

Vortrag Gruppe 2

Gerald Wintersberger

Rene Löschl

Philipp Grasböck

Stefan Wallner

Inhalt

- ◆ Ausführen von JUNIT Tests
- ◆ JUNIT Testberichte
- ◆ Troubleshooting for JUNIT
- ◆ Documenting Testing (IEEE 829)

Ausführen von Junit-Tests (1/3)

- ◆ Textbasierender Testrunner
- ◆ AWT-basierender Testrunner
- ◆ SWING-basierender Testrunner
- ◆ Verwendung von Ant
- ◆ Junit-addons Testrunner

Ausführen von Junit-Tests (2/3)

- ◆ **<junitreport> Task**
XML-Ausgabe wird mittels XSL-stylesheet formatiert. Ausgabe ähnlich dem Javadoc.

[Home](#)
Packages
[tum.ape.app.test](#)
[tum.ape.app.test.processPattern](#)
[tum.ape.db.impl](#)

Classes
[DBParserTest](#)
[DProjectTest](#)
[ExportTest](#)
[ImportTest](#)
[TestAPE_Core](#)
[TestAPE_Project](#)
[TestApeKernProcessPattern](#)
[TestApeKernProcessPatternDescription](#)
[TestApeKernProcessPlan](#)

Unit Test Results

Designed for use with [JUnit](#) and [Ant](#).

Summary

Tests	Failures	Errors	Success rate	Time
41	1	0	97.56%	29.465

Note: *failures* are anticipated and checked for with assertions while *errors* are unanticipated.

Packages

Name	Tests	Errors	Failures	Time(s)
tum.ape.app.test	5	0	1	3.374
tum.ape.app.test.processPattern	22	0	0	11.521
tum.ape.db.impl	14	0	0	14.570

Ausführen von Junit-Tests (3/3)

◆ Ausgabe des Namens des aktuellen Tests

```
public class MyNameisTest extends Testcase {  
    protected void runTest() throws Throwable {  
        System.out.println(„Starting test“ + toString());  
        super.runTest();  
        System.out.println(„Ending test “ + toString());  
    }  
}
```

◆ Ausführen jedes Tests in einer eigenen JVM

```
<target name=“eachSuiteInSeparateJvm“>  
    <junit>  
        <formatter type=“plain“ />  
        <classpath>  
            <pathelement location=“test/classes“ />  
            <pathelement location=“d:/junit3.8.1/junit.jar“ />  
        </classpath>  
        <batchtest fork=“yes“ todir=“logs“>  
            <fileset dir = „test/source“>  
                <include name=“junit/cookbook/running/**/*.*.java“ />  
                <exclude name=“**/GlobalData.java“ />  
            </fileset>  
        </batchtest></junit></target>
```

JUnit Testberichte (1/3)

- ◆ Oft sind Testberichte von Testläufen erwünscht
- ◆ TestRunner unterstützt lediglich Testausgaben
 - in Konsole oder
 - in Java Fenster (grün/roter Balken)
- ◆ Zwei grundlegende Rezepte:
 - JUnit erweitern (nächste Folie)
 - Ant einsetzen (übernächste Folie)

JUnit Testberichte (2/3)

◆ junit.framework.TestCase erweitern + Log Tool

◆ Log4Unit: junit.log4j.LoggedTestCase erweitern

- Benützt, falls verfügbar, Log4J
- Erweiterter junit.logswingui.TestRunner

◆ Angepasstes Ausgabeformat

- junit.framework.TestListener implementieren
- junit.framework.TestRunner erweitern

JUnit Testberichte (3/3)

- ◆ Schlichte Textausgabe (<junit> task)
 - <formatter>.type = (plain | brief | xml)
- ◆ XML Ausgabe mit XSLT transformieren
 - Schablone anpassen oder ersetzen
- ◆ HTML Ausgabe (<junitreport> task)
 - Transformation: XML → HTML (javadoc-like)
- ◆ Ant's JUnit Ausgabeformat erweitern
 - JunitResultFormatter implementieren

Troubleshooting JUNIT (1/2)

Probleme die Auftreten könnten

- ◆ Was tun bei der
*Fehlermeldung: **NoClassDefFoundError** ?*
- ◆ Auffinden der Tests
- ◆ Ausführen der custom test suit und test setup
- ◆ Auftreten des ersten assertion fails

Troubleshooting JUNIT (2/2)

- ◆ Probleme beim Reload der Klassen zwischen den Tests
- ◆ Probleme beim Reload von Klassen in Zusammenhang mit XML

Immer noch Hilfe nötig?

- ◆ Pair programming
- ◆ <http://groups.yahoo.com/group/junit/>

Documenting Testing (IEEE 829) (1/2)

Was ist der IEEE 829 Standard?

- ◆ Standard für die Dokumentation von Softwaretests (Vorlage)
- ◆ Framework
- ◆ Definitionen
- ◆ Struktur

Documenting Testing (IEEE 829) (2/2)

Hinweise zur Anwendung von IEEE 829

- ◆ Zuerst über Problem nachdenken
- ◆ Verstehen der Abschnitte im Standard
- ◆ Anpassen der Vorlagen an Bedürfnisse
- ◆ Nichts ist verpflichtend im Standard

The background is a light blue grid. There are several thin blue lines: a vertical line on the left, a horizontal line near the top, and another horizontal line near the bottom. There are also small blue circular ornaments at the intersections of these lines. The text is centered in the upper half of the slide.

**Vielen Dank
für eure Aufmerksamkeit**