

# Kelly Shiptoski

Email: kship at seas upenn edu

Github: <https://github.com/krs85>

Website: <https://krs85.github.io/>

## Research Interests

- Operating systems
- Parallelism and concurrency
- Determinism and reproducibility
- Distributed systems
- System design and development

## Skills

- **Languages:** Rust, C++, C, Java, Python
- **Linux Systems Programming**

## Education

2017 – Present	Ph.D., Computer Science, <b>University of Pennsylvania</b> <a href="#">Architecture and Compilers Group</a> Advised by <a href="#">Dr. Joseph Devietti</a>
2017 – 2019	M.S.E., Computer Science, <b>University of Pennsylvania</b>
2012 – 2017	B.S., Computer Science, B.A., Mathematics <b>Drexel University</b> Graduated cum laude

## Publications

[Reproducible Containers](#), Omar S. Navarro Leija, **Kelly Shiptoski**, Ryan Scott, Baojun Wang, Nicholas Renner, Ryan Newton, and Joseph Devietti. International Conference on Architectural Support for Programming Languages and Operating Systems (*ASPLOS '20*), March 2020.

## Industry Research Experience

**Research Intern, VMWare Research Group**, Summer 2020.

## Research Projects

### Process Cache:

- A system for providing automatic caching of computation at the process level (*WIP*).
- Written in Rust, utilizing asynchronous futures and ptrace.
- Leading design and implementation of the project.

### Distributed Differential Datalog (D3log) - VMWare Research Group:

- D3log is an extension of the Differential Datalog language (a language built upon Datalog, specifically designed for incremental computation), which provides automatic distribution of Differential Datalog computations across compute nodes.
- Contributed to the distributed runtime (written in Rust) by adapting the distributed API to allow for incremental on-the-fly reconfiguration of the nodes within the network, expanding the fault tolerance guarantees of the runtime.

### Reproducible Containers (DetTrace):

- A container abstraction for Linux which guarantees both determinism and reproducibility for any unmodified Linux program run through it. Written in C++ and utilizes ptrace.
- Extended the scheduler from serialized execution to parallelization of system-call-free regions of execution, reducing the overhead of compute-bound workflows to under 2%.

## Teaching

- **Graduate Teaching Assistant** for Computer Architecture (CIS 501), University of Pennsylvania, Spring 2019.
- **Teaching Assistant and Recitation Leader** for Intro to Computer Science (CIS 110), University of Pennsylvania, Spring 2018.

## Other Experience

- **Computer Science Instructor** for Penn GEMS (Girls in Engineering, Math, and Science) Camp, University of Pennsylvania, June 2018.
- **Software Engineering Intern** for Thomson Reuters, Summer 2016.
- **Software Engineering Intern** for Bentley Systems, Inc., Summer 2015.
- **Software Engineering Intern** for Independence Blue Cross, Summer 2014.