iPhone App Dev

Lesson 3

Resource

https://github.com/bryanttang/iOS-Class-2015-6/tree/master/lesson3

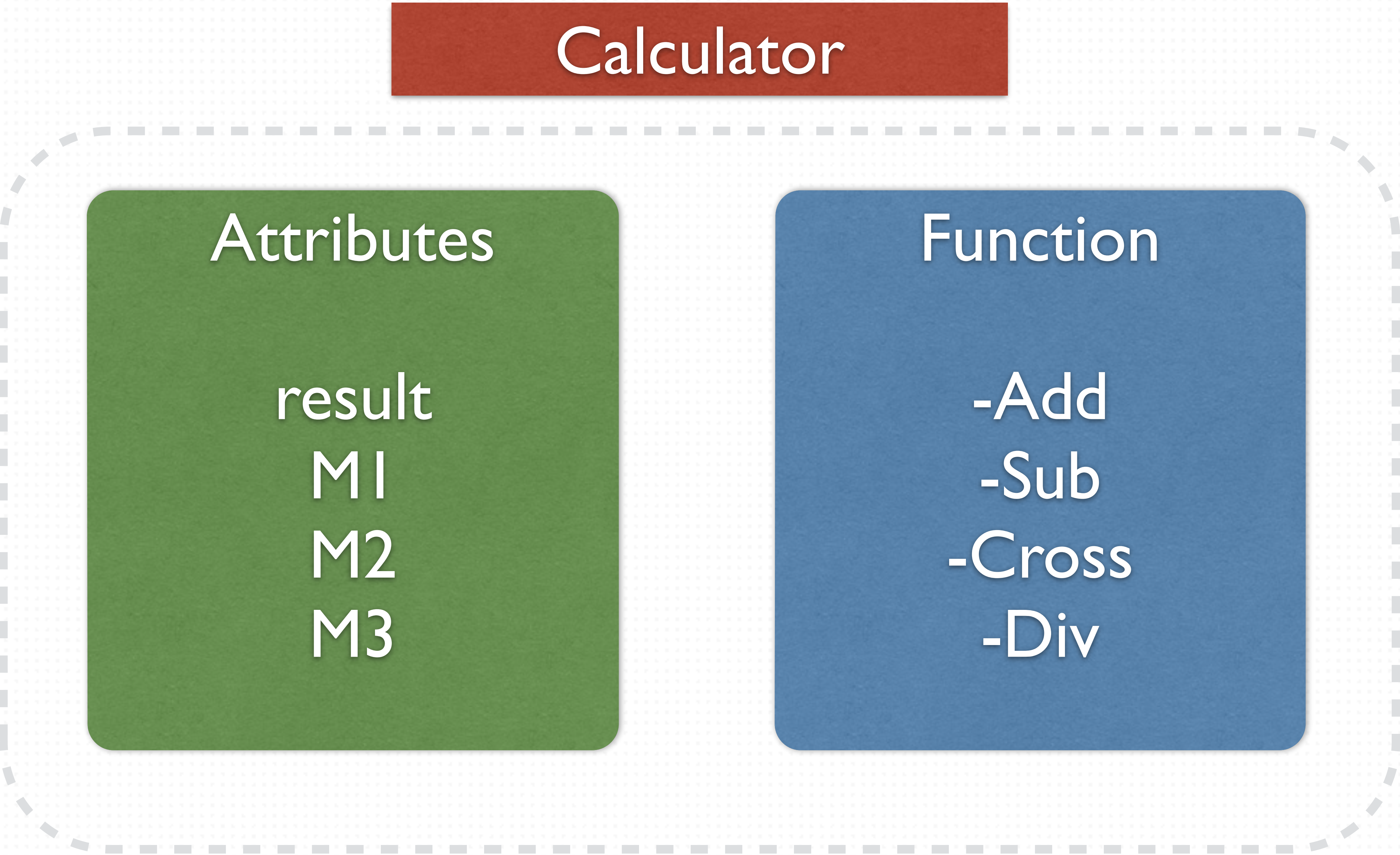
Practice

✓ Show your Converter

Summary

- Review
- Class (Advance)
- UIView
- UIViewController
- Gesture
- Animation

Basic Class



Calculator

Attributes

result
M1
M2
M3

Function

-Add
-Sub
-Cross
-Div

Class(Advance)

- Example [UIView alloc]

NSObject + alloc:

UIView + animateWithDuration: animations:

Class(Advance)

- Class Method

Declare:

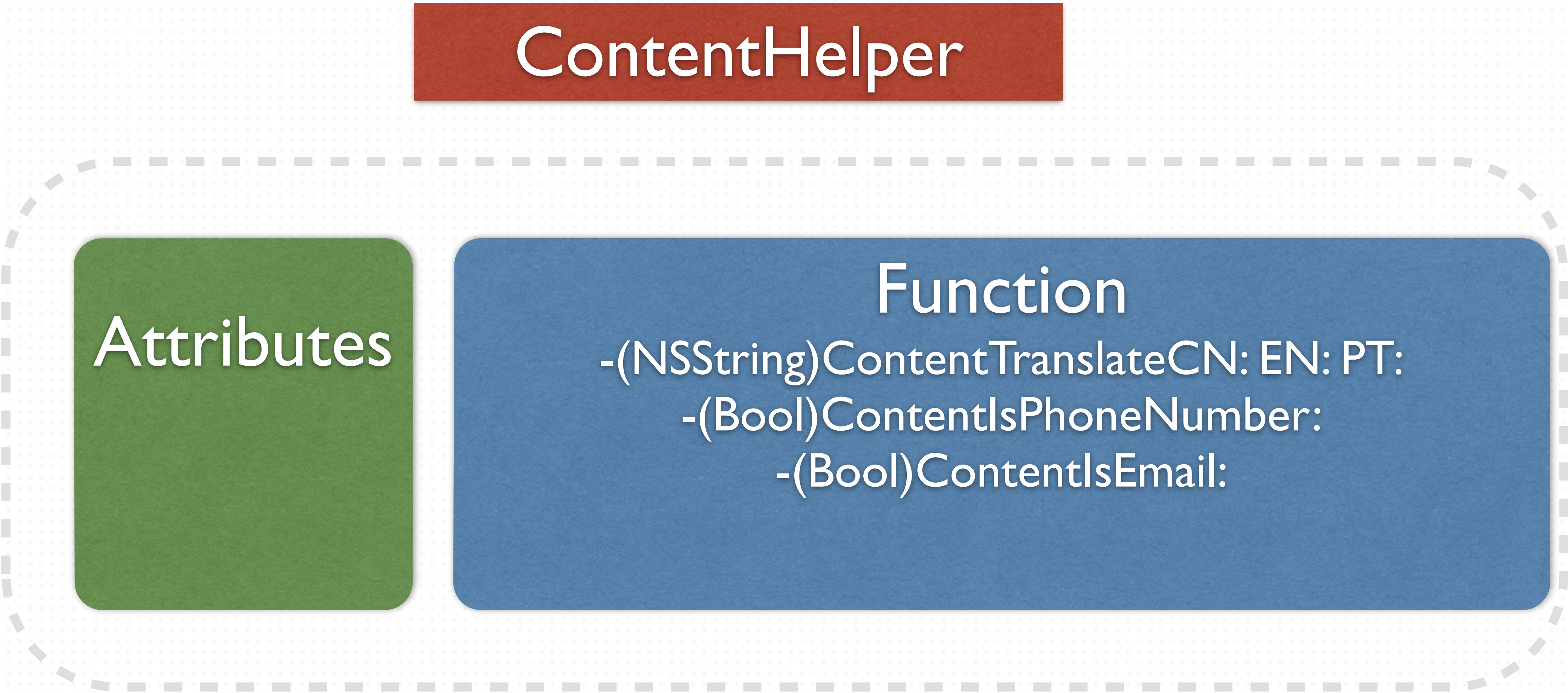
```
+ methodName:
```

Implement:

```
+ methodName:(id)params{

}
```

Class(Advance)

- Example: TranslateHelper



ContentHelper

Attributes

Function

-(NSString)ContentTranslateCN: EN: PT:

-(Bool)ContentIsPhoneNumber:

-(Bool)ContentIsEmail:

Character

- Self and Super

Car	Bus
Car’s super is Object	Bus’s super is Car
Car’s self is Car	Bus’s self is Bus

Character

- Example:

```
self.color = [UIColor blueColor];

[super init];
```

Setter & Getter

```
Car.color = [UIColor redColor];

UIColor *color = Car.color
```

Instance Property

- What is Setter and Getter exactly?
  - Getter

```
- (NSString *)something
{
 return something;
}
```

- Setter

```
- (void)setSomething:(NSString*)newSomething
{
    something = newSomething;
}
```

Instance Property

something is one of property inside ObjectA

For:

```
@property (strong) NSString *something;
    id a = ObjectA.something;
    ObjectA.something = otherthing;
```

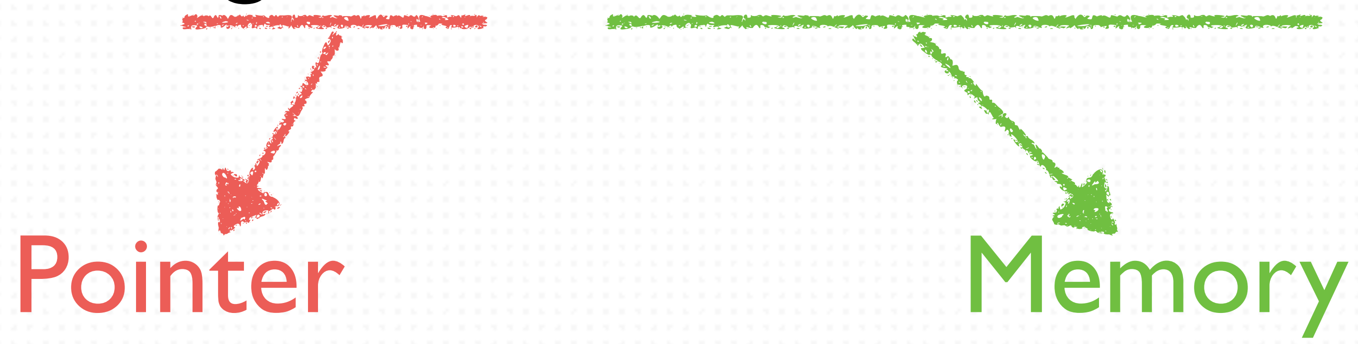
For:

```
NSString *something;
    id a = ObjectA.something;
    ObjectA.something = otherthing;
```

Variable Ownership

- Strong and Weak?

```
NSString *name = @“Jobs”
```

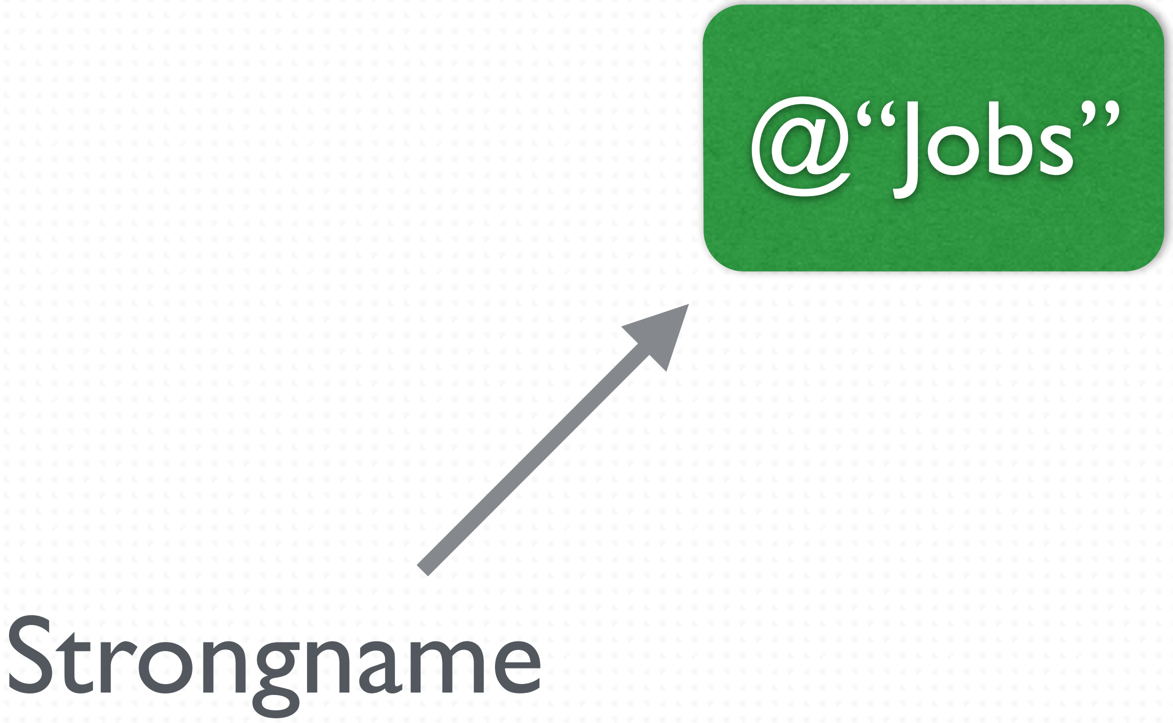


Pointer

Memory

Variable Ownership

- Strong

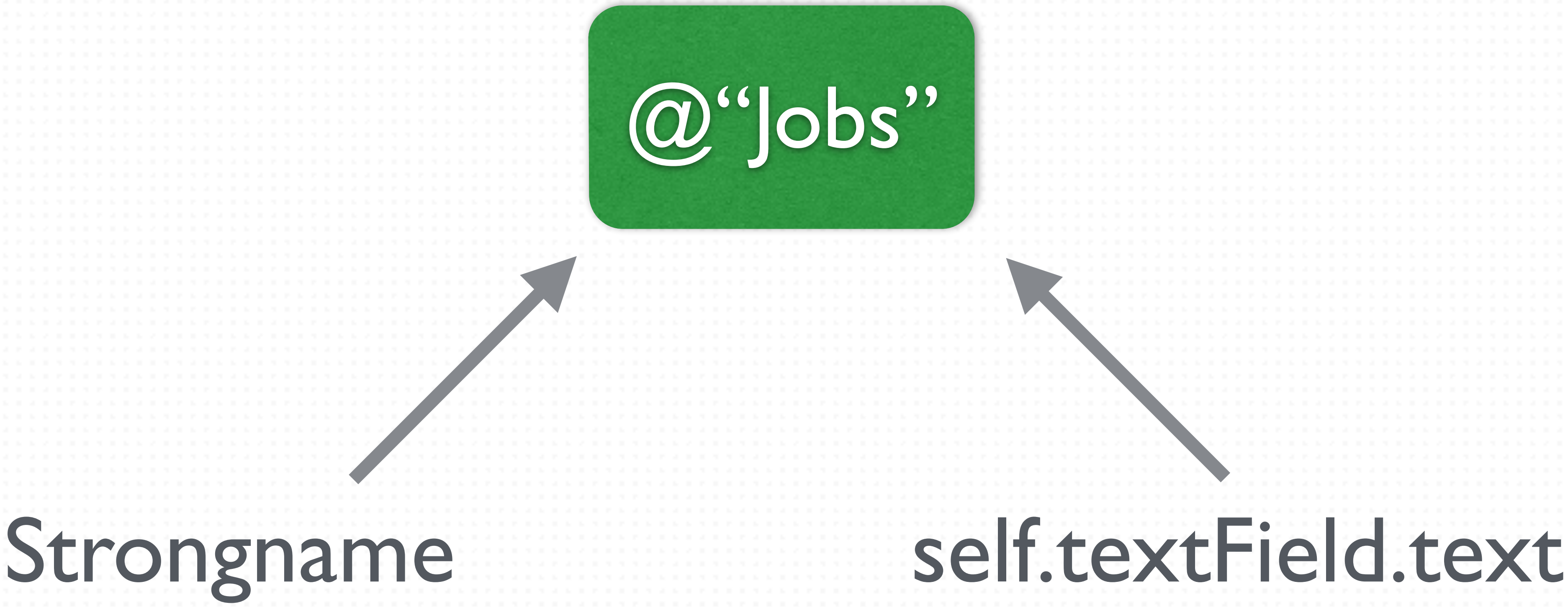


@“Jobs”

Strongname

Variable Ownership

- Strong



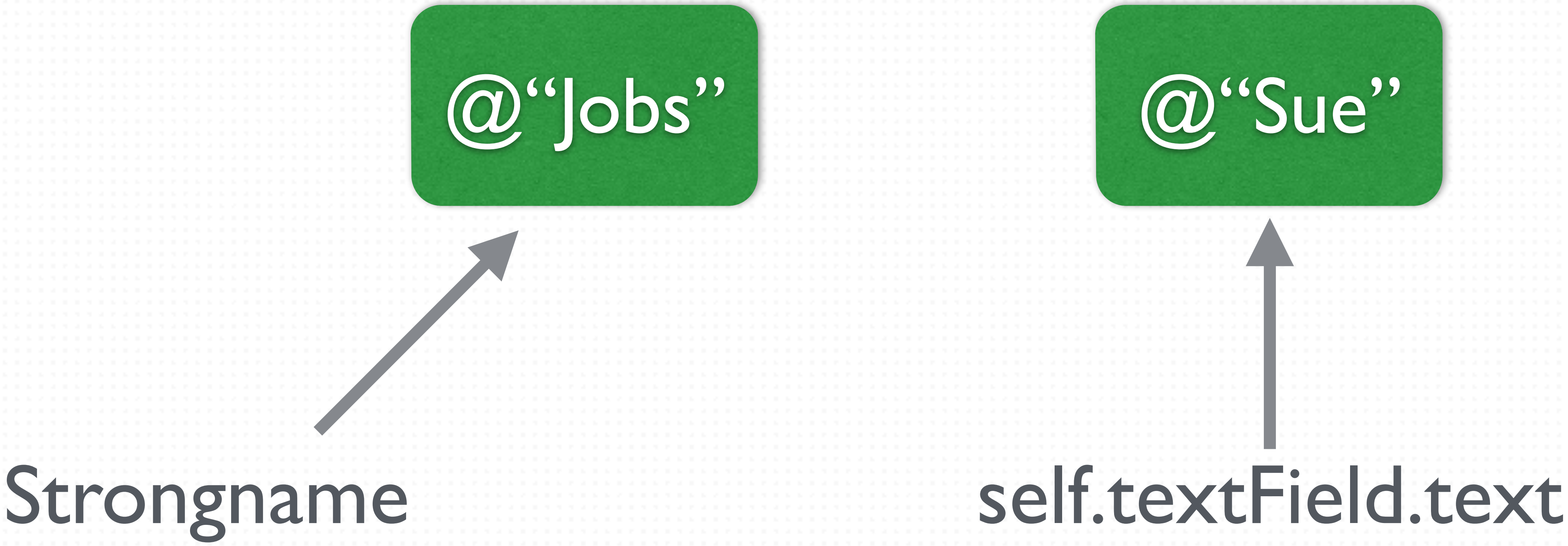
@“Jobs”

Strongname

self.textField.text

Variable Ownership

- Strong



@“Jobs”

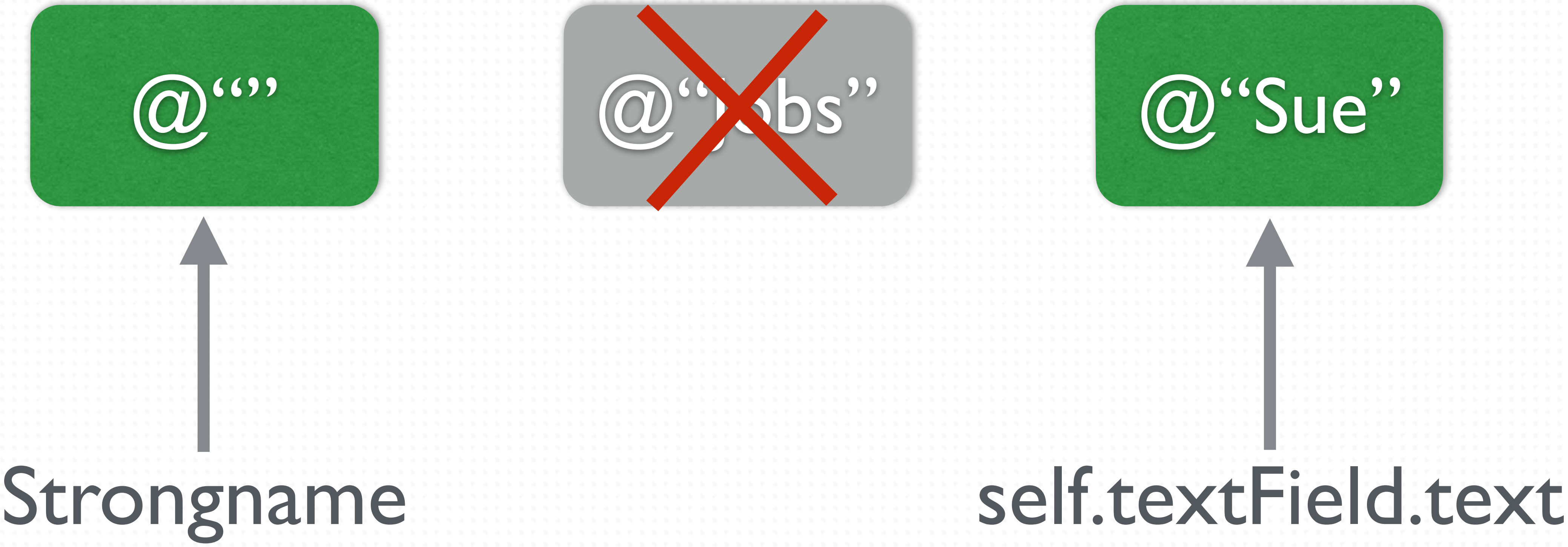
@“Sue”

Strongname

self.textField.text

Variable Ownership

- Strong



@“”

~~@“Jobs”~~

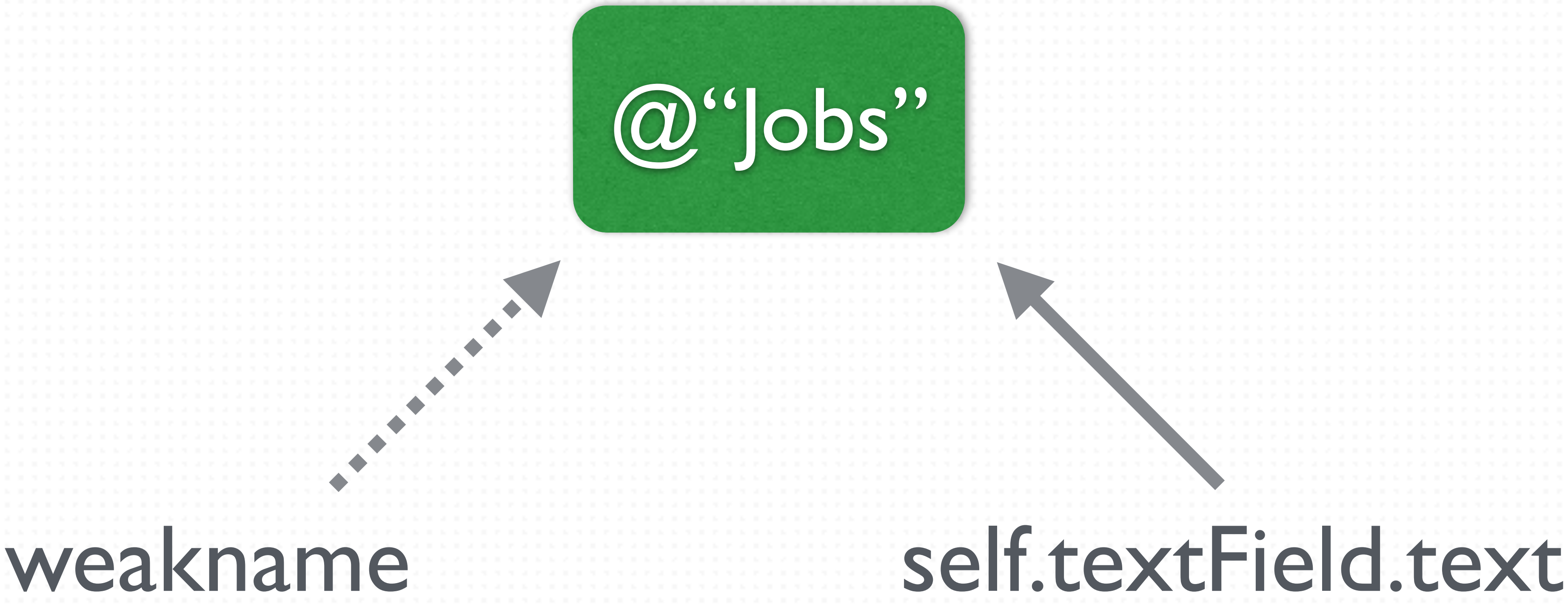
@“Sue”

Strongname

self.textField.text

Variable Ownership

- Weak



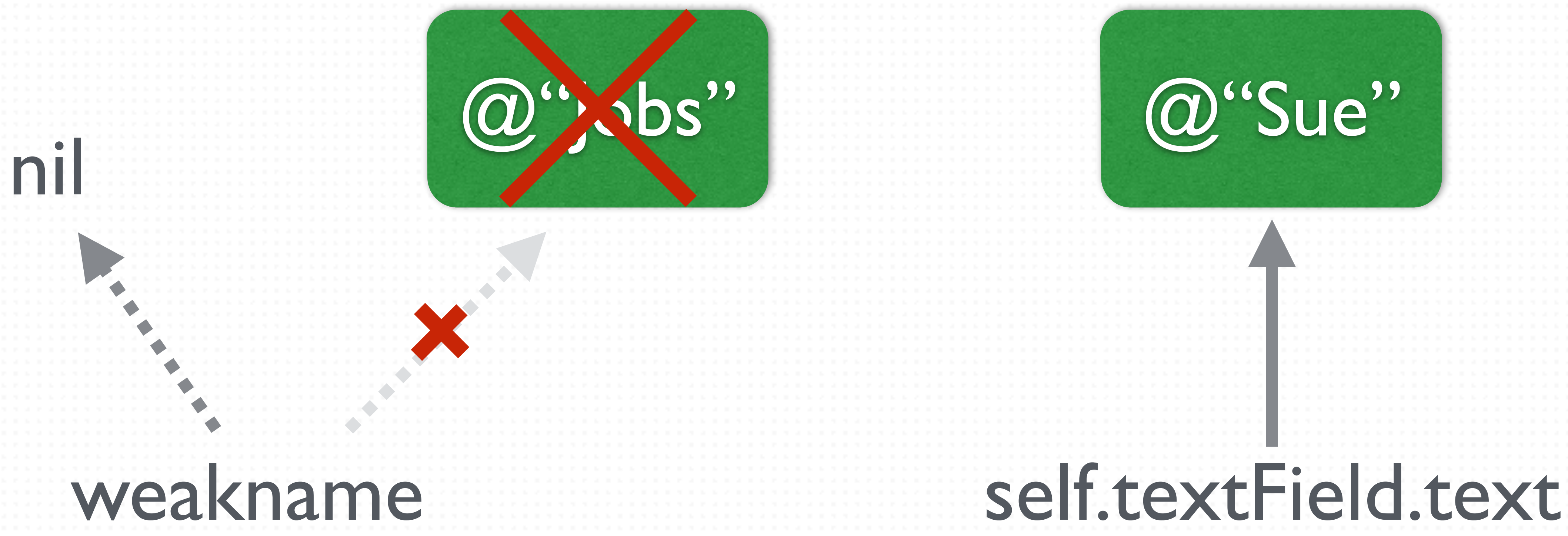
@“Jobs”

weakname

self.textField.text

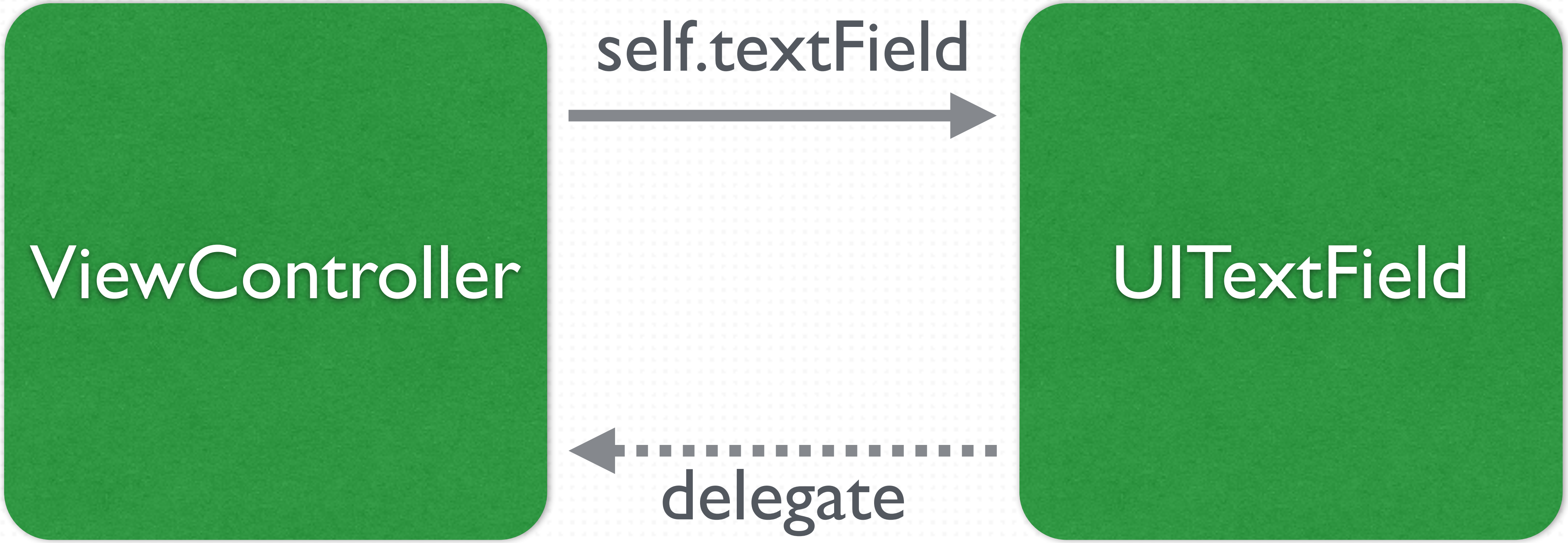
Variable Ownership

- Weak



Variable Ownership

- Example

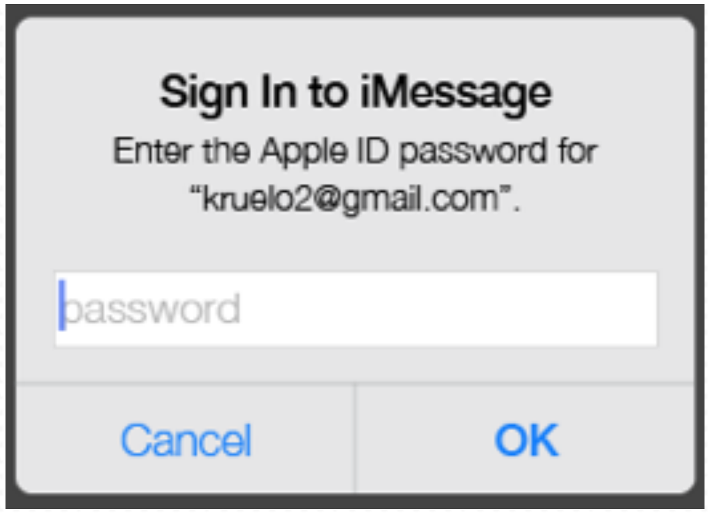


UI

UI is what you see

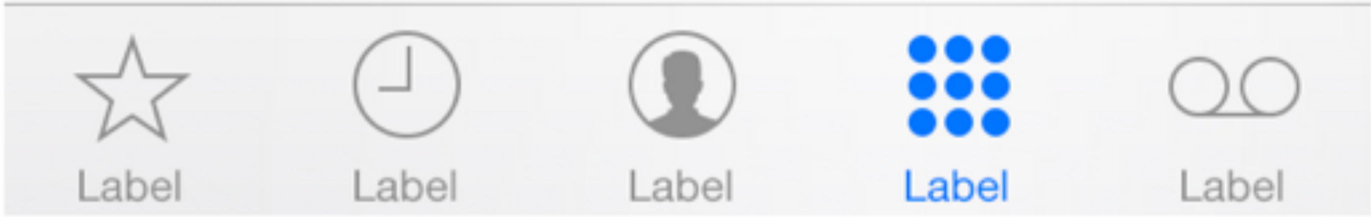
UIView

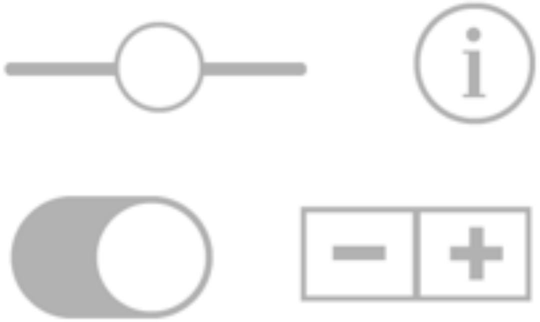
- What is UIView?

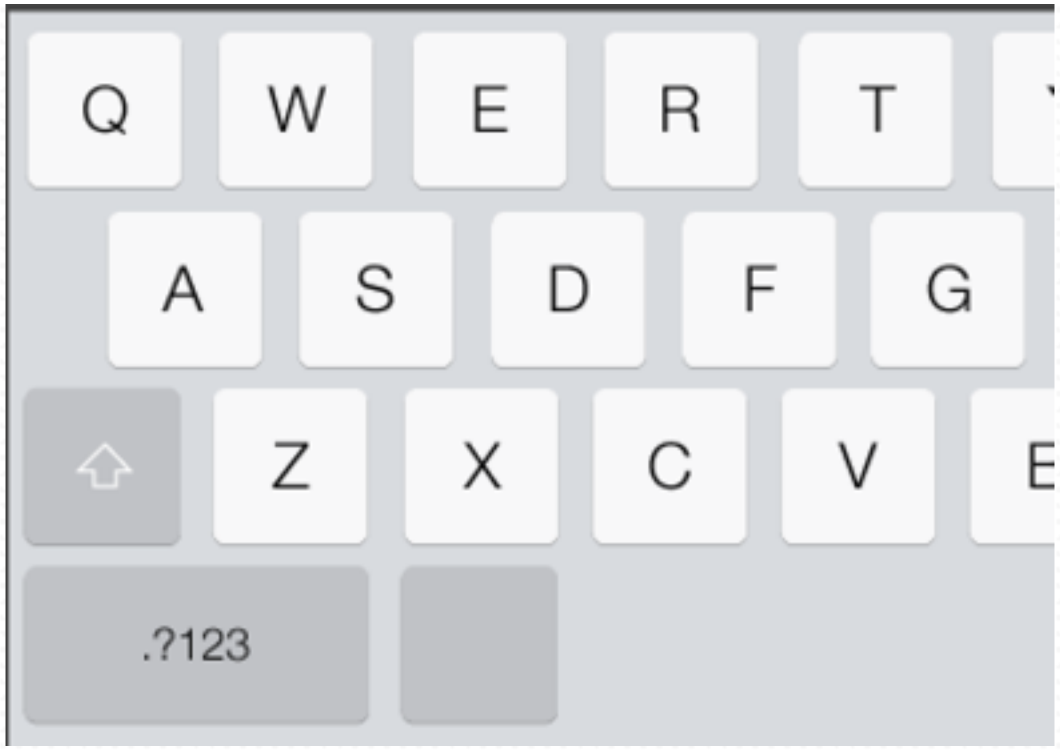


Sign In to iMessage

Enter the Apple ID password for "kruelo2@gmail.com".







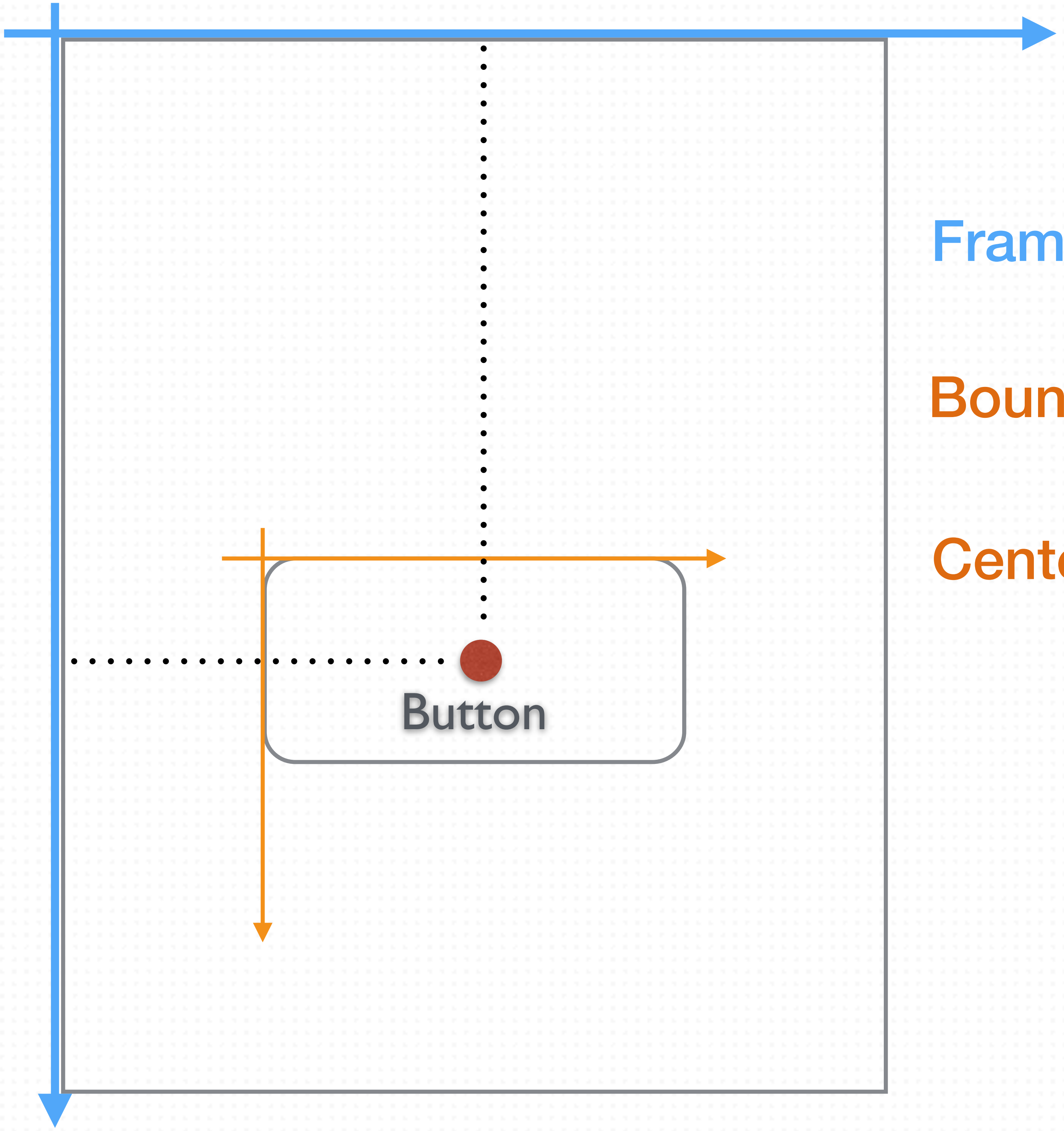
UIView

- UIButton
- UILabel
- UITextField
- UIImageView
- UITableView
- …

UIView-Attributes

- Frame (size, position), Bounds, Center
- Background color, alpha, Hidden
- Transform

Frame,Bounds,Center



Frame: (20,100, height,weight)

Bound: (0,0)

Center: (20+weight/2, 100 +height/2)

UIView-Behavior

- Method:

    -addSubview:

    -animateWithDuration: animations:

- Event: -touchesBegan: withEvent:

UIButton

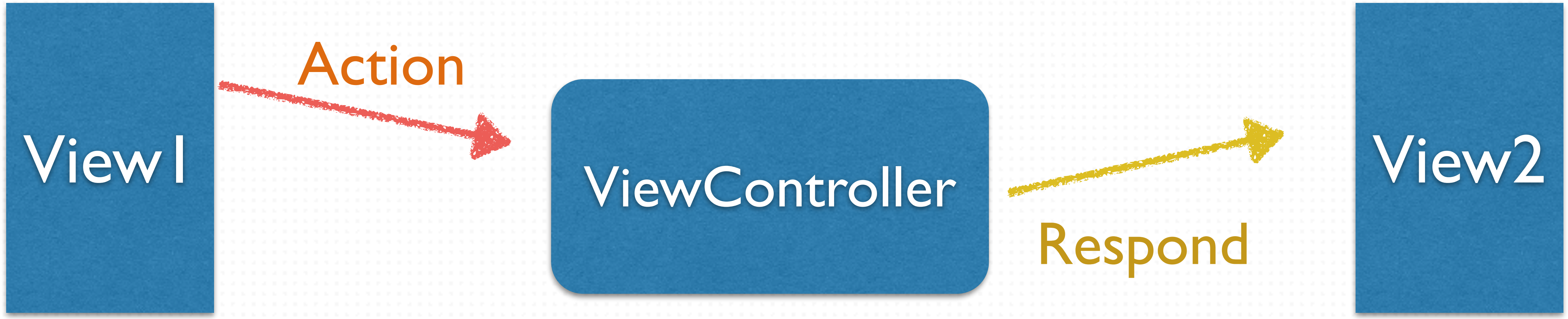
- UIView Attributes + Target (delegate)
- Method: -setTitle: forState:
- Event: -touchUpInside: ,   -touch

Controller

What would controller do?

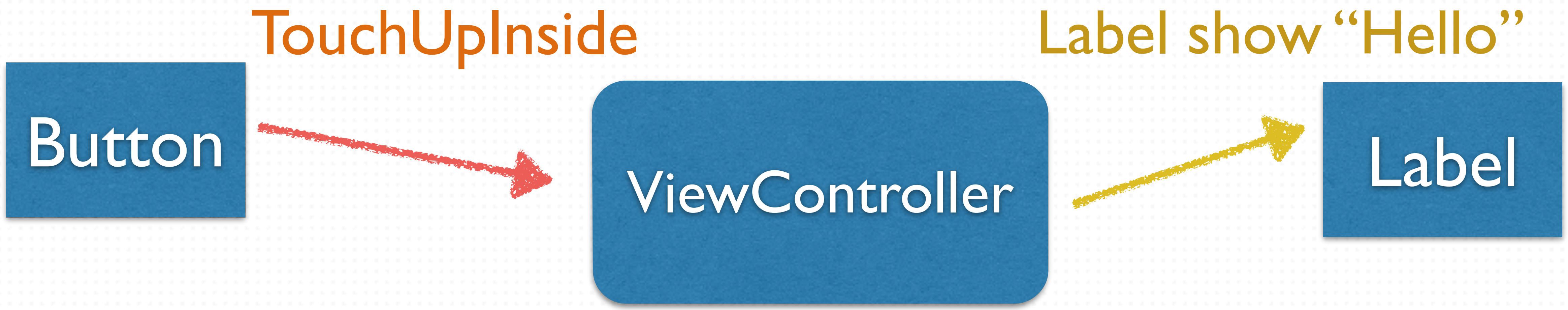
UIView & UIViewController

- How a ViewController control a view?



UIView & UIViewController

- Example: Button

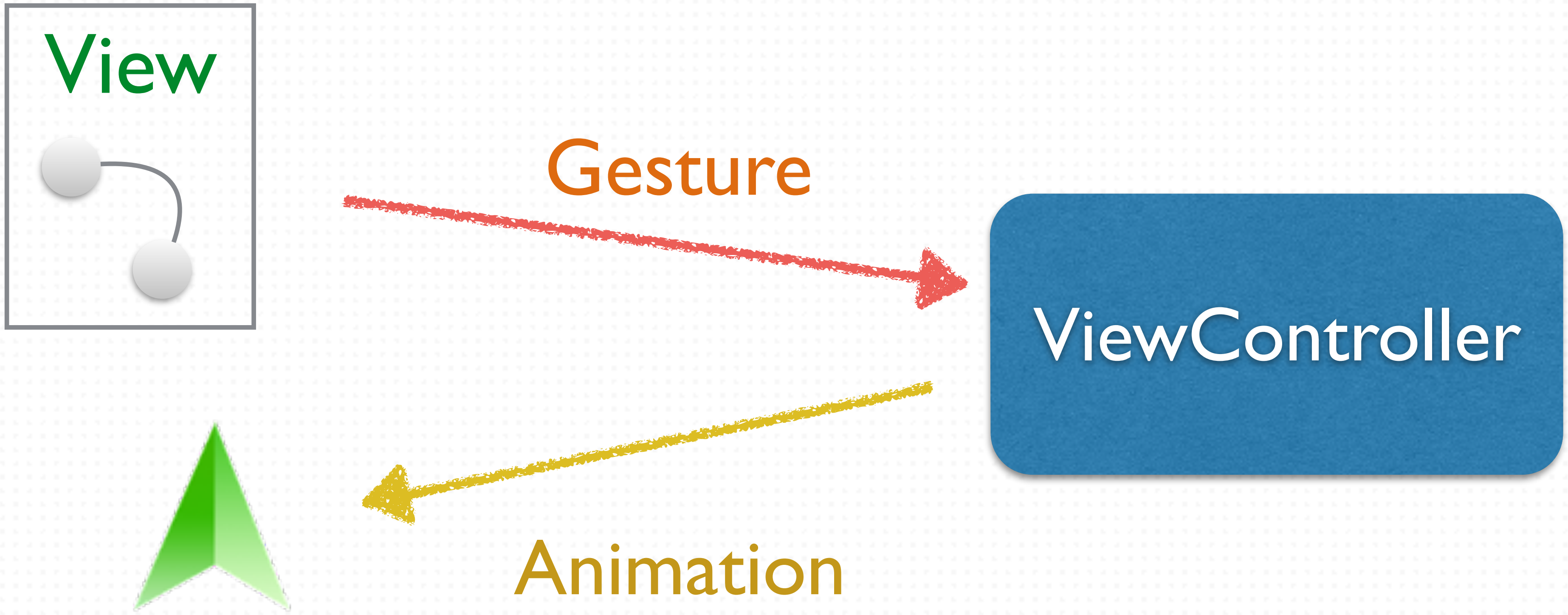


Gesture Recognition

Demo

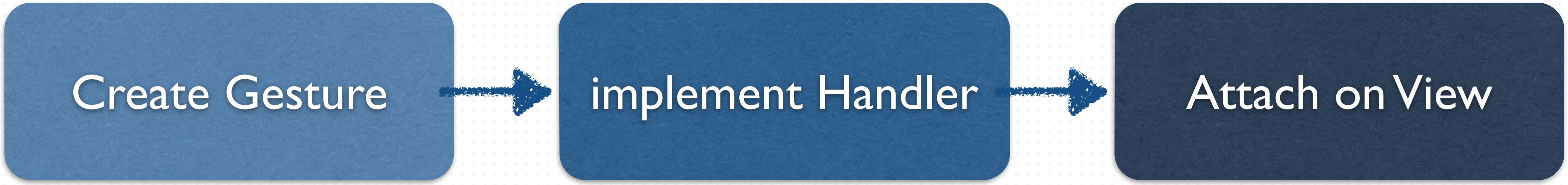
UIView & UIViewController

- Example: Gesture on



UIView & UIViewController

- How to?



Create Gesture

implement Handler

Attach on View

Gesture Recognizer

- Tap

```
UITapGestureRecognizer *tapGesture = [[UITapGestureRecognizer alloc] initWithTarget:self action:@selector(tapGestureHandler:)];
tapGesture.numberOfTapsRequired = 2;
[button addGestureRecognizer:tapGesture];
```

```
- (void)tapGestureHandler:(UIGestureRecognizer*)gestureRecognizer
{
    NSLog(@"Tap Gesture Triggered. %d fingers tapped.", gestureRecognizer.numberOfTouches);
}
```

@selector

```
@selector(sendMessage:to:)

- (void)sendMessage:(id)msg to:(id)somebody
```

@selector

```
@selector(helloWorld)

- (void)helloWorld
```

```
@selector(helloWorld:)

- (void)helloWorld:(id)param
```

Gesture Recognizer

- Long Press

```
UILongPressGestureRecognizer *longPressGesture =
[[UILongPressGestureRecognizer alloc] initWithTarget:self
action:@selector(longPressHandler:)];

[button addGestureRecognizer:longPressGesture];
```

Gesture Recognizer

- Swipe

```
UISwipeGestureRecognizer *swipeGesture = [[UISwipeGestureRecognizer alloc] initWithTarget:self action:@selector(swipeGestureHandler:)];
swipeGesture.direction = UISwipeGestureRecognizerDirectionLeft;
[self.view addGestureRecognizer:swipeGesture];
```

Gesture Recognizer

- Rotation

```
UIRotationGestureRecognizer *rotationGesture = [[UIRotationGestureRecognizer alloc] initWithTarget:self action:@selector(rotationGestureHandler:)];
[self.view addGestureRecognizer:rotationGesture];
```

```
- (void)rotationGestureHandler:(UIRotationGestureRecognizer*)gestureRecognizer
{
    float degree = gestureRecognizer.rotation * 180 / M_PI;
    NSLog(@"Rotating: %fdeg", degree);
}
```

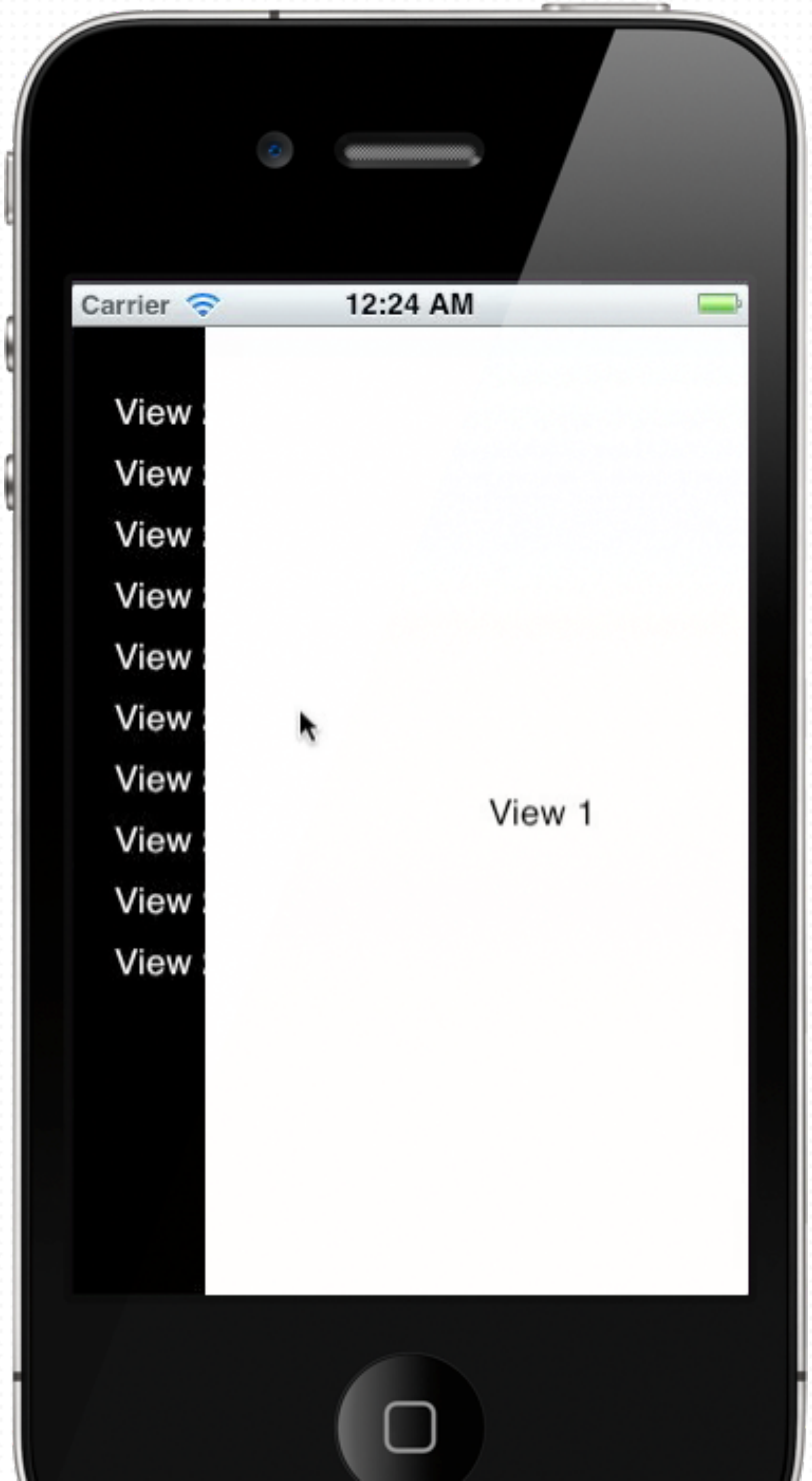
Gesture Recognizer

- Pan

```
UIPanGestureRecognizer *panGesture = [[UIPanGestureRecognizer alloc] initWithTarget:self action:@selector(panGestureHandler:)];
[self.view addGestureRecognizer:panGesture];
```

```
- (void)panGestureHandler:(UIPanGestureRecognizer*)gestureRecognizer
{
    NSString *translation = NSStringFromCGPoint([gestureRecognizer translationInView:self.view]);
    NSString *velocity = NSStringFromCGPoint([gestureRecognizer velocityInView:self.view]);
    NSLog(@"translation: %@, velocity: %@", translation, velocity);
}
```

View Panning



View Panning

- ViewDidLoad

```
- (void)viewDidLoad
{
    [super viewDidLoad];

    UIPanGestureRecognizer *panGesture = [[UIPanGestureRecognizer alloc]
initWithTarget:self action:@selector(panGestureHandler:)];
    [self.view addGestureRecognizer:panGesture];
}
```

View Panning

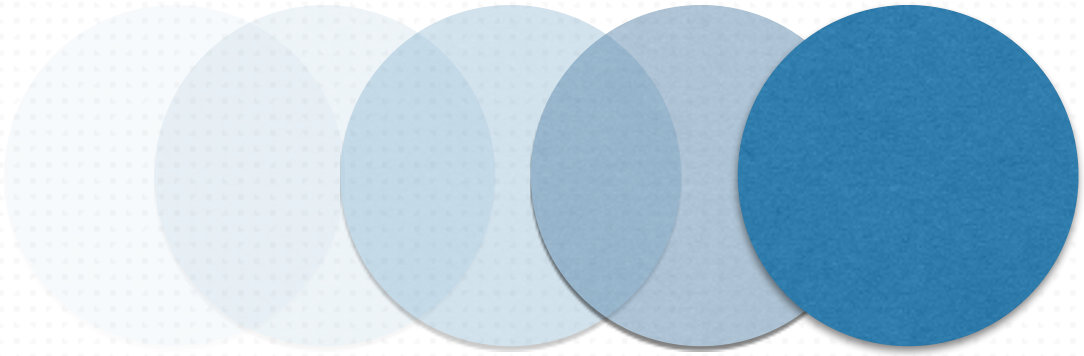
- Handling panning gesture

```
- (void)panGestureHandler:(UIPanGestureRecognizer*)gesture
{
    CGPoint translation = [gesture translationInView:self.view];
    NSLog(@"%@", NSStringFromCGPoint(translation));

    CGRect frame = self.fgView.frame;

    // gesture ended.
    if (gesture.state == UIGestureRecognizerStateEnded)
    {
        frame.origin.x = 0;
    }else {
        frame.origin.x = frame.origin.x + translation.x;
    }

    // transform the frame.
    self.fgView.frame = frame;

    [gesture setTranslation:CGPointZero inView:self.view];
}
```

Animation

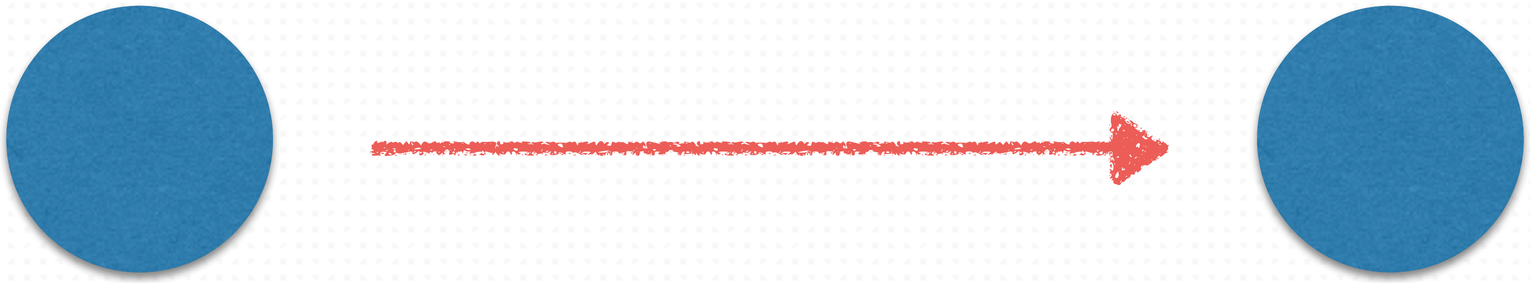
Animation

- Give UIView a life - Move!



Animation

- UIView transit from one state to other state

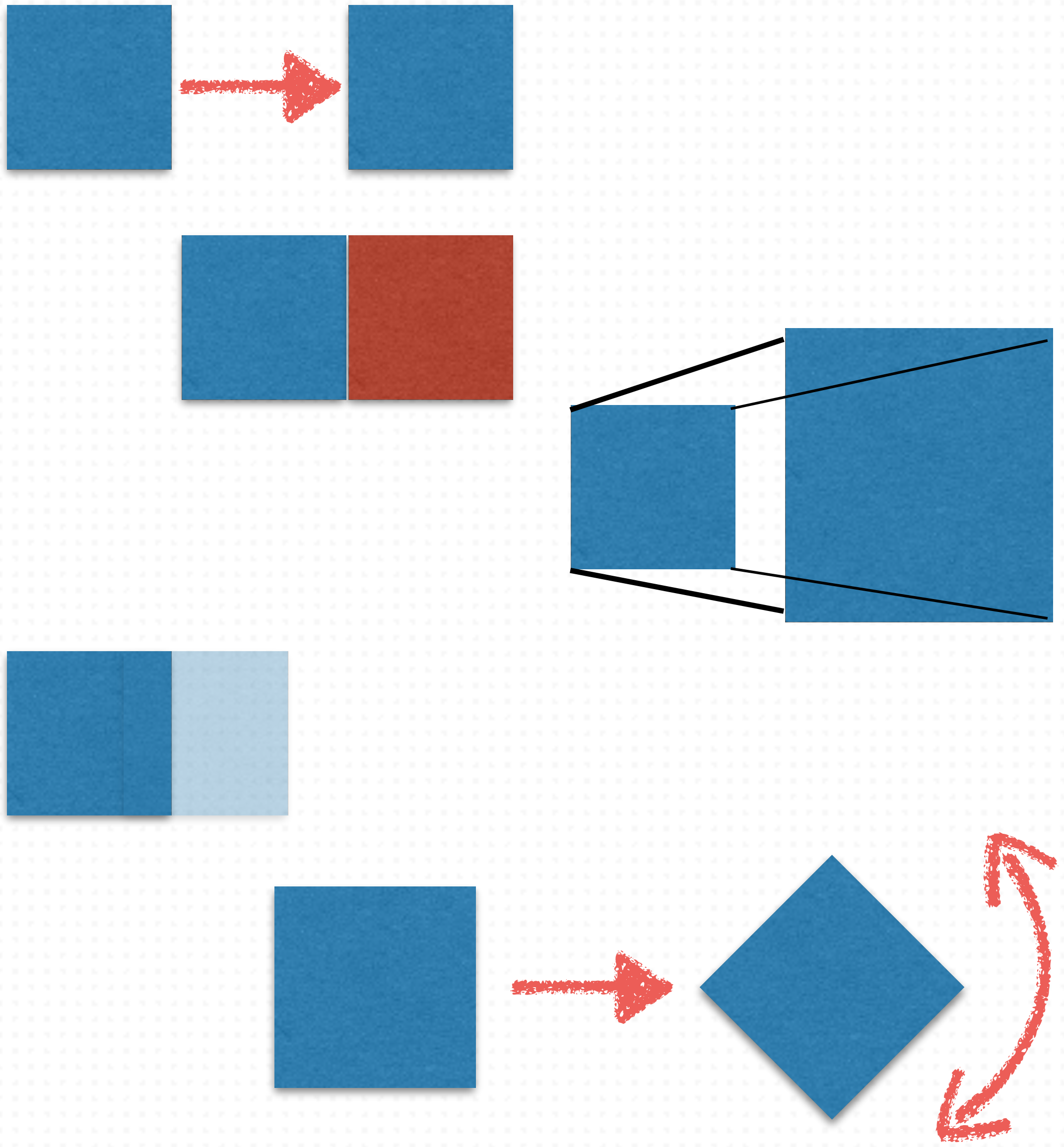


Time: t=0s t=2s

State: p(30,100) p(120,100)

Animation

- State
  - Position
  - Color
  - Scale
  - Opacity
  - Rotation



Animation

- Implement an animation on UIView

```
view.frame = CGRectMake(0,200, 40, 40);
```

```
[UIView animateWithDuration:0.4 animations:^{
    //UIView that wants to have animation
    view.frame = CGRectMake(100, 200, 40,40 );
}];
```

Animation

- Multistage animation (Nested animation)

```
[UIView animateWithDuration:0.5
                delay:0.0
               options:UIViewAnimationOptionBeginFromCurrentState
             animations:^{
                    //####  first animation  ###
             }
             completion:^(BOOL finished){[UIView animateWithDuration:0.5
                                                  delay:0.0
                                              options:UIViewAnimationOptionBeginFromCurrentState
                                              animations:^{
                                                 //#### second animation ###
                                              }
                                              completion:^(BOOL finished){//## and so on.. ##
                                              }];}];
```

Animation

- Transform
  - Translation
  - Rotation
  - Scale

Animation

- Make Translation Matrix

```
CGAffineTransform CGAffineTransformMakeTranslation ( CGFloat tx, CGFloat ty);
```

- Make Rotation Matrix

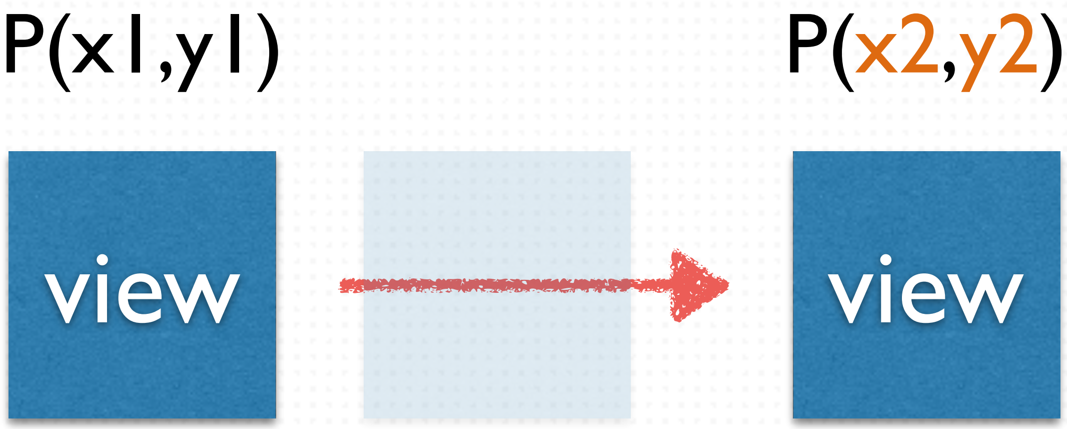
```
CGAffineTransform CGAffineTransformMakeRotation ( CGFloat angle);
```

- Make Scale Matrix

```
CGAffineTransform CGAffineTransformMakeScale ( CGFloat sx, CGFloat sy);
```

Animation

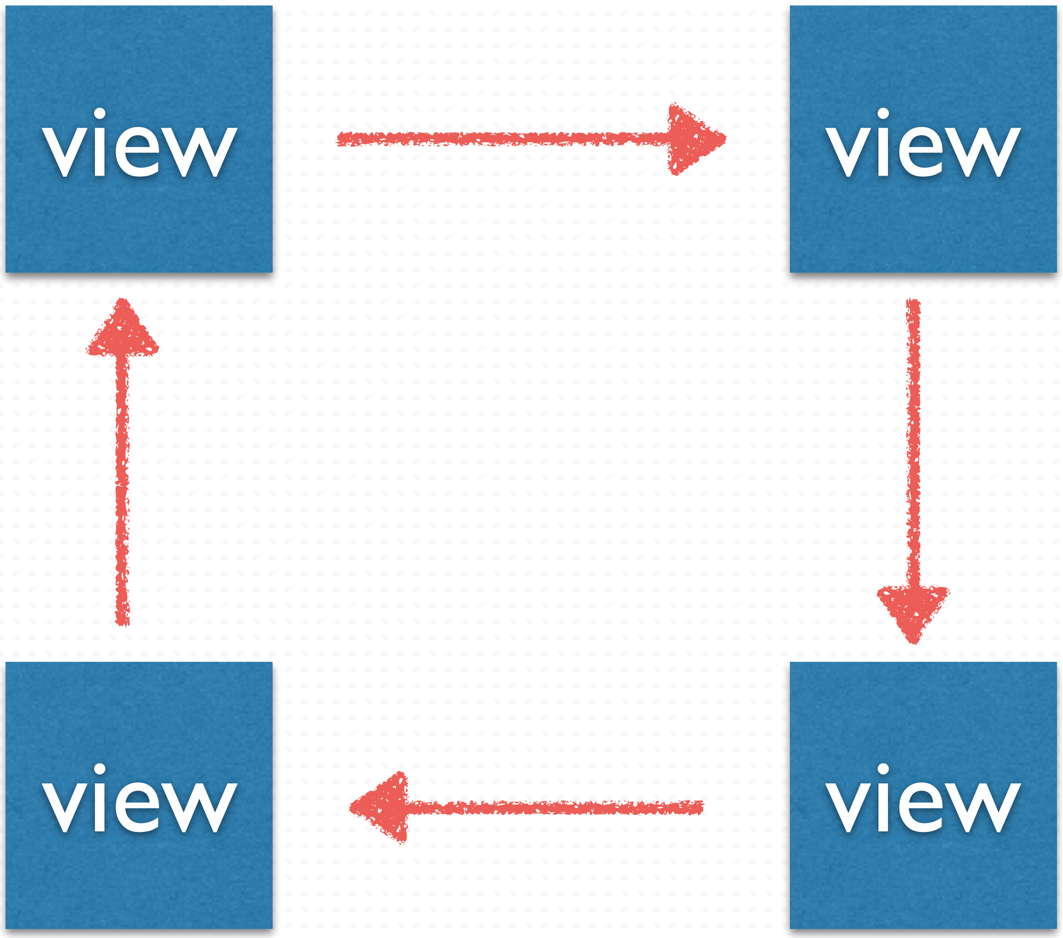
- Translation a UIView



```
view.transform = CGAffineTransformMakeTranslation(x2, y2);
```

Animation

- How about?



Exercise

- ✓ Think about how to make use of gestures.
- ✓ Design an app with gesture features and animation.
- ✓ Present it to the class in next lesson.