☎ 857-266-4119
✉ anorwell@gmail.com
Cambridge, MA
anorwell.com
github.com/ANorwell

Arron Norwell

Summary

A results-oriented engineer with proven experience designing and building robust, scalable, and concurrent backend software. Expertise with distributed computing, SQL and no-SQL database application development, and performance optimization. Experience leading teams and managing important projects that were delivered successfully and on-time.

Skills

- HTTP REST APIs (XML, JSON)
- Relational DBs (Postgres, MySQL)
- Distributed Systems (ZeroMQ, Zookeeper)
- Version Control Systems (git, SVN)
- Scalability and Performance Optimization
- Agile (Scrum, Kanban)
- No-SQL (Redis, Cassandra, MongoDB)
- Cloud Applications (AWS, S3, OpenStack)
- Build + Deploy Tools (Make, Rake, Chef)
- HTML5 + CSS; Ruby on Rails

Languages Java, Scala, C, C++, Ruby, Javascript, Perl, Python, elisp, Go, R

Experience

Nov 2013 - Present **Senior Software Engineer**, *Datto*, Cambridge, MA.

- Designed and implemented stability, scalability, and feature improvements for a large distributed systems architecture, able to support backups for over a million users daily and built in Ruby on Rails using Redis, PostgresQL, Cassandra, and Amazon S3.
- Acted as project technical lead for a team of seven developers on a project to deliever a full Backup and Restore Solution for Office 365. Project was completed successfully and on-time. Presented the project to hundreds of partners at the DattoCon 2015 conference.
- Designed and was main contributor to open-source projects for an internally-used Apache Cassandra client and schema management tool.
- Designed and implemented an extendable API client and authentication framework that handled tens of millions of requests per day against many different APIs.
- Led a team involved in building a permissions management system as part of an effort to convert to a service-oriented architecture.
- Provided direction on the engineer hiring process by designing the technical practical completed by candidates, improving the company interview process, and interviewing candidates.

Apr 2012 - Nov 2013 **Software Developer / Team Lead**, *mDialog*, Toronto, ON.

- Designed and implemented components of a distributed, asynchronous, and highly scalable video streaming platform built in Scala, Java, and C, using Redis, PostgreSQL, Apache Zookeeper, and ZeroMQ.
- As project lead, worked with another developer to deliver a multiple-component project successfully and on-time to a large external client; acted as external-facing technical point of contact for that product.
- Supervised teams of developers as part of an Agile development process, responsible for design and implementation of new applications and features.
- Horizontally and vertically scaled a software platform from zero users to over 250,000 concurrent video streams running on clusters of over fifty machines; designed software components proven to scale vertially to thousands of requests per second.
- Analyzed, profiled, and created tools for visualizing the performance of Java Virtual Machine (JVM) applications.
- Open-sourced several internally-developed Scala libraries, including an asynchronous Redis client, an asynchronous HTTP client and server, and a ZeroMQ library.
- Interviewed potential new hires and participated in code reviews.
- mDialog was acquired by Google in April 2014.

May 2009 - Aug 2010 **Software Development Engineer in Test**, *Microsoft*, Mountain View, CA.
- Engineered tests for VOIP platform infrastructure used by large corporate clients to handle thousands of calls per day
- Led project to Design and implement VOIP test framework for quick and flexible creation and execution of tests, with web UI for test reporting
- Created automated graphing tools for visualization of performance data.
- Languages: Perl, C++, Javascript, Python, R

May 2008 - Aug 2008 **Software Development Engineer in Test**, *Tellme Networks*, Mountain View, CA.
- Designed and implemented test automation framework for back-end VOIP system.
- Developed various automated test utilities.

Education

Sept 2010 - Feb 2012 **M.Sc. - Computer Science**, *University of Toronto*, Toronto, Ontario, Canada.
- Thesis: *A Threshold for Clusters in Real-World Random Networks*
- GPA: 4.0
- Seminar Talks Given:
  Balanced Graph Partition Problems, Mar 2011
  The Diameter of a Scale-Free Graph, Nov 2011
  A Threshold for Clusters in Real-World Random Networks, Dec 2011

Sept 2006 - May 2009 **B.Sc. - Combined Honours, Mathematics and Computer Science**, *University of British Columbia*, Vancouver, British Columbia, Canada.
- GPA: 92
- Courses: Computer Graphics, AI, Programming Languages, Software Engineering, Advanced Algorithms, Numerical Methods for PDEs, Compilers and Interpreters

Selected Projects/Open Source

fileUploader.js (2014 - Personal Project), *http://github.com/ANorwell/nginx-resumable-upload*.
A HTML5 javascript library resumable file uploads, designed for use with nginx's resumable upload module.

Stieltjes (2013 - Personal Project), *http://github.com/ANorwell/stieltjes*.
A minimal, asynchronous UDP-based Scala cleint for Riemann, the distributed application monitoring system.

anorwell.com (2010-13 - Personal Project), *http://anorwell.com*.
My personal website is a minimally-featured blog tool written from scratch. Implemented feature-equivalently in two versions: Ajax with a Python/MySQL backend, and PHP. Supports upload, storage, and display of posts and music, as well as comments using Facebook's comment system.

Brando (2013 - Contributor), *http://github.com/chrisdinn/brando*.
An Akka-based Redis client in Scala; supports sharding and pipelining.

InterviewStreet (2012 - Personal Project), *http://github.com/ANorwell/InterviewStreet*.
Implementations of solutions to various InterviewStreet coding problems. Several interesting graphing and network algorithms; implemented in Ruby and C.

Graph.js (2011-12 - Personal Project), *http://anorwell.com/graph*.
A Javascript Graph creation and visualization API using HTML 5 Canvas. Allows for flexible appearance and manipulation of graphs. The example app created using this API allows users to draw, save (either using HTML5 LocalStorage or to the cloud),and share their graphs.

Graphics (2009 - Personal Project), *https://github.com/ANorwell/graphics*.
An implementation of some graphics algorithms and design patterns in C++ and using OpenGL/GLSL. Includes a surface subdivision algorithm that acting on an implementation of half-edge data structure, a scene manager, shader manager, quaternion camera, hit detection, and small shader library including a Phong shader. Compiles to a program with a game-like interface in which you can fly around graphics scenes.