jfront inc.

Andrew Eidsness

andrewe@jfront.com   (613) 286-6398
http://www.jfront.com/consultants/andrewe.pdf

I have extensive experience developing embedded software in several industries including telecom and consumer devices. I have also created a number of software development tools to aid in the creation of embedded software.

Work Summary   [details on page 3]

June 2007 – present   **jfront incorporated**   Software Developer
I have undertaken contracts with QNX Software Systems, JDSU, DRS Technologies, Zeligsoft Inc., Espial Group Inc., Protecode Inc., EMS Satcom, Edgewater Computer Systems, and Rational Software Corporation.

Oct   2004 – May 2007   **JDSU**   Senior Embedded Software Developer
JDSU is the largest producer of optical components for the telecom industry. I was a lead software developer for telecom grade circuit packs. These packs (a collection of one or more optical components on a single PCB) are custom developed for several large telecom clients.

Jan   2004 – Oct 2004   **IBM**   Java Software Developer
At IBM I was a member of software team that developed the Eclipse project (www.eclipse.org) and made several key contributions to the core UI for eclipse release 3.0.

July 2001 – Dec 2003   **Innovance Networks Inc**.   Embedded Software Developer
My responsibilities were centred on the design, implementation, and support of the core OAM framework that was used by the embedded applications to manipulate network elements. In particular, I was responsible for the application interface, the object model server, several supporting applications, and an inter-processor reliable event subsystem.

Mar 2000 – July 2001   **jfront incorporated**   Consulting Services
The initial goal of jfront was to productize a Smalltalk to Java compiler technology that I developed as my honours project. Following that attempt I used the company as an entity for contracts as follow.

Dec 2000 – Feb 2001   **Rational Software Corp.**   Senior Software Engineer [contract]
I was contracted prior to product release to implement fixes to customer reported problems with the code generator. In addition to fixing these problems, I performed a general inspection of the code in order to find and repair as yet unreported bugs.

Sept 2000 – Nov 2000   **Espial Group Inc.**   Java Software Developer [contract]
I was contracted to aid the development of a Java based server that stores Java applications in a central database. These applications are downloaded to a proprietary Java environment, which runs on a variety of embedded devices (e.g., cell phones, set-top boxes, PDAs, etc.).

Apr 1998 – Mar 2000   **ObjecTime Limited**   Software Developer
During my tenure I was involved with several projects; from maintenance and evolution of the Smalltalk based tool, to implementation of new features in the C++ based code generator. These projects allowed me to explore the implementation of Object Oriented features in a variety of languages.

Education

1993-1998   Bachelor of Computer Science, Carleton University (co-op option)

Consulting Projects

I have undertaken a variety of consulting projects which can be broadly categorized as embedded development using C/C++ and Eclipse-based tools development using Java.

Eclipse CDT
I've implemented features in client's private forks as well as the upstream repository at eclipse.org. I've generated a list of details for all public contributions at jfront.com/cdt. One large area of focus has been enhancing the CDT indexer to support indexing of Qt meta-data.

Event Framework
I designed and implemented a publish/subscribe library for use in a linux-based embedded system. Shared memory was used to hold the subscription table and event payloads. Unix domain sockets were used for signaling.

Fault/Alarm Framework
I designed and implemented a fault detection and reporting framework for use on an Analog Devices DSP. The framework applied hysteresis to the raw readings and hold-offs to the detection times.

Embedded State Machine Frameworks
I have implemented multiple state machine frameworks in C for use in embedded devices. The state machines typically support storage for the currently active state and allow a way to associate behaviour with transitions between the states.

Logging Framework
I designed and implemented a logging framework which wrapped syslogd. A unique feature of the framework was the ability to suppress messages in the client application, rather than only in syslogd. This reduced inter-application messaging which improved overall system performance.

Code Generators
I have implemented multiple production C/C++ code generators for Eclipse-based modeling tools. As part of developing these code generators I implemented a technique for integrating with the Eclipse CDT editor. I presented this technique at an Eclipse Demo Night, the slides are available at jfront.com/consultants/20100526-eclipse-demo-night.pdf.

Open Source Porting
I have ported a number of open source libraries for embedded use. In all cases these projects needed to be adapted to the platform's build environment, the types needed to be mapped, and existing API needed to be implemented. Some of the projects that I've ported in this manner include:

- OpenSSL
- MatrixSSL
- libPNG
- libJPEG
- zlib
- Expat (XML parser)
- FreeType
- LZMA

Skills

Languages	Java, C, C++, Make, Bourne shell script, and Smalltalk
RTOS	QNX, VxWorks
Environments/Tools	Eclipse, CDT, Jenkins, valgrind, cygwin and Unix (gnu tool-chain), Linux
Source Control	Mercurial, Subversion, Rational ClearCase, Continuus, CVS, and MS Source Safe
Architecture	Commercial experience with the design, evolution, and migration of complex, multi-threaded software architectures and implementation frameworks.
Miscellaneous	I have implemented and maintained multiple shipping code generators.

Work Experience

May 2007 –
present

jfront incorporated
Contracts with JDSU, DRS Technologies, Zeligsoft Inc., Espial Group Inc., Protecode Inc., EMS Satcom, Edgewater Computer Systems, and Rational Software Corporation

> **SKILLS USED**
> C/C++, Java, Eclipse, CDT, QNX, VxWorks, junit, gnu tool-chain, flex, bison, server development, multi-threaded development, API development, XML, xslt

Several of the contracts have involved application development using the Eclipse framework. These contracts, along with my previous experience on the Eclipse team, have provided me with an excellent working knowledge of using Eclipse to build industrial strength applications.

Several of my contracts have also involved the design, implementation, and maintenance of C/C++ code generators. In addition to generation of the code, I have deep experience in generating Makefile artifacts to allow the generated code to be built into binaries that are able to execute in a variety of RTOS environments.

I have prepared and presented on developing code to interact with the Eclipse CDT environment. The primary focus of these presentations was live-interaction with the AST that is used by the CDT editors.

Oct 2004 –
May 2007

JDSU
Senior Embedded Software Developer

> **SKILLS USED**
> C/C++, QNX, ctest, gnu tool-chain, flex, bison, SNMP, DHCP, server development, API development, ITU-T G.700, G.800, X.710, X.734, ifconfig, XML, xslt

I was a member of a team that developed, manufactured, and supported custom circuit packs for various large telecom clients.

A typical project started with a set of customer requirements in the form of their network management protocol (e.g., an SNMP MIB). I performed a vertical role which included working with the project management team to flesh out the requirements, working with the FPGA, hardware, and optical module teams to ensure the completed circuit pack will meet the high level requirements, and implementing the required software using industry best practices.

Key contributions included reducing the size of our standard load by half while dramatically improving performance. I was consistently utilized as the prime software developer for troubleshooting problems with fielded packs and delivered several software workarounds for underlying hardware component failures.

Jan 2004 – Oct 2004	**IBM** **Java Software Developer**

> SKILLS USED
>
> Java, junit, eclipse, User Interface development, API development, CVS, Open source software development

I was a member of the team that invented and is still the largest contributor to the world-class Eclipse development platform (www.eclipse.org). In particular I was responsible for several new features and bug fixes in the 3.0 release.

As a member of the core UI team I was responsible for implementing and supporting the code behind key visual components. I was also one of the key contributors to the Rich Client Platform (RCP), which was an initiative to move the utility of the eclipse platform beyond its traditional application as an IDE.

One of my roles as a member of the core Platform UI team was to interact with the underlying Runtime and SWT development teams, as well as with the higher level application-specific teams. As part of that collaboration I co-authored an article on applying branding elements to RCP applications (http://www.eclipse.org/articles/Article-Branding/branding-your-application.html).

July 2001 – Dec 2003	**Innovance Networks Inc.** **Embedded Software Developer**

> SKILLS USED
>
> C++, VxWorks, Solaris, cppunit, gnu tool-chain, flex, bison, multi-threaded development, server development, API development, CMISE (ITU-T X.710, X.734), distributed applications, Solaris DOORS, purify, gdb/ddd

Innovance developed the world's most advanced DWDM optical-mesh transport equipment for long-haul telecom networks. Our team provided the core OAM framework used by the embedded applications to manipulate the network elements.

A differentiating feature of our framework is presentation of the underlying distributed information model in a consistent and transparent fashion throughout the heterogeneous (Solaris and VxWorks) cluster of platforms composing a network element. Our implementation of the ITU-T X.710 CMISE standard provided a uniform API even though the true source was often located several processor hops, and possibly thousands of kilometers, distant. My responsibilities in this area ranged from the development of a clean application interface, to the implementation of significant portions of the server that provided the framework services on every processor in the network.

One of my key contributions was the design and implementation of a subsystem that provides reliable propagation of events throughout the network. This implementation was required to detect and manage the loss of underlying communication channels to release applications of this responsibility.

Dec. 1999 – July 2001	**<u>jfront incorporated</u>** **Consulting Services (founder)**

> SKILLS USED
>
> C++, Java, and Smalltalk Development, EJB (jboss), JSP (tomcat), Testing (junit), XML, make, parser generators (flex/bison)

This is a company I founded to pursue contracting opportunities. During this period I divided my time between revenue generating contracts, and professional development. The software I developed as learning exercises is available on the company's website at www.jfront.com; what follows is a brief description of the projects in which I had greatest interest.

Java Semaphore: Java has built-in support for inter-thread synchronization, however it is not the simplest mutual exclusion mechanism for all algorithms. The jfront.util.Semaphore class is a 100% Java based implementation of an easy to use semaphore. This implementation makes heavy use of Java's built-in functionality to ensure efficient performance.

jfront Sequence Generator: Most databases provide a platform proprietary utility to generate unique primary keys. Using the proprietary mechanism ties the application (e.g., the EJB) to a specific platform. I developed this component (an EJB) to be a platform independent solution.

jfront.ejb.* package: There are several bean requirements described by the EJB specification. Some of the requirements are not at all obvious and can lead to subtle bugs (e.g., a stateless session bean must provide an #ejbActivate method, however it is never invoked). I designed this package to encapsulate the requirements and allow the compiler to detect most of these types of errors.

jfront rawjava: This is a C++ library for parsing Java source code. The original application was the front-end to a Java compiler, but it has since been used in several other applications, such as the front-end to a code analysis utility. For this project I used flex and bison as parser generators to implement the complete syntax from the Java Language Specification.

For all of these products, I've performed all stages of development, from initial conception and documentation through to final testing and distribution. Many are available with complete source code for review at www.jfront.com/products.

Dec 2000 – Feb 2001	**<u>Rational Software Corporation</u>** **Senior Software Engineer [contract]**

> **SKILLS USED**
>
> C++ Development, UML, make, Parser Generators (bison/flex), MS DevStudio, Rational Rose & Rational ClearCase, Code Refactoring

Rational produces a variant of the industry standard Rational Rose UML modeling tool -- known as Rational Rose RealTime, targeted at the production of real time software systems. Starting from a high level model the tool produces executable code that runs reliably and efficiently in real time embedded applications.

I was contracted prior to product release to fix customer reported problems within the code generator. In addition to fixing all reported problems, I performed a general inspection of the code; finding and repairing as yet unreported bugs.

This project was an ideal setting in which to demonstrate the value that jfront inc. is able to offer. The code generator is the heart of the client's product, customers would not tolerate unresolved issues – high quality repairs were essential. Further, the product release date was previously announced – any slip in the schedule would be visible by the client's customers. I succeeded by delivering high quality repairs that exceeded the client's release schedule.

Sept 2000 – Oct 2000	**<u>Espial Group Inc.</u>** **Java Software Developer [contract]**

> **SKILLS USED**
>
> Java Development, java.io package, java.net package, javadoc, Low-level Interaction with HTTP, MS SourceSafe, J2EE Exposure, Mentoring

I was contracted to aid the development of a server that stores Java applications in a central database. These applications are downloaded to a proprietary Java environment, which runs on a variety of embedded devices (e.g., cell phones, set-top boxes, PDAs, etc.).

I was responsible for the design and implementation of a protocol to allow the client device to communicate reliably with the application server. This protocol used http as the transport layer and implemented a series of specialized commands to provide the needed features. The project made heavy use of the http protocol as well as Java's input and output streams.

This project implemented a new piece of functionality in a major release of the client's product. The client considered these features vital to their future success.

April 1998 – March 2000	**<u>ObjecTime Limited</u>** **Software Developer**

> **SKILLS USED**
>
> Smalltalk & C/C++ Development, Embedded Development Techniques, Code Refactoring, UML, Application Architectures, Code Migration, Application of Design Patterns, Interpretation of Language Standards, Compiler Implementation, MS DevStudio, gnu tool-chain, MS SourceSafe, RCS

ObjecTime (now part of IBM) produces a tool for creating executable, real time applications from architectural diagrams. This tool was initially implemented in Smalltalk and was then successfully migrated to a C++ based implementation.

During my tenure I was involved with several projects; from maintenance and evolution of the Smalltalk based tool, to implementation of new features in the C++ based code generator. These projects allowed me to explore the implementation of Object Oriented features in a variety of languages.

One of the most significant projects was the implementation of a C++ based code generator. In addition to being a member of the team that created and maintained the code generator (a C++ implementation that produced C++ code) I was solely responsible for design and implementation of a C code generator (a C++ implementation that produced C code). During implementation of this code

generator I was able to effectively leverage the C++ generator's code, demonstrating significant code and architectural re-use.

Publications

Presentation: <u>Models, Generated Code, and Eclipse</u>
Presented at Ottawa Eclipse Demo-Night, May 26, 2010
Slides: http://www.jfront.com/pubs/20100526-eclipse-demo-night.pdf

Eclipse Corner Articles: <u>Branding Your Application</u>
Co-Authored with Pascal Rapicault, Sept 2004.
http://www.eclipse.org/articles/Article-Branding/branding-your-application.html

University Honour's Project: <u>Generating Java Class Files from Smalltalk Objects</u>
Submitted April 19, 1999
http://www.jfront.com/pubs/jasper.pdf