Jianxin Qiu

jianxin.qiu@outlook.com · imtsuki · jxqiu

Education

University of Toronto, Master of Engineering 2022 – 2023 (Expected)

- Major: Computer Engineering, *Department of Electrical and Computer Engineering*

Beijing University of Posts and Telecommunications, Bachelor's Degree 2017 – 2021

- Major: Data Science and Big Data Technology, *School of Computer Science*, GPA: 90.66/100

Skills

- **Programming Languages**: not limited to any specific language, and experienced in Rust/C/C++, comfortable with Python/TypeScript/Java/Assembly (in random order).
- **System**: familiar with operating system concepts and design, have experience in optimizing performance at kernel level.
- **Distributed Systems**: taken courses MIT 6.824 and ECE1724, understand consensus algorithms like Raft, have experience in distributed system development.
- **Developing Tools**: experienced in Linux-based programming, have experience with team tools like Jira, Git, etc.
- **Open-source Contributions**: contributed to `@rust-lang, @rust-osdev, @jupyter, @pingcap`, etc.

Work Experience

ByteDance, Beijing, China 06/2021 – 10/2021

(**Lark Messenger Infrastructure**) Rust Engineer Intern

- Worked with the infrastructure team to develop the cross-platform Rust client backend of Lark Messenger.
- Responsible for the calendar component, developed and landed new features such as a new "create, edit and subscribe to calendars" experience.
- Optimized the size of compiled binary by identifying code bloat at assembly level, like extracting logic from macros and refactoring functions where the impact of static polymorphism is significant, with an overall reduction of ~1MB.
- Designed and implemented a new lock-free task queue for asynchronous task execution using channels and const generics, that supports arbitrary async executor, `Future` payload and priority scheduling, replacing the old thread-based implementation.
- Refactored the internal SQL binding codegen tool to support typechecking e.g. `Nullable -> Option<T>` by parsing the schema definition into actual abstract syntax trees (ASTs).

Alibaba Cloud, Hangzhou, China 07/2020 – 08/2020

(**OLAP Database Group**) Database Engineer Intern

- Independently developed Flink connector for ClickHouse, using optimizations like parallel direct shard writing, that outperforms the default JDBC connector by 100% in most common scenarios.

SmartX Inc., Beijing, China 09/2019 – 01/2020

(**Distributed Storage Systems**) R&D Intern, C++

- Improved the long task execution module, like backup storage parallelization, QoS and task status management.
- Implemented Hadoop-like command-line tools for the NFS interface of the storage service.

Network and Big Data Technology R&D Center, Tsinghua University 02/2020 – 07/2020

(**RISC-V Trusted Execution Environment (TEE)**) Research Intern

- Implemented committed instruction flow collection based on RocketChip running on FireSim using Chisel.
- Analyzed memory allocation patterns of Tensorflow and Tensorflow Lite.

Portfolios

- **xv7**: An operating system implemented in Rust. Implemented UEFI Bootloader, memory management and process management, and achieved memory safety in kernel with the help of Rust's safe abstractions and lifetimes. Made contributions to `rust-osdev`, an organization aiming at providing tools useful for OS development in Rust.
- **Hedgehog Lab**: An in-browser, jupyter-like JavaScript execution environment that received over 2,200 stars on GitHub.
- **Reddens**: A rasterization renderer implemented in Metal and Swift.
- **Rust Analyzer Contributor**: The offical Rust language server implementation for IDEs. Added support for IntelliJ-like inlay parameter name hints for call expressions.