Daniel Speyer

340 W 87st apt 2B
New York, NY 10024
dspeyer@gmail.com

History

Symbiont -- *CI/CD Lead* -- 2021-2023

Designed, built and maintained both unit and full integration tests for our distributed byzantine system, including terraform/scripting for running the tests and mergify for integrating them into our source management process. Also managed our installation automation (both RPM+ansible and kerberos) and our command line interface.

Entrupy -- *Software Engineer* -- 2018 - 2019

Worked on a variety of computer vision and data-pipelining projects relating to counterfit recognition and physical object fingerprinting. Did work on image recognition (signal augmentation followed by genetic algorithm), texture recognition (mostly performance improvements, some over 100x), and an automated testing platform (mostly AWS and Docker).

New York Genome Center -- *Programmer and Generalist* -- 2016 - 2017

Worked on the DNA.Land project, which gathered genomic data directly from individuals who had been sequenced by 23&Me and similar companies, then analyzed it. I worked on all aspects of the project, including performance (of existing tools), reliability, visualiztion, general user interface and statistical toolbuilding.

Turing Solutions -- *Temporary Central Technical Officer* -- summer 2014

This was an early-phase startup applying machine learning to a wide variety of problems. I made preliminary plans regarding both platform and target market, then began work on a project to classify music into genres. The project was not completed that summer (nor could it have been) but I developed a viable design and some preliminary code that was able to categorize widely different styles.

Columbia University -- *Student* -- 2013 - 2018

See Education section

Google -- *Site Reliability Engineer* -- 2007 - 2012

Maintained and created tools to maintain a variety of Google services, including blogsearch, igoogle, colossus (internal distributed filesystem) and bigtable (internal nonrelational distributed database). When needed, modified the services themselves to be more maintainable. Focused heavily on high-availability and high-performance extremely distributed systems. Performed oncall duties for the services, and ran simulated outages to train others in oncall handling. Wrote documentation, both for my own tools and pre-existing ones. Interviewed candidates.

Tools were mainly related to monitoring, testing and deployment. Scope ranged from quick one-offs to multi-Kloc projects with formal design docs and design reviews. Most were written in Javascript and internal languages, but some involved python and C++.

Microsoft -- *Software Development Engineer* -- 2005 - 2007

Investigated and addressed battery life of multimedia scenarios, including DVD playback, audio playback, and video playback on a variety of hardware platforms and configurations. On most of these cases, I achieved at least a 10% improvement. I also worked on tablet input services and GDI graphics from a responsiveness perspective. All these problems involved a lot of profiling, analysis (both writing scripts to interpret long traces and using statistics to understand the results), test automation, and reading existing code.

Then worked on profiling graphics code. Built tools to investigate D3D performance, including integration with general profiling tools and entirely new ways of visualizing timing data. Worked closely with people who were investigating performance of specific graphics performance issues.

Microsoft -- *Software Development Engineer (Intern)* -- summer 2005

Designed, coded and documented a system to test laptop battery life by running down to battery failure and then automatically recharging. Integrated the system into the existing test infrastructure and documented it. The primary technologies involved were C++, SNMP and HTTP. I then expanded the system for the simpler power control needs of the rest of the client performance lab (rebooting of Blue-Screened systems).

Information Systems and Internet Security Laboratory (Polytechnic University) -- *Research Intern* -- summer 2004

Helped to design and built ForNet, a distributed logging system for large IP networks. Designed a protocol on top of XML and TCP to describe noteworthy network events in a general way, then wrote a library to implement the protocol. Also wrote a system by which different data-collection modules could collaborate to answer a user's query automatically. All this work was essentially on my own initiative, with supervision mainly to ensure compatibility within the team.

Cosmic Ray Energetics And Mass Project (Univ. of Maryland, with CERN, NASA and others) -- *System Administrator* -- 2002 - 2005

Maintained a heterogeneous network of GNU/Linux and Windows. Provided SSH, SFTP, NFS, lp, SMB, http, ftp, and e-mail services. Also handled disk monitoring (including RAID), hardware upgrades, introduction of new systems (including purchasing), security, documentation, and handling miscellaneous user complaints.

CyberCamps -- *Counselor* -- summer 2002

Taught programming (C++) and robotics (from a kit, programmed in Stamp Basic); organized activities (mainly sports and crafts) and looked after campers (including recreation, meal and dorm times). I also performed what little maintenance our local computer network required. My students were between 12 and 16, but I also looked after younger students.

Cyberfun Computer Camps / Jobs & More -- *Technical Consultant* -- summers 2001 and 2000,

Set up and maintained a heterogeneous network first of Windows and GNU/Linux systems, both software and hardware. Ran NFS, FTP, SMB, HTTP, VNC and IP-Masquerading services..

Education

Columbia University -- *Courses*

While not part of a formal degree program, I have completed over 60 credits of graduate-level classes in computer science (mostly systems and machine learning), bio-informatics and neurology. I hold a cumulative GPA of 3.7.

University of Maryland at College Park -- *Bachelor of Science*

I hold a Bachelor of Science degree with a computer science major and math minor from the University of Maryland. In addition to those subjects, I took high-level courses in biology, physics and anthropology. I was a member of the computer science honors program, a presidential scholar, and on the dean's list six out of the eight semesters. I graduated with both departmental and college honors.

Publications

Rapid re-identification of human samples using portable DNA sequencing. With Zaaijer, S., Gordon, A., Piccone, R., Groen, S. C., & Erlich, Y. (2017) *eLife, 6, e27798.* http://doi.org/10.7554/eLife.27798. I designed, coded and tested the initial algorithm for this, then the co-authors added relative support and tested more thoroughly.

DNA.Land is a framework to collect genomes and phenomes in the era of abundant genetic information. With Yuan, J., Gordon, A., Aufrichtig, R., Zielinski, D., Pickrell, J. & Erlich, Y. (2018) *Nature Genetics 50, 160-165* https://doi.org/10.1038/s41588-017-0021-8 (preprint: https://www.biorxiv.org/content/early/2017/05/09/135715). I helped build the DNA.Land project itself (from optimizing C++ to designing visualizations in D3); the co-authors evaluated its success.

Github

My Github is https://github.com/dspeyer and contains a mixture of research projects, personal projects, and toys.

Keywords

GNU/Linux, javascript, jquery, react, d3, python, django, flask, numpy, tensorflow, keras, opencv, c, c++, cuda, lisp, bash, perl, html, css, sql, sqlalchemy, postgres, nosql, docker, tcp/ip, circleci, mergify, git, terraform, gcp, aws, ansible, data science, machine learning, bayesian statistics, mcmc, stan, convolutional neural networks, genetics, vcftools, blast.

These are all technologies I have done nontrivial work in. Some may be too trivial to be worth listing, but I tried to err on the side of including things that some overworked resume-filterer might include in a grep statement. There are plenty more that I've done a little with, and unlimitedly more that I'm willing to learn.