Daniel Speyer

340 W 87st apt 2B
New York, NY 10024
(646) 233 2135
dspeyer@gmail.com

History

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 - present

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 (mainly RedHat 8.0 and 9.0, and Fedora Core 2) and Windows (98, 2000 and XP). The network also included Solaris and Digital Unix workstations under third-party control. Most GNU/Linux systems ran SSH, SFTP, and NFS servers, and many performed lp and SMB printer-sharing. One GNU/Linux server also handled http (public and private), anonymous ftp, and e-mail functions. Others performed IP-masquerading. My duties included disk monitoring (including RAID), hardware upgrades and introduction of new systems (including purchasing), security, inventory control (coding a web-based system for this), 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 95, Windows 98, and GNU/Linux (Redhat 6.2), then of Windows 98, Windows 2000, and GNU/Linux (Mandrake 8.0 and Suse 7.0). Also assembled and modified hardware. Ran NFS, FTP, SMB, HTTP and VNC servers (all intranet), and IP-Masquerading to the internet through a DSL connection. Installed and maintained productivity and gaming software, and trained staff in its use. Performed regular backups, wrote documentation, and responded to specific failures.

Education

Columbia University -- *Courses*

While not part of a formal degree program, I have completed 34 credits of graduate-level classes in computer science (mostly systems and machine learning), bio-informatics and neurology. I have received no grade lower than a B.

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.

Selected Technical Skills

Deep knowledge of C, C++ (including templates), Javascript and Linux/Unix.
Fluency in C, C++ (including template metaprogramming and C++11), Javascript, Python, and HTML.
Proficient with Bash, Matlab, Lisp, Perl, CSS, SQL and Windows.

Github

Samples of my work can be found in my github account at https://github.com/dspeyer

The ubiq_genome, causality, spatialtable, profiling and vpr-gpu projects are all coursework, and are deliberately less maintainable than code I expect to look at again in more than three monthes. Ideologyturing was a request from a friend; the pithos fork is for my own use and the rest are projects of personal interest.