

Brian R. Gaeke ----- brg@dgate.org

## Summary

- \* Systems software engineer with 20+ years of experience specializing in compilers, debuggers, and safety-critical embedded software. Core developer of the Green Hills MULTI Debugger and IDE for 16 years. Led NVIDIA's Ada/SPARK formal verification program for autonomous vehicle software, achieving ISO 26262-certified toolchain deployment across all execution environments. Early contributor to the LLVM compiler infrastructure. Experienced in build engineering, CI systems, and cross-functional technical leadership.

## Education

- \* Master of Computer Science, University of Illinois, Urbana-Champaign (UIUC). Completed Dec. 2004. GPA 3.935. Passed oral qualifying exam for the Ph.D. degree, Feb. 29, 2004.
- \* Bachelor of Arts in Computer Science and Linguistics, University of California, Berkeley (UCB), with high honors in computer science and high distinction in general scholarship. Completed May 25, 2002. GPA 3.846.

## Academic Honors

- \* Phi Beta Kappa, 2001. Upsilon Pi Epsilon, 1999. Honorable Mention, Computing Research Association Outstanding Undergraduate Award, 2002.

## Highlights

- \* I have been developing systems software in C and C++ since the early 1990s.
- \* I have 16+ years of experience in debugger, simulator, and IDE development.
- \* I have 4+ years of experience in formal verification and safety-critical software development using Ada/SPARK, including ISO 26262 certification.
- \* I have 3+ years of experience in compiler development, including early work on the LLVM compiler infrastructure.
- \* I have taught computer science at the undergraduate, graduate, and high-school levels.

## Recent Software Development Experience

- \* Sep. 2021-present: Embedded Virtualization Development Tools Architect, NVIDIA, Santa Clara, CA. Served as company-wide technical champion for Ada/SPARK adoption across DRIVE OS (NVIDIA's safety-critical automotive software platform), reporting directly to a Senior Director. Led a cross-functional program spanning formal verification, toolchain engineering, ISO 26262 certification, process development, and training, in close collaboration with toolchain vendor AdaCore and multiple internal platform teams. More recently transitioned to automotive build and release engineering.
  - \* Drove Ada/SPARK adoption from early prototype to shipping in ISO 26262-certified products in DriveOS 6.0 and 6.5; built the first Ada/SPARK library committed to DRIVE OS and delivered the first Ada/SPARK code to ship in the real-time hypervisor.

- \* Negotiated and executed AdaCore toolchain certification contracts to qualify compilers, runtimes, and static analysis tools for ISO 26262; upgraded toolchains to certification-candidate releases and integrated qualified runtime libraries across all DRIVE OS execution environments (QNX, Linux, real-time hypervisor).
  - \* Designed and implemented build system support for mixed C/C++/Ada/SPARK codebases -- compiler options, warning policies, static analysis rules, proof configuration, prover safeguards -- so teams received process-conformant Ada setups by default; integrated CI static analysis and formal prover to run automatically for safety configurations.
  - \* Built a prototype Ada source analysis tool using libadalang -- NVIDIA's first custom libadalang tool for DRIVE OS -- capable of generating interface control documentation directly from Ada specification files.
  - \* Resolved deep technical problems across platforms: loop-invariant-induced stack overflows, non-thread-safe runtime behavior, Ada elaboration and secondary-stack issues in hypervisor contexts, and signal-handling failures in upstream libraries traced to Ada runtime dependencies.
  - \* Led compliance and qualification process for AdaCore tools as part of DriveOS 6.0; reviewed all AdaCore safety manuals with element owners; obtained SBOMs for cybersecurity requirements; contributed to process tailoring and disclosure documentation.
  - \* Authored the SPARK process document, NVIDIA supplement, style guide, static analysis rules, developer FAQ, bindings guidance, interface design docs, and CI documentation.
  - \* Built NVIDIA's Ada/SPARK training curriculum: five LMS modules with labs and quizzes; collaborated with AdaCore on custom video training content.
  - \* Organized Formal Methods Week 2024 (182 attendees, 14 presenters from NVIDIA, Microsoft, AdaCore, and academia).
  - \* Negotiated AdaCore subscription renewals; initiated AI/ML joint working group with AdaCore.
  - \* Supported DriveOS 5.2 ISO 26262 safety verification: unit tests, code and integration inspections, structural coverage gap analysis; developed GDB-on-ARM-VM debugging process for processor-in-the-loop unit tests; fixed GDB Mode bugs in Debug Server and built a GDB traffic logger.
  - \* Drove gnatcov coverage builds and analysis CI builds from consistently failing to green (<2% failure rate); contributed architectural consultation and code review for TegraConfig, a feature-flag and configuration differentiation system for DriveOS 7.2+.
  - \* Built CI infrastructure analysis tools (job failure checker, Gantt chart generator, build time analyzer); planned and executed build capacity changes across 20+ build configurations; filed 60+ bugs; technical mentoring and code reviews for multiple engineers.
- \* Sep. 2017-Aug. 2021: Senior Software Architect, Green Hills Software, Santa Clara, CA. Continued as lead developer and maintainer of JIT simulators and core features of the MULTI Debugger and IDE. Owned nightly test infrastructure, test scripts, and supporting machines.
- \* (2019-20) Developed a fast trace compression algorithm for ARM debug targets.
  - \* (2016-19) Built a continuous test coverage system from scratch: database, query tool, job distributor, web interface, Git/

Subversion integration, and integration with internal test hardware; supported source line coverage, decision coverage, and MC/DC.

- \* (2015-17) Implemented ARMv7 and ARMv8 instruction set simulators and accompanying INTEGRITY RTOS BSPs.
  - \* (2019-20) Ported Windows build and test systems from Cygwin to native Windows.
  - \* (2018-20) Added dynamic target selection, reservation, and automatic reflashing/reconfiguration to test infrastructure.
  - \* (2017-18) Implemented support for debugging across power cycles and debug link state change events.
  - \* (2016-17) Focused bug fixing to enable 64-bit IDE and compiler shipments.
- \* Feb. 2005-Sep. 2017: Systems Software Engineer / Senior Software Engineer (promoted 2008), Green Hills Software, Santa Barbara, CA. Lead developer for JIT simulators for PowerPC and ARM targets. Responsible for maintenance and enhancement of core features of the MULTI Debugger and IDE. Developed, tested and documented Eclipse/CDT plugins for GHS compiler, debugger and simulator. Supported customers and other engineers by answering technical questions, diagnosing and fixing bugs, and providing patches. Responsible for testing of nightly debugger builds and maintenance of test scripts. Responsible for setup and maintenance of test machines. Responsible for CVS and Subversion version-control servers along with another engineer. Assisted recruiting department with college visits, interviews, career fairs, on-campus information sessions and testing. Mentored 2 engineering interns and 1 new full-time engineer.

#### Research and Teaching

- \* May 2003-Feb. 2005: Graduate Research Assistant, LLVM Project, Computer Science Dept., UIUC. Implemented a trace-based dynamic runtime optimizer and portions of the x86 static and JIT back-ends in the LLVM compiler infrastructure -- foundational work that predated LLVM's widespread adoption in industry.
- \* Oct. 2000-May 2002: Undergraduate Research Assistant, IRAM Project, EECS Dept., UCB (supported by Lawrence Berkeley National Laboratory). Performance simulation and benchmark tuning for a novel processor architecture.
- \* Teaching: Graduate TA, compilers course, UIUC (Spring 2003). Undergraduate Tutor, CS 61C (Machine Structures), UCB (Summer 1999). Co-Lecturer, Academic Talent Development Program, UCB (Summer 1998).

#### System Administration/IT Experience

- \* Aug. 1998 - Aug. 2002: Programmer/Analyst, Instructional & Electronics Support, EECS Department, UCB. Custom programming supporting EECS classes. Overhaul of course software for large undergraduate classes. System administration and user support for Unix and Windows NT servers & labs used by faculty and staff in the EECS department. Maintenance of vendor-supported and unsupported software, including CAD and software-development tools. Hardware upgrades and installations. Consultation with faculty/teaching staff to assess instructional needs.
- \* Fall 1997 - Fall 1998: Vice President and Coordinator of Unix Help Sessions, Computer Science Undergraduate Association, UCB. System administration for 2,500-user FreeBSD server and workstations running DEC ULTRIX, FreeBSD, and Solaris. Organized and taught regular

seminars on topics in computer science and system administration.  
Publicity, flyering, outreach.

- \* Summer 1995 - Spring 1997: Member of Help Desk student staff, Cincinnati Country Day School, Cincinnati, OH 45243. Maintenance of: on-campus Ethernet, Sun Solaris-based campus-wide E-mail and World-Wide Web server, and Mac OS-based campus-wide AppleShare server. Help-desk technical support for 600 Toshiba Pentium-based Windows 95 laptops and two labs of Mac OS machines.

#### Other Software Development Experience

- \* Spring - Summer 1997: Contract Programmer, MIS Department, Channel 48 WCET-TV, Cincinnati, OH 45214. Designed, installed, tested, and documented a low-cost custom firewall and backup facility using free software.
- \* Summer 1995: Contract programmer, University of Cincinnati College of Law Library, Cincinnati, OH 45221. Designed kiosk to allow law students to create their own web home pages.

#### Open-Source Development Projects

- \* May 2001 - 2018: Principal architect of VMIPS (<http://vmips.sourceforge.net>), an open-source MIPS R3000 whole-system simulator.
- \* Sep. 2000 - 2010: Wrote, tested, and released WebNews (<http://webnews.sourceforge.net>), an open-source Usenet News reader written in Perl. WebNews is used by the Berkeley EECS department to give off-campus students access to class discussion groups.
- \* May 1995 - March 1997: Volunteer support programmer, The NetBSD Foundation, Inc. Worked with the team porting the free 4.4 BSD Unix derivative, NetBSD, to the Motorola 68K-based Macintosh systems (NetBSD/mac68k): Maintained and enhanced the bootloader. Wrote first edition of bootloader manual. Implemented support for Farallon ethernet cards and various other kernel patches.

#### Languages and Other Skills

- \* English native. Some reading and conversational ability in French.
- \* Programming languages: C, C++, Ada/SPARK, Python, Perl, Java, Lisp/Scheme, MIPS/ARM/x86 assembly, Unix shell scripting, HTML, TeX.
- \* Formal methods and safety: SPARK formal verification (GNATprove), static analysis (CodePeer, GNATcheck), ISO 26262 automotive functional safety processes, safety manual review, tool qualification/certification.
- \* Build systems: make, gprbuild (AdaCore), gbuild (Green Hills), TMake/nvmake (NVIDIA); CI pipeline development, debugging, and capacity planning; coverage measurement (gnatcoverage).
- \* Operating systems and platforms: Linux, QNX (safety-certified embedded RTOS), embedded real-time hypervisors, Windows, macOS, FreeBSD; NVIDIA DRIVE OS, INTEGRITY RTOS.
- \* Tools: Green Hills MULTI IDE (core developer, 16+ years), GDB, libadalang (Ada source analysis), Git, Subversion, Eclipse/CDT, GNAT Studio.

References available upon request.