James Craig Burley

resume-li191015@burleyarch.com

Objective

Architect/Developer role (contract or full-time) involving design, implementation, and deployment of software systems, including operating systems, developer/tester toolchains, and communications protocols.

Profile

- Currently focusing on low-latency architecture and on enhancing Joker (a Clojure interpreter written in Go) to automatically support/wrap the entire Go standard library and, optionally, 3rd-party packages.

- Primary architect/designer and project leader for various projects improving server availability, performance, and capacity at Bloomberg LP, via improved TCP and IPC utilization; Coverity, Purify, valgrind, and a custom fuzzer to validate code; and Splunk, profiling tools, and heatmaps to analyze effects and performance.

- Shepherded substantial portions of large-scale refactoring of Bloomberg LP systems-level run-time libraries (mostly written in C and C++) to improve packaging.

- As a consultant, drove lag time to port a code-generation module, in a Verilog compiler written in C, down from estimated 6 months to nearly nothing via partial automation coded in Lisp.

- Pioneered novel approach to writing test-automation infrastructure using reflection and other techniques while at Microsoft. Also served as Scrum Master.

- As a volunteer for the Free Software Foundation, wrote and maintained a widely-used GCC compiler front end and run-time library.

Recent Systems and Programming Languages

- Unix, OS X, Windows

- Clojure, Go, C, C++, C#

- Networking protocols (such as SMTP)

- Shell programming and scripting (GNU Bash, PowerShell, Perl, GNU Emacs Lisp)

- Software-development/deployment technologies (Git, dpkg, Visual Studio)

- Network and system administration and maintenance (Splunk, Linux, Windows)

Recent Professional Experience

Bloomberg LP, New York, NY
Senior Software Architect, Systems Infrastructure, January 2012 — March 2017

Architected, designed, debugged, and implemented core infrastructure protocols, communications software, libraries, and test-architecture infrastructure. Goals included: supporting more clients and client capabilities; reliable infrastructure; and improved developer workflows. Stack included proprietary multi-threaded applications (written in C, C++, and legacy Fortran) with heavy use of shared memory and related synchronization primitives, leveraging Git, Subversion, dpkg, Splunk, Coverity, Purify, valgrind, etc.

Microsoft Corporation, Cambridge, MA

Senior Software Developer in Test, Application Virtualization, *March 2008 — October 2010*

Architected, designed, and implemented Test Automation Infrastructure for Application Virtualization (App-V), focusing primarily on core virtualization components (such as Registry virtualization), by leveraging C# features, such as reflection, and WCF to easily create numerous effective, reliable, and easily-maintained automated tests to exercise the product during early development. Served as Scrum Master. Assessed and made recommendations concerning source-control and test-infrastructure systems under consideration for adoption.

Designed and implemented product-installation testability hooks (in C++) to enable fully-automated failure injection. Found numerous product bugs, including in late-cycle changes that would otherwise have delayed RTM dates, via code review.

James Craig Burley, Software Craftsperson
Compiler/toolchain R&D, IT Support, and Training

Sole Proprietor, August 1989 — March 2008; October 2010 — December 2011

Clients included DRH Internet, Inc., Dallas, TX (client support and software development); Reflexion, Woburn, MA (SMTP server enhancements); Cadence Design Systems, Chelmsford, MA (porting code-generation modules of NC-Verilog from native HP-PA RISC to SPARCv8); Archetype, Waltham, MA (page-layout software development and API documentation); PictureTel, Peabody, MA (wrote high-level assembler for custom video processor); Lehigh University (added Interval Arithmetic support to g77, funded by Sun Microsystems).

Other

- Various and sundry projects and contributions on GitHub

- Software Architecture Fundamentals training (May 2018)

- Approximately 10 years' experience with optimizing-compiler design and internals, run-time libraries, debuggers

- Over 5 years' experience with operating-system internals (kernel and filesystem)

- Over 5 years' experience as a technical writer, mostly in Lead and Manager roles

- Created companion videonotes on concurrency for "Operating Systems: Internals and Design Principles, 8th Edition" by William Stallings

- Nominated for the Free Software Award (1998 and 1999)