

# James Leitch

Whitehorse, YT | 424-901-3053 | me@rickvanprim.com | www.rickvanprim.com

## Profile

I am a software engineer and generalist, having worked professionally in the software industry since 2010. While the majority of my experience is working with performance conscious native code, I enjoy trying to understand and work on the whole picture, from the actual code being written, to the build and deployment automation, and developer workflow behind it. I am always looking for a challenge, and I enjoy working on fresh and interesting problems with other smart, motivated, and passionate people. I thrive in a cross-discipline environment and love helping content creators realize their vision.

I am presently working for my company, Consuming Chaos, developing a game using all the experience I've amassed. I am looking for full time work to continue to grow my skills and to help fund my sushi addiction.

## Skills

### Strong

- Rust / C++ / C
- Bazel
- Git
- Containerization (Docker)
- Native Code Optimization
- Collaboration with Content Creators
- FFI (Foreign Function Interface) / Scripting Integration

### Proficient

- Nix / NixOS
- Kubernetes / Skaffold / Helm
- Windows / MacOS / Linux / iOS / Android
- GitHub Actions
- Postgres / SQL
- TypeScript / React / Redux
- HTTP REST / GraphQL
- Cross Compilation
- Perforce

### Familiar

- Unity / Unreal Engine
- C# / Swift / Objective-C / Go
- 3D Math
- AWS
- QUIC
- Node / Node Native Modules
- Protobuf / gRPC
- Oculus VROS
- Jenkins
- Buck Build

## Experience

### Consuming Chaos

#### Founder — March 2023 to Present

- Developing a free-to-play PvP game targeting Windows, Mac, iOS, and Android.
- Software is being developed fully in Rust, which includes a 3D client targeting multiple platforms and a scalable backend running in the cloud.
- Tech stack includes NixOS development infrastructure, VSCode Remote targeting NixOS servers, Bazel build system (including cross compilation), and Skaffold + Kubernetes deployment.

### Meta

#### Software Engineer (E5 - Tech Lead) — November 2021 to March 2023

- Provided technical leadership on the development and bi-weekly delivery of a web app responsible for managing, auditing, and visualizing networking equipment.
- Drastically improved onboarding time and continuous integration by leading a build/development environment unification initiative.

- Formalized release process with Quality Assurance and the other members of leadership, and in turn raised the confidence in release quality.
- Mentored junior engineers on software craft and architecture, with an emphasis on ensuring correctness and thinking about invariants.

### **Software Engineer (E5) — August 2020 to November 2021**

- Worked with my Product Manager to refine a loosely defined objective into a set of concrete deliverables.
- Developed and shipped Meta Quest Developer Hub's Perfetto integration and performance tab.

## **Apple**

### **Software Engineer (ICT4) — October 2018 to August 2020**

- Shipped Reality Composer's support for USDZ as an export format.
- Worked with content creators and the Reality Composer team to optimize and deliver a collection of Augmented Reality experiences for the TV show For All Mankind.

## **Riot Games**

### **Software Engineer (Mid) — September 2016 to June 2018**

- Worked with several R&D teams to onboard them to the internal playtest tool and provided ongoing support.
- Replaced manually administered "patch boxes" used by many R&D teams with an automated solution using Terraform.

### **Software Engineer (Associate) — June 2014 to September 2016**

- Contributed to the release of Gnar, Azir, Kalista, Rek'Sai, Tahm Kench, Aurelion Sol, and Ivern on League of Legends' Champion Team.
- Refactored League of Legends' animation system from the old proprietary data format to the new consistent game-wide format, including updating the bespoke tool and migrating all data.
- Operated in a live service development cycle that required releases every 2 weeks.

## **Electronic Arts**

### **Software Engineer (SE 1) — May 2013 to May 2014**

- Worked on numerous gameplay systems, including implementing a data driven buff system with client/server synchronization in the Frostbite engine.
- Shipped Plants vs Zombies: Garden Warfare.

### **Software Engineer (SE 1) — June 2012 to April 2013**

- Worked on C# gameplay for an unannounced Unity game for Quicklime Studios prior to cancellation.

## **SMART Technologies**

### **Software Developer (SD 1) — August 2011 to May 2012**

- Worked on an R&D prototype of a multi-user, collaborative, infinite canvas web app.

### **Software Developer Intern — May 2010 to August 2011**

- Worked on educational applications for the SMART Table.

# Community Contributions

## Blog

### Consuming Chaos Blog ([blog.consumingchaos.com](http://blog.consumingchaos.com))

- Posts detailing my experiences setting up managed developer environments, toolchain provisioning, cross compilation, and any other notable subjects I encounter during development of Consuming Chaos' first game.

## Open Source

### [github.com/bazelbuild/rules\\_rust](https://github.com/bazelbuild/rules_rust)

- Fixed some linker invocation issues that arose during cross compilation.
- Added a check to prevent accidentally targeting a platform that `cargo-bazel` hasn't been configured to target.

### [github.com/rust-lang/rust-analyzer](https://github.com/rust-lang/rust-analyzer)

- Added support for ProblemMatcher to handle diagnostics directly from `rustc` instead of just `cargo`.
- Miscellaneous fixes and enhancements to improve support for non-Cargo build systems.

## Miscellaneous

### Technical Editor on *Game Coding Complete, 4<sup>th</sup> Edition*

- Reviewed technical accuracy and provided subjective feedback to the authors, Mike McShaffry and David "Rez" Graham.
- Code reviewed the same code embedded alongside the chapters of the book.

## Education

### BSc in Software Engineering — 2012

University of Calgary