementation, architecture leadership for both the Java and Ruby implementations, and participated in the development of the Ruby components.

Consulting Engineer - Technical Architect

Reva Systems, Chelmsford, MA

April 2004–May 2008

Reva Systems is a venture-funded startup that makes an appliance (called the Tag Acquisition Processor or "TAP") that controls RFID readers. I designed the architecture and was the implementation tech lead for the "northbound" half of the appliance which interfaces with enterprise systems. This was an interesting development effort because the software had to both operate as a "sealed box" network component and as a platform for user-developed software. Figuring out how to embed J2EE technology was an interesting challenge.

I was the tech lead for the design and development of the Reva Multisite Manager which is a J2EE system that manages the clusters of Tag Acquisition Processors within an enterprise. The RMM polls the network of TAPs for status and statistics, stores that information in a database, and provides a web user interface for network health analysis and problem resolution.

In addition to software development (since Reva was a small startup at its inception) I managed much of the corporate IT infrastructure including file sharing, backups, and intranet services. Reva's network supports GNU/Linux, Windows, and Mac.

Consultant

Self-employed

June 2002–April 2004

After Black Ink Systems went out of business I became an independent software consultant. I worked on several projects including building and operating an online antique clock auction, building an online vacation packaging system, and performing a "post-mortem" analysis of a large Federal Government transaction-processing system.

Chief Technology Officer

Black Ink Systems, Cambridge, MA

November 2001–June 2002

Black Ink Systems was a venture-funded startup that built financial analysis products for communications carriers. The software delivered real time financial data to business people which allowed them to make timely decisions, which in turn helped them improve their margins.

Black Ink's product, the "Black Ink Margin Suite" was a Java/J2EE "mother ship" with distributed data collection front-ends that fed data to the mother ship over asynchronous queues. It had web user interfaces, and a relational database. It was interesting in its ability to reconcile the very different (and usually conflicting) requirements of interactive reporting and high volume real-time processing, including cost calculation and programmable alerts and alarms.

Co-founder and Chief Technology Officer

SaltFire, Inc., Arlington, MA

January 2000–November 2001

SaltFire's "Accurio" was a J2EE transaction processing system designed for use where a complex "value chain" delivered products or services. It was implemented as a set of EJB's which performed business logic and a set of interfaces, to users and to other systems. The user interfaces used XML and XSLT to build web pages. The system interfaces allowed other systems to integrate using either EJB or XMLRPC interfaces.

As CTO I was responsible for our products' architecture and technology (and much of the development, too). It was cool stuff - a distributed object system using Java, J2EE and XML technology internally and Web technology for user interfaces.

Director / Manager

*Kenan Systems / Lucent, Cambridge, MA
March 1996–January 2000*

Kenan Systems was a small, independent software company; their flagship product was Arbor/BP, a billing system for communications carriers. Kenan doubled in size annually so I wore many hats including: lead technical architect, head of the benchmark group (responsible for benchmarking and capacity planning), head of several development groups (database, client/server, web applications), lead technical architect, and head of MIS. In 1999 Kenan was purchased by Lucent Technologies for 1.48 Billion Dollars, and was sold in 2002 to CSG Systems.

From March 1999 through January 2000 I built an Advanced Development organization which focused on improving our products' user interfaces and technology architecture. We built a Java/CORBA Web user interface and provisioning front-end for our existing billing system, and substantially improved the ease of use of our interfaces. I also managed the Internet development team who enhanced our products to make them more suitable for deployments in the internet industry.

From May 1996 through July 1998 I managed Kenan's "Technical Infrastructure Group" which was responsible for providing information services for the 200+ employees in the Cambridge office. TIG provided support for a wide range of user activities on a variety of platforms including Windows, Windows NT, and 5 different varieties of Unix. During my tenure we moved from Novell to SMB, and migrated the entire company (including 4 remote offices) from cc:Mail to a standards-based email solution.

From March 1996 through March 1999 I managed the development of the core components (invoice calculation, API's, database schema, and Web UI's) of Arbor/BP. BP is a very complex, terabyte-scale product which has been deployed worldwide. Under my leadership the Cambridge-based development group grew from 7 to 31 people, and Kenan successfully deployed more than 50 systems, added support for Oracle RDBMS, added full multi-currency support (including support for the Euro), and scaled to more than 50 million billing events per day at our largest site.

Engineer / Manager

*GTECH Corporation, West Greenwich, RI
August 1986–February 1996*

From February 1994 through February 1996 I managed the development of GTECH's online transaction processing application. This is a multi-tier application with proprietary transaction processors, a database back end, and Windows PCs for user interface. It's capable of 24-hour operation in a Lottery environment and transaction rates in excess of 2000 per **second**. It is fully fault-tolerant, up to and including losing the entire primary data center to a natural or man-made disaster. I managed a 'Reference Application' in much the same way that I did when I was in my previous role in the embedded group. This was much more powerful, though, since it included all of the various clients and servers at the same version so they all played together.

From April 1991 through February 1994 I ran the 12-person group that developed applications for GTECH's Lottery point-of-sale terminals. The application drives the user interface, peripheral control, and communications for the terminal. Prepared budgets, project plans, and performance reviews, hired staff and (where necessary) carried out layoffs. Established the policies and procedures which allowed us to do development on the terminal application as a product rather than a set of individual applications.

From August 1986 through April 1991 I wrote firmware in C and assembly language (Z80, 64180, 8031, 68000) for GTECH's terminals and peripherals. I had increasing responsibility from Associate Engineer through Group Leader.

Education**BSE in Computer Science and Engineering**

University of Pennsylvania

Miscellany

Married, two kids, 1.5 cats.

Enjoy casual street and track motorcycle riding. I try to get off the bike only when it's standing still.

Giving back: GNU Project volunteer coordinator, Tecschange volunteer, CodePath panelist and mentor.