



NOW LEARNING

Software Engineering Management, Data Science, GCP, Kubernetes  
Operations without high-level tooling, Preact, Kotlin.

“I have experience in a lot of technologies; however, that isn’t as interesting as the fact that I understand the following at a conceptual level: The HTTP protocol, regular expressions, pragmatic test driven development, when, why, and how to use design patterns, most design patterns exist to work around mostly OOP language gaps, shell exit codes and assert(), the unix/linux shell environment, database indexing vs inverted indexing and materialized views, hash tables, hash functions and their relationship to inverted indexes, and finally, the relationship between Linux containers (LXC) and Docker. Almost everything else today is an implementation detail and a convenience on top of these concepts.”

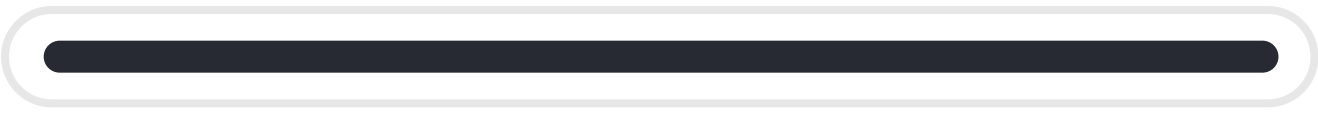
PROGRAMMING LANGUAGES

Languages I’ve shipped to production, refactored to another language, authored an open-source library for, or I am actively learning.

JAVASCRIPT (V8/NODE.JS, BROWSER/DOM)



JS COMPILED (TYPESCRIPT, BABEL, COFFEESCRIPT)



PYTHON



RUST



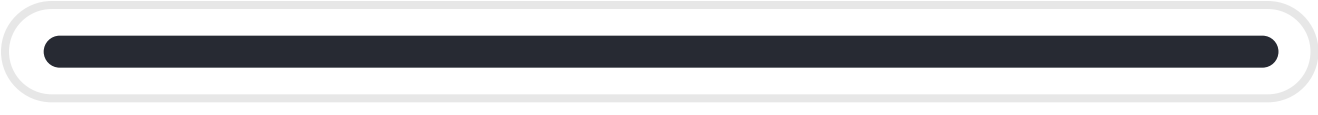
ERLANG



PHP



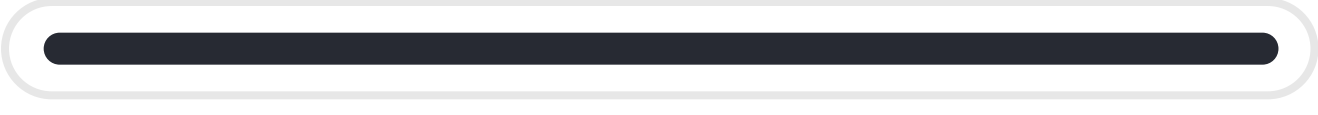
ELIXIR



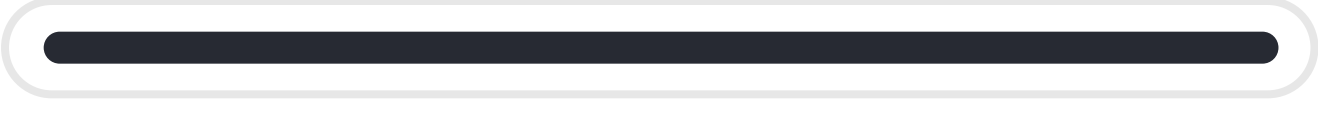
KOTLIN



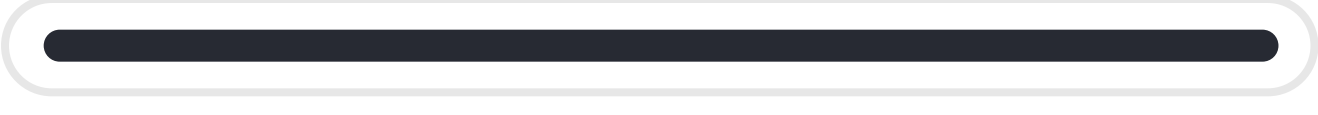
GO



JAVA (VERSION 8)



AWK / SED



SQL (DDL/DQL/DML, SPROCS, TRIGGERS, VIEWS)



SHELL SCRIPTING (SH, BASH, ZSH, FISH)



MARKUP / STYLE (HTML/5, CSS/3, SASS, LESS, JSX)



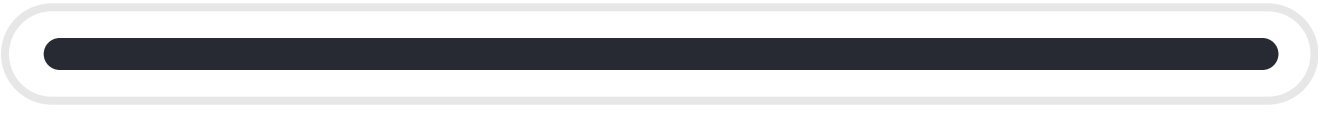
DATABASES

Databases I’ve deployed and/or managed in production scenarios, or I am actively learning.

RDBMS (POSTGRESQL, MYSQL, SQL SERVER, ORACLE)



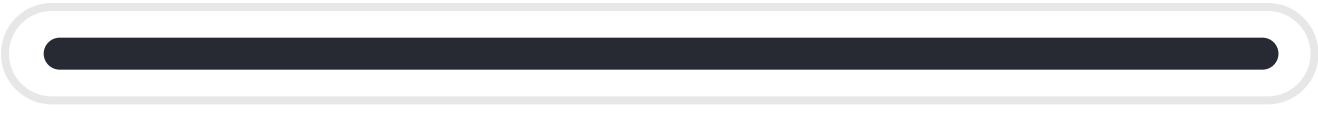
NOSQL (DYNAMODB, REDIS, MONGODB)



GRAPH (NEO4J)



SEARCH (ELASTICSEARCH)



Wilbert (Wil) Moore III

POLYGLOT SENIOR SOFTWARE ENGINEER, DEVOPS, SRE



github.com/wilmoore



linkedin.com/in/wilmoore



wil.moore@polyglotsoftware.io

WORK EXPERIENCE

My contract and direct hire experiences with detailed depictions of notable projects I’ve delivered and in some cases, the reasons behind the tech decisions.

Senior Full-Stack Software Engineer, devOps, SRE (CONTRACT) MAR 2018 - FEB 2019  
Nordstrom - Denver, CO Insight Global

[API ARCHITECTURE + IMPLEMENTATION, BACKEND, DEVOPS] Designed and implemented a near real-time messaging system with message fanout arriving via SSE. SSE (EventSource) was choosen over Websockets as my experiments and research revealed that SSE is superior in simplicity of debugging, testing, and deployment over Websockets, especially when client-to-API two-way messaging is not needed. Deployment is automated via Terraform and CloudFormation. The RESTful Swagger API (OpenAPI) written in YAML format is fronted by AWS API Gateway and proxies to AWS Lambda which persists messages to DynamoDB for long-term storage, which subsequently emits an event to a DynamoDB Stream and is picked up by an AWS Lambda function which produces to a Kafka topic. The Kafka topic is watched by a Node.js worker process and eventually fanned out to clients via SSE.

[FULL-TEXT SEARCH SCHEMA DESIGN, IMPLEMENTATION, BACKEND] Designed and implemented an Elasticsearch schema and cluster for a search/filtering strategy which replaced an existing Redis cluster which was previously merging results in-flight which was less than efficient and not very flexible in terms of agility or surviving change. The Elasticsearch cluster was deployed in AWS on EC2 instances. I crafted and documented several Elasticsearch queries per the team’s requirements as well as delivered performance metrics via Artillery-based load testing. Integrated a simple Elasticsearch HTTP client along with the queries into an existing Node.js/TypeScript API (not originally maintained by me or my main team – helped another team out with this implementation).

[DB SCHEMA DESIGN, DB INDICES, DEVOPS, BACKEND] Contributed various production features and enhancements to a Node.js Apollo GraphQL API deployed to AWS ElasticBeanstalk fronted by an Nginx load balancer in a VPC. I increased response throughput by +3x (at least) by way of adding database indices and pagination in hot portions of the source. The performance enhancement was verified with pre and post load testing with Artillery. Wrote custom express middleware as a provider for mocking the calls to the Node.js GraphQL API for Cypress.io testing a React front-end application.

[DOCUMENTATION, DEVOPS, SRE] Authored sections of a Docker/Kubernetes deployment tutorial. Authored an incident response guide: a guide for on-call SREs which included examples of previous incident responses along with screenshots. Built dashboards, monitors and alerts in Datadog and Kibana, with integration to PagerDuty and automated with terraform.

Senior Full-Stack Software Engineer (CONTRACT) OCT 2017 - MAR 2018  
Charter Communications - Greenwood Village, CO Insight Global

[API ARCHITECTURE + IMPLEMENTATION, BACKEND, DEVOPS] Designed and implemented micro-service APIs to automate test account creation through orchestration of several internal APIs with Node.js and the IBM/StrongLoop Loopback API framework. Developed custom Loopback middleware and mixins for uploading and persisting attachments in base64 format.

[DB SCHEMA DESIGN, DB INDICES] Using raw SQL DDL, designed and created MySQL database schemas to support new features and enhancments in the API.

[DOCUMENTATION] Authored a new developer onboarding guide which covered getting started with contributing to the project, getting around the company’s campus, and how to setup the VPN.

Senior Full-Stack Software Engineer (CONTRACT) FEB 2016 - SEP 2017  
Shutterstock - Denver, CO Sharp Decisions (formerly CN-Tec)

[API ARCHITECTURE + IMPLEMENTATION, DB SCHEMA DESIGN, DB TUNING] Maintained and refactored an e-commerce Node.js API on top of the express framework backed by a MySQL database using test-driven development (TDD). Implemented a currency exchange API with Ruby and the Sinatra framework backed by a MySQL database using test-driven development (TDD). Contributed to Node.js orchestration APIs and built JavaScript API clients using test-driven development (TDD), including npm packaging and release into a private npm registry. Evolved the schema of a MySQL database through features and enhancements through DDL and DML SQL change scripts. Introduced a message queuing strategy through use of the RabbitMQ message broker. Built architecture diagrams and facilitated architecture reviews, and authored developer guides.





NOW LEARNING

Software Engineering Management, Data Science, GCP, Kubernetes  
Operations without high-level tooling, Preact, Kotlin.

# Wilbert (Wil) Moore III

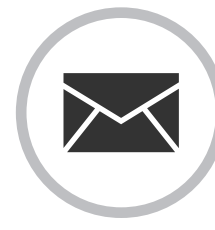
POLYGLOT SENIOR SOFTWARE ENGINEER, DEVOPS, SRE



github.com/wilmoore



linkedin.com/in/wilmoore



wil.moore@polyglotsoftware.io

“Unfortunately, many software developers suffer the following (I must concede that I’ve at times struggled with this as well): we sometimes fall in love with technologies, languages, and frameworks which can cause tunnel vision. **Even more unfortunate is the fact that many hiring officials have and continue to blindly dismiss great candidates on the premise of one easy to train technical gap; these people may posses more passion for the product than even the hiring official, and these gems, these excellent people are dismissed before they can shine their light ... and this over some slight gap in experience with an esoteric tool which could be trained up on in a few hours for a few dollars...sorry, but you can’t train a lack of passion for the cause.**”

## CLOUD TECHNOLOGIES

Technologies that I’ve worked with in the cloud from the perspective of architecture as well as implementation.

COMPUTE (AWS EC2/ELASTICBEANSTALK, HEROKU)



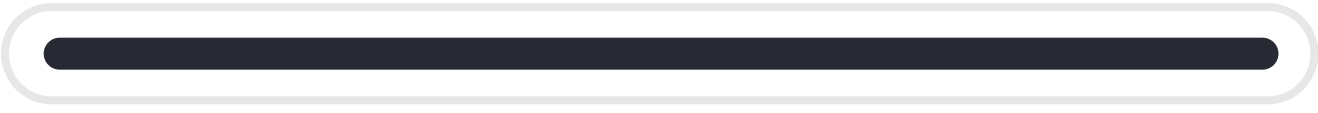
FUNCTIONS (AWS LAMBDA, SERVERLESS FRAMEWORK)



NOSQL (AWS DYNAMODB, ELASTICACHE)



RDBMS (AWS RDS POSTGRESQL & MYSQL)



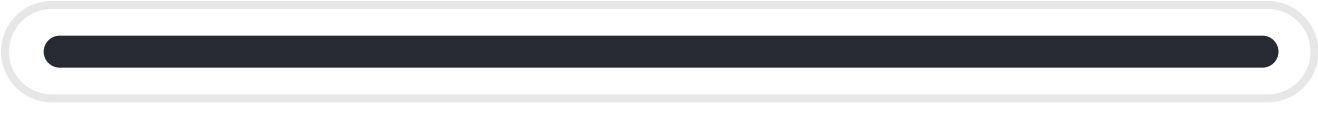
STORAGE (AWS S3, RACKSPACE CLOUD FILES)



QUEUE (AWS SQS, RABBITMQ)



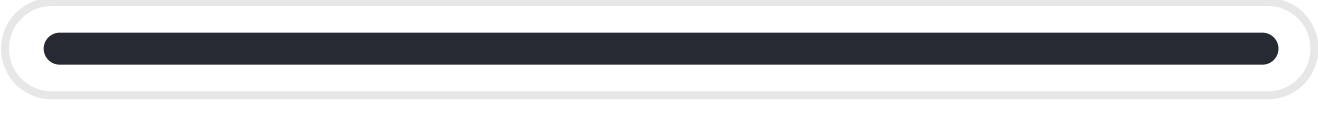
MESSAGING (AWS SNS)



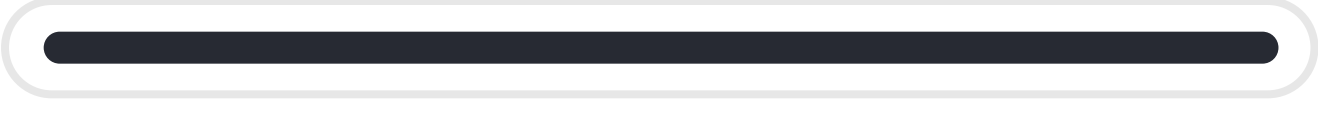
DATA WAREHOUSE (AWS REDSHIFT)



CONFIGURATION (TERRAFORM, AWS CLOUDFORMATION)



LOAD BALANCING (AWS ELB)



DNS (AWS ROUTE 53)



CDN (AWS CLOUDFRONT)



MONITORING (AWS CLOUDWATCH, DATADOG, KIBANA)



STREAM (AWS KINESIS, KAFKA)



NETWORK (AWS VPC)



## WEB SERVICES

Web service (or micro-service) technologies in which I’ve implemented the producing side (the service itself) and/or a consuming client.

REST & RESOURCE-BASED SERVICES (OPENAPI/SWAGGER)



GRAPHQL (APOLLO-BASED & VANILLA NO-FRAMEWORK)



SOAP (CLIENTS)



## WORK EXPERIENCE

My contract and direct hire experiences with detailed depictions of notable projects I’ve delivered and in some cases, the reasons behind the tech decisions.

Senior Full-Stack Software Engineer  
Aetna (formerly iTriage) - Denver, CO

(CONTRACT) MAY 2015 - FEB 2016  
Sharp Decisions (formerly CN-Tec)

[API ARCHITECTURE + IMPLEMENTATION, BACKEND, DEVOPS] Implemented and deployed multiple **RESTful APIs with Swagger, Node.js, Hapi, and Docker**. Contributed to **analytics client libraries** writing in **JavaScript** and **Objective-C**. Contributed to a **Ruby/Sinatra** RESTful API. Contributed to an analytics platform built on top of **AWS Redshift**. Built several reusable project **templates** and **boilerplates** for quickly **scaffolding** backend **APIs** including documentation with careful consideration for developer experience (**DX**). Service instrumentation with StatsD, end-to-end testing with WebdriverIO/Selenium. Performed **R&D prototyping** with **Elasticsearch** for a project that unfortunately never materialized.

[FRONT-END] Implemented a browser-based **single-page-application (SPA)** secure messaging dashboard application utilizing the Angular framework.

Senior Full-Stack Software Engineer  
Welltok - Denver, CO

(CONTRACT) AUG 2014 - MAY 2015  
Sharp Decisions (formerly CN-Tec)

[API ARCHITECTURE + IMPLEMENTATION, BACKEND, DEVOPS] Designed and implemented Several micro-service APIs with Node.js, Express which powered the core platform for CafeWell Concierge. Concierge which was comprised of a native mobile application for health care benefits recipients and a web application for administration. APIs were deployed via **Docker** to **AWS ElasticBeanstalk** with some assets being deployed to **AWS S3** and ditributed through **AWS CloudFront**. The administration dashboard was written in **Ruby/Rails** and the **Rails Admin** package with deployment to **AWS ElasticBeanstalk** with **Docker**.

[SEARCH / AUTO-COMPLETE] Utilized **Elasticsearch** to build a search / auto-complete service which was exposed via a **Node.js RESTful API** using the **Koa** framework. The **auto-complete** service was consumed by the Concierge mobile application.

[EXTERNAL SERVICE API INTEGRATIONS] As Concierge was meant to be a “cognitive” application, there was some integration with IBM Watson (cognitive computing engine that that won Jeopardy).

Designed and implemented a **RESTful API** using the **Node.js Koa framework** with integration with the **Active.com API** (http://developer.active.com/docs/read/v2\_Activity\_API\_Search). Packaged as a **Node.js module, Open-sourced the code**, and published to the **npm registry** (https://www.npmjs.com/package/active-activity-search).

[ETL] Migrated a legacy Python **ETL** project that was not written for scale. Re-implemented the ETL project as **data transformations** powered by **Node.js** streams. **Packaged as a Node.js module, Open-sourced the code, and published to the npm registry** (https://www.npmjs.com/package/text-pipe).

Senior Front-End Software Engineer  
Time Warner Cable - Broomfield, CO

(CONTRACT) AUG 2012 - MAY 2015  
TEKsystems

Built an **HTML5 single-page-application (SPA)** and **HLS streaming video streaming video player** with custom scrubbing features. This system was targeting a **whole-home IP TV** (cable set-top box) system targeting millions of customer homes. Utilized **test-driven-development (TDD)** to implement and deliver features, bug-fixes, enhancements, and performance tuning.

**CI** & and releases using **Puppet, VirtualBox, and Jenkins**. Unit/Integration testing w/ **Mocha, Jasmine, Sinon, Chai, and PhantomJS**. Build automation with **GNUMake** and **continuous integration** with **Jenkins. Node.js** for utility scripting, testing, and backend stubbing.

Designed & Developed various HTTP API clients using **jQuery AJAX, Superagent, Promises**, higher-order functions, and “README”-driven development. API clients for retrieving and playing HLS streams as well as clients for retrieving emergency (**EAS**) notifications and application updates. Originally utilized **WebSockets** which were switched to **Server Sent Events, Mocha, Grunt, Gulp, and GNUMake**.





NOW LEARNING

Software Engineering Management, Data Science, GCP, Kubernetes  
Operations without high-level tooling, Preact, Kotlin.

# Wilbert (Wil) Moore III

POLYGLOT SENIOR SOFTWARE ENGINEER, DEVOPS, SRE



github.com/wilmoore



linkedin.com/in/wilmoore



wil.moore@polyglotsoftware.io

“After about your **4th** or **5th programming language, testing framework, cloud service, template engine, or relational database**, you start to realize, **ramp up** on new technologies **no longer takes years or months** ... it takes **minutes** to get the **general concept**, **days** to **build something** useful, and **weeks** to start **working at expert level**.”

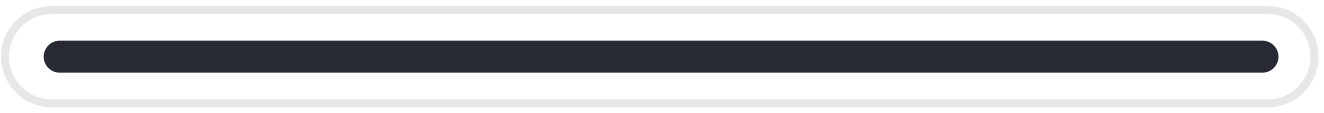
## TEST FRAMEWORKS

Automated test frameworks that I have used and feel comfortable writing new or modifying existing test cases.

JS (TAPE, MOCHA, JASMINE, TESTEM, KARMA, JEST)



RUBY (MINITEST, RSPEC)



PHP (PHPUNIT)



## API FRAMEWORKS

API libraries and frameworks that I've worked with.

NODE.JS (EXPRESS, KOA, FASTIFY, CONNECT, HAPI)



NODE.JS (APOLLO GRAPHQL, STRONGLOOP LOOPBACK)



PYTHON (FALCON)



RUBY (SINATRA)



## WEB & MOBILE FRAMEWORKS

API libraries and frameworks that I've worked with.

REACT, REACT NATIVE



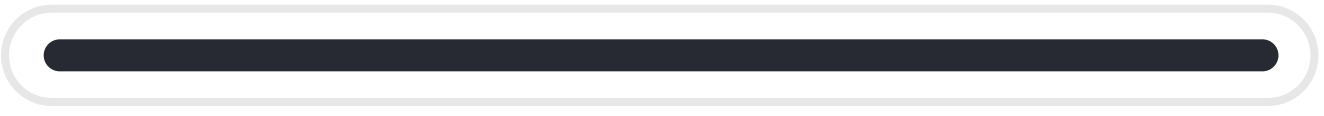
VUEJS



ANGULAR



BACKBONE



## CI/CD

Continuous Integration and continuous deployment systems that I've used, setup, and/or maintained.

JENKINS



GITLAB CI



SEMAPHOR, CIRCLE, TRAVIS



## WORK EXPERIENCE

My contract and direct hire experiences with detailed depictions of notable projects I've delivered and in some cases, the reasons behind the tech decisions.

### Senior Full-Stack Software Engineer

DEC 2011 - AUG 2012

Net-Results - Golden, CO

[FRONT-END, BACK-END, UI/UIX, DEVOPS] UI/UX design and implementation for a multi-tenant browser-based **single-page-application (SPA)** for customer **cloud file management** interface on top of **AWS S3** with a **PHP 5.3** backend.

[FRONT-END, BACK-END] Contributed to a **REST API** for **account provisioning** and client maintenance with **jQuery** and backed by **MongoDB**. Contributed to a **multi-tenant** browser-based **single-page-application (SPA)** for building marketing landing pages using **drag-and-drop** using HTML5, CSS3, and **YUI3**.

### (Team Lead) Senior Full-Stack Software Engineer

JUN 2010 - AUG 2011

Navigant (formerly Ethos Partners) - Denver, CO

[API ARCHITECTURE + IMPLEMENTATION, BACKEND, DEVOPS] Lead a **team of 3** developers in successfully re-architecting a legacy monolithic web application that **allowed health care providers create health risk assessments (HRA)** with custom business rules, **branching**, and demograph intake.

The legacy version included no architecture diagrams, no documentation, 1% test coverage, and an abyss of poorly designed MySQL database tables which were missing key indices. The original software was written with PHP 4.x which my team replaced with **PHP 5.x** using the **Zend Framework**, **95% test coverage**, an increase of **94%**.

### Senior Software Engineer

(CONTRACT) AUG 2012 - MAY 2015

Time Warner Cable - Broomfield, CO

TEKsystems

Built an **HTML5 single-page-application (SPA)** and **HLS streaming video streaming video player** with custom scrubbing features. This system was targeting a **whole-home IP TV** (cable set-top box) system targeting millions of customer homes. Utilized **test-driven-development (TDD)** to implement and deliver features, bug-fixes, enhancements, and performance tuning.

**CI** & and releases using **Puppet**, **VirtualBox**, and **Jenkins**. Unit/Integration testing w/ **Mocha**, **Jasmine**, **Sinon**, **Chai**, and **PhantomJS**. Build automation with **GNUMake** and **continuous integration** with **Jenkins**. **Node.js** for utility scripting, testing, and backend stubbing.

Designed & Developed various HTTP API clients using **jQuery AJAX**, **Superagent**, **Promises**, higher-order functions, and "README"-driven development. API clients for retrieving and playing HLS streams as well as clients for retrieving emergency (**EAS**) notifications and application updates. Originally utilized **WebSockets** which were switched to **Server Sent Events**, **Mocha**, **Grunt**, **Gulp**, and **GNUMake**.