

Beginning iOS Development

A Deeper Look

Daniel Norton
8/17/2011 2:30 PM



Beginning iOS Development, MYC in iOS

Agenda

2

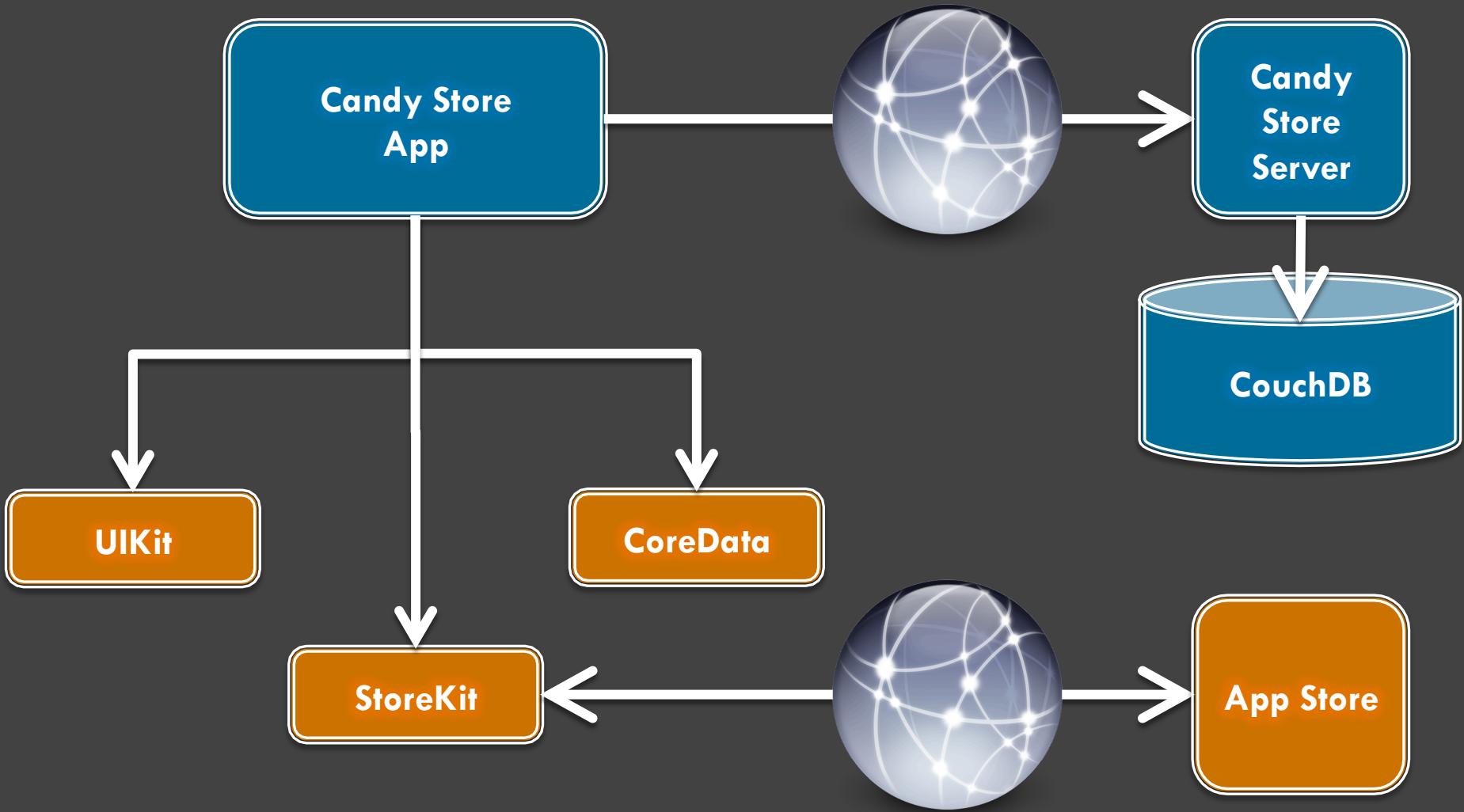
- Advanced Objective-C Language Features
 - Categories
 - Protocols
 - Delegates
 - Blocks
- View Animation
- Web Service Communication
- Core Data
- What's new in iOS5
- Additional Resources

github: `danielnorton`
blog: `framewreck.net`
twitter: `@daniel_norton`



`github.com/danielnorton/CandyStore-app`
`github.com/danielnorton/candystore-server`

The Demo App



Candy Store Architecture

A few caveats ...

5





Categories

How many times have you written code like this?

```
ShopItemDetailViewController *controller =  
    [[ShopItemDetailViewController alloc]  
     initWithNibName:@"ShopItemDetailViewController" bundle:nil];  
  
UIAlertViewController *alert = [[UIAlertViewController alloc] initWithTitle:@"Candy Store"  
                                         message:message  
                                         delegate:nil  
                                         cancelButtonTitle:@"OK"  
                                         otherButtonTitles:nil];  
[alert show];  
[alert release];
```

How many times have you written code like this?



```
ShopItemDetailViewController *controller =  
    [ShopItemDetailViewController newWithNibName];
```

```
[self popup:message];
```

Code

Uses of Categories

- Helper messages
- Extend iOS framework
- Private messages
- Common style



Protocols

Example iOS Protocols

- NSCoding
- UITableViewDataSource
- UITableViewDelegate
- UIApplicationDelegate
- UIAlertViewDelegate
- MKAnnotation
- NSFetchedResultsSectionInfo

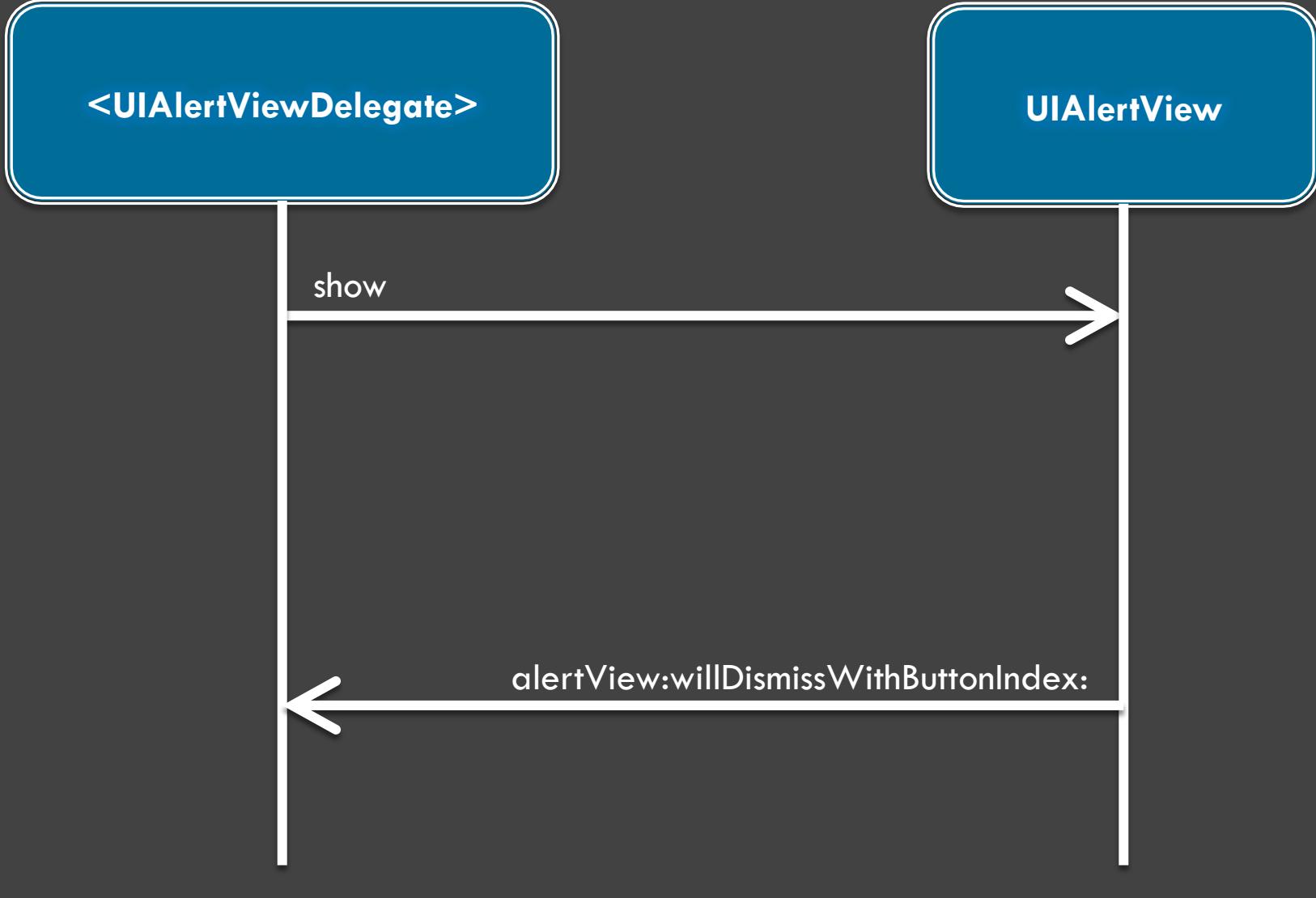
Code

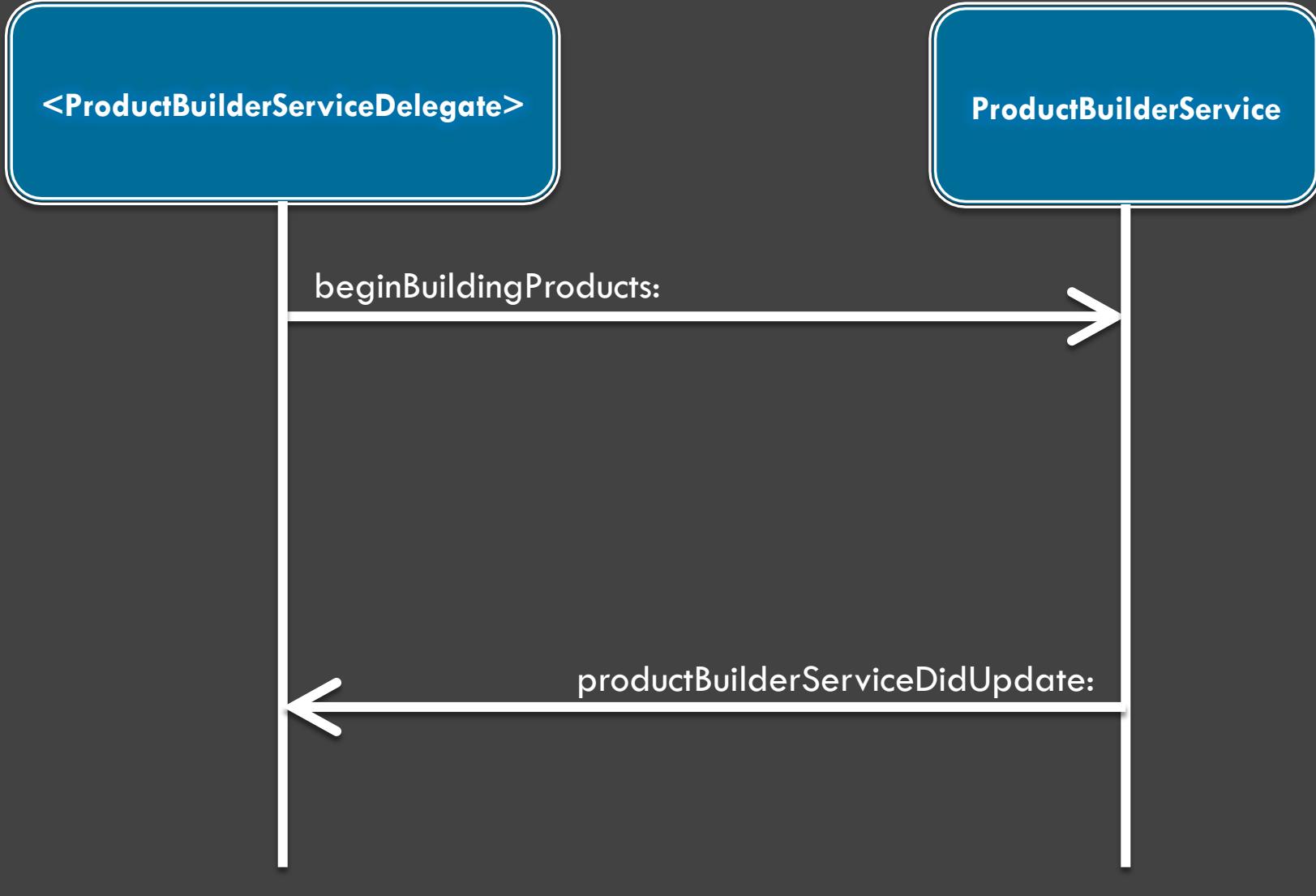
Implementing iOS Protocols

- Decorate a type as conforming to a protocol
- Documentation
- Declare a variable that conforms to a protocol



Delegates





Code

Delegates

- Delegate protocols
- Delegate properties
- Sending messages to a delegate
- Optional delegate messages



Blocks

Block Syntax

18

Return Type	Variable Name	Message Signature	Input Variables
int	(^add)(int, int)	= ^ (int one, int two) { return one + two; };	

Code Block

Calling the
block

```
int three = add(1, 2);
```

Block Syntax

19

```
void (^sayHi)() = ^() {  
    NSLog(@"hi");  
};
```

```
void (^simpler)() = ^{  
    NSLog(@"Hello");  
};
```

Block Syntax

20

```
NSNumber *(^makeCount)(NSArray *) = ^(NSArray *counted) {  
    NSNumber *number = [NSNumber numberWithInteger:counted.count];  
    return number;  
};
```

Block Syntax

21

```
NSArray *array = [NSArray arrayWithObjects:@"one", @"two", @"three", nil];  
  
[array enumerateObjectsUsingBlock:^(id obj, NSUInteger idx, BOOL *stop) {  
    NSLog(@"item: %@", obj);  
}];  
  
NSString *match = @"one";  
  
NSPredicate *pred =  
    [NSPredicate predicateWithBlock:^BOOL(id evaluatedObject, NSDictionary *bindings) {  
        NSString *item = (NSString *)evaluatedObject;  
        return [item isEqualToString:match];  
}];  
  
NSArray *filtered = [array filteredArrayUsingPredicate:pred];
```

Block Syntax

22

```
NSArray *array = [NSArray arrayWithObjects:@"one", @"two", @"three", nil];  
  
NSString *greeting = @"Hello";  
_block int count = 0;  
[array enumerateObjectsUsingBlock:^(id obj, NSUInteger idx, BOOL *stop) {  
  
    count++;  
    NSLog(@"%@", greeting, obj);  
}];
```

Code

Blocks

- Enumerating NSArray and NSDictionary
- Method pointer



View Animation

```
[UIView animateWithDuration:0.33f
                      delay:1.0f
                    options:UIViewAnimationOptionCurveEaseIn
      animations:^{
        [self setFrame:newFrame];
    }

    completion:^(BOOL finished) {
        [self.titleLabel setAlpha:1.0f];
    }];
}
```

View Animation

Code

[View Animation](#)

- Change location and size of a button



Web Service Communication

NSURLConnection

NSURLResponse

NSURLRequest

NSMutableURLRequest

NSData

NSMutableData

Web Service Communication Classes

NSURLConnection

```
NSURLConnection *aConnection =  
[ [NSURLConnection alloc] initWithRequest:request  
delegate:self];
```

- `(void)connection:(NSURLConnection *)connection didSendBodyData:(NSInteger)bytesWritten totalBytesWritten:(NSInteger)totalBytesWritten totalBytesExpectedToWrite:(NSInteger)totalBytesExpectedToWrite;`
- `(void)connection:(NSURLConnection *)connection didReceiveResponse:(NSURLResponse *)response;`
- `(void)connection:(NSURLConnection *)connection didReceiveData:(NSData *)data;`
- `(void)connection:(NSURLConnection *)connection didFailWithError:(NSError *)anError;`
- `(void)connectionDidFinishLoading:(NSURLConnection *)connection;`

NSURLConnection Delegate

Code

Web Service Communication

- Connect to a web service
- Edit HTTP Body, Method, etc.
- Parse return JSON

HTTPRequestService

RemoteServiceBase

AppProductRemoteService

Example Candy Store Implementation



Core Data

NSManagedObjectContext

NSFetchedResultsController

.xcdatamodel

NSManagedObjectID

NSManagedObject

.sqlite

NSManagedObjectModel

NSPersistentStoreCoordinator

Core Data

NSManagedObjectContext

NSFetchedResultsController

.xcdatamodel

NSManagedObjectID

Product

.sqlite

NSManagedObjectModel

NSPersistentStoreCoordinator

Core Data

Code

Core Data

- Model schema
- Typed Model Classes
- CRUD
- Fetched results controller
- Versioning

Gotcha's When editing the model without versioning

- Reset Simulator and/or delete app from device
- Xcode -> Product -> Clean

What's New in iOS5

37

How are the topics we discussed today impacted by new features and technologies in iOS5?

- Automatic Reference Counting
- Core Data
 - Nested Contexts
 - Nonatomic persistant stores

Additional Resources

38

- iOS Developer Library
 - Core Data Programming Guide
 - iOS Development Guide
 - iOS Application Programming Guide
 - Coding Guidelines for Cocoa
 - What's New in iOS
- WWDC 2011 Videos
 - Blocks and Grand Central Dispatch in Practice

Review

39

- Advanced Objective-C Language Features
 - Categories
 - Protocols
 - Delegates
 - Blocks
- Animation
- Web Service Communication
- Core Data
- What's new in iOS5
- Additional Resources

github: danielnorton
blog: framewreck.net
twitter: @daniel_norton



github.com/danielnorton/CandyStore-app
github.com/danielnorton/candystore-server

Thanks, guys! Enjoy the rest of devLink 2011!