Advanced Gesture Recognition

Fail early, fail often

Josh Shaffer and Brad Moore
iPhone Frameworks Engineers

What You'll Learn

- View interaction—on touches and gestures and views
- Subclassing—defining custom gestures

UIResponder Review

```objc
- (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event;
- (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event;
- (void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event;
- (void)touchesCancelled:(NSSet *)touches withEvent:(UIEvent *)event;
```

UIResponder Review

UIWindow: - (void)sendEvent:(UIEvent*)event;

- (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event;

- (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event;

- (void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event;

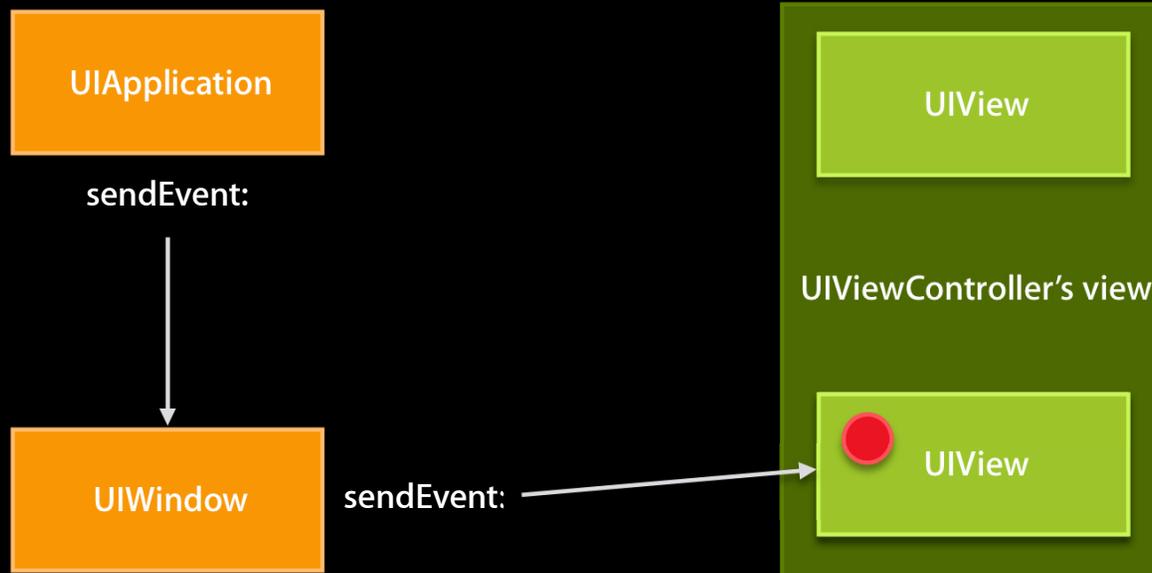- (void)touchesCancelled:(NSSet *)touches withEvent:(UIEvent *)event;

UIResponder Review
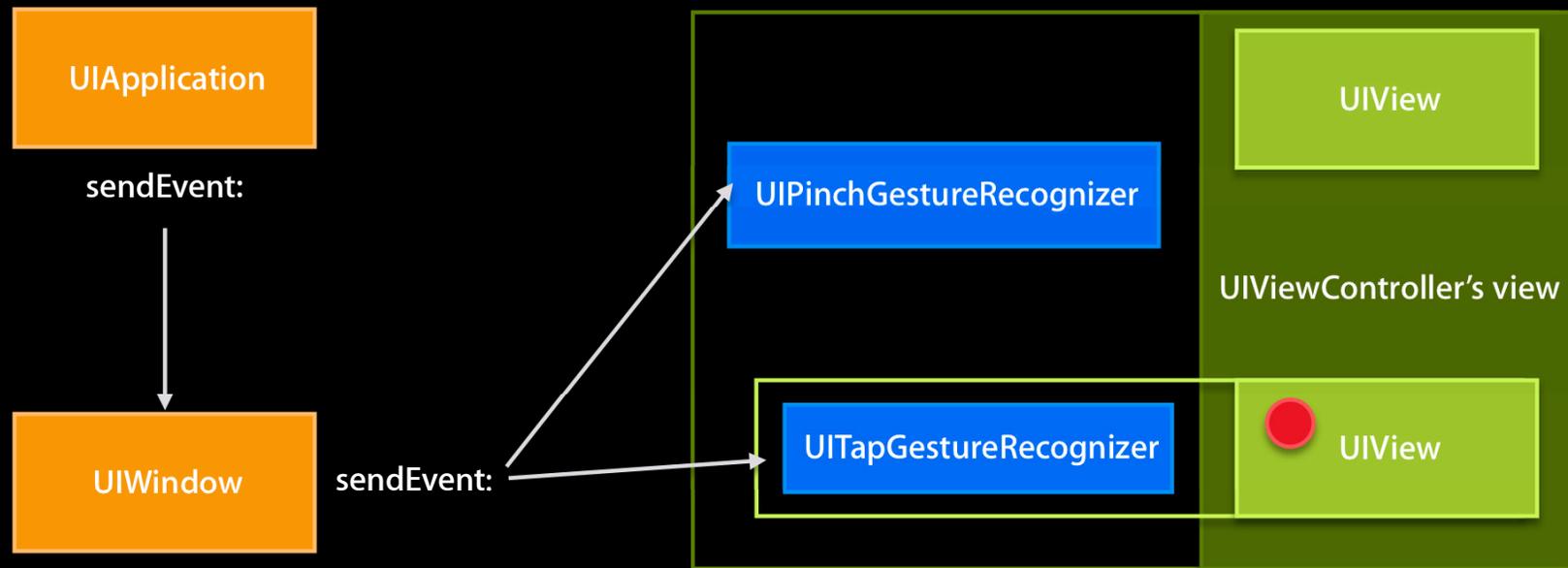
UIApplication: - (void)sendEvent:(UIEvent*)event;

UIWindow: - (void)sendEvent:(UIEvent*)event;

- (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event;
- (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event;
- (void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event;
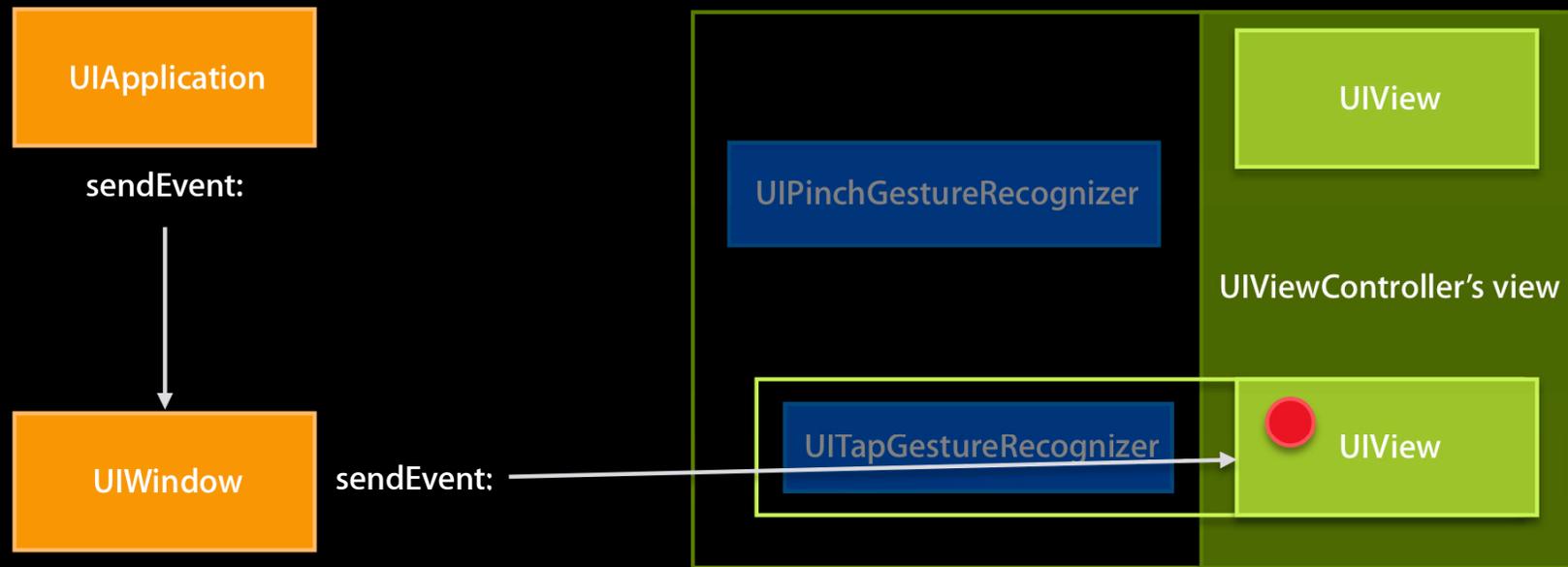- (void)touchesCancelled:(NSSet *)touches withEvent:(UIEvent *)event;
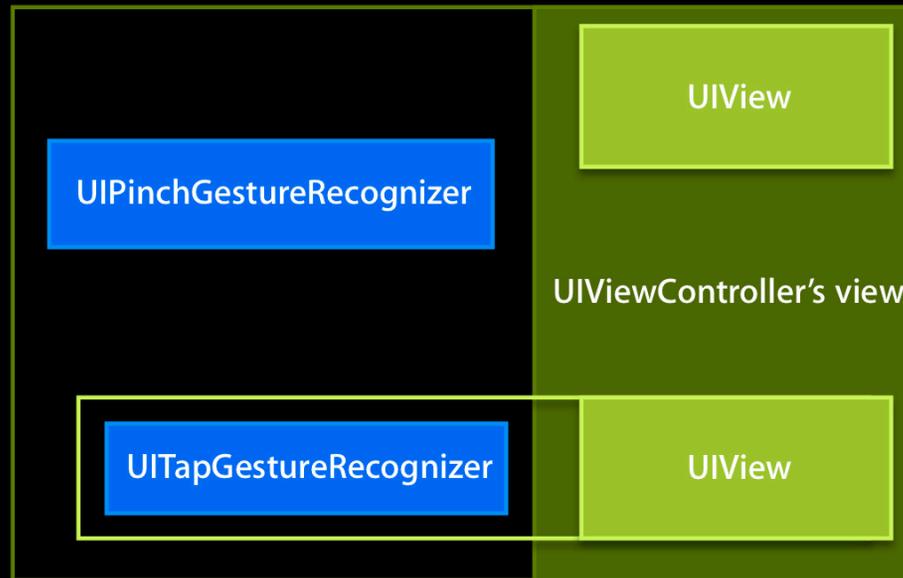
Touch Delivery with Views

UIApplication

sendEvent:

UIWindow   sendEvent:

UIView

UIViewController's view

UIView

Touch Delivery with Gesture Recognizers

UIApplication

sendEvent:

UIWindow

sendEvent:

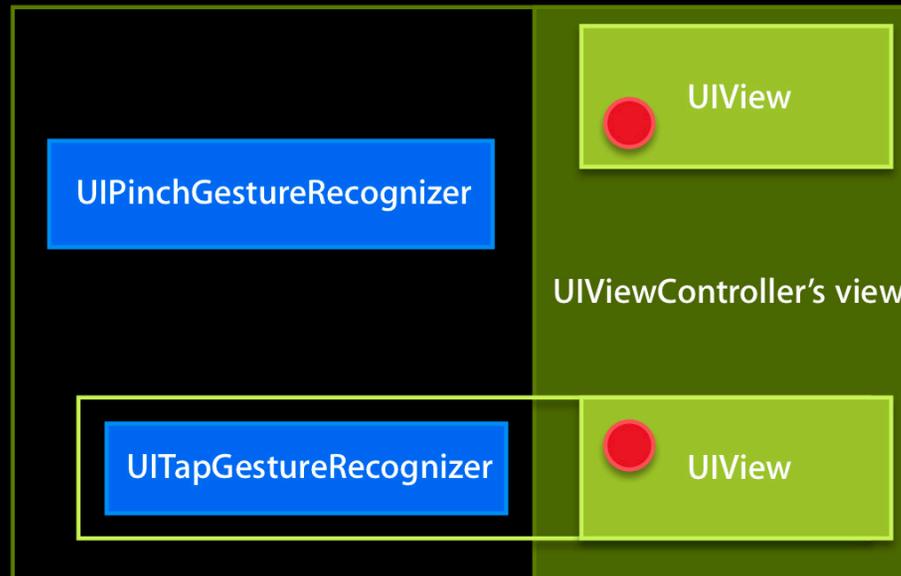UIPinchGestureRecognizer

UIView

UIViewController's view

UITapGestureRecognizer

UIView

UIView

9

Touch Cancellation

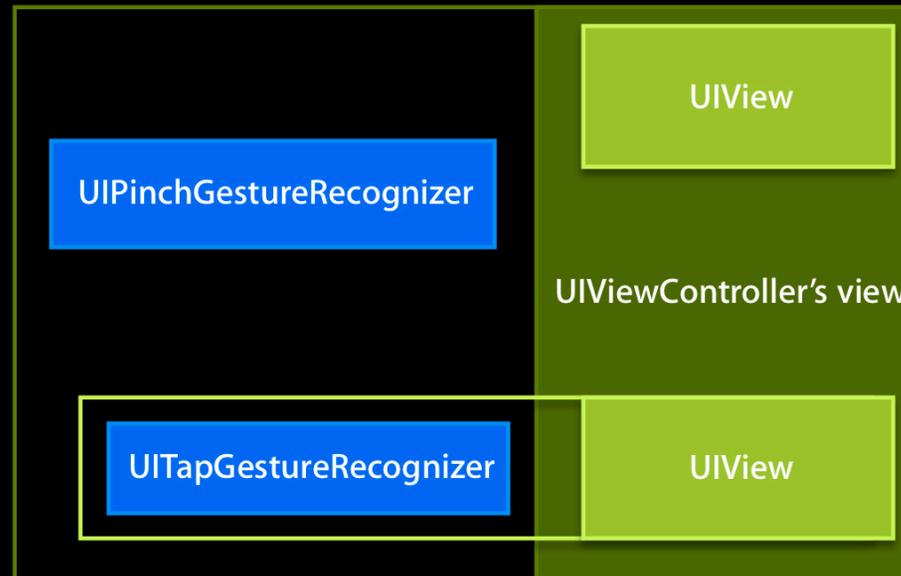| Recognizers	touchesBegan:	touchesMoved:	touchesMoved:	touchesMoved:	touchesEnded:
Tap	Possible	Possible	Failed		
Pinch	Possible	Possible	Began	Changed	Ended
UIViews	touchesBegan:	touchesMoved:	touchesCancelled:		

UIPinchGestureRecognizer

UIView

UIViewController's view

UITapGestureRecognizer

UIView

Interaction with Views
Canceling Touches

```
@property BOOL cancelsTouchesInView;
```

No Touch Cancellation

Recognizers	touchesBegan:	touchesMoved:	touchesMoved:	touchesMoved:	touchesEnded:
Tap	Possible	Possible	Failed		
Pinch	Possible	Possible	Began	Changed	Ended
UIViews	touchesBegan:	touchesMoved:	touchesMoved:	touchesMoved:	touchesEnded:

UIView

UIPinchGestureRecognizer

UIViewController's view
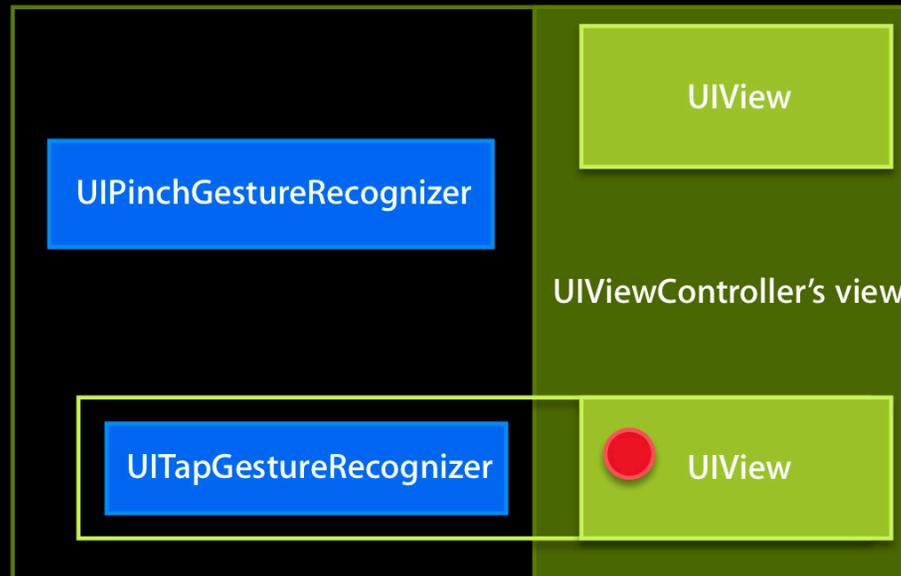
UITapGestureRecognizer

UIView

Interaction with Views
Delaying Touches

```
@property BOOL delaysTouchesEnded;

@property BOOL delaysTouchesBegan;
```

Delay Touches Ended

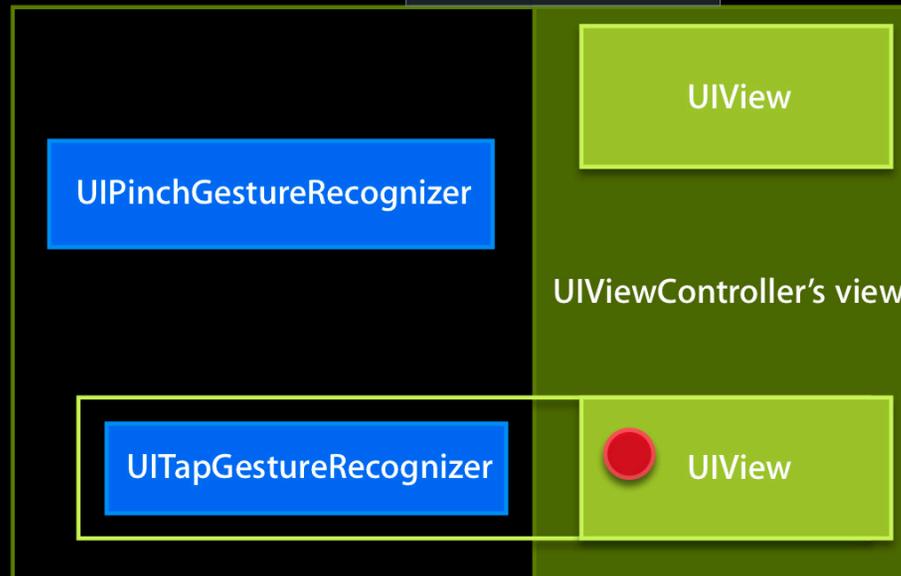| Recognizers	touchesBegan:	touchesMoved:	touchesEnded:	
Double Tap	Possible	Possible	Possible	Failed
Pinch	Possible	Possible	Failed	
UIView	touchesBegan:	touchesMoved:		touchesEnded:

Delay Touches Began

Recognizers	touchesBegan:	touchesMoved:	touchesMoved:	touchesMoved:	touchesEnded:
Tap	Possible	Possible	Failed		
Pinch	Possible	Possible	Possible	Possible	Failed
UIViews			touchesBegan:	touchesMoved:	touchesEnded:
			touchesMoved:		

UIView

UIPinchGestureRecognizer

UIViewController's view

UITapGestureRecognizer

UIView

Subclassing UIGestureRecognizer

#import <UIGestureRecognizerSubclass.h>

Subclassing UIGestureRecognizer
UIGestureRecognizerSubclass.h

- UIGestureRecognizer is not a subclass of UIResponder

```
- (void)touchesBegan:     withEvent:
- (void)touchesMoved:     withEvent:
- (void)touchesEnded:     withEvent:
- (void)touchesCancelled: withEvent:
```

Subclassing UIGestureRecognizer
Changing states

- UIGestureRecognizer.h

  ```
  @property(nonatomic,readonly) UIGestureRecognizerState state;
  ```

- UIGestureRecognizerSubclass.h

  ```
  @property(nonatomic,readwrite) UIGestureRecognizerState state;
  ```

Subclassing UIGestureRecognizer
Failure is unavoidable

UIGestureRecognizerStatePossible

UIGestureRecognizerStateFailed

UIGestureRecognizerStatePossible

Subclassing UIGestureRecognizer
Discrete state transitions

UIGestureRecognizerStatePossible

UIGestureRecognizerStateRecognized

UIGestureRecognizerStatePossible

Subclassing UIGestureRecognizer
Continuous state transitions

UIGestureRecognizerStatePossible

↓

UIGestureRecognizerStateBegan

↓

UIGestureRecognizerStateChanged

↓

UIGestureRecognizerStateEnded

↓

UIGestureRecognizerStatePossible

Subclassing UIGestureRecognizer
Continuous state transitions

UIGestureRecognizerStatePossible

UIGestureRecognizerStateBegan

UIGestureRecognizerStateChanged
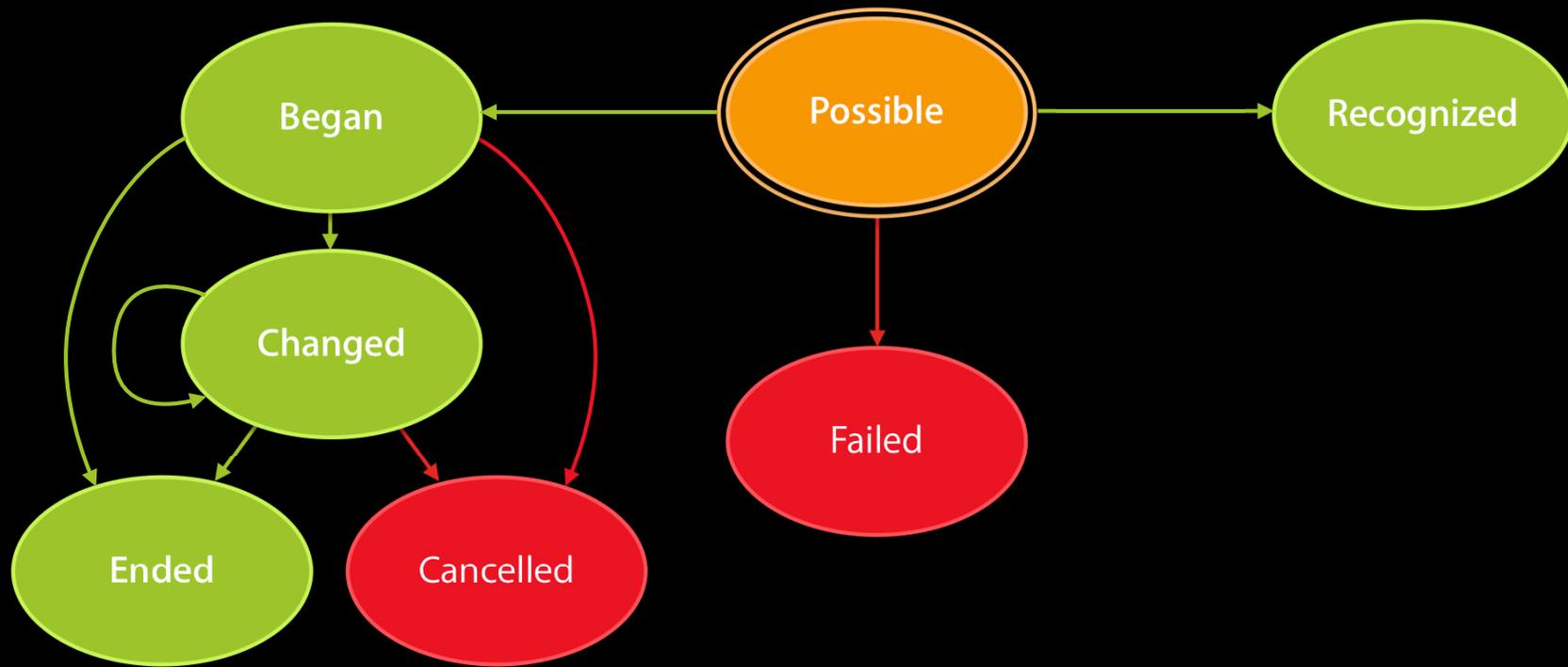
UIGestureRecognizerStateCancelled

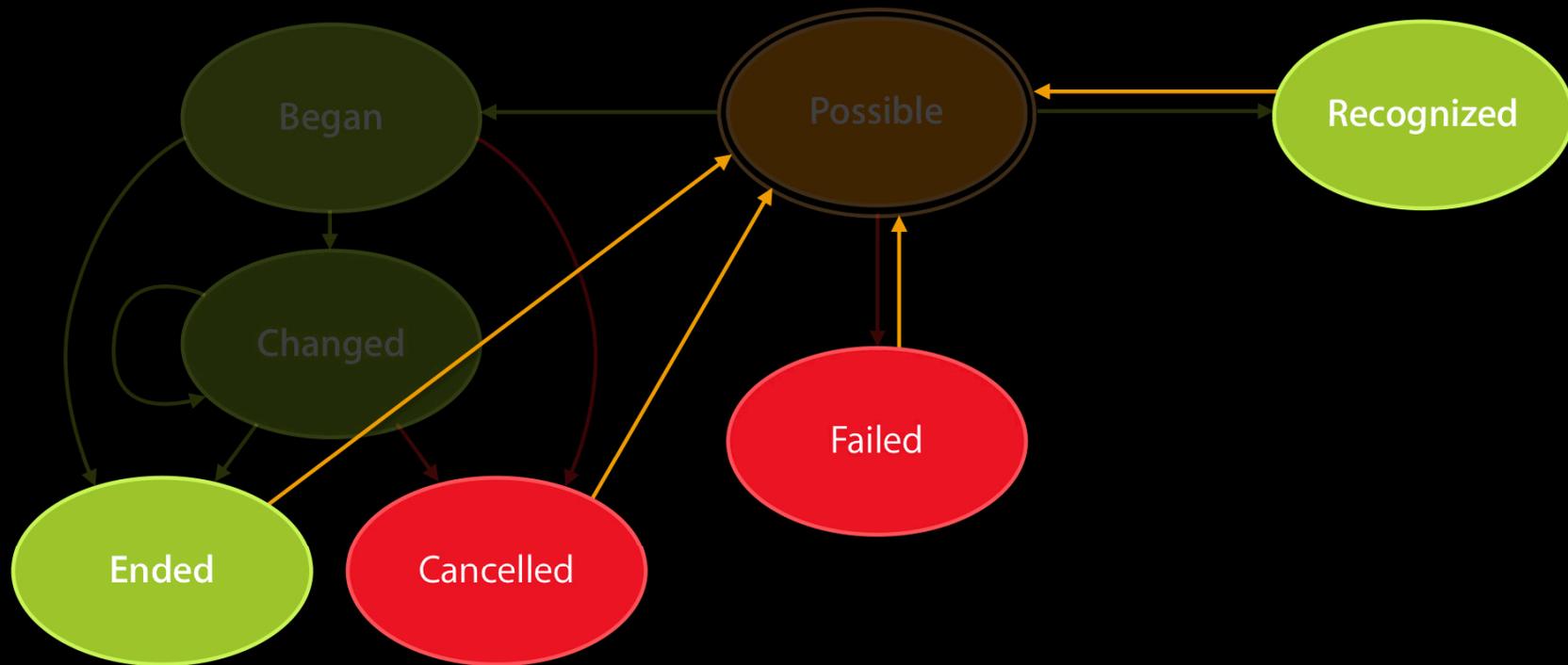UIGestureRecognizerStatePossible

Subclassing UIGestureRecognizer
Transitions as a state machine

Subclassing UIGestureRecognizer
Transitions as a state machine

Subclassing UIGestureRecognizer
Automatic reset

- State set to UIGestureRecognizerStatePossible

- Subclass resets state

  - `(void)reset;`

- All failure requirements must be fulfilled again

- All existing touches ignored

Subclassing UIGestureRecognizer
Ignoring touches

- Stop further delivery

  - `- (void)ignoreTouch:(UITouch*)touch forEvent:(UIEvent*)event;`

- Manually ignoring prevents cancellation

Subclassing UIGestureRecognizer
Effects of state changes

State	Actions Performed	Reset
UIGestureRecognizerStatePossible		
UIGestureRecognizerStateBegan	✓	
UIGestureRecognizerStateChanged	✓	
UIGestureRecognizerStateEnded	✓	✓
UIGestureRecognizerStateCancelled	✓	✓
UIGestureRecognizerStateRecognized	✓	✓
UIGestureRecognizerStateFailed		✓

Writing a Gesture Recognizer

How not to recognize taps

Subclassing UIGestureRecognizer
Simple tap gesture recognizer

```objc
#import <UIKit/UIGestureRecognizerSubclass.h>

@interface SimpleTapRecognizer : UIGestureRecognizer {}

- (void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event {
    if ([[event touchesForGestureRecognizer:self] count] == 1 &&
        [[touches anyObject] tapCount] == 1) {
        self.state = UIGestureRecognizerStateRecognized;
    } else {
        self.state = UIGestureRecognizerStateFailed;
    }
}


- (void)touchesCancelled:(NSSet *)touches withEvent:(UIEvent *)event {
    self.state = UIGestureRecognizerStateFailed;
}
```

Subclassing UIGestureRecognizer
Simple tap gesture recognizer

```
- (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event {
    if ([[event touchesForGestureRecognizer:self] count] > 1) {
        self.state = UIGestureRecognizerStateFailed;
    }
}

- (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event {
    if ([[touches anyObject] tapCount] != 1) {
        self.state = UIGestureRecognizerStateFailed;
    }
}
```

Subclassing UIGestureRecognizer
Preventing exclusion

```
@property (nonatomic) NSUInteger numberOfTapsRequired;

- (BOOL)canPreventGestureRecognizer:(UIGestureRecognizer *)gestureRecognizer {
    if ([gestureRecognizer isKindOfClass:[SimpleTapRecognizer class]] &&
        [gestureRecognizer numberOfTapsRequired] > [self numberOfTapsRequired]) {
        return NO;
    }

    return YES;
}

- (BOOL)canBePreventedByGestureRecognizer:(UIGestureRecognizer *)recognizer {
    return YES;
}
```

Best Practices
How to fail gracefully

- Perform calculations for recognition in screen coordinates
- Convert to local view coordinates when necessary

Demo

Brad Moore
iPhone Frameworks Engineer

More Information

Bill Dudney
Application Frameworks Evangelist
dudney@apple.com

Documentation
UIGestureRecognizer Class Reference
http://developer.apple.com/iphone/library/documentation/uikit/reference/
UIGestureRecognizer_Class/Reference/Reference.html

Apple Developer Forums
http://devforums.apple.com

Related Sessions

Simplifying Touch Event Handling with Gesture Recognizers	Pacific Heights Wednesday 3:15PM
Mastering Table Views	Pacific Heights Thursday 11:30AM

Labs

Cocoa Touch Lab	Application Frameworks Lab C Thursday 9:00AM –11:15AM
Cocoa Touch Lab	Application Frameworks Lab C Friday 9:00AM –11:45AM
Gesture Recognition Lab	Application Frameworks Lab A Thursday 2:00PM