10/06/17

Swiftly Into ARKit

Slide Deck Link

http://bit.ly/2fWY6m3

Project Link

http://bit.ly/2fW00D8

Who Are We?

- Johan Ospina
  - Next Gen Apps Engineer at Wayfair
- Email: jospina@wayfair.com
- LinkedIn: https://www.linkedin.com/in/johanos/
- GitHub: https://github.com/johanos

- Zachary Gay
  - Senior iOS Engineer at Wayfair
- Email: zgay@wayfair.com
- LinkedIn: linkedin.com/in/zachary-gay-a91b3033

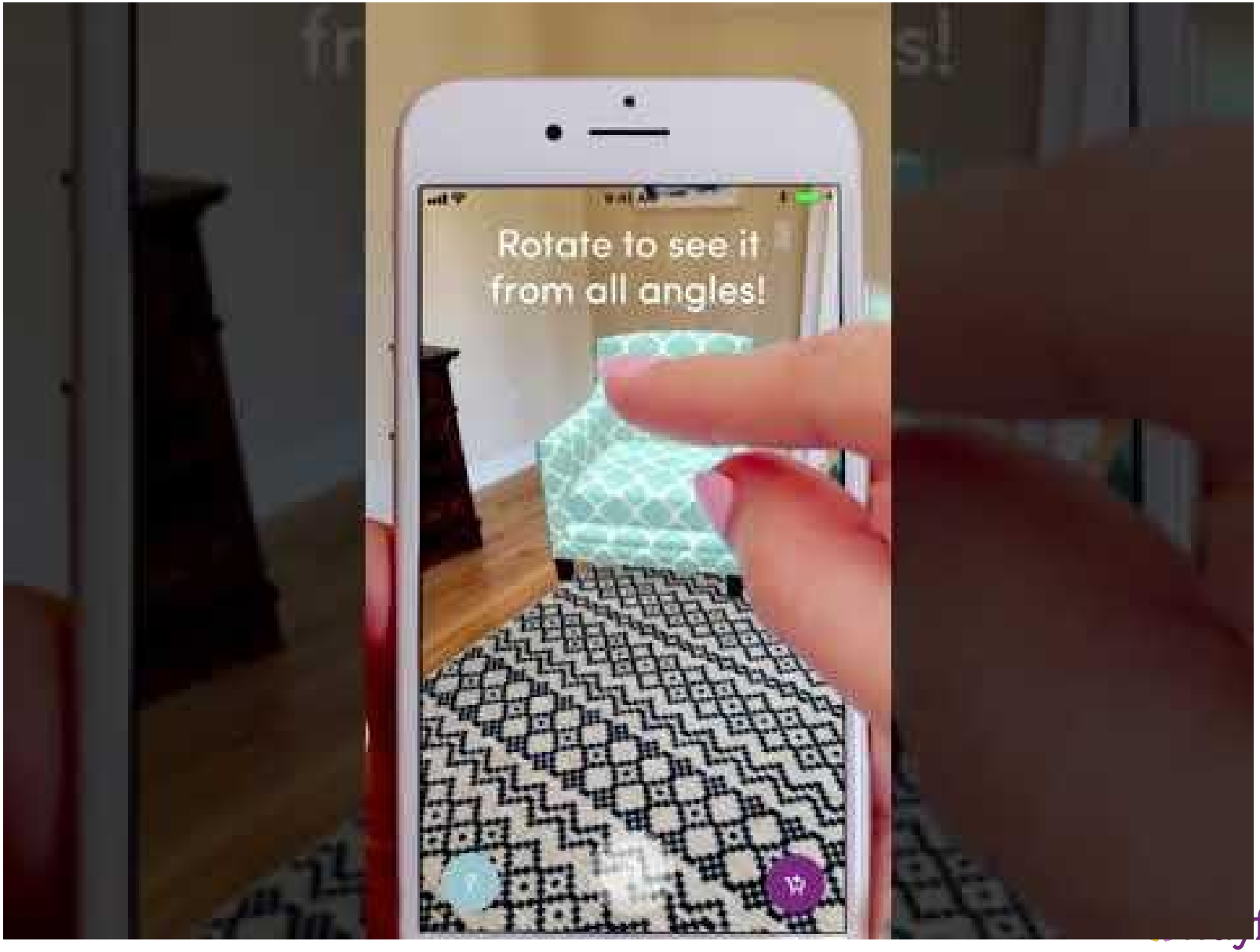
Goals And Expectations

- Understand the basics of ARKit:
  - Tracking
  - Scene Understanding
  - Rendering (SceneKit)

- Expectations:
  - Working Knowledge of iOS
  - Basic Understanding of Swift

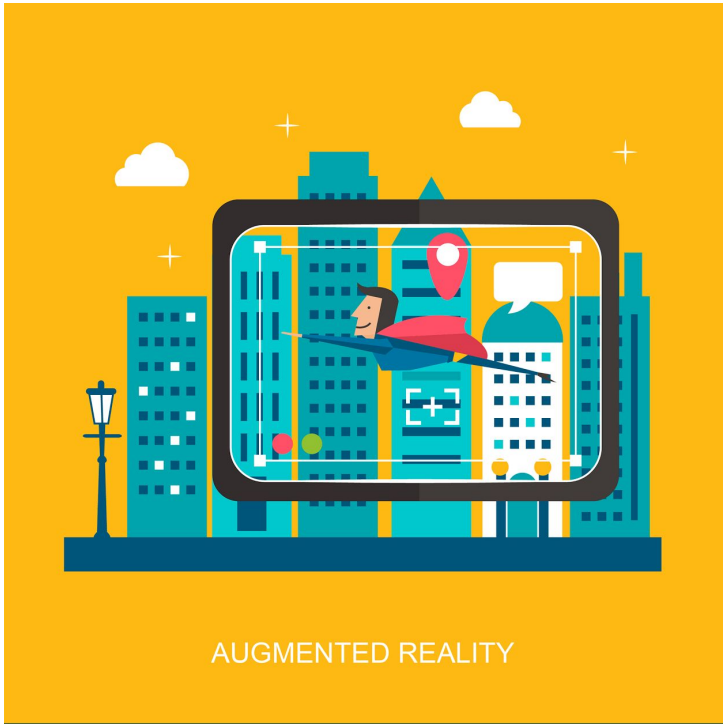
- Extra Credit:
  - Gestures
  - Model Management

AR at Wayfair



What is Augmented Reality?

Augmented Reality adds simulated experiences into a real environment

Augmented Reality	Virtual Reality
Real Environment + Virtual Elements	Virtual Environment + Virtual Elements
	

What is ARKit?

- System-level framework for AR
- Provides High-Level API for developers
- Released with iOS 11
- Supported Devices:
  - iPhone 6S or newer
  - iPad Pro or newer
  - (A9 SoC and above)

How Does ARKit Work?

- No Additional Hardware
- Existing Technologies:
  - Ambient Light Estimation
  - Sensor Fusion
  - Computer Vision
  - Game Engines
- Combined To Facilitate AR:
  - Tracking
  - Scene Understanding
  - Rendering

Tracking

- Updates Position and Orientation Of Device
- Measures Physical Distances
- Finds Feature Points

https://devstreaming-cdn.apple.com/videos/wwdc/2017/602pxa6f2vw71ze/602/602_introducing_arkit_augmented_reality_for_ios.pdf

Scene Understanding

- Tracks Ambient Light Conditions
- Computes Average Colors
- Plane Detection
- Hit Testing

https://devstreaming-cdn.apple.com/videos/wwdc/2017/602pxa6f2vw71ze/602/602_introducing_arkit_augmented_reality_for_ios.pdf

Rendering Engines

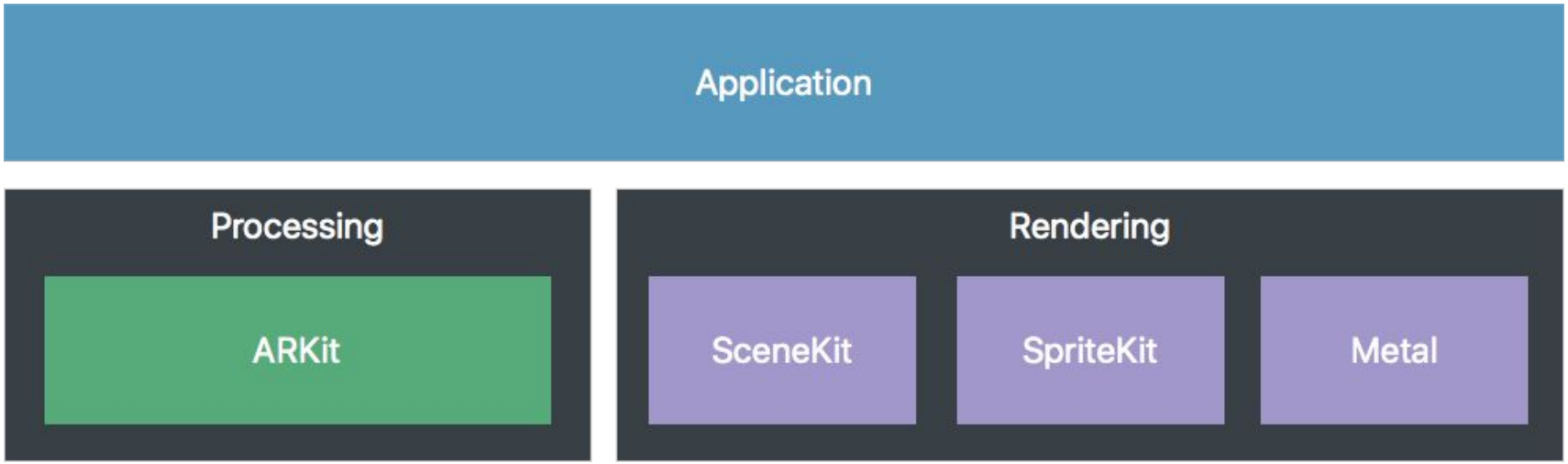
1. **SceneKit**

2. SpriteKit

3. Unity

4. Unreal

Essentially

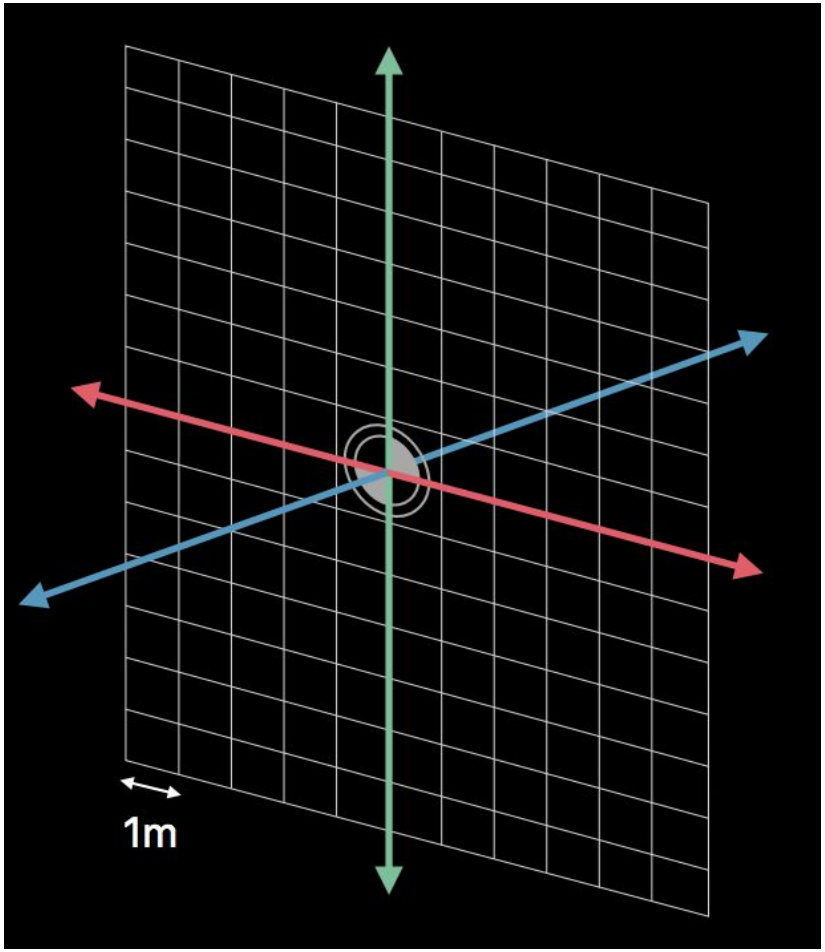


https://devstreaming-cdn.apple.com/videos/wwdc/2017/602pxa6f2vw71ze/602/602_introducing_arkit_augmented_reality_for_ios.pdf

The App

Tracking

- Position and Orientation
- Physical Distances
- Relative to starting position

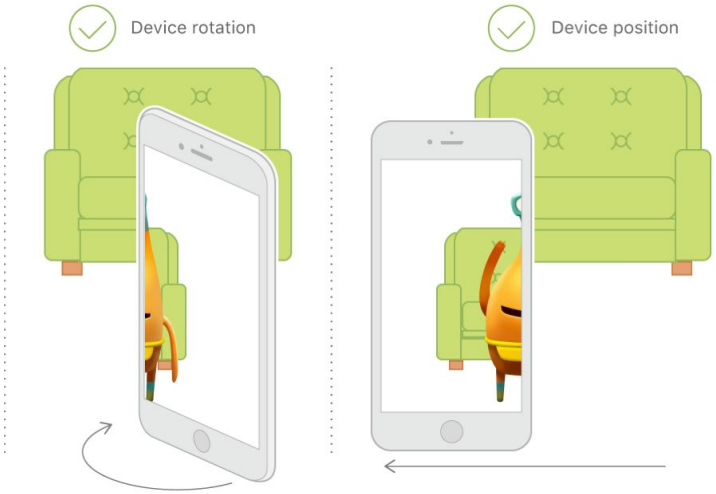


Session Based API

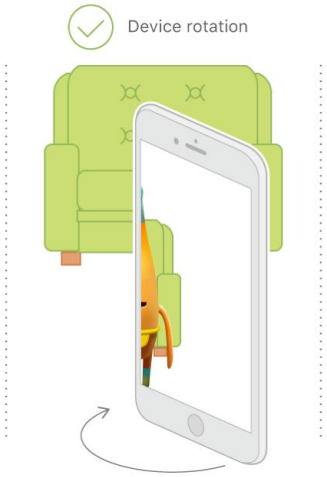
- ARSessionConfiguration Base Class

- ARFaceTrackingConfiguration
  - iPhone X Only

- ARWorldTrackingConfiguration
  - 6 Degrees of Freedom
    World Tracking



- AROrientationConfiguration
  - 3 Degrees of Freedom
    Orientation Tracking



Running an AR Session

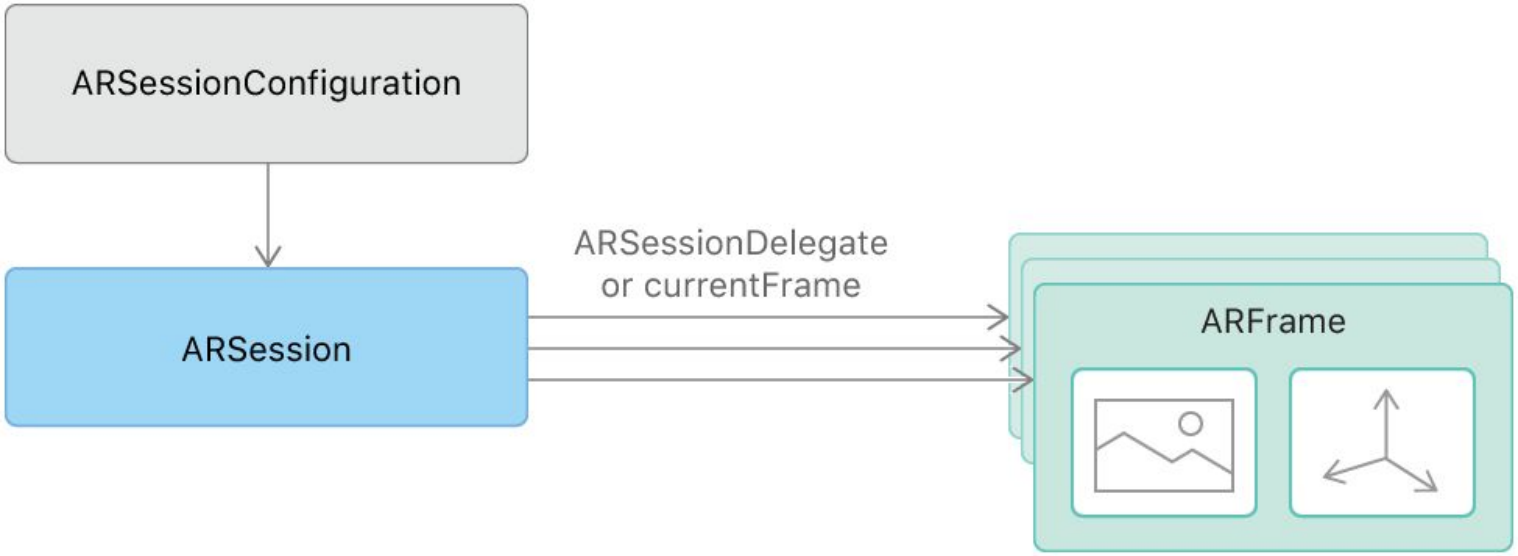
```
// Run your session
session.run(configuration)

// Pause your session
session.pause()

// Resume your session
session.run(session.configuration)

// Change your configuration
session.run(otherConfiguration)
```

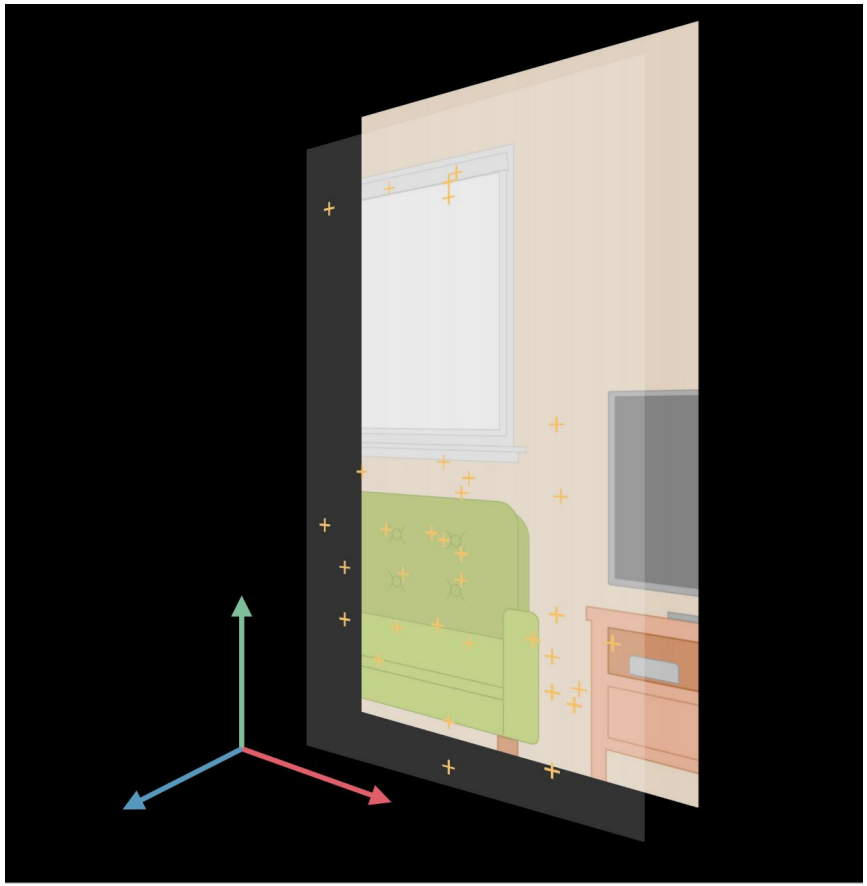
ARFrame



https://docs-assets.developer.apple.com/published/ffb3831f78/c162c528-dc03-494d-a5da-c23a8691a98e.png

ARFrame

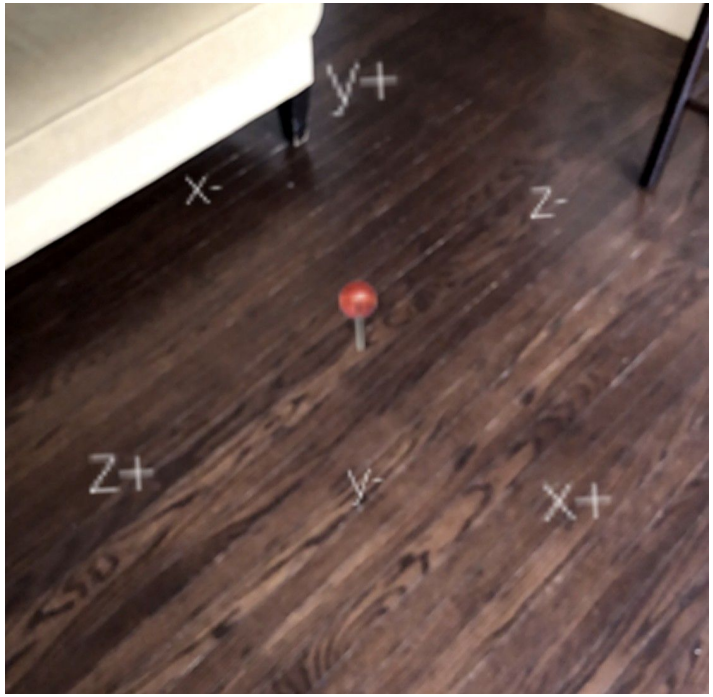
- Camera Image (Background)
- Tracking Info
- Scene Information



http://www.techeblog.com/index.php/tech-gadget/innovative-camera-has-no-display-splits-apart-to-let-you-manually-frame-a-photo

ARAnchors

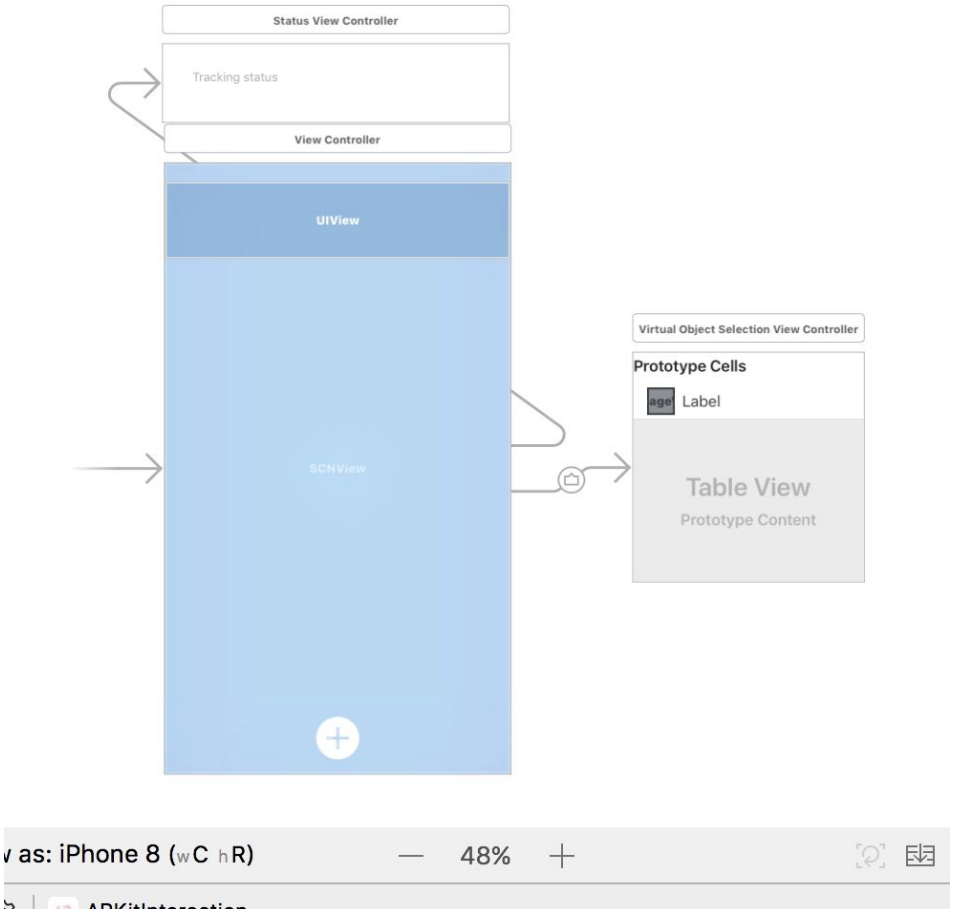
- Real-World Position and Orientation
- Persist through life time of session
- Plane detection actively adds them



https://medium.com/@yatchoi/getting-started-with-arkit-real-life-waypoints-1707e3cb1da2

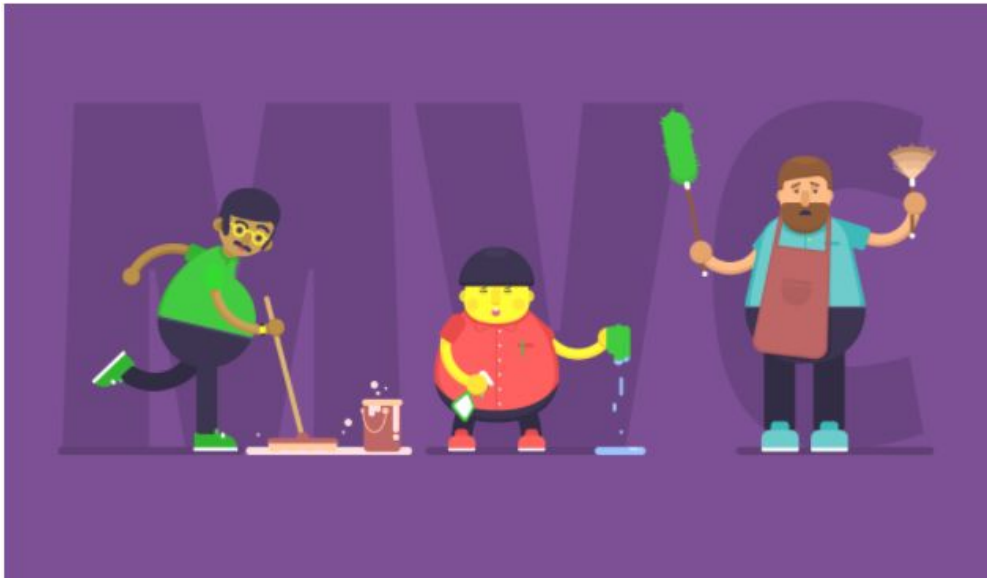
Main.storyboard

- UI Elements are connected to variables
- One way of doing this



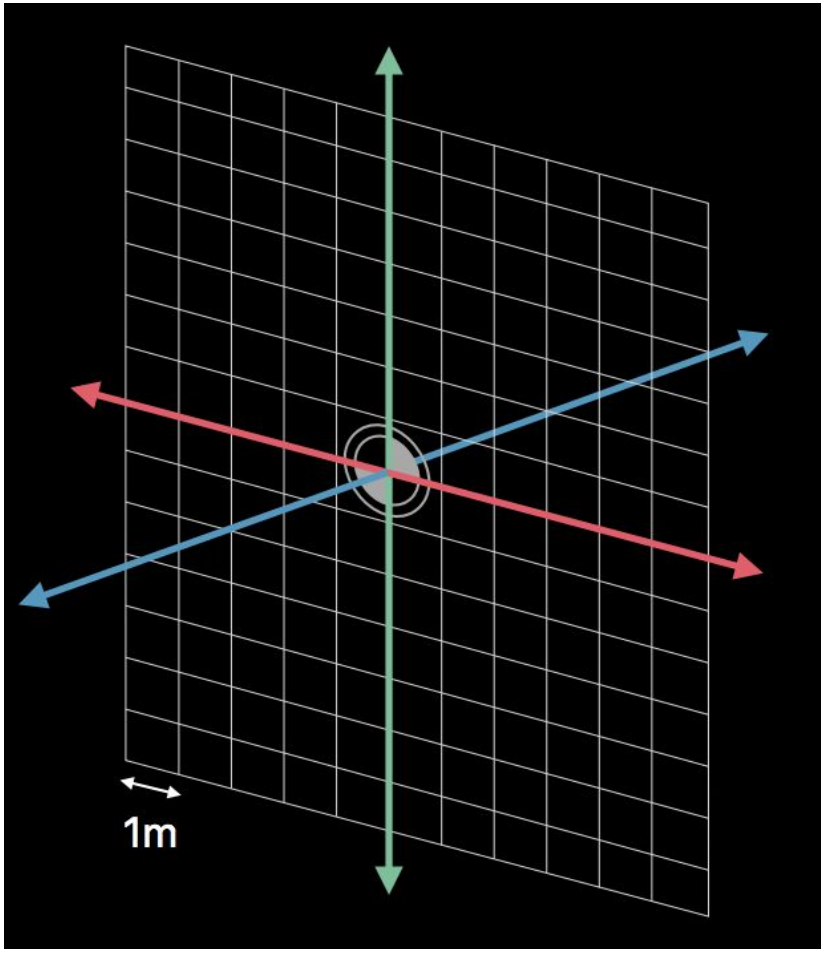
ViewController.swift

- **Controls setting up the Session Configuration for the ARSCNView**
- **Runs the ARKit Session**
- The Brain of the Operation
- Populates data on 3D models



Tracking

- That's all you need!
- Pretty boring



Scene Understanding

- Light Intensity
- Average Color
- Plane Detection

https://holographica.space/wp-content/uploads/602_introducing_arkit_augmented_reality_for_ios.pdf

ARSCNViewDelegate

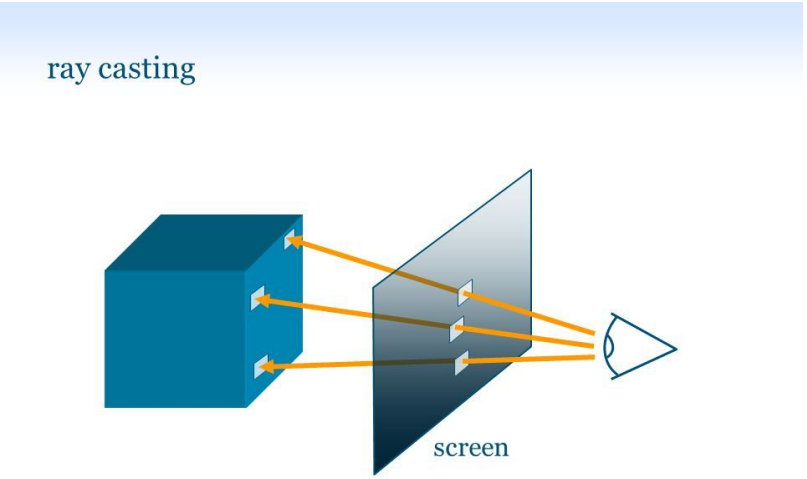
- renderer(_ renderer:, updateAtTime time:)
- renderer(_ renderer:, didAdd node:, for anchor:)
- renderer(_ renderer:, didUpdate node:, for anchor:)

ARSession Delegate

- session(_ session:, cameraDidChangeTrackingState camera:)
- session(_ session:, didFailWithError error:)
- sessionWasInterrupted(_ session:)
- sessionInterruptionEnded(_ session:)

Hit Testing (Ray Casting)

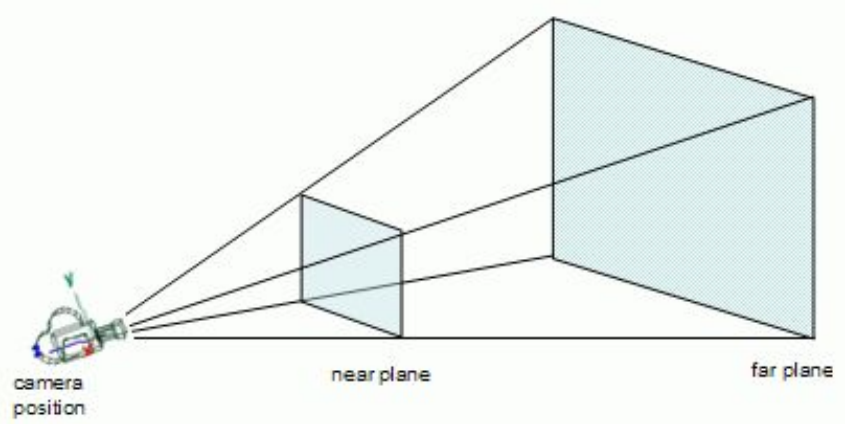
- Intersect ray with real world
- Using Scene info
- Ordered by Distance
- Hit-Test types



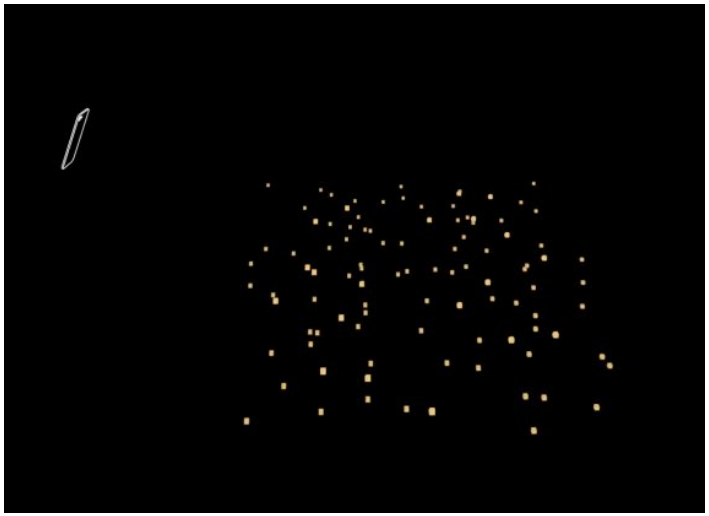
http://images.slideplayer.com/18/6096431/slides/slide_83.jpg

VirtualObjectARView.swift

- Manages the ARView for our app with a little extra sauce
- Manages all of the Hit Testing that we do into the world.



http://www.lighthouse3d.com/wp-content/uploads/2011/04/vf.gif



https://holographica.space/wp-content/uploads/602_introducing_arkit_augmented_reality_for_ios.pdf

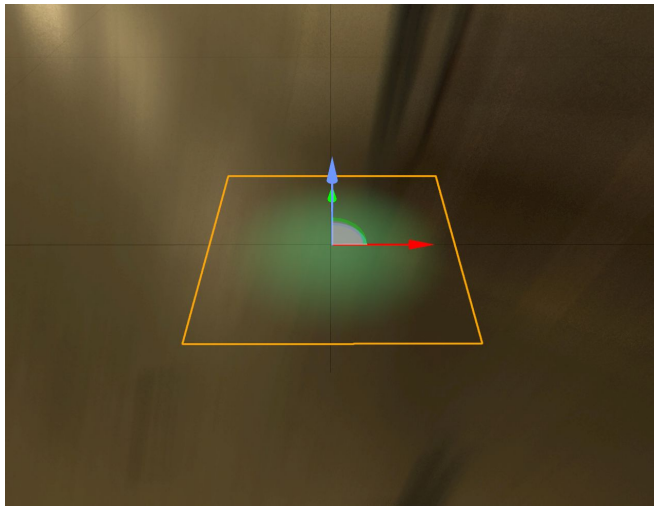
Rendering Engine - SceneKit

FocusSquare.swift

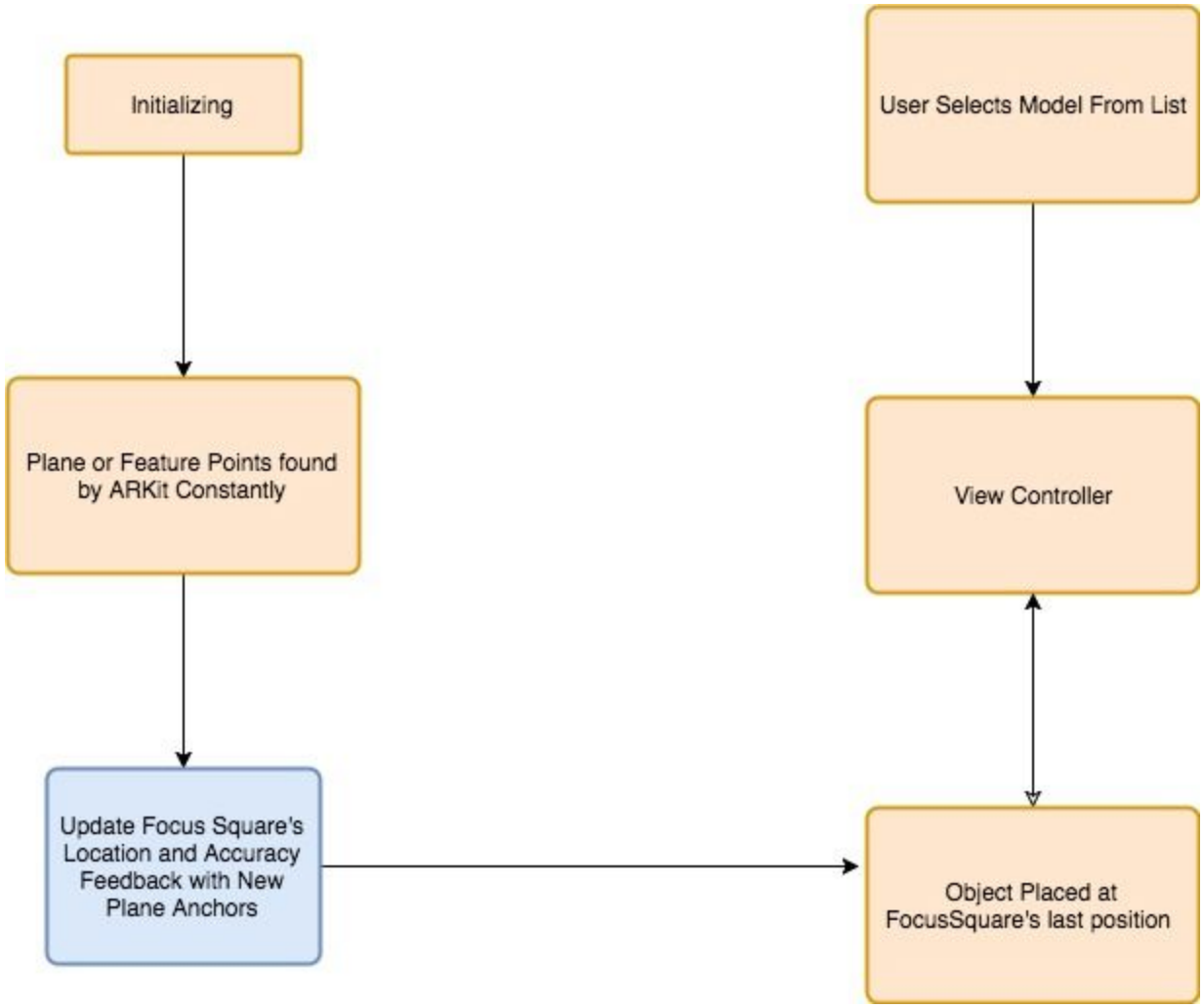
- Keeps track of last known valid positions
- Gives the user feedback of where they can place objects

Uses Resource: FocusSquare.scn

- Normal, Plane Object
  - Surface Shader causes the glow (Extra credit)



General App Flow



VirtualObject.swift

- Our “Content”
- Wrapper around an SCNReferenceNode Object
- Extends and enhances how stable the placed content is in our experience

http://compass.xboxlive.com/assets/0f/eb/0feb7a56-f8c1-45e1-a412-a7edf8141f4e.jpg?n=Microsoft_Hololens_1200x630.jpg

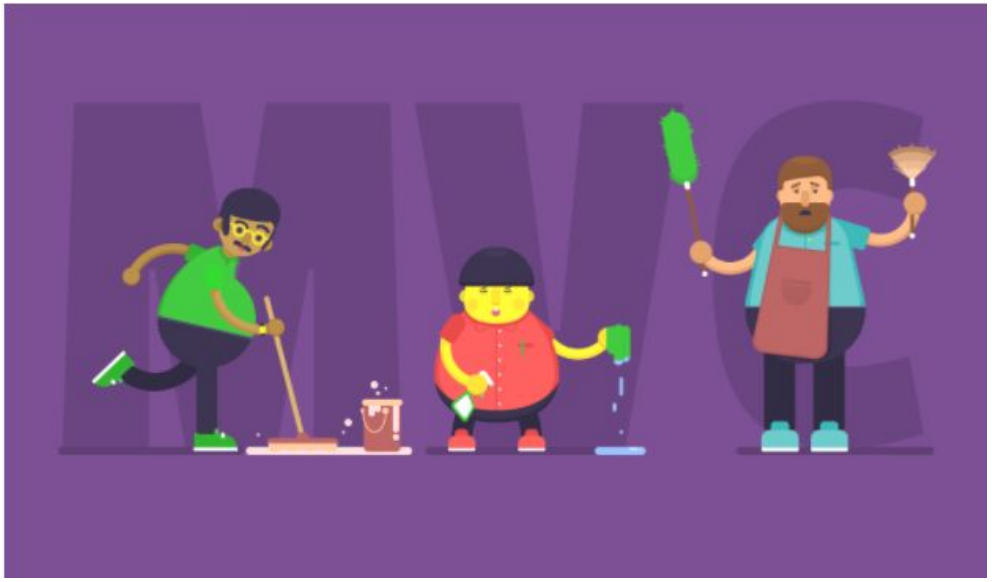
VirtualObjectLoader.swift

- Pretty small
- Handles the loading of 3D content into our application
- Does it on a separate thread to avoid “loading lag”

http://www.jasonyormark.com/2017/05/01/5-things-ive-learned-about-being-a-great-manager/

ViewController.swift

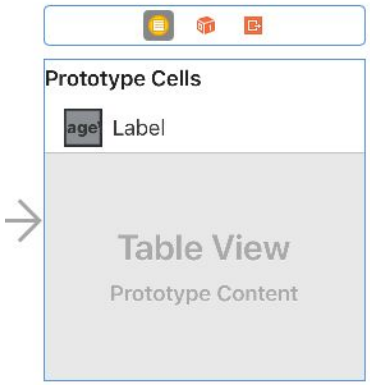
- **Populates data on 3D models**
- The Brain of the Operation
- Controls setting up the Session Configuration for the ARSCNView
- Runs the ARKit Session



UIManagement

- 2 Classes/Files
  - StatusViewController.swift
  - VirtualObjectSelection.swift

- Not important to understand how it works right now
- Controls our reset button, feedback label, and object selection table



In Main.storyboard

Questions?

https://ruben.verborgh.org/images/blog/boring.jpg

Resources

Master Yoda(https://sketchfab.com/models/9ada9c6edc1c4509bb413b903c0824f4) by Adam Beamish(https://sketchfab.com/earthen) is licensed under CC Attribution-NonCommercial(http://creativecommons.org/licenses/by-nc/4.0/)

https://sketchfab.com/models/ad73cea7bc2049f7975e6e1a3296be5d#

Crow toy(https://sketchfab.com/models/c6efada51a1d4721b4bee0bdaabdc276#)

developer.apple.com


wayfair®