Karl Hiner

612-747-9175 | karl.hiner@gmail.com | LinkedIn | GitHub | Portfolio

Education

Portland State University — Portland, Oregon
Bachelors of Science, Computer Science | GPA: 3.82 — 2010 – 2013

Georgia Institute of Technology — Atlanta, Georgia
Master's, Computational Sciences and Engineering | GPA: 3.9 — Sep. 2022 – graduating Apr. 2024

Skills

Languages: C++, Python, TypeScript/JavaScript, Java, Ruby, Julia, SQL

Technologies: Git, ImGui, React, Postgres, Docker, JAX, PyTorch, Node

Expertise: Full-stack development, technical leadership, physical modeling and simulation, applied machine learning

Experience

Axiom Data Science — Portland, Oregon
Lead Software Engineer — Feb 2020 – Jul 2022, Full-time

Lead developer on the next generation of the Research Workspace, a web application for collaboratively managing data for science projects. Designed and implemented the following major projects:

- An extensive admin application supporting many internal Research Workspace use-cases.
- A user authentication and authorization service backend and UI library, used across multiple Axiom services.
- An internal set of packages to improve and modularize UI components.
- A remote browser service acting as an internal QA tool to help quickly discover regressions across our many portals and pages, as well as a remote browser screenshot service for generating animations for portal maps.
- Major features and improvements for Axiom's data portals, including a cross-portal feature called *Map Views* enabling users to create, edit, share and publish multiple portal map instances.

Cozy — Portland, Oregon
Senior Software Engineer — Oct 2018 – Jan 2020, Full-time

- Led development of major components of Cozy's web app and payments system.
- Transitioned components and services to be presented and consumed by Apartments.com after Cozy's acquisition by CoStar Group.
- Mentored junior engineers. Led product development efforts on large projects, including project and sprint planning and organizing team efforts.

Self — Portland, Oregon
Independent Study — Dec 2017 – Oct 2018, Full-time

I took an unpaid sabbatical to focus on learning about fields I am passionate about. After returning to professional life, I continued to learn and read papers in my free time, and finally decided to pursue a Master's degree, aiming to transition into software domains that deeply interest and inspire me.

- Released an Android app for sample-based music production.
- Studied digital audio signal processing, machine learning, C++, and Python, and produced in-depth Jupyter notebooks for each chapter of eight technical books on these topics.
- Completed online courses in deep learning and statistics.
- Developed digital audio workstation software.
- Implemented a declarative static site generator in React, and used it to build my portfolio/blog website.

New Relic - Applied Intelligence Services Team — Portland, Oregon
Senior Software Engineer — Oct 2016 – Dec 2017, Full-time

- Researched and developed products for the Applied Intelligence team, including dynamic baselines, error profiles and host outlier detection.
- Leveraged machine learning and statistical techniques on data from many monitoring sources to provide customers with actionable information and context to find, understand and fix software problems quickly. Architected, built, shipped and maintained frontend components, backend services and APIs.

- Architected, built, shipped and maintained frontend components, backend services and APIs.

New Relic - Mobile Product Team Portland, Oregon
Software Engineer / Senior Software Engineer Apr 2014 – Oct 2016, Full-time

- Designed, built and maintained several APIs, UI features and high-throughput services, mostly using Java, Ruby and React.
- Acted as the technical lead on significant features including activity tracing, version trends, as well as features for crash reporting, network reporting, and real user monitoring.

New Relic - Mobile Team Portland, Oregon
Junior Software Engineer May 2013 – Apr 2014, Full-time

- Co-produced the front-end for the Mobile product Rails application.
- Developed data collection and aggregation service features.
- Implemented components of the Android agent responsible for sourcing and reporting data from Android applications.

Thetus Corporation Portland, Oregon
DevOps Intern Jan 2013 – May 2013, Half-time

- Automated host environment setup and web-app deployment.

Jive Software Portland, Oregon
Software Engineering / Quality Engineering Intern Jan 2012 – Dec 2012, Half-time

- Developed features and fixed bugs for the Jive platform.
- Designed and implemented an automated test framework for Jive's web application on a small team.
- Co-implemented a modular and extensible testing framework, and wrote many functional tests using this framework.
- Worked with teams to generate test plans.

Projects

MeshEditor | C++/Vulkan/ImGui | GitHub

Real-time mesh viewer and editor with rigid body audio modeling, and interactive RealImpact dataset explorer supporting comparison of audio models with real-world impact recordings.

mesh2audio | C++/OpenGL/ImGui | GitHub

Real-time modal audio synthesis from 3D meshes, with interactive vertex excitation.

Jupyter notebooks | Python | GitHub

Python Jupyter notebooks covering each chapter of several books, including:

- Gareth Loy's Musimathics *Vol 1. Vol 2.*
- Julius O. Smith's *Mathematics of the DFT*, *Intro to Digital Filters*, and *Physical Audio Signal Processing*

Generating Music with WaveNet and SampleRNN | Python | GitHub

Exploring musical raw audio generation using these popular models.

JAXdsp | Python/JAX/TypeScript/React | GitHub

Parameterize audio graphs in real-time to model an incoming/outgoing audio stream pair with differentiable DSP components, with data and audio over WebRTC.

FlowGrid | C++/ImGui |GitHub

Immediate-mode interface for the Faust functional audio language, backed by a persistent store supporting navigation to any point in the project history in constant time.

BeatBot | Java/C/OpenGL | GitHub

A sample-based music production app for Android, with an OpenSL audio/effects backend implemented in C, and a custom OpenGL-based 2D UI designed to minimize draw call submissions for optimal performance on low-end devices.

Auto-Sampler | MaxMSP/Ruby/Javascript/C | GitHub

A Max4Live instrument that streams looping audio segments matching the pitch of incoming MIDI notes in real-time.