

~~STAssertAwesome~~  
XCTAssertAwesome

Unit testing with Xcode 5

**Why?**





- Start-Check, laut ansagen!**
- 1 Ballast und Trimmung geprüft
  - 2 Fallschirm richtig angelegt und eingehängt
  - 3 richtig angeschnallt
- Innen**
- 4 Klappen freigelegt und verriegelt, Wöbklappen
  - 5 Höhenmesser, Funk eingestellt
  - 6 Ruderkontrolle
- Außen**
- 7 Koffer ab
  - 8 Windrichtung
  - 9 auf Sicht bin ich gefasst
  - 10 Haube verriegelt, Hauben-Nutabwurf betätigt
  - 11 Startraum frei
- Volle ENTFRÄSUNG des Haubenverschlusses vor dem Start prüfen!
- Lande-Check, laut ansagen!**
- 1 Flughöhe kontrollieren
  - 2 Fahrwerk raus
  - 3 Fahrt erhöhen
  - 4 Flugzeuge im Landeanflug und auf der Landebahn



ASK 21, D-3766



ATR 720 COMM

1 2 2,3 7

Modell ASK 21  
 Werk Nr. 21469  
 Zulassung Geschwindigkeit in  
 Wölkchen  
 Zulassung Geschwindigkeit in  
 Wolken  
 Zulassung Geschwindigkeit in  
 Wolken  
 Zulassung Geschwindigkeit in  
 Wolken  
 Zulassung Geschwindigkeit in  
 Wolken





**Start-Check, laut ansagen !**

Person

- 1 Ballast und Trimmung geprüft
- 2 Fallschirm richtig angelegt und eingehängt
- 3 richtig angeschnallt

Innen

- 4 Klappen freigängig und verriegelt, Wölbklappen
- 5 Höhenmesser, Funk eingestellt
- 6 Ruderkontrolle

ausßen

- 7 Kuller ab
- 8 Windrichtung
- 9 auf Seilriß bin ich gefaßt
- 10 Haube verriegelt, Hauben-Notabwurf bekannt
- 11 Startraum frei

Volle Einrastung des Haubenverschlusses vor dem Start prüfen!

**Lande-Check, laut ansagen !**

- 1 Flughöhe kontrollieren
- 2 Fahrwerk raus
- 3 Fahrt erhöhen
- 4 Flugzeuge im Landeanflug und auf der Landebahn

# Release checklist

- App does not crash on start
- App is compatible with iOS 6, iOS 5, ...
- App behaves well-defined when internet connection is slow
- ...

**How to automate this?**

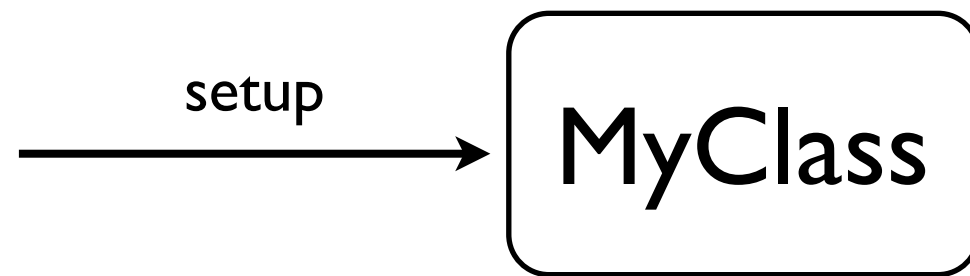
# State tests

# State tests

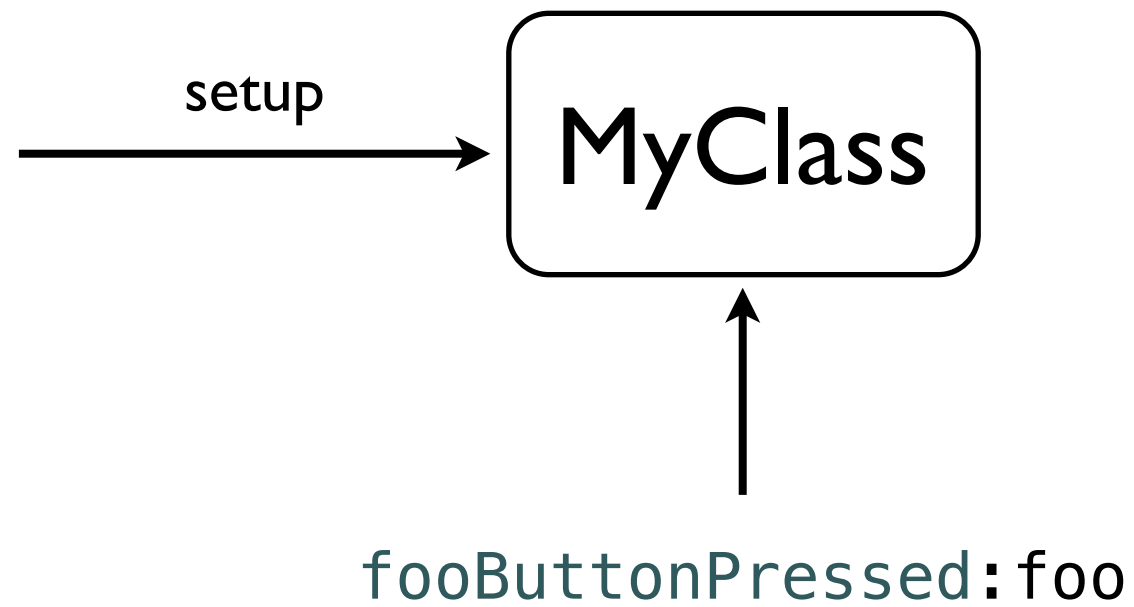
MyClass



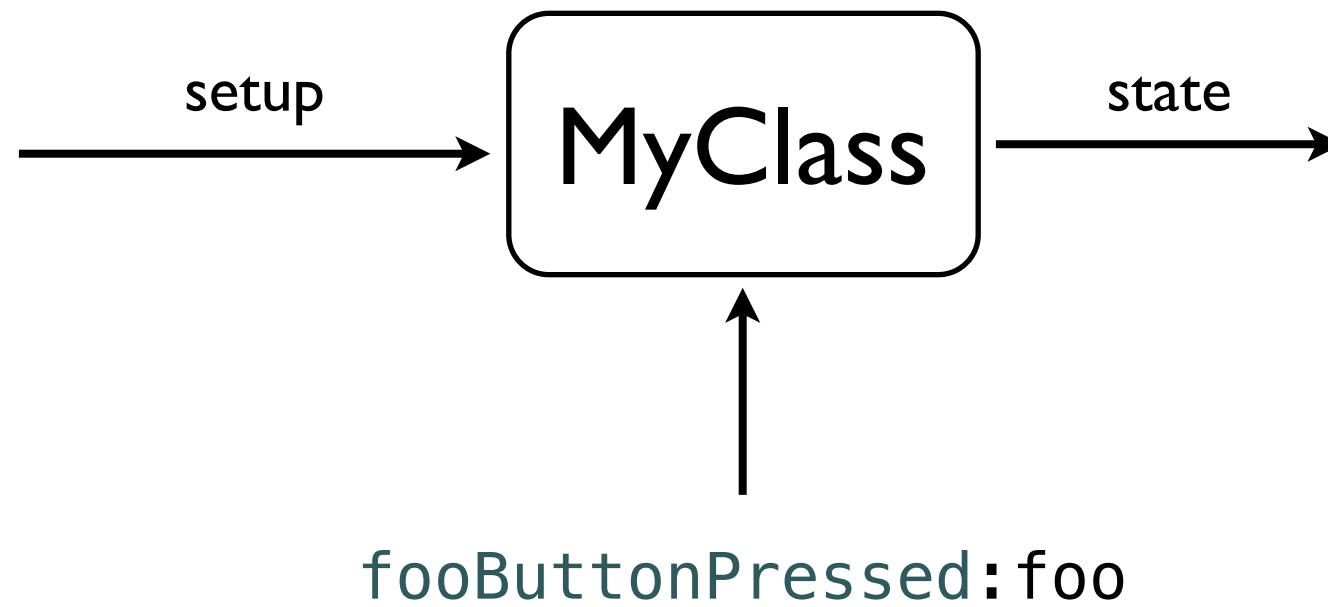
# State tests



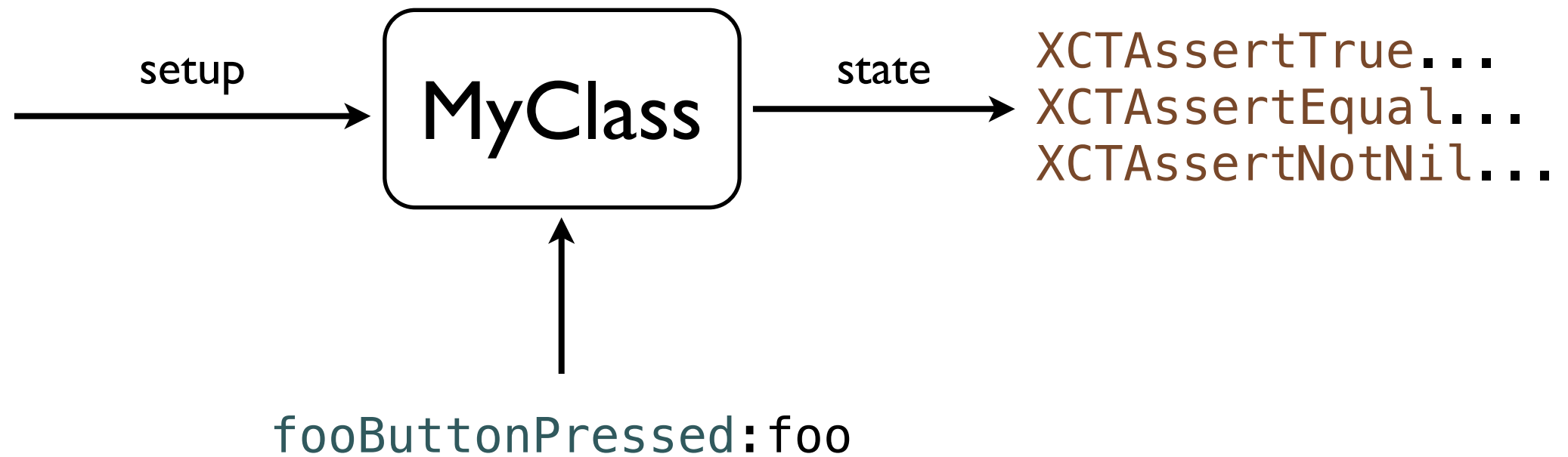
# State tests



# State tests



# State tests





**Demo**

**An object is not an  
island...**

# Behavior tests

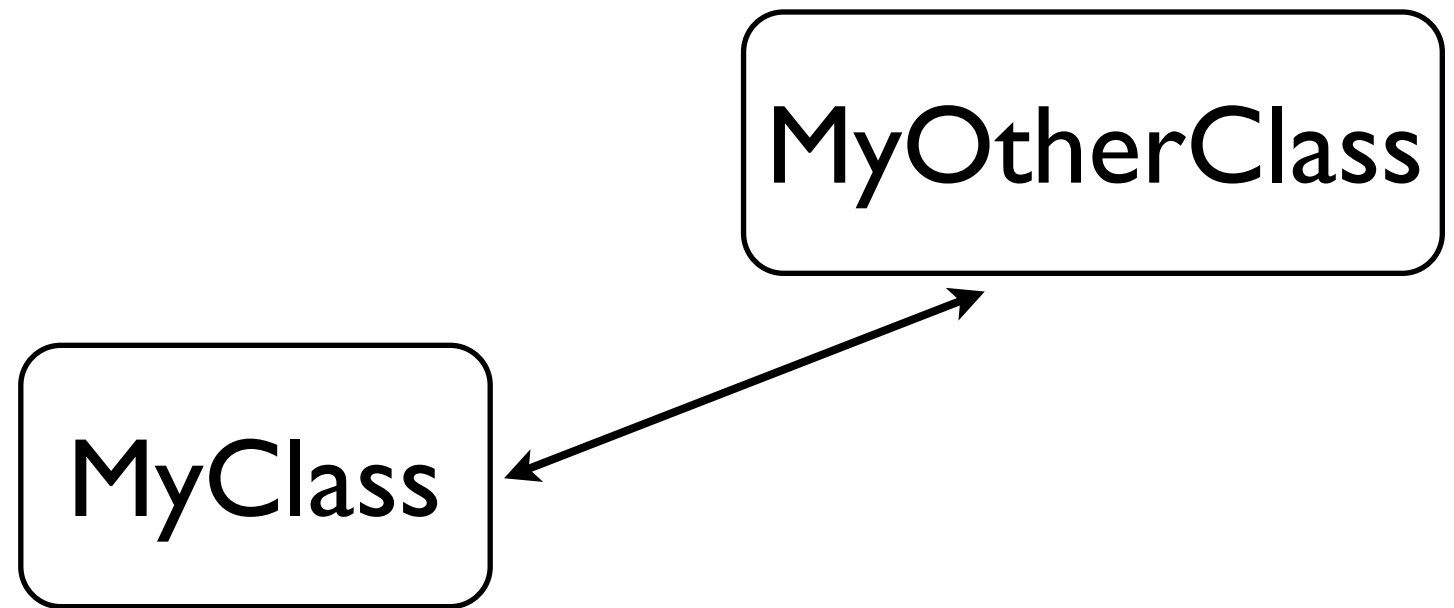
# Behavior tests



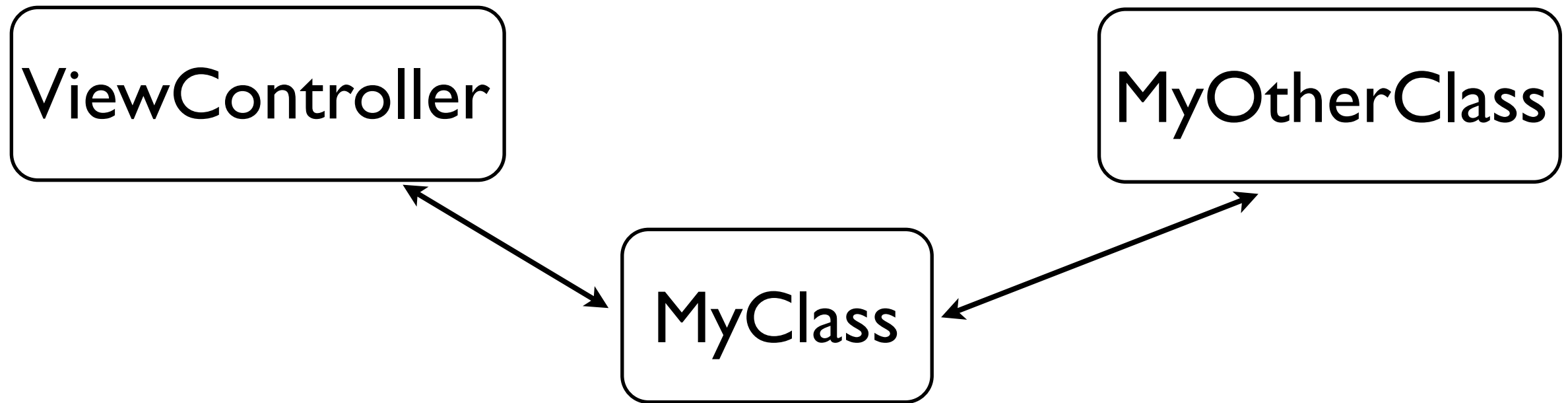
MyClass



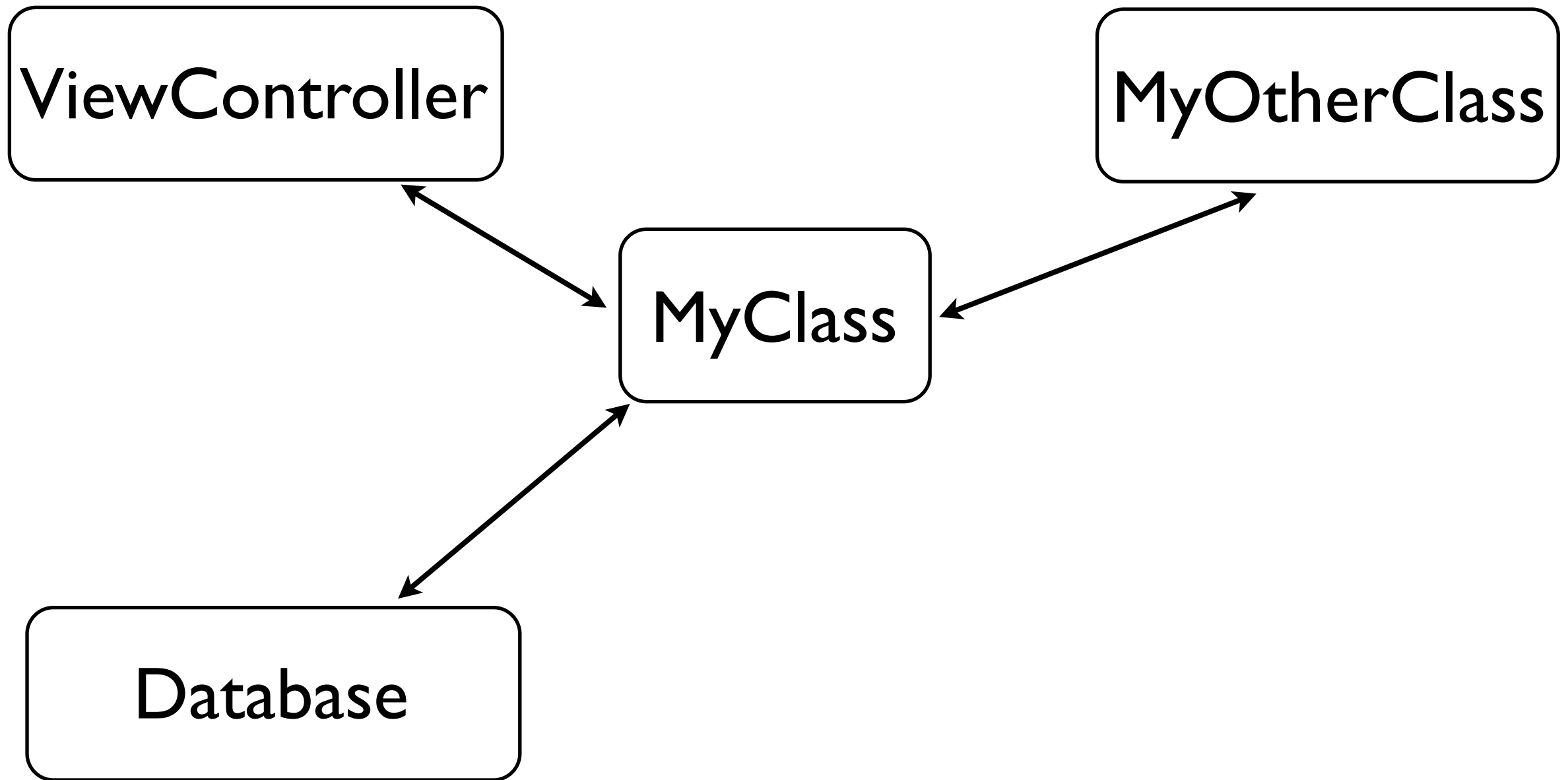
# Behavior tests



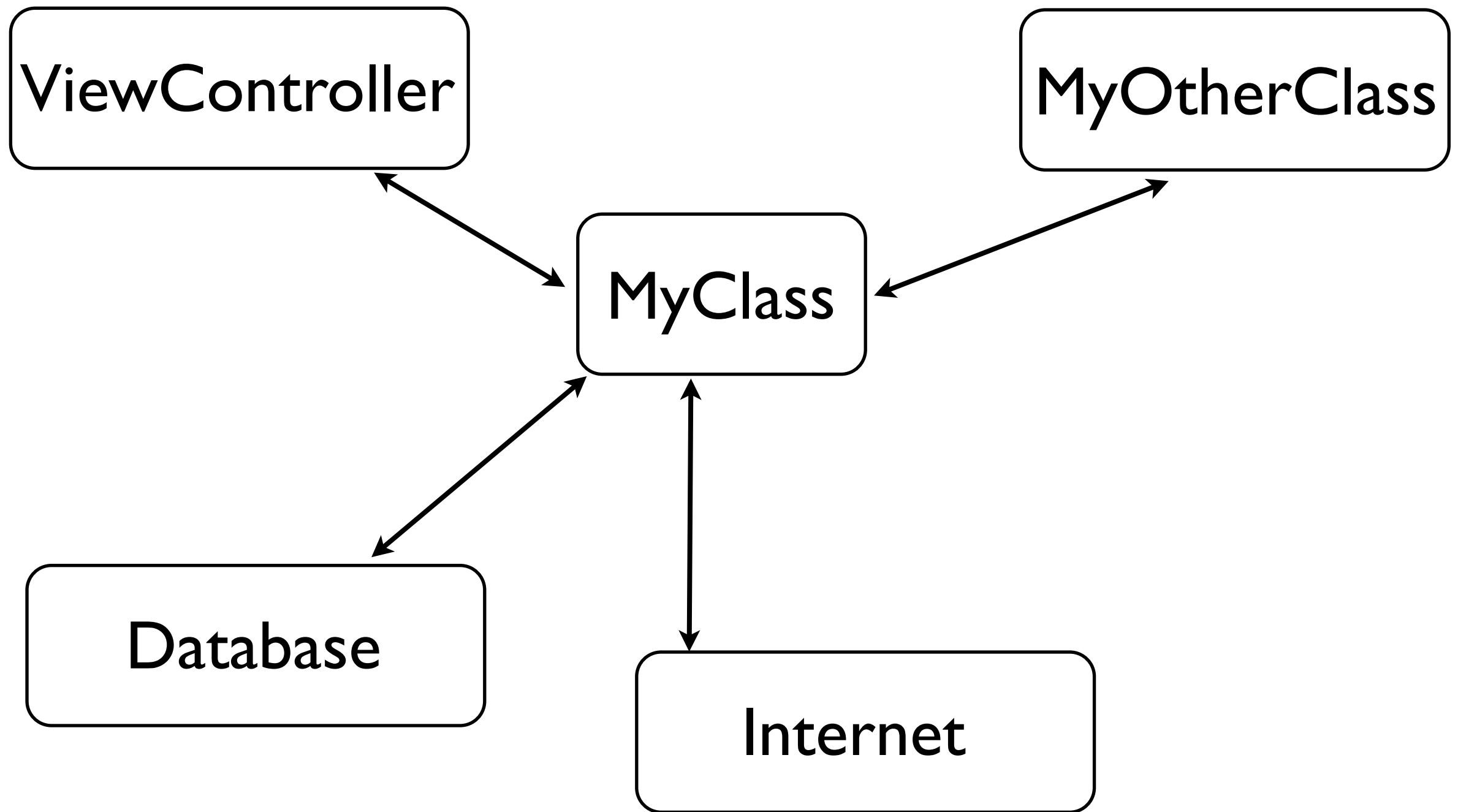
# Behavior tests



# Behavior tests

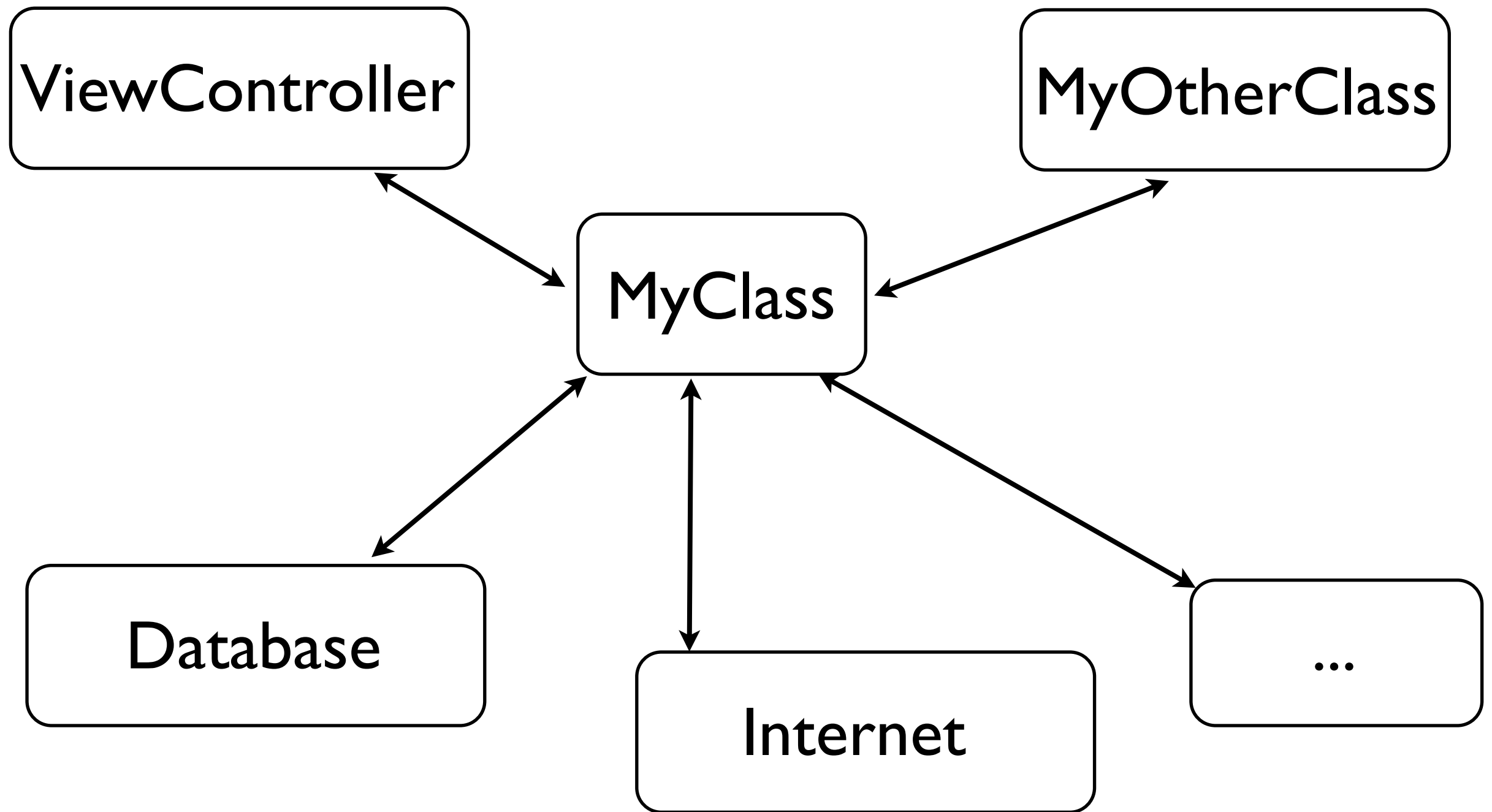


# Behavior tests

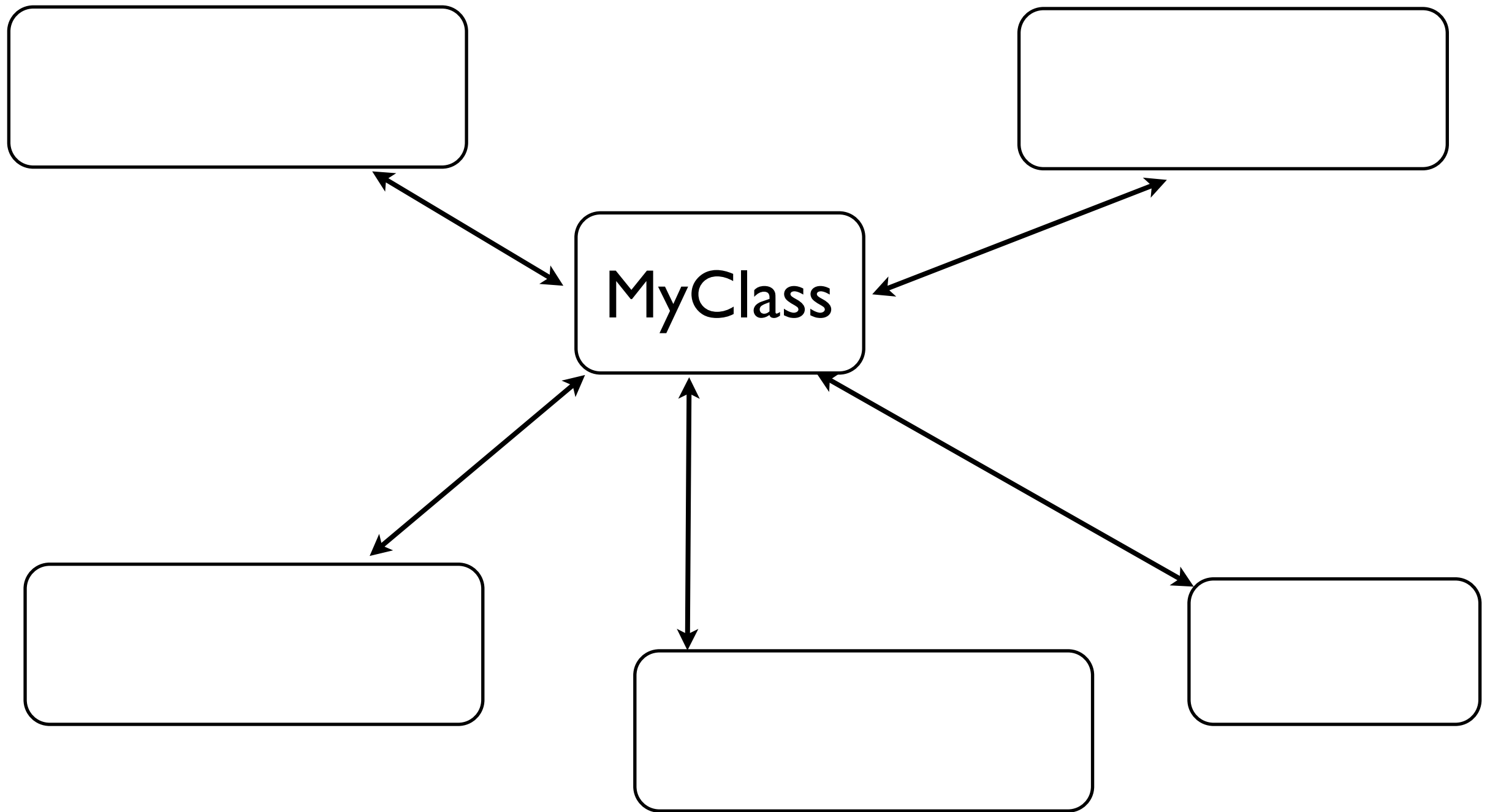




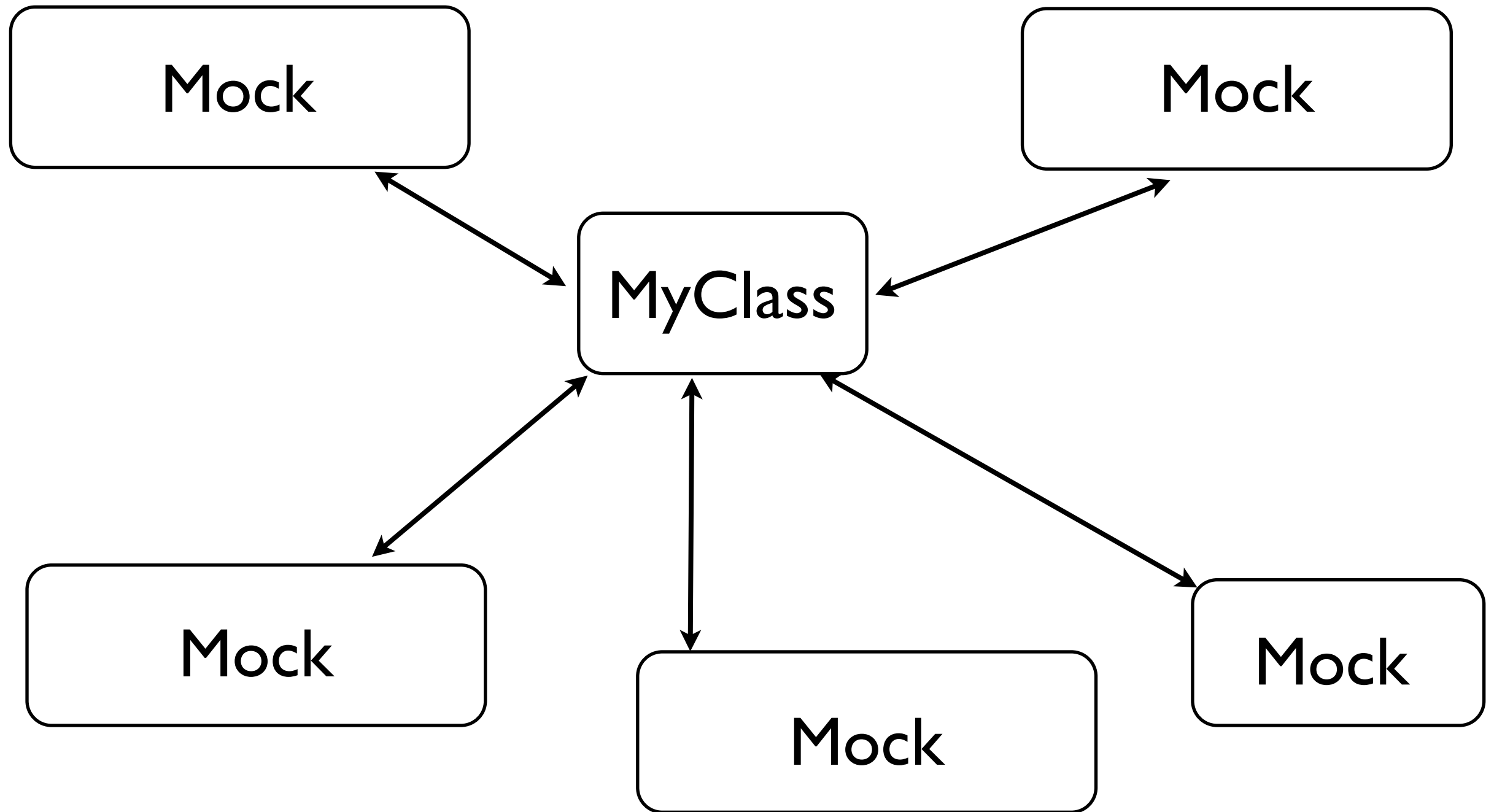
# Behavior tests



# Behavior tests



# Behavior tests



# Mock via subclassing

```
@interface MockMapView : MKMapView
@property(n nonatomic, assign) BOOL addOverlaysWasCalled;
@property(n nonatomic, retain) NSArray* overlaysAdded;
@end

@implementation MockMapView

-(void)addOverlays:(NSArray *)overlays
{
    self.addOverlaysWasCalled = YES;
    self.overlaysAdded = overlays;

    [super addOverlays:overlays];
}

@end
```



# Mock via OCMock

```
NSArray* expectedOverlays = @[ ... ];  
id mockMapView = [OCMockObject mockForClass:[MKMapView class]];  
[[mockMapView expect] addOverlays:expectedOverlays];  
[testClass setMapView:mockMapView];  
  
[testClass doStuffWithMapView];  
  
[mockMapView verify];
```



- Create a mock

- Create a mock
- Tell it what should be called

- Create a mock
- Tell it what should be called
- Set up your class with mock

- Create a mock
- Tell it what should be called
- Set up your class with mock
- Call method under test

- Create a mock
- Tell it what should be called
- Set up your class with mock
- Call method under test
- [mock verify] will throw exception if expectations were not met.

**Demo**



# More Info

- Get OCMock at: <http://ocmock.org>
- <http://alexvollmer.com/posts/2010/06/28/making-fun-of-things-with-ocmock/>

# More Info

Graham Lee:  
Test-driven iOS Development

Addison Wesley  
228 Pages

