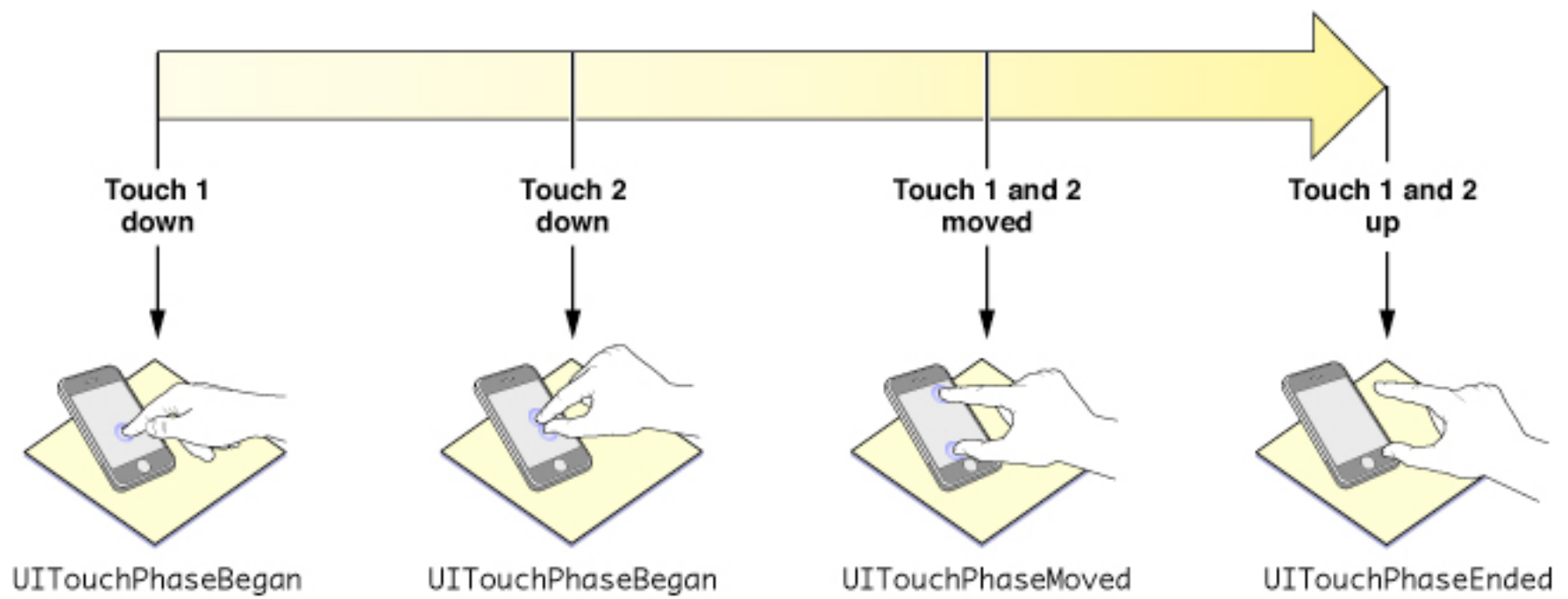


# iPhone Game Development

gestures, basic animations, transitions, UILabel

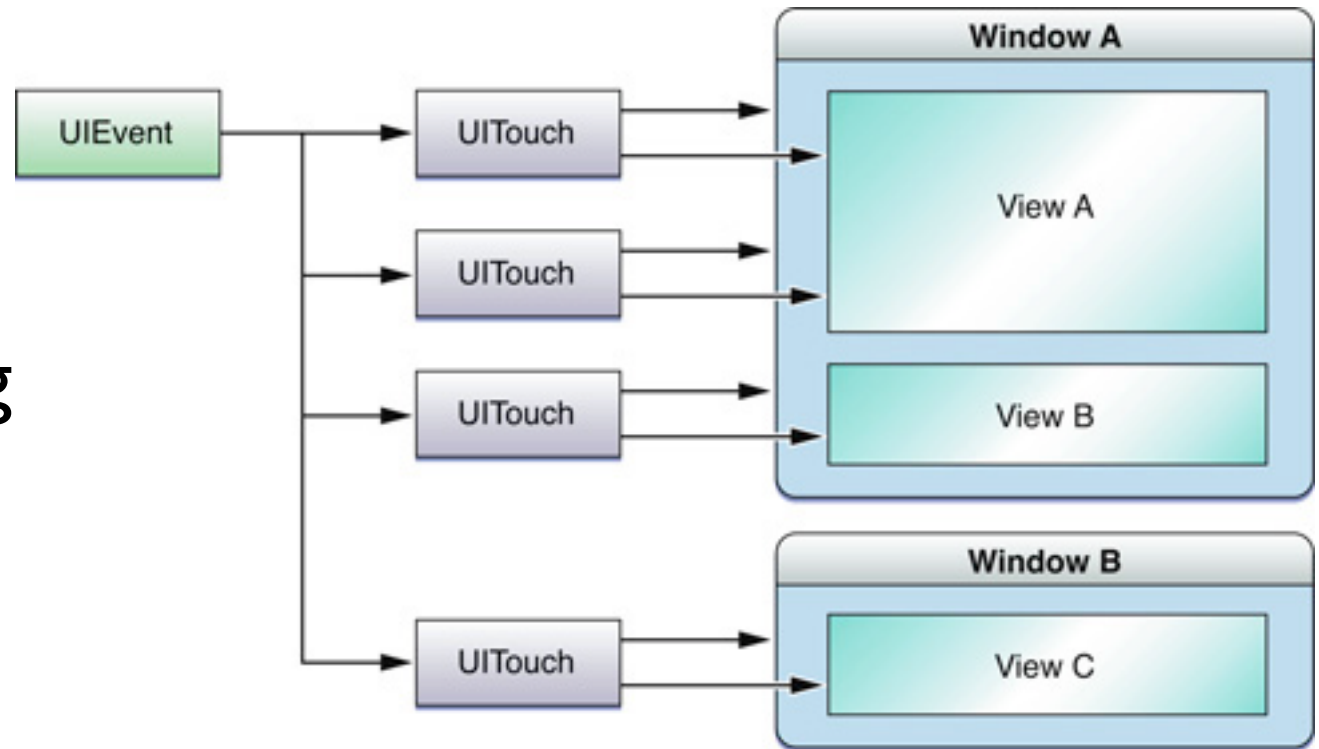
# basic touches

- touchesBegan
- touchesMoved
- touchesEnded
- touchesCanceled



# gestures

- tapping
- pinching
- panning / dragging
- swiping
- rotating
- long press



<http://developer.apple.com/library/ios/#documentation/EventHandling/Conceptual/EventHandlingiPhoneOS/GestureRecognizer/GestureRecognizer.html>

Fails to recognize gesture — all gesture recognizers



Recognizes gesture — discrete gestures

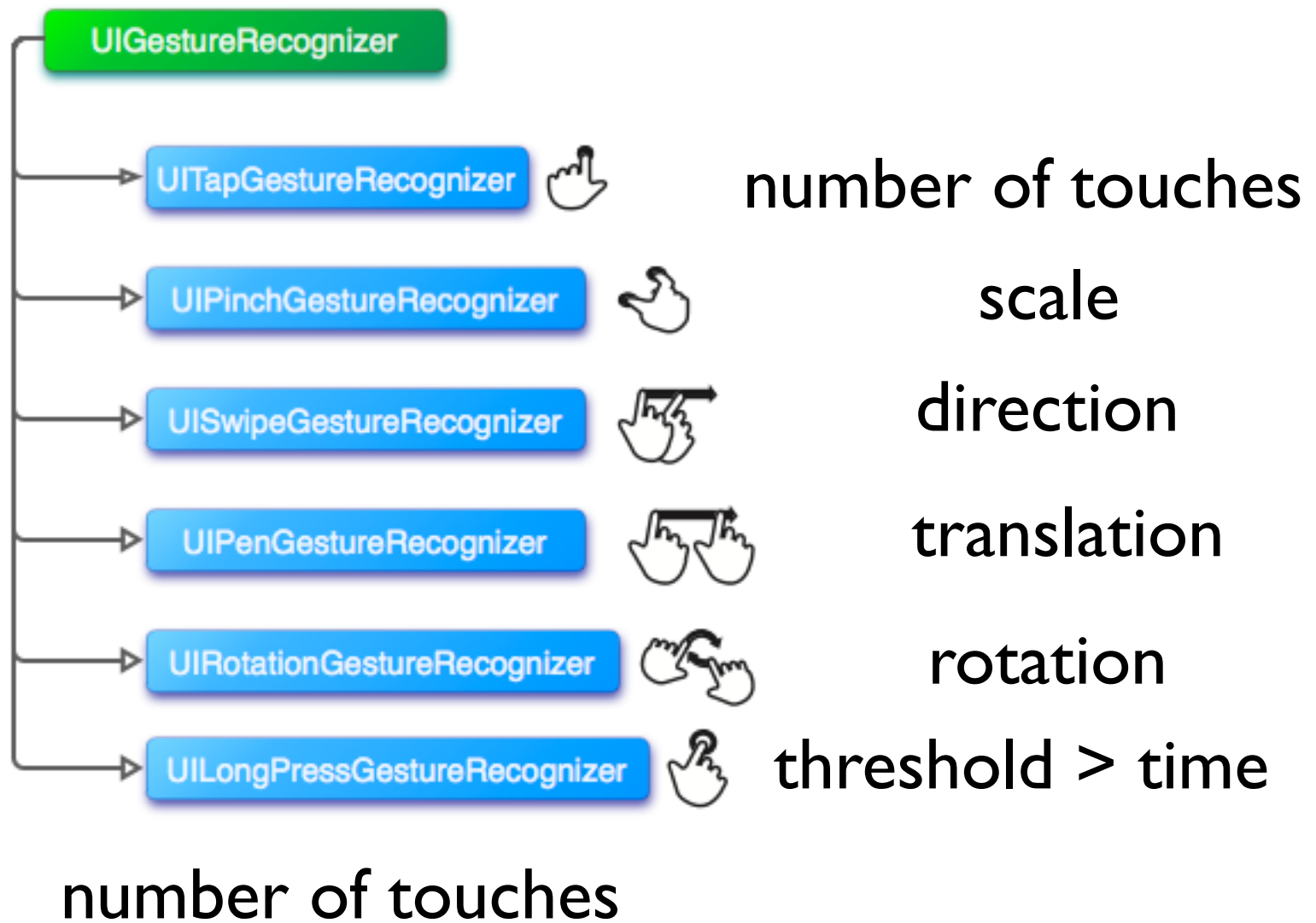


Recognizes gestures — continuous gestures



Gesture cancelled — continuous gestures





```
UISwipeGestureRecognizer *SR=[[UISwipeGestureRecognizer alloc
initWithTarget:self action:@selector(handleGesture:)];
[SR setDirection:UISwipeGestureRecognizerDirectionUp];
[self addGestureRecognizer:SR];
```

# simple animations

```
[UIView  
  animateWithDuration:0.1  
  animations:^(self.alpha=0.0;)  
  ];
```

```
animatable properties  
@property frame  
@property bounds  
@property center  
@property transform  
@property alpha  
@property backgroundColor  
@property contentStretch
```

```
[UIView animateWithDuration:1.0  
  delay:3.0  
  options:UIViewAnimationOptionCurveEaseInOut  
  animations:^(self.alpha=0.0;)  
  completion:NULL];
```

```

enum {
    UIViewAnimationOptionLayoutSubviews          = 1 << 0,
    UIViewAnimationOptionAllowUserInteraction  = 1 << 1,
    UIViewAnimationOptionBeginFromCurrentState = 1 << 2,
    UIViewAnimationOptionRepeat                 = 1 << 3,
    UIViewAnimationOptionAutoreverse            = 1 << 4,
    UIViewAnimationOptionOverrideInheritedDuration = 1 << 5,
    UIViewAnimationOptionOverrideInheritedCurve = 1 << 6,
    UIViewAnimationOptionAllowAnimatedContent   = 1 << 7,
    UIViewAnimationOptionShowHideTransitionViews = 1 << 8,

    UIViewAnimationOptionCurveEaseInOut         = 0 << 16,
    UIViewAnimationOptionCurveEaseIn           = 1 << 16,
    UIViewAnimationOptionCurveEaseOut          = 2 << 16,
    UIViewAnimationOptionCurveLinear           = 3 << 16,

    UIViewAnimationOptionTransitionNone        = 0 << 20,
    UIViewAnimationOptionTransitionFlipFromLeft = 1 << 20,
    UIViewAnimationOptionTransitionFlipFromRight = 2 << 20,
    UIViewAnimationOptionTransitionCurlUp      = 3 << 20,
    UIViewAnimationOptionTransitionCurlDown    = 4 << 20,
    UIViewAnimationOptionTransitionCrossDissolve = 5 << 20,
    UIViewAnimationOptionTransitionFlipFromTop = 6 << 20,
    UIViewAnimationOptionTransitionFlipFromBottom = 7 << 20,
};
typedef NSUInteger UIViewAnimationOptions;

```



# transitions

```
CGContextRef myContext= UIGraphicsGetCurrentContext();  
[UIView beginAnimations:nil context:myContext];  
[UIView  
    setAnimationTransition:UIViewAnimationTransitionCurlDown  
    forView:self  
    cache:YES];  
[UIView setAnimationCurve:UIViewAnimationCurveEaseInOut];  
[UIView setAnimationDuration:1.0];  
[UIView commitAnimations];
```

```
typedef enum {  
    UIViewAnimationTransitionNone,  
    UIViewAnimationTransitionFlipFromLeft,  
    UIViewAnimationTransitionFlipFromRight,  
    UIViewAnimationTransitionCurlUp,  
    UIViewAnimationTransitionCurlDown,  
} UIViewAnimationTransition;
```

```
typedef enum {  
    UIViewAnimationCurveEaseInOut,  
    UIViewAnimationCurveEaseIn,  
    UIViewAnimationCurveEaseOut,  
    UIViewAnimationCurveLinear  
} UIViewAnimationCurve;
```

# UILabel

- UILabel is a UIView with a text property

## Accessing the Text Attributes

text property

font property

textColor property

textAlignment property

lineBreakMode property

enabled property

## Sizing the Label's Text

adjustsFontSizeToFitWidth property

baselineAdjustment property

minimumFontSize property

numberOfLines property

## Managing Highlight Values

highlightedTextColor property

highlighted property

## Drawing a Shadow

shadowColor property

shadowOffset property

## Drawing and Positioning Overrides

– textRectForBounds:limitedToNumberOfLines:

– drawTextInRect:

## Setting and Getting Attributes

userInteractionEnabled property

# exercise

- make a custom view
- add gesture recognizer
- show in a UILabel the type of gesture that was applied