Chris MacGregor

Principal Software Engineer

206-650-0686
chris@cybermato.com
www.cybermato.com/chris
in/chrismacgregor

Expertise

- Robust, maintainable, extensible, reusable, scalable, high-performance, high-quality, tested, documented software
- C/C++/Assembly in embedded and real-time systems large and small, ranging from Linux on client/server networked systems using 64-bit ARM multi-core CPUs to an RTOS (or no OS) on 8-bit microcontrollers with 12k SRAM
- Architecture, design, and implementation of solid, stable systems, from requirements & research to production
- In-depth low-level optimization to maximize performance of embedded computing systems for e.g. real-time image compression & processing, utilizing experience from many years of work on compiler back ends
- High performance image processing on embedded (CPU- & memory-constrained) and desktop platforms
- Kernels, device drivers, compilers, simulators, development tools
- Libraries; C/C++ API design, implementation, and documentation; runtime environments; firmware
- Technical leadership and mentoring while also working directly on architecture, design, and code
- Learn existing codebases quickly
- Long history of successful work either independently or as lead or member of a remote and/or local team, large or small

Experience

OXFORD NANOIMAGING (ONI) (Oxford, UK and San Diego, CA – Fully remote) 2022 – 2026

- As the Principal Software Engineer overseeing the C++ app and board firmware, provided technical direction to the software team along with significant direct contributions to architecture, design, and implementation. Also provided technical and other advice to management as desired.
- Led team in rewriting user-mode C++ & CUDA app to manage microscope hardware and process images.
- Rewrote entire imaging pipeline to solve stability problems and improve performance (over 100 fps at full resolution).
- Implemented numerous significant app infrastructure improvements, and directed implementation of many more.
- Led C++ software team in adding support for a new microscope model with significant hardware changes.

GOOGLE (Seattle, WA) 2013 – 2021

- Researched and implemented high-fidelity software emulation of 16550A UART for Google Compute Engine (cloud).
- Proposed and led successful project to provide SSH-based interactive serial console access to Google Compute Engine VMs: https://cloud.google.com/compute/docs/instances/interacting-with-serial-console & wrote much of the code.
- Led deep research into heap allocation behavior.
- Various other projects subject to NDA.

WALLINGFORD IMAGING SYSTEMS (Seattle, WA): 2010 – 2013

- Designed & implemented custom video capture and image processing, optimized for maximum fidelity (lossless compression), flexibility, and performance (e.g. up to 14 fps in full 5 megapixel resolution on BeagleBoard xM).
- Fixed and enhanced kernel mode image sensor drivers and platform drivers in Linux kernels from 2.6.3x to 3.x.

CYBERMATO CONSULTING (Seattle, WA and remote): 2003 – 2022

- Designed & implemented numerous significant new features and enhancements for an open source video editing program to adapt it to video analysis of real-world biological behaviors.
- Designed & implemented Linux interfaces to a variety of scientific instruments for oceanographic research groups.
- Ported a Windows-based embedded system to Linux, introducing portability layer, refactoring/restructuring.
- Revamped and optimized legacy web-based front end and data-entry interface to scientific database.
- Built multi-platform (Windows, Mac OS X, Linux) offline data-entry application for scientific database.

ORFLO TECHNOLOGIES (Hailey, ID – Fully remote): 2011 – 2013

- Repaired and enhanced multi-platform (Windows, Mac OS X) application for device control and data analysis.
- Led software architecture, design, prototyping, and implementation for new products in development.

AMAZON.COM (Seattle, WA): 2008 – 2012

- Designed and implemented device-side system software for Kindle (e.g., book-downloading infrastructure).
- Advanced product development work on unannounced products.

DIGEO (Kirkland, WA): 2005 – 2008

- Fully automated build of cross-compilers and complete custom Linux distro from scratch for ARM, MIPS, PPC, & x86.
- Designed & implemented OpenGL subset, including scaling+rotating+flipping blits using C++ templates.
- Modified Linux device drivers to improve performance and implement missing features.

TIGERWAVE NETWORKS (Seattle, WA): 2003 – 2007

- As CTO & co-owner, handled all engineering, admin, sales, tech support, sysadmin, security, software, and hardware.

SONY ELECTRONICS (Mountlake Terrace, WA): 2001 – 2005

- Designed, implemented, tested, documented, and deployed several large new software components for advanced research projects and prototypes, using C++ in a Linux-based embedded environment.
- Client/server-structured configuration info subsystem; infinitely multi-zoned flexible debug/trace output management system; SIP-based VoIP+video communication management system; LCD monitor firmware rewrite for new product.
- Designed, implemented, tested, documented, and deployed fully automated system for testing various aspects of TV functionality, such as compliance with V-Chip requirements (EIA-608B).

BSQUARE CORP. (Bellevue, WA): 1994 – 2001

- Compiler back end implementation for ARMv4 & Thumb (assistant technical lead) and TriCore (technical lead).
- Designed creative solution to maintain backward source-level compatibility despite incompatible calling convention.
- Designed and implemented ARM disassembler and peephole optimizer.
- Implemented Hitachi SH-3 code generation, low-level optimization, & bit assembly using Microsoft front end & linker.
- Designed, implemented, tested, and deployed an assembler for the SH-3, compatible with Hitachi's assembler.
- Interfaced with various customers, primarily Infineon, ARM, Hitachi, and Microsoft.
- Initiated, maintained, and broadened an effective and cooperative relationship with the engineers at Microsoft maintaining tools being ported. Delicately handled some difficult personalities. Most of the interaction was via email.
- Enhanced C runtime in numerous ways; debugged and fixed various WinCE kernel bugs.
- Proposed & implemented several initiatives improving development environment, including many new tools.
- Wrote a sophisticated set of customizable, extensible build scripts for the development tools (in Perl 5).

GTE NETWORK MANAGEMENT OPERATIONS (Bothell, WA): 1993 – 1994

- Designed, implemented, tested, and documented powerful new CM tool using Perl 4, RCS, and Sybase (SQL database).

STRATUS COMPUTER, INC. (Marlboro, MA): 1987 – 1993

- Developed and maintained toolset to automate source control/configuration/build/test management, still used in 2013+.
- Developed and maintained powerful make front end (in Perl) and makefile framework, still in use in 2013+.
- Significantly enhanced & ported GNU assembler (gas) for PA-RISC on proprietary OS.
- Co-developed and maintained a tcsh-like shell for proprietary OS.
- Developed large C library (still in use) to: provide or emulate standard Unix system calls and runtime functionality missing from proprietary OS; provide ANSI & POSIX compliance; ease porting of standard tools and other software.
- Designed, implemented, and tested an i80860 instruction-level simulator, using GDB as a front end.
- Maintained and enhanced an MC88100 simulator used for all in-house MC88100 software development.
- Designed, implemented, and exhaustively tested high-performance MC88100 assembly language runtime routines: variable-alignment data moves and compares, varying-length string operations, PL/I data type conversions, etc.

Education

BS in Computer Science (with Distinction), Worcester Polytechnic Institute (MA), 1990