# Resume of James Craig Burley

**97 Arrowhead Circle, Ashland, MA 01721; james-resume@jcb-sc.com**

## OBJECTIVE

Architect/developer role involving research, design, implementation, and shipping of software and networking systems, including operating systems, developer/tester toolchains, communications protocols, and group-ware.

## SUMMARY

- Extensive experience in software product architecture, design, implementation, testing, documentation, and maintenance
- Successful track record in project leadership (including Scrum) and management
- Demonstrable willingness to roll up sleeves and get the job done, for clients as well as team members
- Proven ability to deliver solutions under stressful circumstances without necessarily sacrificing quality or other long-term benefits
- Skilled at identifying or recognizing opportunities for improvements in process flow (including individual and team use of productivity tools) and at analyzing, presenting, and leveraging such opportunities as appropriate

## EXPERTISE

### Programming languages

- C, C#, C++, Java, Lisp, Verilog, VHDL, Fortran, PL/I, and assembler; experience designing languages
- Shell programming and scripting (such as GNU Bash, PowerShell, Perl, GNU Emacs Lisp)
- Acquainted with Python as well as with functional languages (exemplified by ML and Haskell) and logic-based languages (such as Prolog)

### Systems

- Unix (especially Linux) and Windows, including OS internals
- CPU instruction sets (CISC, RISC, and VLIW), memory and cache designs, I/O interfaces
- Windows Communication Foundation (WCF), TCP/IP, and networking protocols (such as SMTP)
- Software-development technologies (such as Git, Perforce, Visual Studio, MSBuild, GNU Make, and InstallShield)
- QA/Test technologies (such as Visual Studio Team Test (VSTT) and Driver Verifier)
- Documentation tools
- Network and system administration and maintenance (such as Linux, Windows, and Hyper-V)

# EMPLOYMENT HIGHLIGHTS

**James Craig Burley, Software Craftsperson** — http://www.jcb-sc.com/
Compiler/toolchain R&D, IT Support, and Technical Writing
*Sole Proprietor, August 1989 — March 2008; October 2010 — present*

Currently focusing on architecting and prototyping components of a network and computing infrastructure that scales up to 10 Billion people. This has included researching languages, protocols, user-interface issues, and other technologies. My initial areas of focus include developing a deeply-embeddable shell, including programming and communications environment, and leveraging these to create Unix, Web, and mobile apps that provide highly flexible, low-latency, and potentially ad-hoc, collaboration among peers.

Clients have 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 documentation); PictureTel, Peabody, MA (wrote high-level assembler for custom video processor); Lehigh University (added Interval Arithmetic support to g77, funded by Sun Microsystems); and (*circa* 1985) O'Reilly Media, Inc.

Also:

- Authored GNU Fortran 77 (g77), a widely-used ANSI FORTRAN 77 compiler, including the accompanying manual. Leadership responsibilities included cultivating and directing software developers, test-case authors, and end users.
- Contributed substantially to GNU Compiler collection (GCC) development and direction. This included participating in the EGCS "fork" from, and subsequent rejoining with, GCC.
- Authored and published several widely-deployed patches to qmail, a free, popular Mail Transfer Agent.
- Served as an advocate-at-large regarding a variety of issues revolving around email and network connectivity; computer language, toolchain and operating-system design; and the emergence of free software, also called Open Source Software (OSS).
- Began researching and designing a low-latency, high-reliability, Spam-resistant email protocol intended for use initially by IT professionals, which has evolved in conception to my current project.

**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. Conceived and implemented a way to leverage this infrastructure to achieve deep reliability testing, without having to run batteries of specialty tests over very long periods of time, using existing automation in combination with Driver Verifier and its fault-injection mode. Wrote PowerShell and other code to assist porting of numerous team-wide test cases to a new infrastructure. Developed approaches and plans for testing the product. 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 the product (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.

### Numerix Corporation, Newton, MA

*Senior Software Engineer, February 1988 — August 1989*

Designed and implemented code-transformation software to translate Fortran code for "outboard" mini-supercomputers with no built-in Fortran I/O or character-manipulation facilities. Helped architect and design run-time and debugging environment for this system.

Worked on proprietary Fortran Optimizing Compiler, Fortran Debugging Compiler, and related software. Became corporate expert on interface to third-party Fortran Front End. Maintained VLIW instruction-scheduling subsystem within Optimizing Compiler, focusing on straight-line optimizations.

*Manager of Documentation, March 1985 — February 1988*

Designed line of documents for Numerix NMX-432 (aka MARS 432) and other "mini-supercomputer" processors and related software, including VLIW opcode descriptions and assembler manual; wrote many of the documents.

### Prime Computer, Inc., Framingham, MA

*Senior & Lead Technical Writer, September 1982 — March 1985*

Designed, wrote, reviewed, edited, and led projects in Technical Publications department of super-minicomputer vendor. Major effort was "Advanced Programmer's Guide," Volumes 0 through III. Cultivated, hired, and supervised technical writers.

*Software & Senior Software Engineer, February 1978 — February 1982*

Designed, implemented, and maintained: internal components of PRIMOS operating system; run-time library to manage queuing of email, file transfer requests, batch jobs, and so on, using semaphores and shared memory as components of multi-user data base; and BATCH (job-submission subsystem). Also proposed key networking-related improvements for SPOOL (printing subsystem) and enhanced and maintained the file-utility program FUTIL.

# OTHER

Other projects included porting GCC/g77 to cross-development platforms; researching potential for optimizing circuit simulation using memory prefetching and related techniques in compiler-generated code; writing productivity-enhancing tools such as disassemblers for VLIW machines, IEEE FP conversion routines, and unused-pin-locating routines for net-lists; documenting VLIW ROM code; designing a 3270-terminal-based user interface to SAS, a statistical software package, running on IBM 370 MVS; writing custom software to format, print, and index technical documentation; and teaching the basics of low-level computer programming to adults in a technical-writing curriculum.

Work-related nominations and awards include: nominations for the Free Software Award (1998 and 1999); the Award of Distinction from the Society for Technical Communications (STC), New England Chapter (*circa* 1985); and Prime Excellence Awards from Prime Computer (*circa* 1980).