

# Introduction to Scene Kit

Demitri Muna

*CocoaHeads • 13 September 2012*

# Introduction to Scene Kit for Non-Artists

Demitri Muna

*CocoaHeads • 13 September 2012*

# What is Scene Kit?

- 3D API
- Cocoa layer that abstracts OpenGL
- Makes relatively easy 3D tasks simple
- Spiritual descendent of QuickDraw 3D
- Basic shapes (“primitives”) provided
- Integrated with Core Animation



**Wil Shipley**  
@wilshipley

 Follow

Now it can be said: SceneKit is the best, most revolutionary API in 15 years. Well-designed, fast, nearly flawless.



# What Isn't Scene Kit?

- A modeler— BYOG (bring your own geometries)
- Well documented
- Available on iOS (yet?)
- No physics engine (but hooks available)

# Why Not Just Use OpenGL?

- OpenGL is *very* low level— deals with arrays of raw numbers
- No key-framing — you draw every step with a timer when objects move
- Very steep learning curve

# Building a 3D Scene

Minimum to create a 3D scene:

- a camera
- a light
- a 3D object

(And the SceneKit and QuartzCore frameworks)

Well, that sounds simple enough...



# Scenes and Nodes

- There is an `SCNScene` object which contains everything in the scene (i.e. the 3D world).
- A scene contains a “root node” which can hold objects and hierarchies.
- Nodes can be placed hierarchically within other nodes. Imagine placing a flag object onto a sphere object – you want the flag to move as a “child” of the sphere when it moves and not have to maintain its movement independently.
- Nodes have properties such as position, rotation, opacity, etc.

# *Demo I*

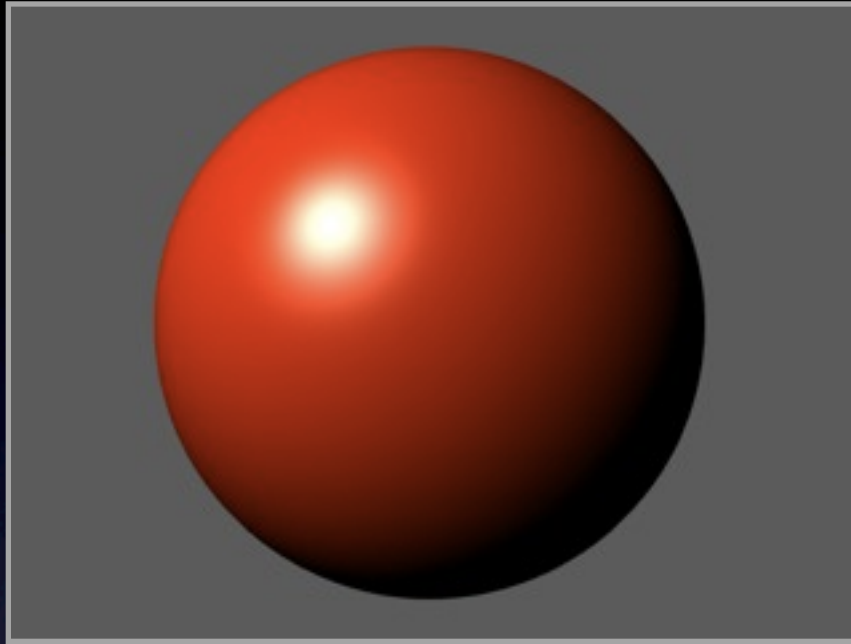


# Demo 1 Steps

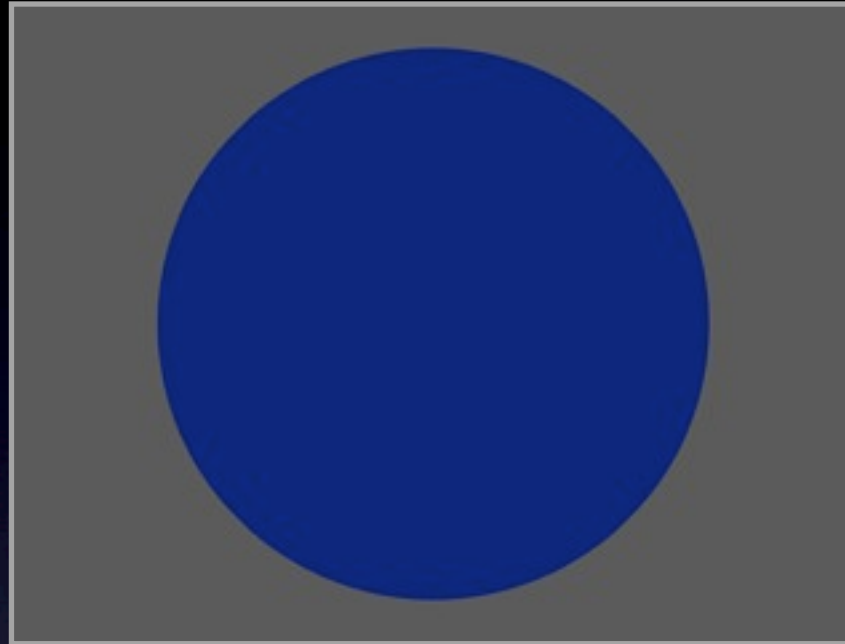
## Steps performed in demo app

- add QuartzCore, SceneKit frameworks
- subclass NSViewController
  - add `#import <SceneKit/SceneKit.h>`
- in xib, drag new SCNView to window and fill view
- set NSViewController's view to the SCNView

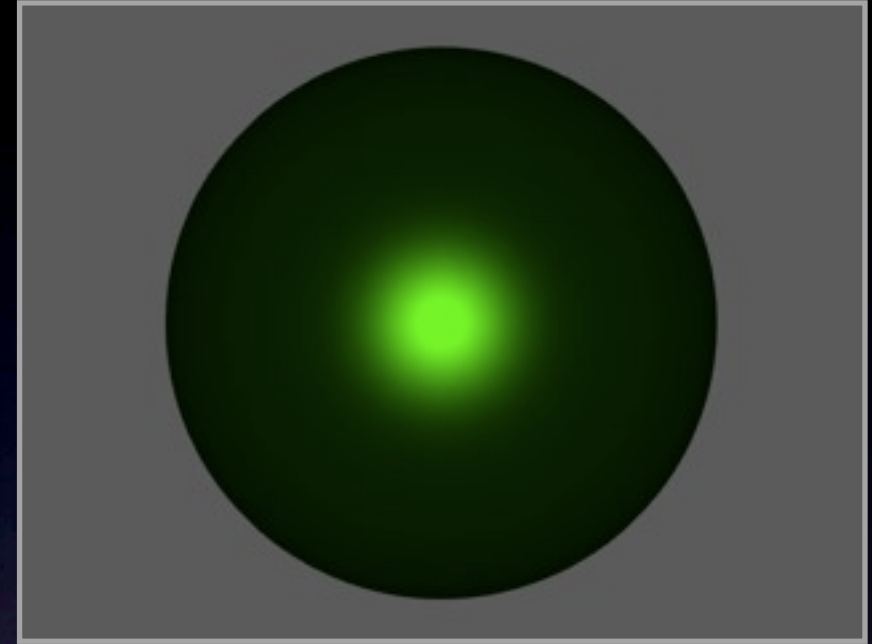
# Lighting



diffuse



ambient



specular

Types describe the reflective properties of the material:

- Diffuse: directed light, reflects in all directions, shows matte surfaces
- Ambient: flat lighting, coming from (thus reflecting to) all directions
- Specular: directed light, shows shininess of surface, reflects in a single direction

# Materials

- An object's material can be as simple as a flat color, an image, or an image with texture (bump map) and translucency.
- A single object can have different materials to reflect each type of lighting.



# Text

- Text can be rendered from any font (and retains correct kerning!).
- Made 3D through extrusion.
- Has a “chamferRadius” property, i.e. amount of rounding of edges.
- Any unicode character (except emoji) can be made into a geometry...

→ ⊙ Δ ⇄ ☺ ✓ ♪ ♯ ∩ ⊕

# *Demo 2*

# Digital Assets

- Scene Kit supports COLLADA 3D objects in DAE (digital asset exchange) format.
- These files can contain full scenes with objects, textures, and animation.
- Apple's sample code best example of how to do this.
- Be creative— you can do quite a lot with primitive geometries.
- Google's SketchUp provides access to thousands of models provided by the community (of wildly varying quality).

File → 3D Warehouse → Get Models...



# CAAnimation

- Scene Kit object properties are animatable through CAAnimation using the exact same API that you are familiar with.
- OpenGL doesn't support key-framing, but Scene Kit does with CAAnimation.

# Hit Detection

- Scene Kit makes hit detection easy through `NSEvent`.
- Returns object hit by mouse as well as world, local, and texture coordinates.
- For example, one can easily convert a mouse click to a lat/lon position on a globe.

# *Demo 3*



# Caveat

- Be aware to limit the size of your resources (e.g. image materials).
- Something that looks great on your iMac may render into garbage on a MacBook Air.
- Be sure to test on low-powered (integrated) GPU machines before shipping!

# Resources

## Apple Links

<https://developer.apple.com/videos/wwdc/2012/?id=504>

[http://developer.apple.com/library/mac/#documentation/3DDrawing/Conceptual/SceneKit\\_PG/Introduction/Introduction.html](http://developer.apple.com/library/mac/#documentation/3DDrawing/Conceptual/SceneKit_PG/Introduction/Introduction.html)

## Tutorials

<http://iphonedevdevelopment.blogspot.com/2012/08/an-introduction-to-scenokit.html>

<http://weblog.bignerdranch.com/754-scenokit-in-mountain-lion/>

## Examples

<http://www.notesfrommandy.com/2012/08/>

# Resources

## Rotation of individual scene objects

<http://stackoverflow.com/questions/4251716/iphone-mac-accelerate-framework-vector-scale-and-vector-normalization>

<http://www.raywenderlich.com/12667/how-to-rotate-a-3d-object-using-touches-with-opengl>

<http://www.tecgraf.puc-rio.br/~mgattass/fcg/material/shoemake92.pdf>

See: GLKQuaternion\* functions (introduced OS 10.8, iOS 5).  
Good opportunity to use the Acceleration framework (e.g. calculating dot and cross products).