

# David Simmons | Résumé

5499 Zephyr Ct, Arvada, CO 80002

☎ +1 720-552-8086    ✉ simmons@davidsimmons.com

🌐 <https://www.davidsimmons.com/>    🌐 simmons

## Objective

---

I'm looking for new opportunities principally in the area of low-level systems programming, where my background and expertise offers the most value, while also being able to offer my skills in other areas (for example, web app development) as the need arises.

## Experience

---

### ○ Sparco Media

*Partner / Software Engineer*

**Denver, CO**

*September 2005–present*

As a partner at Sparco Media, I provide custom software development services to our customers in the consumer electronics industry. This usually involves a lot of C coding, device drivers / Linux kernel hacking, audio/video programming, and other software skills that help our clients participate in the ongoing consumer electronics revolution. Some examples of work I performed for customers include:

- Linux device drivers, including supporting an HDMI receiver chip (for Intel), a no-interrupt eMMC driver for storing panic dumps (for Verizon), and numerous kernel modules to support diagnostics.
- Linux systems programming to provide software update infrastructure (Verizon), OpenGL ES graphics demos (Intel), and manufacturing tests (Verizon).
- Android customization work for manufacturers, including an advanced enterprise security framework (Intel) and adapting AOSP builds to support television devices (Verizon). Also, I performed Android application work to support management of wireless speaker systems (Summit Semiconductor).
- Developed a Java/Spring web app to collect internal orders for developer components, forecast future demand, and schedule manufacturing (Intel).

### ○ Intel Consumer Electronics Group

*Software Engineer (contracts)*

**Beaverton, OR (remote from Denver, CO)**

*June 2003–December 2003, July 2004–June 2005*

- Developed an SDL-based graphics driver to support the 2D graphics acceleration of Intel's custom video hardware, allowing customers to develop visual experiences that seamlessly blend graphics and video. This driver was used to good effect in the Toshiba HD-A1, the world's first HD-DVD player.
- Made modifications to the Intel iSCSI reference implementation to support the wider set of SCSI commands needed for media playback and key negotiation, thus allowing music CD's and DVD's to be played over the network.
- Developed a graphical user interface for television devices.
- Implemented an XML-based message passing library for communication between backend multimedia components.
- Implemented a build and assembly system for deploying customized Linux distributions.

## Skills

---

- **Programming Languages:** Rust, C, C++, Assembly, Java, Python, Perl
- **Programming concepts:** UNIX systems programming, Linux device drivers, Network protocols, Digital audio and video, Security best practices, Android development, Web applications, Documentation
- **Industry software:** Linux, Standard GNU and UNIX tools, Vim, Apache, Nginx, Eclipse, Git, PostgreSQL, MySQL

## Research interests

---

Like many experienced practitioners of software engineering, I spend significant time outside of work hours improving my skills and thinking about how our profession can deliver better products and services. Among the areas that are of particular interest to me are:

- **Secure and reliable software.** Conventional software engineering techniques have failed to meet expectations for security and reliability, leaving our systems vulnerable and companies exposed to costly liability issues. I've spent considerable time becoming proficient in the Rust programming language, which I hope will eventually supersede C and C++ for high-performance systems programming, as it greatly reduces such risks. I also keep a keen eye on ideas for improving the software development process and techniques for assuring software integrity (e.g. reproducible builds).
- **Advanced networking.** We've lately seen a renaissance in network protocols, as recently developed schemes such as HTTP/2, WebRTC, and QUIC provide new ways for our systems to communicate with each other. A pet project of mine is the development of a pure-Rust implementation of SCTP, the underlying transport protocol of WebRTC data channels, as a building block for peer-to-peer applications based on open standards.
- **Application of cryptography.** I am not a cryptographer, but I do try to follow the latest news in the field and look for opportunities to use well-tested cryptosystem implementations to improve security and privacy for my users.