iOS and Android development with Unity3D

Ralph Barbagallo
Founder, FLARB LLC

Who am I?

The Platform Problem

Why bother with Android?

The Android Customer

Have you ever paid to download an app on your phone?
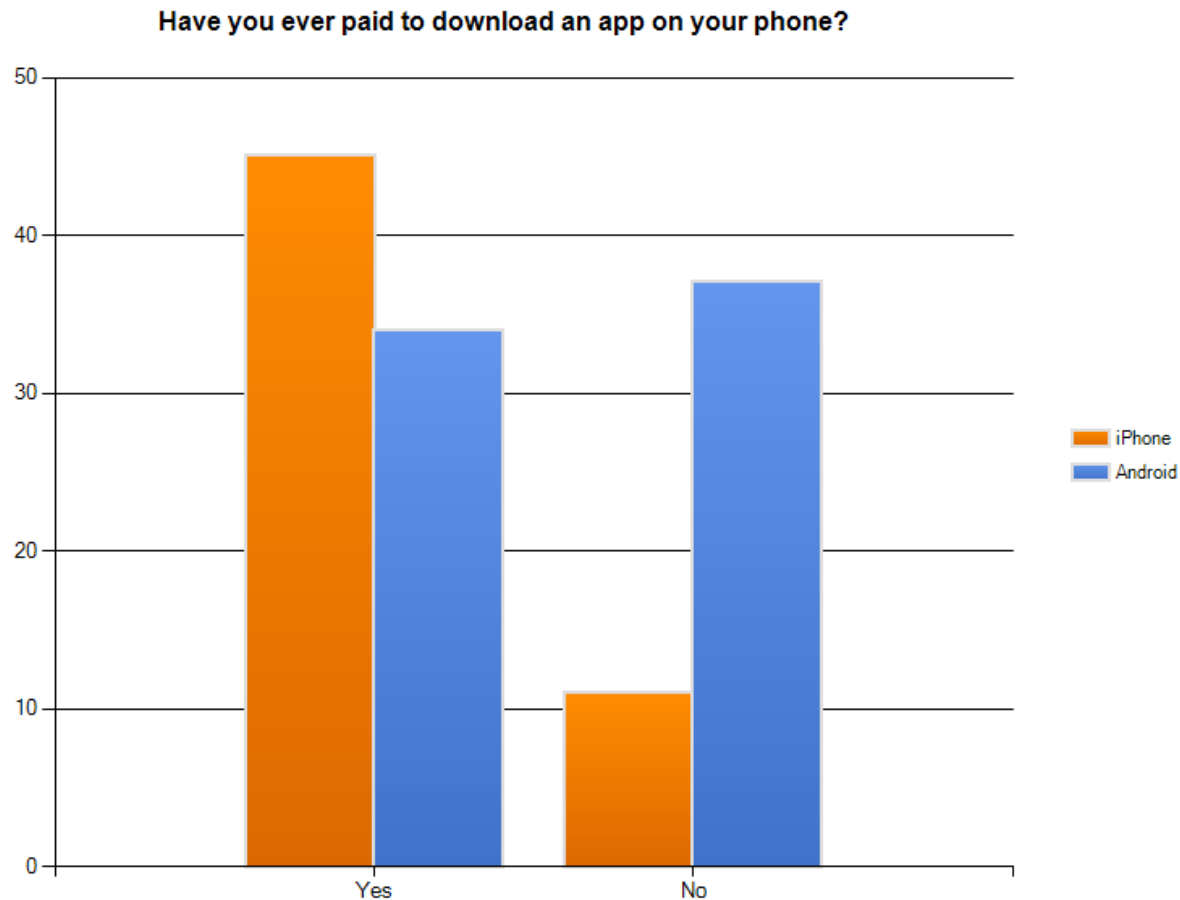
Future Platform Shifts

Choosing A Solution

- Most Platforms With Least Code

- Licensing Model That Won't Eat Margins

- Broad Community Support

HTML5

- HTML5 Frameworks
  - PhoneGap
  - GameSalad
  - Game Closure, AppMobi, etc.

- HTML5 Still Sucks
  - Horrible performance
  - Gated by mobile browsers

Cocos2d

- Cocos2d-X
  - Straight C++ version of Cocos2d
  - Great 2D library, widely used
  - Amazing updates and support
  - No web support

Unreal Developer's Kit

- UDK
  - Don't need a Mac!
  - Much better renderer (As of Unity 3.5)
  - Platform-specific features from Epic only
  - 25% of your revenue to Epic

Cross Platform Options: Misc.

- Others
  - Titanium Studio
  - Corona
  - Flash

Why I Chose Unity3D: Tech

- Best support for critical platforms
  - Mobile (iOS, Android)
  - Web (NaCL, Flash, Web Player)
  - Desktop (Steam, Mac App Store)
  - Console

Tech

- Native Plug-ins
  - Can support any platform-specific functionality
    - Amazon App Store
    - OpenFeint
    - Write your own plug-in for whatever you need

Support

- Awesome forums
- Awesome feedback from support
- Twitter!

Asset Store

Business Model

- Business Model
  - Flat fee per-seat
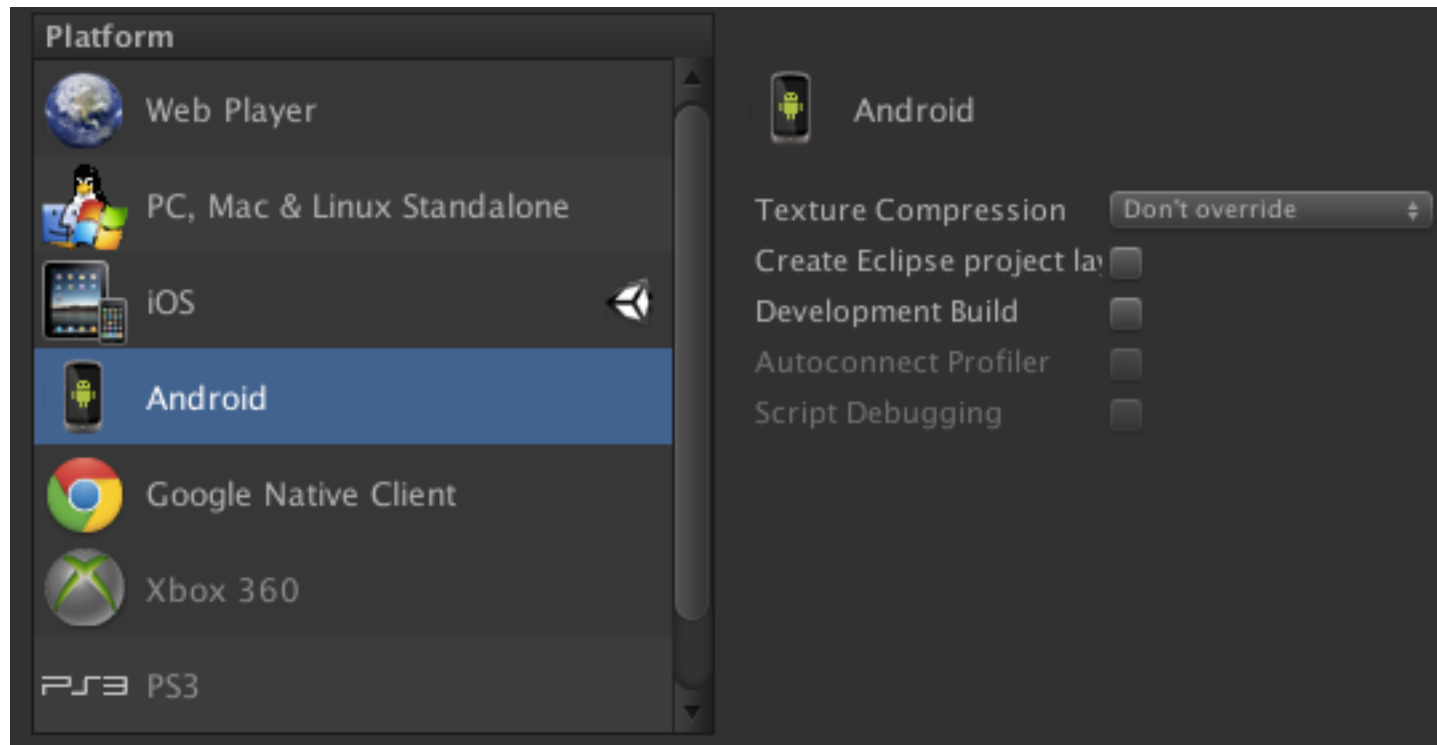 - No rev-share

Brick Buddies

Camera Birds
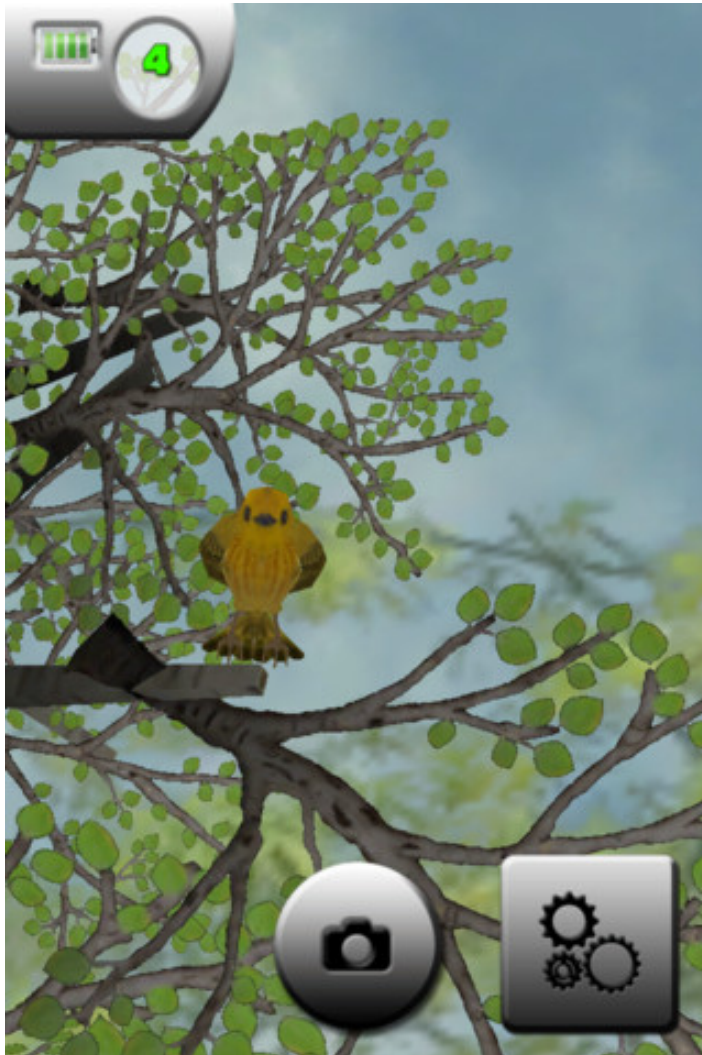
iOS to Android with Unity3D

Plug-ins

- Used mostly cross-platform plug-ins

- Access platform specific features (Game Center etc.)

prime
[31]

Platform Specific Code

- Only spent a few days of re-factoring

- Swapped out iOS plug-ins for Android

- Combination of runtime platform checks and #IF compiler directives

- AndroidJavaClass!

AndroidJavaClass

```
AndroidJavaObject activity = new
AndroidJavaClass("com.unity3d.player.UnityPlayer").GetStatic<And
roidJavaObject>("currentActivity")


mWindowManager =
activity.Call<AndroidJavaObject>("getSystemService","window");


mConfig =
activity.Call<AndroidJavaObject>("getResources").Call<AndroidJav
aObject>("getConfiguration");
```

Screen Sizes

- iOS has 3 aspect ratios to deal with

- Made interface work in "safe zone"

- For Android's common 16:10 ratio, scaled NGUI interface to fit

Performance and Testing

android.hardware.touchscreen
android.hardware.touchscreen.multitouch
android.hardware.wifi

This application is available to over 1125 devices.   Show devices
1 device is manually filtered.

Unity's Android Test Devices

Unity at SIGGRAPH 2012:
- Nexus One (Adreno 205)
- Samsung Galaxy S 2 (Mali 400)
- Nexus S / Galaxy Nexus (SGX 540)
- Motorola Xoom (Tegra2)

Cross Platform Export Tools

- Per-Platform Asset Settings
  - Compression settings
  - Filtering

- Cache Server

- Multi-Platform Toolkit
  - Platform-specific assets
  - Build-time asset changes

In Summary

- Best Business Model

- Widest Platform Support

- Best Community Support

- Dead Simple Porting Process

Thanks

Email: ralph@flarb.com

Twitter: @flarb

Blog: www.ralphbarbagallo.com

Site: www.flarb.com