

# PAUL SADAUSKAS

## ›SENIOR WEB APPLICATION ENGINEER

✉ EMAIL  
paul@sadauskas.com

🌐 WEBSITE  
blog.theamazinggrando.com

🐙 GITHUB  
github.com/paul

## Passionate about the craft of creating software

My passion is creating software that is a joy for fellow developers to both use and contribute to. I am motivated to hone my craft of software development, to learn as much as I can about better ways to accomplish complicated tasks, and to build tools that are elegant, robust, and reliable.

## ›NOTABLE PROFESSIONAL SKILLS

Full-Stack Web Application Development  
Ruby & Rails  
Web APIs/services, ReST, JSON+LD, Hydra  
Service Oriented Architecture & Distributed systems  
System Administration & Large-scale deployments, AWS  
Javascript, CSS, Ember.js  
Open Source Contributor  
PostgreSQL and other relational and non-relations DBMSs

## ›MAJOR OPEN SOURCE PROJECTS

### Resourceful

Implementation of a fully RFC2616-compliant HTTP/1.1 client in Ruby  
Includes support for advanced features of HTTP, including support for redirects and caching with pluggable backends  
Used to build other abstractions, such as object mappers for REST/Hypermedia APIs

### progress\_bar

A very simple implementation of a CLI progress bar, similar to the output of cURL  
Mentored contributors to keep thier pull requests small, focused, and well-tested  
It was originally intended primarily for personal use, but is now a dependency of several popular projects

### DataMapper

Heavily involved in design and implementation of DataMapper, and preparing for release of 1.0  
Responded to issues and pull requests, and mentored new developers to get involved  
Redesigned and implemented DataMapper Adapters interface  
Designed and implemented Migrations plugin

### Merb

Heavily involved in design and implementation of Merb, preparing for release of 1.0  
Implemented support for conditional GET requests (304 response/Etag)  
Implemented correct handling of HTTP Accept header including negotiating client and server priorities

### Collectd

Contributed the tokyo-tyrant collection plugin  
Contributed the write-http plugin, which will submit collected measurements via libcurl to an HTTP server

Additionally, I have made small code contributions to numerous other OSS projects, including Rails, Rack, Awesomewm, and Arch Linux.

## ›CONFERENCE TALKS

Forms Don't Have to be this Complicated Mountain.rb - Oct 2010

How HTTP Already Solved All Your Performance Problems 20 Years Ago MountainWest RubyConf - Mar 2010

Writing Adapters for DataMapper MountainWest RubyConf - Mar 2009

## ›BUZZWORDS

REST, HTTP, Ruby, Ruby on Rails, JSON, JSON+LD, Hydra, SOA, HTML, HAML, CSS, SASS, Javascript, Coffeescript, Ember.js, Backbone, Shell scripting, Bash, ZSH, Vim, PostgreSQL, MySQL, SQLite, Redis, Memcached, AWS, EC2, SQS, ELB, Apache, NGinx, Chef, agile, Git

## ›PROFESSIONAL EXPERIENCE

- Wrote a full-stack Rails application (Postgres, Ruby, CSS, JS) to provide testers with a one-time use email address
- Added features to the open-source QueueClassic and QueueClassicAdmin gems
- Troubleshooted performance issues relating to load from a "stampeding herd" effect from testers
- Prototyped a replacement for a core component of the UI in Ember.js

- Architected and implemented a queue/worker system in Ruby to import users' AWS Cloudwatch data into Librato Metrics
- Contributed to and coordinated feature development on the Librato open-source client libraries
- Drove greater adoption of hypermedia concepts in the Librato API, both external and internal
- Took charge of a website security audit and fixes to user-facing websites and APIs
- Evaluated various JavaScript frameworks to use in a new frontend dashboard

- Design and implemented hypermedia APIs for extracted micro-apps to communicate with the main app and each other
- Made upgrades to various aspects of the GitHub API, and communicated those changes to users
- Improved tooling for generation of documentation of the GitHub API

- Took part in upgrades and enhancements to the email newsletter and notification delivery system
- As a member of the API team, drove adoption of hypermedia APIs as part of a SOA refactor
- Helped refactor a large Rails app by following the principles of TDD and Martin Fowler's *Refactoring*

- Implemented a "ChatOps" continuous deployment system using Campfire and Chef
- Created from scratch the Chef cookbooks and recipes to deploy the entire Strobe infrastructure to AWS
- Set up a JenkinsCI cluster for continuous builds of Ruby/Rails, C++ and Clojure applications
- Assisted other team members in getting their projects in CI and Chef for deployment
- Helped implement a CDN/Caching proxy for customer assets in Clojure and Riak

- Rewrote an existing monolithic Perl web application into multiple Rails applications
- Architected and developed new applications to interact with the main apps by hypermedia APIs
- Accomplished an order of magnitude improvement in capability by implementation of HTTP caching
- Revamped procedures to ship and deploy applications in various languages to remote customer sites
- Implemented new features in various NoSQL databases, TokyoTyrant, MongoDB, and Redis
- Encouraged the team to adopt automated Unit and Functional testing, as well as agile methods
- Engaged in continuous deployment on non-cloud physical hardware at various customer locations
- Involved as primary architect of all layers of the stack, from Javascript UIs to database schemas

- On a contract basis, implemented numerous web applications in PHP and later in Rails
- Created specifications and designs from (sometimes ambiguous) customer and stakeholder requirements

## DOWNLOAD RESUME

[Markdown/txt](#)[PDF](#)[Source](#)