Overview

- Introduction
- Android Tools
- Game Development
- OpenGL ES
- Marketing
- Summary
- Questions
Introduction

Disgruntled Rats incorporated on March 31st, 2011.

We are a small company that specializes in game development for mobile devices.

Sean Godinez - Software Engineer - Graphics and C/C++

Michael Boldischar - Software Engineer - Java and Linux

Brian Morgan - Music, Sound, Marketing, Business, and Advertising
Question!

Who discovered quaternions and when?
Android Tools

- Eclipse - Integrated Development Environment (IDE)
- ADT - Android's Eclipse Plugin
- Ant (show off our build environment)
  http://www.disgruntledrats.com/?page_id=309
  Why Ant? - Larger group projects
  The Eclipse IDE works well for one developer
- Can invoke Ant scripts from cron scripts
- Hudson - Automatic build tool
  Invokes Ant when new code is checked in
- Send emails when code doesn’t compile
- Android SDK - includes ADB
  http://developer.android.com/sdk/installing.html
Sprites

A sprite is a texture mapped to a quad. They can be animated by updating their position / texture coordinates over time.

- Basic
  - Plants, grass, signs, etc
- Axial
  - LOD Technique for far off mountains, castles, etc
- Screen-Aligned
  - HUDs
  - Text
- View Point
  - Point Sprites => Particle Effects
  - Clouds
World-Oriented Sprites

Screen-Aligned & Axial Sprites
We developed a sample sound project that builds and deploys with Ant.
Follow the following link to our website to get started:
http://www.disgruntledrats.com/?page_id=545

Let's read some code!
Data Formats

- **Text-based**
  - Pros - easy to parse
  - Cons - bloated model files

- **Binary**
  - Pros - very compact, tightly packed data
  - Cons - harder to parse, have you tried byte shifting in Java lately?
3D Models - File Types

What are some of the file types?

- **3DS - Autodesk 3DS Max (.3ds)**
  - Pros - de-facto standard for modelling, hierarchical data
  - Cons - all meshes are made from triangles, closed standard

- **COLLADA**
  - Pros - open standard for transferring model data
  - Cons - very complex, lots of elements and relationships, would take a long time to fully implement a parser, 3D Studio Max exports lose some animation data

- **Milkshape, SketchUp, X3D, etc...**
Question!

Who's hotter than J-WOWW?
AI & Physics

● AI Behaviors
  ○ Seek and Evade
  ○ Cohesion and Separation
  ○ Alignment
● A-Star & Dijkstra Search Algorithms

● Classical Mechanics
  ○ Position and its derivatives
  ○ Force
In Game User Interfaces

- **Standard Android UI and widgets**
  - Pros - Library is already written for you!
  - Cons - None

- **Build Your Own - Orthographic Sprites, Font Textures, or Font Algorithms**
  - Pros - Build it once, apply it anywhere in a game (e.g. a newspaper, quests, etc...)
  - Cons - Time consuming process, fonts are not trivial
Version Checking

You should check which version of OpenGL ES is supported by the current device.

```java
/**
 * Checks to see if the OS supports the given version of OpenGL ES.
 * @param version the version to check against in hex
 * @return true if the version is supported by the OS, otherwise false
 */
private static boolean supportsOpenGLEsVersion(int version) {
    ActivityManager activityManager = (ActivityManager)
            activity.getSystemService(Activity.ACTIVITY_SERVICE);
    int foundVersion = activityManager.getDeviceConfigurationInfo().reqGlesVersion;
    System.out.println("OpenGL ES version found: " + foundVersion);
    return foundVersion >= version;
```
Question!

Who would win in a fight and why?

Gordan Freeman

Chuck Norris
What are they?

- "A point is represented by a set of floating-point numbers called a vertex" - OpenGL Red Book

How do we get all the data associated with vertexes to the GPU?

Remember these calls?

```c
glBegin(GL_TRIANGLES);
glVertex3f( 0.0f, 1.0f, 0.0f); // Top
glVertex3f(-1.0f, -1.0f, 0.0f); // Bottom Left
glVertex3f( 1.0f, -1.0f, 0.0f); // Bottom Right
glEnd(); // Finished Drawing The Triangle
```

Why are they a bad idea on a mobile device?
Vertex Buffers and Textures

What is the best way to pack data?
  ● It depends.

The Goal: Switch between buffers as little as possible during rendering.

Separate buffers for vertex, normal, texture data: Tradeoff between simplicity and performance.

Let's read some code!
http://www.disgruntledrats.com/?page_id=545
Transformations

● Right Hand Rule vs Left Hand Rule
● Column Vector vs Row Vector
● Column Major vs Row Major
● Translation
● Rotation
● Scale
● 4x4 Transformation Matrix
● Rotation about arbitrary axis
● Quaternions
● Inverting Matrices
● Transpose of an Inverted Matrix
● Transformations in a Scene Graph
● OpenGL ES 2.0 does NOT have matrix stack
● Check out our website for a tutorial on scene graphs
4x4 Transformation Matrix

- **Translation**
  \[
  \begin{bmatrix}
  1 & 0 & 0 & dx \\
  0 & 1 & 0 & dy \\
  0 & 0 & 1 & dz \\
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]

- **Scaling**
  \[
  \begin{bmatrix}
  s_x & 0 & 0 & 0 \\
  0 & s_y & 0 & 0 \\
  0 & 0 & s_z & 0 \\
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]

- **Rotation about X-axis**
  \[
  \begin{bmatrix}
  1 & 0 & 0 & 0 \\
  0 & \cos \theta & -\sin \theta & 0 \\
  0 & \sin \theta & \cos \theta & 0 \\
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]

- **Rotation about Y-axis**
  \[
  \begin{bmatrix}
  \cos \theta & 0 & \sin \theta & 0 \\
  0 & 1 & 0 & 0 \\
  -\sin \theta & 0 & \cos \theta & 0 \\
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]

- **Rotation about Z-axis**
  \[
  \begin{bmatrix}
  \cos \theta & -\sin \theta & 0 & 0 \\
  \sin \theta & \cos \theta & 0 & 0 \\
  0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]
Question!

Who's hotter than Jessica Alba?
Camera

- Projections (Orthographic, Perspective, Isometric)
- Constraints on Calculating Axis (Up x View)
- View Matrix transforms from World Space to Eye Space
  - rotates \((u, v, n)\) to align with world \((x, y, z)\)
  - translates to origin

\[
V = \begin{pmatrix}
    u_x & u_y & u_z & 0 \\
    v_x & v_y & v_z & 0 \\
    n_x & n_y & n_z & 0 \\
    0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
    1 & 0 & 0 & -e_x \\
    0 & 1 & 0 & -e_y \\
    0 & 0 & 1 & -e_z \\
    0 & 0 & 0 & 1
\end{pmatrix}
\]

\[
V = \begin{pmatrix}
    u_x & u_y & u_z & d_x \\
    v_x & v_y & v_z & d_y \\
    n_x & n_y & n_z & d_z \\
    0 & 0 & 0 & 1
\end{pmatrix}
\]

where \(d = (-e \cdot u, -e \cdot v, -e \cdot n)\)
OpenGL 1.x Graphics Pipeline

Existing Fixed Function Pipeline

API
- Primitve Processing
- Vertex Buffer Objects

Transform and Lighting
- Primitve Assembly
- Rasterizer

Triangles/Lines/Points

Texture Environment
- Colour Sum
- Fog

- Alpha Test
- Depth Stencil
- Colour Buffer Blend
- Dither

Frame Buffer
OpenGL 2.0 Graphics Pipeline

ES2.0 Programmable Pipeline

API

Primitive Processing

Vertex Buffer Objects

Triangles/Lines/Points

Vertices

Vertex Shader

Primitive Assembly

Rasterizer

Fragment Shader

Depth Stencil

Colour Buffer Blend

Dither

Frame Buffer
Other Considerations for OpenGL ES 1.0 versus 2.0

- Android Support
  - OpenGL ES 1.0
    - Supported since Android 1.0
  - OpenGL ES 2.0
    - Supported since Android 2.2
- 80% of devices now support Android 2.2 or greater
- 90% support 2.0
- More devices are supporting OpenGL ES 2.0 every day
What was the name of the end boss in World of Warcraft's Molten Core?
Vertex Shader

![Figure 1-2: OpenGL ES 2.0 Vertex Shader](image)
Primitive Assembly

**Figure 7-6** OpenGL ES Primitive Assembly Stage
Rasterization

**Figure 1-3** OpenGL ES 2.0 Rasterization Stage

- **From Primitive Assembly**
  - Point-Sprite Rasterization
  - Line Rasterization
  - Triangle Rasterization

  Output for each fragment—screen \((x_w, y_w)\) coordinate, attributes such as color, texture coordinates, etc.

  To Fragment Shader Stage
Fragment Shader

Figure 1-4 OpenGL ES 2.0 Fragment Shader
Per-Fragment Operations

Figure 1-5  OpenGL ES 2.0 Per-Fragment Operations
Google does not currently have an Android emulator that supports OpenGL ES 2.0, but nVidia does!

http://developer.nvidia.com/tegra-resources#tools

Android Triangle Demo:

Question!

What is the game engine in the new Battlefield 3 FPS?
Marketing
Summary

- Game development is challenging. Lots of math.
- No one is hotter than Jessica Alba.
- You are probably excited to get home.
References

● Google Android SDK
● OpenGL ES 2.0 Programming Guide
● Real-Time Rendering, 3rd Edition
● University of Minnesota, CSCI 5107/5108
● Android Versions
  ○ http://developer.android.com/resources/dashboard/platform-versions.html
  ○ http://developer.android.com/guide/topics/graphics/opengl.html
  ○ http://developer.android.com/resources/dashboard/opengl.html
● Before vertex buffers
  ○ http://nehe.gamedev.net/tutorial/your_first_polygon/13002/
References Continued

- WOW Picture
  - [http://www.wowwiki.com/Ragnaros_%28tactics%29](http://www.wowwiki.com/Ragnaros_%28tactics%29)
- Gordan Freeman
  - [http://www.rankopedia.com/CandidatePix/40373.gif](http://www.rankopedia.com/CandidatePix/40373.gif)
- Chuck Norris
- Jessica Alba
- J-WOWW
- Battlefield 3
  - [http://videogames.techfresh.net/battlefield-3-gameplay-trailer/](http://videogames.techfresh.net/battlefield-3-gameplay-trailer/)
Questions?