

---

## Keith Gable (ZiggyTheHamster)

---

Passionate software engineer with 19 years of professional experience across a wide array of tools. Led an open source project in middle school. Engineered and architected an industry-leading enterprise podcast hosting and monetization SaaS platform which Amazon acquired in 2021. Rubyist since 2004. Vintage computer enthusiast.

---

## Info

---

📧 ziggythehamster  
✉ keith@ziggythehamster.sh  
📧 @ZiggyTheHamster@ruby.social

---

## Work Experience

---

### ♦ ART19, an Amazon company

2021-06 - Present » Senior Software Development Engineer  
2014-06 - 2021-06 » Chief Architect

Responsible for the design, implementation, deployment, monitoring, securing, and management of an enterprise podcast content management and ad delivery platform used by some of the industry's largest shows and advertisers. This platform allows the real-time insertion of targeted advertisements and other pieces of content, which gives potentially every listener a totally unique listening experience. Led a small, agile team initially consisting of the CTO and myself to build a platform that handles billions of transactions per month on a Ruby on Rails monolith. We have since been acquired by Amazon and have grown to have a team of more than 50 people who I work with and oversee. Other technologies in use, which I drove the adoption of, are Terraform, Docker, Consul, Apache Flink, Apache Kafka, Prometheus, Grafana, Linux, PostgreSQL, AWS, Ember.js, and HAProxy.

### ♦ Gallo Digital, LLC

2014-06 - 2014-10 » Senior Software Engineer

Worked on Leverage, a platform for law firms to litigate mass action cases. Principally designed, developed, secured, and deployed several services that the larger application suite depends on, including a custom-built electronic signature service, which is utilized by case members to sign and fill out documents and by lawyers and paralegals to define which fields are required to be filled out by case members. The primary technologies in use were Ruby on Rails, PostgreSQL, Redis, and Elasticsearch.

### ♦ DecisionPoint Systems, Inc.

2013-06 - 2014-06 » Senior Developer and Solution Architect  
2011-05 - 2013-06 » Developer

Responsible for the design, implementation, deployment, and security of both our own SaaS products and custom-developed software applications that were done for customers under contract. Managed a team of developers. Designed RESTful APIs for customers and methodically worked to accurately define the requirements of our customers. Worked with customer engineering teams and their leadership to scope and design a solution that met their needs and budget. Utilized both cross-platform and native toolkits to develop mobile applications across a vast array of devices and markets. Primarily utilized Ruby on Rails to develop web applications both to develop our products and to deliver for customers. Migrated from physical on-premises and co-located servers to AWS. Managed this infrastructure on AWS. Assisted customers with migrating from their legacy systems to AWS, including changing from Solaris to Red Hat Enterprise Linux. Supported first and third party software on niche operating systems like CCI Multiuser DOS, Windows CE, Symbian, and QNX. Integrated with AS/400 and z/OS systems on behalf of customers.

---

## Highlights

---

Ruby • Linux/Unix/BSD  
AWS • Terraform  
Consul • Packer  
JavaScript • Kafka  
Chef • Scala  
PostgreSQL • MySQL  
Redis • SQLite • CSS  
HTML • CouchDB • Druid  
Flink • KiCad • GitLab  
Ember.js • C# • SystemD  
Semantic Web • InfoSec

## Talks & Presentations

### ♦ Lessons Learned

#### Running Sidekiq at Scale - RubyConf 2024

This talk went over some hard lessons I learned after more than a decade of using Sidekiq at scale. I went over some basic mistakes like sharing the Sidekiq database with the rest of my application as well as some issues that I only encountered once I was doing millions of jobs per hour. I did the talk in character and spent the rest of the conference in character as well.

#### ♦ Oklahoma State University Institute of Technology - Okmulgee

2009-02 - 2011-05 » Application Developer and Webmaster  
2008-02 - 2009-02 » Temporary Webmaster  
2006-11 - 2008-02 » Computer Support Student Worker

Managed, wrote, or maintained all internal and external University web applications and content. Produced text and video content for use on University websites. Modernized antiquated PHP-based website by rewriting the backend in Ruby on Rails and redesigning the site's pages using HTML5 and CSS3. Piloted the first mobile website in the Oklahoma State University A&M system. Maintained campus VMware vSphere infrastructure. Managed interns and work studies. Served as a sponsor for an on-campus club and a student group's Applied Research and Development project. Provided support for faculty, staff, and students with computer issues on campus. Supported critical infrastructure systems and performed field repairs of critical systems and networks. Assisted in the development and implementation of a workstation imaging solution. Wrote JCL and FOCUS on the MVS mainframe to more tightly integrate internal and external facing web applications with our MVS-based campus ERP (SunGard SCT).

---

#### Education

#### ♦ Oklahoma State University Institute of Technology - Okmulgee

2006 - 2009 » Bachelor of Technology, Information Assurance and Forensics  
2006 - 2008 » Associate in Applied Science, Information Technologies  
Option in Information Security  
2006 - 2008 » Associate in Science, Information Technologies

#### ♦ Indian Capital Technology Center - Sallisaw

2005 - 2006 » Cisco Certified Network Associate I and II  
Did not complete III and IV. The CCENT, which covers this curriculum, did not yet exist.

---

#### Selected Projects

#### ♦ Airport Special Service Request App

Architected an application, using Ruby on Rails, Ember.js, Node.js, Android, and WebSockets to enable a customer to more efficiently serve special service requests (unaccompanied minor, wheelchair to seat, wheelchair to gate, etc.) at John F. Kennedy International Airport. In the pilot, the customer was able to reduce the time that airport patrons were waiting on them by 80%, and was able to nearly eliminate situations where airport patrons were waiting on personnel who would never come, which had the possibility of affecting ground operations if patrons were unable to leave the aircraft without assistance.

#### ♦ ART19

Utilizing Ruby on Rails, Node.js, Apache Kafka, Ember.js, PostgreSQL, AWS, Redis, and many other technologies, I designed and built the majority of an industry-leading enterprise podcast hosting and monetization platform. The platform serves traffic for and performs analytics on more than a billion transactions per month, and does so with performance that surpasses that of third party ad servers we integrate with. Thousands of creative, ad sales, and other industry professionals use the platform to create, distribute, and monetize hundreds of thousands of hours of content with genres ranging from comedy to news and sports. We host many of the top companies in the podcasting market and help them deliver high quality content and advertising to listeners worldwide. Because of the limitations inherent in podcasting, it is challenging to architect systems which can work with this magnitude of audio data while maintaining ambitious performance goals. We filed a patent application

---

#### Certifications

#### ♦ Amazon Web Services Partner Network

2014-04 »  
AWS Technical  
Professional  
Accreditation

#### ♦ Apple

2014-05 »  
Apple Certified  
Associate - Mac  
Integration 10.9  
2014-05 »  
Apple Certified  
Support  
Professional 10.9  
2014-05 »  
Apple Certified  
Technical  
Coordinator 10.9  
2013-02 »  
Apple Certified  
Associate - Mac  
Integration 10.8  
2013-02 »  
Apple Certified  
Support  
Professional 10.8  
2013-02 »  
Apple Certified  
Technical  
Coordinator 10.8

#### ♦ CompTIA

2013-02 »  
Cloud+  
2011-11 »  
A+ (CE)  
2011-11 »  
Network+ (CE)  
2011-11 »  
Storage+ powered  
by SNIA  
2009-06 »  
Network+  
2006-05 »  
A+

(US20170054780A1), of which I am a listed inventor, for how our technology works. This was ultimately abandoned for business reasons that I am not privy to.

#### ♦ Inventory Tracking Application

This application, for a large cosmetics retailer, allowed store associates and managers to more easily perform inventory functions. Because the retailer had a very large capital investment in their existing equipment, it had to run on Windows CE 5 and 6. I was initially involved on the server side, which served a RESTful API to clients from a Ruby on Rails application, and integrated with the retailer's AS/400 back-office systems. After a catastrophe involving the engineer tasked with the C#-based mobile application which should have been caught by their manager but wasn't until we entered UAT and discovered that they had been faking network connectivity by manually updating all of the screens with each day's test data and discarding the API call responses, I was brought in and tasked with rescuing the project. I was able to deputize a few of my team's engineers to help, and we were able to essentially rewrite the application in about a month. This did not delight the customer as much as I had hoped, because we were transparent about the dishonesty, but we were able to pilot the application at a dozen stores in the US and Canada. I left the company before we shipped it to all of the stores, but I observed associates in a store using the application a couple years later, so they must have.

#### ♦ Leverage

Leverage is a web application, written in Ruby on Rails and using Redis, PostgreSQL, and Elasticsearch, that allows law firms to litigate mass action cases when class action cases are not possible or appropriate. Without software that streamlines working with potentially thousands of similar -- but different -- cases, there are many cases that are not economically viable to pursue, leaving those that are wronged without any viable recourse. The largest case that I am aware of that used this application was related to the Volkswagen Diesel Emissions scandal, and played an important role in owners of affected vehicles (including myself) being fairly compensated for VW's dishonesty around "TDI Clean Diesel" vehicles. I joined the company when Leverage was less mature and was unable to handle this kind of scale, and I was able to scale the architecture out while reducing the per-litigant operational cost. I also built a few key components, including a service that allowed litigants to e-sign documents which did not depend on any third party service, a technical requirement caused by privacy and security concerns that lawyers at the firm that sponsored this application raised.

#### ♦ SentralCommand

This product, written in native Android, iOS, Windows Phone, and Ruby on Rails, presents design staff with a canvas in their web browser with which they can create many kinds of mobile applications. This product was most notably used by The Richards Group to deliver the TGI Friday's consumer application on iOS and Android. The front-end was my first Ember.js application, and I worked on it in collaboration with one other engineer.

---

END