

RESUME OF: Chris Macgowan
952 Linwood Avenue, Saint Paul, Minnesota 55105
Home Phone: 651.225.9743
macgowan@pobox.com

OBJECTIVE: A challenging position as Software Engineer with an emphasis on Object Technology

EDUCATION: B.S. Degree in Electronics Engineering Technology
Mankato State University, Mankato, Minnesota, 1985.

SOFTWARE ENGINEERING SKILLS:

Strong understanding of the relationships between software design, product testing, maintenance, support and the many factors that are involved in developing a complete software solution.

Developed various client-server, client and embedded systems under Windows, Unix, OS/2, and real-time operating systems. Applied design methodologies and wrote technical specifications using OOAD artifacts and UML. These applications were developed using C/C++, MFC, ATL and COM. Developed, documented and applied automated test scripts using C/C++, PERL and WinRunner.

Database experience includes Microsoft SQL Server, MySQL, DB2/2, Informix and Microsoft Access. Network Operating systems including Unix, Windows NT/2000, OS/2 and Novell NetWare.

Developed firmware and hardware for embedded systems based on 8-bit microprocessors/microcontrollers.

Member and/or leader of various software development teams. Helped develop in-company standards for quality processes, design methods, test procedures and product specifications. Coordinated and managed product testing and support. Authored and edited operating manuals and system documentation.

EMPLOYMENT EXPERIENCE:

Meteorlogix LLC
Software Engineer
19 May 2003 to Present
Minneapolis, Minnesota

Member of the Data Development Group, responsible for design, development, testing and implementation of software components to support data acquisition, infrastructure and product development for Meteorology Software Products in a Windows and Linux environment.

Designed, developed, tested, documented and implemented various data acquisition components. The components handled various weather data provided by the National Oceanic & Atmospheric Administration (NOAA), National Weather Service and third parties. Prior to this implementation data acquisition requirements were added directly to specific applications. Using components allows the components to be reused and improved maintenance. The components were designed using OOAD methodologies and UML artifacts such as Use Cases, Class, Collaboration and Sequence Diagrams. The data audition component will receive data using a file or socket. The data is then parsed and written to a database table using ADO. These components were build as MFC or ATL-COM DLLs using Microsoft Visual C++.

Designed, developed, tested, documented and implemented the Convective Sigmet Processor. A Convective Sigmet is a polygon drawn above a storm to show storm characteristics. The Sigmet Processor Component provided data processing functions to the AviationSentry Product. Created classes and methods to support data management, object derivations, quality control and integrated tested. The components were designed using

OOAD methodologies and UML artifacts. The component was built as a ATL-COM Dll using Microsoft Visual C++.

Designed and developed the Bloomberg Contribution File Manager (Bcfm). The Bcfm Application added header information to the top of datafiles being processed. Using OOAD methodologies and UML artifacts designed the classes used in the application; CBcfmApp, CMonitor, CProcess, CMxReadyFile, CMxHeartBeat, CMxAlert, CMxEventLog and CMxException. The Bcfm Application was built using C++ on the Redhat Linux Operating System. Common components were built into Shared Objects.

Developed the Bloomberg Contribution File Manager Client (Bcfm Client) to manage the Definition Files that are used by the Bcfm Application to create the header information. The BcfmUtil application was created using Microsoft Visual C++ with MFC for the Windows Operating System. Database support was provided using ADO.

As a Member of the Infrastructure Group developed software development guidelines, infrastructure components and created a class library. The software development guidelines were a collection of documents outlining best practices in software engineering. Identified and initiated a collection of shared components. The component interfaces were documented and published in a class library. Continue to manage and maintain the class library. Tasked with research of best solutions to collect process metrics and implement process control into the data processing environment. Completed prototypes of a metrics collection component, metrics server and a peer-peer process control component.

Telex Communications Incorporated
Consulting Software Engineer
August 2002 to September 2004 (Consulting)
Minneapolis, Minnesota

Designed, developed and tested the RE-OneLink Wireless Receiver Controller. The RE-OneLink is a Windows application that monitors and controls wireless receivers. Worked with marketing and engineering to develop product specifications. The RE-OneLink was designed using OOAD methodologies and UML. Classes were designed and written to support USB communications, receiver data, receiver events, event monitoring, error handling and the GUI presentation. Created the InstallShield application. The application was written using Visual C++ and MFC.

TravelersExpress Corporation
Consulting Software Engineer
April 2002 to April 2003
Minneapolis, Minnesota

Designed, developed and tested communications enhancements to the In Store Integration Simulator (ISIS). The ISIS application is a client program used to demonstrate command and communication protocols for point of sale terminals. Implemented a asynchronous sockets class to support TCP/IP communication using Windows Sockets. Developed additional enhancements to provide improved configuration, registry support, message logging and user interface. ISIS is a Win32 console application written in C and C++.

Colder Products Corporation
Consulting Software Engineer
October 2001 to January 2002
Saint Paul, Minnesota

Designed, developed, tested and implemented the HostController Windows Application to control and collect data from an embedded system. The HostController was designed to exercise the embedded system and

Resume of Chris Macgowan

provide a sample data acquisition application to demonstrate the embedded systems capabilities. Classes were designed and written to support RS-232 serial communication, data collection and management, embedded command processing, error handling and the GUI presentation. The HostController Application was written using Visual C++ and MFC.

VERITAS Software Corporation
Staff Software Engineer
March 1999 to May 2001
Saint Paul, Minnesota

Participated in a team responsible for design, development, testing and implementation of software features for the VERITAS NetBackup Product. These features supported the backup of Lotus Notes and general client enhancements.

Designed the Progress Viewer ActiveX Component. This software component was designed to be reused in the NetBackup Client and NetBackup Job Tracker Applications. Gathered specifications and created requirements for design using OOAD methodologies and UML. Implemented classes to support server messaging, event handling and GUI presentation. A prototype of the ActiveX Component was written in Visual C++.

Designed, Developed and Tested the NetBackup Client Find Dialog. The Find Dialog presented objects available to be restored based in the users search criteria. Design specification were written using OOAD methodologies and UML. This feature required new interfaces to the server to support the search component and also development of the GUI. This feature was developed using ATL. Worked with Technical Publications to update the documentation.

Designed and developed an enumeration component to identify Lotus Notes Objects for backup on the client. This software component used the Lotus Notes API and the NetBackup Universal Browsing System to create a list of objects that is sent to the server for backup. Worked with Test Engineering to design, develop, document and apply automated test scripts using PERL to test component. This software was written in C to provide platform independence between Windows and Unix.

Developed the client GUI features required to support Lotus Notes on the NetBackup Client. The GUI allowed the user to select/set parameters prior to starting the backup. The NetBackup Client was written as an MDI Application using the Microsoft Document/View Architecture. UI Features tested using WinRunner. These software components were written using MFC.

Target Corporation
Senior System Developer
October 1994 to March 1999
Saint Paul, Minnesota

Member of various teams responsible for software design, development, testing, implementation and support for software systems in Target Stores. This environment included 800 stores, each with a single server running Windows NT and many clients.

Applied object oriented techniques to develop common business classes to be re-used in the general retail operating environment. These classes supported data access, messaging, process control and event logging. Data access classes used ODBC and CRecordset to access the database. Created test applications to verify business rules of each object. The client application and business objects (server) were developed using Visual C++ and MFC.

Designed, developed, tested, implemented and supported a Retail Chargeback System. The program is used to

Resume of Chris Macgowan

manage returned product in the store. Developed functionality to support general returns, shipping, automated chargeback creation and UPC scanning. Worked with QA to build test scripts using PERL and WinRunner. Worked with clients in a usability lab and providing leadership in documentation and version control. The application was developed using C in OS/2 and DB2/2.

Higher Education Assistance Foundation / TGA
Software Consultant
June 1992 to August 1994
Saint Paul, Minnesota

Developed, maintained and provided end-user support for human resources, inventory and loan management client-server systems written in Clipper, Power Builder and C. Authored end-user on-line and written documentation.

Siemens AG
Software Engineer
Jan 1991 to February 1992
Berlin, Germany

Designed, developed, implemented and supported the Quality Assurance Management System. The application was designed to collect, maintain and report on quality assurance data collected during the production of fiber optic transceivers. The software was developed on an DOS/Netware environment using Clipper. Authored end-user on-line and written documentation. I was employed by Siemens AG as part of a professional exchange program between Germany and the United States sponsored by the The Carl Duisberg Society.

Smarte Carte Inc.
Design Engineer
December 1985 to January 1989
Saint Paul, Minnesota

Designed, prototyped and tested three microprocessor based embedded systems for luggage cart rental systems in airports. The Intel 8051 based systems were used to manage data acquisition, control the operation of cart rental/return mechanisms, coin dispensing/accepting devices, bill acceptor, credit card reader and a cart identification system. Application software was written using an 8051 Cross Assembler and C Compiler.

INDEPENDENT SOFTWARE PROJECTS:

Media Facilities Management System
Software Design and Development
April 1998 to Present
Saint Paul, Minnesota

The Media Facilities Management System (MFms) is designed to management media equipment, facilities, media, people, and their transactions. The MFms System is designed using OOAD methodologies and UML. The Client Server Application includes server software and client software. The server is written using Visual C++ and MFC. The client is written in Java and uses TCP/IP to communicate with the server. Database support is provided with ODBC and DAO. Shared components are written using C++, VB and ATL. The install package is written with InstallShield. The application is currently being prepared to be released as Open Source.

References will be provided upon request.
v26