Pete Wilcox

pete@pcwilcox.com (510) 393–4074
http://www.pcwilcox.com github/LinkedIn: pcwilcox

Education

- **University of California, Santa Cruz** Santa Cruz, CA
  Doctor of Philosophy in Computer Engineering; GPA: 3.75 *Sep. 2019 — Present*
  - **Honors**: Eugene Cota-Robles Fellowship
- **University of California, Santa Cruz** Santa Cruz, CA
  Bachelor of Science in Computer Science; GPA: 3.98 *Sep. 2017 — Jun 2019*
  - **Honors**: Phi Beta Kappa, *summa cum laude*, Dean's List

Experience

- **University of California, Santa Cruz** Santa Cruz, CA
  Graduate Student Researcher *Sep. 2019 — Present*
  - **Storage Accelerator**: Developed computational storage device simulator in C using SPDK and QEMU. Ongoing research in collaboration with UCSC Genomics Institute to develop computational storage devices in order to accelerate genome sequence alignment.
  - **System Administrator**: Developed best practices guide for research group server cluster. Provide ongoing maintenance and sysadmin for servers running Linux OS on academic networks. Perform needed hardware maintenance and upgrades on Dell and Tyan rack-mounted servers as well as consumer desktop computers.
- **TidalScale** Los Gatos, CA
  Kernel Engineer Intern *Summer 2019*
  - **Functional Hypervisor Testing**: Designed and built a functional hypervisor test facility for verifying model-specific register implementation in the hyperkernel. Implemented system as a custom Linux kernel module and device driver, written in C and Python.
  - **Model-Specific Register Support**: Utilized hypervisor test mechanism to implement and verify emulation for model-specific registers in the TidalScale hyperkernel.
  - **Virtual CPU Migrations**: Optimized virtual CPU migration algorithms and data structures in order to reduce migration packet size, improve code readability, and optimize maintainability.
  - **VMCS Compatibility**: Developed kernel mechanism for verifying compatibility of virtual machine control structures between servers in a TidalScale cluster.

  Software Engineer Intern *Summer 2018*
  - **Server Management Tools**: Developed management, migration, and deployment tools for WaveRunner servers.
  - **WaveRunner**: Worked with large code base in C and Go to identify and eliminate bugs. Implemented version feature enhancement and provided support for internal NAS feature.
  - **Server and Network Admin**: Configured and deployed WaveRunner server clusters; setup and administered network infrastructure. Collaborated with sales engineers to support customer installations.
  - **Documentation**: Developed and documented best practices for TidalScale Admin Guide.
- **University of California, Santa Cruz** Santa Cruz, CA
  Assembly Programming Tutor *Jan. 2018 — Jun. 2018*
  - **Individual Tutor**: Provide direct assistance to undergraduate students in MIPS assembly and logic circuit design.
  - **Group Tutor**: Demonstrate best practices to beginning students in lab sections by reinforcing lecture topics.

Projects

- **Relational Database**: Page-oriented relational database management system using the underlying OS filesystem in order to implement page file, relation, index, and query execution modules. Built in C++ for Database Systems course.
- **Distributed Key-Value Store**: Distributed, fault-tolerant, in-memory key-value store using Docker containers, providing guarantees of eventual consistency, availability for writes, and large storage capacity. Allows easy modification of system configuration at runtime via REST API. Implements full unit test coverage and continuous integration using CircleCI. Developed using Go, Python, and Docker for Distributed Systems course.
- **OC Compiler**: A compiler for the C-like language OC for Compiler Design course. Developed scanner using Flex and parser using Bison. Implemented string tables, symbol tables, and syntax tree generators in C++.
- **Popper**: Open-source CLI tool to make devops tools and workflows more accessible to academic research. Designed and implemented remote badge service feature using Go, Python, Docker, and shell scripts.

Programming Skills

- **Languages**: C, C++, Python, Go, Java, Assembly
- **Technologies**: SPDK, QEMU, FreeBSD, Linux, Windows, Git, Docker, networking