

Scott M Anderson

smanders@outlook.com · 435/512-4070

<https://www.linkedin.com/in/andersonscottm/>

SUMMARY

Principal Software Engineer with a track record of architecting scalable, cross-platform systems and transforming development ecosystems. Proven leader in modern build systems, CI/CD, and software supply chain security, including SBOMs and automated compliance. Drove 10x gains in developer productivity and product throughput by introducing key technologies, restructuring large-scale codebases, and standardizing development environments. Deep expertise in C++, CMake, containerization, and high-performance systems, with experience spanning embedded, UI, and multimedia domains.

EXPERIENCE

Principal Software Engineer Priority 6 – Current

- Architected a reusable CMake build platform with CI pipelines, leveraging Git, GitHub, and Docker (see <https://github.com/externpro/externpro>)
- Developed extensive GitHub Actions for automated build, testing, and security compliance
- Created world-class CMake build platform supporting multiple processor architectures, OSes, and compiler releases
- Implemented software supply chain security with SBOMs and attestation for production deployment

Software Engineer Space Dynamics Laboratory – 20+ years

- Transformed development ecosystem by pioneering 10 key technologies, enabling 10x productivity growth and scaling from 3 annual to 50+ product releases
- Reorganized 100+ static libraries, eliminating circular dependencies and enabling 10x product growth and unlocking 5x developer efficiency; the Architect of C4ISR division's code library structure
- Introduced Docker containerization, standardizing development environments across Windows/Linux for 50+ engineers, reducing onboarding time from 3 days to 1 hour
- Implemented Git/GitHub Enterprise, preserving 15+ years of SVN history and enabling modern continuous integration and deployment (CI/CD) that cut deployment time by 80%
- Spearheaded cross-platform agile development, guiding teams on Windows, Linux, and Solaris to meet release deadlines
- Delivered end-to-end solutions spanning UI design, code development, and hardware integration
- Expanded image compression capabilities with JPEG 12-bit, JPEG 2000, and JPEG XR integration and developed applications that showcase the capabilities offered

Software Engineer Sorenson Media, Inc. – 5+ years

- Designed & implemented an extensible video codec framework to run under several multimedia architectures (QuickTime, Video For Windows, DirectShow, MPEG-4)
- Responsible for architecture, implementation, and optimization of core encoding algorithms which differentiate compression performance
- Led an engineering team in implementing the Sorenson Video 3 codec based on H.26L specification
- Codec implementations: H.263, H.261, MPEG-4 Visual (Simple and Advanced Simple Profiles), H.264 (aka JVT, MPEG-4 Part 10 AVC)
- Extensive optimizations: MMX, SSE (Streaming SIMD Extensions), SSE2, AltiVec - outperformed Intel's IPP library

PREVIOUS RELEVANT EXPERIENCE

Graduate Research Assistant – Utah State University ECE Dept

System Administrator – Sorenson Vision, Inc.

Science Engineering Research Semester (SERS) Intern – Oak Ridge National Laboratory

Software Tester – Space Dynamics Laboratory

EDUCATION

UTAH STATE UNIVERSITY

Master of Engineering in Electrical Engineering

- Coursework: Signal Processing, DSP Real-Time Processors, Networking, and Communications
- Research: Video Compression (Motion Estimation / Vector Quantization)

Bachelor of Science in Electrical Engineering

- Minor in Computer Science Magna Cum Laude (3.81 GPA)
- National Merit Scholar and Superior Student Scholar, University Honor Roll & Dean's List
- Outstanding Pre-professional Student in Electrical Engineering Department

TECHNICAL SKILLS/CERTIFICATIONS

Build Systems & DevOps

- CMake: Advanced Modern CMake, ExternalProject, CPack, cross-platform builds, custom functions/macros
- CI/CD: GitHub Actions, automated testing, multi-platform builds
- Containerization: Docker, Docker Compose, multi-architecture builds, GitHub Container Registry
- Infrastructure as code: Dockerfile optimization, container orchestration
- Package Management: SBOM generation, software supply chain security, dependency management
- Build Automation: Multi-compiler support (GCC, Clang, MSVC), dependency management
- Cloud Platforms: GitHub hosting, container registries

Version Control & Collaboration

- Git: GitFlow, submodules, large-scale repository migration, release workflows
- GitHub: Enterprise administration, CI/CD pipelines, Actions, Container Registry, issue management
- SVN: Legacy system migration, history preservation

Programming Languages

- C/C++: 25+ years, cross-platform development, performance optimization
- Assembly: Intel assembly (MMX, SSE, SSE2, AltiVec optimizations)
- Shell Scripting: Bash, Python, build automation, DevOps workflows

Cross-Platform Development

- GUI Frameworks: wxWidgets, Motif migration
- Operating Systems: Windows, Linux (multiple distributions), Solaris, macOS
- Architectures: x86/x64, ARM, cross-compilation

Multimedia & Image Processing

- Codecs: H.264, H.263, MPEG-4, JPEG 2000, JPEG XR, Sorenson Video, custom implementations
- Optimization: SIMD instructions, DCT algorithms, performance profiling
- Compression: Lossless/lossy compression, PSNR analysis, algorithm comparison

Development Tools & Security

- AI-Assisted Development: Windsurf Cascade, accelerated development workflows
- Testing: Cross-platform automated testing, regression detection
- Security: Static analysis, security patches, vulnerability management, SBOMs
- Documentation: Technical writing, API documentation, process documentation

Security & Certifications

- Security Clearance: Secret (current), Top Secret/SCI (previous)
- CompTIA Security+