

Brian Alliet

1158 S CLINTON AVE, ROCHESTER, NY 14620
585 880-5276

brian@alliet.com

Objective

To obtain a position where I can use my skills and experience to solve interesting problems and develop software using those solutions.

Skills

- Languages: Bourne shell, C, C++, C#, C--, Haskell, Java, JavaScript, ML, Objective-C, Pascal, Perl, PHP, Scheme, SQL, VBA/VBScript
- Operating Systems: *BSD, Linux, OS X, Windows
- Databases: Microsoft SQL Server, MySQL, SQLite
- Technologies: ASP.NET, Classic ASP, COM

Experience

ZAK SOFTWARE **2000-PRESENT** **SOFTWARE ENGINEER / SYSTEM ADMINISTRATOR**

Duties include web application development (using a variety of server side languages, including Classic ASP, ASP.NET, PHP, and Perl), desktop application development (using Delphi), system administration (internal web, mail, DNS, and database servers, as well as file/print servers at customers' sites) of Windows, Linux, and FreeBSD servers, and customer support (everything from simple password resets to helping clients develop and debug their code). Built hosting infrastructure up from a single server to over a dozen servers spread across the country. Worked with customers at every stage of the development process from assessing their needs to delivering the finished product.

Selected Projects

PEMDATA **www.pemdata.org** **2008 - 2010**

A website developed in partnership with the Image Permanence Institute at the Rochester Institute of Technology to store and analyze temperature and humidity data (collected from temperature/humidity monitors) for libraries, archives, and museums so they can better care for their collections. PEMdata assesses risks based on the environmental data collected and the types of materials stored at each location. It can answer questions such as "Is it too damp? Will paper curl?" and "Is it too dry? Will paper become brittle?", "Will metallic objects corrode?". PEMdata is written in Classic ASP and was optimized for performance using custom COM controls (written in C) where appropriate. It can analyze (on the fly) years of environmental data in seconds.

LAMBDAVM **wiki.brianweb.net/LambdaVM** **2004 - 2007**

A JVM backend for GHC (The Glasgow Haskell Compiler). LambdaVM allows Haskell applications to run, unmodified, on the JVM. A new backend was added to the compiler to emit JVM bytecode from STG code (one of GHC's intermediate languages) and the standard libraries were modified to use the Java class library for all primitive operations (such as reading and writing files) which depended on native code.

ORG.IBEX.CLASSGEN **darcs.brianweb.net/classgen** **2004 - 2005**

A JVM bytecode generation and analysis library written in Java. It can fully parse, transform, and emit Java class files with an easy to use API. Used to output JVM bytecode in NestedVM (see below)

Brian Alliet

ORG.IBEX.CRYPTO
crypto.ibex.org
2004

A small, fast, and simple pure Java implementation of SSL/TLS and associated protocols and algorithms (AES, DER, HMAC, RC4, RSA, X509, etc) with an API that is a drop in replacement for java.net.Socket. Used to support RPC calls over SSL in Ibex project (see below) where native SSL support isn't always available.

NESTEDVM
nestedvm.ibex.org
2003-2004

A binary translator which converts MIPS machine code to JVM bytecode. NestedVM was developed to allow libraries written in unsafe languages (such as C) to be used safely and securely in the JVM sandbox. A cross compiler is used to compile any GCC supported language (C, C++, and FORTRAN to name a few) down to MIPS machine code, which is then translated to pure JVM bytecode, with no dependencies on JNI or any native code. A runtime library (implemented partly in C, and partly in Java) was developed to support all ANSI C code (plus many POSIX extensions, such Berkeley sockets) and well as well as conveniently invoke functions and access data from Java. Used for a variety of purposes (JPEG decoding with libjpeg, font rendering using FreeType, CAB file decompression using libmspack, etc) in the Ibex project. Presented at the ACM SIGPLAN 2004 Workshop on Interpreters, Virtual Machines and Emulators.

BOEHM GC
www.hpl.hp.com/personal/Hans_Boehm/gc/
2003

A conservative garbage collection for C and C++. Contributions include the initial implementation of thread and dynamic library support on Darwin/OS X. Mach (the microkernel Darwin is based on) and dyld calls are used directly to work around limitations (at the time) of Darwin's pthread and dlopen support. This port also allowed GCJ (the GNU compiler for Java) to gain Darwin/OS X support.

IBEX/XWT
www.ibex.org
2001 - 2005

Ibex (formally known as XWT) is an application development platform based on XML and JavaScript. User interfaces are laid out in XML, with the application logic written in JavaScript (and usually accompanied by server side code accessed via XMLRPC or SOAP for the heavy lifting). Ibex runs anywhere there is a JVM, and has native Windows, Linux, and OS X ports. A Java applet is used to launch Ibex applications directly from the browser. Contributions include a custom JavaScript engine, org.ibex.js (the performance and code size of third party JS engines were unacceptable), an entire widget library (buttons, radio boxes, list boxes, etc) which can emulate the look and feel of many native user interfaces, a native OS X port with OpenGL hardware acceleration, as well as routine development, bug fixes, and user support.

Publications

Complete translation of unsafe native code to safe bytecode
Proceedings of the 2004 workshop on Interpreters, virtual machines and emulators.
www.brianweb.net/papers/nestedvm.pdf

Education

Rochester Institute of Technology
Master of Computer Science
2012 (Expected)

Rochester Institute of Technology
Bachelor of Information Technology
2004