

UnitTests & ObjC

von Martin Häcker <mhaecker@mac.com>

Berlin 5.8.2004

Programm

- Was sind UnitTests (nur wenn gewünscht)
- Wie geht das in Objective C

Was sind UnitTests? Theorie

- Alt

- Neu

- Einfach

- Automatisch

- **Vorher**

Das Problem mit Fehlern

- Einfach zu machen
- Schwer zu finden
- Je später desto Schwer
- Fehler "Schnell" finden ist der Ansatz!

Anatomie eines Tests

- XXXTest (Wenn OO)
- setUp
- testXXX (Davon viele)
- tearDown

Was testen / was nicht?

- Grundsätzlich erst mal alles
- Man merkt sehr schnell, welche Tests helfen Fehler zu finden
- Die schreiben
- Stetes auf und ab

Design for Testability

- Tests sind auch nur ein "Client"
- Schwer zu Testen <-> Schwer zu benutzen
- "Lots of little Pieces!" (Kent Beck)
- Das hat auch noch andere Vorteile

Andere Tests

- Akzeptanz-Tests
- Code-Beweise
- Model-Checking
- Source-Anotation
- Interesse?

Was sind UnitTests?

Praxis

Praktisch

- Mein (bisher) einfachster Testing-Framework:

```
#define assertEquals(expected, actual) ({ \
    if ( ! Matrix_isEqual(expected, actual)) { \
        printf("\n%s:%d:%s Test Failed! ", \
            __FILE__, __LINE__, __FUNCTION__); \
    } \
    else { \
        printf("."); \
    } \
})
```

Beispiel: Python

```
import unittest
from PalindromTester import *

class TestTesting (unittest.TestCase):

    def testSplitInTwo(self):
        self.assertEqual("", "", splitInTwo(""))
        self.assertEqual("a", "a", splitInTwo("aa"))
        self.assertEqual("aa", "aa", splitInTwo("aaaa"))
        self.assertEqual("", "", splitInTwo("a"))

    def testUnify(self):
        self.assertEqual("", unify(""))
        self.assertEqual("", unify(" "))
        self.assertEqual("a", unify("  a"))
        self.assertEqual("a", unify(" a "))
        self.assertEqual("abab", unify(" a b ab"))
```

Beispiel: Java

```
public class ResourceTest extends TestCase {
    private Resource resource;

    protected void setUp() throws Exception {
        Element resourceElement = XMLBuilder.getFirstResource();
        resource = new Resource(resourceElement); }

    public void testType(){
        assertTrue(resource.isNodeResource()); }

    public void testNodeAmount() {
        assertEquals(15, resource.getNodeAmount()); }

    public void testArchitecture() {
        assertEquals("Alpha", resource.getNodeArchitecture()); }
}
```

Beispiel: Objective C

```
#import "MyPlayerTest.h"
#import <MyUnit/MyTestingMacros.h>
#import "MyPlayer.h"

#define is ==

@implementation MyPlayerTest (ForceDiscoveryUnderZeroLink) @end
@implementation MyPlayerTest

-(void)testClothing {
    id player = [[MyPlayer alloc] init];
    should([player hasClothing:MyShoesAndSocks] is NO);
    [player setClothing:MyShoesAndSocks to:YES];
    should([player hasClothing:MyShoesAndSocks] is YES); }

@end
```

OC-Unit

- Ein Framework von Sen:te
- Benutzt für die restlichen Beispiele
- <http://sente.epfl.ch/software/ocunit/>

Stück für Stück

```
#import <Foundation/Foundation.h>  
#import <SenTestingKit/SenTestingKit.h>
```

```
@class Person;
```

```
@interface TestPerson : SenTestCase  
{  
    Person *person;  
}
```

```
@end
```


Stück für Stück (2)

```
#import "TestPerson.h"  
#import "Person.h"
```

```
@implementation TestPerson
```

```
- (void) setUp {  
    person = [[Person alloc] init]; }  
}
```

```
- (void) tearDown {  
    [person release]; }  
}
```

```
- (void) testFullName {  
    [person setFirstName:@"Pablo"];  
    [person setLastName:@"Picasso"];  
}
```

```
    STAssertTrue ([[person fullName] isEqual:@"Pablo Picasso"],  
                  @"Full name should equal Pablo Picasso.");  
}
```

```
}  
@end
```

Beispiel

Links

- <http://c2.com/cgi/wiki?UnitTest>
- <http://www.sente.ch/software/ocunit/>
- <http://www.cocoadev.com/index.pl?UnitTesting>
- <http://junit.org/index.htm>

Danke!