

Bruce Atherton

xxxx Foul Bay Road
Victoria, BC V8R 5E2

Home Phone: (250) 592-xxxx

Work Phone: (250) 385-xxxx x240

Email: bruce@callenish.com

Professional Objective

I am a skilled systems architect and project leader with 20 years' experience creating software and 9 years' experience planning, leading, co-ordinating, and directing a team. I have a broad base of technical knowledge and excellent people skills, and am always looking to learn new things and meet new challenges.

Management Skills

Administrative: Developing project plans, Writing funding proposals, Budgeting, Managing funds, Delivering on commitments, Meeting deadlines.

Leadership: Supervising and focusing teams, Creating culture change, Promoting empowerment, Team building, Creating a shared team vision, Strategic planning.

Computer Skills

Programming Languages: Java, Python, C, C++, Perl, bash, awk, sed, Javascript, SQL.

Java/J2EE: Servlets, EJB, JSP, Custom Taglibs, JSTL, JDBC, JNDI, RMI, JAXP, Model 2, Struts, Blueprints, Applets, Swing.

XML: SAX, DOM, XPath, XSL:FO, XForms, SVG, XSLT, XML-RPC, SOAP.

Server Administration: Apache, Weblogic, Oracle AppServer, Tomcat, Zope, SMB/Samba, POP3/IMAP Mail Servers, NTP, Mixed Linux/Solaris/Windows/Novell Network, Cross-system log file integration, Ongoing security audit, System Monitoring.

Methodologies: Continuous Integration, Process Control, n-Tier Distributed Computing, Nonboolean text search, Artificial Intelligence, Model-View-Controller, Extreme Programming, Design Patterns, Aspect-Oriented Programming, Multi-threaded Programming, Object Oriented Design, Web Services, Open Source.

Process and Change Management: Source Code Management, Issue Tracking, Automated Continuous Builds, Attribute-generated Artifacts, Quality Metrics, Automated Deployment, Automated Testing, Performance Monitoring, Logfile Generation and Analysis.

Analysis: Requirements Gathering, Writing Specifications, Modelling (UML), Process Reengineering, Test Plans, Presentations.

Research Skills: Developing and Testing New Technologies, Identifying Opportunities Provided by New Technologies, Finding Options in Existing Technologies, Discovering Causes and Solutions for Obscure Bugs.

Artificial Intelligence: Neural Networks, Expert Systems, Natural Language Processing, Fuzzy Logic.

Documentation: Publications, Code Documentation, Training Manuals, Wiki Maintenance, Web site Maintenance.

Employment History**Senior Developer**

Avue Technologies

August 2001–Present

Process Designer, System Architect, J2EE Developer, Test Engineer, Network Administrator, and Change Agent. Member of a team that provides a human resources web application for various U.S. government agencies.

Projects:

- Created a continuous integration and release process that allows a sustainable schedule of new stable releases every week and bug fix releases every day or two.
- Created a change management system that allows exact control of code changes and associated issue records that are promoted to each environment and to each release.
- Ported a 140,000-line application from Weblogic to Oracle App Server with a 2-week deadline.
- Set up a system of secure transfers between systems and networks to allow critical metrics such as CPU usage, cluster behaviour, and free memory to be monitored from a central location.
- Wrote a Python script to emulate a web browser in order to create an archive of all of the dynamically created HTML

and PDF content of Avue.

- Wrote a system which pulls all of the Job announcements off of a government web server, extracts data about each job that is then stored in a database, and emails former applicants whose profile matches the job requirements.
- Created a system whereby applicants can publish their own resume site.

Achievements:

- Reshaped the source code management, release strategy, performance monitoring, bottleneck identification, and deployment abilities of the Avue Technologies web application.
- Introduced many architecture improvements to Avue's J2EE system.
- Automated systems so that server environments operate with a minimum of user intervention.

Systems Architect

University of British Columbia Student Services

January 1998–September 2001

Member of a team that created a three-tier J2EE system for faculty, staff, and students at the University of British Columbia. The client side is HTML/JavaScript, Java Applets, and a Java Application. The web tier is Apache and Weblogic hosting Servlets and JSPs. The app tier is Weblogic Application Server serving EJBs. The database is Oracle.

Projects:

- Created a front-end client applet.
- Designed and wrote an interprocess communication system which used custom marshalling services to provide three tier application services. Eventually this was replaced by an Application Server.
- Created a continuous integration and release management system
- Created a regression testing and load testing environment using OpenSTA
- Using Weblogic's ZAC and InstallAnywhere, created an automated application distribution system.
- Using Python, created a performance monitoring tool with live graphing on the web.

Achievements:

- Reshaped the source code management, release strategy, bug tracking, performance monitoring, and update delivery of the UBC Student Services web application.
- Introduced Debian Linux servers and Solaris servers into the development environment and integrated them into a mixed Novell, NT, and HP-UX environment.

Team Leader

UBC Faculty of Law Artificial Intelligence Research (FLAIR) Project

1990–1997

Member of a research and development team discovering and applying new computer technology, particularly artificial intelligence techniques, to legal research, ultimately developing and shipping a Win32 commercial product.

Projects:

- Wrote a high speed non-boolean text retrieval system using the Vector Space Model.
- Helped write a system to automatically categorize and disambiguate terms in legal text. Terms are divided into legal terms, cited cases, cited statutes, and facts. Each category has unique methods for finding synonyms.
- Created a system using these newly researched technologies to automatically mark up a legal document with SGML in each of the four categories: case, statute, legal term, and fact. The SGML contained a canonical form for each phrase and, in the case of cited cases, provided a hyperlink back to the original judgement.
- Worked on a system to identify similar statutes between jurisdictions and across time.
- Worked on a system to automatically identify how a judge was citing a case, whether they were overruling it, distinguishing their case from the cited one, or any of several other types of citation.
- Designed and implemented a custom database system composed of a combination of inverted indices, hash tables, and specially sorted tables laid out in a particular way on a CDROM such that it could handle a query on tens of millions of records using fuzzy text matching and return results in a few seconds.
- Designed and implemented a build system to create custom Very Large Databases composed of tens of millions of records optimized for retrieval time. The system was an early example of Grid computing. Databases took over a week to build, with various machines joining and dropping out of the build process during that time.
- Developed a source code management and continuous build system for a multiplatform C and C++ codebase.
- Member of the commercialization team that translated the research product into one that could be marketed.

Achievements:

- When the project was about to be cancelled due to financial exigencies, took on the task of analysing financial position and forecasts, thus ensuring the project continued.
- Identified how the culture of the team was optimized for research, and moved the team to a culture which was optimized for delivering a product.
- Produced a novel method of retrieving data from a CDROM, for which a patent application was made.

- Identified new products which these technologies could realize, and refocused the team to producing them.
- Produced several articles on this work.

Systems Analyst

Target Software

1989–1990

Launched a new consulting arm for this value added reseller.

Projects:

- Developed Code Generation Templates for SBT.
- Wrote Informix SQL programs.
- Modified Fourgen Accounting systems.
- Installed and supported UNIX systems.

Owner

Stainless Steel Software

1984–1989

As an independent computer consultant, I provided computer-related services to my clients. Clients included several courier companies, a computer VAR, a number of engineering firms, a software developer, and one of the world's largest mining companies.

Projects:

- A communications subsystem for a point of sale system for retail chains.
- An object-oriented database with a multidimensional spreadsheet.
- An automated document transfer system.
- A vision recognition system.
- A low-level disk manager for the Amiga.

Open Source Projects

Founding Member

Apache Ant Project Management Committee

Project Lead

RRFiller Sourceforge Project

Contributor

Subversion Version Control System

Contributor

W3C XForms 1.0 Candidate Recommendation

Publications

- Bruce Atherton. "Build a Home Theatre PC for your Living Room Parts 1 through 4". *Toronto Computes*. Canada Computer Paper. September - December 2002.
- Bruce Atherton. "Artificial Intelligence Technologies Explained: Expert Systems and Case Based Reasoning". *CSALT Review*. Canadian Society for the Advancement of Legal Technology. July 1996.
- Bruce Atherton. "Artificial Intelligence and the Law: The Faculty of Law Artificial Intelligence Research Project". *CSALT Review*. Canadian Society for the Advancement of Legal Technology. April 1996.
- Bruce Atherton. "Artificial Intelligence and Legal Discourse: The Flexlaw Legal Text Management System". *Artificial Intelligence and the Law Journal*. Kluwer Academic Publishers. 1995. 55-95.

Education

Bachelor of Technology. Completed Significant Coursework
British Columbia Institute of Technology

Graduate Level Computer Science. Audited several courses
University of British Columbia

Miscellany

Excellent references available on request