iOS Gesture Recognizers

CocoaConf Dallas: April 2013

Jonathan Penn

@jonathanpenn

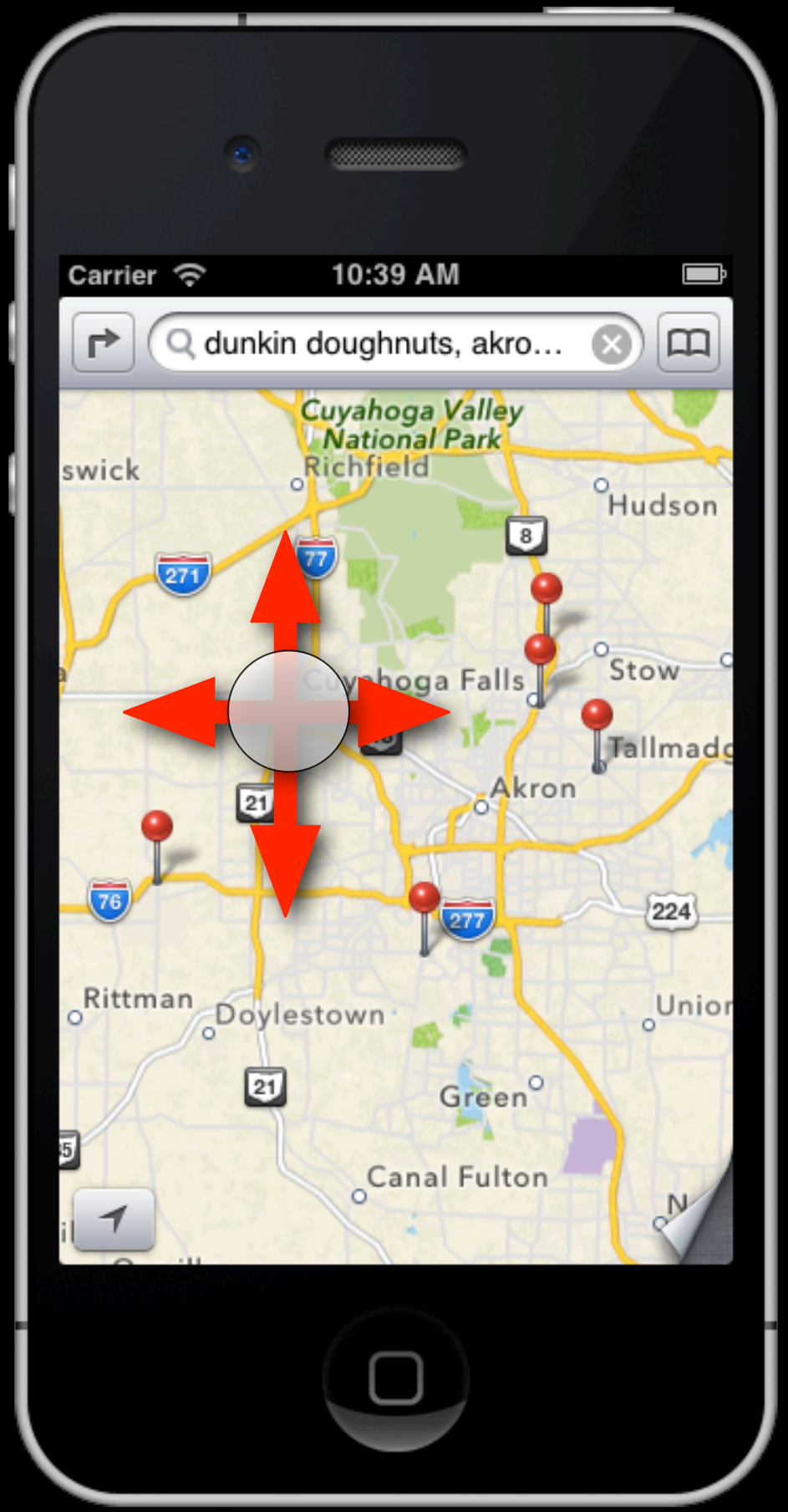
Navel Labs

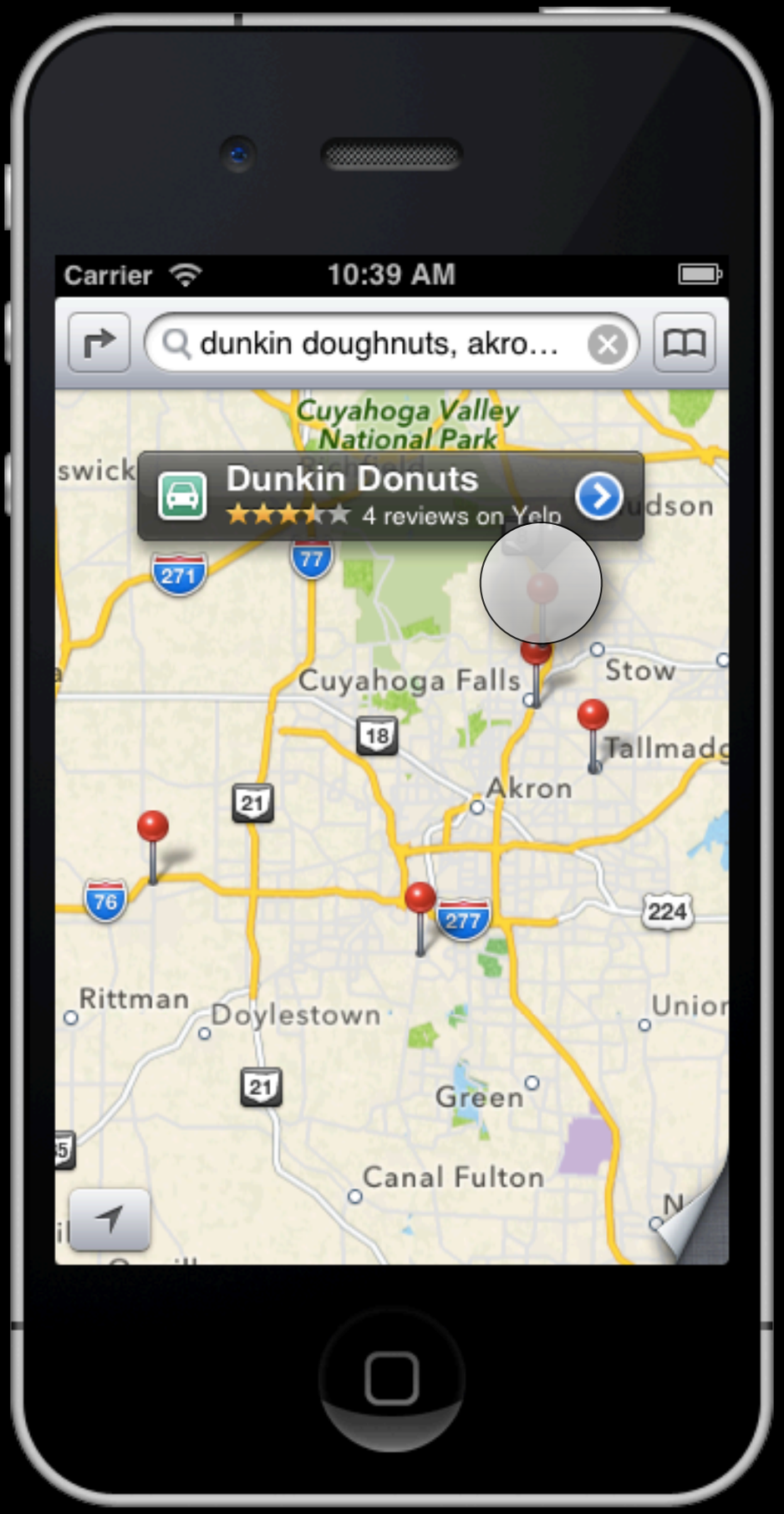
COCOA MANIFEST

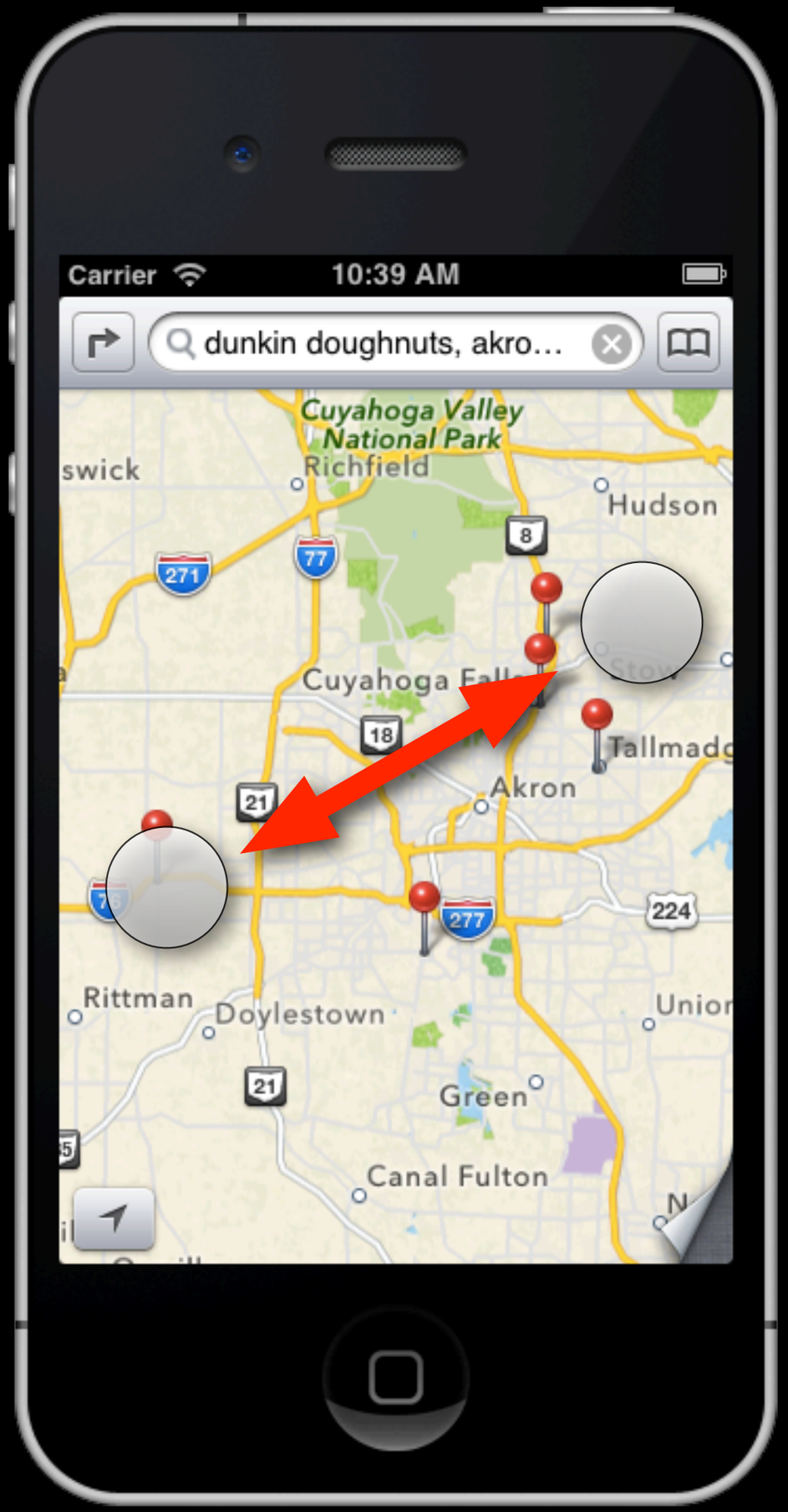
Slides n’ Sample Code

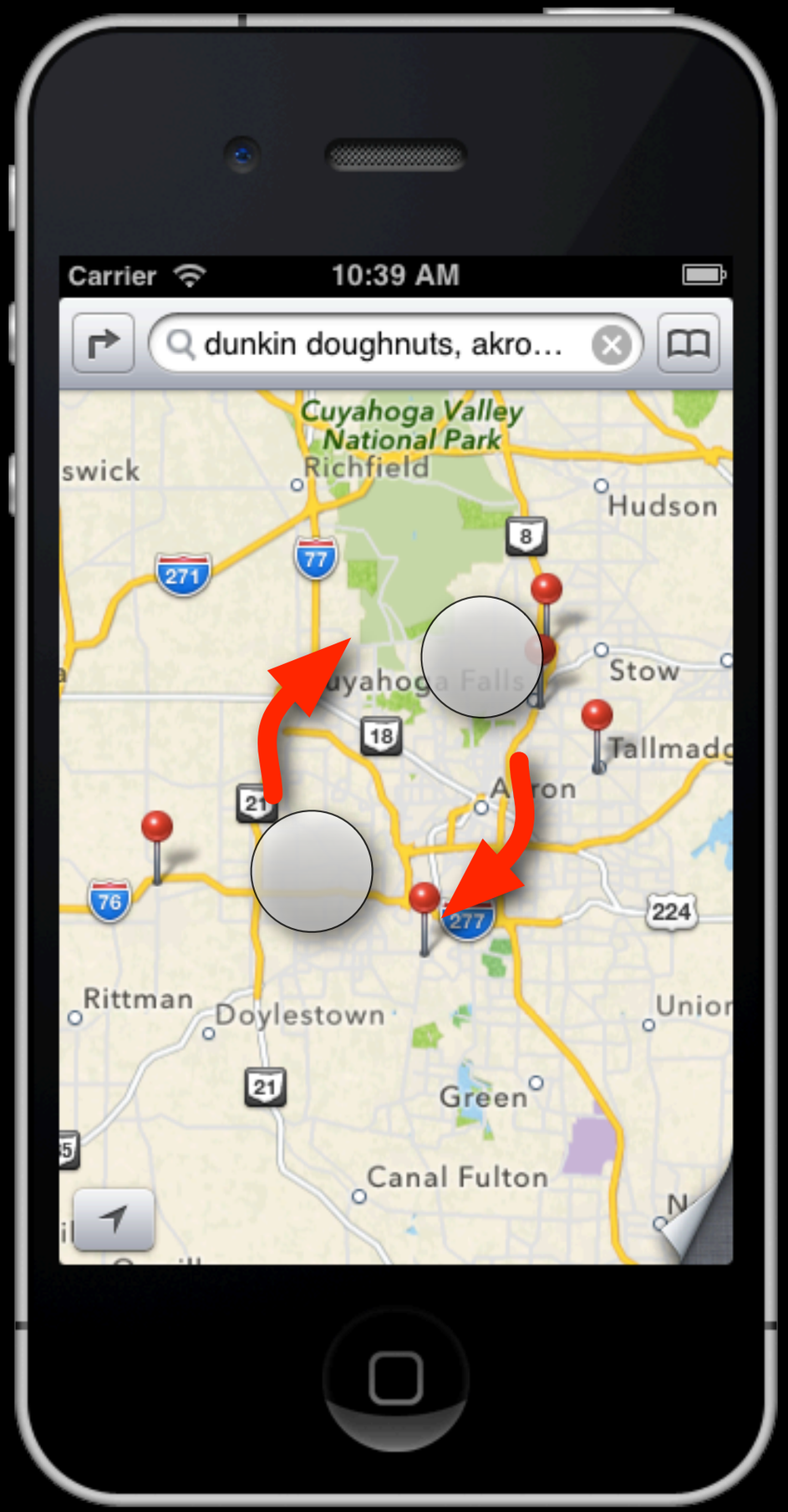
cocoamanifest.net/features

The problem...


Carrier
10:39 AM
dunkin doughnuts, akro…
Cuyahoga Valley
National Park
Richfield
Hudson
Stow
Akron
Rittman
Doylestown
Green
Canal Fulton
271
77
8
21
76
277
224


Carrier
10:39 AM
dunkin doughnuts, akro…
Cuyahoga Valley
National Park
Dunkin Donuts
4 reviews on Yelp
271
77
Cuyahoga Falls
Stow
18
Akron
21
76
277
224
Rittman
Doylestown
Green
Canal Fulton


Carrier
10:39 AM
dunkin doughnuts, akro…
Cuyahoga Valley
National Park
Richfield
Hudson
swick
271
77
8
Cuyahoga Falls
18
Tallmadge
21
Akron
277
224
Rittman
Doylestown
Union
21
Green
Canal Fulton


Carrier
10:39 AM
dunkin doughnuts, akro…
Cuyahoga Valley
National Park
Richfield
Hudson
Stow
Rittman
Doylestown
Green
Canal Fulton
271
77
8
18
21
76
277
224

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

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

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

```
- (void)touchesBegan:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    remember when, where for the touches
}
```

```
- (void)touchesMoved:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    if one finger:
        is user panning?
            adjust offset
    if two fingers:
        am I rotating? what angle?
        am I pinching? what scale?
}
```

```
- (void)touchesMoved:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    if one finger:
        is user panning?
            adjust offset
    if two fingers:
        am I rotating? what angle?
        am I pinching? what scale?
}
```

```
- (void)touchesMoved:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    if one finger:
        is user panning?
            adjust offset
    if two fingers:
        am I rotating? what angle?
        am I pinching? what scale?
}
```

```
- (void)touchesMoved:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    if one finger:
        is user panning?
            adjust offset
    if two fingers:
        am I rotating? what angle?
        am I pinching? what scale?
}
```

```
- (void)touchesMoved:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    if one finger:
        is user panning?
            adjust offset
    if two fingers:
        am I rotating? what angle?
        am I pinching? what scale?
}
```

```
- (void)touchesMoved:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    if one finger:
        is user panning?
            adjust offset
    if two fingers:
        am I rotating? what angle?
        am I pinching? what scale?
}
```

```
- (void)touchesEnded:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    did we already pan?
    did we already rotate?
    did we already pinch?
    did we move to far for a tap?
    did we wait to long for a tap?
    if not: then TAP!
}
```

```
- (void)touchesEnded:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    did we already pan?
    did we already rotate?
    did we already pinch?
    did we move to far for a tap?
    did we wait to long for a tap?
    if not: then TAP!
}
```

```
- (void)touchesEnded:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    did we already pan?
    did we already rotate?
    did we already pinch?
    did we move to far for a tap?
    did we wait to long for a tap?
    if not: then TAP!
}
```

```
- (void)touchesEnded:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    did we already pan?
    did we already rotate?
    did we already pinch?
    did we move to far for a tap?
    did we wait to long for a tap?
    if not: then TAP!
}
```

```
- (void)touchesEnded:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    did we already pan?
    did we already rotate?
    did we already pinch?
    did we move to far for a tap?
    did we wait to long for a tap?
    if not: then TAP!
}
```

```
- (void)touchesEnded:(NSSet *)touches
           withEvent:(UIEvent *)event
{
    did we already pan?
    did we already rotate?
    did we already pinch?
    did we move to far for a tap?
    did we wait to long for a tap?
    if not: then TAP!
}
```

Phew!

But wait, there's more...

What we want...

```
- (void)panned:(id)coordinates
{
    // ...
}


- (void)tapped:(id)coordinates
{
    // ...
}


- (void)doubleTapped:(id)coordinates
{
    // ...
}
```

UIView

target/action

```
- (void)tapped:
```

How do they work?

possible

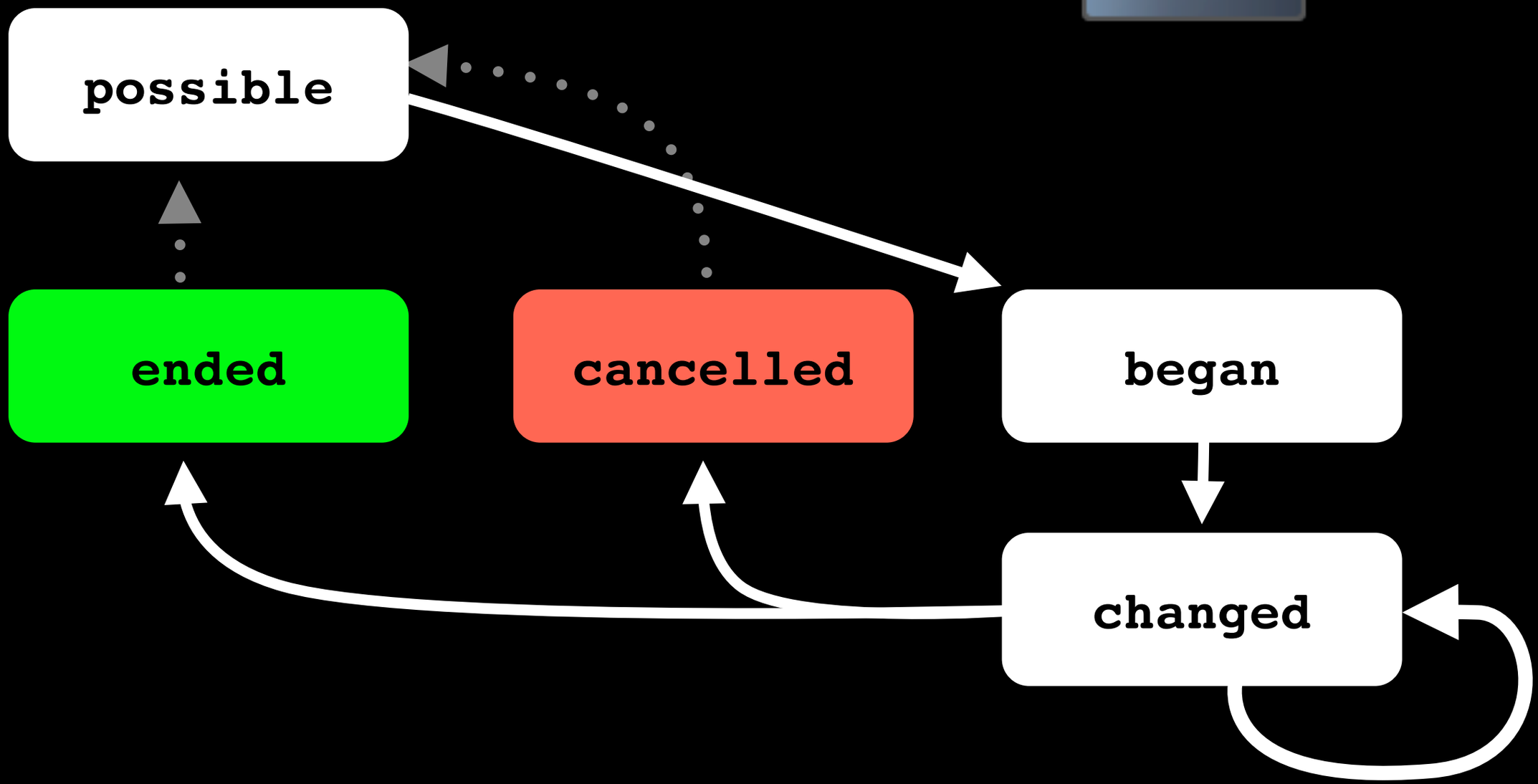
recognized

failed

possible

recognized

failed


possible
ended
cancelled
began
changed

```
- (void)panned:(id)recognizer
{
    // ...
}

- (void)tapped:(id)recognizer
{
    // ...
}

- (void)doubleTapped:(id)recognizer
{
    // ...
}
```

A long press by any other name
takes just as long.

By our powers combined....

delegate

target/action

target/action

UIViewController
<UIGestureRecognizerDelegate>

```
- (BOOL)gestureRecognizer:...
   shouldRecognizeSimultaneouslyWithGestureRecognizer:...
```

```
- (void)pinched:(id)recognizer
{
    // ...
}
```

```
- (void)rotated:(id)recognizer
{
    // ...
}
```

Reduce. Reuse. Recognize.

To the Demo!

Walkthrough Example

github.com/jonathanpenn/GestureExample

Mark's Gesture Lab

github.com/markd2/GestureLab

@jonathanpenn

jonathan@cocoamanifest.net

for more info...
cocoamanifest.net/features