GEORGE CORNEY

2 Millway Close ◇ Oxford ◇ OX2 8BJ
07542 045079 ◇ haxiomic@gmail.com

INTRODUCTION

Previous clients include Microsoft, LG, Atlantic Productions and the Met Office. My work both personal and contracted has reached millions of users and featured on sites like The Next Web, Gizmodo and FastCoDesign.

I specialize in web development and computer graphics, however, I'm never shy to grapple with a new domain to meet the demands of a project.

I'm an advocate of open source and you can find my contributions under the handle 'haxiomic'.

EXPERIENCE

LUSH – Interactive Installation & Other Projects March 2019 - August 2020
React, WebGL, GLSL, C++, emscripten, iOS, Android, TypeScript, 3D *Contracting*

- LUSH commissioned an interactive fluid art piece to be projected on a wall within a new flagship store opening in Shinjuku Japan. In addition, the piece was to be embedded as an experience within in their native mobile app LUSH Labs (iOS and android)
- The mobile app build mandated small binary sizes and close-to-the-metal performance, the in-store interactive required integrating with native C++ vision SDK and the project schedule necessitated a rapid turnaround time.
- I used GLSL to implement optical-flow tracking and the interactive fluid simulation and haxe to build cross-platform native OpenGL views that could be embedded easily within native and web apps with a low-footprint. This approach represents an alternative to heavy game engines like Unity and slower-iteration languages like C++ or Rust. It's an approach I'm trying to pioneer and develop open source projects to make it easier.
- Other projects involved compiling FFmpeg to WebAssembly so WebGL→MP4 video encoding could be performed in-browser, React for building UIs to host WebGL experiences and Babylon.js for PBR rendering.

VALIS - Genomics Startup December 2017 - March 2019
React, WebGL, GLSL, Text Rendering, TypeScript *Contracting*

- Developing a WebGL-based genomics data viewer, capable of navigating through gigabytes of data in realtime. Traditional genomics data is often heavy to load and slow navigate, this project involved developing a pipeline to enable streaming and rendering genomics data so it can be explored interactively within a web browser.
- To achieve best performance, a WebGL UI, animation and text rendering system was developed from the ground up.
- The VALIS viewer is now in-use at the ENCODE project to preview genomics data in the browser.

Microsoft June 2016 - August 2017
WebGL, GLSL, Physically Based Rendering, C++ *Contracting*

- Working on Microsoft's internal cross-platform physically-based rendering engine used in products including Office, Windows 10 shell and Paint3D.

- I developed the PBR pipeline for the WebGL version of the engine and much of my work on the WebGL engine is now available in BablyonJS.
- During this time also I worked on optimizing rendering and launch performance in MS Paint's successor, Paint3D.

Alchemy VR Feb 2016 - June 2016
VR, WebVR, C++, three.js, WebGL, GLSL, node.js *Contracting*

- Alchemy VR are transitioning to distributing their VR films directly to users (rather then through exhibitions), I was contracted to develop the cross-platform Netflix-style VR video player and VR storefront to enable this.
- To provide a cross-platform base (where bundling a webview wasn't an option) I implemented the WebGL API over GLES using the V8 JavaScript engine (C++), this enabled us to reap the benefits of developing for the web (live feedback, easy distribution, easier to hire developers etc), whilst allowing access to native code and APIs where necessary.
- WebVR was in the early experimental phase but much of the specification had been outlined. I used the early specification as a roadmap for implementing the native VR app, so that transitioning to in-browser WebVR would be a relatively painless process
- I was also responsible for developing the media delivery systems, which involved writing a content delivery server in node.js and developing an in-house DRM system.

Atlantic Productions Dec 2015 - Feb 2016
three.js, WebGL, GLSL, HTML, SCSS *Contracting*

- Contracted to develop interactive features for a companion website to Atlantic Productions recent series "David Attenbrough's Great Barrier Reef". Features included a WebGL globe displaying animated wind flow, atmospheric temperature and ocean current data

DinahMoe Nov 2015
WebGL, GLSL, three.js, Front-end Web *Contracting*

- Contracted by DinahMoe to develop portions of an interactive WebGL film, Canada Goose's "Out There". The project included developing a system to render WebGL output to video.

LG & Responsive Ads Sept 2015
WebGL, GLSL, Front-end Web *Contracting*

- As part of an LG OLED TV ad campaign, I was contracted to produce an interactive WebGL fluid and particle simulation in collaboration with responsiveads.
- The ad was required to run smoothly in all modern browsers, including mobiles and underpowered devices. The computationally intensive nature of the simulation required heavy optimization and an adaptive quality system.
- Traditionally this sort of simulation requires extensions to WebGL (such as floating point textures), to enable maximum compatibility, techniques were developed to pack simulation data into 4 byte textures which were available in every WebGL instance (whereas extensions are not).
- A early preview version of the ad is available.

Met Office June 2015
WebGL, GLSL *Consulting*

- The Met Office Informatics Lab were developing a browser-based 3D visualization of live weather in the UK. I was brought in to consult on techniques for high-performance volumetric rendering and approaches to resolve performance issues their initial WebGL ray marching implementation.

Alchemy VR June 2015, Sept 2015

Java, C++, Android, Mobile *Contracting*

- · Alchemy VR is a recently formed branch of Atlantic Productions, their VR premier "David Attenborough's First Life" was to be shown on 80 Gear VRs in the Natural History Museum from the 12th of June (2015).
- · Alchemy asked me to solve a number of critical problems two weeks before their premier deadline. I was tasked with developing a custom VR video player and bypassing the built-in Oculus Home without resorting to rooting the devices.
- · The custom VR video player was developed using C++, Java Native Interface and Oculus's Mobile SDK and the bypass was developed with Java and the Android API.

fffunction Apr 2015

JavaScript, backbone.js, HTML, SCSS *Contracting*

- · fffunciton is a digial design agency in the Southwest with clients that include Roland UK and the Bristol museums group (BMGA).
- · I was brought in to contribute to a browser-based book reader and preview app (commissioned by Oxford University Press). The app was developed with backbone.js, node.js and Grunt.
- · My role involved developing a page layout engine and viewer thumbnail alongside bug fixes.

Hive Nov 2013 - Jul 2014

Objective-C, OS X Reverse Engineering, UI & UX, JavaScript *Startup*

- · Hive was a team collaboration app I worked on with a small group during university. It was an experiment in developing the ideal collaboration tool. The philosophy was that team cohesion could be improved by reducing boundaries between computers; the goal was to be able to push content (including running programs) from one device to a teammate's immediately and intuitively (in a similar manner to moving windows between multiple displays).
- · The project won the Cisco Open Collaboration Challenge and was accelerated for 3 months at dotforge.
- · My role in the project was development of 'State-sharing' (which involved reverse engineering OS X's state saving feature), UI & UX design, and the development of the native OS X app.
- · I left the project after the release of OS X Yosemite which contained features (Continuity and Handoff) which competed with our core technology 'State-sharing'.

SELECTED OPEN SOURCE PROJECTS

WebGL Fluid September 2014

GLSL, Haxe, Lime, JavaScript, C++, WebGL

- · This project is a GPU fluid and particle simulation written in Haxe and GLSL, targeting HTML5 for browsers and C++ for desktop and iOS. The simulation solves the Navier-Stokes equation for incompressible flow over a grid with the Jacobi method and uses the velocity field to advect over 1 million particles.
- · The motivation for this project was to explore using WebGL for high performance physics simulations and to investigate the performance factors involved.
- · It's been played with approximately **2 million times** by **1.6 million users**, achieving a total of **8 million** pageviews.
- · It has reached the front page of Reddit (twice) and featured in articles on
  - FastCoDesign
  - The Next Web
  - Gizmodo

- engadget

GLSL Parser in Haxe Mar 2015 - Present
GLSL, Haxe, JavaScript, C, Context Free Grammars, LALR *Work in Progress*

- The aim of this project is to provide a cross-platform GLSL parser (and parser generator) that supports the GLSL reference language grammar.
- The motivation for the development was to enable compile-time transformations (such as minification or transpilation) of GLSL source, as well as tighter integration with the host codebase when working with the Haxe compiler.
- The project can be accessed and tested on github github.com/haxiomic/haxe-glsl-parser.

EDUCATION

University of Sheffield
B.Sc in Physics with Astrophysics ⋄ *First Class*

TECHNICAL STRENGTHS

Programming Languages	JavaScript, GLSL, C++, C, Objective-C, Haxe
Technologies & APIs	HMTL5, WebGL, Git
Platforms	Android, iOS, OS X, Linux
Skills	UI & UX design, Reverse Engineering

AWARDS

BAFTA - Best Digital Creativity - David Attenborough's Great Barrier Reef Dive (Atlantic Productions) February 2017
Android, Oculus, VR

Sudo Challenge VR/AR hackathon 1st place November 2017
WebGL, pixi.js, medical technology