JIAJUN MAO

jiajunm@uchicago.edu ◇ 475-529-9896 ◇ jiajunmao.github.io

EDUCATION

University of Chicago *Sept 2020 - June 2024 (Expected)*
Joint BS/MS in Computer Science, System Specialization
Courses: Machine Learning for System, Advanced Operating Systems, Unsupervised Learning, Mathematical Toolkit, Intro to Database, Fundamentals of Deep Learning

Georgia Institute of Technology *Aug 2019 - Aug 2020 (Transferred)*
B.S. in Computer Science, Intelligence and System Arch. Concentration
Courses: Computer Organization and Architecture, Data Structure and Algorithms, Machine Learning

RESEARCH EXPERIENCES

Implement Multi-Level Erasure Code in HDFS and ZFS September 2023 - Present
Multi-Level Erasure Coding Project, UCARE *Chicago, IL*

- Working on MLEC, a hierarchical erasure-coded system that aims to combine the benefits of both local level and network level erasure-coded systems. MLEC achieves high durability, conserves network bandwidth in events of failures, and increase encoding throughput.
- Implementing MLEC, and all of its placement schemes and repair schemes in **Hadoop distributed filesystem** and **ZFS**.

Investigate Storage Characteristics of DP, RAID, SODP and LRC in MLEC March 2023 - June 2023
Multi-Level Erasure Coding Project, UCARE ◇ Advanced OS Course Project. *Chicago, IL*

- Implemented Single Overlapped Distributed Parity as a placement scheme in the MLEC Simulator. Studying its characteristics, such as repair speed, data durability, and network bandwidth consumption, against other placement scheme such as **MLEC, Azure LRC, distributed parity, and RAID**.
- Collected over 500 data points through simulations for the above mentioned characteristics. Used these statistics to benchmark MLEC's performance against other erasure coded systems.

Network Bandwidth Consumption Study of MLEC Aug 2022 - March 2023
Multi-Level Erasure Coding Project, UCARE *Chicago, IL*

- Studied how network bandwidth, especially cross-rack network bandwidth, becomes a constraint on the data durability for network level erasure-coded storage systems. Studied trade-off between repair speed and impact on customer traffic.
- Implemented network bandwidth module to simulate network bandwidth bottleneck effect on network level parity repairs, and collect metrics regarding cross-rack and intra-rack bandwidth consumption during MLEC repair. Demonstrated MLEC's bandwidth conserving property.

Development and Maintenance of MLEC Simulator Jan 2022 - Present
Multi-Level Erasure Coding Project, UCARE *Chicago, IL*

- Developed and maintained a policy-based, Monte-Carlo simulator that is capable of simulating different erasure coding and placement schemes. Currently support local RAID, network RAID, local DP, network DP, LRC, MLEC, SODP. Approximately 13,000 LoC.
- Used Python's multiprocessing library to parallelize and accelerate the simulation speed so that we can simulate systems with high durability.

PUBLICATIONS

Design Considerations and Analysis of Multi-Level Erasure Coding in Large-Scale Data Centers
Meng Wang, **Jiajun Mao**, Rajdeep Rana, John Bent, Garrett Wilson Ransom, Anjus George, Jun Li, Haryadi S. Gunawi. **Supercomputing, 2023**.

INDUSTRY EXPERIENCES

AWS Data Protection, Amazon Web Services May 2022 - Aug 2022 ◇ June 2023 - Aug 2023
Software Development Intern *Seattle, WA*

- Designed and developed a serverless, event-driven **distributed system** for analyzing runtime metrics for AWS Backup services, required to handle more than **50 million** entry ingest for every 8 hours.
- Authored and reviewed **system design** documents that detailed on how to achieve scalability, data consistency, and maintainability for the aforementioned system.
- Built an end-to-end data visualization pipeline around the system to display collected metrics through AWS CloudWatch dashboards and widgets. Setup alarms to automate the ticketing and DevOps process.
- The system is built on top of DynamoDB, S3, CloudWatch, SQS, and Lambda.

Intelligrated, Honeywell Jan 2022 - April 2022
Machine Control Business Logic SDE Intern *Mason, Ohio*

- Designed and developed microservices based on **Java** and **RabbitMQ** that reliably manages and executes business logics for logistical distribution centers.
- Interfaced with **C++** code and Programmable Logic Controller (**PLC**) to drive motors, scanners, and printers to execute proper logic with decision time less than **30ms**.
- Cooperated with a team of 2 to develop and deliver a full suite conveyor solution that helps customer process 30+ cartons per minute.
- Contributed to the business logic Java library by implementing a distributed pub-sub structured file modification watcher.

Shepherd Money Nov 2022 - Present
Founding Engineer, Backend Lead *San Francisco, CA (Remote)*

- Implemented the deployment infrastructure and pipelines through **GitLab CICD**, **Docker** and **Kubernetes**. Automated and standardized quality and coverage testing through **JUnit** and **Jest**.
- Implemented HA and load-balancing for the website frontend through **Nginx**.
- Lead a team of 4 to architect and implement the backend software stack. Responsible for managing the transition from **JavaScript** backend to a robust, scalable **Java** backend. 35,000 LoC.
- Product managed release cycles and feature roadmaps. Working toward a MVP launch at the end of December 2023. Demo *(very demo-y)* can be accessed at https://www.shepherdmoney.com.

PROJECTS

Predicative Pre-fetching of Static Website Resources for Content Delivery Networks September 2023 - Present
Machine Learning for System Course Project ◇ Team Lead

- Lead a project that attempts to implement a predicative pre-fetch mechanism for static website resources (fonts, style sheets, JS scripts, HTML) in a simulated CDN environment. Aiming to improve hit ratios of CDN caches.
- Used LSTM and attention to predict the future usage of a resource given past access traces. Built a trace generator to generate synthetic CDN access traces as real-world data is hard to obtain.

IunGo March 2022 - Present
Fun/Startup-ish Project ◇ Founder, Lead Developer

- Working with 2 other developers in developing a social app that aimed to provide centralized academic/community resources and information to university students.
- Spring Boot, **Hibernate** and **MariaDB** are used for the backend. Backend adopts a microsesrvice architecture with 4 isolated service components communicating through HTTP calls being used as RPC.
- **React** and **TypeScript** are used for the frontend development. Nginx is used to service the frontend content and assets.
- Bare-metal Kubernetes are used for both frontend and backend deployment.
- Source code is not open sourced. Demo website can be accessed at https://web-dev.kube.iungo.ink.

TECHNICAL SKILLS

Programming Languages	C/C++, Rust, Java, Python, (Java/Type)Script, SQL
Backend Stacks	Hibernate, Spring Boot, MyBatis, HikariCP
Frontend Stacks	React, Vue.js, Axios, React Native
DevOps & Containerization	Kubernetes, Docker, Docker Swarm, GitLab CICD, Jenkins
Cloud Tech	AWS EC2, S3, DynamoDB, Lambda, SQS, CloudWatch, SNS, StepFunctions, EBS, EFS