William Bundy

Targeting a software engineering position—experienced with systems-level C/C++, SIMD, and GPU technologies. Will has been building games and tools on his own ever since he started learning the craft 11 years ago.

+1 (909) 380-2894 williambundy.github.io
Redlands, California william.tc.bundy@gmail.com

Graphics Projects

`Traceteroids` (C, Win32, GLSL)

2017-2018—Raytraced Asteroids in under 100k

- Implemented an analytic raytracer running on the graphics card (in a fragment shader).
- Implemented without any external libraries, including the C runtime or external math libraries; fits in under 64k when compressed.

`wiggle` (C++, FreeType, msdfgen)

2018—Font information tool, image atlaser

- Goal: create high-quality representations of fonts that are easy to integrate in games: exports glyphs, metrics, and kerning info
- Implemented a multi-channel signed distance field renderer and layout engine for the Latin alphabet.

Tools and Libraries

`wb_alloc.h` (C, Win32, Unix)

2017-18—Single-header custom allocator library

- Goal: dramatically simplify low-level memory management without adding myriad dependencies.
- Provides multiplatform abstractions for virtual memory, without relying on system headers.
- Implements several common allocator types: linear, stack, pool, and a tagged heap.

`VMFSketch` (C, SDL2)

2017—Rapid prototyping for 3D levels

- Goal: a tool for quickly laying out floor plans and exporting them to Source games or other VMF-based editors.
- Inspired by vim—designed a modal interaction model for keyboard + mouse usage.
- Novel approach to brush structure guarantees valid geometry.

Selected Game Projects

`DustyWalkingSim` (C, SDL2, GLSL)

2017—2D top-down exploration game

- Designed and implemented a 2D physics engine for rigid bodies.
- Supports near-infinitely large worlds with implementation of an infinite grid and sweep-and-prune broadphases working together—using wb_alloc.h.
- Implemented and refined a method for subpixel antialiasing on pixel art.

`Sparse` (C#, SDL2)

2014-15—Survival/engineering platformer

- Heavily modifies the Farseer Physics engine (a Box2D port) to improve serialization.
- Iterated on an extensible game data loader. Using JSON, you can define, inherit from, and override any definition loaded.
- Designed and implemented a highly configurable procedural generation system for 2D worlds.

Work, Other Skills and Education

- 2016-present: Teaching children's art and design classes at a local art association.
- 2014-15: Tutoring students in Java at the University of Redlands.
- Strong skills in C, C++, C#, Python, and Lua; knowledgeable about JavaScript, Java, ActionScript/HaXe
- Experienced with modern C/C++ compilers and debuggers, SIMD, cache optimization, OpenGL.
- 2013-2015: Two years at the University of Redlands, pursuing a Bachelor of Music in Tuba Performance and a Bachelor of Science in Mathematics.