





Hello NSCoder I

Introducción al desarrollo para iOS

NSCoder

- Aficionados al mundo de la manzana
- NSCoder Night
 - Cada jueves 19:00 - Frankfurt de Pelayo
- Twitter: @nscoder_bcn
- http://groups.google.com/group/nscoder_bcn
- <http://nscoder-bcn.tumblr.com/>



Programa (Mañana)

- Introducción iOS “Hello NSCoder”
- Web Service “What’s up, NSCoder”
- Gestión de Memoria
- Debugging and Memory Leaks
- Testing



Programa (Tarde)

- Presentación de Omnis Studio
- Introducción a la programación con Omnis Studio
- Programación de una aplicación cliente Omnis con XCode para iPhone/iPad



Referencias

- Material
 - github
- Apple Developer
- NSCoderCenter
- 85% Cocoa



Resultado

- App que nos localiza
- Conecta a un Web Service
- Devuelva eventos a nuestros alrededores
- Table View
- Detail View



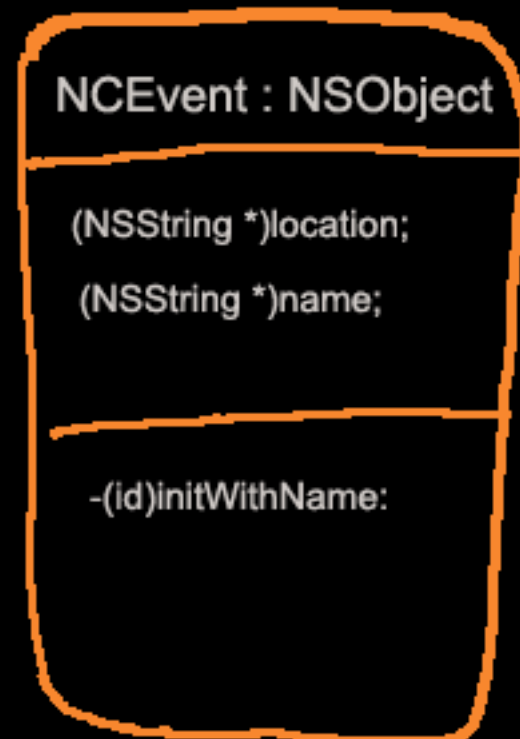
Requirimientos

- Conocimientos Objective C / iOS
- MacOS X
- iOS SDK
- XCode / Interface Builder



Objective C

- Class / Instance
- Método / Variable
 - Message
- .h / .m



```
- (id)initWithName:(NSString *)name;
+ (id)eventFromPList:(NSString *)listPath;
```



Objective C

- Memory Allocation
 - release / retain / nil
- Message
 - receiver / selector / arguments

```
NCEvent *event = [[NCEvent alloc] initWithName:name  
                  andLocation:location];  
  
[event release];  
event = nil;
```



Objective C

- Properties
 - property / synthesize
 - nonatomic
 - copy, retain, assign, readonly

```
@property (nonatomic, copy) NSString *name;  
@synthesize name;  
  
self.name = @"Hello NSCoder";  
[self setName:@"Hello NSCoder"];
```



Objective C

```
@interface NCEvent : NSObject {  
}  
  
@property (nonatomic, copy) NSString *name;  
  
- (id)initWithName:(NSString *)name;  
  
@end
```

.h

```
@implementation NCEvent  
@synthesize name;  
  
- (id)initWithName:(NSString *)name {  
    ...  
}  
  
@end
```

.m



Objective C

- Inheritance
- Protocol

```
@protocol NCEventDelegate <NSObject>  
  
@required  
- (void)didFinishUpdatingData;  
  
@end
```

.h

```
@interface NCController : NSObject <NCEventDelegate> {  
}  
  
@end
```

.h



iOS

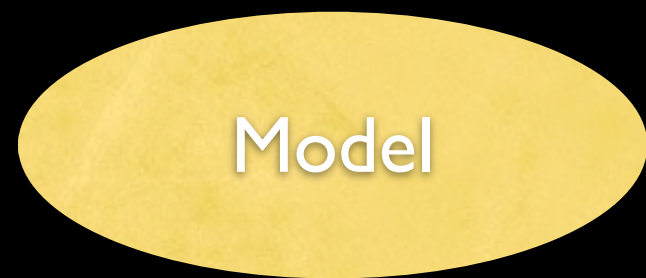
- Frameworks
 - Foundation
 - UIKit
- UIView / UITableView
- UINavigationController
 - root / push / pop



iOS - MVC



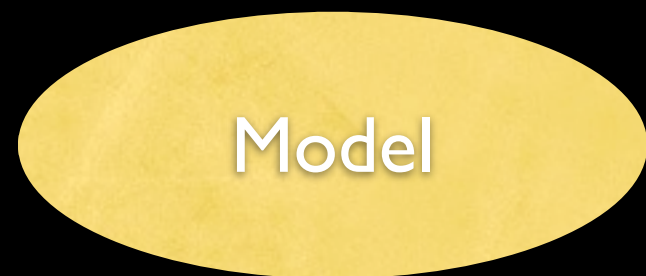
iOS - MVC



Data



iOS - MVC



Data



UI



iOS - MVC

Logica

Controller

Model

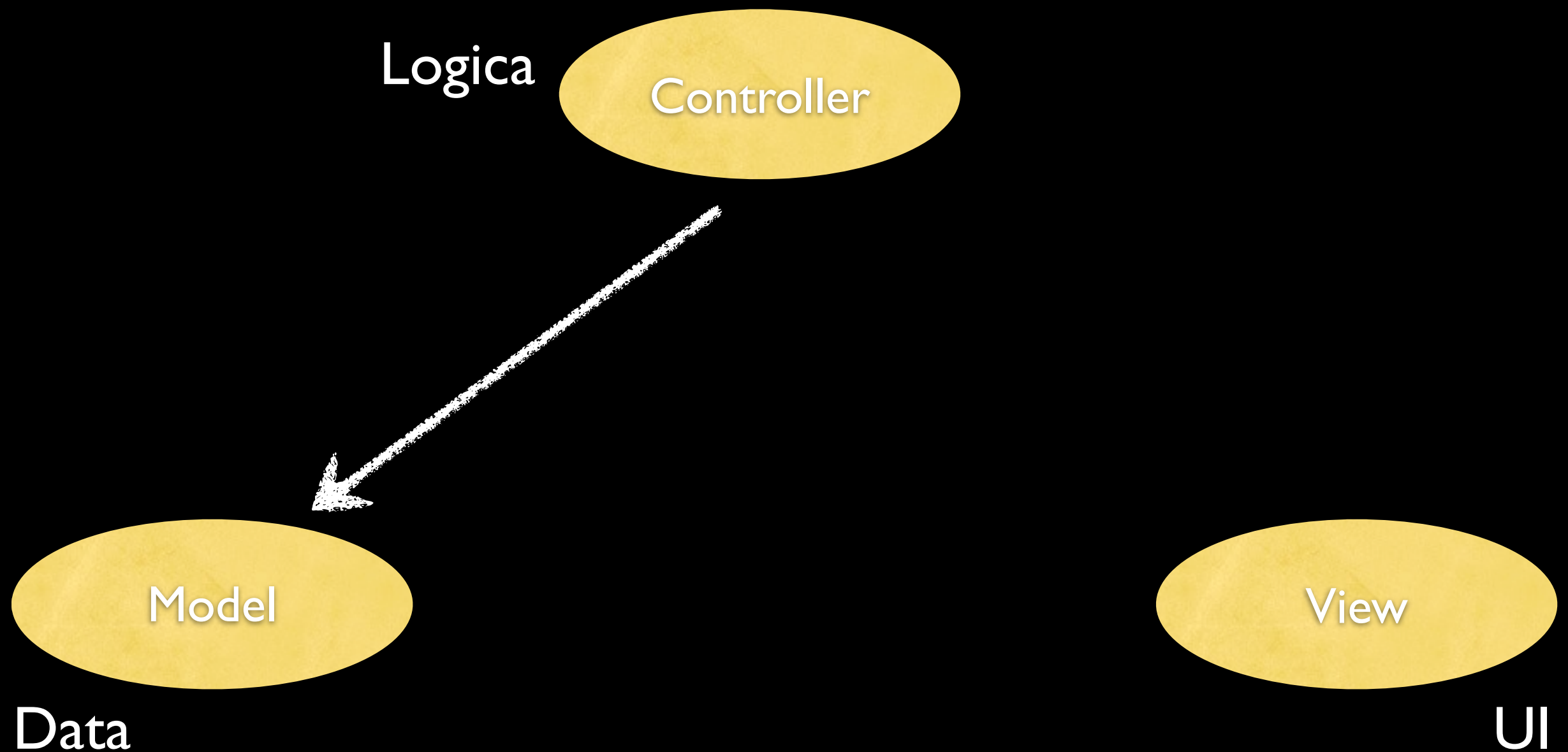
Data

View

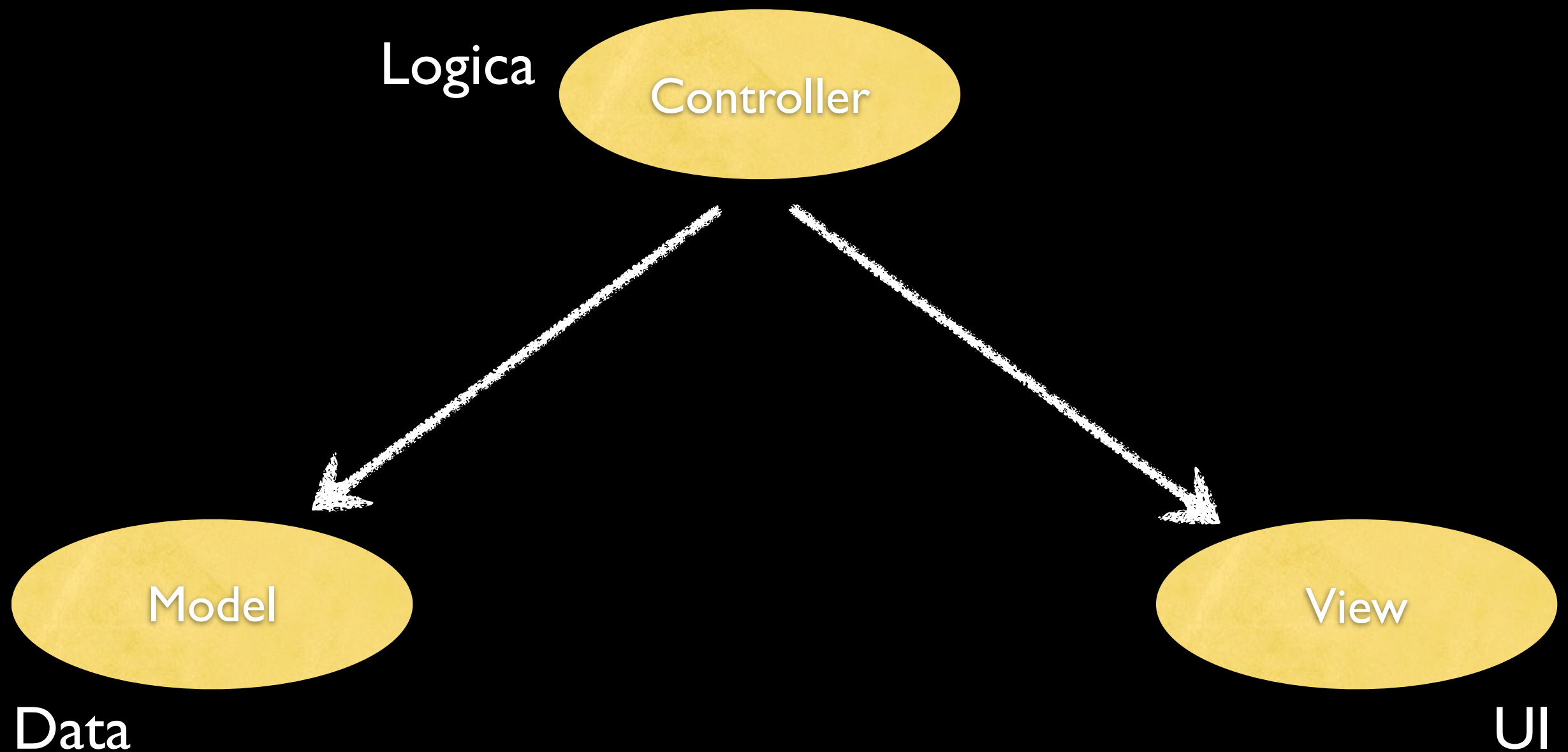
UI



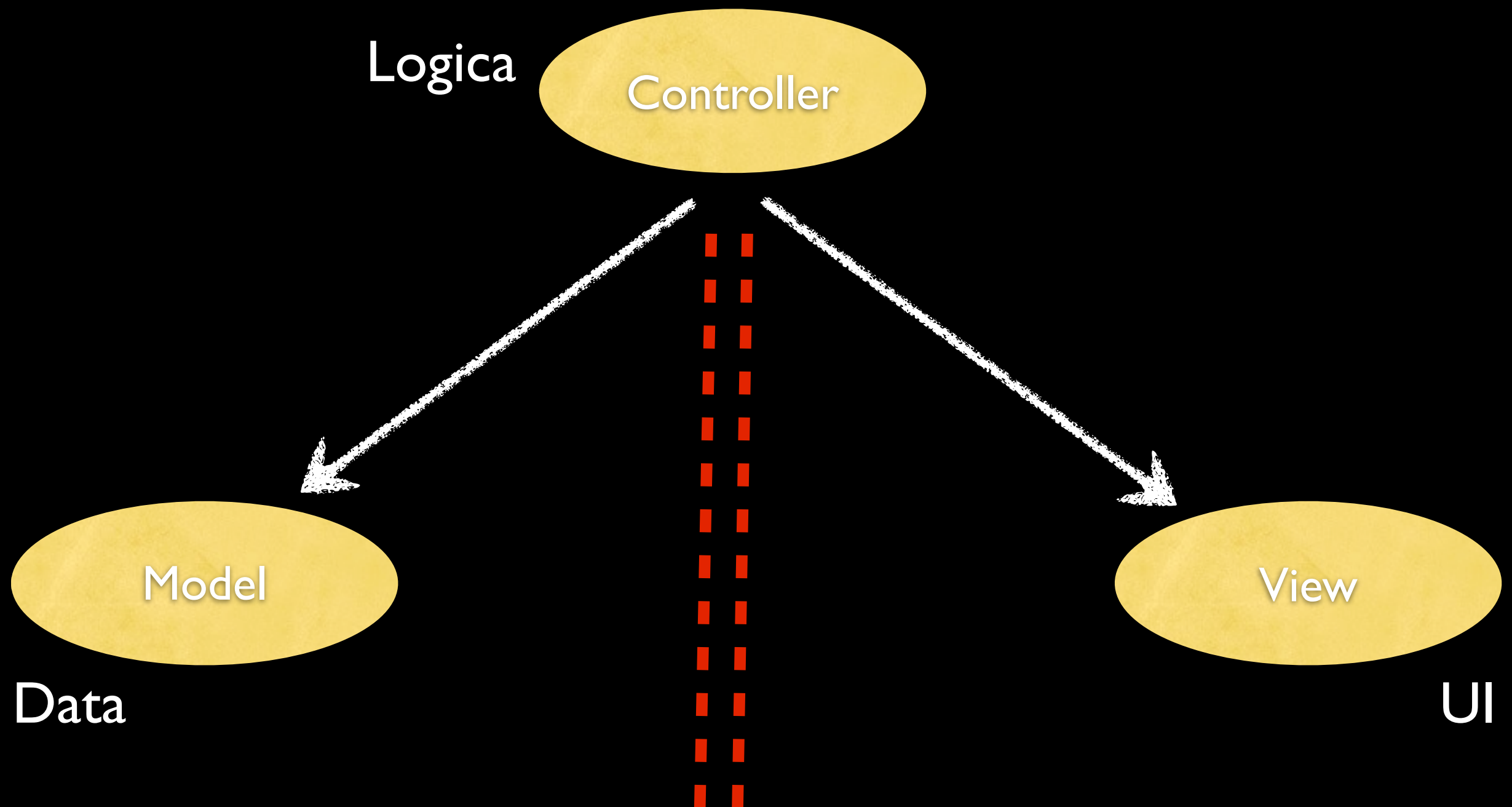
iOS - MVC



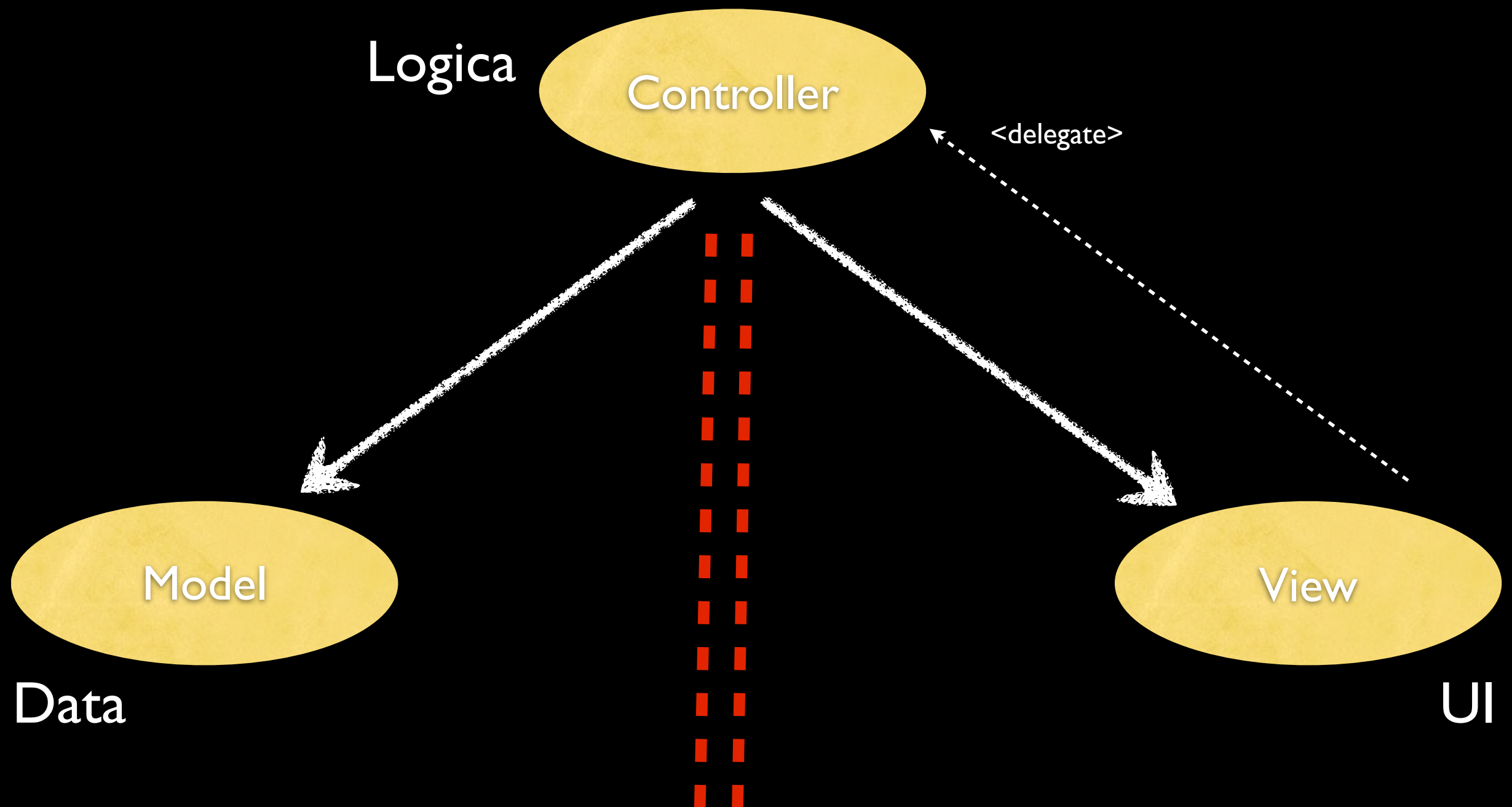
iOS - MVC



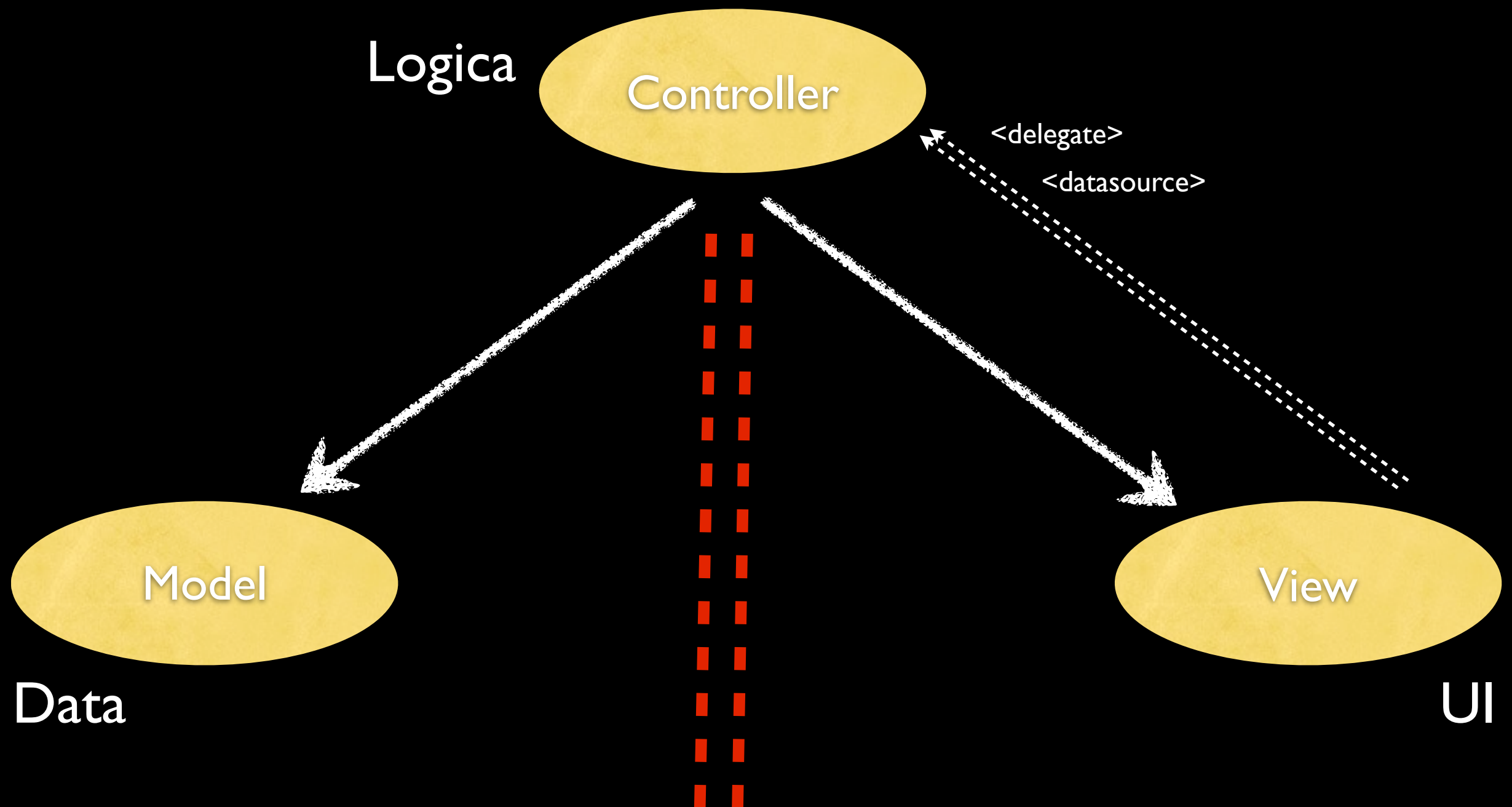
iOS - MVC



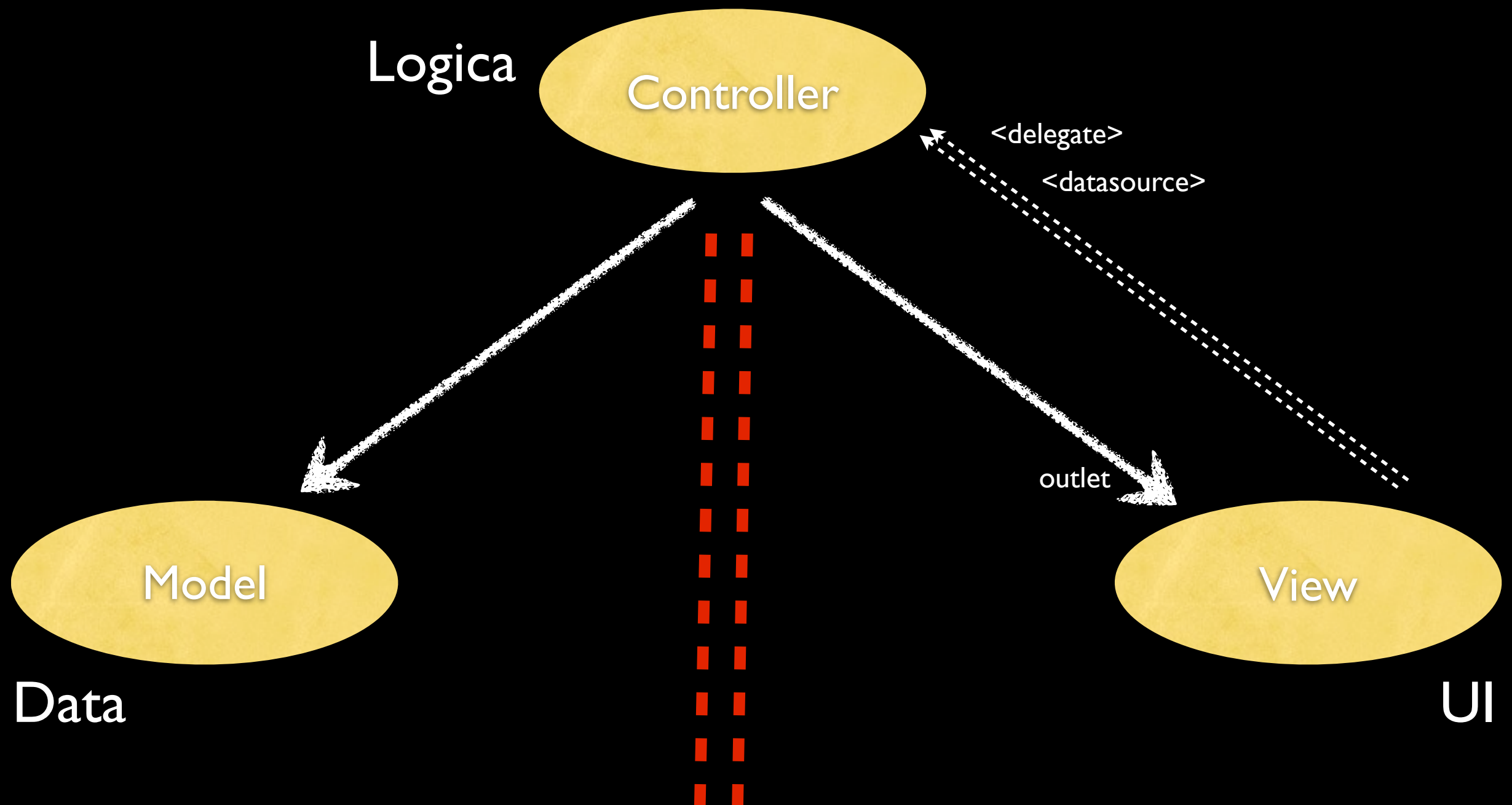
iOS - MVC



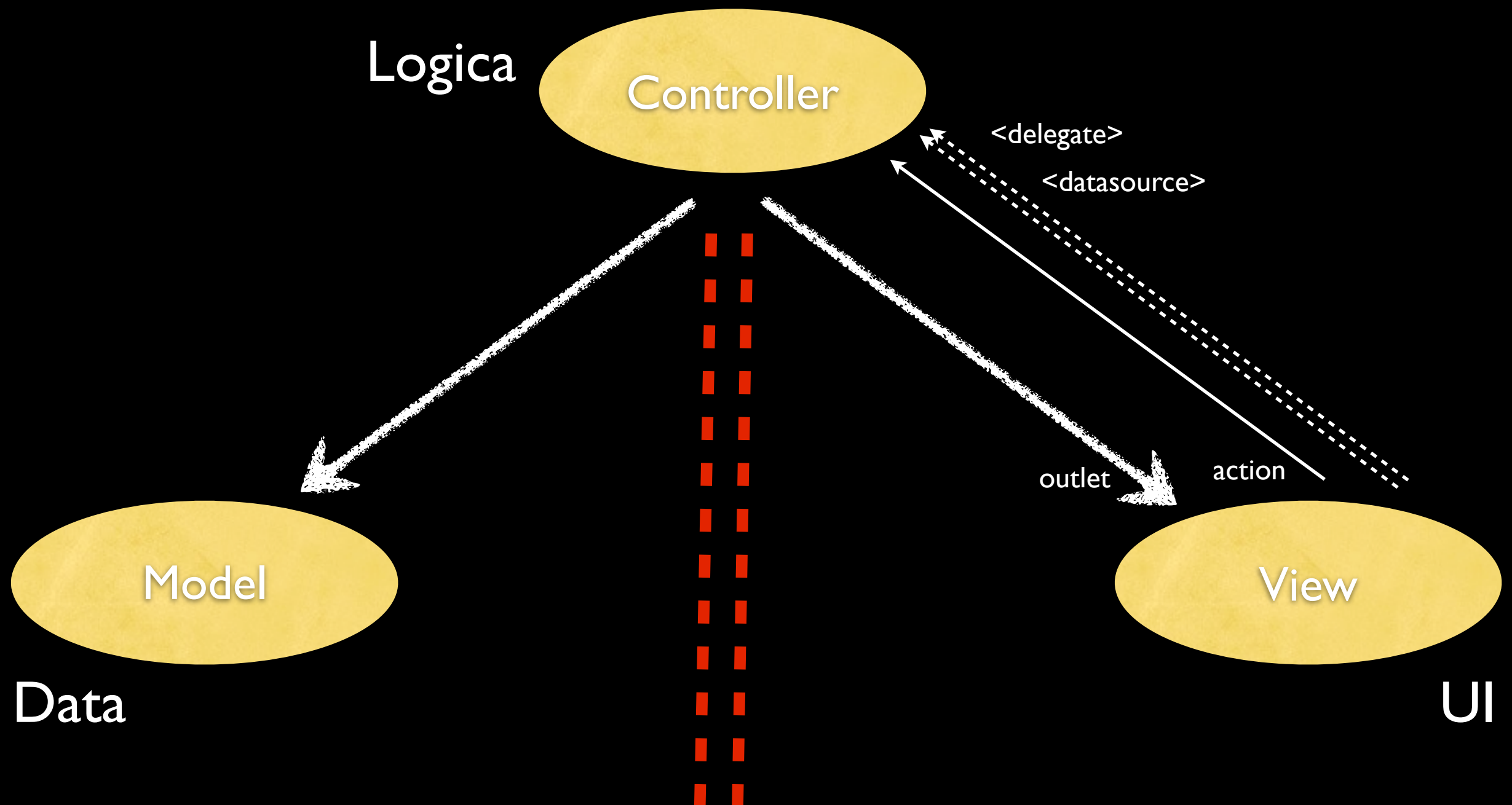
iOS - MVC



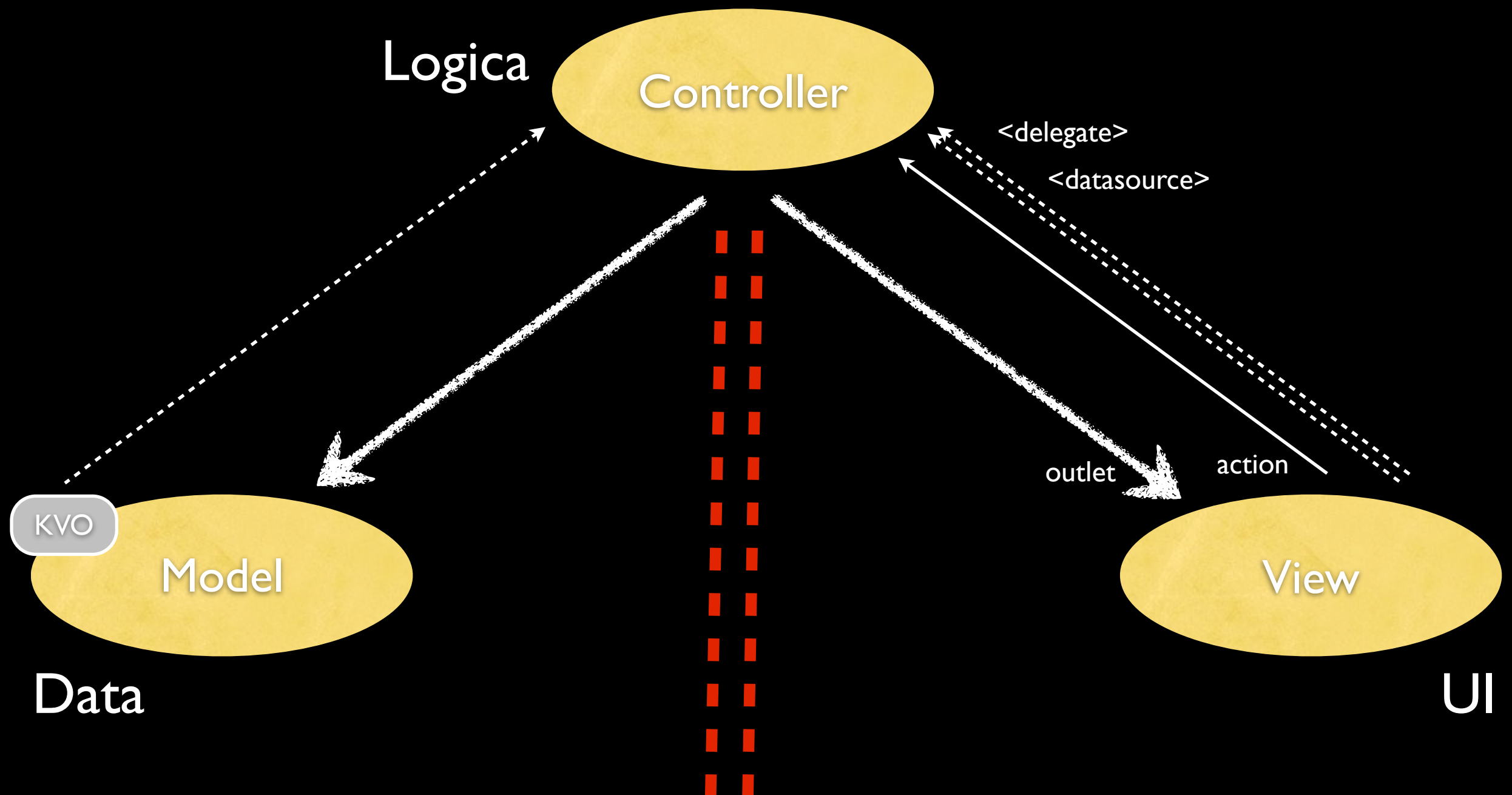
iOS - MVC



iOS - MVC



iOS - MVC



Pasos

- Crear Proyecto
- Crear DetailView & Controller
- Crear NCEvent objeto
- Add NavigationController
- Crear TableView
- Push/Pop DetailView



XCode: Crear Proyecto

- IDE for Objective C
- Templates
 - View-based Application
- `com.nscoderbcn.HelloNSCoder`
- Include Unit Tests



IB: DetailView

- Herramienta para crear GUI
- xib/nib files
- IBOutlet / IBAction



Gracias!

- @nscoder_bcn
- Alfonso: @alfonsmata
- Andreas: @aquarioverde

