QIYANG HE

he615@purdue.edu · qiyanghe1998 · homepage · linkedin

EDUCATION

Purdue University, West Lafayette, IN — Jan. 2021 – Present
Ph.D. student in Computer Science *GPA: 4.0 / 4.0*

Southern University of Science and Technology (SUSTech), Shenzhen — Sep. 2016 – Jul. 2020
B.Eng in Computer Science and Engineering *GPA: 3.86 / 4.00 Rank: 3 / 146*

RESEARCH INTERESTS

1. Query Processing & Optimization, Query Compiler, Streaming Processing, Incremental View Maintenance
2. Multi-Dimensional Indexes, applying Machine Learning models on indexes for Query Processing.

PUBLICATION

Efficient Incrementialization of Correlated Nested Aggregate Queries using Relative Partial Aggregate Indexes (RPAI)
Supun Abeysinghe, Qiyang He, Tiark Rompf
Proceedings of the 2022 International Conference on Management of Data (SIGMOD 2022).

Reachability types: tracking aliasing and separation in higher-order functional programs
Yuyan Bao, Guannan Wei, Oliver Bračevac, Yuxuan Jiang, Qiyang He, Tiark Rompf
Proceedings of the ACM on Programming Languages, Volume 5 (OOPSLA 2021).

SELECTED HONORS AND AWARDS

1 Gold, 1 Silver and 2 Bronze Medals in ACM-ICPC Asia Regional Contest	2017 – 2018
Bronze Medal, National Olympiad in Informatics, China	Jul. 2015
Student Volunteer for PLDI 2021	June. 2021

WORK EXPERIENCE

Research Intern, Pinterest, San Francisco, USA — May 2022 – Aug. 2022
1. Design an efficient protocol for incremental backup between **RocksDB**-based service and **AWS S3**, handling failover, resource management, concurrency, and split-brain problems.
2. Implement a prototype on one node for the previous protocol in **Rocksplicator** with small space (one copy of data in total in cloud) and 10x speedup for backup. (**C++**, **S3**, **RocksDB**)

Research Intern, SenseTime, Shenzhen & Beijing, China — Jan. 2020 – Dec. 2020
1. Help to implement a distributed storage system for image data. (**Go**, **C++**)
2. Benchmark for auto-schedulers of neural network, including **Ansor**, **autotvm**, **Pytorch** and **FlexTensor** (**PyTorch**, **C**, **Python**, **TVM**)

RESEARCH PROJECT

Survey for Learned Multi-dimensional indexes — Sep. 2022 – Now
Mentor: **Prof. Walid G. Aref** Purdue University
1. Have read 10+ papers about applying machine learning techniques to optimize current data structures or design new indexes to support dynamic or static query processing
2. Categorize all the indexes into several classes by workload, and optimized direction (e.g. search or storage).

A Logical Plan based Efficient Backend for Incremental View Maintenance — Oct. 2021 – Now

Mentor: **Prof. Tiark Rompf** Purdue University

1. Leverage generative programming techniques (**LMS**) to add a new C++ codegen backend for **SparkSQL** to support incremental view maintenance **SQL** queries.

2. Propose new operator group-thetajoin combing group-join and predicate to optimize nest-aggregate queries (achieve 10x speedup compared to **DBToaster** on **TPC-H**). (**Scala**, **SQL**, **SparkSQL**, **C++**)

Efficient Incrementalization of SQL Queries with Nested Aggregates — Apr. 2021 – Nov. 2021

Mentor: **Prof. Tiark Rompf** Purdue University

1. Find out redundant computation for incremental view maintenance of nested-aggregate correlated queries.

2. Build novel tree-based and hash-based indexes to improve the incrementalization efficiency with up to 1000x over the **DBToaster**. (**Scala**, **SQL**, **Pandas**)

Improving data ingestion performance in Apache AsterixDB — Jul. 2019 – Sep. 2019

Mentor: **Prof. Michael J. Carey** UC Irvine

1. Decouple the data intaking and data parsing in the data feed of AsterixDB.

2. Make the data parsing parallel and get about 2x speedup over the current AsterixDB (**Java**)

Course Project

Implementing a distributed key-value store system based on dslabs — Jan. 2021 – May. 2021

Lecturer: **Prof. Yongle Zhang** Purdue CS 505 Distributed Systems

1. Implement exactly-once **RPC** protocol for messages sending between clients and servers and **Primary-Backup** protocol in unreliable network conditions based on the previous RPC protocol.

2. Implement **Multi-Paxos** (leader election and consensus) protocol, **Sharding** operations and transactions by two-phase commit protocol for the previous messaging system. (**Java**)

Teaching

Lab instructor for CS180 Problem Solving and Object-Oriented Programming	Aug. 2022 – Dec. 2022
Lab instructor for CS505 Distributed Systems	Jan. 2022 – May. 2022
Teaching Assistant for CS381 Introduction To The Analysis Of Algorithms	Aug. 2021 – Dec. 2021

Technical Skills

Programming Languages: C, C++, Java, Python, Scala, SQL, Bash

Systems and Libraries: Pandas, Git, Latex, SparkSQL, RocksDB