

# Code Coverage mit PHPUnit & XDebug

Mit diesem Beitrag möchte ich euch Code Coverage in Verbindung mit PHPUnit und XDebug vorstellen. Code Coverage (zu deutsch: Testabdeckung) ist ein nützliches Hilfsmittel, um zu überprüfen, wie der zu testende Code mit Testfällen abgedeckt wird. Man kann z.B. leicht erkennen, welche Codeteile nicht von den PHPUnit Tests abgedeckt werden.

**== Installation ==**

Die Installation bezieht sich auf Ubuntu (UNIX) und kann bei anderen Systemen abweichen. Als erstes installieren wir uns PHPUnit:

## Quellcode

1. sudo apt-get install phpunit

Danach sollte PHPUnit systemweit über den folgenden Befehl aufrufbar sein:

## Quellcode

1. phpunit

Nun installieren wir noch XDebug:

## Quellcode

1. sudo pecl install xdebug

Als nächstes müssen wir noch unsere PHP-Konfigurationsdatei anpassen:

## Quellcode

1. sudo gedit /etc/php5/apache2/php.ini

Wir fügen nun folgendes in die Datei ein:

## Quellcode

1. [xdebug]
2. zend\_extension=/usr/lib/php5/20090626/xdebug.so
3. xdebug.remote\_enable=1
4. xdebug.remote\_handler=dbgp
5. xdebug.remote\_mode=req
6. xdebug.remote\_port=9000
7. xdebug.remote\_host=127.0.0.1

Den absoluten Pfad zur Extension kann man über "find" herausfinden:

## Quellcode

1. sudo find / -name xdebug.so

Am Ende müssen wir Apache neustarten:

## Quellcode

1. sudo /etc/init.d/apache2 restart

Um sicherzustellen, dass XDebug erfolgreich eingerichtet ist, reicht uns folgender Befehl:

## Quellcode

1. php -v

Als Ausgabe sollte wir diesen Text hier erhalten und zwar mit dem XDebug Copyright Text:

## Quellcode

1. PHP 5.3.2-1ubuntu4.5 with Suhosin-Patch (cli) (built: Sep 17 2010 13:49:46)
2. Copyright (c) 1997-2009 The PHP Group
3. Zend Engine v2.3.0, Copyright (c) 1998-2010 Zend Technologies
4. with Xdebug v2.1.0, Copyright (c) 2002-2010, by Derick Rethans

== PHPUnit Test erstellen ==

Wir erstellen uns folgende Beispielklasse und Beispiel-Testklasse:

## Quellcode

1. cd /var/www/
2. sudo gedit exampleClass.php

Inhalt der Beispielklasse:

## Quellcode

```
1. <?php
2. class ExampleClass
3. {
4.     private $exampleVar = "";
5.     private static $onExampleMode = true;
6.     public function getExampleVar()
7.     {
8.         return $this->exampleVar;
9.     }
10.    public function setExampleVar($var)
11.    {
12.        if($var === 'abc')
13.        {
14.            $this->exampleVar = $var;
15.        }
16.        else
17.        {
18.            $this->exampleVar = 'xyz';
19.        }
20.    }
21.    public function isOnExampleMode()
22.    {
23.        return self::$onExampleMode;
24.    }
25. }
```

```
29. }
30. ?>
```

Alles anzeigen

Jetzt die dazugehörige Testklasse:

## Quellcode

1. sudo gedit exampleClassTest.php

Inhalt der Testklasse:

## Quellcode

```
1. <?php
2. require_once('exampleClass.php');
3. class ExampleClassTest extends PHPUnit_Framework_TestCase
4. {
5.     private $exampleInstance = '';
6.     protected function setUp()
7.     {
8.         $this->exampleInstance = new ExampleClass();
9.     }
10.    public function testExampleVarIsStringAbc()
11.    {
12.        $this->exampleInstance->setExampleVar('abc');
13.        $this->assertEquals('abc', $this->exampleInstance->getExampleVar());
14.    }
15. }
16. }
17. }
18. }
19. ?>
```

Alles anzeigen

== Coder Coverage Report erstellen ==

Die Erstellung des Code Coverage Reports ist relativ simpel und reduziert sich auf einen Befehl:

## Quellcode

1. # 'report' ist der Name des Verzeichnis in dem der Report gespeichert wird
2. sudo phpunit --coverage-html report exampleClassTest.php

Dann können wir den erzeugten Report anschauen. Dazu öffnen wir einfach die im Ordner "report" erzeugte Datei "exampleClass.php.html" mit irgendeinem Browser, wie z.B. Firefox.

Ergebnis:

[Blockierte Grafik: <http://image-upload.de/image/fCmRh1/e837a6aa79.jpg>]

[Blockierte Grafik: <http://image-upload.de/image/OAOwjm/d4c19979b7.jpg>]

Durch die farbliche Hervorhebung sehen wir nun leicht, welche Codeanteile durch unsere Testfälle abgedeckt sind und wie oft sie durchlaufen wurden.