



Overview

Seeking a position with a personally unfamiliar technology to grow as a developer. Knowledgeable about data-driven design for rapid iteration, working in small teams, and being a highly independent abstract-problem solver. I enjoy making great tools that developers enjoy using.

Languages

C, C++, C#
SQL, U-SQL
Lua, JavaScript
Bash

Developer Tools

CMake, QMake
Visual Studio, Qt Creator
CLion, Valgrind
Git, Mercurial, SVN

Mathematics

Linear Algebra
Statistics, Probability
Calculus, Physics
Algorithms

Domain-Specific

Azure, Web Services
GTest, Cont. Integration
OpenGL, Compilers
Linux systems, ALSA

Work Experience

Microsoft - Software Engineer

Jun '15 - (Current)

- Working with Azure technologies to enable internal services (Cloud Storage, REST APIs, Services).
- Deployed and hotfixed preproduction and production environments for internal packaging systems.
- Designed and implemented an API for tracking and reporting packaging payloads (C#, SQL Server).
- Delegated and gave feedback on multiple features based on payload API (design, review, feedback).
- Crunched large amounts of data for processing and visualization (SQL, U-SQL, KendoUI).
- Built extensions when existing tools lacked functionality (U-SQL extensions, C# applications).
- Designed, architected, and maintained several general-purpose, high-impact data streams.
- Made servicing decisions based on the results of the data sets (filed bugs, monitored releases).

Open Source

OpenSK - Cross-Platform Streaming Toolkit

May '16 - (Current)

- Creating utilities using the API as it is being developed to understand impact of API design decisions.
- Became proficient with Advanced Linux Sound Architecture (ALSA), specifically PCM streams.
- Iterating on API specifications as more intricacies of the abstraction become apparent.

Karma - Modern OpenGL Framework

Jan '15 - Apr '15

- A configurable multi-pass deferred renderer with class encapsulations of the OpenGL API.
- Physically Based Rendering, Image Based Lighting, Filmic Tonemapping (via. Naughty Dog).
- Per-Fragment Motion Blur, Area Lights, SSAO, Soft Shadows, swappable BRDF subroutines.
- Published technical documents outlining implementation (<http://www.trentreed.net/topics/opengl/>).

School Projects

Ping - 3D Perception-Based Horror

Jul '13 - Jul '14

- Wrote a reflection system that allowed deeper type-introspection of C++ classes.
- Leveraged reflection system for easy Lua interfacing (types, function, methods, C++/Lua binding).
- Created a cross-platform, type-safe, hot-swappable resource management system.

School

DigiPen Institute of Technology - Redmond, WA

Aug '11-May '15

Bachelors of Science in Computer Science in Real-Time Interactive Simulation

- Magna Cum Laude, Student of the Year (2014), Dean's List (several)