



\*software  
projekte  
prozesse

# eck\*cellent IT

\* LINEAS Project Services heißt jetzt **eck\*cellent IT** - besuchen Sie uns auf [www.eckcellent-it.de](http://www.eckcellent-it.de)



**eck\*cellent IT GmbH** Standorte in Braunschweig und Dortmund / Germany:  
Theodor-Heuss-Straße 2 38122 Braunschweig Phone +49 (0)531.70 22-20 00  
Emil-Figge-Straße 76 44227 Dortmund Phone +49 (0)231.97 42-55 80



- \* Thema: QS in Java-Projekten: Open Source Test Tools



Referent: Thomas Rustemeyer  
Diplom-Informatiker

# Software-Testing

Unter Software-Test versteht man die gezielte Prüfung von Systemen und Teilen auf die Erfüllung vorher definierter Anforderungen

- **Testplanung** bereits mit Abschluss der Anforderungsanalyse
- **Testdurchführung** ab dem ersten erfolgreichen Build (auch lokal)
- **Dokumentation** aller Testaktivitäten (nicht nur Fehler!)
- **Testmanagement** bezeichnet die systematische Termin- und Ressourcenplanung und Dokumentation
- **Testabdeckung** ist die Abdeckung von Anforderungen durch Tests



# Anforderungen

- Funktionale Anforderungen
- Nicht-funktionale Anforderungen
  - Performance
  - Usability
  - Stabilität / Robustheit
  - Sicherheit
  - Dokumentation (!)
  - Code-Qualität (!)
  - Entwicklungsprozess (!)



# Testmethoden

## > Unit-Tests

- > Testen die korrekte Funktionsweise einzelner Module (Klassen, Methoden, Funktionen)
- > Erstellung durch den Entwickler während der Entwicklung
- > Testen gegen die Feinspezifikation der Module

## > Integrations-Tests

- > Testen das Zusammenspiel einzelner oder aller Module
- > Überprüfen Schnittstellen



# Testmethoden

## > System-Tests

- > Testen das Gesamtsystem
- > Finden ab dem ersten erfolgreichen System-Build statt
- > Abnahme erfordert stabile Umgebung (Testsystem, Code-Freeze)

## > Reviews / Audits

- > Prüfung bestimmter Aspekte eines Systems
- > Können bereits ab Projektstart stattfinden

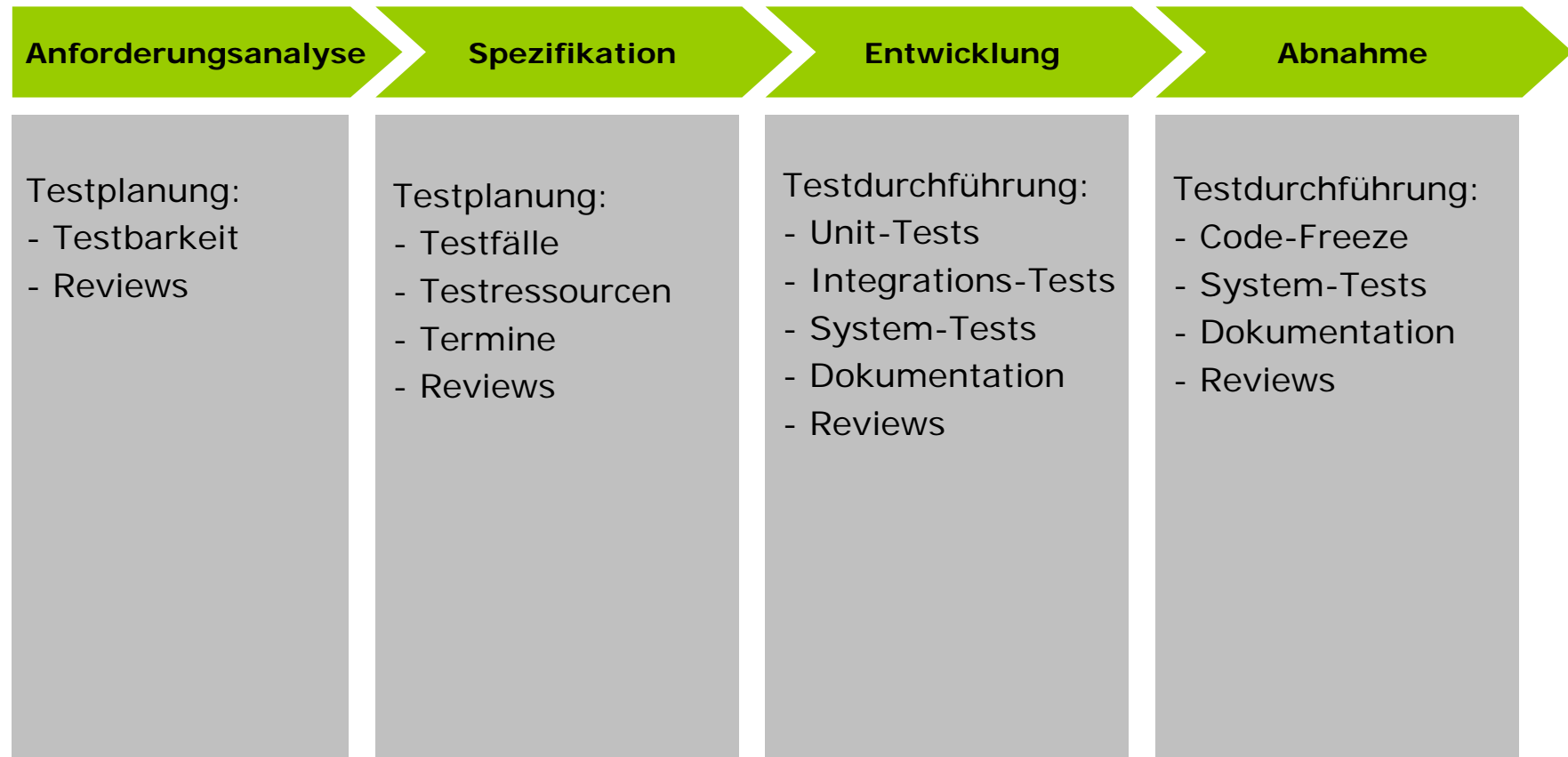


# Testmethoden

	Unit	Integration	System	Review
Funktional	+	+	+	
Performance	o	o	+	o
Usability			+	+
Stabilität		+	+	
Sicherheit		o	o	+
Dokumentation				+
Code				+
Prozess				+



# Testzyklus



# Java Testtools und -bibliotheken

Unit-Tests	Integrations-Tests	System-Tests	Reviews
<ul style="list-style-type: none"><li>&gt; JUnit</li><li>&gt; EasyMock</li><li>&gt; Cobertura</li><li>&gt; CruiseControl</li></ul>	<ul style="list-style-type: none"><li>&gt; JUnit</li><li>&gt; EasyMock</li><li>&gt; Cactus</li></ul>	<ul style="list-style-type: none"><li>&gt; JMeter</li><li>&gt; The Grinder</li><li>&gt; Selenium</li><li>&gt; Marathon</li></ul>	<ul style="list-style-type: none"><li>&gt; CheckStyle</li><li>&gt; Findbugs</li></ul>



# Java Testtools

## > JUnit

- > De-facto Standard für Unit-Tests
- > Gute Integration in alle gängigen Entwicklungsumgebungen

## > EasyMock

- > Mock-Objekte für externe Abhängigkeiten
- > Geeignet für JSP Taglibs, Servlets, ...

## > Cobertura

- > Bestimmung der Test-Abdeckung von Unit-Tests während der Ausführung
- > Line-Coverage und Branch-Coverage
- > Reports mit Auswertung auf Zeilenebene



# Cobertura

http://cobertura.sourceforge.net/sample/

Live Search

File Edit View Favorites Extras ?

Coverage Report

### Coverage Report - All Packages

Package	# Classes	Line Coverage	Branch Coverage	Complexity
<b>All Packages</b>	55	75% 1625/2179	64% 472/738	2.319
<a href="#">net.sourceforge.cobertura.ant</a>	11	52% 170/330	43% 40/94	1.848
<a href="#">net.sourceforge.cobertura.check</a>	3	0% 0/150	0% 0/76	2.429
<a href="#">net.sourceforge.cobertura.coveragedata</a>	13	N/A N/A	N/A N/A	2.277
<a href="#">net.sourceforge.cobertura.instrument</a>	10	90% 460/510	75% 123/164	1.854
<a href="#">net.sourceforge.cobertura.merge</a>	1	86% 30/35	88% 14/16	5.5
<a href="#">net.sourceforge.cobertura.reporting</a>	3	87% 116/134	80% 43/54	2.882
<a href="#">net.sourceforge.cobertura.reporting.html</a>	4	91% 475/523	77% 156/202	4.444
<a href="#">net.sourceforge.cobertura.reporting.html.files</a>	1	87% 39/45	62% 5/8	4.5
<a href="#">net.sourceforge.cobertura.reporting.xml</a>	1	100% 155/155	95% 21/22	1.524
<a href="#">net.sourceforge.cobertura.util</a>	9	60% 175/291	69% 70/102	2.892
<a href="#">someotherpackage</a>	1	83% 5/6	N/A N/A	1.2

Report generated by [Cobertura](#) 1.9 on 6/9/07 12:37 AM.

Internet 100%



# CruiseControl

- Tools für Continuous Integration
- Sofortiger Checkout, Build, Testlauf
- Mail-Benachrichtigung über Build-Ergebnisse
- Web-Oberfläche für Status



# Testautomatisierung

## > Apache Jakarta JMeter

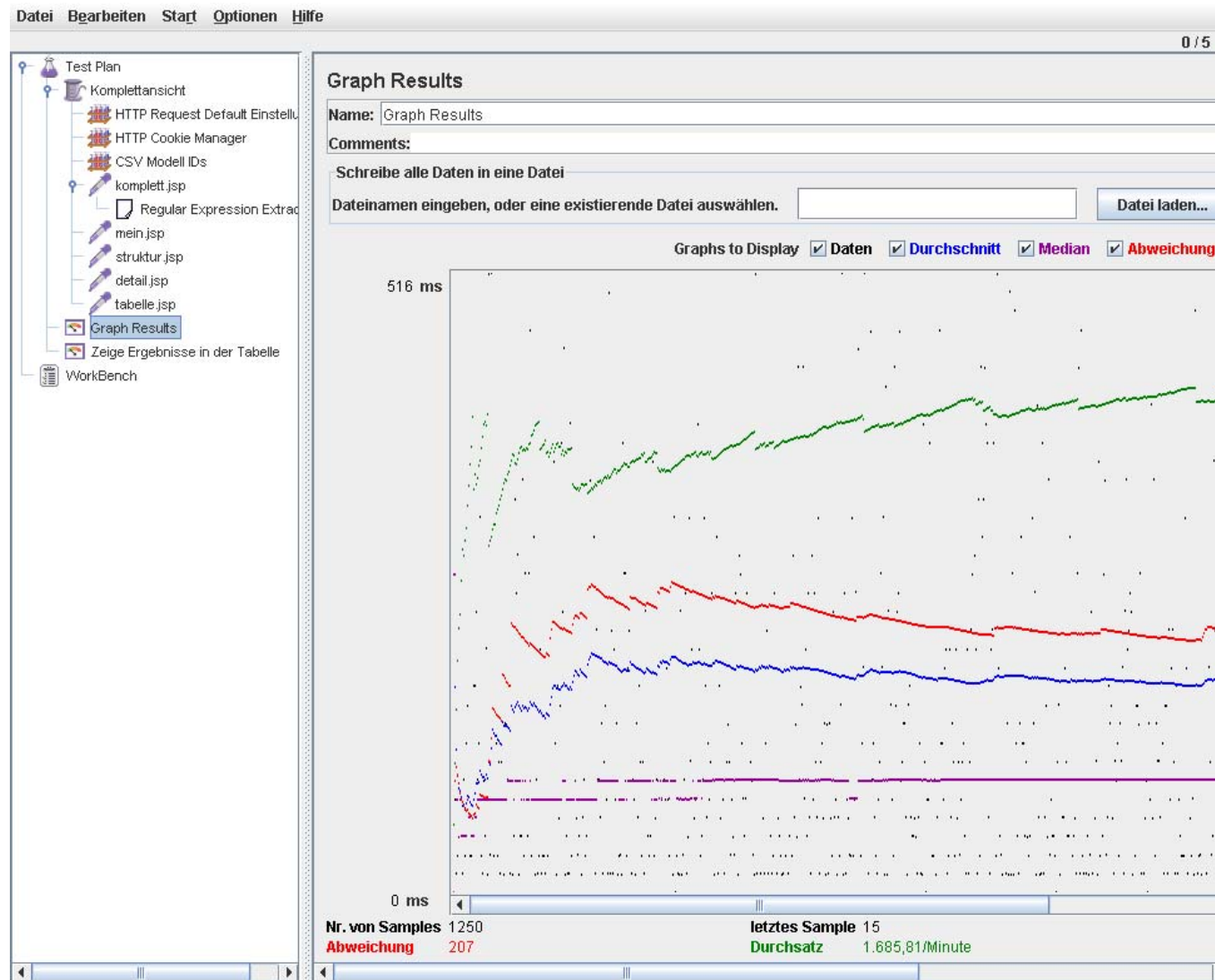
- > Automatisiertes Testen von Java-Anwendungen, insbesondere Web
- > Einfaches Zusammenstellen linearer Testabläufe (HTTP-Requests)
- > Einfache Funktionstests, Lasttests und Performancetests

## > The Grinder

- > Automatisiertes Testen von Java-Anwendungen
- > Jython-Testskripte auch für komplexe Abläufe und Funktionstests
- > Hohe Parallelisierbarkeit durch Agent-Architektur



# JMeter



# The Grinder

**Grinder Konsole**

Datei Aktion Verteilung Hilfe

Proben Interval: 1000 ms

Ignoriere 0 Proben

Endlose Probensammlung!

**Sammle Proben: 2**

**0,00 TPS**

Gesamt

- ms (Mittel)
- 0,00 TPS (Mittel)
- 0,00 TPS (Spitze)
- 0 Transaktionen
- 2 Fehler**

**Test 1 (Database connection)**

- ms (Mittel)
- 0,00 TPS (Mittel)
- 0,00 TPS (Spitze)
- 0 Transaktionen
- 2 Fehler**

**Test 2 (FIRSTspirit connection)**

- ms (Mittel)
- 0,00 TPS (Mittel)
- 0,00 TPS (Spitze)
- 0 Transaktionen
- 0 Fehler

**Test 3 (Create deployment queue entry)**

- ms (Mittel)
- 0,00 TPS (Mittel)
- 0,00 TPS (Spitze)
- 0 Transaktionen
- 0 Fehler

**Test 4 (Check deployment queue entry)**

- ms (Mittel)
- 0,00 TPS (Mittel)
- 0,00 TPS (Spitze)
- 0 Transaktionen
- 0 Fehler

**Test 5 (Test successful deployment)**

- ms (Mittel)
- 0,00 TPS (Mittel)
- 0,00 TPS (Spitze)
- 0 Transaktionen
- 0 Fehler



# Testautomatisierung

## > Marathon

- > Automatisiertes Testen von Swing-Anwendungen
- > Recording von Bedienungsabläufen
- > Steuerung über Jython-Skripte

## > Selenium

- > Automatisiertes Testen von Web-Anwendungen
- > Steuerung des Browsers über JavaScript, dadurch Ausführung aller Bestandteile einer HTML-Seite (inklusive JavaScript)
- > Recording von Bedienungsabläufen
- > Überprüfung von Darstellung, Sichtbarkeit, Inhalt



# Marathon

The screenshot displays a software development environment. On the left, a window titled "Simple Widgets" shows a form with fields for "First Name", "Password", "Email", "Address", "Country", "Gender", "Annual Income (\$)", "Languages", and "Password Expiry". The "First Name" field contains "Dakshinamurthy Karra".

In the center, a "Navigator" window shows a tree view of "TestCases" with a list of Python files including "SampleApp.py", "buttondemo.py", "buttonhtmldemo.py", "checkboxdemo.py", "colorchooserdemo.py", "comboboxdemo.py", "customcomboboxdemo.py", "dialogdemo.py", "dialogdemo\_nonmodal.py", "dragcolortextfield.py", "dragcopy.py", "dynamicreedemo.py", "formattedtextfielddemo.py", "formatterfactorydemo.py", "framedemo.py", "glasspanedemo.py", "htmldemo.py", "labeldemo.py", "layeredpanedemo.py", "listdemo.py", "menudemo.py", "menugluedemo.py", and "menulayoutdemo.py".

On the right, an "Editor" window shows a Python test case:

```
useFixture(default)

def test():
    window('Simple Widgets')
    select('First Name', 'Dakshinamurthy Karra')
    select('Password', 'Password')
    select('Email', 'kd@jaliansystems.com')
    select('Address', '''No.599, M.M.Layout
Kavalbyrasandra
Bangalore - 560 032''')
    sleep(200)
    click('Female')
    select('Female', 'true')
    select('Annual Income #{$#}', 'Between 10K and 20K-Between 20K and 30K')
    click('English')
    select('English', 'true')
    click('Hindi')
    select('Hindi', 'true')
    select('Slider', '35')
    close()
```

Below the editor is a "Results" window showing "No Errors". At the bottom, a status bar indicates "Playing" and a table with columns "File", "Location", and "Insert".



# Selenium

The screenshot shows a Firefox browser window with the Selenium IDE extension installed. The browser is displaying a Google search for "selenium ide". The Selenium IDE interface is overlaid on the browser, showing a table of commands and a log of actions.

Command	Target	Value
open	/webhp?hl=en	
type	q	selenium ide
clickAndWait	btnG	
verifyTextPr...	selenium-ide.openqa...	

verifyTextPresent  
selenium-ide.openqa.org/ - Find

Log Reference UI-Element Rollup

verifyTextPresent(pattern)  
Generated from isTextPresent(pattern)  
Arguments:  
• pattern - a pattern to match with the text of the page  
Returns:  
true if the pattern matches the text, false otherwise

Done

Web Images Maps News Shopping Gmail more Sign in

Google selenium ide

Web Results 1 - 10 of about 218,000 for selenium ide. (0.04 seconds)

**Selenium IDE: Selenium IDE**  
Selenium IDE is an integrated development environment for Selenium tests. It is implemented as a Firefox extension, and allows you to record, edit, ...  
[selenium-ide.openqa.org/](#) - 4k - Cached - Similar pages

**Selenium IDE: Download**  
Selenium IDE: Download.  
[selenium-ide.openqa.org/download.jsp](#) - 2k - Cached - Similar pages

**Selenium IDE Intro**  
There is a Firefox extension called Selenium IDE made by the folks over at OpenQA.org. It is a very easy to use and powerful tool for controlling, ...  
[www.dynamitemap.com/selenium/](#) - 9k - Cached - Similar pages

**Software Testing - Selenium IDE**  
Selenium IDE is the easiest way to use Selenium and most of the time it also serves as a starting point for your automation. Selenium IDE comes as an ...  
[www.testinggeek.com/selenium.asp](#) - 34k - Cached - Similar pages

**Introduction to the Selenium IDE**  
If you would like to try out Selenium and the Selenium IDE, you should check out this tutorial (complete with screencasts) that will get you up and running ...  
[www.dynamitemap.com/articles/2008/introduction-to-the-selenium-ide](#)



# Review Tools

## > Checkstyle

- > Automatische Prüfung auf Einhaltung von Code Conventions
- > Gute Anpassbarkeit von Regelsätzen
- > Formale Prüfung, keine Inhaltsprüfung

## > Findbugs

- > Statische Code-Analyse
- > Aufdeckung von verschiedenen Fehlern, Schwachstellen, Bad Practices, Ressourcenproblemen



## Zusammenfassung

- Eine durchgängige Qualitätssicherung von Projekt-Anfang bis -Ende ist wichtig
- Die QS muss verschiedene Aspekte des Systems testen
- Tests müssen geplant und strukturiert durchgeführt werden
- Tests lassen sich durch geeignete Tools automatisieren
- Reviews können durch Tools unterstützt werden





\*software  
projekte  
prozesse

# eck\*cellent IT

\* Vielen Dank für Ihre Aufmerksamkeit!

