

Sean Werkema

607-206-6326 * seanwerkema@gmail.com * Downingtown, PA

Goal

I want to design, build, and maintain meaningful, high-quality commercial software with a team of talented, dedicated, and smart colleagues. I've coded millions of lines in dozens of languages over the last couple decades, and I'm now looking for a senior-level engineer, architect, or lead role where my coding skills and experience can make a positive difference in the world.

Technologies & Skills

- **Languages:**

Expert: C# (3.5 to 7), JavaScript (classic to ES7), C (K&R to C99), C++ ('90s to C++1x), HTML (0.9 to 5+), CSS (1.0 to 3+), assembly (x86), regexes (.NET, JS, PCRE, Oni), MS SQL.

Also some: Python, PHP, Perl, Ruby, VB.NET, Lisp, Scheme, Yacc/Bison, Lex/Flex, Pascal, BASIC, assembly (x64, ARM, SPARC, 68K, 6502), shell script, Makefiles, MySQL, PostgreSQL. And some Java.

Research: Learning Rust and Go. Creating Smile.

- **Frameworks & Libraries:**

JavaScript: ReactJS (+ecosystem), NPM and Babel, jQuery (+plugins), “no framework” JS and the DOM, various ReactJS widget libraries, many other NPM packages.

.NET: .NET Framework 3.5-4.7, .NET Core 1-2, ASP.NET WebForms & MVC, ADO.NET, WinForms, P/Invoke, JSON.NET, Dapper, NUnit, XUnit, Polly, StackExchange Redis.

Others: Zlib, LibPNG, Oniguruma, Boehm GC, Intel Binary Decimal library, GD2, IJG JPEG library, GMP, OpenGL, SDL, GLEW, GLM, STB, DirectX, MFC, COM, and a lot more.

- **Platforms/OSes:** Linux (since v1.1), Windows (3.x to 10), Browser DOM (1.0 to IE to evergreen & CanIUse), WSL, Cygwin.
- **Development Environments:** Visual Studio (VC98 to VS19, w/ and w/o ReSharper), GCC (1.0 to 5.4) + Make, Vim + Unix terminal, Git, Subversion, VS Code.
- **Data Stores:** SQL, Redis, AWS S3, Couchbase
- **Algorithms & Concepts:** Trees and graphs, compression, optimization, encryption, security, design patterns, fuzzy pattern matching, search engines, Unicode, compilers, lexical analyzers, parsers, memory allocators, garbage collectors, image processing, 3-D graphics, and a lot more.

Employment History

2009-2019: HomeNet Automotive (Cox Automotive) (www.homenetauto.com)

- *Principal Software Engineer (2015-2019):* Began decomposing HomeNet's flagship product *IOL*, a monolithic backend processing system and ASP.NET WebForms website, by lifting, refactoring, or rewriting its major pieces to become cloud microservices. Designed, documented, implemented, and instructed in major refactors, enhancements, and architectural changes. Trained team on new techniques and new tools with custom coursework. Replaced core components with new custom cloud services without failures or downtime. Began replacing UI with new custom ReactJS-

based framework. Implemented a new Redis-based data store for IOL's most critical processing load.

- *Team Lead (2015)*: Led a specialized small development team to develop a new product, *AIS*. Designed the UI and backend architecture for it. Mentored team members, reviewed work, and helped out with the hardest coding under tight deadlines. *AIS* was released to great acclaim in early 2016.
- *Senior Developer (2012-2014)*: Optimized backend and frontend performance of IOL. Led team in refactors and enhancements. Designed and documented major architectural changes. Trained team on new techniques with custom coursework. Built dozens of new products.
- *Frontend Developer and Frontend Lead Developer (2009-2012)*: Designed architecture of and implemented frontend UI for IOL. Created UI standards for the company. Enhanced ASP.NET with many custom tools and components to improve productivity of frontend team.

2006-2009: Founder & President, Sixty-Five Software

Designed, implemented, debugged, and supported its flagship product, SpaceMonger, a data-analysis-and-visualization program. Wrote 250,000 lines of C++. Supported customers via e-mail and a forum, resolved bug reports, distributed several upgrades. Licensed the rights to the technology to StarDock.

1996-2008: Freelance website designer and programmer.

Created and maintained custom Web sites under contract, using PHP and CGI.

1993-1997: IT support, Arkema (www.arkemagroup.com)

Installed, configured, repaired, and trained employees on computers and software for the North American headquarters of Arkema, a large chemical corporation.

Independent Research

I have spent my spare time over the last 20 years designing and implementing *Smile*, a new programming language based on a dynamic syntax parser set atop a custom object-oriented-Lisp 1 core. *Smile* is an attempt to remove *all* boilerplate from programming by reducing software implementations to their purest form: whiteboard pseudocode. *Smile* draws its heaviest inspiration from Scheme, Ruby, and JavaScript; and from Knuth's *Literate Programming* (1992); but it has a look-and-feel unlike any of them, and it includes many unique innovations (such as dynamic proof of operators, explicit type lists, punctuation minimization, and dynamic syntax). *Smile* is not yet ready for public consumption, but the current C implementation may be found in my GitHub account.

Honors

- Knuth Check recipient in 2001 for an error in Volume I of *The Art of Computer Programming*.
- "Jedi Master" innovation awards in 2010 and 2014 from HomeNet Automotive. HomeNet hackathon winner in 2015.

Education

Pennsylvania State University, State College, PA
Bachelor of Science in Computer Science