

Christian Hergert <christian@hergert.me>

OBJECTIVE

To build world class developer tools for GNU/Linux and GNOME.

COMPUTER SKILLS

Languages: C, C#, Vala, \LaTeX , JavaScript, Python, Shell, SQL.

Server Software: Apache, Dovecot, Jabberd2, KVM, MongoDB, Qemu, Postfix, NFS, Nginx, PostgreSQL, Redis, Samba, Twisted, Xen, Squid and Varnish.

Operating Systems: GNU/Linux, FreeBSD, Solaris, AIX, OS X, Windows

FREE SOFTWARE

Individual Contributor 2002-current

GNOME Desktop Environment - OpenGL based desktop for Linux.

- Contributor to core Linux libraries such as **GLib** and **Gtk+**.
- **Builder**, a new IDE to simplify writing software for both GNOME and GNU/Linux.
- **Animation** engine for Gtk+.
- **Realtime graphing** solutions that are power and memory efficient.

Author 2014-current

Builder - An IDE for GNU/Linux

- Designed and implemented the foundation and features of Builder.
- More than 80,000 lines of GObject-based C code.
- Bindings for Python and JavaScript to allow scripting and extending Builder.
- Extensive language features for C/C++ in an editor that is lightning fast.
- Many upstream contributions to GLib and Gtk+.

Author 2013-2014

libmongoc, libbson - C library for communicating with MongoDB.

- High-performance, cross-platform MongoDB C driver.
- Bindings for Python, Ruby, and PHP which were considerably faster than competition.
- Works on GNU/Linux, FreeBSD, NetBSD, OpenBSD, Solaris 10+, AIX, Mac OS X, and Windows XP+.

Author 2011-current

GOM - Object database for GObject utilizing SQLite.

- Asynchronous library written in C allowing GObject to transparently serialize and deserialize an object graph to and from SQLite.
- Works from C and Python.

EXPERIENCE

Senior Software Engineer 2013-2014

MongoDB, Palo Alto, CA

- Author of the MongoDB high-performance, cross-platform C library (libbson and libmongoc).
- Mentor to team on writing scalable and maintainable software.
- Steered engineering staff on how to optimize software that communicates between language domains.
- Prototyped custom solutions for customers when things went wrong. Familiar with debugging file systems and on-disk formats.

- Public speaker on MongoDB technologies at meetups and notable conferences such as FOSDEM.

Senior Software Engineer 2011–2013
Catch.com, San Francisco, CA

- Wrote custom map/reduce engine to analyze MongoDB databases directly using the on-disk format. Faster and more memory-efficient than other solutions. Used for generating metrics to drive business decisions.
- Wrote and maintained API server, written using Python Twisted and my own MongoDB driver. Handles data synchronization to millions of mobile devices from a single machine.
- Designed and implemented mobile sync protocol.
- Designed and implemented wire protocols for push notifications to iOS and Android devices.
- Designed and implemented scalable HTTP server to notify web clients of changes to backend storage.
- Custom patches to Nginx.
- Scalable data design for MongoDB and Redis.

Member of Technical Staff 2009–2011
VMware, Inc., Palo Alto, CA

- Implemented SmartCard support for VMware View.
- Animation support for Linux desktop applications.
- PulseAudio support for Virtual Machines.
- Integration with commercial GStreamer products.
- Implemented custom mainloop that supports Microsoft Windows, Mac OS X, and Linux.
- Contributed to shared library that runs on desktop Linux, Mac OS X, iOS, Microsoft Windows, and Android to interact with remote desktops.
- Contributed to implementation of Windows DCOM wire protocol allowing Windows to manage a SmartCard attached to Linux-based thin-clients.
- Contributed to PCoIP remote desktop implementation on Linux.

Software Developer 2009
Unity 3D, Copenhagen, Denmark

- Helped port MonoDevelop to Mac OS X.
- Bug fixes for Gtk+ native quartz engine on Mac OS X.

Linux Software Engineer 2006–2009
MySpace.com, Beverly Hills, CA

- Ported MySpace .NET infrastructure to run on Mono and Linux, with potential annual savings to the company in millions of dollars.
- Realtime analysis of server metrics allowing many issues to be resolved without administrator intervention.
- Network monitoring of hundreds of thousands of switchports using a custom SNMP implementation supporting external MIBs.
- Custom network graphing and alerting using librrd.
- Indexing and analysis of hundreds of gigabytes of logs per day.

Linux Software Engineer 2005–2006
Medsphere Systems Inc., Mission Viejo, CA

- Managed operations for hardware deployed on customer sites.
- Designed and implemented VPN solutions used by customers to securely connect for remote administration.

- Contributor to Gtk# based desktop client used by doctors.
- Designed and maintained infrastructure services such as Asterisk PBX, Scalix email, SourceForge, munin, and OpenNMS.

Software Engineer

2003-2005

Mosaix Communications, Seattle, WA

- Designed and built a virtual router solution running on commodity x86 servers using Linux and both User-mode-linux and Xen. This saved the company hundreds of thousands of dollars per point-of-presence.
- Designed and implemented a network management solution to allow customers to remotely manage network configuration, iptables based firewalls, SMTP spam and virus detection, IPSec, and provisioning of new networks.
- Designed and managed data-center deployments.
- Network monitoring of internal, external, and customer premise hardware.
- Used Linux based hardware to terminate T-1, DS-3, and various cell-switched protocols.
- Frequent use of Linux, PostgreSQL, PHP, Mono, and other free software solutions.

LIFE EXPERIENCE

I began programming at age 12 using RedHat Linux. From that day forward I was hooked on the ability to make my imagination reality. Required to become personally and financially independent immediately upon graduation from High School, I began my career and have devoted substantial personal as well as professional time to programming and to the Free Software community.

I've gained valuable skills from many thousands of hours of practical industry experience. I've studied algorithms, language design, and operating system design on my own time and make practical use of that knowledge every day. I excel at conceptualizing the overall architecture of software, and then executing that vision with the creation of code that is both efficient and elegant.

Now, in Free Software fashion, I give back to the next generation by building tools to enable new contributors to develop Free Software.