

<http://membres-liglab.imag.fr/donsez/cours>

JUnit

Tests unitaires en Java

Didier DONSEZ

Université Joseph Fourier (Grenoble 1)

PolyTech'Grenoble – LIG/ADELE

`Didier.Donsez@imag.fr`

`Didier.Donsez@ieee.org`

Sommaire

- Motivations
- JUnit
- API
- Misc
- Bibliographie

Motivations

- Tests unitaires
 - http://fr.wikipedia.org/wiki/Test_%28informatique%29
 - http://fr.wikipedia.org/wiki/Test_unitaire

- eXtreme Programming (XP)
 - Développer les tests unitaires en même temps/avant le programme à tester
 - A partir des spécifications

JUnit

- Basé sur le patron de conception [Gamma95]
 - Séparation du code de test du code métier
 - Le code de test est défini, développé et livré séparément

- Site Web
 - <http://www.junit.org>

API junit.framework.*

- Interface
 - Test
- Classes
 - TestCase *étends Assert et implémente Test*
 - TestSuite *(suite de TestCase)*
 - Assert *(méthodes assertXXX() et failZZZ())*
- Méthodes
 - MyTest.testYYY()
 - YYY = spécifique à l'application testée
 - Assert.assertXXX(...)
 - XXX = True, False, Equal, ...
 - TestCase.setUp(), TestCase.setUpBeforeClass()
 - initialise des objets tels que des connexions JDBC, ... avec les tests
 - TestCase.tearDown(), TestCase.tearDownAfterClass()
 - relâche les objets pris au setUp() après les tests
 - TestSuite.addTest(Test)
 - Ajoute un test à une TestSuite

Exemple : Money

```

import junit.framework.*;
public class MoneyTest extends TestCase {
    private Money f12CHF;
    private Money f14CHF;
    public MoneyTest(String testName){
        super(testName);
    }
    protected void setUp() {
        f12CHF= new Money(12,"CHF");
        f14CHF= new Money(14,"CHF");
    }
    public void testSimpleAdd () {
        Money expected = new Money(26,"CHF");
        Money result= f12CHF.add(f14CHF);
        assertTrue(expected.equals(result));
    }
    ...
}

```

Étend TestCase

Nom du test

Invoquée avant chaque test

Une des méthodes de test

Assertion à vérifier

API

■ TestSuite

- Organiser/Composer/Regrouper plusieurs Tests

```
TestSuite suite = new TestSuite("MoneyTestSuite");  
suite.addTest(MoneyTest.class);  
suite.addTest(AnotherMoneyTest.class);
```

■ ActiveTestSuite

- Chaque test est exécuté par une thread indépendante

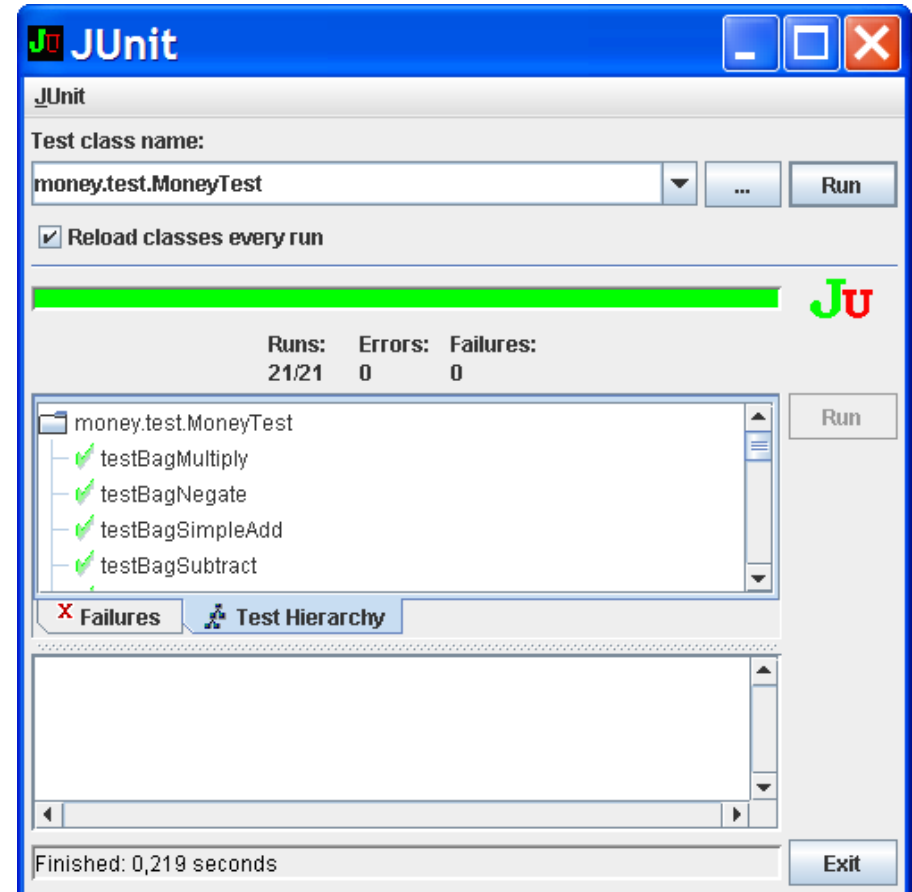
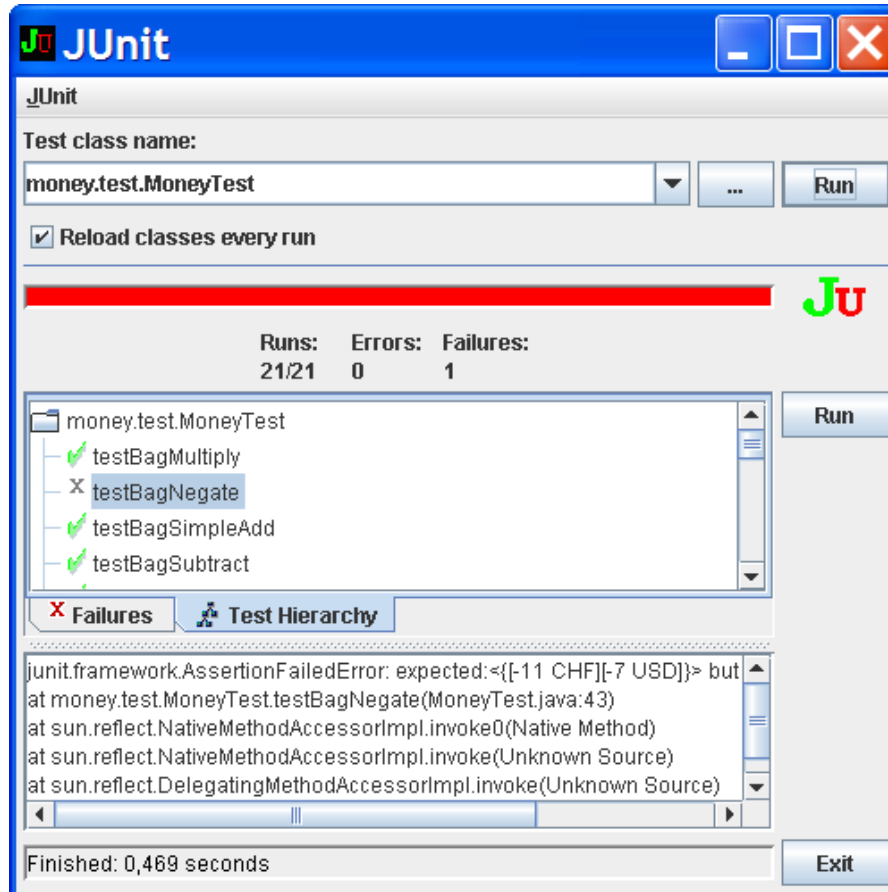
■ RepeatedTest

- Performance ou erreur intermitente

```
suite.addTest(new RepeatedTest(new MoneyTest("testSimpleAdd"),100));  
suite.addTest(new RepeatedTest(new MoneyTest("testComplexAdd"),100));
```

TestRunner

- Utilitaires (textuel et graphique) pour lancer une suite de tests
 - `java junit.textui.TestRunner money.MoneyTest`
 - `java junit.swingui.TestRunner money.MoneyTest`



JUnit 4

- Motivation : Exploiter Java 5.0
 - Generics, @OP (Meta-programmation)
- API org.junit.*
 - Étendre TestCase n'est plus nécessaire
 - @Test → for a test methods
 - @Test(expected=IndexOutOfBoundsException.class)
 - @Test(timeout= 1000)
 - @Suite
 - @Suite(tests={MoneyTest.class, SimpleTest.class})
 - @Suite(pattern="*Test")
 - @Suite(package="org.junit.test.*")
 - @Ignore → to ignore a test
 - @Ignore(reason= "...")
 - @Before, @After → setup/teardown
 - @BeforeClass, @AfterClass → one time setup/teardown
 - class Assert → méthodes static assertXXX()
 - @Theory → globally true assertion

Exemple : Money

```
import org.junit.*;
public class MoneyTest4 {
    private Money f12CHF;
    private Money f14CHF;

    @Before
    protected void setUp() {
        f12CHF= new Money(12,"CHF");
        f14CHF= new Money(14,"CHF");
    }

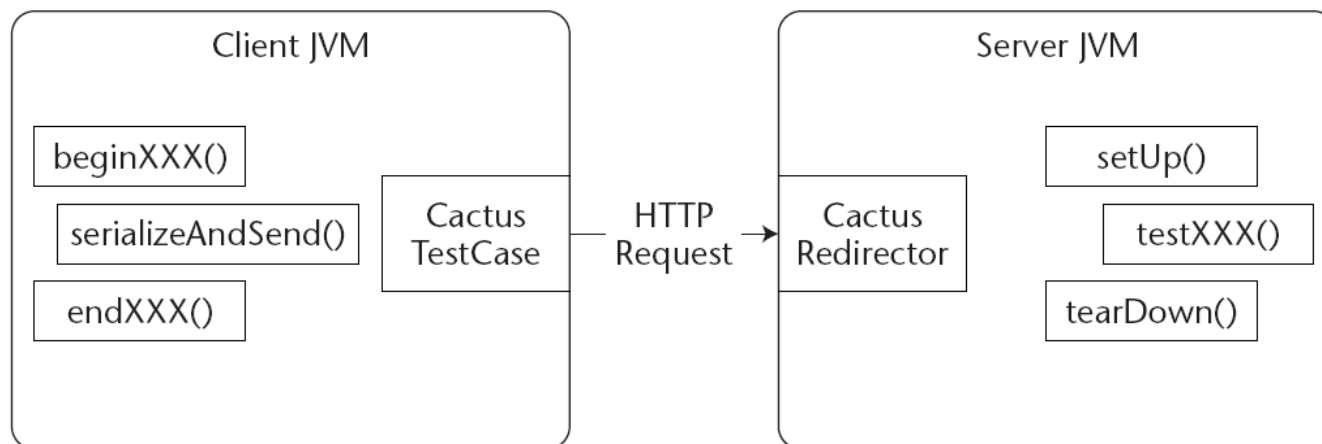
    @Test
    public void testSimpleAdd () {
        Money expected = new Money(26,"CHF");
        Money result= f12CHF.add(f14CHF);

        Assert.assertTrue(expected.equals(result));
    }

    ...
}
```

Tests unitaires JavaEE

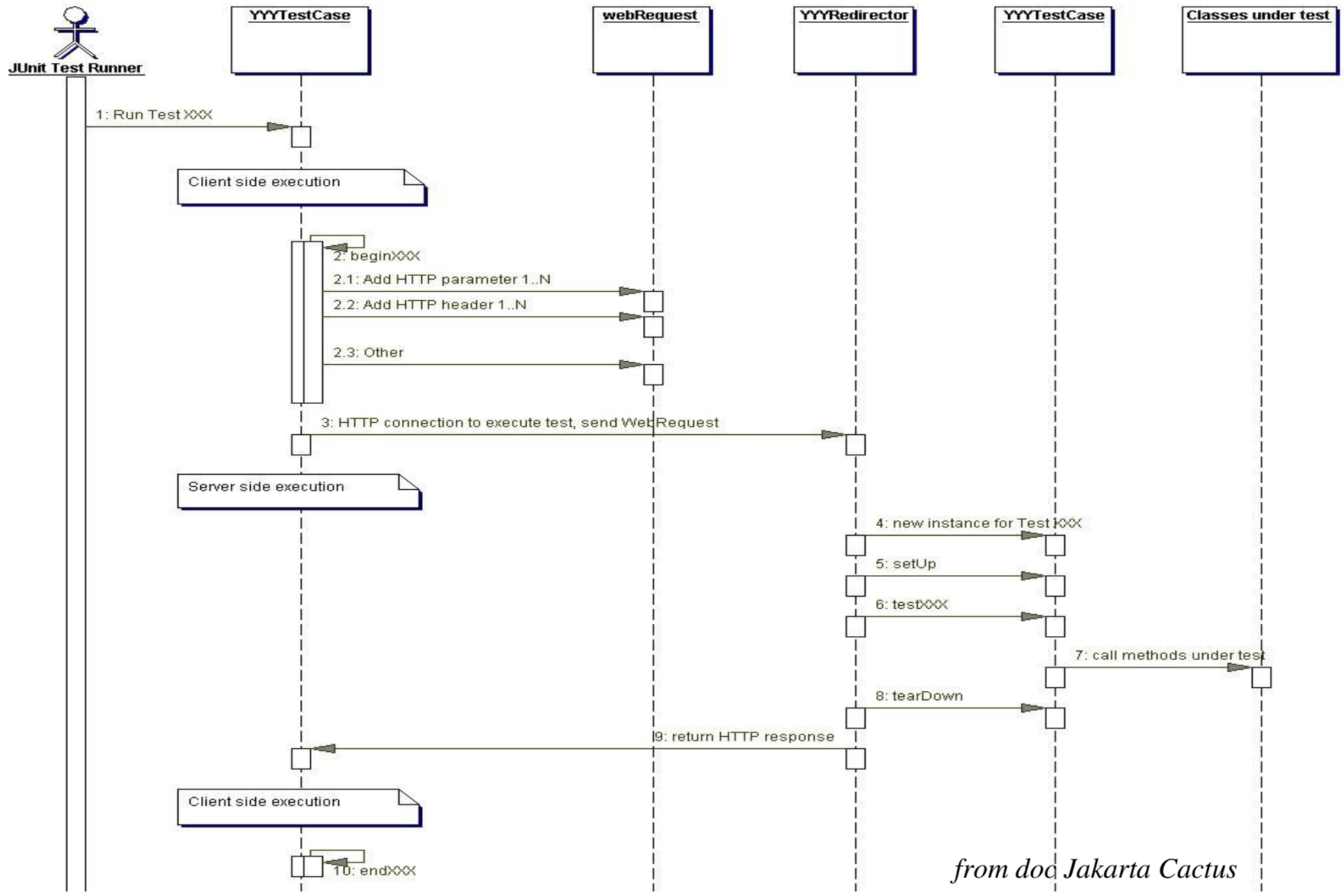
- Test unitaires de Servlets, JSP, Taglibs, Filters, EJB,
- Jakarta Cactus <http://jakarta.apache.org/cactus/>
 - Architecture



- **A lire**

- <http://www.theserverside.com/tt/articles/content/JakartaPitfalls/JakartaPitfallsChapter1.pdf>
- http://jakarta.apache.org/cactus/writing/howto_testcase.html

Jakarta Cactus



Tests unitaires avec Spring

- Motivation
 - difficulté de tester la logique applicative des EJB en dehors d'un container EJB (2 ou 3).
 - POJO (Plain Old Java Object)
- TODO

Tests unitaires de bases de données

- Tests unitaires sur des bases relationnelles
- DBUnit <http://dbunit.sourceforge.net/>
 - Basé sur JUnit et JDBC
 - *DbUnit has the ability to export and import your database data to and from XML datasets. Since version 2.0, DbUnit can work with very large datasets when used in streaming mode. DbUnit can also help you to verify that your database data match the expected set of values.*

JUnit avec Ant

- Tache Ant <junit>
 - Lance les tests depuis un build.xml
- Exemple

```
<junit printsummary="yes" haltonfailure="yes">
  <classpath>
    <pathelement location="${build.tests}"/>
    <pathelement path="${java.class.path}"/>
  </classpath>

  <formatter type="plain"/>

  <test name="my.test.TestCase" haltonfailure="no" outfile="result">
    <formatter type="xml"/>
  </test>

  <batchtest fork="yes" todir="${reports.tests}">
    <fileset dir="${src.tests}">
      <include name="**/*Test*.java"/>
      <exclude name="**/AllTests.java"/>
    </fileset>
  </batchtest>
</junit>
```

JUnit avec Ant

- Tache Ant `<junitreport>`
 - Fusionne plusieurs rapports JUnit
- Exemple

```
<junitreport todir="./reports">  
  <fileset dir="./reports">  
    <include name="TEST-*.xml"/>  
  </fileset>  
  <report format="frames" todir="./report/html"/>  
</junitreport>
```


JUnit avec Maven

■ Plugins Surefire de Maven

■ maven-surefire-plugin

- Lance les testcases JUnit lors de la phase de test du build
 - <http://maven.apache.org/plugins/maven-surefire-plugin/>

■ maven-surefire-report-plugin

- Format les rapports de test lors de site:site

■ mvn test -Dtest=MoneyTest

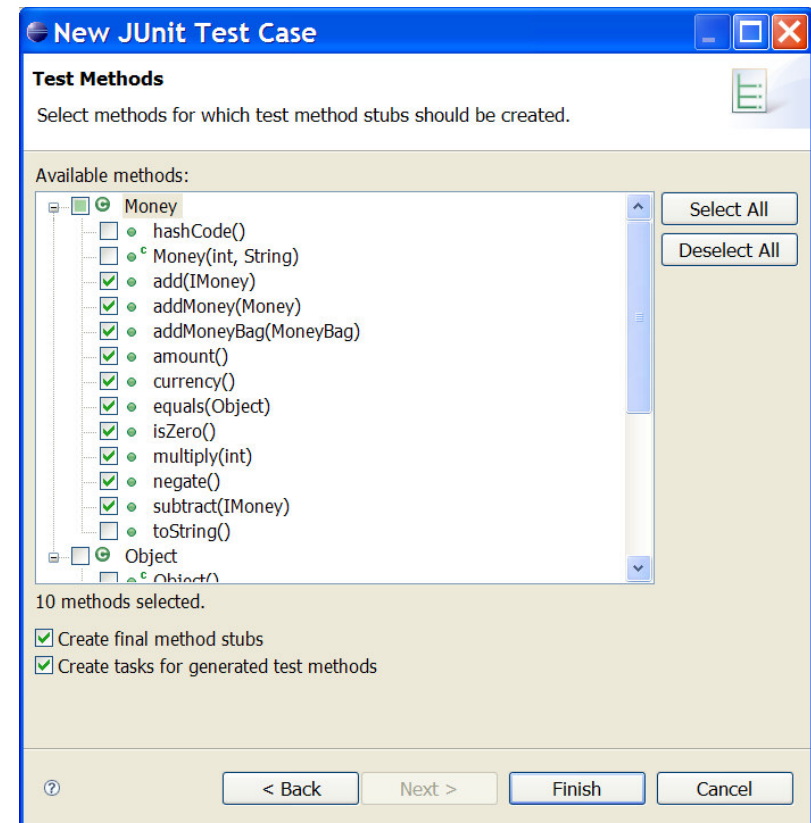
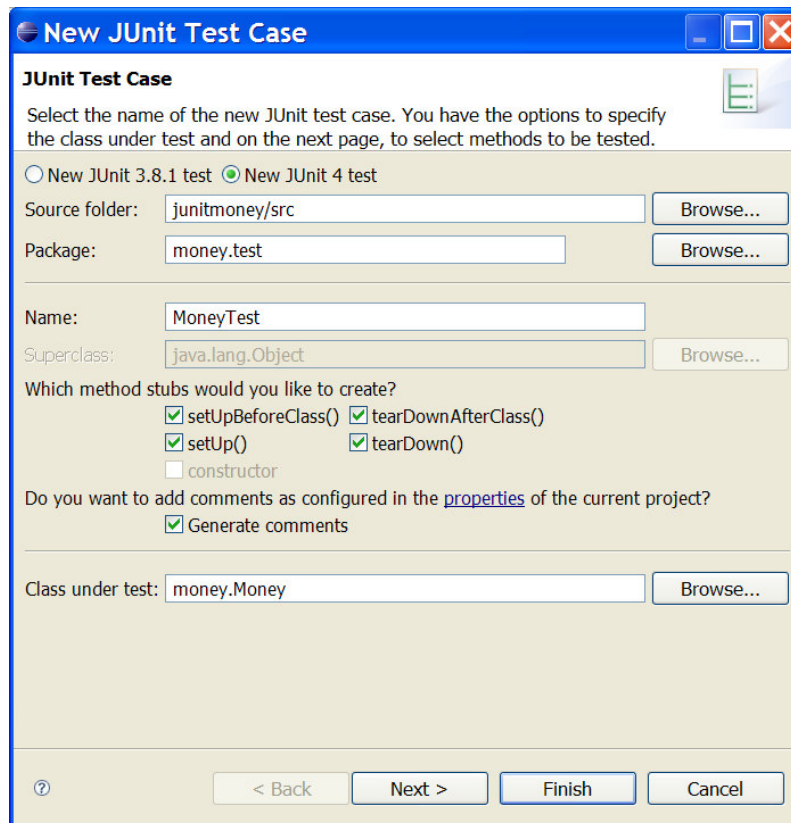
■ Autres

■ Cactus Maven plugin

■ DBUnit Maven plugin (goals: operation,export,compare)

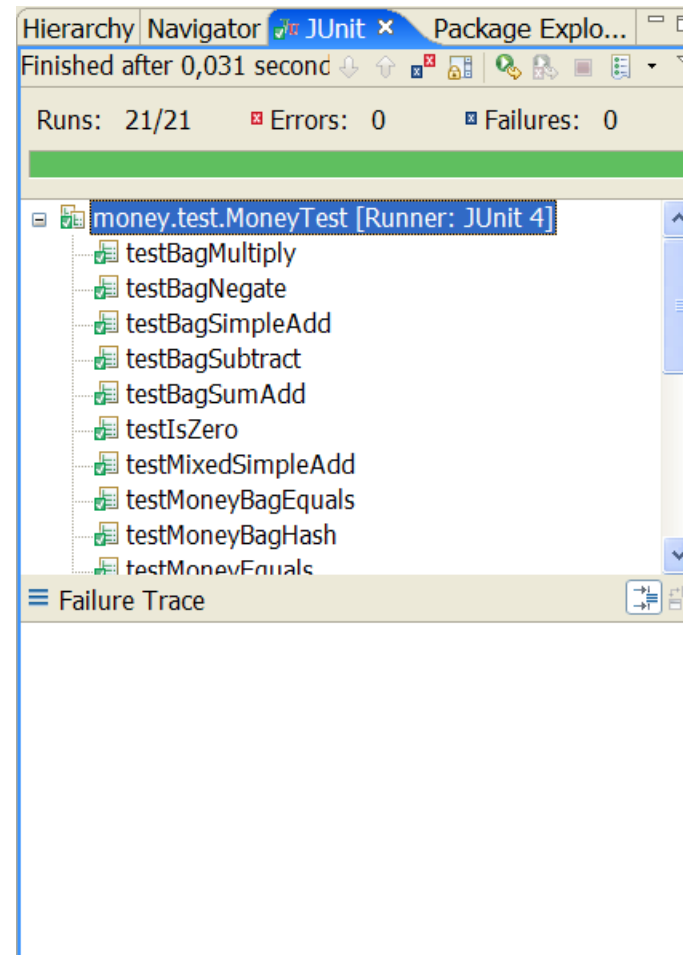
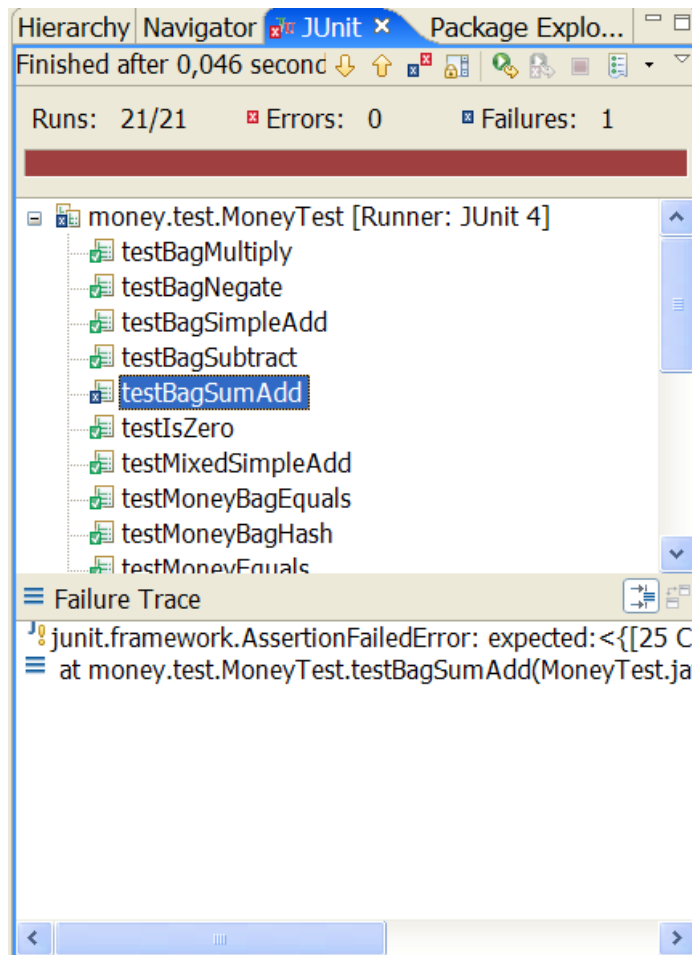
JUnit depuis Eclipse (i)

- Plugin intégré à Eclipse
 - Création de TestCase dans tout projet Java



JUnit depuis Eclipse (ii)

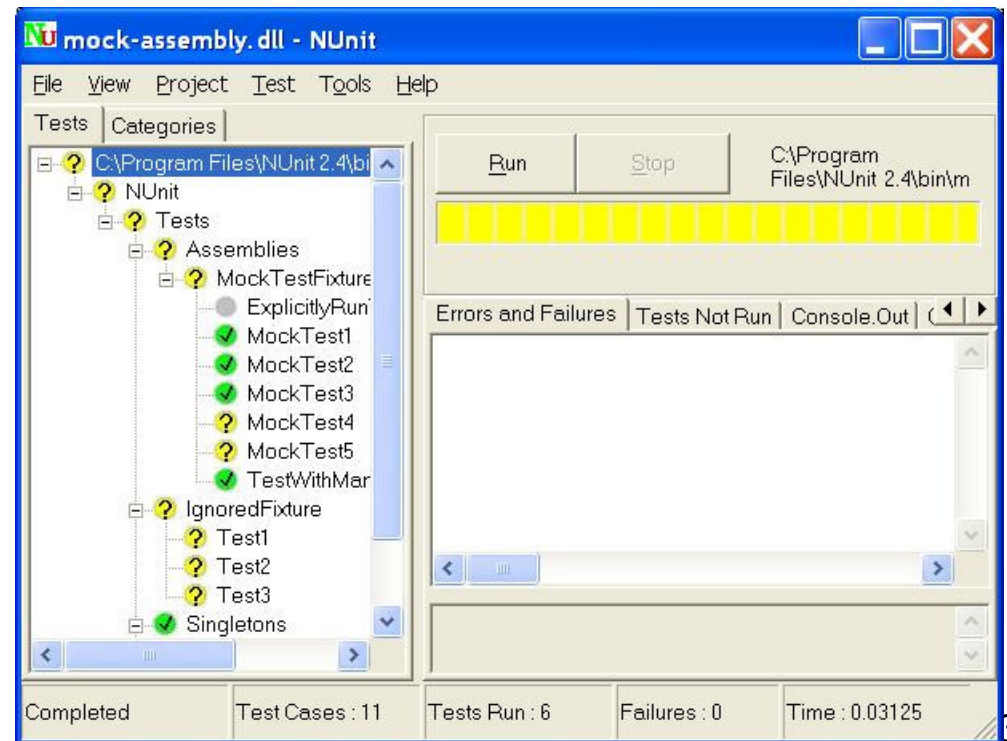
- et Lancement des tests



NUnit

- Comme « JUnit » mais pour .NET !
 - Langages : C#, J#, VB.NET, MC++, ...
 - Plateformes: .NET et Mono (dernières versions)
 - Namespace NUnit.Framework

- Site Web
 - <http://www.nunit.org>



Bibliographie

- Ted Husted, Vincent Massol, “JUnit in Action”, Pub. Manning, 2003, 384 pages, ISBN: 1930110995
- J. B. Rainsberger, “JUnit Recipes: Practical Methods for Programmer Testing”, Pub. Manning, 2004, 752 pages, ISBN: 1932394230
- Eric Burke, Adrian Coyner, « Java Extreme Programming Cookbook », Pub O'Reilly