

Eric Spevacek

Senior Technical Artist

Eric Spevacek

1310 E Spring St
Seattle WA 98122 USA

1.651.216.5821
ericSpevacek@gmail.com
April 2020

Skills & Proficiencies

Experienced technical artist specialized in content workflows and engine systems. Diverse history of game engine expertise including, Unity, Unreal Engine 4, and proprietary AAA engines. Comfortable jumping into large projects and tackling problems with a multidisciplinary focus.

Languages

Python, C#, C++, Maxscript

Creative Software

Unreal, Maya, Unity, 3ds Max, Substance Suite, Adobe Suite, Houdini

Productive Software

Visual Studio/Code, PyCharm, Perforce, Subversion, Jira

Analytical Software

RAD Telemetry, PIX for Xbox/PC

Experience & Work

Undead Labs

Senior Technical Artist | JULY 2016 - PRESENT

SEATTLE, WA, USA

Systems - Unannounced Project

- Prefabs
 - Artist/designer UE4 composition tools for placing reusable content across open world maps
- World Interchange Files
 - UE4 extension to move ownership of core actor types to in-house scene format
 - Ensures data-consistency and optimization across large scale of content and optimal loading/streaming behavior
- Procedural Biome
 - Houdini-based procedural scatter tool based on "biome" rules to automate placement of natural elements (foliage, vegetation, rocks, cliffs) across open world map
 - C++ custom bridge between UE4 and Houdini to roundtrip data transfer

Systems - Studio/Shared

- Material Archetypes
 - Source-to-UE4 dialog-free exporter and packing process that automates texture + material creation and setup
 - Enables batch re-processing to source art to re-pack textures to new or changed archetypes

- Launchpad
 - Persistent PySide2 GUI tray application responsible for P4 workspace management, creative software and in-house tool distribution, dev environment customization
- Content Checker
 - UE4 Editor Tool for validating asset data in content browser
 - Replaces built-in “Map Check” tool; provides a superior UI/UX to make identifying and fixing errors more actionable
- Automated Performance Testing
 - Nightly performance runs of the latest build to measure CPU/GPU against target benchmarks
 - Nightly data snapshots to provide analysis in offline tools
 - Reports easily understood email digests and heatmaps for production and art staff to diagnose and triage issues in the build
- DCC Environments & Bootstrapping
 - Standalone Python tools for wrangling art content from a diverse set of source packages (Quixel Mixer + Bridge, Substance Painter + Designer, Adobe Photoshop, Maya, ZBrush)
 - Standardization of texture channel packing, material/shader archetypes

Systems - State of Decay 2

- VFX Component Pool
 - C++ run-time manager; fixed-size component pool that provides “fast path” to recycled objects for high-frequency VFX systems (gunfire, decal responses, gibs, footsteps) to save on CPU cycles and to throttle overloads of requests
- Interior Streaming
 - C++ run-time streaming logic for managing the state of the closest set of interior spaces to players to reduce global memory and CPU overhead
 - Exposes art-tunable “blackout” meshes to help mitigate streaming pop-in/out
- Vehicle Destruction
 - Art-driven blend-shape based pipeline for run-time vehicle destruction
 - Python tools for baking texture atlases of morph target offsets to drive material blends
- Enlighten Middleware Integration & Support
 - Optimized and maintained the UE4 integration of a 3rd-Party Real-Time Global Illumination solution
 - Created custom UE4 tools for lighting artists to wrangle complex and performance-sensitive content setup requirements

Other Responsibilities

- Runtime Performance
 - Open world streaming support
 - CPU/GPU optimization guidance
- Standards & Best Practices Documentation
 - Help raise the quality bar of art production to meet current standards and trends in art production
 - Improve code quality and habits across engineering by providing examples, templates, and patterns to work within the UE4 codebase

- Tools CI (Build, Test, Distribute)
 - Maintain a set of Jenkins build processes to automate tools distribution
- UE4 Editor Extensions
 - Maintain a set of frameworks to provide “paths of least resistance” to extending editor tools
- Gameplay Sketches
 - Independently prototype demo-able systems (through UE4 Blueprint/C++) to design/art staff with the goal of reducing risk and gaining confidence in implementation details

Blizzard Entertainment

Senior Technical Artist I | FEB 2016 - JULY 2016

Technical Artist | JULY 2014 - FEB 2016

IRVINE, CA, USA

Systems

- Python Package Manager
 - Simpler, in-house alternative to “pip” that enabled a team of 10 tech-artists to rapidly contribute to, version, and share a library of code across production branches
 - Built streamlined process for committing and deploying changes; acted as a force multiplier for the entire tech art team
- Diff Doctor
 - PyQt GUI-based standalone build delta tool to guide the internal QA process of finding string leaks of unannounced content and other unintended changes to the live game
- Launcher
 - Persistent PyQt GUI tray application for syncing builds, launching tools, and managing local state of development branches in rapid live-service
 - Queries back-end web services for build metadata
- Art Source Navigator
 - PyQt GUI side-kick tool that provided fast partial and exact look-up of 3ds Max files in the source art directory tree

Other Responsibilities

- Internal Tools Visionholder
 - Modernize art and data pipeline tools to remove inefficiencies and bottlenecks across discipline boundaries (e.g. allow an artist to test their new mount in the game without needing to put in a request for technical design to enter new data fields)
 - Guide development decisions to meet the demands of “games as a service” requiring ongoing, frequent content delivery (6 week release cycle)
 - Provide UI/UX oversight of new tools development and encourage best practices to guide early design iterations

- Runtime Performance Steward
 - Guide environment and character art staff to improve performance of content
- Art Debugging Mentor
 - Teach and guide artists to use editor tools to diagnose problems in art process (bakes, normal errors, UV issues)
- Warcraft III Asset Port
 - Developed a set of scripts to port forward legacy art assets to the latest tools to be released as a free art pack to the Starcraft II modding community

InContext Solutions / Technical Artist

FEB 2013 - JUNE 2014, CHICAGO, IL, USA

Systems

- Procedural Art Tools
 - Maya Python toolkit to help expedite the modeling and UVing process of common art content (aisles, end caps, walls)
- Source Project Browser
 - In-Maya Python tool to navigate and discover source art content via Perforce
- Maya to Unity Scene Exchange Format
 - One-click scene exporter that preserves asset references as “GameObject” packages in Unity

Other Responsibilities

- Rapid Prototyping
 - Per-client based prototyping and code support for custom “gameplay” requests

Shipped Titles

State of Decay 2 + State of Decay2: Juggernaut Edition

MAY 2018 + MAR 2020 - UNDEAD LABS - SENIOR TECHNICAL ARTIST

Heroes of the Storm

JUN 2015 - BLIZZARD ENTERTAINMENT - SENIOR TECHNICAL ARTIST I

Starcraft II: Legacy of the Void

NOV 2015 - BLIZZARD ENTERTAINMENT - TECHNICAL ARTIST

Tessallation

MAY 2013 - INDIE - GAME DEVELOPER

Involuntary Runner

DEC 2012 - INDIE - GAME DEVELOPER

Diablo III

MAY 2012 - BLIZZARD ENTERTAINMENT - TECHNICAL ART INTERN

Education

DePaul University / BS Computer Games Development

GRADUATED JUNE 2013, CHICAGO, IL, USA