Daniel Hugh McInnes

0415 208 480

d_h_mcinnes@hotmail.com

Personal Summary

I have dedicated my career to becoming the world's greatest software engineer. I'm not there yet, but every year I'm getting closer. I read every book I can find on the subject. I want to make software-intensive systems better, cheaper, and faster.

Professional Experience

October 2017 – May 2019: Software Engineer at tritium

Developed the HMI for high powered car charging (sole developer). Internationalization, localization. **C++, Qt, Linux, QtScxml, State Charts, QtLinguist, Cmake, VSTS.**

OCPP 2.0 Client development. Implement the OCPP comms protocol between a high powered charger and any OCPP server. **C++, Linux, CMake, VSTS**.

OCPP 1.6J Customer Integration Testing. Verify that our 'OCPP client' works with different vendors' servers. **C++, Linux**.

July 2017 – August 2017: Senior Software Engineer at Topcon Positioning Systems

Precision seeder development. **C++, Qt, Linux.**

Aug 2014 – June 2017: Software Engineer - System Lead at AgJunction

System Lead for the Headland Automation project. **C++, Qt, Linux.**

Safety-critical system development - ISO25119 analysis and compliance. Risk mitigation.

PCU2Demo project - developed touch screen terminal for tractor control. **C++, Qt, Linux**.

Created Trade Show Demo Stand, a standalone system containing a steering wheel, joystick, control unit, and GUI. This allows sales staff to demonstrate guidance systems to customers. **C++, Qt, Linux**.

Created eDriveSim, a desktop tractor simulation tool to exercise all aspects of in-house guidance systems. **C++, Qt, Linux.**

November 2011 – July 2013: Project Engineer at Leica Geosystems Mining

Greenfield commissioning of mine automation system at Peabody's Wilpinjong coal mine.

Project Engineer - Upgraded the Leica system at Xstrata's Ravensworth coal mine. Diagnosed issues, liaised with FMS coordinator and Leica's product development team. Rolled out new onboard hardware platform "UHP" across the haul truck fleet.

Software Engineer - Developed software for the mining industry using **C++**, **Objective C, ruby, PostgreSQL, Sqlite**.

Developed server-side r**uby** scripts handling design file transfer.

Added a **REST**ful interface to the MineOPS product using **NGINX** and **FastCGI**.

Integrated 3rd party attendance / swipe-card system with the MineOPS product.

Wrote interface for **MSSQL** database - RESTful interface.

Modified the 'Auto - Lineup' algorithm to assign workers to equipment across multiple pits (**Objective C, Ruby, Postgres, Sqlite**).

June 2010 – October 2011: Software Engineer at invensys

Developed design documentation, software and test cases for the flagship 'Train Control Management System' for the Melbourne metro system and the London Underground. Interfaced multithreaded application software with a single threaded SCADA package. Added to the Timetable Interface, Timetable Archiving, Service Performance, Fleet Allocation, and Crew Roster systems. Bug fixes for the SystematICS SCADA package. **Safety-critical**, **real-time Boost C++** on a **Linux** (ubuntu) x86 platform.

2008 – April 2010: Senior Software Engineer at Queensland Rail

Developed requirements, design specification, software and system test documentation for the 'GPS Stage 3' project. In this project I extended the GPS functionality of the 'Controller Workstation' software used in the control room. I added numerous features which improved the logistics and safety of running bulk coal from mines to ports. **Safety-critical, real-time C++** on a Windows XP platform.

2007-2008: Software Engineer at RSA – The Security Division of EMC

Modified and extended the automated test system to run test cases on multiple platforms. Migrated an existing build, test and release system (a collection of **shell** and **perl** scripts) to Anthill pro.

Implemented new features as requested, writing cross platform code in **C** for 50+ platforms (Multiple flavours of Windows, Linux and Unix).

2005 – 2006: Software Engineer at Safenet (SFNT) Data security, Encryption, Smart Cards.

"Internationalized" ProtectDrive product – converted from MBCS to UNICODE character sets. (**C++**, multiple platforms)

Developed 16 bit PCI driver for a smart card reader (TI PCI6621) on PC platform. (**C**)

Developed 16 bit USB drivers for smart card readers (Kobil, HP USB keyboard/smart card reader) on PC platform.(**embedded, C**)

Converted existing DOS based utilities to the COM framework.(**C++**)

Debugged existing USB host controller software.(**C**)

2003-2005: Software Engineer at Dawson Technologies Power generation, Engine control.

Developed CANBUS, MODBUS ASCII, MODBUS RTU drivers.(**embedded, C, real time**)

Developed the 'AC Card' system, performing AC power measurements using TI TMS320F2812 digital signal processor. (**embedded**, **C++**, **real time**)

Developed the 'Remote Display Module' system. This unit connects to a CANBUS network and displays diagnostic and control data from several systems on an LCD screen.(**embedded, C**, **real time**)

Developed all interrupt service routines for several Motorola HCS12 based products. Developed all device drivers (ADC via SPI port, LED displays via SPI port, LCD panel drivers via general I/O, Real Time Clock interfaces, Keypad Interfaces etc.) (**embedded, C**, **real time**)

Upgraded software for the Remote I/O module – chip changed from Motorola HC12 to HCS12. (**embedded, C**, **real time**)

Upgraded software for the Temperature Scanner module – chip changed from Motorola HC12 to HCS12.(**embedded, C**, **real time**)

2001 – 2003: Software Engineer at Xerox (XRX) Photocopiers, Document solutions.

Implemented the 'Protocol Report' and 'Cover Letter' features. (**embedded, C**)

Maintained and debugged existing software for the high speed 'Electronic Pre-Collation' memory for the 'Video' subsystem. (**embedded, C++**)

Maintained and extended the capabilities of an existing DOS-based 'PC User Interface' – a tool for diagnosing faults and changing system settings within the print engine. (**embedded, C++**)

Designed, documented and implemented a new Windows-based diagnostics tool for the print engine using **C++** in a multi-threaded environment.(**embedded, C++**)

Education

2013 – (ongoing). University of Queensland
Master of Engineering (Software Engineering)

2010 – 2012 University of Queensland
Graduate Certificate in Software Engineering

1995–2000 Griffith University. Brisbane, QLD

Bachelor Of Engineering in Microelectronic Engineering

Tools Used

C#:

Microsoft Visual Studio 2012.

C/C++ tools:
Microsoft Visual Studio 2012, 6.0, .net 2005.
'Gnu' compiler.
Programming Research QA C/C++
Imagecraft HCS12 ‘C’ compilers
Texas Instruments code composer studio C++ (TMS320F2812)
Numega TrueCoverage
Boost C++ libraries

Databases:
PostgreSQL
MSSQL
Sqlite

Web Servers:
NGINX
lighttpd

Software Configuration Management tools:
Git

Mercurial
Subversion
Others: Rational Clearcase, Rational Synergy, Rational Change, Teamware, CVS, Seapine Surround

Requirements Management Tools:
DOORS
Rational Requisite Pro

Design Tools:
Artisan (UML package)
Rational Requisite Pro
Poseidon (UML package)
StarUML (UML Package)
BoUML (UML Package)

Development Methodologies:
Agile
Team Software Process
Personal Software Process
Rapid Prototyping
'Fagan' defect-free software delivery process

Operating Systems:
VxWorks
RunTime (proprietary Xerox operating system)
Nucleus
Linux (ubuntu, debian, OpenWrt)
Windows XP → Windows 7
I have developed custom embedded operating systems for several projects.

Quality Tools:
Valgrind
Rational Purify (dynamic verification tool – detects memory leaks etc.)
Coverity Prevent (static analysis tool)

Embedded CPUs
Motorola HCS12 (automotive)
Texas Instruments TMS 320F2812 (DSP)

Other Interests

Investing, stocks, and CFDs.
Boating and fishing in and around Moreton Bay, Jumpinpin, Fraser Island and the Gold Coast
Bushwalking (Mt Barney National Park)
Camping
Tennis