



Mastering Xcode for iPhone OS Development

Part 2

Marc Verstaen

Sr. Manager, iPhone Tools

Tale of Two Sessions

- Part 1: Orientation: Tour of complete development cycle
- Part 2: Mastery: Details of several critical areas

It's So Yesterday

Develop

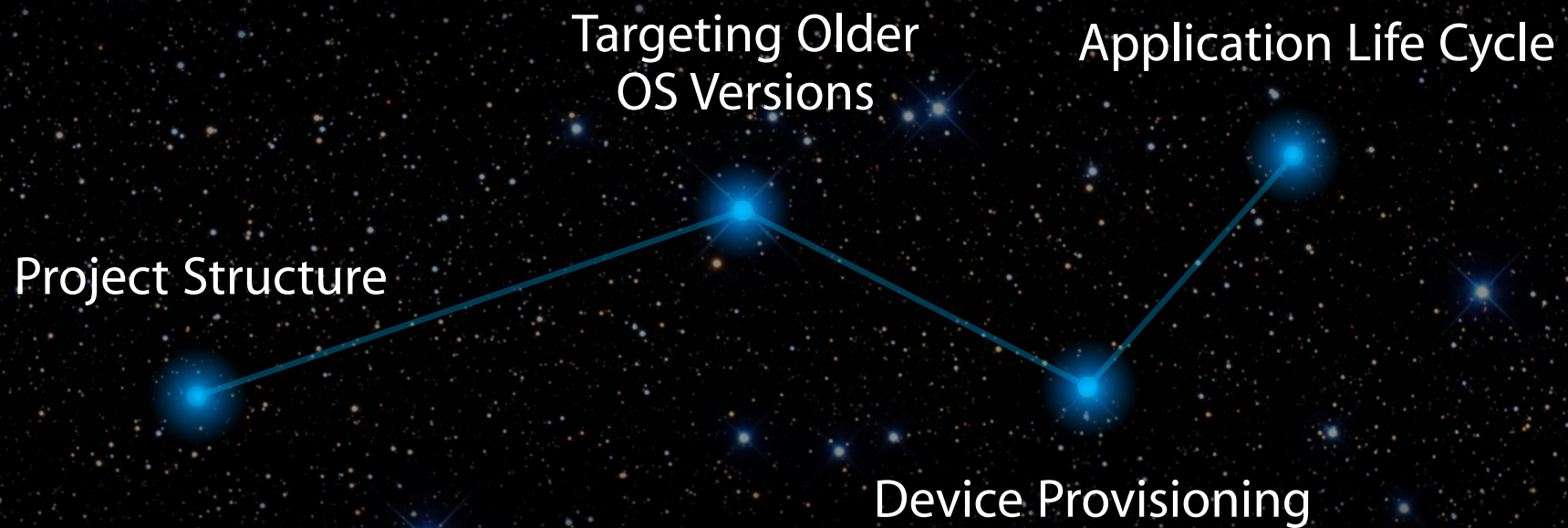
Provision

Distribute

Build



Agenda



Project Structure

Jacob Xiao
iPhone Tools Engineer

Project Structure



Targeting Older
OS Versions

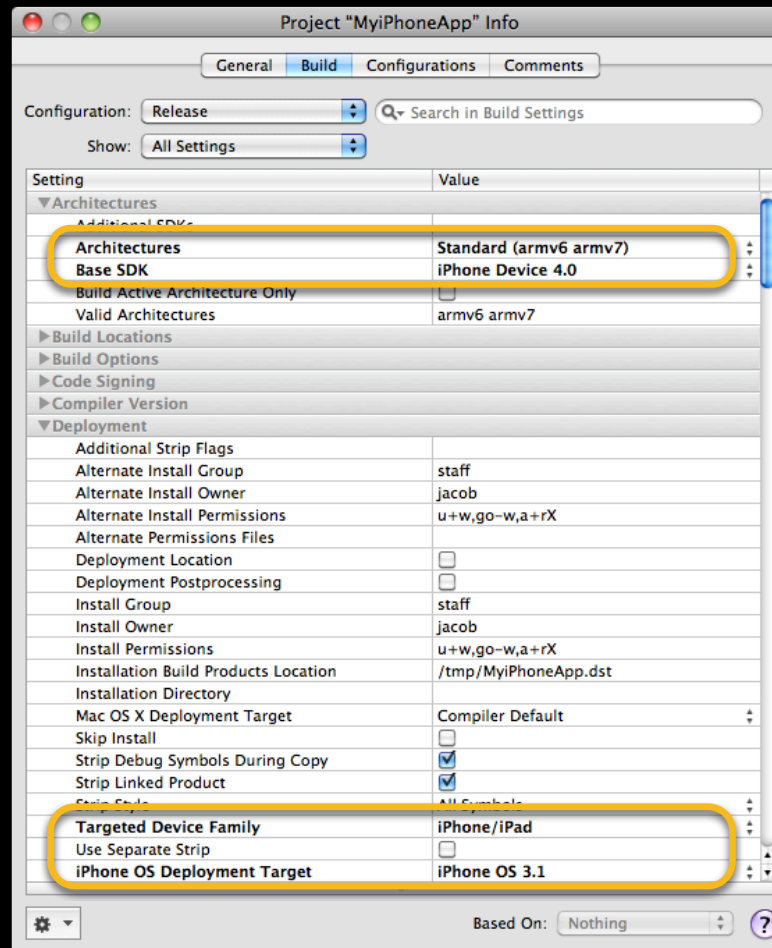
Application Life Cycle

Device Provisioning

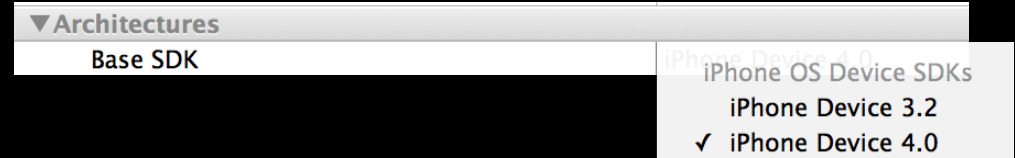
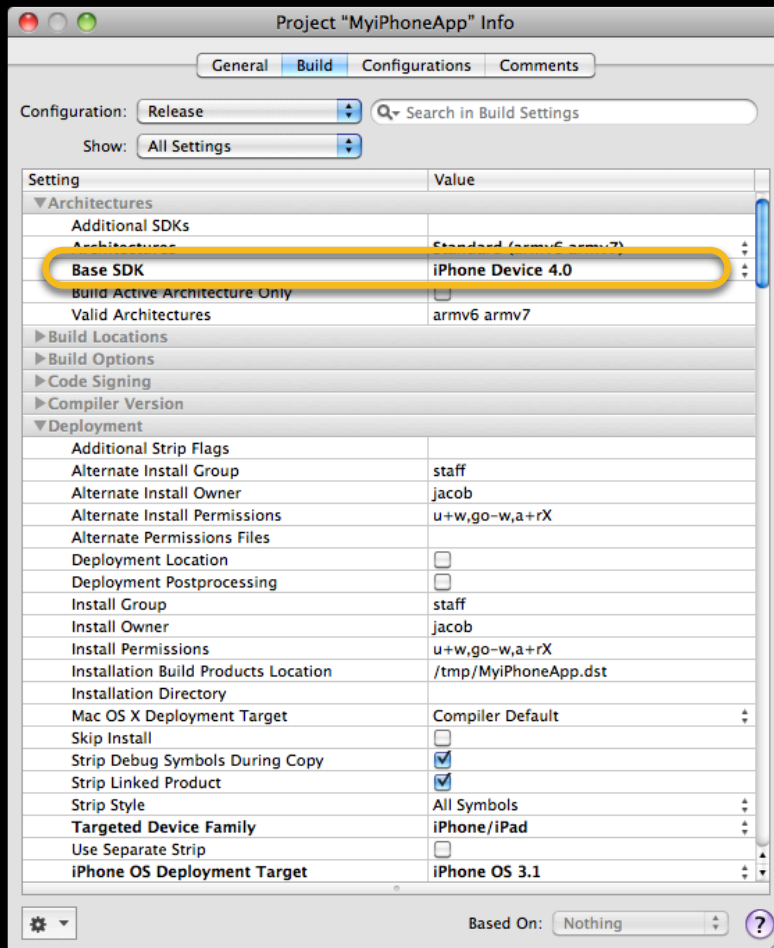
Project Structure

- Important project settings
- Bringing your app to iPhone and iPad

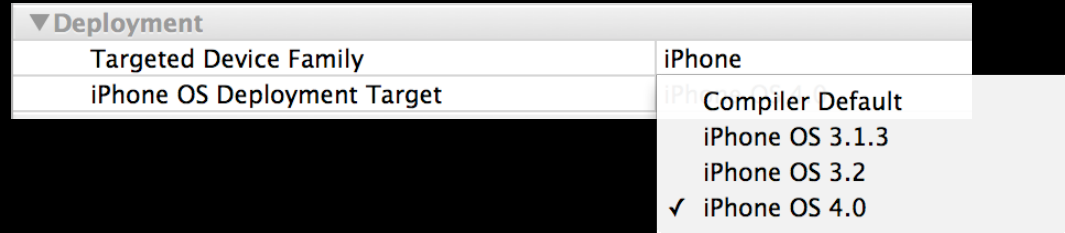
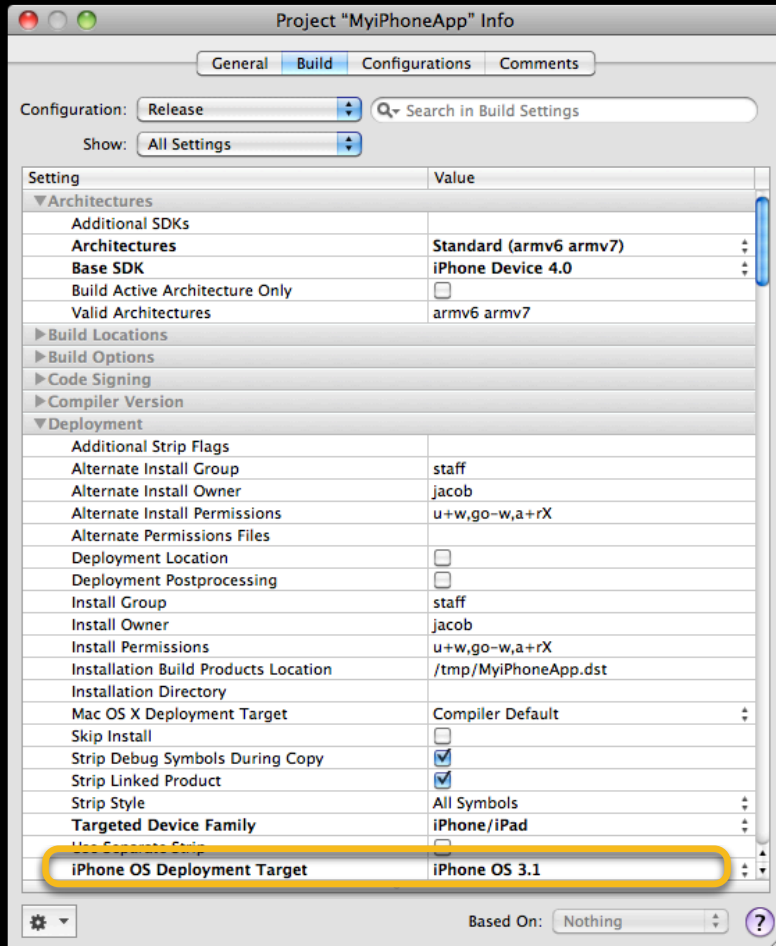
Project Settings



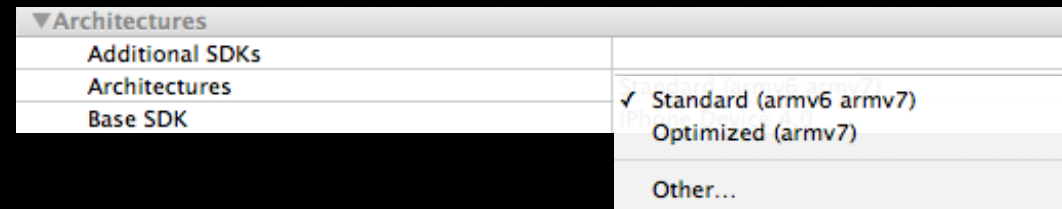
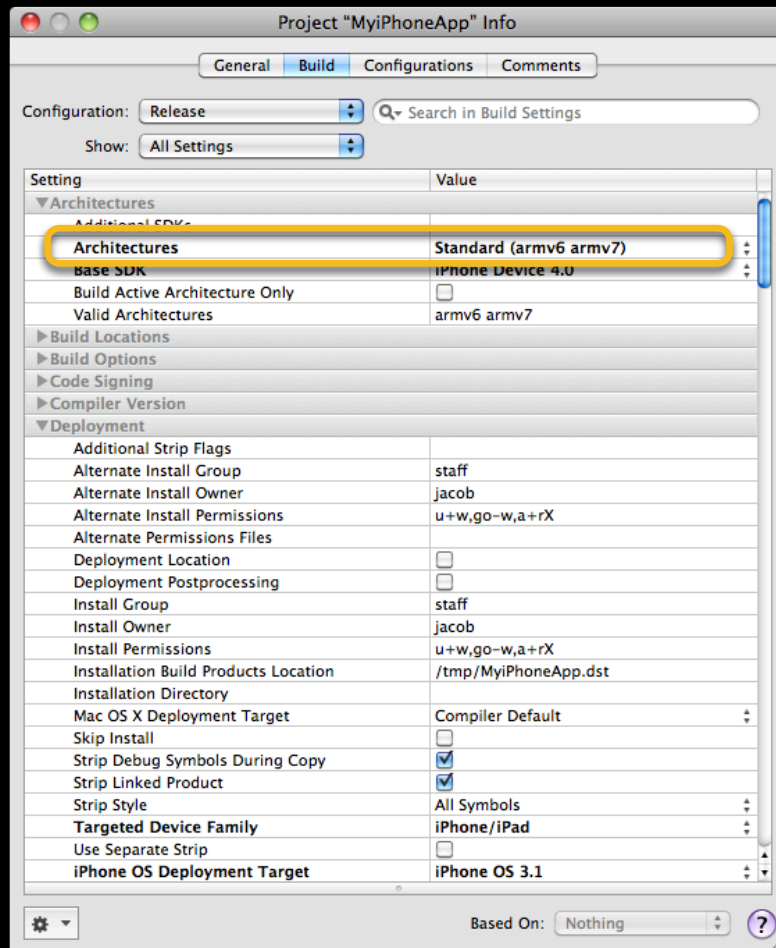
Base SDK



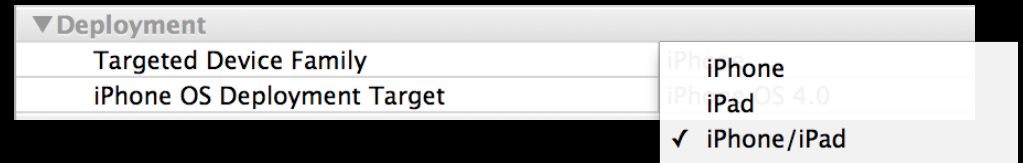
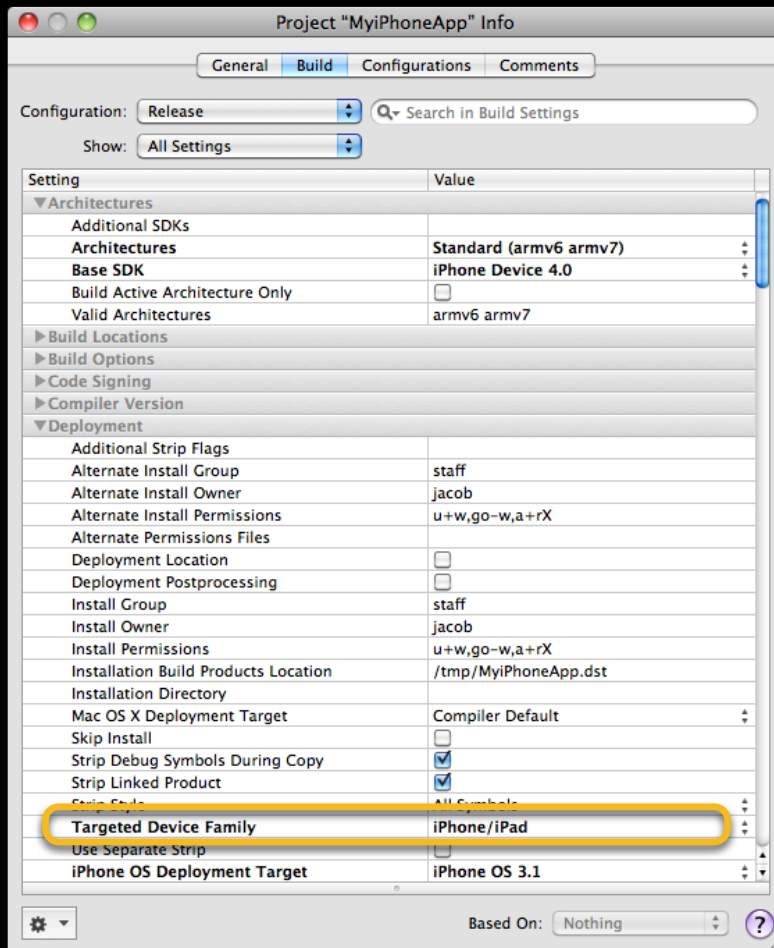
Deployment Target



Architectures



Targeted Device Family



Example App

- Base SDK: 4.0
- Deployment Target: 3.1
- Architectures: Standard
- Targeted Device Family: iPhone/iPad

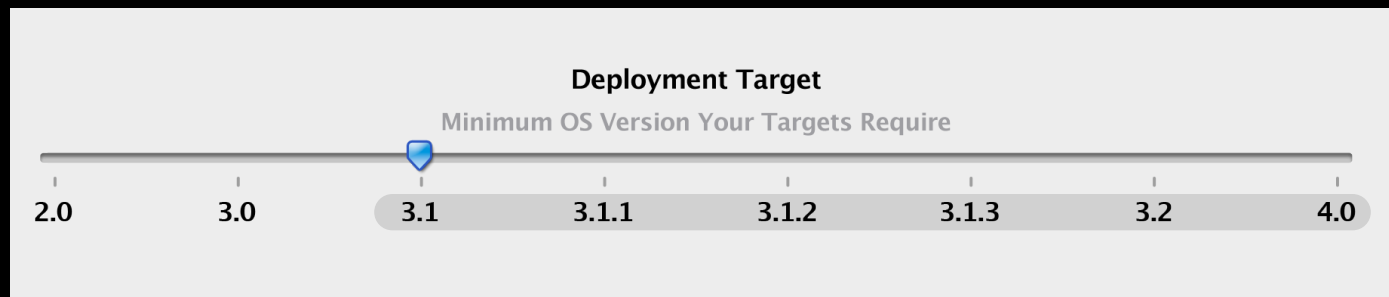
Example App

- Base SDK: 4.0
- Deployment Target: 3.1
- Architectures: Standard
- Targeted Device Family: iPhone/iPad



Example App

- Base SDK: 4.0
- Deployment Target: 3.1
- Architectures: Standard
- Targeted Device Family: iPhone/iPad



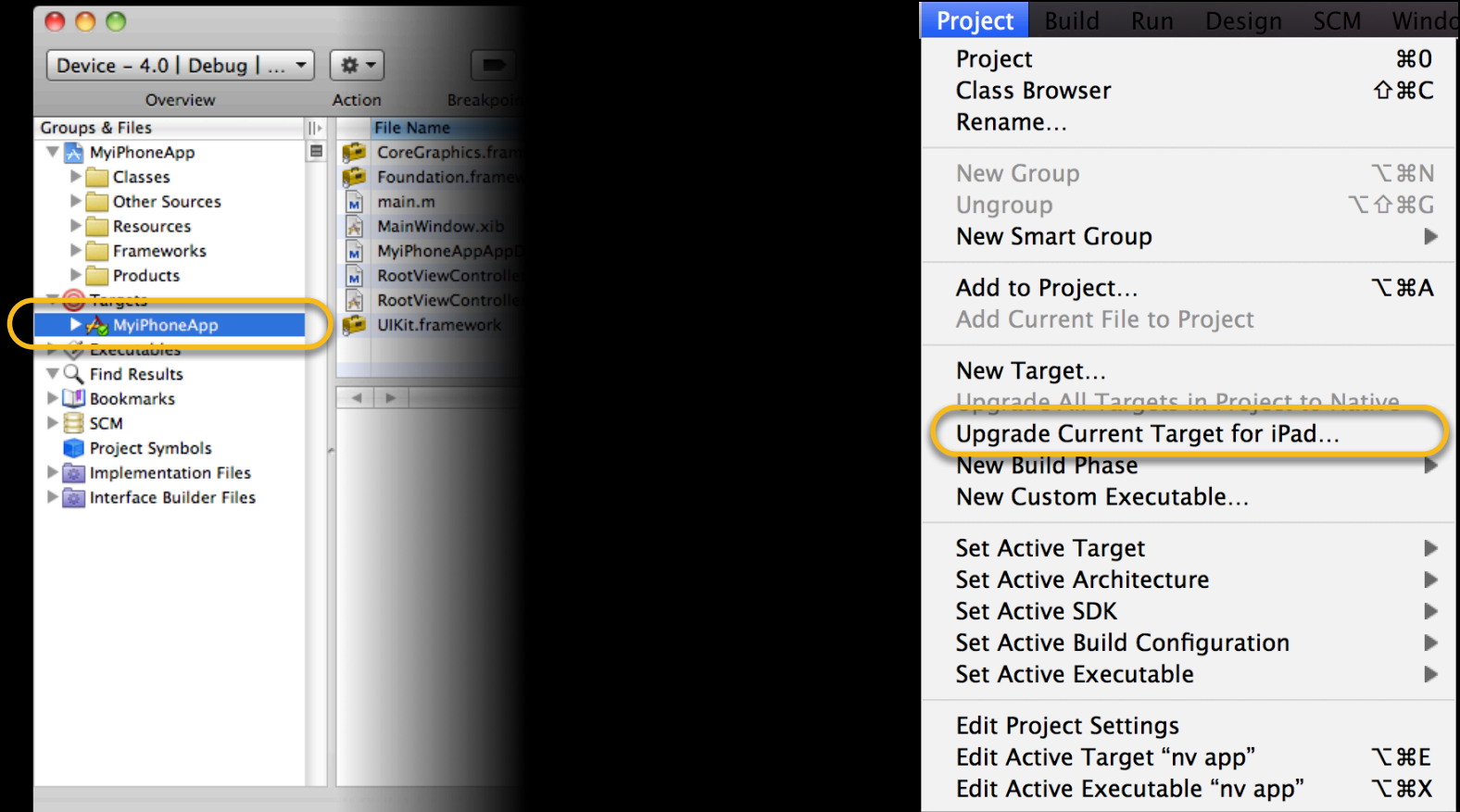
Universal Apps



Multiple Apps



Upgrading a Project to iPad

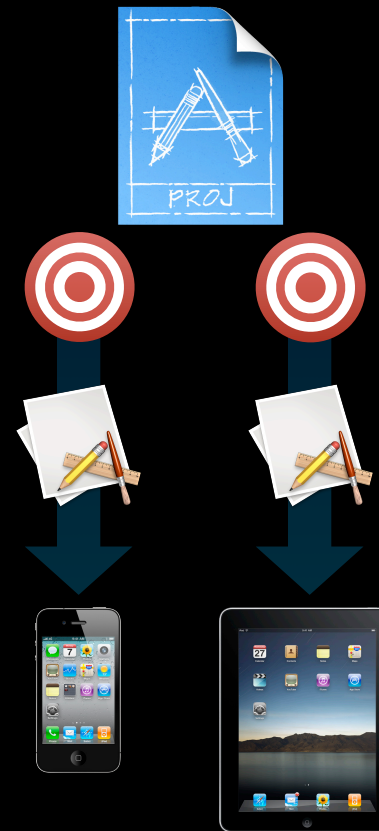


What's the Difference?

Universal



Multiple Apps



Demo

Structuring Your Project

Summary

- Use the latest base SDK
- Set Deployment Target to earliest OS you want to support
- Use “Standard” Architectures
- Use Targeted Device Family to choose allowed devices
- Choose Universal or Multiple Apps

Targeting Multiple iPhone OS Versions

Clark Cox
iPhone Tools Engineer

Project Structure

Targeting Older
OS Versions

Application Life Cycle

Device Provisioning

Use New Classes, Selectors, and Symbols

iPhone



iPod touch



iPad



Classes

```
Class myClass = NSStringFromClass(@"UILocalNotification");  
if(myClass) {  
    UILocalNotification *alarm = [[myClass$Notification alloc] init];  
    ...  
    [app scheduleLocalNotification: alarm];  
    [alarm release];  
}
```

Methods

```
UIDevice *device = [UIDevice currentDevice];  
BOOL multitaskingSupported = NO;  
if([device respondsToSelector: @selector(isMultitaskingSupported)]) {  
    multitaskingSupported = [device isMultitaskingSupported];  
}
```

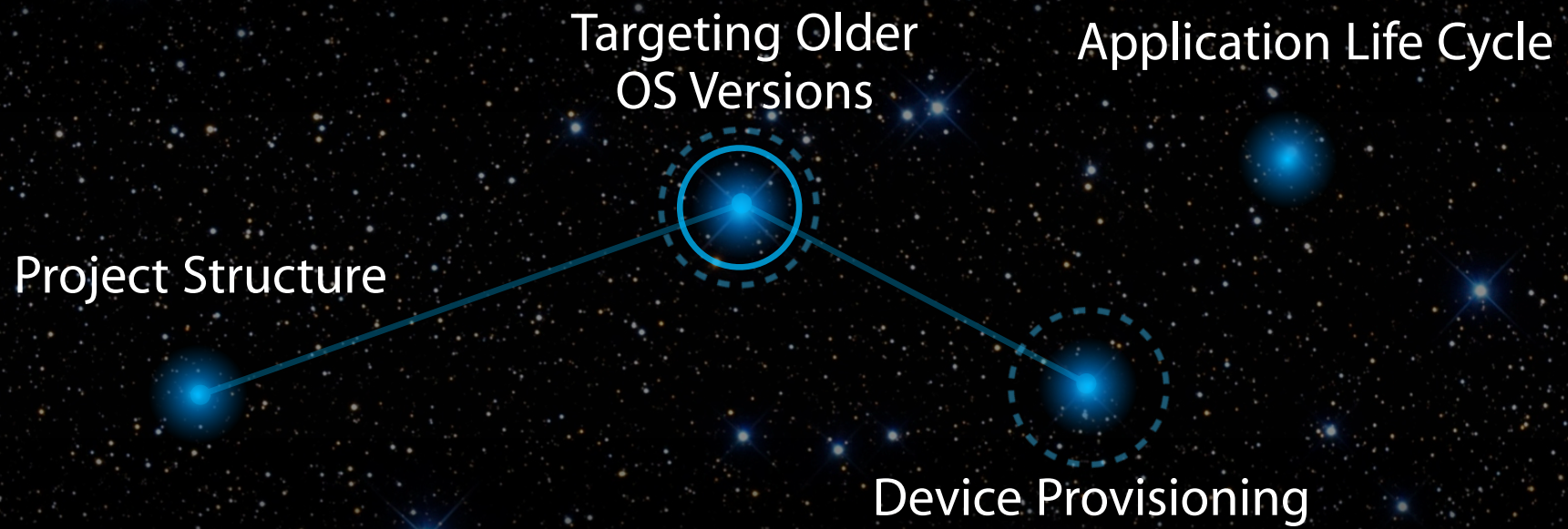
Functions

```
if(&UIGraphicsBeginPDFContextToFile != NULL) {  
    if(UIGraphicsBeginPDFContextToFile(...)) {  
        ...  
        UIGraphicsEndPDFContext();  
    }  
}
```

Constant Symbols

```
- (void)handleNotification:(NSNotification *)notification {  
    if(&UIApplicationDidEnterBackgroundNotification != NULL) {  
        NSString *name = [notification name];  
        if([name isEqualToString: UIApplicationDidEnterBackgroundNotification]) {  
            ...  
        }  
    }  
}
```

Always test on a device!



Device Provisioning



- Automatic and custom provisioning profile
- Distribution provisioning profiles
- Developer profile (introduced in 3.2)

Automatic Provisioning Profile

- Create a CSR if required
- Create a team provisioning profile
- Download provisioning profiles



Custom Provisioning Profiles

- In App Purchase
- Push Notifications
- Game Center
- Keychain data sharing
- etc.



Developer Profile



Developer Profile



Developer Profile

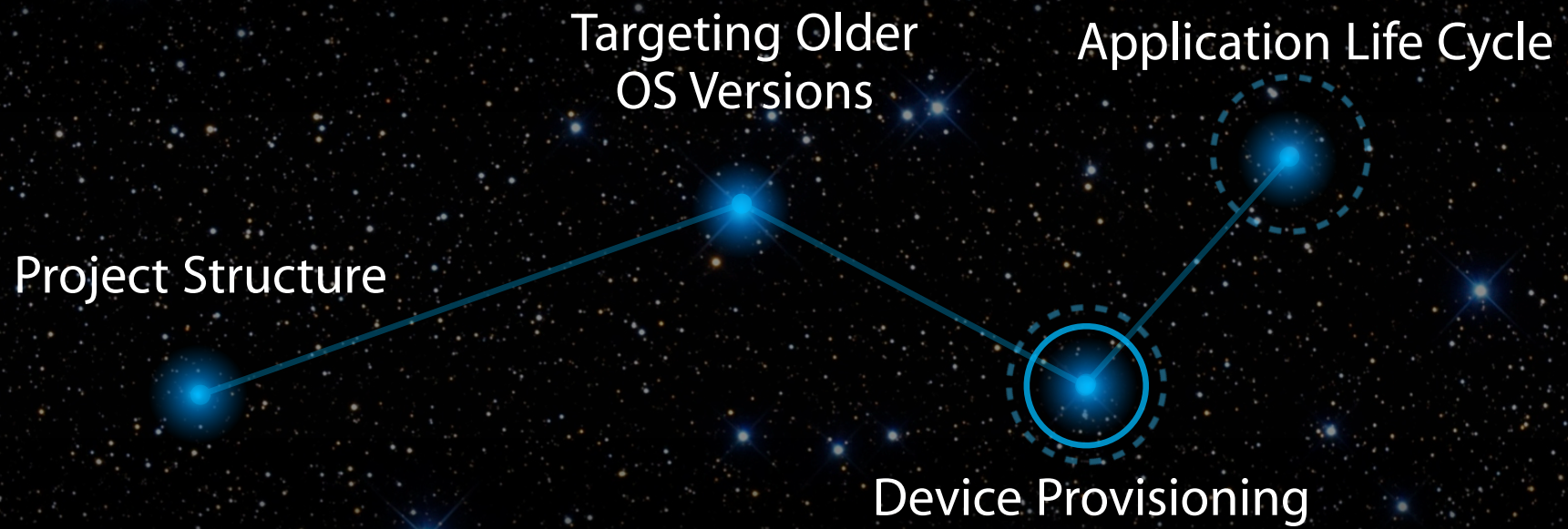


Demo

Automatic Provisioning Profile

Application Life Cycle

Chris Skogen
iPhone Tools Engineer



Application Life Cycle

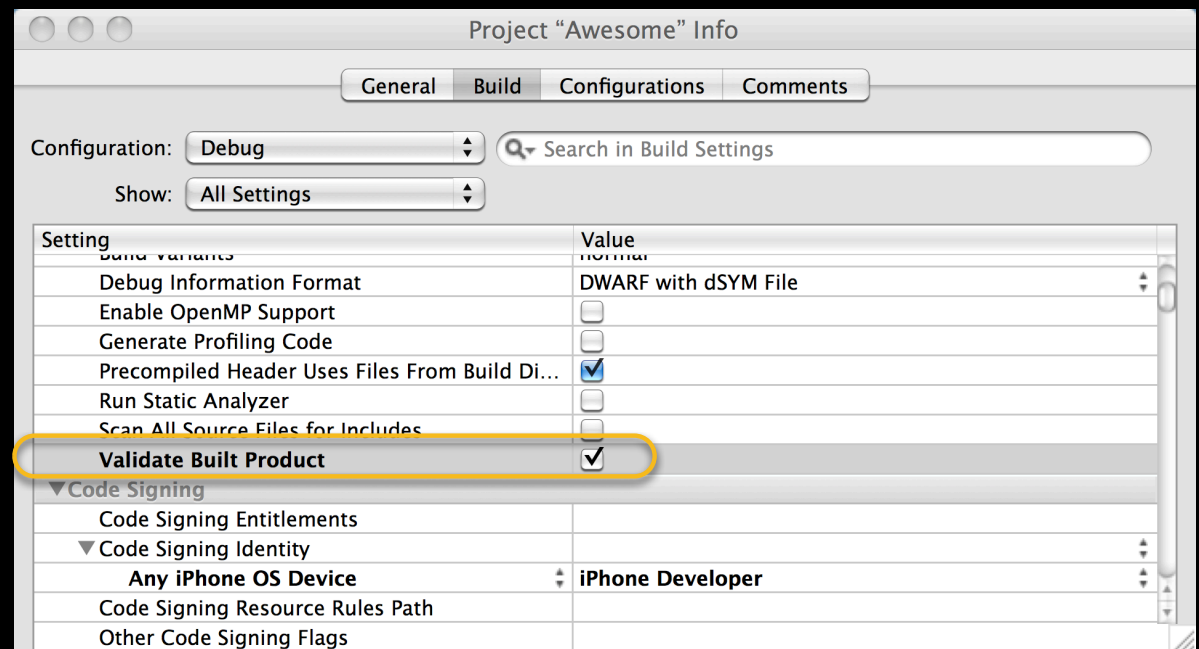


- Build time validation
- Build and archive
- Online validation
- Submit to the store
- Use crash logs

Build Time Validation



- It's a build setting
- Do it all the time



Build Time Validation



- Check icons
- Check plist
- Check Bundle ID



Archived Application

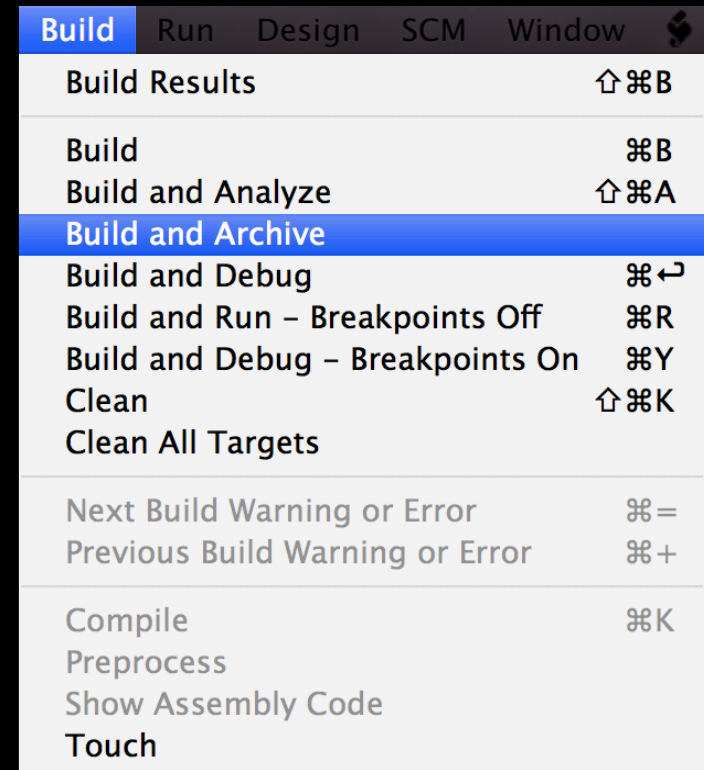
New
3.2



Archived Applications



- Build > Build and Archive
- Preserves app build and associated data
- Source for app distribution
- Needed throughout app life cycle



Online Validation



- Application needs to be registered with iTunes Connect
- Distribution profile required
- Runs all automated App Store review tests

Validation Results

- Detects many different types of errors
 - Bundle version needs to be increased
 - Missing icon sizes
 - Incorrect bundle identifier
 - Code signing
 - Entitlements
 - And many more
- Suggests fixes

Uploading to the Store



- Same process as online validation
- Submit marketing info through iTunes Connect

Crash Reports

- Obtain from
 - iTunes Connect
 - Direct from app users
- Symbolicate in Xcode
 - Requires .dSYM file matching app
 - Archived application preserves for you
 - Drop crash log in Organizer

Demo

App Distribution

Conclusion

Marc Verstaen

Sr. Manager, iPhone Tools

Summary

- Take full advantage of Xcode
 - Structuring your project
 - Testing your app
 - Managing your app's entire life cycle

More Information

Michael Jurewitz

Developer Tools Evangelist

jurewitz@apple.com

iPhone Dev Center

iPhone Development Documentation and Resources

<http://developer.apple.com/iphone>

Developer Forum

iPhone Developer Tools Forum

<https://devforums.apple.com/community/iphone/devtools>

Apple Developer Forums

<http://devforums.apple.com>

Related Sessions

Mastering Xcode for iPhone OS Development, Part 1	Mission Tuesday 2:00PM
Managing Mobile Devices	Nob Hill Tuesday 3:15PM
What's New in Instruments	Presidio Wednesday 11:30AM
Automating User Interface Testing with Instruments	Marina Wednesday 2:00PM
Designing Apps with Interface Builder	Mission Wednesday 2:00PM
Introducing Xcode 4	Mission Wednesday 3:15PM
App Publishing with iTunes Connect	Mission Thursday 10:15AM

Labs

Xcode for iPhone Development Lab	Developer Tools Lab B Wednesday 2:00PM
Automated User Interface Testing Lab	Developer Tools Lab A Wednesday 4:30PM
Interface Builder Lab	Developer Tools Lab B Thursday 9:00AM
Xcode 4 Lab	Developer Tools Lab A Wednesday 9:00AM
iPhone OS Performance Lab	Developer Tools Lab A Thursday 4:30PM

Q&A



The last slide
after the logo is
intentionally left
blank for all
presentations.