



# OpenGL: A Love Story

Cass Everitt  
OpenGL Enthusiast

# Through 3D Colored Glasses

- .A brief look way back**
- .The march to desktop maturity**
- .Mobile grows up**
- .Detour: the ubiquitous internet**
- .What is an application?**
- .Web / Desktop / Mobile / Cloud – OpenGL is everywhere!**

# **% whoami**

- **IrisGL, OpenGL vis work in college**
- **NVIDIA**
  - Content technology, architecture, and Cg teams
- **Id Software – RAGE, some mobile**
- **Epic Games – UE3 for iOS and Tegra**
- **NVIDIA – Content technology**
  
- **Been around the block, but haven't strayed too far from 3D generally or OpenGL specifically**

# Then the Earth cooled...

- **SGI caught lightning in a bottle with IrisGL**
  - Simple, clear, easy to get started...
- **The API needed a little cleaning up to be an industry standard**
- **Established the ARB, drove standardization, birthed OpenGL!**
- **Same stuff we were doing in IrisGL, but it seemed cooler for some reason...**

# Enter the PC

- **My first apps ran on machines that cost \$\$\$**
- **By 1999, I was running OpenGL apps on sub-\$1k home PC**
  - TNT – not my first 3D accelerator, but the first one that accelerated 3D...
- **SGI never caught back up**
  - It took a while before PC won in every sense
  - But the writing was on the wall

# Nice long run

- **PC took the baton and sprinted – for years**
  - Multitexture, texture compression, cube maps, combiners, shaders, floating point render targets, MRT, geometry shaders, tessellation, compute, etc.
- **And not just features**
  - performance grew at a rate difficult to balance with CPU horsepower...

# News of OpenGL's death has been greatly exaggerated

- **If I had a nickel for every time someone told me that OpenGL was dead...**
- **And yet it never was. Not really even close.**
  - Because people stick with an API that doesn't break their code
  - Especially people who write professional applications!
- **But in the consumer space, OpenGL was very under-represented.**
  - Usual complaint: inconsistent/buggy drivers...
  - Sigh... this is still a valid complaint, as evidenced by WebGL on Chrome using D3D.

# Why didn't OpenGL just die?

- **Always industry pressure to keep exactly one open standard alive**
- **Some platforms had no ready alternative to OpenGL**
- **Large, expensive professional apps mostly stayed OpenGL**
- **OpenGL did not innovate, but it doggedly kept parity with D3D**
- **Lean times for consumer apps and games, but workstation business flourished**



# The plateau

- **GL3 and GL4 added some incredible new features, and went to great lengths to fill out skimpy aspects of the spec**
- **But no corresponding console hardware update, so focus largely remained on GL2 class functionality**
- **Tantalizing speculation on radical changes to rendering pipeline were entertained with Larrabee, but never materialized**

# A new upstart

- **Apple released a revolutionary new device.**
- **No 3rd party apps. No APIs. But despite lots of skepticism, it was a runaway success.**
- **Then, a year later came the 3rd party app support. And the world was suddenly different.**
- **Old fixed function OpenGL was cool again, because you could do it on your phone!**
- **Made us old farts feel giddy with nostalgia. :-)**

# Mobile skyrocket

- **Apple discovered that people like the “real” internet on these crazy non-traditional computing devices.**
  - Phones, tablets, pods.
  - Who knew a monolithic device with a multi-touch screen could be so right?
- **So did Google. And now there is such intense focus on that space, you know it is where to look for innovation.**
- **But the equation is a little different. Crazy performance, long battery life**
- **Best user experience isn't exactly like desktop, so more time researching how users wants to use device and app**
- **Devices already multi-core, faster and more RAM-laden than the desktops of few years back. Wow. They are just computers at the OS level, but with a radically different UI model.**

# Mobile > PC?

- **I know which is faster and which has more RAM**
- **But it is so cool having**
  - location services, accelerometer, compass, gyroscope, cameras, internet, 3D accelerator
  - in a device that's always with you!
- **Hard to compete with what those features can offer you as an application developer.**
- **Q: Did you ever think when cameras were first in cell phones, that they were pointless?**
  - I sure did. No internet. No way to upload images.
- **Do you now think cameras in cell phones are useful now? ·**
  - The Facebook app for iPhone changed my mind.
  - Magic is all these technologies together – greater than the sum of the parts

# The internets

- **Holy crap, internet technologies have matured a lot in the last 7 years or so. I guess I just wasn't paying attention**
- **It's come a long way from when I ran Trumpet Winsock on my Windows 3.1 machine with 8-bit color to use NCSA Mosaic over a 14.4kbps modem**
  - Also walked to school barefoot in the snow, uphill both ways...
- **So many technologies, so interconnected..., so robust and powerful, touch so many people**
  - Javascript, SQL, CSS, HTML5, PHP/Perl/Python, JSON, web services, ...
- **Mobile is the fastest growing part of that tapestry**
  - And it brings the network closer to the user
  - The internet is not just a "computer" thing anymore
  - It's always at your fingertips

# What is an application?

- **Written a Facebook application lately?**
  - Or Twitter or Linked-In integration?
- **The lines have been blurred.**
- **Applications and services co-mingle**
  - “We will add your distinctiveness to our own.” – Locutus of Borg
- **My hobby app is a satellite tracker for handheld...**
  - Internet fetch (http) for orbital elements
  - Added Facebook integration
  - Using Google Translate web services for localization
  - Keeping favorites list “on the cloud”
  - Generate “active users” map via pbuffer OpenGL rendering on Linux server
  - Could leave all this off, but user experience would suffer – really needs the internet

# Single player experience

- **Less buzz here, but still very important**
- **Even here though, elements of the networked ecosystem creeping in**
  - Social networking, leader boards, etc valuable outside the single player experience
- **Apps will have different levels of integration**
  - But all will have access to a lot more of it than before...
  - And it will change the way users want to experience game play

# OpenGL is everywhere

- **Graphics on mobile – OpenGL ES**
- **Graphics on desktop – OpenGL**
- **Graphics in the browser – WebGL**
- **Graphics on workstation – OpenGL**
- **Graphics on the cloud – OpenGL w/ pbuffers**
- **OpenGL is the core technology available everywhere**
  - Because it is Open
  - Which makes it attractive for software development
  - Reach more people with less platform-specific code
- **Mark my words – this internet thing is going to be BIG!**
  - And now the experience will include all 3 dimensions!