

Sef Kloninger

John Josef Kloninger (full name)

sefklon@gmail.com
sef.kloninger.com @sefk
github.com/sefk

Highlights

Hands-on engineering manager who loves developing software and systems for the Internet. My expertise is building, scaling, and operating distributed systems. My roots are in backend/ops but I've dabbled in full stack. I've led teams of large teams 100+ at Akamai and VMware, and teams of 5-25 at Google, Ning, and Stanford. I built and launched an open-source online learning platform in eleven weeks. My 2012 [blog post](#) was the top post on Hacker News for a day. BS Computer Science, Stanford.

Work Experience

Google

Mountain View, CA

Engineering Manager

(9/2015-present)

Search. Managed teams delivering the largest and most used datasets in the Search Data Warehouse. Processed all user activity of billions of users of Search and Assistant. Delivered a new system to scale out datasets for every team in Search.

Google Cloud (2018-21). Managed development for [Cloud IAM](#) and [Firebase Rules](#). Built 1P and 3P authorization services for Google Cloud products. Fail-close dependency for every Cloud service, met stringent performance and availability SLO's. Lead teams of developers and managers teams of up to 25. Owned Google's [Common Expression Language \(CEL\)](#), a small embedded policy language for fast and safe evaluation.

YouTube (2015-18). Scaled up and added key features to the central Data Warehouse of telemetry from [over a billion active users](#). Owned systems providing big data processing services, collaborating across teams and locations. Ran thousands of ETL pipelines daily. Primary decision maker on warehouse operations, resource management, capacity planning, and security.

Wavefront

Palo Alto, CA

VP Engineering

(9/2014-3/2015)

Led a small team of engineers building a pre-launch service for monitoring infrastructure at scale. Developed backend features in Java and frontend features in JavaScript (Bootstrap, FlightJS, Require). Operated sizable Amazon Web Services deployment with Chef; cleaned up security groups and IAM roles. Cut infrastructure costs by 50%. Went on to be acquired by VMware.

Stanford Online

Palo Alto, CA

Head of Engineering, Open Source Platform

(6/2012-8/2014)

Engineering manager and developer on [Open edX](#), Stanford's open-source platform for Massive Open On-Line Courses (MOOC's). Senior technical member of the Stanford's new Online Education effort, developing strategy and presenting to senior leaders and trustees. Reported to the Vice Provost for Online Learning. Over Summer 2012 built and launched Class2Go, the first open-source MOOC platform. Ran successfully for 200k students until merged with edX in April 2013.

- **Engineering Manager:** Line managed small engineering team. Ran agile process with weekly sprints and demos; established code review and releases. Hired three engineers and many students. Set up [GitHub](#), [GitHub Issues](#), Huboard and Google Docs. Established *just enough* process.
- **Dev/Ops and Release:** architect of the run-time system built on AWS (EC2, S3, RDS, SES), ansible/chef, and celery/SQS. Lived and breathed Git. Managed all releases. Owned capacity and costs, fixed production and security issues.
- **Core Engineer:** core app features: queueing, OAuth integration, and operations tools.

Ning

Palo Alto, CA

VP Infrastructure Engineering and Operations

(12/2010-4/2012)

Operations. Oversaw the team running a medium-sized platform: 100K sites, 300M requests/day, 60M uniques/month, 240TB of user-generated content. Maintained four nines of availability. Established set of core technical metrics: incidents, availability, and performance. Built a mini-NOC. Defended against DDoS attacks by choosing protection service provider after build vs. buy analysis.

Infrastructure. Line managed senior engineers on the core Ning platform: relational and key/value content, blob storage, messaging, service discovery, mail, activity, and API's. ~50 RESTful Java services. Ran project to convert Oracle to sharded MySQL. Improved performance by restructuring JS and CSS and targeted rewrites. Responsible for development and support of the [Ning API](#).

Left shortly after Glam Media acquisition (12/2011).

VMware

Palo Alto, CA

Senior Director, Desktop Virtualization Engineering

(5/2008-12/2010)

Delivered server and client virtualization enterprise software to improve how corporate Windows desktops are deployed and managed. Hired to line-manage a small engineering team, grew to leading engineering for the Desktop Business Unit with 140 engineers in the US, UK, India, and China. Managed managers, directors, and three principal engineers (VMware's most senior technical role).

Delivered major releases that grew Desktop from \$10M to \$100M. Owned engineering for enterprise products, [VMware View](#) remote virtual desktops and [ThinApp](#) Windows application virtualization; personal virtualization with [Workstation](#) (Windows and Linux) and [Fusion](#) (Mac). Technical sponsor of two strategic transactions: acquisition of RTO Software for profile virtualization and OEM of Teradici's PCoIP remote display protocol.

Akamai Technologies

San Mateo, CA (2007-08)

Senior Director, Service Performance (2005-08)

Cambridge, MA (1999-07)

Founded the Service Performance group in Networks & Operations for advanced ops and high-touch customer engagements. Grew organization from six to 35 engineers and architects in US and India.

Improve service reliability by driving bulletproofing and scaling initiatives. Managed Akamai's incident response process. Define and measure core platform quality metrics, presented quarterly to CEO and Chief Scientist. Owned high-touch customer escalations and events, including: 2006 Olympics, March Madness, Microsoft patch Tuesdays, Hulu's launch, and Apple iPhone launch. Designed a unique cache hierarchy setup for long-tail social networking content.

Director and Senior Director, Engineering (1999-05)

Engineering Manager of 20-50 engineers and managers. Led technical hiring during company growth, then managed transitions and motivated teams during staff reductions. Mentor to other managers, senior engineers, and architects. Established processes for development, QA, and release engineering. Oversaw technical integration of two acquired companies.

Hands-on technical leader of Akamai's core technology. Handled many urgent service incidents; one of five crisis managers called in for the big problems.

- Communication infra.
- Load balancing
- Linux kernel
- Security systems
- Coherence protocols
- Operations tools
- HTTP and DNS
- Fault tolerance
- Performance

Senior Software Engineer (1999)

Lead engineer on HTTP content server. Scaled technology and features for rapid pre-IPO growth.

FarSight Financial Services

Cambridge, MA (1998-99)

Senior Performance Engineer

Enhanced performance of a website for retail on-line stock trading. Profiled systems, rewrote critical code paths (C++), and optimized database queries and schemas (Sybase).

Venturcom

Cambridge, MA (1996-98)

Principal Software Engineer

Principal architect, engineer, and team lead of Component Integrator, a tool for customizing and embedding Windows NT and CE. Product bought by Microsoft, later released as the [Windows Embedded Studio](#). Venturcom eventually became Ardenne, acquired by Citrix.

McKinsey & Company, IT

Cambridge, MA (1993-96)

Principal Software Engineer

Led development of client/server human resource system for internal use. Enhanced and optimized custom database replication system (Oracle). Attended week-long training sessions on communication skills, writing, and problem solving.

Veritas Software

Santa Clara, CA (1990-93)

Software Engineer (Part-Time)

Developed and tested Visual Administrator tool suite (C, Motif, Xt).

Activites

- 2014-2018: [Peninsula School](#) Board of Directors, Secretary and Executive Committee member.
 - 2016-current: [Bridge Foundry](#) Advisory Board member.
-

Patents

- Content Delivery Network (CDN) Content Server Request Mechanism With Metadata Framework Support ([US 7,240,100](#) and [US 8,122,102](#))
 - Internet content delivery service with third party cache interface support ([US 7,010,578](#))
 - Method and System For Providing Content Providers With Information About How Their Users Access the Internet ([WO 2002/017139](#))
 - Enterprise Content Delivery Network Having a Central Controller For Coordinating a Set Of Content Servers (pending, [US 2004/0073596](#))
-

Education

Stanford University, B.S. Computer Science, 1993

Research: "Verification of SCI Cache Coherence Protocol" under Prof. David Dill.

Nov 2022