J. R. Hill

hiljusti@so.dang.cool — 206.948.5692

▷ interests and goals

I'm a polyglot programmer interested in extreme force multiplication and enabling the "best tool for the job" languages whether they're cutting-edge or ancient. For the past couple years I've enjoyed building support for new languages and toolchains at Amazon, modernizing its old and brittle enterprise-scale codebase, and jumping into critical CVE mitigations to provide expert mitigations.

▷ skills

programming languages

Breadth ⇀ I have experience with more than 100 programming languages spanning many paradigms. I've written mentored students in 28 programming languages, and created real-world applications in dozens of languages.

Depth ⇀ I have jumped head-first into both green-field and legacy projects. TODO

▷ career highlights

amazon, code foundations, 2020–now

Lone Wolf ⇀ Unless otherwise specfied as a larger effort, all work on this team is my own personal contribution.

Improving the Java language experience was my primary focus here. JVM languages like Java, Kotlin, and Scala are the primary languages of Amazon, making up roughly 80% of the code base (by project), and about 60% of active commit activity. We experienced high turnover in management during this period, I had 5 changes in direct manager in 2021 alone, and it wasn't possible to properly fund and fill seats for a full development team. (Fingers crossed for 2023.)

Community Leader ⇀ Of course, there's no way to make changes in an enterprise-scale system used by more than 80,000 engineers without coordination and making lots of friends. I've fostered numerous interest groups and established shared responsibilities with key technical leaders and subject-matter experts. The largest group, "Gradle Hackers" is a collection of 132 engineers from various business units across Amazon, AWS, and subsidiaries with a strong core of contributors and advisors.

Improved Java Language Experience ⇀ I made Java suck less for the whole company through large-scale software curation, community management, and modernizing templates. I also maintained and made tactical changes to foundational software used transitively by >500,000 software projects. I've also directly contributed to more than 6,000 software projects directly through manual and automated changes.

More Tools in Amazon's Toolbox ⇀ In Amazon's older "Brazil" codebase, I codified templates for basic Rust, Go, Kotlin, and NodeJS usage patterns. I also built support for Zig, and assisted the Kotlin and Scala communities. In Amazon's new "Peru" codebase, I built support for Zig, Apache Maven, Apache Ant/Ivy, CMake, Clojure. I also provided the earliest examples of Android and Lambda projects as well as the earliest examples for building C, Fortran, and X86 64-bit Assembly (GNU Assembler and NASM) projects.

"Builder Tools 102" ⇀ Began a cross-org effort (Learning tech, Tech writers, Builder Tools Product Management, etc) to supplement and extend newhire training for software engineers in areas where a 101 course was sorely lacking. This became a 102 course launched in 2023.

Centralized CVE Mitigation ⇀ I provided large-codebase-compatible patches to critical 3p software.

aws commerce platform, 2018–2020

Replication Validator ⇀ TODO

Cutover Service ⇀ TODO

aws support, 2014–2017

Operational Tools ⇀ TODO: TT Kiosk, Link Alchemizer – Still not replaced as of 2023!

Automated Refund Application ⇀ (As a Technical Customer Service Agent) Proposed and implemented automation via JavaScript to very manual web-based refund tool that saved AWS >$1mm in 2014 operating costs.

▷ charity and mischief

Exercism ⇀ I am occasionally active on exercism.org, a learning and mentoring platform for programming languages. I've mentored students here in 28 languages. I also have contributed to multiple language tracks and helped develop and launch tracks for Fortran and Zig.

Rail ⇀ Rail is an experimental stack machine I'm crafting in Rust for concatenative programming languages. It currently has two experimental language frontends: *dt* (Joy-like) and *stap* (Lisp-like).

Sigi ⇀ Sigi is an organization tool for terminal lovers who hate organizing. It's a todo-list generator distributed through multiple Linux package managers.