

# ERIC WALSH

SOFTWARE ENGINEER   📍 LONG BEACH, CA   📞 7143515785

## ◦ DETAILS ◦

Long Beach, CA  
7143515785  
[ericmatthewwalsh@gmail.com](mailto:ericmatthewwalsh@gmail.com)

## ◦ LINKS ◦

[ericmwalsh.com](http://ericmwalsh.com)  
[github.com/ericmwalsh](https://github.com/ericmwalsh)  
[linkedin.com/in/eric-walsh-31552250/](https://linkedin.com/in/eric-walsh-31552250/)

## ◦ SKILLS ◦

ETL/OLAP

Programming

Databases

Infrastructure

Leadership

## ◦ HOBBIES ◦

- Racing Cars
- 3D Printers
  - Travel
  - Cooking



## PROFILE

I am searching for an engaging position in software engineering and development where a strong background in programming, architecture, and leading projects can be used to improve operations.



## EMPLOYMENT HISTORY

### Senior Software Engineer at Affirm, San Francisco

November 2022 — January 2025

Originally hired as a Senior Software Engineer and then promoted to Staff Software Engineer for the Storage and Replication team within the DSS (Data, Storage, and Services) department.

My main responsibilities were building and maintaining our data replication pipeline and managing our Snowflake instances. My work consisted of a mixture of data, infrastructure, and software engineering:

- My team was responsible for taking all "live" SQL (MySQL, PostgreSQL) and DynamoDB data and replicating that data to AWS S3, Apache Iceberg, and Snowflake so that the replicated data could be used for analytics and data science purposes.
  - I led a multi-team effort to build out systems to obtain CDC records and entire DB snapshots of source data. We utilized AWS DMS (Database Migration Service) to obtain SQL CDC records and AWS Kinesis streams in order to obtain DynamoDB CDC records. We obtained DB (both SQL and DynamoDB) snapshots using Python CRON jobs via Apache Airflow 2.0. All of this replicated data was stored in AWS S3 and all of the infrastructure was defined in Terraform.
  - I led our team of developers in creating the pipeline that migrated all of the replicated AWS S3 data to Apache Iceberg and finally to Snowflake. The pipeline was built on top of Apache Airflow 2.0 and utilized Luigi (Python) for batch job processing.
  - I led the team that was responsible for setting up all our replication and analytics services/infrastructure in our EU business region. We debuted our brand new EU region support in Q4 2024 by launching Affirm services in the UK.
- My team was also responsible for managing our Snowflake instances which included administration (creating/deletion/support/management) and permissions controls.
  - I led the process of integrating our Snowflake instances with OKTA for easier internal access provisioning. I also led the process of integrating our Snowflake instances with SailPoint in order to allow for automated permissions control (instead of having to manually approve hundreds of permissions requests a week). In order to accomplish this we created a wrapper on top of the (Snowflake) Permifrost library which allowed us to manage permissions and SQL grants programmatically.
  - I worked with the analytics and data science teams to setup the Looker integration and Monte Carlo integration. I also set up the logging pipeline that delivered metrics and logs to CloudWatch/ElasticSearch.

<https://www.affirm.com/>

## **Software Engineer (SDE II) at Amazon Music, Culver City**

July 2020 — June 2022

I transferred internally within Amazon from Ring to Amazon Music. When I joined Amazon Music I worked in the newly formed Podcasts department as a member of the Podcast Search team. Amazon Music Podcasts did not go live until the end of 2020: I helped build out the original podcast search aggregation pipelines and APIs and was also part of the production launch.

During my time at Amazon Music:

- I helped built out the podcast aggregation ETL pipeline that ingested podcast data from millions of different sources and then transformed and loaded that data into our search back end system. The pipeline was built using Java (across AWS SQS, Lambdas, and ECS) and the ingested podcast data was stored in S3 before being transformed and loaded into Redshift, DynamoDB, PostgreSQL, and ElasticSearch.
  - All of the infrastructure we created was expressed in code (using AWS Cloudformation) and each of our services were split up into separate AWS accounts for security/isolation purposes.
- I joined the team prior to public launch and helped build out integral parts of the search APIs. After Podcasts became public I led the team that continued feature development and scaling for the search services/APIs. We saw month over month increases in listening and search usage over the next year and a half while I led the team.
- In order to improve the quality of our search results I worked with multiple internal teams to build out our Podcast metric reporting and aggregation system. Using the various engagement metrics (clicks, listens, plays, etc.) we were able to calculate a rough "popularity score" to use to assist in ranking search results. We worked through numerous rounds of A/B experiments to optimize the calculation of this search popularity score. Ultimately this persistence led to higher quality search results and we saw a noticeable increase in podcast search conversion percentage for the platform.

<https://music.amazon.com/>

## **Software Engineer (SDE II) at Ring (Amazon), Santa Monica**

November 2018 — July 2020

I worked as a senior software engineer on the Ring Customer Service engineering team. My main responsibility was building and maintaining our internal customer support tool (used by our customer service phone agents in our call centers) using Ruby on Rails and Java hosted on AWS. I traveled to our call center locations (in Arizona and Philadelphia) once a quarter in order to meet with the our customer service team in order to gather feedback and plan out new features for the customer support tool.

I also worked with the Ring Payments team to create the Ring customer service payment system using Amazon Connect (IVR) and AWS Lambda (using Java/DynamoDB). This payment system allowed our customer service agents to take secure payments over the phone by having customers say or type their card number in rather than relaying it over the phone to an agent.

<https://ring.com/>

## **Senior Software Engineer (SE4) at TrueCar, Santa Monica**

October 2016 — November 2018

I worked as the senior back end engineer in the ALG department at TrueCar (who had owned ALG at that time). *ALG (Automotive Lease Guide, focuses on vehicle residuals) is currently owned by JD Power.*

My initial responsibility was converting over legacy internally-hosted Java APIs to new Ruby on Rails APIs hosted on AWS. After finishing the conversion project I spent the next year working with the Data Engineering and Data Science teams to build an internal

price forecasting tool called Stormwatch. My main responsibility consisted of building out an ETL pipeline for car purchase prices using data sourced from dealerships, banks, and other financial institutions. This data was obtained from the different vendors on a regular basis and transformed to a master format before being loaded/stored in AWS S3, AWS Redshift, and finally made accessible to analysts via Tableau.

I led a team of engineers building out the APIs on top of the transformed purchasing data so other TrueCar teams could leverage and integrate the pricing data into different TrueCar tools.

<https://www.truecar.com/>

<https://www.jdpowervalue.com/alg-automotive-insights-outlook>

### **Software Engineer at HYFN, Hermosa Beach**

February 2016 — September 2016

Originally hired as a Software Engineer and then promoted to Lead Software Engineer for the Twitter Ads team on the HYFN8 project.

I helped build the Twitter integration for our internal tool social media marketing tool known as HYFN8. This tool was written in Ruby and Python with a MySQL DB and allowed our internal advertising team to manage social media ad campaigns (for major brands like Nike, Taco Bell, and Doritos) across multiple social media platforms. I led the team that built the Twitter integration which leveraged the Twitter API/Twitter Ads API.

<https://www.linkedin.com/company/hyfn/>

### **Software Engineer at Spinlister, Santa Monica**

December 2014 — February 2016

I was one of the three original software engineers that built and maintained Spinlister after it re-opened in 2013 as a tech startup. I worked as a full stack engineer building out the (customer-facing) website using Ruby on Rails. I also built out the Ruby on Rails APIs that powered our web app and native iOS and Android applications. One of the most important projects I completed before leaving was splitting up our monolithic Ruby on Rails app (hosted on Heroku) into a separate Rails back-end API with an EmberJS front end application component (all hosted on native AWS).

<https://en.wikipedia.org/wiki/Spinlister>

### **Software Engineer at America's Printer, Buena Park**

June 2014 — September 2014

I was hired as a software engineer to work on a small internal team (three engineers) where I spent my time maintaining and developing two Ruby on Rails projects:

- an internal CRM that tracked print orders from start to finish (lead generation, order creation, production life cycle, order fulfillment)
- a customer facing site that allowed customers to place orders for custom printing products and track their orders

<https://americasprinter.com/>

### **Software Engineer at Broadview Mortgage Corporation, Orange**

June 2013 — June 2014

Originally hired as a intern and then promoted to full time software engineer after three months.

I helped build and maintain an internal tool for mortgage loan application management using PHP and Drupal 7. I also built single page applications with a small team using LAMP and Ruby on Rails.

<https://www.broadviewhl.com/>