

# stub和mock的比較

### 源地址

Mocking frameworks: stubs vs mocks

<http://mindinthewater.blogspot.com/2010/02/mocking-frameworks-stubs-vs-mocks.html>

## 1. Mocks

- (1) mock 是藉由提供一連串想要的函式呼叫和對應的傳回值, 來建立所想要測試的流程.
- (2) 所以你可以把mock 想像成是戲劇中的演員, 他必須要依據劇本來演出. 劇本內會描述一些預期所要演出的劇情是什麼, 也就是演員必須說出劇本內所描述的句子

### (3) 範例

[Test]

```
public void Eat_uses_banana_provider()
{
    var mocks = new MockRepository();
    var bananaProviderMock = mocks.DynamicMock();
    bananaProviderMock.GetBanana(); // record method call expectation
    LastCall.Return(new Banana()); // tell mock what to do
    bananaProviderMock.Replay(); // stop recording, go to replay mode

    // actual test
    var monkey = new Monkey(bananaProviderMock);
    monkey.Eat();

    // verify the expectations that were set up earlier
    bananaProviderMock.VerifyAll();
}
```

## 2. Stubs

- (1) 和mock不同, 沒有"test script"的概念.
- (2) 你只需要設定固定的答案給被呼叫的函式
- (3) 不需要設定任何的預期行為

### (4) 範例

[Test]

```
public void Eat_uses_banana_provider()
{
    var bananaProviderStub = MockRepository.GenerateStub();
    bananaProviderStub.Stub(x => x.GetBanana()).Return(new Banana());

    // actual test
    var monkey = new Monkey(bananaProviderStub);
    monkey.Eat();

    // make assertion about how the stub was used
    bananaProviderStub.AssertWasCalled(x => x.GetBanana());
}
```

盡可能的只使用stub而不是mock, 原因如下

3. Mock威力太大, 容易讓你的測試難以了解
4. 通常會用mock來做測試時, 測試的script會比較複雜, 因此若有問題時會花比較多的時間去檢查
5. 除非你很清楚這一連串動作的預期結果, 否則使用mock的測試通常很容易導致測試結果不是你所預期的