

iPhone Dev with OpenGL



Paul Solt

Agenda

- iPhone Developer University Program
- Xcode
- Objective-C Primer
- OpenGL ES
- Demos
- Resources/Questions



iPhone Developer University Program

- RIT is registered in the University Program
- Test and debug applications on devices
- Distribute applications
 - Internally at RIT

```
@synthesize direction;
@synthesize staticImage;
@synthesize activeImage;

- (id)initWithFrame:(CGRect)frame
    if (self = [super initWithFrame:frame])
        // Initialization code
        NSLog(@"Init me");
    }
    return self;
}

- (void)setupStaticImage:(NSString*)name
    self.staticImage = [self loadImageWithName:name];
    self.activeImage = [self loadImageWithName:name];
    self.image = self.staticImage;
}

// Load an image. It will display
- (UIImage *) loadImageName:(NSString*)name
    UIImage *image = [UIImage imageNamed:name];
    if(!image) {
        NSLog(@"Error loading the image");
    }
    return image;
}

- (void)touchesBegan:(NSSet *)touches
    direction = CGPointMake(0,0);
```

Xcode

- Install the Xcode iPhone 3.1.2 SDK from **developer.apple.com/iphone**
 - Free to develop with the iPhone Simulator
- Interface Builder - GUI
- iPhone Simulator - “virtual iPhone”



Xcode Pro Tips

- Header file = Command + double-click symbol
- Quick Help = Option + double-click symbol
- Reference Document = Command + Option + double-click symbol
- Switch between header/source files = Command + Option + Up-arrow
- Learn something you don't know = Google Search

Objective-C Primer

- Read “*The Objective-C 2.0 Programming Language*”
- Extension to C
- Mix C++ and Objective-C in a single file

Objective-C Primer

- C++/Java

```
public void increment(Object sender) {  
    count++;  
    textField.setIntValue(count);  
}
```

- Objective-C

```
- (void)increment:(id)sender  
{  
    count++;  
    [textField setIntValue:count];  
}
```

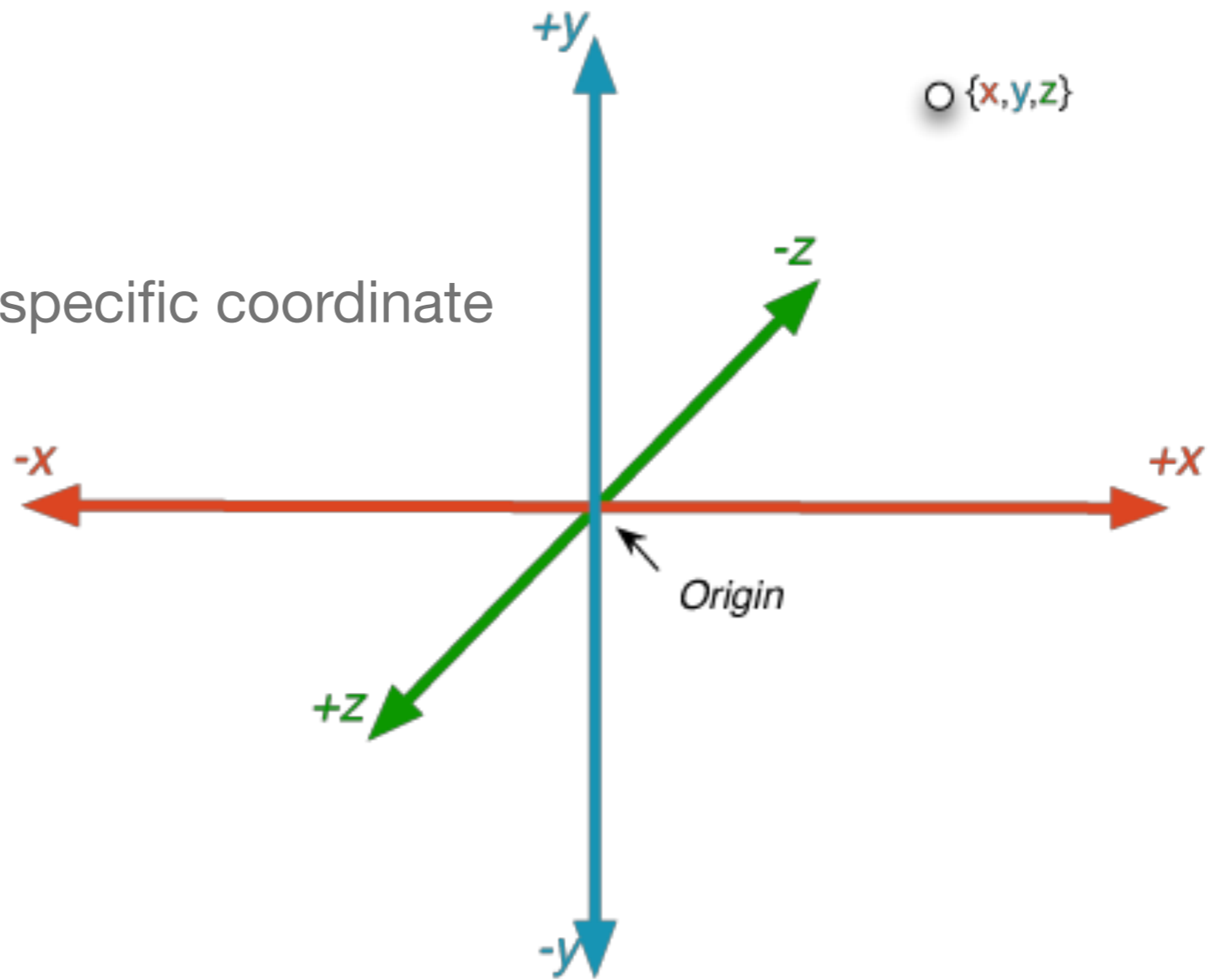
OpenGL ES

- Direct Mode vs. Immediate Mode
 - OpenGL ES only supports Immediate mode!
- Data types
 - GLfloat, GLint, GLboolean, GLenum, GLbyte
- Vertex - A point in 3D
 - arbitrary scale

Coordinates

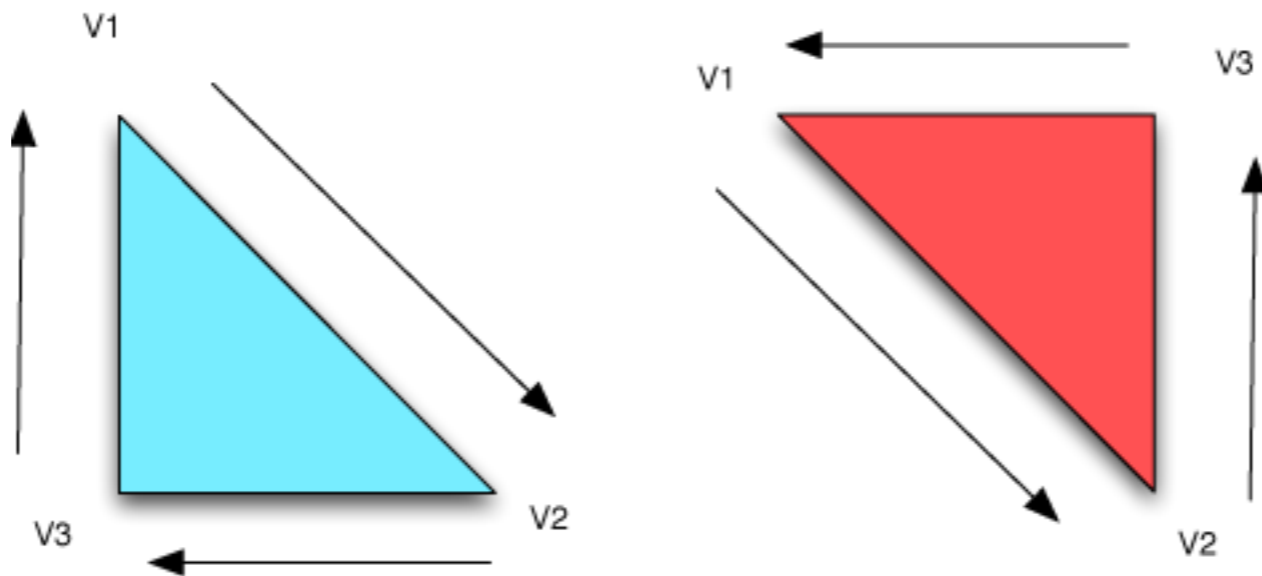
```
GLfloat vertex[3];  
vertex[0] = 15; // X  
vertex[1] = 20; // Y  
vertex[2] = 5;  // Z
```

- Indices 0,1,2 represent the specific coordinate



Winding and Back face Culling

- Clockwise = back face winding
- counter-clockwise = front face winding
- Culling = Triangle removal



Vertex Array

- An array of vertices
- Example:
 - Triangle = 3 vertices
 - 5 triangles would require $5 * 3 = 15$ vertices
 - 15 vertices * 3 coordinates per vertex = 45 GLfloats

OpenGL ES Code

```
Vertex3D v1 = Vertex3DMake(5.0, 5.0, -3.0);
Vertex3D v2 = Vertex3DMake(0.0, 5.0, -3.0);
Vertex3D v3 = Vertex3DMake(0.0, 0.0, -3.0);
Triangle3D triangle = Triangle3DMake(v3, v2, v1);

glLoadIdentity();
glClearColor(1.0, 1.0, 1.0, 1.0);
glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

glEnableClientState(GL_VERTEX_ARRAY);
glColor4f(1.0, 0.0, 0.0, 1.0);
glVertexPointer(3, GL_FLOAT, 0, &triangle);
glDrawArrays(GL_TRIANGLES, 0, 9);
glDisableClientState(GL_VERTEX_ARRAY);
```

glVertexPointer()

- Size = # coordinates

```
void glVertexPointer( GLint size,  
                    GLenum type,  
                    GLsizei stride,  
                    const GLvoid *pointer )
```

- Type = data type i.e. GL_FLOAT

- Stride = # bytes between vertices (generally 0)

- pointer = first coordinate in the vertex array

```
glVertexPointer(3, GL_FLOAT, 0, &triangle);
```

glDrawArrays()

- mode = Type of primitive
 - GL_TRIANGLES
- first = starting index
- count = number of indices

```
void glDrawArrays( GLenum mode,  
GLint first,  
GLsizei count )
```

```
glDrawArrays( GL_TRIANGLES, 0, 9 );
```

Demo: OpenGL ES Basics

- Shows a simple triangle in a viewport



Demo: Cocos2D



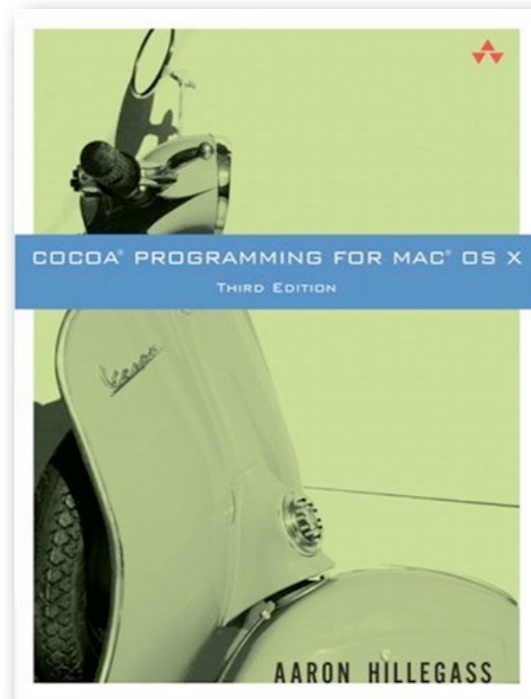
Demo: Ray Tracer iPhone

- Ray tracer written using C++ and OpenGL
- Ported to OpenGL ES
 - Slow on iPhone (133 seconds to render)



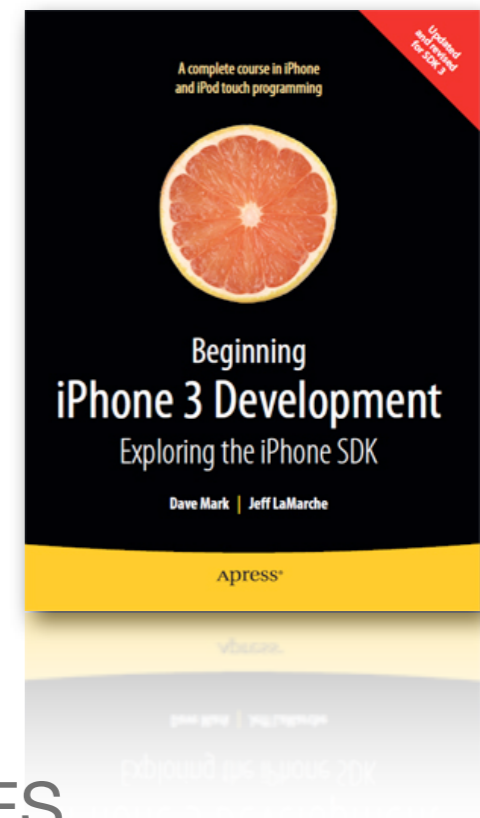
Resources

- [Cocoa Programming for Mac OS X](#) by Aaron Hillegass (Third Edition)
- Stanford iPhone Course (cs193p.stanford.edu)
 - Search “*iPhone Application Programming*” in iTunes



Resources

- [Beginning iPhone 3 Development: Exploring the iPhone SDK](#) by Jeff LaMarche
- Blog: iphonedevdevelopment.blogspot.com
- OpenGL ES Xcode Project Template
 - <http://www.innerloop.biz/code/Empty%20OpenGL%20ES%20Application.zip>

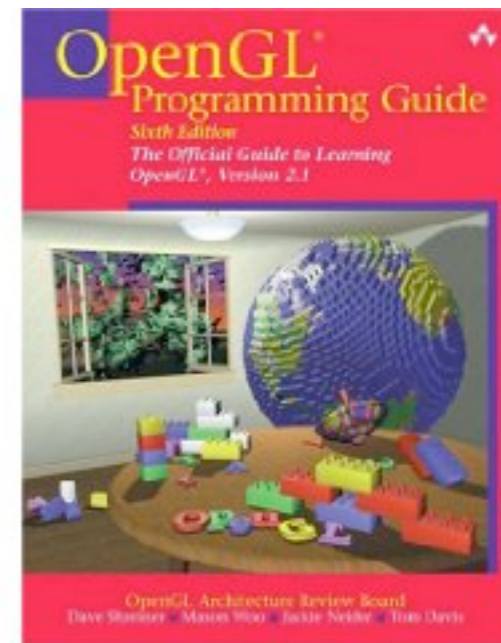


Resources

- OpenGL - “The Red Book”
 - OpenGL(R) Programming Guide: The Official Guide to Learning OpenGL (R), Version 2.1 (6th Edition) (Paperback)

- Cocos2D 

- <http://code.google.com/p/cocos2d-iphone/>



Questions
