Jason Barmparesos

jason@jasonb.dev ● jasonb.dev ● github.com/nsailor

Warsaw, Poland & Remote ● Résumé current as of February 9, 2023

Employment

ByteDance (TikTok), ***Software Engineer – Warsaw, Poland*** . . . . 05/2021 - Present

Worked on the following projects so far:

- A new, distributed, large-scale, approximate nearest-neighbor database built around HNSW in C++.
  - Designed and implemented new features to allow efficiently searching subsets of the data.
  - Through successive optimizations, increased search throughput by more than 10 times.
- A tool for importing data, managing the database, and evaluating its performance in Rust with Tokio.
- A distributed execution engine for dynamically configured video deduplication pipelines in Go.
- Several internal graphical tools for evaluating and debugging video deduplication pipelines in Python and React/TypeScript.

Skills: C++, Rust, Python, Thrift, React

Arm, ***High-Performance Computing (HPC) Tools Summer Intern – Warwick, UK*** . . . . . . 07/2018 - 09/2018

- Worked on the Arm DDT debugger and the Arm MAP profiler in C++ with Qt.
- Implemented new features and fixed several bugs in an agile workflow.
- Developed the foundations for a UI testing framework in Python.
- Wrote a Python program to generate an issue heatmap for each source code file using the JIRA API.

Skills: C++, Python, Qt

Academics

Aristotle University of Thessaloniki, ***Thessaloniki, Greece*** . . . . 2015 - 2021

Master's Degree in Electrical & Computer Engineering

Thesis: *A high-level solver for the compressible Euler equations using Julia*

Developed a Julia package that allows solving certain transport Partial Differential Equations (PDEs) by turning them into Ordinary Differential Equations (ODEs) (Method-of-Lines). The package lets the user define a flux function, an unstructured mesh, and boundary conditions, and then creates a function evaluating the time-derivative based on the current solution. The resulting ODE can then be solved by any applicable method. Supports solving the problem on a CUDA GPU with a single extra function call. For the compressible Euler equations, achieved performance comparable to SU2 on the CPU and x97 acceleration compared to a single CPU thread using CUDA.

Skills: Julia, CUDA, Computational Fluid Dynamics, Aerodynamics

Politechnika Warszawska, ***Warsaw, Poland*** . . . . 2020 - 2021

Erasmus+ exchange. Winter Semester 2020. Faculty of Electronics & Information Technology.

Notable Open Source Contributions

AMDGPU.jl *github.com/JuliaGPU/AMDGPU.jl* — An effort to run Julia code on AMD GPUs with ROCm.

- Expanded coverage of the HSA API by, among others, adding support for memory locking and coarse grained allocations.
- Wrote the initial project documentation.
- Various refactorings and improvements.

Skills: Julia, GPUs, HIP

Notable Side Projects

SoundTouch Accessible tree-structured navigation for the visually impaired. Arranges items on a grid and provides audio feedback in response to mouse pointer motion. Won 1st prize in the Accessibility Hackathon held in Athens, Greece, May 2017. Skills: C++ , Qt 5

Ilma CFD Lattice Boltzmann Method (LBM) solver accelerated with OpenCL. Skills: C++ , OpenCL

Languages

English, Modern Greek, Polish (conversational)