Summary

Engineer with 30 years of experience in full-stack software-development on Linux and legacy unix platforms, embedded systems, DevOps, IT, security, networks and web development; seeking challenges.

Joshua Judson Rosen
19 Pierce Street
Nashua, NH 03060
telephone: (603) 318-5093
e-mail: rozzin@gnu.org

Skills

Languages:

C, C++, Perl, Python, Pike, TCL/Tk, Scheme, Common Lisp, Emacs Lisp, ECMAScript/LiveScript/JavaScript, unix shell (Bourne sh, bash, csh, awk, sed), Logo, Java, ColdC, MOO, SGML, XML, HTML, CSS, SQL, PHP, Pascal, BASIC

Tools & toolkits:

Apache, Roxen, lighttpd, Jabber/XMPP, OpenVPN, Active Port Forwarder, SEC, BIND, ISC DHCP, INN, PAM, CVS, RCS, Subversion, Bazaar, Git, Autoconf, Automake, Libtool, Make, Samba, GLib, GTK+, GObject, wxWindows/wxWidgets, MySQL, PostgreSQL, SQLite, Sybase, Bugzilla, Launchpad, Redmine, Net-SNMP, Scotty, Exim, Postfix, SSH, OpenSSL, PulseAudio, GConf, GPSD & libgps, StatusNet & GNU social, BusyBox, Xenomai, GStreamer, Aravis, ChibiOS, ArduPilot, MAVLink, MAVProxy, CHDK & chdkptp

Technologies & concepts:

Embedded Linux, open source, DevOps, POSIX, portability, multithreading, parallel processing, realtime, high-performance computing, distributed processing, event-systems, metaprogramming, parsers, robotics, wiki, usenet, NNTP, SMTP, IMAP, POP, HTTP, SSL, PGP/GnuPG, x.509, PKI, PKCS, VPN, DNS, mDNS, DNS-SD, ZeroConf, NIS/YP, NSS, TCP/IP, UDP, client/server, mesh networking, DHCP, PPP, routing & firewalling, NAT, SNMP, SMI, MIB, ASN.1, LDAP, PCI, PXE, RAID, RTP, GPS, GIS, Mapping/Cartography, OpenStreetMap, USB, topological sorting, heuristics, Lambda Calculus, UI/UX, internationalization (i18n) & translation/localisation (l10n), revision-control & branch-management, issue-tracking, systems integration, release-management, software-packaging, RPM, dpkg, deb

Operating systems:

GNU; Linux 2.0, 2.2, 2.4, 2.6, 3.x; Ubuntu, Debian, RedHat, LinuxPPC; OpenEmbedded, Ångström, OpenWrt; FreeBSD; Solaris; Irix

Employment history & notable accomplishments

2016–2026: Group Lead, Principal Embedded & Software Systems Engineer at GreenSight (Boston, MA)

- Head of Software development; managed a diverse set of software developers ranging from interns to highly senior engineers across various domains including embedded systems, robotics and control systems, Linux applications and systems administration, and web development.
- Responsible for defining software development, version-control, packaging, release, and distribution policies / best practices and providing training; providing performance reviews and career guidance
- Lead release-management team; responsible for defining release-management policies and procedures, including software packaging policies; administered APT repositories for Ubuntu, Debian, and Raspberry Pi OS across amd64, arm64, and armhf architectures; ensured integrity of software distribution channels by managing GnuPG APT signing keyring.
- Lead embedded systems software development, integration, and release, taking GreenSight's Drone Operating System from an early rapid-prototype stage to a cohesively integrated, fully-functional data-collection appliance with robust and secure mobile connectivity, with units operating at dozens of production customer sites with each drone capable of flying multiple productive missions per day.
- Developed firmware, drivers, and client toolkit for the GreenSight thermal camera.
- Led development of a stack of multi-camera control libraries for reliable management of clusters of Canon cameras based on CHDK using Python and Lua, other cameras supported by GStreamer using Python, and specialty thermal cameras using Python and C
- Developed libraries for driving various devices on Linux and ArduPilot, including smart battery systems and various embedded sensors.
- Developed and maintained systems to provision new product units and perform automated and semi-automated tests during manufacture, repair/refurbishment, and field diagnostics.
- Worked closely with manufacturing, mechanical and electrical engineers, and pilots to ensure that the thermal camera firmware, multi-camera control subsystems, realtime telemetry systems, and autopilot systems all consistently have high-quality releases, with efficient and reliable upgrade-systems to continually improve the user/operator experience.
- Worked closely with customer IT resources, to ensure smooth bringup of new sites, and to quickly identify and resolve--and prevent--networking issues at customer sites that could impact either the efficacy or the safety of drone operations at those sites.
- Reverse engineered control protocols for OTS components, e.g. LTE modem status/control protocols, Sony IMX camera sensor controls
- Designed, implemented and administered VPN connectivity system for remote management of field equipment, including facilities for managing multiple distinct levels of access via a set of x.509 certificate authorities Developed systems for credentialing both products and personnel to allow remote access while protecting the security of the drone fleet, the company, and customers.
- Administered Jira and Redmine issue-tracking systems; defined standard workflows, schemas, and project-creation/-management procedures.

2011–2016: Senior Software Systems & DevOps Engineer at Harvest Automation (Billerica, MA)

- Responsible for development and maintenance of Linux-based operating systems for Harvest's HV-100 and TM-100 robots and peripheral devices. Responsible for all software integration, build, and release processes.
- Responsible for network and server infrastructure at main office, manufacturing, test facilities, and cloud services–including responsibilities for IP subnet partitioning, DNS, DHCP, Wi-Fi, and a multitude of physical and virtual servers.
- Identified open-source software components for development of robot vision, data-management, and other systems; modified open-source software to add features and resolve other issues to meet needs of product development; reduced technical debt by working with upstream authors to integrate changes; ensured license compliance.

- Created web-based mobile UI for the human element of the TM-100 warehouse-automation system, allowing the warehouse control system and robot fleet control system to perform real-time coordination of robotic and human-based tasks.
- Reduced the overhead of field integration and service, and increased visibility into fielded HV-100 systems, by developing an inexpensive, user-friendly data-gathering appliance and accompanying cloud services to collect accumulated logs from robots, transmit them back to Harvest for ongoing analysis, and aggregate them in a convenient archive.
- Simplified customer robot-management and site-management by creating cloud services and client software to manage site-wide deployment operational and configuration data on fleets of autonomous robots.
- Reduced the cost of customer support with a flexible, secure remote support login system for fielded systems.
- Specified and implemented a comprehensive logging framework used throughout Harvest's codebase, including a migration-strategy involving formal deprecations of old APIs and gradual transition to the new APIs to accommodate the team's preexisting workload.
- Defined server system architecture for high throughput, reliability, and scalability; bootstrapped servers using Xen and LVM with software RAID storage.
- Maintained robot UI software framework. Reduced UI translation burden by porting from a proprietary translation infrastructure to standard GNU gettext.
- Developed major user-facing features in time to fill urgent customer needs, including new robot behaviors.
- Streamlined development by migrating from legacy Subversion revision-control systems to Bazaar and Git; trained engineers on best practices and effective use of revision control tools.
- Identified limitations of existing network architecture; planned and carried out a complete overhaul of the corporate IPv4 network to improve reliability, scalability, and general future-proofness; worked with the American Registry of Internet Numbers to acquire a sufficiently-sized routable IPv4 subnet despite the exhaustion of the IPv4 address pool.
- Developed network protocols and UIs for efficient, robust bulk data transfer and realtime wireless robot status/control and telemetry.
- Reduced cost and setup burdens of on-site TM-100 robot system demos by developing a robust but inexpensive, portable, battery-operated Wi-Fi network with 802.11s mesh backhaul.

2008–2011: Advanced Research Engineer at MIT Lincoln Laboratory (Lexington, MA)

- Developed advanced tools to process radar imaging data and to automatically assess and report on situations in the Earth's orbital space via signal-analysis, astrodynamic and other heuristics.
- Performed optimisation and maintenance of parallel-processing and distributed-processing algorithms.
- Led the effort to port existing radar-processing code from Irix/MIPS to Linux/x86-64.
- Designed and implemented a family of portable libraries for parsing, analysis, and interchange of radar data on Irix, GNU/Linux, and Solaris on both big-endian and little-endian architectures.
- Modified upstream open-source software to better meet the needs of research engineers and scientists; worked with upstream authors to integrate changes.
- Reduced development overhead by refactoring large, monolithic projects into individually-releasable modular components and migrating from platform-specific, legacy build-systems to the GNU Autotools.
- Developed data-models and databases using PostgreSQL and Sybase.
- Managed migration from legacy version-control systems like ClearCase and CVS toward Bazaar; developed automated VCS-migration & -integration tools; educated engineers and scientists on version-control workflows.

2006–2008: Software Engineer III at Russound (Newmarket, NH)

- Project Lead on version 3.0 of Russound's Smart Media Server (SMS3): coordinated a geographically-distributed team of developers, provided schedule-estimates, prepared release-candidates and upgrades, and managed the public repository of software-updates. Fixed numerous major bugs in a complex Perl & C++ codebase with dynamically-generated JavaScript and SQL.
- Resolved long-standing issues of audio signal contamination from processor power-state transitions: root cause analysis involved tracing through Linux kernel power-management subsystems, resolution involved identifying and disabling several (normally-useful) power-saving features at fault.

- Developed and integrated embedded Linux software for a ACA-E5, a small ARM-based system with NAND flash storage, using OpenEmbedded & Angstrom; developed software packages capable of automatic, network upgrade.
- Developed hierarchical state machines to manage interactions between a multitude of alarm conditions.
- Modified open-source software to add missing features and resolve other issues.
- Developed automated release-tools to speed delivery of updates and improve their reliability.
- Designed and implemented a secure, user-friendly remote-support system for use by worldwide dealers and tech. support: the system used SSL PKI and included support for wide-area ZeroConf networking.
- Added x.509 certificate-based authentication & access-control to Active Port Forwarder (APF); worked with upstream/third-party maintainers to integrate changes.
- Developed factory infrastructure necessary to ship secure PKI infrastructure in embedded systems. Worked with IT to support deployment. Educated engineering teams about Public Key Infrastructure.

2005–2006: Engineering Technical Specialist at ADC (Nashua, NH; following acquisition of OpenCell Corp. by ADC)

- Designed, specified, and coded a reference implementation for an extensible client/server system for remote-control and analysis of SONET and similar fiber-interface hardware; worked with a team distributed between New Hampshire, Minnesota, and Bangalore to produce other back-end implementations.
- Developed Linux PCI drivers for custom I^{2}C-bus, UART, and fiber-interface cards.
- Developed a `stackable' driver-architecture to minimise driver-integration issues.
- Reduced system complexity by designing and implementing an SNMP event-correlator module with pluggable and chainable rulesets and SQL logging.
- Developed web-based tools for network-management and system diagnostics.

2003–2005: Software Engineer at OpenCell Corp. (Nashua, NH; following spin-off of OpenCell Corp. from Transcept)

- Developed software-release standards and procedures, including software-upgrade mechanisms; oversaw the release-cycle.
- Accelerated development and release of software patches by developing comprehensive branch-management policies and procedures.
- Defined standards for integration of custom software into base OS, ensuring a seamless user-experience.
- Developed SNMP trap-emission scheme and trap-management libraries for C and C++.
- Authored SNMP MIBs.
- Developed SQL databases for tracking SNMP events, system fault-conditions, and network-configuration.
- Provided engineering support to customers.
- Improved product reliability by designing and implementing SNMP- and ICMP-based network watchdog system to automatically detect and correct problems across multiple network layers.
- Developed environmental-chamber tests for hard disks and flash-memory, and a software solution to limit writes to media based on optimal batching and postponement of writes.
- Designed, implemented, and deployed UDP-based automatic network node-registry system with support for update-notification and automatic management of DNS and NIS/YP.
- Developed a centralised automatic build-server & module-level release-system providing a set of build-environments that expanded upon the run-time environments used in products.
    - The actions performed by this system include: analyze changelogs, automatically generate & apply release-tags to software, build & validate source-code snapshots, transfer source-code to the secure build-server via an FTP spool, build the source into discrete binary modules, post the resultant build-logs to a company-interal usenet-server , and (following a successful build) make the binary module available for download by automatic system-upgrade utilities.
- Reduced development-costs by deploying a centralised development-server hosting multiple development-/build-environments with SSH, telnet, VNC, and XDMCP access, giving developers the benefits of self-hosting environments without the expense of individual workstations. The self-contained nature of this environment allowed it to be deployed across several host platforms for wider-scale use.
- Improved inter-office connectivity by building and deploying an SSL-based end-to-end VPN system; managed the SSL certificate authority.

2000–2003: Software Engineer & Linux Systems Administrator at Transcept (Manchester, NH)

- Managed configuration and building of Linux kernels and filesystems for embedded PowerPC products with Compact Flash storage; integrated custom patches.
- Improved visibility into systems by developing a modular, graphical, SNMP-based network-management system with real-time information-display, written in C++.
- Expanded automated-test coverage by developing an extensible graphical shell for management of automatic tests and real-time generation of reports on both desktop and embedded Linux systems.
- Shortened MIB development-cycle by automating the generation of extension-modules for the Net-SNMP agent, with automatic generation of interfaces between those modules and other processes.
- Determined present and future requirements for best practices and continually-improving development, and specified tools necessary to fulfill those requirements:
  - Specified base Operating Systems for new flagship products.
  - Specified system development-tools.
  - Developed self-hosting software-development environments.
- Developed production/manufacturing and software-release tools for capture and deployment (cloning) of hard-disk and flash-memory images via an automated PXE-based network-boot system with a menu-driven user-interface.
- Developed a framework for deployment of prototype networks using routing and DNS zone-forwarding.
- Established a revision-control regime using CVS on Windows and Linux; administered CVS.
- Headed employee-education program on principles and use of revision-management and CVS.
- Boosted productivity by implementing a custom WWW interface for browsing, searching, and reading revision-controlled documents in HTML, PDF, MS Word, and other formats.

1998–2000: Computer Technician at Souhegan Enterprises (Amherst, NH)

- Assembly, setup, and service of personal computers.

1997: Java Programmer for The Concord Consortium (Concord, MA)

- Developed graphing software.

Independent & open-source projects/accomplishments

- In addition to being a lead developer of multiple open-source projects, I contribute on an as-needed basis to various open-source projects including Busybox, GNU social, GJay, Active Port Forwarder, and others; see a list of my upstreamed contributions at openhub.net.
- Lead/Maintainer since 2017 of GNU Robots, a programming game where you construct a program in Scheme or JavaScript for a little robot, then watch him explore a virtual world. The world is filled with baddies that can hurt you, objects that you can bump into, and food that you can eat. The goal of the game is to program the robot to collect as many prizes as possible before it is killed by a baddie or it runs out of energy.
- Project Lead of FoxtrotGPS, a user-friendly GPS, GIS, and mapping application for GNU/Linux and other unix-like systems on GNU/Linux tablet computers and other resource-constrained and embedded devices. FoxtrotGPS is shipped as part of Ubuntu, Debian, Gentoo, the FreeBSD Ports Collection, and other major OS distributions.
- Project Lead on libvisualid, a cross-platform graphics toolkit for efficient, automatic generation of distinctive, highly-recognisable iconography; a reference implementation of VisualIDs.

- Implemented support for VisualIDs as file-icons in the Nautilus file-manager, and as point-of-interest (POI) markers in FoxtrotGPS.
- Designed and implemented a whole-home audio system with wireless remote control using MPD and Remuco; intelligent, automatic playlist-management based on sonic analysis using GJay; and seemless digital audio-distribution via RTP using PulseAudio.
- Ported GJay to OpenWrt for use on Qi Hardware's NanoNote and in other embedded environments.
- Administration of several social websites using StatusNet and GNU social, including community and corporate services.
- Enhancements and bug-fixes to Orrery, an astronomical tool for smartphones, tablet computers, and other resource-constrained and embedded devices—contributions included a rewrite of the build-system for portability using the GNU Autotools, and correct parsing of numeric data for I/O in international contexts.
- Designed and implemented cross-platform, graphical e-mail client software in Python using wxWidgets.
- Designed and implemented cross-platform, graphical game software in Python using Tk.
- Designed and implemented conversational `artificial stupidity robots' in ColdC.
- Startup, administration, maintenance, and lead development of several online multiuser virtual environments (`MUVEs') including MOOs, Colds, and MOOP.
- Algorithmic 3-D artwork using the POV-Ray ray-tracer's scene-description language.
- Mapping for OpenStreetMap.
- Coordinator of the New England regional OpenStreetMap birthday party starting in 2014.

References

Excellent references, which can be made available upon request.