



iPad Development Overview

Chris Parker
UIKit Frameworks Engineer



iPhone to iPad

New UI considerations

- Initial appearance
- Rotation
- Transitions
- Information density and presentation
 - UIPopoverController
 - UISplitViewController

Universal Applications

Writing for multiple SDKs

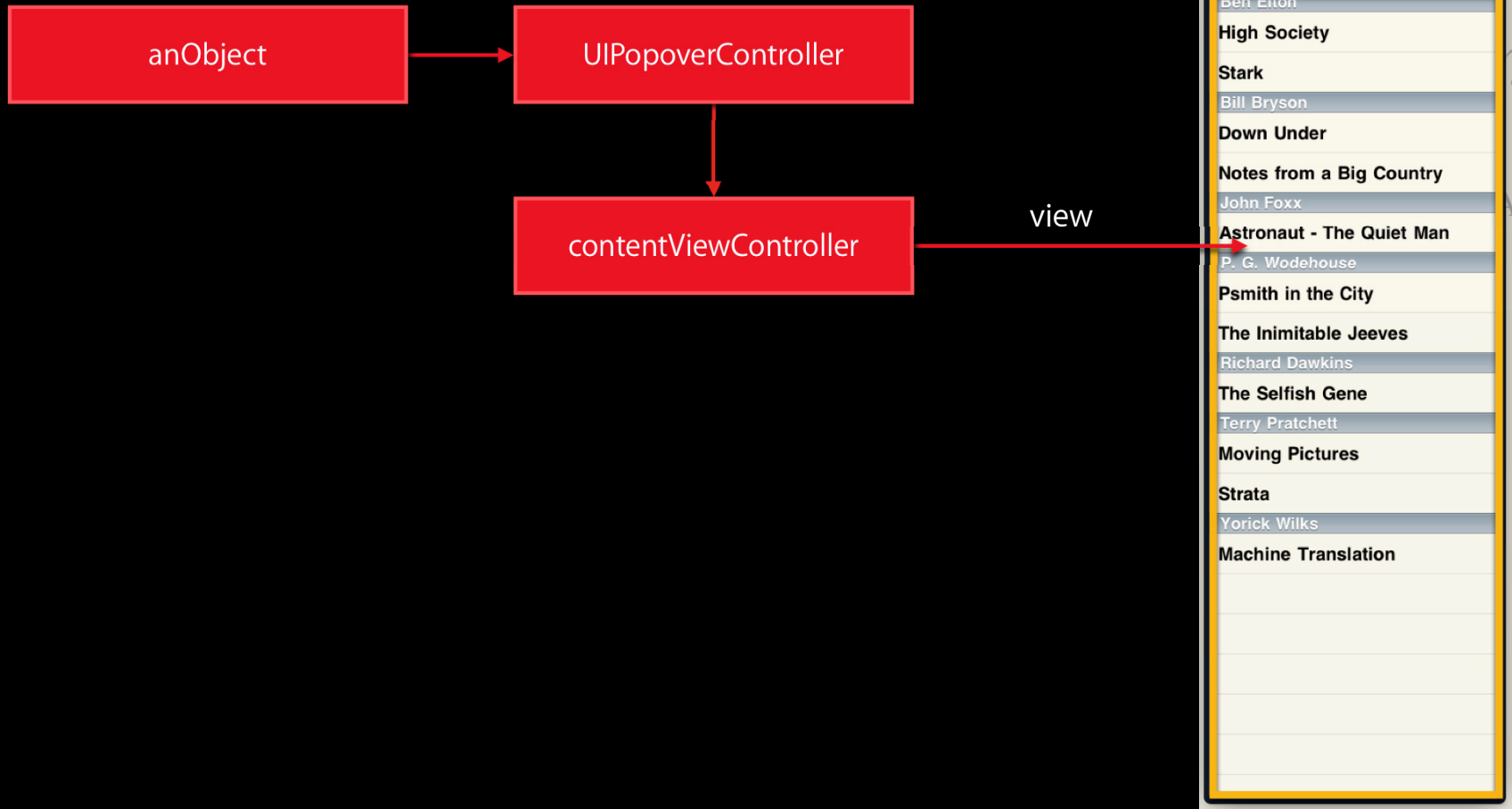
- Xcode
- API
- Symbol availability

What Needs to Change?

- Xcode project settings
 - Base SDK
 - Deployment target
 - Targeted device family
- Refactored code
 - Application delegate(s)
 - Action sending controllers
- Resources
 - Nibs
 - Some images

Popovers and Split View Controllers

UIPopoverController



UIPopoverController

Creation and properties

- Creation

```
- (id)initWithContentViewController:(UIViewController *)vc;
```

- Properties

```
@property (nonatomic, retain) UIViewController *contentViewController;  
@property (nonatomic) CGSize popoverContentSize;  
@property (nonatomic, copy) NSArray *passthroughViews;
```


UIPopoverController

Presentation and dismissal

- Presenting

- (void)presentPopoverFromRect:(CGRect)rect
 inView:(UIView *)view
 permittedArrowDirections:(UIPopoverArrowDirection)directions
 animated:(BOOL)animated;
- (void)presentPopoverFromBarButtonItem:(UIBarButtonItem *)item
 permittedArrowDirections:(UIPopoverArrowDirection)dir
 animated:(BOOL)animated;

- Dismissing

- (void)dismissPopoverAnimated:(BOOL)animated;

Demo: Popovers

UISplitViewController

UISplitViewController

view



UISplitViewController



UISplitViewController

Creation and properties

```
UISplitViewController *svc = [[UISplitViewController alloc] init];
```

```
@property(n nonatomic, copy) NSArray *viewControllers;  
@property(n nonatomic, assign) id <UISplitViewControllerDelegate> delegate;
```

UISplitViewController

Delegate methods

- (void)splitViewController:(UISplitViewController*)svc
willHideViewController:(UIViewController *)aViewController
withBarButtonItem:(UIBarButtonItem*)barButtonItem
forPopoverController:(UIPopoverController*)pc;
- (void)splitViewController:(UISplitViewController*)svc
willShowViewController:(UIViewController *)aViewController
invalidatingBarButtonItem:(UIBarButtonItem *)barButtonItem;
- (void)splitViewController:(UISplitViewController*)svc
popoverController:(UIPopoverController*)pc
willPresentViewController:(UIViewController *)aViewController;

Demo: CoreData Books

[http://developer.apple.com/iphone/library/samplecode/CoreDataBooks/
Introduction/Intro.html](http://developer.apple.com/iphone/library/samplecode/CoreDataBooks/Introduction/Intro.html)

Getting Started

Getting Started

Preparing your iPhone project for iPad

- Xcode project modifications
- Code refactoring
- Resources

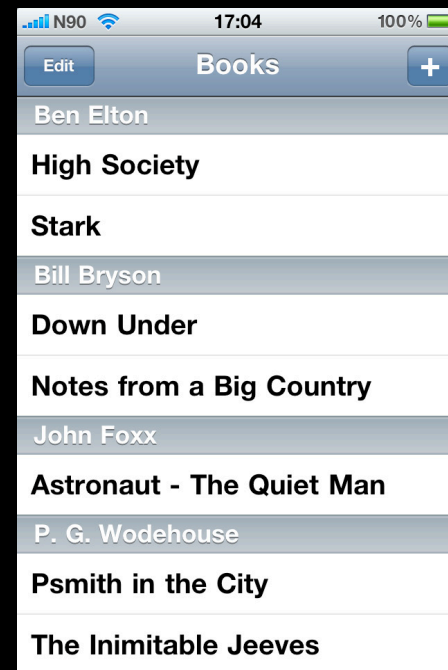
Demo: Converting Your Project

Refactoring

Application delegate

CoreDataBooksAppDelegate

```
managedObjectModel  
managedObjectContext  
persistentStoreCoordinator  
applicationDocumentsDirectory  
window  
navigationController
```



Refactoring

Application delegate

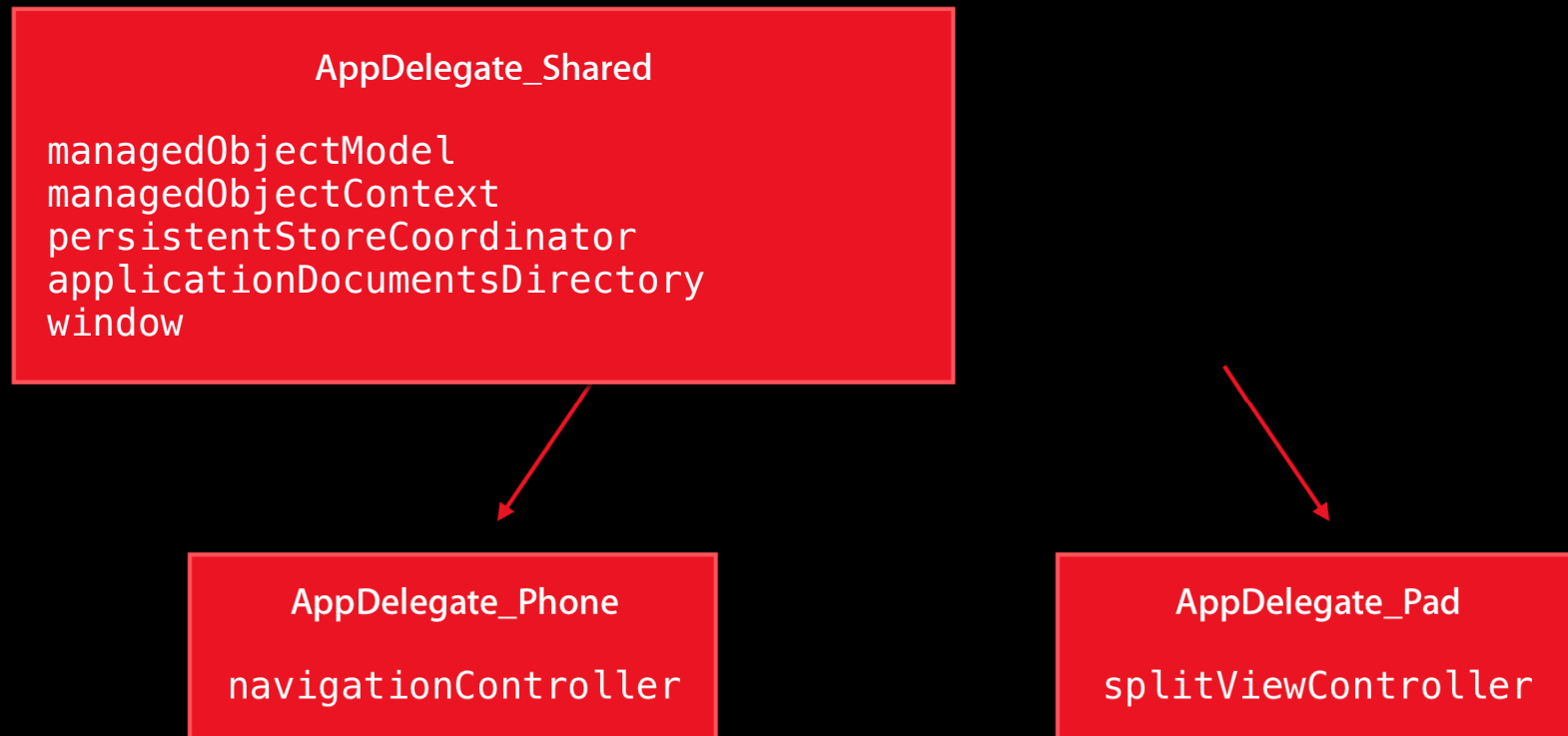
CoreDataBooksAppDelegate

```
managedObjectModel  
managedObjectContext  
persistentStoreCoordinator  
applicationDocumentsDirectory  
window  
navigationController
```



Refactoring

Application delegate



Refactoring

- It's hard to avoid tightly-coupled designs
 - Think about intent
 - Think in terms of actions
- Choose action receivers with care
 - Closest object in your hierarchy
 - ...which knows about all players
 - Responder chain

Refactoring



Refactoring



Demo: Refactoring

What Changed?

- Xcode project settings
 - Base SDK
 - Deployment target
 - Targeted device family
- Refactored code
 - Application delegate(s)
 - Action sending controllers
- Resources
 - Nibs
 - Some images

What Didn't Change?

Model

Model-View-Controller for iPhone OS

Russian Hill
Wednesday 10:15AM

SDKs and Symbols

Symbol Availability

	iPad iPhone OS 3.2	iPhone iOS 4
UIPopoverController	Usable	Creation throws exception
UISplitViewController	Usable	Creation throws exception
UIGestureRecognizer	Usable	Usable

UI Idiom

UIDevice.h

```
typedef enum {
    UIUserInterfaceIdiomPhone, // iPod Touches, also
    UIUserInterfaceIdiomPad,
} UIUserInterfaceIdiom;

#define UI_USER_INTERFACE_IDIOM() ...

if (UI_USER_INTERFACE_IDIOM() == UIUserInterfaceIdiomPad) {
    // Create a popover controller here
    UIPopoverController *popoverController = [[UIPopoverController...
    // etc...
} else {
    // Create an AuthorViewController here and push it
    AuthorViewController *avc = [[AuthorViewController alloc] init];
    avc.author = newAuthor;
    [self.navigationController pushViewController:avc animated:YES];
}
```

UI Idiom

UIDevice.h

```
typedef enum {
    UIUserInterfaceIdiomPhone, // iPod Touches, also
    UIUserInterfaceIdiomPad,
} UIUserInterfaceIdiom;

#define UI_USER_INTERFACE_IDIOM() ...

if (UI_USER_INTERFACE_IDIOM() == UIUserInterfaceIdiomPad) {
    // Create a popover controller here
    UIPopoverController *popoverController = [[UIPopoverController...
    // etc...
} else {
    // Create an AuthorViewController here and push it
    AuthorViewController *avc = [[AuthorViewController alloc] init];
    avc.author = newAuthor;
    [self.navigationController pushViewController:avc animated:YES];
}
```

UI Idiom

UIDevice.h

```
typedef enum {
    UIUserInterfaceIdiomPhone, // iPod Touches, also
    UIUserInterfaceIdiomPad,
} UIUserInterfaceIdiom;

#define UI_USER_INTERFACE_IDIOM() ...

if (UI_USER_INTERFACE_IDIOM() == UIUserInterfaceIdiomPad) {
    // Create a popover controller here
    UIPopoverController *popoverController = [[UIPopoverController...
    // etc...
} else {
    // Create an AuthorViewController here and push it
    AuthorViewController *avc = [[AuthorViewController alloc] init];
    avc.author = newAuthor;
    [self.navigationController pushViewController:avc animated:YES];
}
```


UI Idiom

UIDevice.h

```
typedef enum {
    UIUserInterfaceIdiomPhone, // iPod Touches, also
    UIUserInterfaceIdiomPad,
} UIUserInterfaceIdiom;

#define UI_USER_INTERFACE_IDIOM() ...

if (UI_USER_INTERFACE_IDIOM() == UIUserInterfaceIdiomPad) {
    // Create a popover controller here
    UIPopoverController *popoverController = [[UIPopoverController...
    // etc...
} else {
    // Create an AuthorViewController here and push it
    AuthorViewController *avc = [[AuthorViewController alloc] init];
    avc.author = newAuthor;
    [self.navigationController pushViewController:avc animated:YES];
}
```

What Do You Need To Do?

Refactor

Resources

Write Universal Apps

More Information

Bill Dudney

Application Frameworks Evangelist
dudney@apple.com

Apple Developer Forums

<http://devforums.apple.com>



