



CURRENTLY SEEKING

I am currently employed, but am interested in opportunities as a software engineer in the metropolitan Boston area.

EDUCATION

BSE in Computer Science Engineering from Princeton University, 1998.

PROFICIENCY

Languages: Java, C, Python, Javascript, ActionScript, MXML, Bourne, Perl, C#, C++, assembler. Operating Systems: Unix (including Linux), Windows, Android. Technologies: multithreading, object orientation, sockets, porting and platform-neutral programming; text processing (indexing, auto-categorization, tokenization, word clustering, character normalization, regular expressions, parsing, YACC, Unicode, encodings, internationalization, localization); Web programming (HTML, CSS, HTTP, CGI, AJAX, web services, JSON, Flex, JSP, Servlets); SQL, ODBC, MySQL, SQLite; GUI programming (Swing, AWT, SWT, Wx, GTK+); XML (XSLT, XPath, DOM, SAX, SOAP); embedded programming (Android, Blackberry, J2ME); multimedia programming (MIDI, audio synthesis, image recognition); .NET and JNI; distributed systems (load balancing, failover, mirroring, data segmenting); Android Studio, Gradle, ADB, NDK, Git, Jenkins. Project management: Agile, Scrum, Kanban. Rapid and eager learner of new tools and technologies.

EXPERIENCE

SENIOR SOFTWARE ENGINEER, AMAZON.COM - 2013 to present

Work on Audible products and services.

Contributing architect and developer of *Amazon English*, Android-based educational software that helps native Japanese speakers improve their reading and listening comprehension of English.

Contributing developer of audiobook playback and *Immersion Reading* (synchronized audiobook and e-book) software for Android, Kindle Fire tablets, and the Amazon Fire Phone, including Java-based user interface and networking code, and C++-based audio file and digital rights management code.

Helped build several teams at Amazon by running numerous technical interviews.

SENIOR SOFTWARE ENGINEER AND SOFTWARE DEVELOPMENT MANAGER, SAS INSTITUTE - 2003 to 2013

Work on Teragram, PicoSearch, and SAS text analytics products and services.

Development manager for information retrieval technologies including search, crawling, and document normalization. Led a cross-site team of multiple developers and quality assurance engineers through several product releases.

System architect and principle developer of *Information Retrieval Studio*, a ready to use and extensible search product including crawling, document manipulation, and indexing. Features include a Flex-based administration interface, web services remote control, Python-based plugin infrastructures for document processors and social media site crawlers, and a highly customizable Java servlet-based query interface.

Architect and developer of *Markup Matcher*, a document normalization tool which provides an interactive interface for building and testing XPath expressions (with extended support for regular expressions, date normalization, etc) against sample XML or HTML documents, then publishing the resulting model so that it can be applied to new documents in realtime. Also, implementer of a client/server system for extracting text from documents in other formats including PDF and Microsoft Office, built atop third party tools (Microsoft IFilter, Apache Tika).

Contributing architect, integrator, and crawler team lead for the SAS social media analytics service. Managing liaison for integration of search technology into other SAS products.

Principal developer for integrating Teragram's existing library code into the codebase of the new corporate parent, SAS. Contributing system architect for SAS' next generation unified text analytics software suite. Also, principal integrator of Teragram search, crawling, and categorization software for internal documentation at SAS.

Leader of an international development team for the *MyGADs* knowledge repository portal service. Features include a natural language interface for answering questions, teaching new facts, and issuing directives; an account and permissions system for collaboration and social networking; a full wiki system with version histories, natural language analysis, and full-text indexing of document formats including HTML, PDF, and Word; a hosted programming language for extension applications; a Web 2.0 interface with natural language chat, WYSIWYG wiki editing, and embedded GUIs for extension applications; text-based interfaces to the natural language chat from SMS, Jabber, AOL, Yahoo, and MSN instant messaging, IRC, and programmatic web services; Blackberry, J2ME MIDlet, and WAP mobile interfaces to the whole system including wiki editing and embedded GUIs for extension applications; a distributed architecture supporting load balancing, mirroring, and data segmentation; backup and restore mechanisms; web-based administration; and query statistics analysis. Responsibilities included coordination of developers,

testers, documenters, and IT administrators; system, component, protocol, and database architecting; component implementation; and sales integration.

System architect and principal developer of the *PicoSearch Platinum* hosted search service. Designed and implemented a modular web and FTP crawler, supporting a different set of document and link preprocessors for each customer. Designed and implemented a web-based query interface utilizing XSLT. Integrated Teragram search engine, spellchecking, thesaurus, stemming, and document format conversion software.

Designer and developer of a corpus-based query suggestions system for use in both hosted and OEM search products.

Designer and principal developer of a term clustering and ranking product, based on the frequency of term cooccurrences in a training corpus. Supporting work included design and implementation of a cooccurrence model API and task-specific data caching and paging routines.

Contributing developer of *Direct Answers*, a natural language question answering service based on data mining of public repositories of general knowledge (Wikipedia, CIA World Fact Book, IMDB, weather services, etc). Integrated WordNet as a knowledge source. Implemented a full-featured unit converting calculator within the natural language interface. Implemented a framework for merging results from third party search engines into the set of natural language question answers. Performed integration work for high profile initial customer AOL.

Development work on core search engine software. Designed and implemented support for three powerful query syntaxes, distributed virtual indexes, phrase recognition, character normalization, and multiple mechanisms for index updates. Implemented support for wildcards in queries. Designed and implemented an integrated web interface for real-time statistics monitoring, as well as an offline log analyzer. Implemented query APIs for several computer languages. Technical editor for the search engine documentation.

Development work on core categorization software, including design and implementation of a rule validator, client APIs, and an XML document annotation tool. Technical editor for the categorization engine documentation.

Pre- and post-sales integration of Teragram and third party search engine, categorization, concept extraction, email alert, spellchecking, phrase recognition, stemming, and document format conversion software for high-profile customers including ABC News, Archivas, the New York Times, the Washington Post, CNN, the Associated Press, EBSCO, and the US Department of Education.

Porting of software from Linux to Windows, including creating a Windows build system for Teragram's codebase. Also designed and implemented a wrapper for installing Unix-style daemons as fully-integrated Windows services.

SENIOR SOFTWARE ENGINEER, ALTAVISTA - 1998 to 2003

Work on AltaVista Enterprise Search.

Architect and project lead for a complete redesign of the search query infrastructure. Migrated from a proprietary web server, template engine, and plugin system to a multilayered system of C libraries, wrapped in both JNI and COM, invoked from Servlets, JSP, and ASP scripts, which in turn plugged into multiple third party application engines and web servers. Responsibilities included project design and implementation, as well as management of two other developers, a technical writer, and a quality assurance engineer.

Engine maintainer, plugin author, template master, and documentation writer for a proprietary dynamic web page generation system, responsible for the look and feel of the end users' interface. Implementer of an alternative web services based query interface utilizing SOAP and XSLT.

Architect and implementer of a library for parsing and robustly manipulating three different query syntaxes. Maintainer of a library for converting amongst different character encodings and performing international character normalizations. Resident Unicode expert and localization data maintainer. Designer and implementer of an HTTP proxy which allowed the search product to leverage security benefits of Apache and IIS.

Contributor to core crawling and indexing algorithms; customer-facing technical liaison in many sales and support situations; instructor of training courses for customers as well as internal sales and support groups; and mentor of a junior developer.

SOFTWARE ENGINEER, PRINCETON UNIVERSITY COGNITIVE SCIENCE LABORATORY - 1997 to 1998

Work on the *WordNet* hyper-thesaurus; design and implementation of multi-platform GUI and Web interfaces; programming of backend library routines; porting to X11, Windows, and Mac.

UNDERGRADUATE RESEARCH, PRINCETON UNIVERSITY - 1997 to 1998

Design and implementation of computer recognition systems for musical hand signals, using both video and sensor glove techniques including fuzzy logic. Also, analysis and design of human-computer interfaces for music notation.

INTERN, BELL COMMUNICATIONS RESEARCH - 1995 to 1996

Programming of prototype telephone network control and database software; preparation of traveling demonstrations; and system administration.

OUTSIDE INTERESTS

Music composition and performance; homemade music software; community theater; and volunteering at an animal shelter.