Jack Wrenn

Phone	401-793-6774
Email	jack@wrenn.fyi
Website	jack.wrenn.fyi
Github	jswrenn

Education

Brown University

Computer Science

Ph.D. 2022

Sc.M. 2018

University of Rhode Island

Computer Science, History

B.A. 2015

Employment

Applied Scientist

Amazon Web Services 2021–2025

Interviewed Rust programmers to identify their productivity barriers, and developed a variety of interventions. Led initiative to improve the observability of asynchronous Rust applications. Led cross-industry initiative to improve safety tooling.

Adjunct Professor

University of San Francisco Spring 2024

Co-taught an intensive introduction to computer security that presented the fundamentals of cryptography and systems security through a mixture of theory and practice.

Researcher

Brown University 2015–2021

Contributed to the development of the Pyret programming language. Published five research papers on error reporting, testing, tooling and human factors.

Selected Awards

Client Bug Bounty

Mozilla 2019

Discovered a security issue in Firefox's XSLT engine that enabled attackers to indefinitely execute JavaScript and send network requests (even with JS disabled) *after* the exploited tab was closed.

Selected Projects

Rust Programming Language

An expert in leveraging and evolving Rust to improve the safety and ergonomics of complex programming tasks. Key projects include:

Safe Transmutation

Lead the working group to make bit-reinterpretation casts (e.g., `union`, `mem::transmute`) memory safe. Designed and implemented `TransmuteFrom`, a novel compiler analysis for proving the transmutability of arbitrary types. Maintain zerocopy, a widely-used library providing a safe abstraction for many kinds of bit-reinterpretation casts.

Observability

Led the AWS Rust Platform Team's observability initiative. Collaborated with the Rust Debugging Working Group to prototype and evaluate methods for in-process debugging of stuck asynchronous services. Authored a suite of novel crates for program instrumentation. This work culminated in the design and development Tokio's task dump functionality.

Other Open-Source Leadership

Maintainer of itertools, a popular (>575m downloads) library for ergonomic transformations and summarization of streams of data.

Computing Education

An expert in programming pedagogy and user experience. Contributed to Pyret, a programming language designed by computer science educators:

Error Message Design

Developed Pyret's unique hypertext error messages, which leverage hyperlinks, highlights and in-line code snippets to guide novice programmers towards deeply understanding their errors.

Data Science Support

Developed Pyret's language-level support for manipulating tabular data, used by the data science module of the Bootstrap curriculum, a research-based computer science curriculum for grades 6–12.

Example-Driven Development

Developed a cloud-hosted IDE for Pyret that encourages students to write input–output examples *before* they begin programming, and conducted research to assess its impact.

Evaluation at Scale

Developed infrastructure for assessing the quality of students' Pyret programs and test suites at massive scale on a distributed super-computing cluster, and developed an analysis to validate the robustness of these assessments.

Selected Talks

Safety Goggles for Alchemists [Transcript]

RustConf 2024

This talk charts Rust's path to becoming the first systems programming language with transmutation safety, and highlights how safe transmute is already being put to use to build next-gen systems.