

David Zaebst

SOFTWARE ENGINEER

☎ (740) 412-7333 | ✉ zaebst@gmail.com | 🏠 www.zaebst.com | 📱 zaebst

Skills

Programming Languages	Scala, Ruby, JavaScript, Java, Python
Frameworks	Spark, Ruby on Rails, React, Django
Databases	MongoDB, Elasticsearch, Redis, PostgreSQL, Redshift, Hive
Infrastructure	Ansible, AWS, Terraform, Vault, Consul
Operating Systems	Debian, OS X, Ubuntu

Work Experience

Transmit.Live

New York, NY

VP ENGINEERING

July 2017 - Present

- Built out Transmit Live's social broadcasting and analytics platform as the second engineering hire.
- Designed and implemented ETL for collecting and processing advertising and live streaming metrics.
- Wrote Spark Streaming and Batch processing jobs in Scala to load time series data into MongoDB.
- Worked with Kinesis producers and consumers to collect and persist data from our ad servers.
- Created post-broadcast reporting pages in React and JavaScript ES6 with Excel downloads and D3.js graphs.
- Developed OAuth2 workflows authorizing Transmit Live to schedule and broadcast live video feeds to Facebook, YouTube, and Periscope.
- Introduced websockets with Django channels and Redux middleware to push live streaming status to clients.
- Integrated and mapped external models for broadcasts on social platforms to internal web application in Django REST Framework API backed by PostgreSQL.
- Deployed an Elasticsearch cluster, Filebeat collectors, and Kibana for centralized logging. Handled upgrade from ES 5.5.1 to 6.2.4.
- Configured infrastructure with Terraform, using it to launch ec2 instances, manage security groups, create load balancers, and update DNS records.
- Wrote continuous integration tests using Jenkins and Docker.

Run / Publicis

New York, NY

VP DATA ENGINEERING

May 2013 - May 2017

- Grew a small successful startup as the fifth engineering hire.
- Hired for and managed the Data Engineering team that prepared Run's advertising platform for increased volume after the Publicis acquisition.
- Took ownership of the ETL pipelines and databases used for time series reporting on ad delivery, reconciling client budget against current spend, and billing clients.
- Handled high priority items key to signing new clients, generating revenue, and leading to acquisition. Examples include an API that allowed clients to download their ad serving logs, cost per complete view charging, and ad visibility reporting.
- Rewrote the main Hive ETL workflow in Spark's Scala API. This pipeline ingested hourly json event files and inserted aggregated ad delivery statistics into MongoDB and Elasticsearch. Scaling and stability were critical to meeting client expectations and managing budgets.
- Established team standards for testing, build tools, deployment procedures, code reviews, and git workflows.
- Organized the JIRA queue, wrote tickets, and prioritized items for the Data Engineering team.
- Tuned Spark jobs by identifying tables to be broadcast and ways to partition data.
- Researched and implemented a sharding scheme for MongoDB statistics database.
- Deployed a sharded MongoDB cluster and an Elasticsearch cluster. Each database held more than 2Tb of ad delivery data and delivered on queries from the website with minimal latency.
- Managed backups on the MongoDB and Elasticsearch clusters.
- Ported budget management code for a high throughput http server from Node.js to Java with Netty. Used threads and concurrent data structures to reduce database write load.
- Upgraded databases and database drivers from Mongo 2.x to Mongo 3.0 and from Elasticsearch 0.9 to Elasticsearch 2.4. Code reviewed the upgrade to Mongo 3.4 drivers.
- Worked on a modern Rails stack including Backbone.js, Bootstrap, Phusion Passenger, Sidekiq, Nginx, Redis, and MongoDB.
- Made application endpoints to serve time series data and connected them to the rickshaw JavaScript library to graph ad delivery over different time intervals.
- Upgraded the portal, the site for managing advertising campaigns, from Rails 3 to Rails 4.
- Added features to the portal including VAST companion ad units, bulk advertisement import, and reach and frequency graphs.
- Designed ad server log ingestion pipeline which took the ad serving records in json, transformed them using Scala and scalding and imported the results into a Redshift database optimized for joining advertising user events.
- Developed original click through rate optimization system based on Bayesian statistics using Hive and Ruby.
- Wrote the first implementation of reach and frequency. This calculated the unique people who saw an ad over different time ranges, the average number of times a person would see the ad, and displayed the results in the graphs in the portal.
- Ensured code quality by testing with RSpec, JUnit, ScalaTest, sbt, gradle, and CircleCI.
- Monitored application and server performance with statsd and datadog.
- Used standard tools to track down production issues: pry, chrome developer tools, tcpick, screen, nmap.
- Leveraged Ansible to manage database deployment, VPC setup, and server configurations.

Sumitomo Mitsui Banking Corporation

Jersey City, NJ

LINUX SYSTEMS ADMINISTRATOR

June 2011-August 2012

- Acquired in depth knowledge of systems administration in a highly secure banking environment.
- Developed Perl libraries as CPAN style modules and packaged modules in RPM format. Libraries were designed to lookup the user id specific to an application and environment, initialize the shell environment, and launch processes as daemons.
- Reduced application start time for the deposit processing server from 27 minutes to 6 minutes by launching processes in parallel.
- Gathered requirements and implemented application verification procedures that were manually run by operations.
- Coordinated deployment, network, and security configuration for applications.
- Utilized common diagnostic tools such as strace, trace, gdb, pstree, lsof, wireshark, and netstat.

Morgan Stanley

New York, NY

SENIOR SYSTEMS ANALYST

August 2009-February 2011

- Maintained highly available, distributed, low latency trading systems.
- Developed web application to streamline trader setup, removal, modification and assignment of trading permissions.
- Improved regulatory compliance procedures by generating daily reports comparing trader entitlements to their expected authorizations.
- Designed procedure to update financial products, which processed new listings and delistings, then loaded records into the product cache.
- Implemented trade resubmission program to resubmit rejected trades with corrections.
- Created command line tools to find, modify, and cancel orders in a distributed trading system.

Bear Stearns / J.P. Morgan

New York, NY

LINUX BUILD ENGINEER

October 2007-February 2009

- Mastered the complex process of compiling, deploying, and debugging compiled software.
- Maintained XS modules that connected Perl to C bond pricing libraries allowing developers access to optimized C code from an easy to use Perl interface.
- Increased stability and accountability for builds by setting up a continuous integration server.
- Improved developer productivity with an internal website used to track down undefined symbols.

Education

The City College of New York

New York, NY

M.A. IN PHYSICS

Degree Conferred February 2013

Ohio University

Athens, OH

B.S. IN APPLIED MATHEMATICS AND B.A. IN PHYSICS

Degree Conferred June 2006