

# Nathan HOLLAND

## PERSONAL DATA

---

ADDRESS: 818 N Logan St. Apt. #906, Denver, CO 80203  
PHONE: (201)-317-1487  
EMAIL: [nholland94@gmail.com](mailto:nholland94@gmail.com)  
GITHUB: <https://github.com/nholland94>

## ABOUT ME

---

I am a passionate, self-taught programmer with an interest in many different programming disciplines. Some of the topics that excite me the most are programming languages, compilers, parallelism, and 3d graphics, but I also have a good deal of experience with full stack web development and DevOps tools and practices. Whenever a new challenge faces me, even if I am not familiar with some of the details behind the challenge, nothing stops me from diving into the underlying details or theory and tackling the challenge head on. I am always interested in opportunities to improve my programming skills, both inside and outside of the workplace.

## WORK EXPERIENCE

---

<i>Current</i> AUG 2017	<b>Founder / CTO</b> <i>Bitversity LLC</i> Designed and developed technology for an online course which teaches programming to people with no prior experience. The course revolves around simplified programming languages and a simple virtual machine designed to teach various programming paradigms and techniques rather than specific real world languages. The custom compilers and virtual machines allow for advanced introspection into the code students write and how it executes. Technologies: OCaml, Javascript, Elixir, js_of_ocaml Skills: language design, teaching, compilers, full stack
AUG 2017 NOV 2016	<b>Software Engineer - Compilers</b> <i>Self Employed</i> Worked on the creation of an array programming language with an emphasis on optimizable parallelism. Developed compiler primitives, runtime execution system (through Vulkan), and experimented with various forms of optimization. Researched and implemented optimizations on sparse resource graphs using petri nets. Technologies: OCaml, D, Prolog, Vulkan Skills: language design, compilers, optimization, parallelism, GPGPU programming, execution analysis
NOV 2016 DEC 2013	<b>Software Engineer - Full Stack</b> <i>Granicus</i> Developed internal projects to support legacy software. Spearheaded development on new products. Taught other developers various programming languages, technologies, and techniques. Investigated new technologies for the company and provided analysis. Worked on automated deployment system and tools associated with it. Built build processes for frontend applications. Technologies: Javascript, Ruby, PHP, C#, Go, React, Electron, Chef, AWS, MongoDB, MySQL, PostgreSQL Skills: teaching, leadership, prototyping, full stack

## PROJECTS

---

Bynar	<p>High level GPGPU compute library</p> <p><i>Source:</i></p> <p>Provides a high level interface for execution stages on large data sets. Execution stages are joined together and dynamically run on the CPU or GPU, depending on input data size. Fences and data transfers between stages are automated.</p>
Prometheus	<p>Distributed, automated test running tool</p> <p><i>Closed source</i></p> <p>Manages and executes test suites across multiple testing environments. Supports provisioning environments via the cloud, as well as job distribution to static hosts. Provides a dashboard to QA to automate and dispatch testing jobs, as well as display results and frequency of errors across tests.</p>
Fudd	<p>MySQL binlog deserializer and RabbitMQ publisher</p> <p><i>MySQL library source:</i> <a href="https://github.com/granicus/mysql-binlog-go">https://github.com/granicus/mysql-binlog-go</a></p> <p><i>Other components are closed source</i></p> <p>Built to solve specific problem with legacy software. Streams and deserializes MySQL binlogs and publishes certain model transactions over RabbitMQ. No existing MySQL binlog deserializer worked for the version of required MySQL at the time, so one had to be built from scratch. Performs block deserialization and message publishing concurrently.</p>

## PROGRAMMING LANGUAGES

---

HIGHLY PROFICIENT:	C, OCaml, Elixir, Ruby, Go, Javascript, PHP, Java, Prolog, Lua
MODERATELY PROFICIENT:	Lisp (specifically Scheme), x86 assembly, APL, Ada, D, Python
MILDLY PROFICIENT:	Factor, Rust, Haskell, Red, ATS, Coq

## TECHNOLOGIES

---

WEB DEVELOPMENT:	Rails, Phoenix, React.js, Backbone.js, Underscore.js, WebSockets, Socket.io, Apache, Lighttpd, HTML, CSS
OTHER TECHNOLOGIES:	Linux, SQL, Chef, Makefile, Ocamlbuild, Grunt, Xlib, Wayland, OpenGL, Vulkan

## INTERESTS

---

Programming Languages, Compilers, Optimization, Parallelism, Operating System Kernels, Music Visualization, Music Generation, Virtual Reality, Neural Networks, Vulkan, Game Engines, Puzzles