Aneesh Durg

Email: aneeshdurg17@gmail.com | Website: `aneeshdurg.me` | Github: `github.com/aneeshdurg`

WORK EXPERIENCE

Senior Software Engineer Jul 2023-Present

Bodo.ai — remote

- Developing the core engine which consists of an optimizing compiler and scalable distributed runtime (using **MPI**) for **SQL** and python/pandas workflows.
- Identified optimizations that reduced compile time by **60%** in some benchmarks

Senior Software Engineer/Team Lead Feb 2021-Present

KatanaGraph Inc. — Austin, TX

- Worked on building a distributed graph compute engine that provides AI, analytics, and a database.
- Lead a team of **5** to implement and support graph database querying and ingest.
- Implemented compiler and runtime support for the **Cypher** query language.
- Designed novel high performance algorithms for distributed subgraph pattern matching (tested on ∼**20B** nodes, **44B** edges)
  - Improved performance by **100x** in queries against the **LDBC-SNB** datasets and reduced memory usage by over **95%** on benchmarks simulating specific client workloads.
- Designed hotswap mechanism to allow devs to update katana deployments on **kubernetes**, reducing testing time by **30x**
- Built infrastructure for benchmarking the query engine in isolation from the rest of the product using **slurm**

Member of Technical Staff Aug 2019-Feb 2021

Qumulo Inc. — Seattle, WA

- Worked on building a distributed scale-out filesystem, supporting both on-prem and cloud.
- Designed a solution for reducing server downtime during upgrades by **10x** in a team of four
- Implemented **SMB3.1** support and features, and extended platform support for two new hardware configurations
- Lead migration of **python2** code to **python3**, and introduced enforced type checking via **mypy**

PROJECTS

rainbow python/Cypher

https://github.com/aneeshdurg/rainbow

- Arbitrary compile-time function coloring and callgraph rejection tool powered by **clang** and **Cypher**
- Provides an ergonomic way for users to labels functions and lambdas, and then define relationships between those labels that should be considered invalid. Some example usecases are:
  - label functions that assume locks are held to verify that they are never called without a lock
  - label routines using collective **MPI** operations to ensure that other collective operations aren't called during execution
  - prototype new language features such as async/constexpr without writing custom compiler passes/extensions

spycy python/WASM

https://github.com/aneeshdurg/spycy

- An in-process graph database library for python that implements a **openCypher** frontend
- Provides implementable interfaces for data sources to enable querying real world graphs.
  - Wrote a demo that uses **spycy** and **WASM** to filter HTML nodes using **openCypher**

What Is a Filesystem? Javascript

https://aneeshdurg.me/what_is_a_filesystem

- An online interactive book/vizualization for students learning filesystem concepts.
- Implements a interactive ext2-esque filesystem simulator with animations to illustrate disk accesses.
  - Features a terminal simulator demonstrating how standard `GNU/Linux coreutils` might interact with the disk.

CameraTheremin JavaScript

https://aneeshdurg.me/CameraTheremin

- In-browser webcam gesture-based theremin (a musical instrument) powered by **Javascript** + **WebGL**

EDUCATION

University of Illinois at Urbana-Champaign Aug 2015-May 2019

Recieved BS in **Computer Science & Mathematics** with **High Distinction**