

Sebastian J. Hamel

He/Him



seabassjh | sebastian-hamel |
 +1 (330) 280-1852

Open Source Projects

🔗 Bevy Kajiya

Adds ray-traced rendering to the Bevy Engine

[Project repo](#)

- Enables any novice gamedev to experiment with raytracing
- High Performance- 60+ render frames per second, 60+ engine ticks per second
- User interface abstracts away low-level raytacing pipeline management from the user for ease of use
- Endorsed by 50+ Rust community members (Github stars)

🔗 NBody-WASM-Sim

GPU-rendered astrophysics simulation in the web browser

[Demo](#)

[Project repo](#)

- Performant real-time physics in Rust using linear algebra libs and WebGPU- 3.5x faster than Javascript
- Builds and deploys the demo web page automatically using GitHub CI/CD Actions
- Serves as an open-source template to create GPU-accelerated, interactive web apps
- Endorsed by 70+ Rust community members (Github stars)

🔗 Contributions to KubOS

Simulation support for Rust flight software

[Demo](#)

[Contribution](#)

- Enables Rust-based flight software (KubOS) to run on NASA's software-simulated CubeSat hardware
- Contains Rust bindings for NASA CubeSat emulation HALs

Career

Software Engineer I

Feb 2022 – Nov Present

NASA/ToSC, Command & Control

Kennedy Space Center, FL, USA

- Software engineering of simulation software for Exploration Ground Systems for Artemis II, Artemis III
- Developed and verified software conforming to NASA Class C Software requirements
- Developed and debugged networked software communicating with PLC runtimes using the Common Industrial Protocol (CIP)
- Reverse-engineered legacy Rockwell PLC functions and designed their re-implementation and integration into a custom PLC emulator
- Created design presentations to coordinate strategy between workgroups to emulate various PLC functionality in emulation

Software Engineer I

Jun 2020 – Feb 2022

AFIT, Center for Space Research & Assurance

Wright-Patterson AFB, OH, USA

- Contributed to the first open-source spacecraft flight software framework written in Rust
- Developed CubeSat flight software in C with NASA's cFS framework and also in Rust with KubOS
- Developed CubeSat ground control software which interfaces with databases and TCP/UDP hardware interfaces in Python
- Automated generation of template files ingested by flight software and ground software to accelerate CubeSat mission development

Software Engineering Co-op

Jan 2019 – Aug 2019

NASA IV&V

Fairmont, WV, USA

- Contributed to NASA's open-source flight software simulator (NOS3) with upgraded packages and documentation
- Developed Rust bindings to C++ based hardware abstractions layers for NOS3 simulator
- Modeled CubeSat OEM hardware components into C++ emulators running in NOS3

Competencies

- ✎ Primary Languages: Rust, C/C++, Python
- ✎ Other Languages: Java, TypeScript, GraphQL, IEC 61131-3 Structured Text, Ladder Logic
- 🔄 CI/CD: GitLab CI configurations (YAML)
- 📁 Content Management: Git, GitHub, GitLab
- 🐳 Software: Docker, Confluence/Jira, VersionOne, Visual Studio, VSCode

Education

Bachelor of Science, Computer Science & Engineering

University of Toledo, ABET Accredited

- Summa Cum Laude
- 3.97 GPA

Class of 2021
Toledo, OH, USA