

Jeremy Wall

Email: jeremy@marzhillstudios.com

Site: jeremy.marzhillstudios.com

About Me

I am a self taught coder with a love for languages and good design. I've been working in the industry for a little more than 10 years. I've been learning and coding in the open source community for far longer than that though. From hacking personal customizations in Blender, to coding my own website, to writing about various languages and technologies on that site. I've been an active part of the programming community for more than 20 years.

I'm passionate about the tools used in the craft of software development. Build systems, Test frameworks, Editors/IDE's all fascinate me and I'm constantly looking for new ways to streamline and improve the process of software construction.

Notable Personal Projects

- * UCG a Configuration format compiler at github.com/zaphar/ucg
- * Rust ICMP Socket library at github.com/zaphar/icmp_socket
- * Parsing combinator library at github.com/zaphar/abortable_parser
- * Network quality exporter for prometheus at github.com/zaphar/durnitisp
- * Original author of C# support for bazel.io/ at github.com/bazelbuild/rules_dotnet
- * One of the only people to provide a pull example for the libgit2 rust bindings at pull.rs on github

You can see the rest of my personal coding projects on github.com/zaphar

.

Employment

GoHealth, Inc - VP of Engineering 2021 - Present

I was promoted to VP of Engineering during my 5th year and helped to drive processes for technology adoption, security and compliance. My areas of oversight were our new Customer Retention stack, Shared Services, Tech Support, and Data Engineering.

GoHealth, Inc - Principal Engineer 2016 - 2021

I joined GoHealth to help them as they moved toward a more agile model in the SDLC. They were beginning to embrace Continuous Integration and Continuous Delivery. My responsibilities as Principal Engineer were to:

- Mentor other developers
- Provide technical direction for old and legacy applications
- Help to form and refine their processes around development, releases and QA.
- Help to foster a strong engineering culture.

Shortly after joining GoHealth I also took on the management of a newly forming SRE team. Helping to bootstrap that team and give guidance on best practices for managing new and old deployments of applications in the GoHealth platform. I was also given oversight of the data engineering group around 4 years into my tenure as a Principal there.

VAE, Inc. - Senior Architect 2014 - 2016

The software stack we used at VAE was C# running on both Mono and .Net runtimes. It is a suite of network engineering tools used by Network Engineers and Network Operations to manage and view the current state of their network. While there I helped to:

- Develop software engineering processes.
- Integrate tooling to ensure software quality.
- Provide high level design guidance for products.
- Mentor junior developers.

Google - Software engineer 2008 - 2014

I was part of the DoubleClick acquisition by Google and continued working on the Google Affiliate Network product for the first two years. Wrote most of the code to migrate our data to Google infrastructure and datastores. Launched the first integrations with Google systems.

I then worked on the Project Hosting product on <https://code.google.com/p/>. This product has since been discontinued by Google.

After Project hosting I moved into the Web Server team for the Search Page. I worked on tooling for the templating language and CSS/Javascript/HTML management. While there I worked on integrating a CSS compiler for the Templating engine. This support greatly improved the experience of styling search features and controlling the impact of CSS on a page.

I also worked on evaluating the HTML5 Web Components standards to see how they might improve the search page. This required reading the W3C standards, prototyping templating engine support for them, and running some experiments on the front page.

Pretty much all the technology at Google is custom but relies heavily on non-relational distributed datastores, and highly scalable distributed architectures.

DoubleClick/Performics Software Engineer 2007 -

Worked on the affiliate ad product for Doubleclick/Performics in several different initiatives. The web stack was Linux, Apache, Mod-Perl and used DB2/Oracle/MySQL for the database. While there I helped to:

- Champion and prioritize tasks for a migration from an IBM DB2 database to an Oracle Database
- Code and setup a unit test infrastructure for javascript development.
- Design and write a UI framework to improve development speed and maintainability of frontends to the reporting API.

Healthcom - SubContractor 2007

Worked on a health device information tracking service. The system received health information from various remote devices in patients homes. It tracked the data and events and graphed them for the healthcare providers. It also alerted based off of certain criteria for the data.

The system was written in Perl and consumed web services, ftp uploads, and encrypted network connections to gather the data. The data was stored in Postgresql. The web stack was Apache and Perl on Windows.

Technologies

This is an incomplete list of the technologies I've had experience with just touching the high points. I'm a generalist who goes deep in a few areas and with an emphasis on maintainability and longevity of the systems or code I'm involved with.

Non Relational Datastores

- * CouchBase
- * MongoDB
- * Bigtable

Relational Databases

- * DB2
- * Oracle
- * Postgres
- * Mysql
- * SQL Server

Languages

I know a fairly wide variety of languages not all of which I could use in production settings. Learning and even creating languages is something of a hobby for me. A partial list is here.

- * Rust
- * Go
- * Erlang
- * UCG
- * Javascript
- * Typescript
- * Java
- * Julia
- * Python
- * C#
- * C