Jonathan Christen

Seasoned programmer with 15+ years professional experience. Spent career across all parts of the software industry ranging from UI development to VR to cloud solutions, with a special focus on client-side video game development. Founded Entropy Software LLC as a solo venture in 2020 to build middleware for game developers.

Seattle, WA

linkedin.com/in/jon-christen

WORK EXPERIENCE

Entropy Software LLC

Owner / Developer — March 2020 – Present

- Middleware development for games.
- Plugins sold on Unity and Unreal marketplaces as well as directly to developers.
- Sole owner and developer.
- Working with independent game studios.
- Contracting with external studios fitting into any needed role.

Linden Lab

Senior Software Engineer — March 2017 - February 2020

- Sansar (Metaverse Game)
- Avatar team lead.
- Developed technology improvements for avatars.
- Implemented a neural network-based IK system for VR.
- Architected asynchronous UI task system for avatar editor.

Microsoft Corporation

Senior Software Engineer — November 2015 - March 2017

- Windows Shell
- UI developer for the Windows Ink Workspace.
- Experience with xaml based applications.
- Coordinated cross team development on unannounced project.

Microsoft Corporation

Senior Software Engineer — September 2013 - November 2015

- HoloLens
- One of the lead developers on several demos that showcase the device’s potential.
- Demos have been shown on stage publicly and behind closed doors to countless companies. They have been praised by industry analysis and from executives.
- Optimized our experiences and the Unity 3D engine to run on effectively mobile hardware.

Microsoft Corporation

Software Engineer II — May 2010 - September 2013

- Windows 8 UX
- Architected and developed the UI for the metro version of Internet Explorer.
- Made use of DirectComposition (new to Win8) for hardware accelerated animations.
- Optimized CPU & GPU code paths for ARM devices using XPerf analysis.
- Collaborated with multiple teams to develop a unified experience across Windows.

Microsoft Corporation

Software Engineer — June 2008 - May 2010

- Web Filtering / Tunneling Service
- Cloud service offering secure machine connectivity without the use of additional hardware.
- Designed and built Silverlight admin web console.

Heavy-Iron Studios

Software Engineer Intern — May 2006 - August 2007

- Ratatouille (X360, PS3) and Wall-E (X360, PS3)
- Created various shaders, including Eve's shader in Wall-E and depth of field for cinematics.
- Worked on various effects, including lens flare and Wall-E tread tracks.
- Developed a generic particle system that allowed artists to have great control over particle effects.
- Worked directly with artists, animators, and designers to bring ideas to life.

EDUCATION

University of Southern California

Bachelor of Science in Computer Science and Computer Engineering - 3.85 GPA — 2008

- Focus on computer science and game development.

PROJECTS

DinoFracture — 2020 - Present

- Mesh slicing and Voronoi shattering library for Unity.
- Pure .Net with a native C++ library in the works.
- High performance utilizing novel fracturing techniques.

SmartEngine — 2020 - Present

- Cross platform AI solution for games.
- Graph based neural networks.
- High performance compiled graphs described in human editable Json.
- Graph runs directly on user's hardware either on CPU or GPU via Vulkan.
- Out-of-box support for multiple training methodologies including genetic algorithms and traditional gradient descent.

C++ Reflection Library 2020 - Present

- Supports both compile time and runtime reflection.
- Requires no external tooling. Pure C++17 without any code-gen or post build processing.
- Compile time reflection via type-safe templated functors.
- Runtime has full class type-info and object activation capabilities.
- Dynamic function objects allow for arbitrary binding of parameters with guaranteed runtime safety. Equivalent of dynamic keyword in C#.
- User-defined metadata can be attached to classes with compile time and runtime lookup.
- Requires single macro per reflected class and single macro per reflected field / method.
- Use cases include serialization, UI exposed classes, task graph input / output connections.

SKILLS

- C++
- C#
- Unity Engine
- Unreal Engine
- Optimization
- Software Architecture
- Game Development
- Linear Algebra
- DirectX
- OpenGL
- Vulkan
- VR
- Machine Learning
- TensorFlow
- UI
- Xaml
- Blender