

I have more than ten years of experience as a software engineering lead and senior level individual contributor, with a focus on software architecture and large and complex projects. I have developed and refactored systems that involve extensive concurrency, distributed processing and networking. I am comfortable working in Windows, Unix and embedded environments. I develop software using object oriented design techniques and C++ as the primary implementation language. I am comfortable with a broad range of languages and tools. As a software architect, I am involved in all aspects of the design and development process from product definition to final release testing. I have Leadership skills and substantial knowledge of a broad range of operating systems, programming languages and protocols. My solid background in mathematics and physics as well as significant experience in digital design allows me to work effectively in cross-disciplinary teams and come up to speed rapidly when new challenges arise.

Extensive Experience: Software Architecture, Advanced C++, COM, STL, Win32, Windows Operating Systems, Windows Driver Development, Unix, Perl, ATL, NT Service Programming, TCP/IP, Sockets, Multithreading, C, Embedded Systems, Assembler Languages (Including Intel x86), SCSI, TIFF, JPEG, Data Compression, MAPI, SNMP, XML, VSS, Dicom-3, HTML, Engineering Development in an FDA regulated environment, Embedded Windows, high speed data acquisition, image processing, 3D transformations, MRI. Java,

Some Experience with: Active-X/OLE, Sonet/SDH, MFC, .NET, C#, JavaScript, XSLT, Cryptography (Symmetric and Public Key), Postscript, IEEE802 Ethernet protocols, Lisp, UML, SQL, ODBC, Relational Databases, GIT, web development, stepper motors.

Education: Bachelor of Science Cum Laude from Northeastern University in Electrical Engineering.

Kiva Systems/Amazon Robotics: 2014 to Present

Senior Software Engineer

Configuration Management Portal for Next Generation Warehouse Robotics Platform

- Technical lead developing configuration management portal for SOA/AWS based next generation warehouse robotics platform. Java based system running on AWS hosts using S3 and DynamoDB for storage. JSON data transport and RESTful interfaces between components.
- Gathered requirements from the organization, generated architecture and design approach and validated with internal customers. Currently leading the process of detailed design and implementation of the components.

GE Healthcare, Healthcare Systems: 2009 to 2014 (GE Acquired ONI)

Software Architect

Specialty MRI Products Software Development

- Responsible for the continuing development of the specialty MRI products that were acquired from ONI Medical Systems. Technical lead, software architect and individual contributor.
- Drove the specification for next generation PCIe to MRI scanner hardware interface design developed by a team in Bangalore, India. Developed Windows driver code and interface design for this hardware and carried out hardware/software integration. Worked with the FPGA developer in Bangalore to locate and address issues with the FPGA/Driver integration and develop tool code to diagnose internal FPGA issues. Adapted the main user mode scanner service code to integrate this new hardware into the existing MRI scanner software as a demand loaded plug-in. Implemented these software changes and integration with the new hardware is in progress.

- As software technical lead I have been involved in most activities that the software team has undertaken and acted as fall back for particularly difficult software issues. I have been involved to varying degrees in the software release cycle and training and mentoring the other engineers on the team.
- Redesigned the core MRI scanner service code for substantially greater concurrency and more flexible run-time loading of core components.

ONI Medical Systems Inc: 2005 to 2009

Principal Software Engineer

MRI Scanner Software Development

- Overall responsibility for the technical integrity and direction of the development of software for ONI's MRI scanner products.
- Technical lead for a small team of software engineers.
- Extensive software development in C++ on the XP embedded platform involving both refactoring of existing code and significant evolutionary extensions of that code both as a direct contributor and as the lead providing direction to the rest of the development team.
- The software platform for ONI's MRI scanner involves a wide range of components from high speed data acquisition code with tight recursion intervals and mathematically intensive image reconstruction algorithms to Dicom medical image networking and a full windows user interface.

Groove Networks Inc: 2003 to 2005

Principal Software Engineer

Enterprise Integration Server (EIS) Product

- Technical lead for EIS at Groove.
- The EIS product allowed large enterprises to automate connections between Groove's enterprise peer to peer client products and existing in-house information infrastructure. Coming on board as the technical lead for this product I needed to become familiar with Groove's platform infrastructure and the EIS framework that is built on top of it.
- This code-base contains more than six million lines of highly customized COM/ATL C++ code and supporting JavaScript.
- This is a heavily threaded server communicating over encrypted socket connections and expected to have high availability in a corporate enterprise environment.

Tektronix Inc and Digital Lightwave Inc: 2000 to 2003

Software Architecture Lead.

Software Architecture Lead for the OTS9000 Test System Product Line

- Architecture lead and individual contributor for the software platform of the OTS9000 optical networking test product line.
- Heavily threaded COM based infrastructure with an NT service hosting in-process components and custom kernel mode drivers to manage the instrument modules, a DCOM remoteable user-interface frame hosting ActiveX controls to provide interactive control and a separate modular SCPI interface process providing automated control capability. Total system contains more that 800, 000 lines of C++ code.
- Evolved Windows NT/2000 based software architecture for a compact PCI based modular test instrument from supporting one set of modules developed by one team in a single location to support five development teams in three widely separated locations while successfully releasing seven major versions of the system. Modules developed by the various groups are interchangeable as needed by the customer.
- Extended software architecture, infrastructure and supporting components to support emerging requirements, streamline development and reduce the potential for errors. Documented existing design and streamlined implementation. Redesigned system components as needed to enhance modularity, improve robustness and address new requirements. Implemented new COM components and C++ code and refactored existing components, frameworks and driver code. Created base class hierarchies and template classes to reduce duplication of code and improve system robustness.

- Drove programming and process 'best-practices', instituted and implemented modular automated build process and nightly builds. Restructured the code-base so that code related to individual modules is kept together to make development by multiple teams easier.
- Provided training and mentoring to the rest of the development teams on both platform issues and broader software technologies and techniques.
- Managed two direct reports who worked on the software platform. Gathered initial requirements, developed project charter and implementation plan and coordinated work on the software platform with ongoing development by the module development teams.

Xerox Inc.: 1998 to 2000

Software Technical Lead.

Redesign of the Document Centre Core Software to Target Windows NT Embedded.

- Technical lead in the redesign of 250,000 lines software system from an embedded C/Unix (LynxOS) environment into a C++/Windows NT Embedded environment.
- Initial port was carried out using a Unix compatibility toolkit (NutCracker) where possible to reduce development time. Analyzed existing design, developed the conversion process, converted and enhanced the core libraries and assisted the rest of the team in carrying out the conversion of the balance of the system.
- Leading contributor to the architectural redesign. Provided technical leadership and platform/language training to the team. Ensured the overall technical integrity of the new code-base and related processes.
- Designed shared memory based interprocess communications mechanism as an NT replacement for Unix domain sockets.

Designed and Developed Prototype Modular SNMP Implementation.

- Developed a new modular SNMP implementation using Microsoft SNMP extension agent technology. Design allowed retention of existing code with incremental replacement by new style modules. Ported existing MIB implementations from Epilogue based framework under Unix to Windows SNMP agent environment.

Puma Technology Inc: 1998

Senior Software Engineer.

Outlook Translator Development for Intellisync.

- Extended and enhanced COM based PDA to desktop data synchronization manager for Outlook 98 (MAPI). Implemented substantial performance enhancements and new features.
- Designed and implemented a standard C++ library to support translator development needs.

Howtek Inc.: 1992 to 1998

Senior Software Engineer.

Windows NT Based Dicom-3 Medical Image Server:

- Evaluated options and requirements for software aspects of Howtek radiological film digitizers and related products.
- Designed and implemented multithreaded Dicom-3 database service on Windows NT 4.0 in C++ for storage and retrieval of secondary capture images including full query functionality. The Dicom-3 interface implemented using the MergeCom-3 library.
- Investigations towards a SQL/ODBC based data store and web based maintenance access were pursued but were not productized.

JPEG Codec Support and Dicom-3 Image Transmission Management Library:

- Designed, implemented and validated modules to manage lossy and lossless JPEG transmission of medical images using the Dicom-3 protocol. Dicom-3 connectivity implemented using the MergeCom-3 library. JPEG implementation based on the IJG lossy codec and the Cornell lossless codec. Took on the role of image compression expert for the medical imaging team.

Howtek Scanner Toolkit for Windows NT and 95:

- Designed and implemented the Howtek C/C++ third party scanner interface library providing software developers with thread safe access to all Howtek scanners while isolating developers from extraneous details.

- This library went through three major revisions and has been incorporated into a number of commercial products while maintaining full drop in backwards compatibility.
- Library supports both C and C++ development for third party developers. It is fully thread-safe and includes a 'checked' build with vastly enhanced error trapping.
- Provides a common API for discovery and operation of all models of Howtek scanners. Built on top of ASPI for Win32 for communications with the SCSI bus. Kernel mode SCSI class driver investigated but never productized.

Main Control Processor Firmware for SM2500/Pro-G/DX Flatbed Scanners:

- Designed and developed the core firmware for the Intel 386EX based DX/Pro-G/SM2500 series of flatbed scanners.
- Designed and implemented core operational code for the system.
- Supervised design integrity, integration and testing of all system firmware.
- Implemented low cost stepper motor control for this flat bed scanner.

Main Control Processor Firmware for the SM7500 Drum Scanner:

- Designed and documented the firmware architecture of Howtek's next generation of drum scanners
- Designed and developed the core firmware for the multiprocessor 80186 based SM7500 scanner on top of the AMX real-time kernel. Implemented core firmware support code and libraries.
- Implemented common interprocess(or) communication protocol and library used throughout the system. Communications protocol supported full local/remote messaging transparency (local message queues or shared memory transport).
- Designed and implemented flash memory storage facility with data compression. Ensured overall technical integrity of the system firmware.
- Facilitated the technical interactions of the other four engineers on the project. Provided support and coordination within the firmware team and between the firmware team and the other engineering teams.
- This code base and architecture has been successfully deployed as the firmware supporting at least four subsequent Howtek scanner products using a range of hardware architectures.

Main Control Processor Firmware for the D4000 Drum Scanner:

- Redesigned data acquisition, processing and transport code and flash memory programming code to dramatically improve reliability and performance. Added interrupt support for data acquisition and SCSI transfer management.

GenRad Inc.: 1989 to 1992

Digital Design Engineer

228x High Speed Test Set Sequencer Board

- Designed and implemented a high speed custom microprogrammable controller for the 228x board test system.

Test System Interface Board

- Designed a VXI/VME interface to a GenRad proprietary instrument control bus. Interface provided direct access between busses and a 16 MByte shared memory block for high performance transfers.