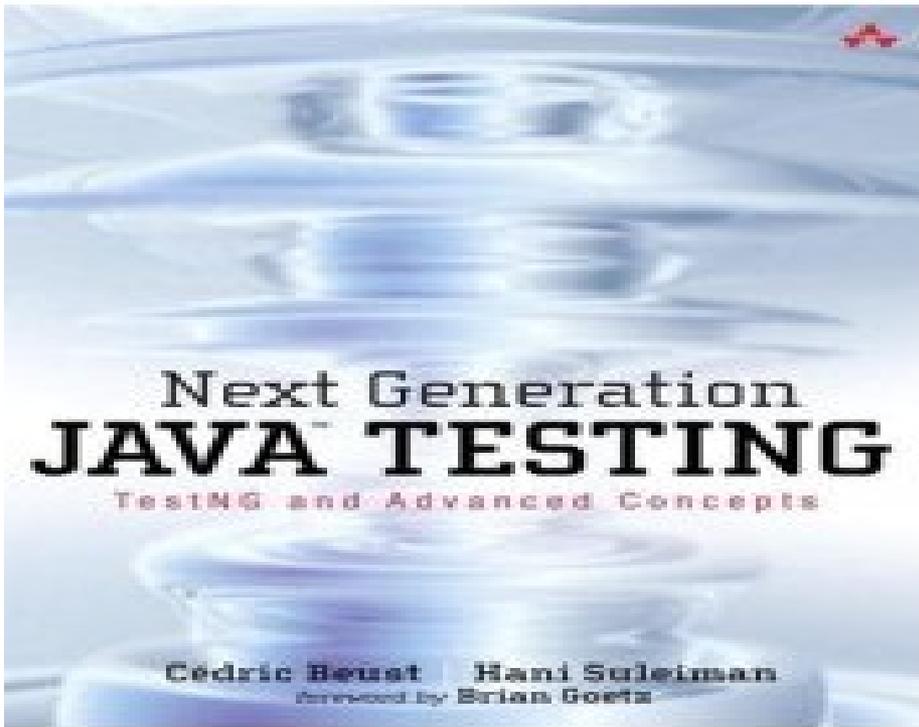


テストフレームワーク「TestNG」について



2011/02/12 作成

佐野 尚之



アジェンダ

- TestNGの特徴 3
- Eclipseのプラグイン使用例 4
- Junitとの違い 8
- 書籍 9
- 参考情報 10



TestNGの特徴

- TestNGはTesting, the Next Generationの略。
- JUnitとJUnitに触発されたJavaのためのテストフレームワーク。
- JUnitは単体テストを重視しているのに対して、TestNGは高レベルなテストや統合テストもできるようになっている。
- データ駆動型テストをサポート(@DataProvider)。
- テストメソッドにアノテーションで指定した値を代入できる引数をサポート。
- 豊富なツールとプラグインによるサポート (Eclipse, IDEA, Maven, Seleniumなど)
- 依存関係のあるテストをサポート。
- 複数のスレッドを使ってテストを並列実行することが可能。
→詳細は@ITのhttp://www.atmarkit.co.jp/fjava/rensai4/testng03/testng03_1.htmlを参照



Eclipseのプラグイン使用例 (1/4)

1、Eclipse3.6.1へのプラグインインストール

- (1). TestNGをダウンロードします。(<http://testng.org/doc/download.html>)
- (2). ダウンロードしたファイルを解凍し、[testng-5.14.10.jar]を任意の場所にコピーします。
- (3). Eclipseを起動します。
- (4). メニューの [ヘルプ] → [新規ソフトウェアのインストール] を選択します。
- (5). 作業対象に[<http://beust.com/eclipse>]を入力します。
- (6). [追加] ボタンをクリックします。
- (7). 名前に[TestNG]と入力して[OK]ボタンをクリックします。
- (8). [TestNG]を選択して[次へ]ボタンをクリックします。
- (9). [インストール詳細]で[TestNG]が表示されていることを確認し、[次へ]ボタンをクリックします。
- (10). [使用条件の条項に同意します]を選択し、[完了]ボタンをクリックします。
- (11). 再起動確認ダイアログが表示されたら、[今すぐ再起動]ボタンをクリックします。



Eclipseのプラグイン使用例 (2/4)

2、Javaのクラス作成

- (1). [ファイル]-[新規]-[Javaプロジェクト]を選択します。
- (2). プロジェクト名に[Counter]、パッケージに[counter]を入力し、[次へ]ボタンをクリックします。
- (3). ライブラリを選択-[外部Jar]ボタンクリック後に[[testng-5.14.10.jar]を追加し、[完了]ボタンをクリックします。
- (4). [Counter]プロジェクトを開き、[src]を選択します。
- (5). 右クリックメニューの[新規]-[クラス]を選択します。
- (6). [名前]に[Counter]と入力して[完了]ボタンをクリックします。
- (7). 以下のコードを入力して保存します。

```
package counter;

public class Counter {

    public int add(int num1, int num2) {
        return num1 + num2;
    }

    public int subtract(int num1, int num2) {
        return num2 - num1;
    }
}
```



Eclipseのプラグイン使用例 (3/4)

3、テストクラス作成

- (1). [Counter.java]を選択し、右クリックメニューの[TestNG]-[Create Test NG class]を選択します。
- (2). [add(int num1, int num2)]と[substract(int num1, int num2)]を選択して、[次へ]ボタンをクリックします。
- (3). [New TestNG class]画面が表示されます。以下の情報を入力して[完了]ボタンをクリックします。

- ・ソース・フォルダー : ¥Counter¥src
- ・パッケージ名 : counter
- ・クラス名 : CounterTest

- (4). テストクラスを作成して保存します。

```
package counter;
import static org.testng.Assert.assertEquals;
import org.testng.annotations.Test;
public class CounterTest {
    @Test
    public void add() {
        Counter counter = new Counter();
        assertEquals(3, counter.add(1, 2));
    }
    @Test
    public void substract() {
        Counter counter = new Counter();
        assertEquals(1, counter.substract(2, 1));
    }
}
```



Eclipseのプラグイン使用例 (4/4)

4、テスト実行

(JUnitと同様、成功するとグリーンバーを、失敗するとレッドバーを表示)

- (1). [CounterTest.java]を選択し、右クリックメニューの[実行]-[TestNG Test(1)]を選択します。
- (2). 以下のようにグリーンバーが表示された場合はテストが成功しています。

Tests: 1/1 Methods: 2/2 (1098 ms)

検索: ✓ パス済み: 2 ✗ 失敗: 0 ⊞ Skip

All Tests Failed Tests 要約

- Default suite (2/0/0/0) (0.12 s)
 - Default test (0.12 s)
 - counter.CounterTest
 - add (0.059 s)
 - subtract (0.061 s)

Failure Exception

```
<終了> CounterTest [TestNG] C:\Program Files\Java\jre6\bin\javaw.exe (2011/02/12 16:37:36)
PASSED: add
PASSED: subtract

=====
Default test
Tests run: 2, Failures: 0, Skips: 0
=====
```



JUnitとの違い

基本的な使い方では違いはあまり無いようですが、機能面では結構違いがあるようです。

機能	JUnit	TestNG
ユーザ定義によるライフ・サイクル	✓	✓
テストの組織化(グループ化など)		✓
分散環境でのテスト実行		✓
並行なテスト実行		✓
データ駆動テスト		✓
依存関係のあるテストの実行		✓
IDEとの統合	✓	✓
Antとの統合	✓	✓
Mavenとの統合	✓	✓
特定ドメインに対する拡張(データベース、HTTPなど)	✓	
活発なコミュニティ	✓	✓

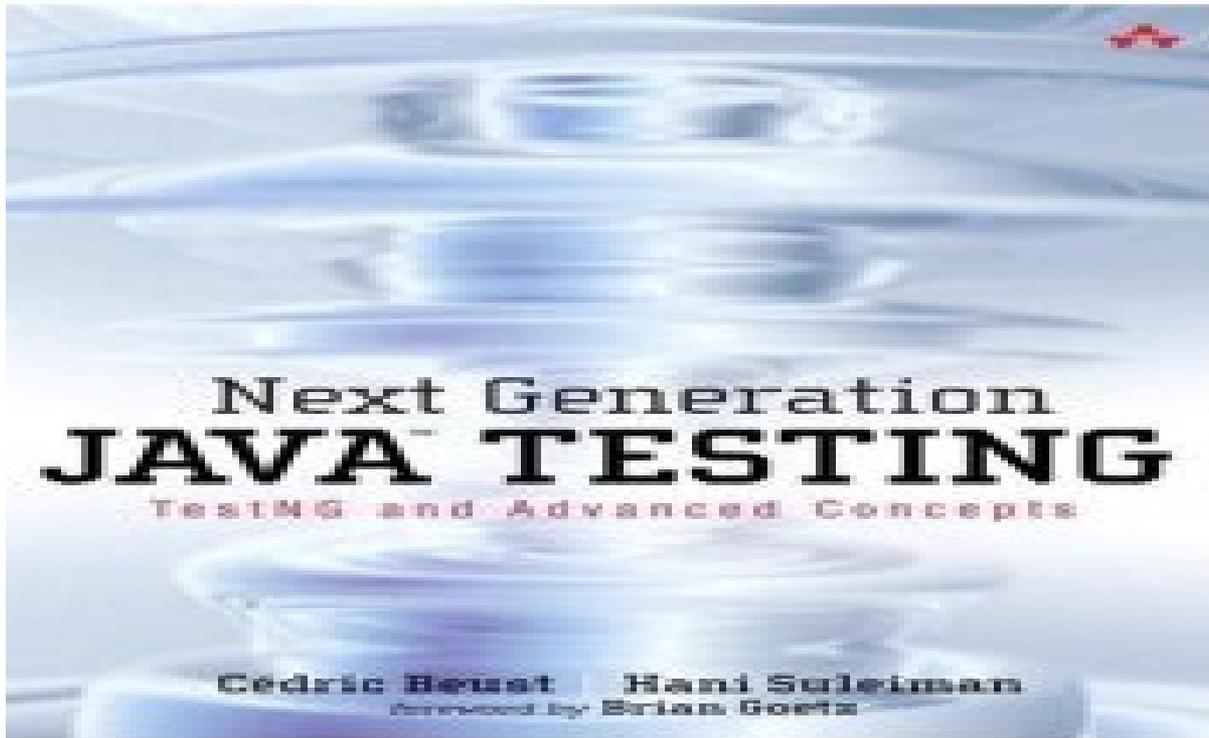
出典([リンク](#))

http://www.infoq.com/jp/news/2008/08/junit_not_deadの図を引用



書籍

Next Generation Java Testing: TestNG and Advanced Concept(英語。2007/10/25。5,185円)





参考情報

TestNG

<http://testng.org/doc/index.html>

TestNG-Eclipse

http://sourceforge.jp/projects/freshmeat_testng-eclipse/

世界一やさしいTestNGの使い方

<http://d.hatena.ne.jp/jyukutyu/20080827/1219802320>

次世代テストフレームワークでテストを変える

http://www.atmarkit.co.jp/fjava/index/index_testng.html

Javaで自動テスト - EclipseでTestNGを使ってみる。

<http://blog.mufu.jp/2008/04/javaeclipsetestng.html>

コード品質を追求する: JUnit 4 対 TestNG

<http://www.ibm.com/developerworks/jp/java/library/j-cq08296/>

TestNGでJavaユニット・テストを楽々行う

<http://www.ibm.com/developerworks/jp/java/library/j-testng/>

TestNG - Wikipedia

<http://ja.wikipedia.org/wiki/TestNG>