

Jon Mayo / Software Engineer

I am a software engineer with over ten years experience developing embedded products running Linux, QNX and vxWorks. I have worked on everything from communications to storage systems, from consumer products to enterprise servers. I desire to work with talented people on products that strive to achieve the best quality possible.

I am experienced at developing UNIX drivers and applications (Linux, BSD, Solaris), embedded systems and networking appliances. I have also used many different processor architectures, including PowerPC, MIPS, ARM, dual core ARM/Cortex-A9, multi-processor x86/i386 and x86-64/AMD64.

Jon Mayo
970 Meridian Ave Apt 38
San Jose, CA 95126
Home: (408) 647-1369
Cell Ph: (408) 216-3226
jon.mayo@gmail.com

EXPERTISE

- Mastery of the C Programming Language. (standard ISO 9899-1990 and 9899-1999 specifications.)
- Over 15 years programming experience. (10+ years as a professional software engineer)
- Experienced in assembly language for i386, x86-64/AMD64/EMT64T, ARM, ARM Thumb, MIPS, and PowerPC.
- Nvidia Tegra 2 AP20 development of both Ubuntu/Debian Linux and Android.
- Ubuntu/Debian packaging system (.deb) using dpkg and apt.
- X11/Xorg core development. XInput drivers, configuration, Xlib and XCB based applications and window manager development.
- Multitouch touch panel driver and gesture development.
- Knowledge of digital and analog electronics. (2 year training course)
- Operation and software development on UNIX and clones: Linux (i386, x86-64/AMD64, MIPS, PowerPC, ARM/Cortex-A9), MacOS X, FreeBSD, NetBSD, OpenBSD, SunOS/Solaris, AIX.
- Source control tools: Git, CVS, Perforce, and Subversion.
- Bug tracking systems: Bugzilla, Jira, TRAC, and others.
- Debugging with JTAG and serial port (remote gdb - GNU Debugger).
- Modifying, porting and extending the Linux kernel (nfsd, scheduler, PCI subsystem, memory manager, TCP/IP, iptables/NetFilter, boot/initialization, multitouch devices and more) to change behavior, fix bugs or improve performance on different architectures (ARM, x86_64, PowerPC, MIPS, i386).
- Drivers for vxWorks, QNX, Linux, DOS, u-boot and boot-loaders.
- CGI/FastCGI web programming in C for embedded devices (vxWorks, Linux)
- Client-Server and Peer-to-Peer networking application implementation and design
- Small custom webserver with built-in TCL scripting using a light weight SQL database (SQLite)
- Software design documentation (troff, html, DocBook)
- u-boot development: driver, diagnostics and new features.
- ANSI FORTH and specialized version such as Open Firmware, pforth and CM Forth.

OPEN SOURCE EXPERIENCE

- Stack-based scripting language that integrates easily with any C program. (Forth clone)
- Custom bootstrap and kernel (picokernel) with non-preemptive scheduler and simple memory management. Designed for embedded devices.
- Abstracted a network server library that could use the following transparently: select() with 3000 connections, kqueue/kevent with nearly 100000 connections, sigtimedwait (POSIX equivalent to kqueue/kevent).
- Three different online text-based adventure/role-playing game servers (also known as a MUD).
- A small limited user-space threading library for POSIX and Windows systems. Allows the creation of thousands of threads without having an impact on system performance. The limitation is that the threads are cooperative, the purpose is to eliminate complex state machines and use an execution context. (uses the POSIX calls setcontext/swapcontext/getcontext) and Win32 CreateFiber/SwitchToFiber.

WORK HISTORY

October 2009 – May 2010:

System Software Lead
Kakai, Inc
5155 Old Ironsides Drive
Santa Clara, CA 95054
(408) 300-8649

Job Description

Led a Linux platform software team to implement device drivers, services and core X11 components.

October 2007 – October 2009:

Senior Software Developer
Lab126, Inc.
a division of Amazon.com
20450 Stevens Creek Blvd.
Suite 400
Cupertino, CA 95014
Phone: 408-790-6400

Job Description

Platform developer for the Amazon Kindle e-book reader. Linux system development, build tools, bootloader development.

June 2007 – October 2007:

Senior Member of Technical Staff (MTS)
VMware, Inc.
3401 Hillview Ave
Palo Alto, CA 94304
(650) 475-5000

Job Description

Developer for ESX Server product, a virtualization product for x86 platform running a custom operating system kernel with a Linux management console.

January 2007 – May 2007:

Senior Software Engineer
PacketMotion, Inc.
110 Baytech Drive, 2nd Floor
San Jose, CA 95134
(408)449-4300

Job Description

Create a new Linux based platform environment for a multiprocessor, clustered networking appliance for deep level packet inspection.

April 2005 – January 2007:

Linux Kernel Developer.
Cisco Systems, Inc
170 West Tasman Dr
San Jose, CA 95134
(408)526-4000

Job Description

Maintain existing open source and proprietary patches for the Linux 2.6 kernel, as well as develop drivers for back plane protocols. Provided engineering support for about sixty open source applications, utilities and development tools. Automated GNU cross compiler builds for i386, MIPS64 and PowerPC.

Modifications to u-boot bootloader to support custom hardware, and enhanced with features for USB booting.

Tools

- GNU C Compiler (gcc): Intel x86, MIPS64 (Cavium and Broadcom SiByte), PowerPC(Freescale).

April 2004 – April 2005:

Linux Kernel Developer / QA Developer
Agami Systems, Inc.
330 Gibraltar Dr
Sunnyvale, CA
(408) 349-0400

Kernel Job Description

Linux kernel developer for an enterprise NAS product. Applied patches from the linux community to fix bugs, including back-porting patched for newer kernel version to the NAS's older kernel. Added a proc interface for statistics collection. Added ioctls to control nfs server behavior. Implemented 64-bit atomic operations for i386. Tracked down and fixed many bugs.

Kernel Tools

- kgdb (kernel debugger)
- GNU C Compiler: Intel x86 and AMD 64-bit architectures (x86_64)

QA Job Description

Test developer for an enterprise NAS (storage) product. Implemented and maintained low-level data path performance and functionality tests in C, shell script, expect and TCL. Debugging of Linux kernel and custom kernel modules using kgdb.

QA Tools

- Scripting languages: expect, TCL, shell script
- Source control tools: Perforce, CVS
- Bug and test tracking systems: bugzilla, testlink
- Test automation harness: STAF

August 2002 – September 2003:

Staff Engineer
Pillar Data Systems
2840 Junction Ave
San Jose, CA 95134
(408)503-4000

Job Description

Software Engineer developing high availability NAS and SAN storage products running QNX and Linux.

Responsibilities

- Performed performance evaluation of Xeon2 and Pentium-4 architectures with Linux and QNX kernels. (for SMP and uniprocessor)
- Designed and developed tracing and debugging tools as a kernel service to allow postmortem diagnostics of the interaction between multiple servers. Required the development of specialized atomic counters for SMP systems.
- Redesigned system CORE dumps to remove certain bulky data sections and to be compressed directly into battery-backed memory.
- Worked on a project to replace QNX kernel messaging interface with a zero-copy interface to eliminate the major bottleneck in performance on QNX. (messaging is used like system calls in QNX)
- Used the IO APIC(Intel Xeon/P4) for sending interrupts/messages between processor. This allows messages to be sent to another processor or broadcast to all processors. This is necessary for effective use of SMP.

Tools

- Source control tools: Perforce, CVS
- GNU Compiler and related tools (gcc, gdb, ld scripts, objdump)

April 2002 – August 2002:

IT Contractor / Software Developer.
 Pioneer Research of America
 2833 Junction Ave # 100
 San Jose, CA
 (408) 437-1800

IT Responsibilities

- Maintain a network of MacOS X and Linux workstations
- Changed network over to use NIS+ (aka ypbind)
- Installation of servers, including a video streaming server and RAID storage server.

Developer Responsibilities

- Implemented the initial EPG(Electronic Program Guide) software for StingRay. (a set-top box personal video recorder)

August 1999 – August 2001:

Embedded Software Engineer.
 DIRECTV Broadband, Inc. (formerly Telocity, Inc)
 Cupertino, CA

Job Description

Embedded Software Engineer for vxWorks and embedded Linux. (Residential DSL Modem / Router / Firewall) Responsible for maintenance and development of various

applications on the DSL Gateway including: DHCP server, HTTP server, software download protocol, project Makefiles, CGIs for HTTP server, bug fixes in nearly any part of the product (application and driver layer), design documentation for new products.

Responsibilities

- Created a protocol specification to allow for remote firmware updates to the DSL modem. This gave the company the power to repair issues without the expense of reissuing a new modem to all customers. Saving the company nearly a million dollars in the first year it was used.
- Managed revisions of the company documentation for the protocol specifications
- Coordinated meetings with other departments to review new protocol specifications.
- Developed testing procedures to be used by QA department.
- One of the main experts on all of the company's products.
- Linux kernel development. Ported Linux to a brand new flavor of MIPS. Worked on TCP/IP subsystem and iptables/NetFilter. Linux kernel drivers and internals.

Tools

- WindRiver Tornado, an integrated debugger, monitor, and shell for embedded systems running vxWorks.
- Debuggers: gdb/ddd and brief experience with SoftICE.
- GNU and BSD Make (Have created very powerful, modular Makefiles for large projects).
- Bourne Shell (sh/bash) Scripts to automate testing and other often repeated tasks.
- Source control tools: CVS, VSS
- ICE (in-circuit emulator) for PowerPC 850 and 860.
- Configuring and compiling GCC as a cross-compiler. (MIPS, PowerPC)