

Running Junit & PowerMock with Mockito through PowerMockRunner from maven

 stackoverflow.com/questions/1796833/running-junit-powermock-with-mockito-through-powermockrunner-from-maven

I am not being able to run Powermock through maven. I'm the PowerMock Mockito and PowerMockRunner for driving a junit test.

Here's the test:

```
@RunWith(PowerMockRunner.class)
@PrepareForTest( { UserLocalServiceUtil.class, ExpandoBridge.class })
public class AlertNotificationsTest {
//...
```

I haven't configured anything special for running the test. My pom references the following deps:

- org.mockito | mockito-all | 1.8.0
- junit | junit | 4.6.0
- org.powermock.modules | powermock-module-junit4 | 1.3.1
- org.powermock.api | powermock-api-mockito | 1.3.1

when I run `mvn -Dtest=AlertNotificationsTest test` mvn says there's no test to run. But if I run the same test class from eclipse, everything runs ok.

Am I doing something wrong?

Here's my pom.xml below (the relevant parts)

```

<dependency>
  <groupId>org.testng</groupId>
  <artifactId>testng</artifactId>
  <version>5.9</version>
  <classifier>jdk15</classifier>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.6</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.mockito</groupId>
  <artifactId>mockito-all</artifactId>
  <version>1.8.0</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.powermock.modules</groupId>
  <artifactId>powermock-module-junit4</artifactId>
  <version>1.3.1</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.powermock.api</groupId>
  <artifactId>powermock-api-mockito</artifactId>
  <version>1.3.1</version>
  <scope>test</scope>
</dependency>
</dependencies>

```

Here's the output from maven

```

| mvn -Dtest=AlertNotificationsTest test
...
[INFO] Surefire report directory: C:\Devel\Java\EP_PORTAL\information-provider\target\surefi
-----
T E S T S
-----
Running TestSuite
Tests run: 0, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.313 sec

Results :

Tests run: 0, Failures: 0, Errors: 0, Skipped: 0

[INFO] -----
[ERROR] BUILD FAILURE
[INFO] -----
[INFO] No tests were executed! (Set -DfailIfNoTests=false to ignore this error.)
[INFO] -----

```

Note: I can run other tests, I just can't run this test. If I make the `AlertNotificationsTest` class extend `junit.framework.TestCase` the class gets picked up by maven, but it seems that it does not get driven by `PowerMockRunner`.

Here's the output of that:

Running TestSuite

[ERROR]: No test suite found. Nothing to run

Tests run: 4, Failures: 2, Errors: 0, Skipped: 0, Time elapsed: 1.053 sec <<< FAILURE!

Results :

Failed tests:

```
testSingleEventNotification(pt.estradasportugal.traffic.services.events.AlertNotificationsTest)
testTwoEventNotification(pt.estradasportugal.traffic.services.events.AlertNotificationsTest)
```

Tests run: 4, Failures: 2, Errors: 0, Skipped: 0

Again, these tests run just fine with Eclipse.

Update I found a possible problem & workaround. I have tests with TestNG and JUnit. If I remove TestNG from my pom and migrate all my tests to JUnit, I am able to run my PowerMock test with `mvn test`. So it seems that there's a problem with maven and the junit/testng combo.

I'd like to be able to run both, but If I don't find a way I'll go and answer my own question. Thanks guys&gals

7 Answers

¿No encuentras la respuesta? [Pregunta en Stack Overflow en español](#).

X
12

I just had this error and worked through the solution. My pom.xml file had the following dependency:

```
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-mockito-release-full</artifactId>
  <version>1.5</version>
  <classifier>full</classifier>
  <scope>test</scope>
</dependency>
```

The problem comes from the fact my code uses JUnit and the above dependency has an external dependency on TestNG. This was stopping my test from running. Why I don't know - you would have though a test framework would have been tested a little bit better!!!

Anyway the solution was to break down the 'full' dependencies to just those required:

```
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-api-mockito</artifactId>
  <version>1.5</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-core</artifactId>
  <version>1.5</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-module-junit4</artifactId>
  <version>1.5</version>
  <scope>test</scope>
</dependency>
```

That solved it. BTW I used `mvn dependency:tree` to understand the associated dependencies.

answered Jan 2 '13 at 17:34



Adam Davies

1,714,325,46

2

I can't reproduce your problem. With the following content in my pom.xml:

```

<repositories>
  <repository>
    <id>powermock-repo</id>
    <url>http://powermock.googlecode.com/svn/repo/</url>
  </repository>
</repositories>
<properties>
  <powermock.version>1.3.1</powermock.version>
</properties>
<dependencies>
  <dependency>
    <groupId>org.powermock.modules</groupId>
    <artifactId>powermock-module-junit4</artifactId>
    <version>${powermock.version}</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>org.powermock.api</groupId>
    <artifactId>powermock-api-mockito</artifactId>
    <version>${powermock.version}</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.6</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>org.mockito</groupId>
    <artifactId>mockito-all</artifactId>
    <version>1.8.0</version>
  </dependency>
</dependencies>

```

And the following test class (skipping the imports):

```

@RunWith(PowerMockRunner.class)
@PrepareForTest( { App.class })
public class AppTest {
    @Test
    public void testApp() {
        assertTrue(true);
    }
}

```

Running `mvn test -Dtest=AppTest` just works fine and give me the following output:

...

TESTS

Running com.mycompany.app.AppTest

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.135 sec

Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

```
[INFO] -----  
[INFO] BUILD SUCCESSFUL  
[INFO] -----  
[INFO] Total time: 3 seconds  
[INFO] Finished at: Wed Nov 25 17:34:32 CET 2009  
[INFO] Final Memory: 9M/79M  
[INFO] -----
```

So the question is: do you have a method annotated with `@Test` in `AlertNotificationsTest` ?

answered Nov 25 '09 at 16:40



Pascal Thivent

481k1139461062

2

I came across this issue as well, but its not a PowerMock issue. My Test class was named XStaticTests.java.

When I run "mvn clean test", this test would not run, it only ran when I specified the test using "-Dtest=..."

The surefire documentation mentions that by default only these patterns are searched for : **`/*Test*.java`** - **includes all of its subdirectories and all java filenames that start with "Test"**. `"/Test.java"` - *includes all of its subdirectories and all java filenames that end with "Test"*. `"/**/*TestCase.java"` - *includes all of its subdirectories and all java filenames that end with "TestCase"*.

Therefore changing the classname to one that ends with one of these, will run when "mvn test" is called, else the surefire plugin needs to be configured with the class name specifically.

answered Sep 14 '12 at 12:48



shark8me

41038

1

Powermock setup looks Ok to me, and the jars seem fine (assuming maven transitive dependencies get the other powermock jars - we have about 6-7 after our ivy resolve gets them)

Eclipse might be using it's own "internal" JUnit library, thus the different behaviours ?

Are the test annotated with `org.junit.Test` ?

answered Nov 25 '09 at 13:50



Phantomwhale

9401128

0

If you look into the source of the Surefire plugin, it does some sneaky stuff. If it finds any TestNG packages in the Classloader, it will opt to run a TestNG TestRunner. I haven't yet seen any examples of both JUnit and TestNG tests running well side-by-side.

answered Oct 4 '10 at 23:51



Nelz

514

0

I had the same problem, and it took me a while to figure out. My setup was pulling in an older version of jboss.javassist, which oddly was preventing the PowerMockRunner from working at all.

It's worth noting that I also have a mixed JUnit / TestNG environment. I previously tried the solution of adding multiple surefire providers, and that didn't work either (using surefire 2.14.1). After upgrading to surefire 2.17, both my JUnit and TestNG tests started running without needing to declare any surefire providers.

Here's my plugin section...

```
<plugin>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>2.17</version>
  <configuration>
    <groups>spring, unit, integration</groups>
    <systemPropertyVariables>
      <java.awt.headless>true</java.awt.headless>

<org.apache.activemq.default.directory.prefix>target/test/</org.apache.activemq.default.directory.prefix>

<log4j.configuration>file:${project.basedir}/src/test/resources/log4j.properties</log4j.configuration>

    </systemPropertyVariables>
    <argLine>${surefire.args}</argLine>
  </configuration>
</plugin>
```

... and the relevant testing deps ...

```

<dependency>
  <groupId>org.mockito</groupId>
  <artifactId>mockito-all</artifactId>
  <version>1.9.5</version>
  <scope>test</scope>
</dependency>
<!--
PowerMock versions are compatible with specific Mockito versions.
https://code.google.com/p/powermock/wiki/MockitoUsage13
-->
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-module-junit4</artifactId>
  <version>1.5.4</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-api-mockito</artifactId>
  <version>1.5.4</version>
  <scope>test</scope>
</dependency>
<!-- without this PowerMock tests don't run in maven -->
<dependency>
  <groupId>jboss</groupId>
  <artifactId>javassist</artifactId>
  <version>3.8.0.GA</version>
  <scope>test</scope>
</dependency>

```

answered Mar 20 '14 at 16:04



Jeff Fairley

4,09432845

-1



There was a problem when mixing both TestNG & JUnit tests. Migrating all tests to Junit solved my problem. Thanks guys.

answered Dec 2 '09 at 12:25



Miguel Ping

13.1k1976126