Russell Klenk
contact@russellklenk.com

Professional Experience

Scientific Games – Contract Software Engineer, October 2012 to Present
Las Vegas, NV

Performed contract game and tools development as well as implementing new system-level features such as jurisdictional win limiting. Technologies: C++, C#, custom tools.

Leap Forward Gaming – Contract Software Engineer, June 2012 to Present
Las Vegas, NV

Implemented a mystery jackpot bonus and multi-level progressive controller in JavaScript along with a gaming RNG and associated regulatory verification and test tools in C/C++. Provided customer support and on-site hardware installation when needed. Technologies: Linux, Node.js, C#, PostgreSQL, Apache Apollo.

International Game Technology – Staff Software Engineer, May 2007 to June 2012
6355 S. Buffalo Drive, Las Vegas, NV

Responsibilities include prototyping game and systems concepts, and researching methods for improving the existing slot game platform and game development process. Later promoted to a team lead role, where I managed a small team of engineers and content developers working on a web-based games platform for casino-style, casual and skill-based games.

Accomplishments:

- Delivered a number of presentations outlining methods for improve API design, code reuse, content loading performance and taking advantage of 3D rendering techniques in 2D games to improve visual quality.
- Implemented a real-time visual profiling system to track down issues with Flash content performance on our Windows CE and XP Embedded platforms, including remote profiling capabilities for monitoring memory usage.
- Managed a small team to deliver a real-time multiplayer tournament solution built around an existing, deployed product in an extremely short timeframe without requiring changes to the existing product.
- Managed a small team to deliver a web-based multiplayer gaming platform and complete content pipeline allowing significantly reduced content iteration times and games supporting runtime content replacement.
- Given responsibility for a troubled, high-visibility project two years behind schedule, re-focused the team to stabilize and release the software within six months. Responsibilities included providing technical guidance to team members, working closely with the QA department to repair a severely damaged relationship and making significant technical enhancements.
- Solely responsible for the design and implementation of a high-performance messaging system designed to enable communication between separate processes on the gaming device and message-oriented-middleware running in the regional data center.
- Received several promotions and special performance bonuses.

Progressive Gaming – Senior Software Engineer, September 2004 to May 2007
920 Pilot Road, Las Vegas, NV

Responsibilities included maintaining, supporting and improving the existing legacy code base, implementing new game play technology, implementing game designs and supporting games through their entire lifecycle, ensuring they met applicable regulatory requirements. Worked closely with other engineers and artists to realize game concepts.

Accomplishments:

- Started implementing games for G2E 2004 (the casino game industry's largest gaming show) within one week of starting, with no prior experience in the casino game industry, no experience developing for real-time embedded systems, minimal training and no documentation, helping to deliver three show-ready games within one month.
- Worked in multiple areas of the code base, from low-level kernel to high-level game play code.
- Worked within a mid-size, aging C/C++ code base of approximately 900,000 lines of code written over 10 years.
- Designed and implemented Windows device drivers in C and pseudo-drivers in C# for various game cabinet devices, including RS-232 bill validators, thermal printers, NVRAM, the system watchdog timer and a custom I/O board.
- Designed and implemented a reliable messaging protocol in C# for interfacing game software to cabinet devices.
- Developed and fielded six full games over a one-year period. These games were deployed in casino properties nationally and internationally.
- Successfully designed, implemented, tested and shipped a new free spin game play mode, packing gaming history data for 100+ game cycles into severely limited non-volatile memory resources.

Virtual Charting – Software Engineer, May 2003 to August 2004
3844 Meadows Lane, Las Vegas, NV

Responsibilities included developing, documenting and supporting version 1.0 of a Tablet PC/.NET-based electronic medical records management software, and serving as the programming lead on the second version of the product (which was also deployed in the field). The company was dissolved at the end of August 2004 for lack of funding.

Accomplishments:

- Developed one of the first electronic medical records management software products for the Tablet PC platform.
- Implemented a TWAIN scanner driver to support high-speed image scanning for .NET, allowing the medical office to scan and capture their existing patient records electronically.
- Implemented a document compositing system utilizing the Microsoft Ink APIs to allow the medical office to continue to write on top of their existing forms, without having to worry about errors due to improper handwriting recognition. Documents could contain a mixture of fields intended for standard keyboard input, handwriting and static images.

Entropy Group – Software Engineer, 2000 to May 2003
2701 Westwood Drive Suite 201, Las Vegas, NV

Responsibilities included designing, developing, testing and documenting a custom real-time web 3D engine and supporting tools for the Windows platform. The company stopped providing software development services in the early part of 2003 to focus on their core strengths of marketing and web development.

Accomplishments:

- Designed and delivered a scene graph-based rendering system in C++, using Direct3D and OpenGL.
- Implemented custom plugins for 3ds max using the C++ API, the IGame C++ API and MaxScript for mesh, animation, material and scene processing and export.
- Implemented a custom material language with shader support for Direct3D 8-level hardware.
- Delivered a Pocket PC application for computing ballistics solutions for one of the company's marketing and web clients.

Education

University of Nevada, Las Vegas, 1999 through 2001

Targeting a B.S. in Computer Science while working full-time. Left to pursue startup opportunities.

Software, Programming Language and API Experience

- More than 10 years of professional experience using C and C++ for games, desktop applications and tools development.
- Six years of experience with the C# and .NET platforms for toolset, application and game development.
- Three years of experience with ActionScript 3 and Java for multiplayer game development.
- Current experience with JavaScript and HTML5 technologies including Canvas, WebGL and Node.js.
- Node.js add-on development and V8 JavaScript engine extension using C/C++.
- Exposure to and limited use of Objective-C, Lua, Python and Ruby.
- Embedded software development for QNX, vxWorks, Windows CE, XP and 7 Embedded platforms.
- Content exporter plugin development using C++ and MaxScript for 3ds max and C for Lightwave3d.
- IDEs and Text Editors: Visual Studio, XCode, Eclipse, NetBeans, FlashDevelop, TextMate, SublimeText 2.
- Build management: CMake, Apache Ant, MSBuild, gyp.
- Revision control: Perforce, Subversion, Mercurial, Git, Team Foundation Server, ClearCase, StarTeam.
- Graphics APIs: Windows GDI, Direct3D 8-11, XNA, OpenGL 2.0+, OpenGL ES, WebGL.
- Microsoft Ink APIs.
- Microsoft Office suite.

Significant Personal Projects and Sample Code

- Implemented the techniques described in the paper Resolution-Independent Curve Rendering Using Programmable Graphics Hardware, by Charles Loop and Jim Blinn using C# and Managed DirectX.
- Implemented an automated multi-platform game content build engine including dependency and reference tracking, minimal rebuild support, and content database management for processing of textures, meshes and other game content types: https://github.com/russellklenk/content.js
- C/C++ libraries for game and tools development: https://bitbucket.org/rklenk/ninjabird-common/
- WebGL.js wrapper and 2D quad batch rendering: https://github.com/ninjabirdstudios/webgl.js