

SCOTT LAMB

Mountain View CA | (650) 335-5971 | slamb@slamb.org | www.slamb.org | github.com/scottlamb

SUMMARY

Hands-on tech lead (TL) of strong teams that build reliable, fast, secure backend systems.

EXPERIENCE

Staff Software Engineer, Google (Jun 2016–Jan 2021)

Senior Software Engineer, Google (May 2010–Jun 2016)

Software Engineer, Google (Jul 2007–May 2010)

Software Engineer (SWE) for 10 yrs, after 3 yrs as a Site Reliability Engineer (SRE). Highlights:

- Led ~6-person team within Identity Infrastructure SWE for 5 yrs, responsible for C++ servers in the critical path of every signed-in request at Google that handled credential issuance, credential, verification, and account lookup. We focused on security, reliability, latency, and efficiency. I advised on, formally reviewed, and authored designs and code for my subteam and the larger org. I set and documented coding and review standards, deployment procedures, and diagnostic techniques. I matched people with projects. I mentored teammates. I gave talks and led live debugging workshops.
- Completed Gmail storage migration, amid 3 yrs as a Gmail Backend SWE. I led a subteam of 3 in writing an improved migrator that allowed us to shut down old systems, delete millions of lines of C++, and end a feature freeze. Over 5+ yrs, many worked toward this moment. They gave me the honor of migrating the last user, and the whole Gmail team cheered.
- Led ~7-person Google Accounts SRE team for 2 years. We were responsible for all of what was later called Identity, including frontends and infrastructure. We ran a global Paxos database (see [older paper](#)). I trained teammates, proposed projects, gave technical advice, and liaised with other teams.
- Participated in primary on-call rotations for Gmail, Accounts, Workspace, Contacts, and Reader. I was an escalation contact for key systems. I set up automated white-box and black-box monitoring. I diagnosed production problems via tools and code inspection. I filed hundreds of production bugs.
- Consistently improved machine efficiency over full tenure. I wrote tools to manage my team's production quotas and job sizes, some of which found broader adoption and inspired later official tooling. I found CPUs "stranded" by a company-wide migration. I overhauled my software's threading model. I load tested and fixed problems that had prevented safely increasing my servers' CPU utilization. I found large opportunities in my own teams' CPU profiles and small opportunities in company-wide CPU profiles. I optimized code and mentored others as they did the same.
- Designed, reviewed, implemented, and ran software written in C++, Java, Python, and Go, including both critical path and management systems.
- Became a skilled user of generations of Google technology including [Borg](#) (internal analog of Kubernetes), [Chubby](#), [MapReduce](#), [Sawzall](#), [GFS](#), [Colossus](#), [Bigtable](#), [Spanner](#), [Dremel](#), [Borgmon](#), [Monarch](#), and [GWP](#).

SCOTT LAMB

Mountain View CA | (650) 335-5971 | slamb@slamb.org | www.slamb.org | github.com/scottlamb

Software Developer, 2Wire (May 2005–Jul 2007)

1 year each on 2Wire's DSL router ("HomePortal") and management server ("CMS"). Highlights:

- Developed new userspace code for the HomePortal, such as a feature to retain statistics gathered at frequent intervals for less frequent collection by CMS. I also adapted open source code to run in our unforgiving environment—no MMU, limited RAM and CPU, long uptimes.
- Built infrastructure for a new HomePortal white box testing group. I configured and enhanced Buildbot, a distributed continuous integration framework. I created a robust framework for flashing new code and launching commands over a serial console after every check-in.
- Replaced CMS's complex threaded handler for a proprietary protocol (OGMP) with svelte async design, doubling performance and eliminating many races & deadlocks.
- Overhauled CMS's load testing. Refactored OGMP tester to verify new implementation, reusing new protocol code. Replaced a broken 10,000-line tester for HTTP-based protocol (CWMP) with 500-line Python-Twisted design, used by QA to find performance and synchronization bugs.

Database Programmer, UI Hospitals & Clinics, Internal Medicine (Sep 2001–Mar 2005)

Half-time student developer on a 5-person team supporting Oracle database needs of 100+ active users. I gathered requirements, designed schemas, implemented Oracle Forms & Reports frontends, documented systems, and introduced version control.

OPEN SOURCE

Moonfire NVR, github.com/scottlamb/moonfire-nvr (2016–present)

Primary author of security camera network video recorder w/ 650+ GitHub ★s, 10 contributors.

- Crafted robust Rust/SQLite-based backend that supports 16+ 5MP RTSP cameras on inexpensive hardware such as the Raspberry Pi 4.
- Drafted Javascript/React-based frontend.
- Documented key aspects of design such as the novel database schema and time handling.
- Handled dozens of user support requests and bug reports.
- Factored out Rust library crates, including ones for RTSP ([retina](#)), HTTP Basic/Digest authentication ([http-auth](#)), and HTTP byte range handling & compression ([http-serve](#)).

EDUCATION

B.S., Computer Science, University of Iowa (Dec 2004)

- Physics Minor
- National Merit Scholarship
- Provost's Scholarship