

MATTHEW BISSON

(Street Address Omitted), Jamaica Plain, MA 02130

Email: [me \[at\] cyberbisson.com](mailto:me[at]cyberbisson.com) | Phone: (###) ###-####

Web: cyberbisson.com | LinkedIn: [/in/cyberbisson](https://in/cyberbisson)

OBJECTIVE:

To contribute my strong technical skills and experience in a position with a growing, dynamic firm.

PROFESSIONAL PROFILE:

- Excellent organizer with solid planning and problem-solving skills.
- Team player, adept mentoring and elevating less experienced teammates.
- Quick learner who can rapidly master all aspects of a job with limited training.
- Enthusiastic, knowledge-hungry self-starter, eager to meet challenges and quickly assimilate newest technologies, skills, concepts, and ideas.

WORK EXPERIENCE:

October 2011 – **VMWARE, INC.** Boston, MA
PRESENT *Staff Engineer — Core Storage*

- Designed, studied, and implemented a prototype “no-stun” Light-Weight Delta (LWD) mechanism in C and C++ for snapshotting Virtual Machines with the goal of replicating them from any storage medium to any other storage medium. This became the snapshot technology that backs the VCDR product.
- Designed, implemented, and documented a Transport API for VMware partners to utilize the LWD system without requiring access to VMware internal libraries. This integrates VMware’s \$137m acquisition of Datrium to provide a high-performance DRaaS solution to and from the cloud.
- Designed and implemented a high-throughput, non-blocking inter-process messaging system using UNIX Domain Sockets (UDS) and TCP/IP. This was written in C with a C++ interface, with the goal of sending messages in the “hot path” of I/O on a Virtual Machine.
- Enabled Microsoft VSS “quiesced” snapshots by implementing a writable snapshot.
- Beginning with the prototype phase, followed a collaborative project with Dell/EMC to do data replication for Virtual Machines on vSAN.
- Provided an asynchronous I/O implementation of the C interfaces used to access vDisks using `libaio`, a kernel-based I/O submission library for Linux and ESXi.

Senior MTS — Disaster Recovery

- Solved various C++ infrastructural problems for vSphere Site Recovery Manager (SRM)—VMware’s second highest selling software (behind only the vSphere management platform itself). Among other things, this involved implementing persistent-connection functionality, maintaining the in-house ORM emitters and support modules for relational database access, enhancing the external API, and developing the remote licensing synchronization and SRM feature-restriction module.
- Implemented `libc` routines for user-space and kernel-space in C and (when beneficial for quick execution) 64-bit ARM (AArch64) assembly. This was implemented on an emulator before real hardware was available.
- Contributed to discussions around VMware’s core C++ infrastructure, implementing changes to SRM or the core code-base as required. Presented SRM’s requirements and point-of-view for the core team’s consideration, and kept abreast of infrastructure changes as they affect SRM.
- Researched and wrote the section for C++ Lambda Expressions in the company-wide coding guide, allowing adoption thereafter.

MATTHEW BISSON

(Street Address Omitted), Jamaica Plain, MA 02130

Email: [me \[at\] cyberbisson.com](mailto:me[at]cyberbisson.com) | Phone: (###) ###-####

Web: cyberbisson.com | LinkedIn: [/in/cyberbisson](https://www.linkedin.com/in/cyberbisson)

August 2008 –
October 2011

RESEARCH IN MOTION

Andover, MA

Software Engineer

- Developed an extensible mechanism to transfer data on the BlackBerry to a desktop or tablet. This was the main e-mail, IM, and PIM service with the first version of the **PlayBook** tablet. Created the BlackBerry Messenger application, as well as various back-end (HTTP server) modules, and an API for 3rd-party use.
- Designed a new device-side Options Application based on detailed user trials and new UI guidelines. Created supporting architecture and API to manipulate how options are presented to the end-user. On an extremely tight deadline, altered over 100 different, unrelated modules to migrate them from the deprecated Options API to the new version—this was to make the integration as seamless for the RIM software engineering team as possible.
- Implemented the mechanism that exposes a 3rd-party Text-to-Speech engine (written in C) so that it conformed to the Java API specified in **JSR-113**—the first such implementation on a mobile device. Coordinated with various groups inside RIM, as well as with the companies that created the speech engine, with the authors of the Java API specification, and with the 3rd-party developers that ultimately used the text-to-speech functionality.

October 2007 –
August 2008

TYCO (FORMERLY INTELLIVID)

Cambridge, MA

Senior Engineer

- As part of a two person team in a small start-up, took responsibility for all interaction between the supported operating systems (SuSE Linux and Windows), the databases (PostgreSQL), and the video analytics software. Among other things, this included networking modules, threading and mutex abstractions, a robust logging and tracing mechanism, and code to store data in a SQL database in real time with billions of writes per day. This work was entirely in C++, and made heavy use of the Boost libraries.
- Implemented and maintained the various tools needed to manage a distributed, multi-threaded video recording and analysis system.
- Built a tool set for debugging running processes to compensate for inadequate debugging software.
- Optimized DVR software and expanded its capability to read files far larger than the previous maximum size of 2 GiB.
- Validated new OS and hardware requirements for a minimum of 16 cameras per tracker at maximum system load.

July 2005 –
August 2007

PALM, INC.

Andover, MA

Software Engineer

- Participated in the first generation **Foleo**—a new category of ultra-mobile phone companion.
- Ported and took ownership of the C++ based Model-View-Controller (MVC) framework that underlies both the Palm OS VersaMail and the Foleo email and PIM applications.
- Collaborated with system SDK team to refine and suggest new APIs.
- Designed new features and applications as a member of the Advanced Technology Group for the Andover Development Center, while maintaining architectural consistency throughout the development cycle.
- Created a scanner and parser using LEX and YACC to parse and translate Palm OS resource files into another representation. This allowed the use of existing Palm OS translations on Foleo, as well as generate C header files for them.

MATTHEW BISSON

(Street Address Omitted), Jamaica Plain, MA 02130

Email: [me \[at\] cyberbisson.com](mailto:me[at]cyberbisson.com) | Phone: (###) ###-####

Web: cyberbisson.com | LinkedIn: [/in/cyberbisson](https://in.linkedin.com/in/cyberbisson)

June 2003 –
July 2005

KEANE, INC. (FORMERLY ARCSTREAM SOLUTIONS)

Cambridge, MA

Consulting Software Developer

- Designed, developed, and maintained a module in Win32 C++ to download books and perform actions to both Palm OS and Pocket PC devices as directed by an HTTP server; this module was the one and only connection unifying the hand-held device with an organized library of downloadable medical volumes, magazines, PDA tools, and other content.
- Designed and developed a drug interaction tool for Pocket PC devices using eMbedded Visual Studio and MFC that finds interactions between approximately 9,000 drugs; users can search between two drugs, two classes of drugs, or find all interactions for one specific drug, then navigate to a series of monographs describing the interaction.
- Maintained the Palm OS version of the drug interaction tool, including the module that generated PDB files from Oracle data.
- Designed and implemented an Oracle stored procedure-based lookup module and data model to link ICD-9-CM clinical codes to medical content; the module is flexible and efficient enough to support over 150,000 requests per hour with response times below 1 second.
- Enabled Digital Rights Management (DRM) of downloadable eBooks.
- Designed XSL style-sheets and leveraged Xalan-J to implement a Java based server component that transforms and caches XML data into a PDA readable format that is downloaded by thousands of users on a daily basis.

Spring 2001;
Summer, Fall
2000

AVAILANT

Cambridge, MA

Junior Software Engineer

- Ported COM and Win32 applications to the Linux platform for the C++ based High-Availability software, ActiveManager.
- Implemented inter-process communication and scalability by creating a C++ module for packaging information and sending it via HTTP.
- Fulfilled a variety of needs during the project by creating utility classes for managing multiple threads, monitoring the status of those threads on a timed interval, as well as for many other necessary tasks.

Junior Release Engineer

- Debugged both run-time and compile-time errors in the nightly builds.
- Improved and documented an obsolete build environment by developing several script-based utilities that could be run on a nightly basis.
- Wrote scripts for installation and packaging.
- Maintained the defect-tracking database.

Winter 2000;
Spring 1999;
Summer, Fall
1998

RATIONAL SOFTWARE CORPORATION

Lexington, MA

Quality Assurance Intern

- Created a Perl WebDAV/HTTP client to test a new web server protocol (**Delta-V**), as well as an XML parser to read the information transferred by the WebDAV server.
- Simplified command-line applications by creating graphical user interfaces.
- Aided developers by developing programs to exercise low-level file functionality in ClearCase.
- Assisted with development on the product, ClearCase, to translate Trusted POSIX Offsets to Windows NT Security Identifiers, thereby allowing access from a fluid Windows NT domain structure.

MATTHEW BISSON

(Street Address Omitted), Jamaica Plain, MA 02130

Email: [me \[at\] cyberbisson.com](mailto:me[at]cyberbisson.com) | Phone: (###) ###-####

Web: cyberbisson.com | LinkedIn: [/in/cyberbisson](https://in.linkedin.com/in/cyberbisson)

EDUCATION:

NORTHEASTERN UNIVERSITY

Boston, MA

College of Computer Science

1997 – 2002

Bachelor of Science in Computer Science, 2002

Related Courses: Analysis of Programming Languages, Compiler Design, Analysis of Algorithms, Automata, Computer Organization and Design, Functional Programming and Its Applications, Distributed Operating Systems, Computer Graphics, Operating Systems Concepts, Artificial Intelligence, Algorithms & Data Structures, Software Design and Development, Object Oriented Programming and Design.

COMPUTER KNOWLEDGE:

Languages: C++, C, Python, Java, Perl, LISP, Scheme, Bourne Shell Script, HTML, SQL.

Systems: VMware (ESXi, vCenter), Linux (Gentoo, Ubuntu, SuSE, Red Hat), Apple macOS, Microsoft Windows, BSD (FreeBSD, OpenBSD), UNIX (AIX, Solaris, SCO, HP-UX, Irix, Compaq TruUnix, DEC UNIX).

Software: GNU Compiler Collection (GCC, GDB), Clang, Microsoft (Visual Studio, WinDBG), Git, Perforce, LEX, YACC, Oracle (*log*, *9i*), PostgreSQL, \LaTeX .

Open Source: Contributed to GNU Emacs, ClangFormat, FVWM2.

Special Projects: Created a debugging tool to print stack traces for all threads in a running Win32 process. Created Palm OS Resource file compiler, “Quirk19” compiler, 3D Ray Tracer. Self taught Oracle DBA, Java and Visual BASIC.

References Furnished on Request