Macoun'10

Komplexe Projekte mit Cocoa

Christian Kienle

[co coa:ding]

Motivation

- Komplexe Projekte sind die Regel
- Bessere Codequalität
- Bessere Wartbarkeit

“Dance like nobody is watching.”

“Code like nobody is watching.”

“Code like **everybody** is watching.”

Jeff LaMarche

Dimensionen

- **Kürze**
- Leistungsumfang
- Ausführgeschwindigkeit
- Entwicklungsaufwand
- Robustheit
- Flexiblität

Quelle: Call Me Fishmeal (Wil Shipley)

80-zu-20-Regel

Guter Code

Return Early oder Return Late

```
@implementation ReturnLate

- (id)doSomethingInterestingWith:(NSString *)input
{
    if(input != nil) {
        if([input length] > 0) {
            if(m != nil) {
                return [input stringByAppendingString:m];
            }
        }
    }
    return nil;
}
@end
```

```
1678
1679                    if(itemName)
1680                    {
1681
1682                       // RessourceTyp
1683                       NSInteger resourceType = [(NSNumber*)CFDictionaryGetValue(dictionaryRef, kCFFTPResourceType)
                               integerValue];
1684
1685                       if(resourceType == 4) // Ordner?
1686                       {
1687
1688                          NSString *subDirectoryPath = [NSString stringWithFormat:@"%@%@/",
                                   encodedRemoteDirectoryLocation, [itemName stringByAddingPercentEscapesUsingEncoding:
                                   encoding]];
1689
1690                          [contentList addObject:subDirectoryPath];
1691
1692                          // Freigeben
1693                          if(autoreleasePool)
1694                          {
1695
1696                             [autoreleasePool drain];
1697                             autoreleasePool = [[NSAutoreleasePool alloc] init];
1698
1699                          }
1700
1701
1702                          NSArray *subpathContent = [[TAFTPTransfer FTPTransfer] ftpSubpathsOfRemoteDirectory:
                                   subDirectoryPath
1703                                                                                    userName:remoteUserName
1704                                                                                  userPassword:remoteUserPassword
1705                                                                                    encoding:encoding];
1706
1707                          if(subpathContent)
1708                          {
1709
1710                             [contentList addObjectsFromArray:subpathContent];
1711
1712                          }
1713
1714                       }
1715                       else if(resourceType == 8) // Datei?
1716                       {
1717
1718                          NSString *subDirectoryPath = [NSString stringWithFormat:@"%@%@", encodedRemoteDirectoryLocation
                                   , [itemName stringByAddingPercentEscapesUsingEncoding:encoding]];
1719
1720                          [contentList addObject:subDirectoryPath];
1721
1722                       }
1723
1724                    }
1725
1726                    // ... und weiter
1727                    offset += index;
```

```
@implementation ReturnEarly

- (id)doSomethingInterestingWith:(NSString *)input {
    if(input == nil) {
        return nil;
    }
    if([input length] == 0) {
        return nil;
    }
    if(m == nil) {
        return nil;
    }
    return [input stringByAppendingString:m];
}
@end
```

```
@implementation ReturnEarly
- (BOOL)validateDoSomethingInterestingWith:(NSString *)input {
    return !(input == nil || [input length] == 0 || m == nil);
}

- (id)doSomethingInterestingWith:(NSString *)input {
    if(![self validateDoSomethingInterestingWith:input]) {
        return nil;
    }
    return [input stringByAppendingString:m];
}
@end
```

Singletons

- Erschweren das Testen
- Erhöhen Potential für Seiteneffekte
- Erschweren Parallelisierung
- Verschleiern Abhängigkeiten

```
#import <objc/runtime.h>

#define SYNTHESIZE_SINGLETON_FOR_CLASS_HEADER
(__CLASSNAME__)\
   \
+ (__CLASSNAME__ *)sharedInstance;  \
+ (void)purgeSharedInstance;


#define SYNTHESIZE_SINGLETON_FOR_CLASS(__CLASSNAME__) \
   \
static volatile __CLASSNAME__ *_sharedInstance = nil; \
   \
+ (__CLASSNAME__ *)sharedInstanceNoSynch \
{  \
   return (__CLASSNAME__ *)_sharedInstance; \
```

Categories

- Konservativ nutzen
- Methoden mit Präfix versehen
- Storage mit “Associative References”

Private Methoden

```
@interface Person : NSObject

- (void)doSomethingPrivate;

@end
```

```
@implementation Person

- (void)doSomethingPrivate {
    NSLog(@"Hallo Welt");
}

@end
```

```objc
@interface Person : NSObject
@end
```

```objc
@interface Person ()

- (void)doSomethingPrivate;

@end

@implementation Person

- (void)doSomethingPrivate {
    NSLog(@"Hallo Welt");
}

@end
```

Privater Setter
Öffentlicher Getter

```
@interface Controller : NSObject
@property (copy, readonly) NSString *title;
@end
```

```
@interface Controller ()

@property (copy, readwrite) NSString *title;

@end

@implementation Controller
@synthesize title;

- (void)doSomething {
    self.title = @"Hello World";
}
@end
```

“Abstrahierungswahn”

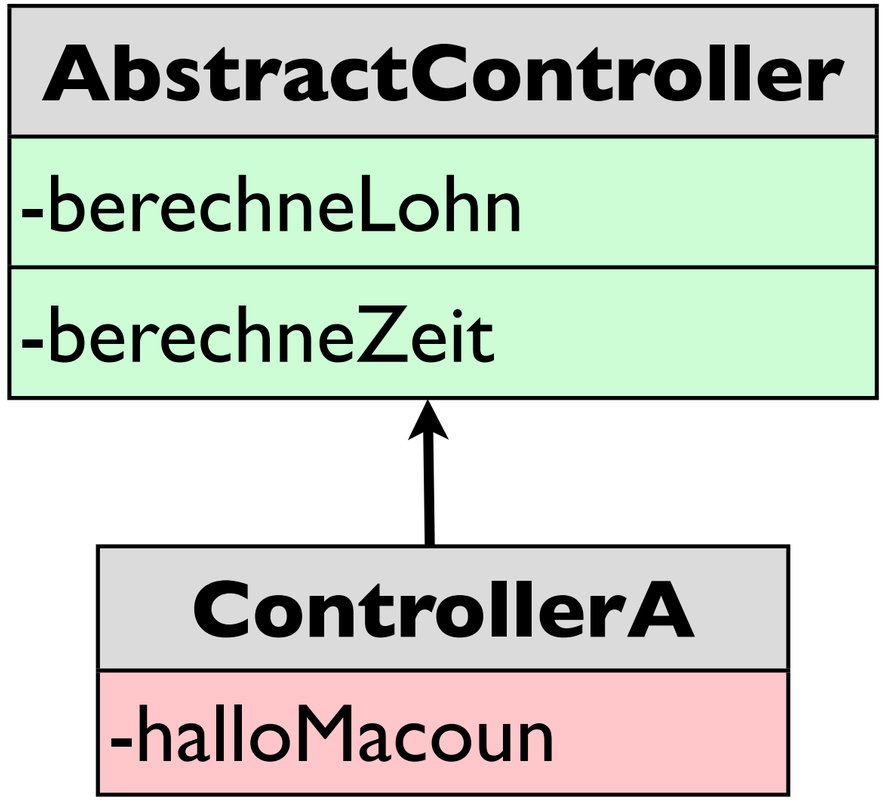
- Im Zweifel: Konkret statt abstrakt
- Vorsichtig abstrahieren

“Abstrahierungswahn”

ControllerA
-berechneLohn
-berechneZeit
-halloMacoun

ControllerB
-berechneLohn
-berechneZeit
-berechneGewinn

“Abstrahierungswahn”

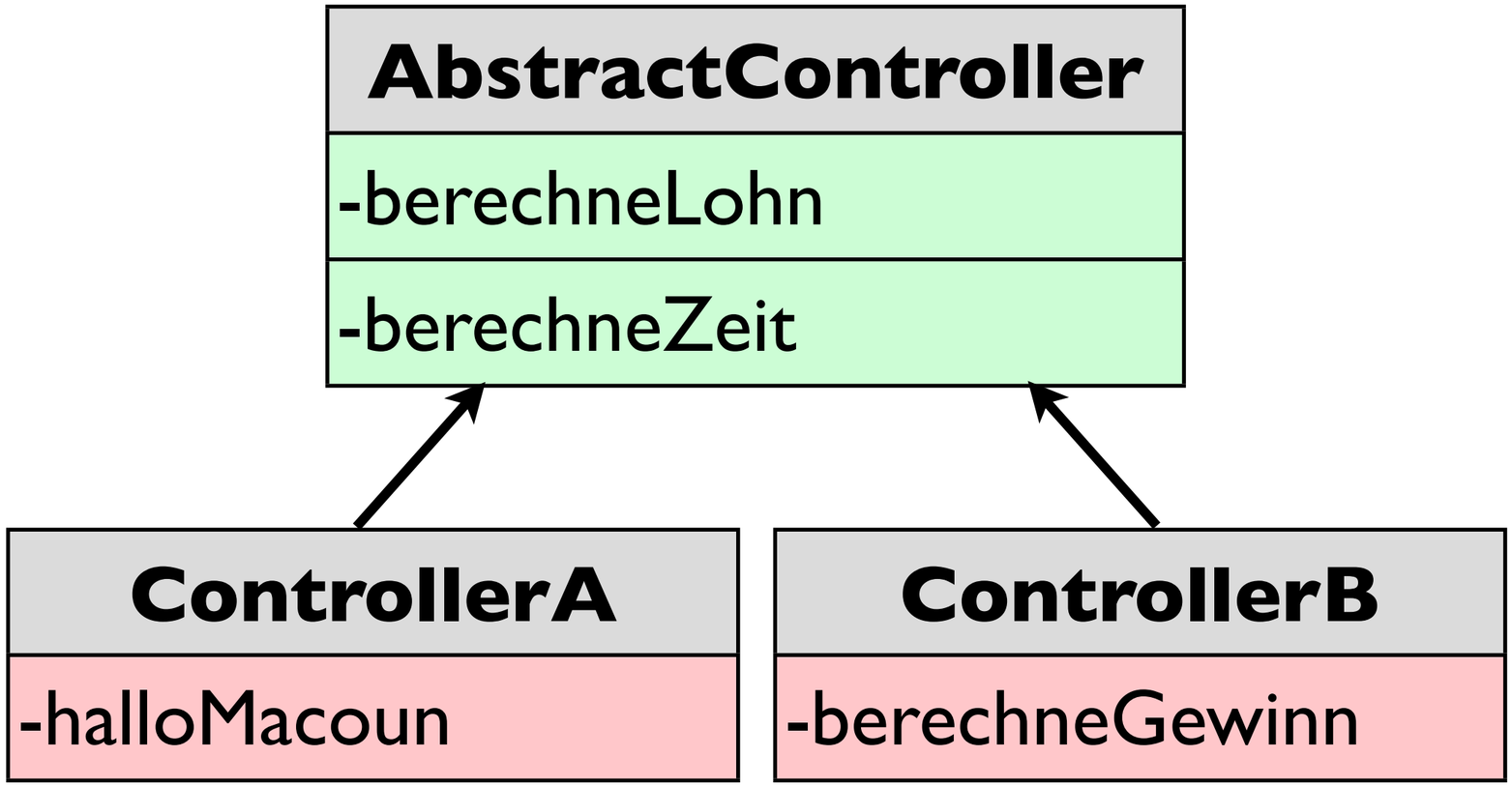


“Abstrahierungswahn”

ControllerA
-berechneLohn
-berechneZeit
-halloMacoun

ControllerB
-berechneLohn
-berechneZeit
-berechneGewinn

“Abstrahierungswahn”



Window- und ViewController

- Controller-Layer aufteilen
- Vereinfacht das Testen
- Viel Flexibilität für wenig Aufwand


iPhoto
LIBRARY
Events
Photos
Faces
Places
RECENT
Nov 17, 2007
Last 12 Months
Last Import
Flagged
Printing
Trash
ALBUMS
Kienle
Englisch
All Faces
Riker
View
Confirm Name
Edit
Rotate
Flag
Hide
Slideshow
Book
Calendar
Card
MobileMe
Facebook
Flickr
Email
Search
1 photo selected

Vielen Dank

Macoun'10