



Pleiten, Pech und PatternTesting

Ein Drama in 5,5 Akten

oliver.boehm@agentes.de
christian.heise@agentes.de



**neugierig?
ängstlich?**

re

re

Pleiten

Pech

**Pattern
Testing**

Pattern
Testing





Prolog

```
...
declare
  vertical_veloc_sensor: float;
  horizontal_veloc_sensor: float;
  vertical_veloc_bias: integer;
  horizontal_veloc_bias: integer;
  ...
begin
  declare
    pragma suppress(numeric_error, horizontal_veloc_bias);
  begin
    sensor_get(vertical_veloc_sensor);
    sensor_get(horizontal_veloc_sensor);
    vertical_veloc_bias := integer(vertical_veloc_sensor);
    horizontal_veloc_bias := integer(horizontal_veloc_sensor);
    ...
  exception
    when numeric_error => calculate_vertical_veloc();
    when others => use_irs1();
  end;
end irs2;
```



Laurent DEJAULT

<http://www.flickr.com/photos/lolodoc/3845299578/>



Pattern



Front End

Middle Tier

Back End

<http://www.flickr.com/photos/ann/1329361/>

AntiPattern



Lava-Flow



By Dan Hecker
WildPhotons.com

```
try {  
    ...  
} catch (MalformedURLException e) {  
    // TODO Auto-generated catch block  
    e.printStackTrace();  
}
```

AntiPattern Testing

```
try {  
    ...  
} catch (MalformedURLException e) {  
    // TODO Auto-generated  
    e.printStackTrace();  
}
```



J2SE 5.0 Monitoring & Management Console: 2209@localhost

Connection

Summary Memory Threads Classes **MBeans** VM

MBeans

Tree

- ▶ Catalina
- ▶ JMImplementation
- ▶ Users
- ▶ java.lang
- ▶ java.util.logging
- ▼ patterntesting.runtime.monitor
 - ClasspathMonitor
 - ProfileStatistic

Attributes Operations Notifications Info

Name	Value
BootClasspath	java.lang.String[7]
Classpath	java.lang.String[12]
ClasspathClasses	java.lang.String[1316]
DoubletClasspath	java.lang.String[4]
Doublets	java.lang.String[17]
LoadedClasses	java.lang.String[253]
LoadedPackages	java.lang.String[291]
UnusedClasses	gui.Hauptframe \$ BerechneHypoButtonListener gui.Hauptframe \$ BerechneZinsButtonListener gui.Hauptframe \$ HypoButtonListener gui.Hauptframe \$ ZinsButtonListener onlinerechner.AutoRechnerAction onlinerechner.HypoZinsAction onlinerechner.WaehrungsRechner onlinerechner.WaehrungsServer onlinerechner.ZinsAction
UnusedClasspath	java.lang.String[5]
UsedClasspath	java.lang.String[7]

Refresh

J. Dan K...
WildPhotons.com

ProfileStatistic

Label	Unit	LastValue	Hits	Avg	Total
<code>patterntesting.sample.World.main(String[])</code>	ms.	20	1	20	20
<code>patterntesting.sample.World.printHelloAgain()</code>	ms.	0	1	0	0
<code>new patterntesting.sample.World()</code>	ms.	0	0	0	0
<code>patterntesting.sample.World.printBye()</code>	ms.	0	0	0	0

Walt



It is not allowed to use the JDBC API from here!



HTTP Status 500 -

type Exception report

message

description The server encountered an internal error () that prevented it from fulfilling this request.

exception

```
java.lang.NullPointerException
    patterntesting.sample.jfs2010.web.LoginServlet.doPost(LoginServlet.java:34)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:637)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
```

note The full stack trace of the root cause is available in the Apache Tomcat/6.0.20 logs.

Apache Tomcat/6.0.20

User

Password

```
protected void doPost(HttpServletRequest request,
    HttpServletResponse response) throws ServletException, IOException {
    String password = request.getParameter("password");
    String userName = request.getParameter("user");

    User user = LoginService.login(userName, password);
    ServletContext context = getServletContext();
    RequestDispatcher dispatcher = context.getRequestDispatcher("/overview.jsp");
    request.setAttribute("user_login", user.getName());
    dispatcher.include(request, response);
}
```

java.lang.NullPointerException

at sample.jfs2010.web.LoginServlet.doPost(LoginServlet.java:35)

at sample.jfs2010.web.LoginServletTest.testDoPost(LoginServletTest.java:52)

A decorative border with a red, scalloped, fabric-like texture frames the top and sides of the page. The background is white.

**Erster Akt:
Ein Königreich
für einen
Test**

```
public void testDoPost() throws IOException {
    LoginServlet servlet = new LoginServlet();
    MockHttpServletRequest req = new MockHttpServletRequest();
    MockHttpServletResponse resp = new MockHttpServletResponse();
    req.setParameter("user", "guiseppa");
    req.setParameter("password", "topsecret");
    servlet.doPost(req, resp);
    log.info("Yeah, no exception!");
}
```

java.lang.NullPointerException

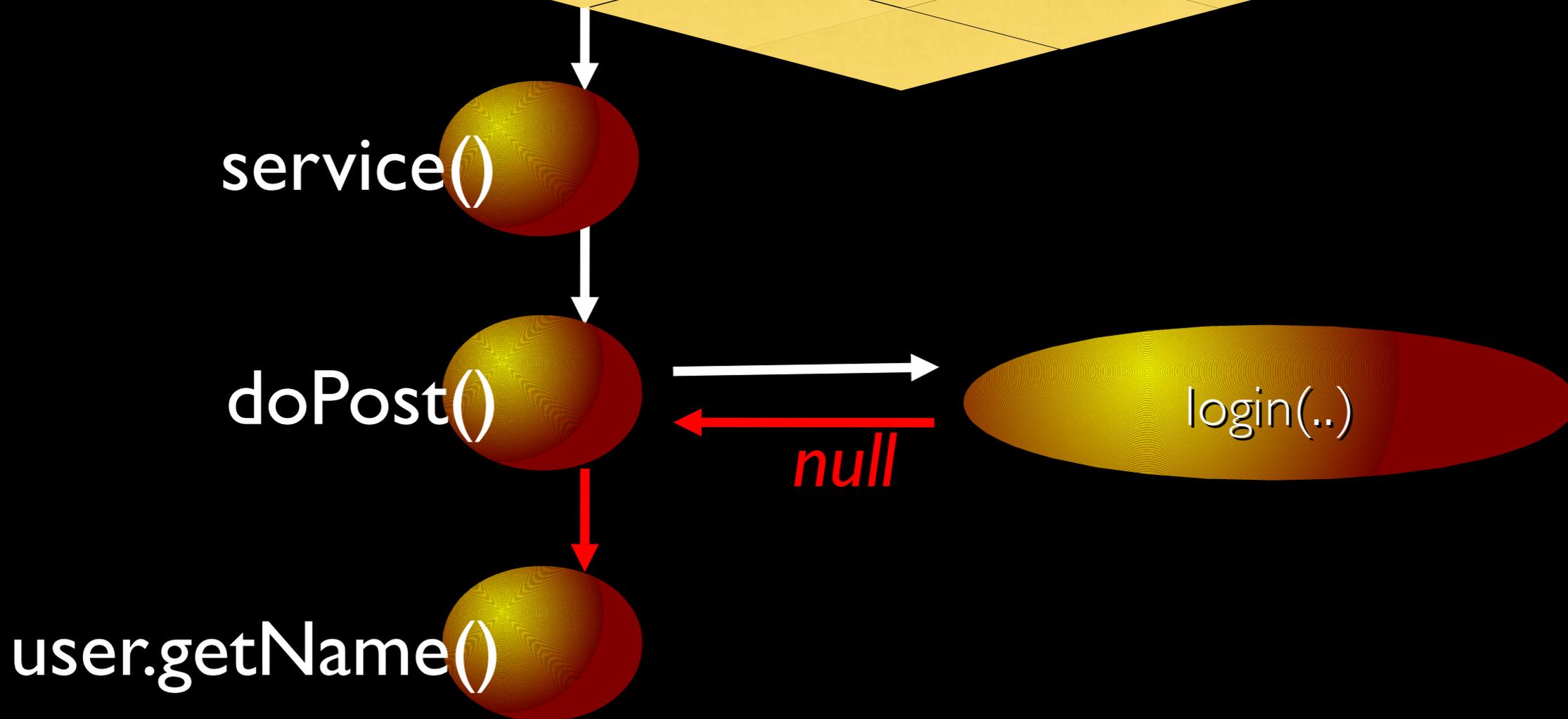
at sample.jfs2010.web.LoginServlet.doPost(LoginServlet.java:35)

at sample.jfs2010.web.LoginServletTest.testDoPost(LoginServletTest.java:52)

The image shows a stage with a white floor and a red curtain backdrop. The text is centered on the white stage area.

**Akt 2.0:
PatternTesting
betritt die Bühne**

ohne PatternTesting



ohne PatternTesting

doPost() ist schuld!?

service()

doPost()

NPE

user.getName()

login(..)

null

Properties for patterntesting-samples-jfs2010

type filter text

- ▶ Resource
- ▶ AspectJ Build
- AspectJ Compiler
- Builders
- CVS
- Deployment Assembl
- JAutodoc
- Java Build Path
- ▶ Java Code Style
- ▶ Java Compiler
- ▶ Java Editor
- Javadoc Location
- ▶ JavaScript
- JSP Fragment
- Project Facets
- Project References
- Run/Debug Settings
- Server
- Service Policies
- Targeted Runtimes
- ▶ Task Repository

AspectJ Build

Inpath Aspect Path Output jar

Aspects added to the aspect path are applied to types in the project:

ng/patterntesting-check-rt/1.0.0/patterntesting-check-rt-1.0.0.jar
ting-rt/1.0.0/patterntesting-rt-1.0.0.jar
sting/patterntesting-exception/1.0.0/patterntesting-exception-1.0.0.jar

Add JARs...

Add External JARs...

Add Variable...

Add Class Folder...

Add Library...

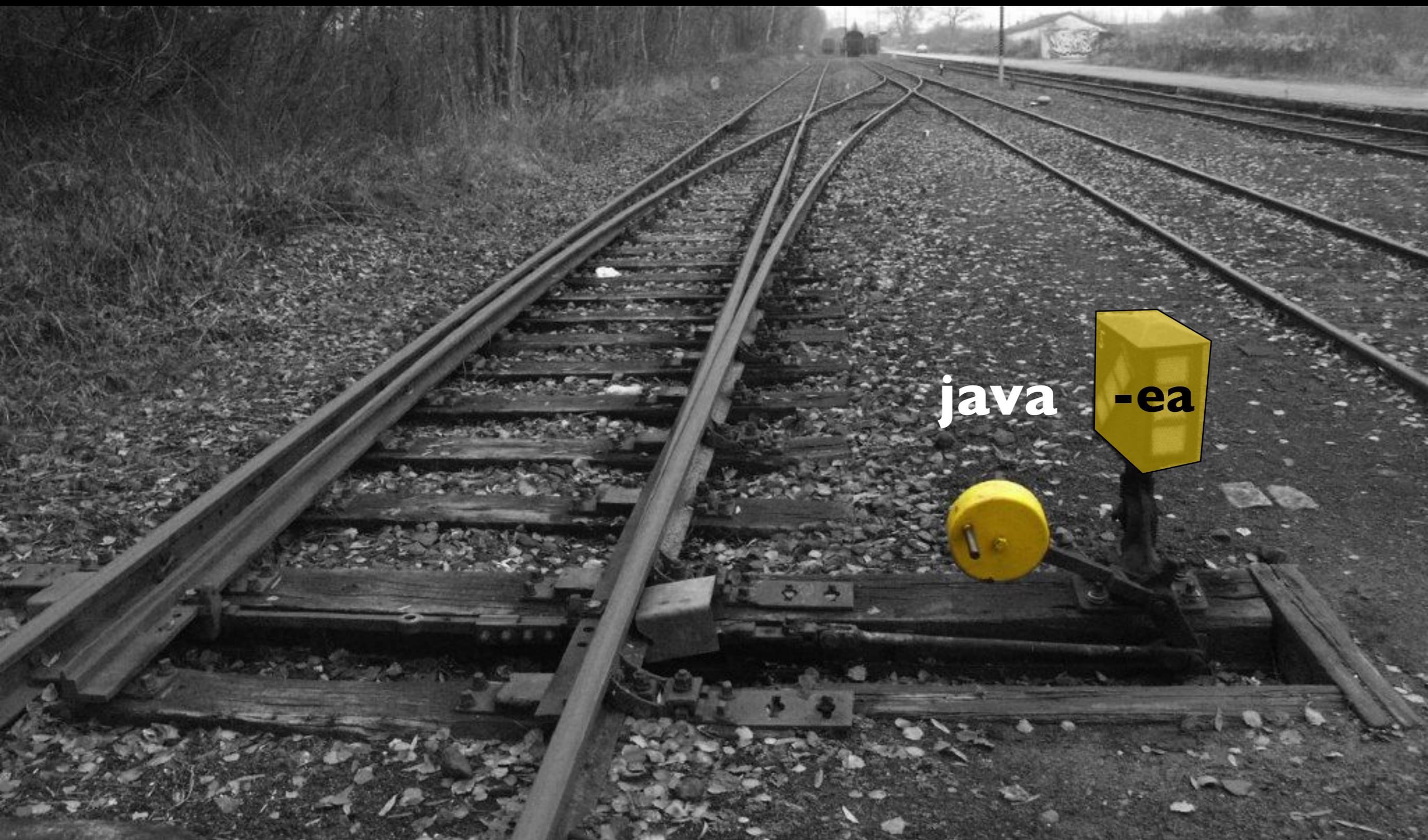
Add Project...

Edit...



Cancel

OK



java

-ea

mit PatternTesting



service()

doPost()

AssertionError:
login() returns null!

java.lang.AssertionError: **User LoginService.login(String, String) returns null!**

...

at sample.jfs2010.LoginService.login(LoginService.java:62)

at sample.jfs2010.web.LoginServlet.doPost(LoginServlet.java:26)

...

```
public static User login(final String name, final String passwd) throws IOException,
    LoginException {
    User user = null;
    URL url = getLoginURL(name);
    HttpURLConnection connection = (HttpURLConnection) url.openConnection();
    int rc = connection.getResponseCode();
    log.debug("RC=" + rc);
    if (rc != 404) {
        user = new User(name);
    }
    return user;
}
```

```
public static User login(final String name, final String passwd) throws IOException,
    LoginException {
    URL url = getLoginURL(name);
    HttpURLConnection connection = (HttpURLConnection) url.openConnection();
    int rc = connection.getResponseCode();
    log.debug("RC=" + rc);
    if (rc == 404) {
        throw new LoginException(name + ": login failed");
    }
    return new User(name);
}
```



HTTP Status 500 -

type Exception report

message

description The server encountered an internal error () that prevented it from fulfilling this request.

exception

```
javax.servlet.ServletException: javax.security.auth.login.LoginException: guiseppe: login failed
    patterntesting.sample.jfs2010.web.LoginServlet.doPost(LoginServlet.java:34)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:637)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
```

root cause

```
javax.security.auth.login.LoginException: guiseppe: login failed
    patterntesting.sample.jfs2010.LoginService.login(LoginService.java:65)
    patterntesting.sample.jfs2010.web.LoginServlet.doPost(LoginServlet.java:28)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:637)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
```

note The full stack trace of the root cause is available in the Apache Tomcat/6.0.20 logs.

The image shows a stage with a white floor and a red curtain backdrop. The text is centered on the white stage area.

**Pause:
Ein Blick
hinter die Bühne**

AOP-Begriffe

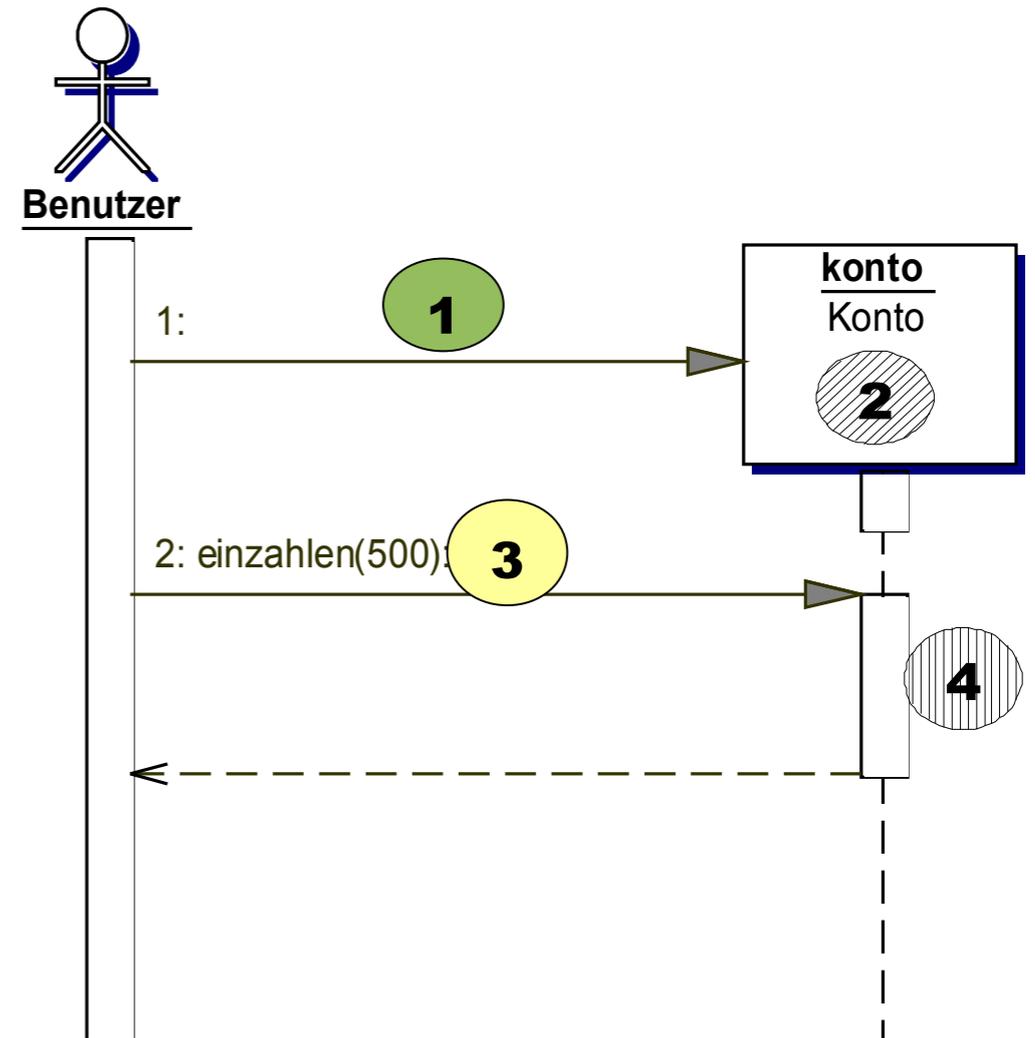
- **Joinpoint**
 - Punkte im Programm, an denen Code erweitert oder modifiziert werden soll
- **Pointcut**
 - eine Auswahl von Joinpoints
- **Advice**
 - der Code für den Pointcut
- **Aspect**
 - Konstrukt, in dem das obige abgelegt wird

Joinpoints können sein

- Aufrufen einer Methode
- Ausführen einer Methode
- Zugriff auf eine Variable
- Behandeln einer Exception
- Initialisierung einer Klasse
- Initialisierung eines Objekts

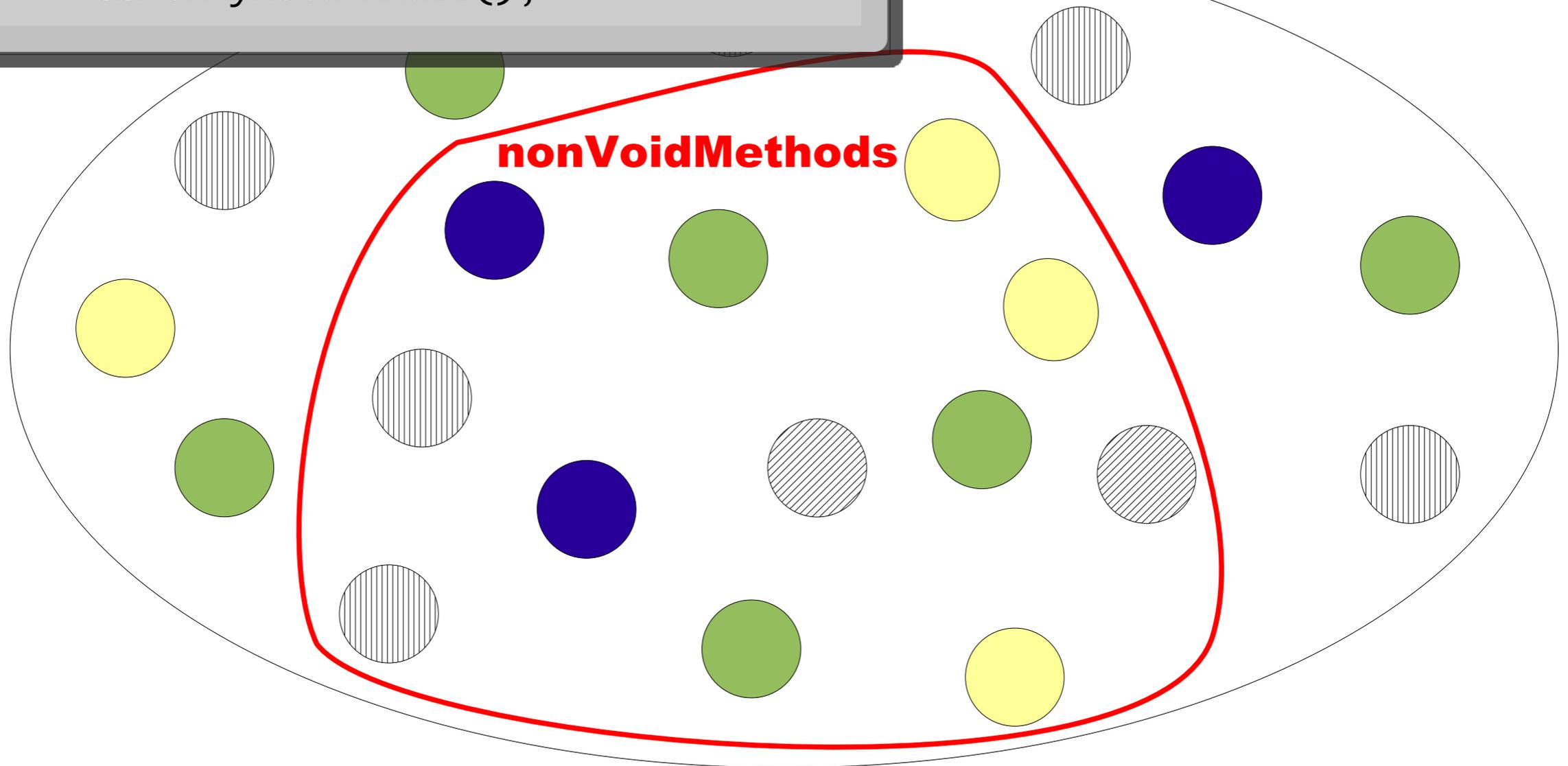
- (1) Konstruktor aufrufen
- (2) Objekt initialisieren
- (3) Methode aufrufen
- (4) Methode ausführen

```
Konto konto = new Konto();  
konto.einzahlen(500.0);
```



Pointcuts

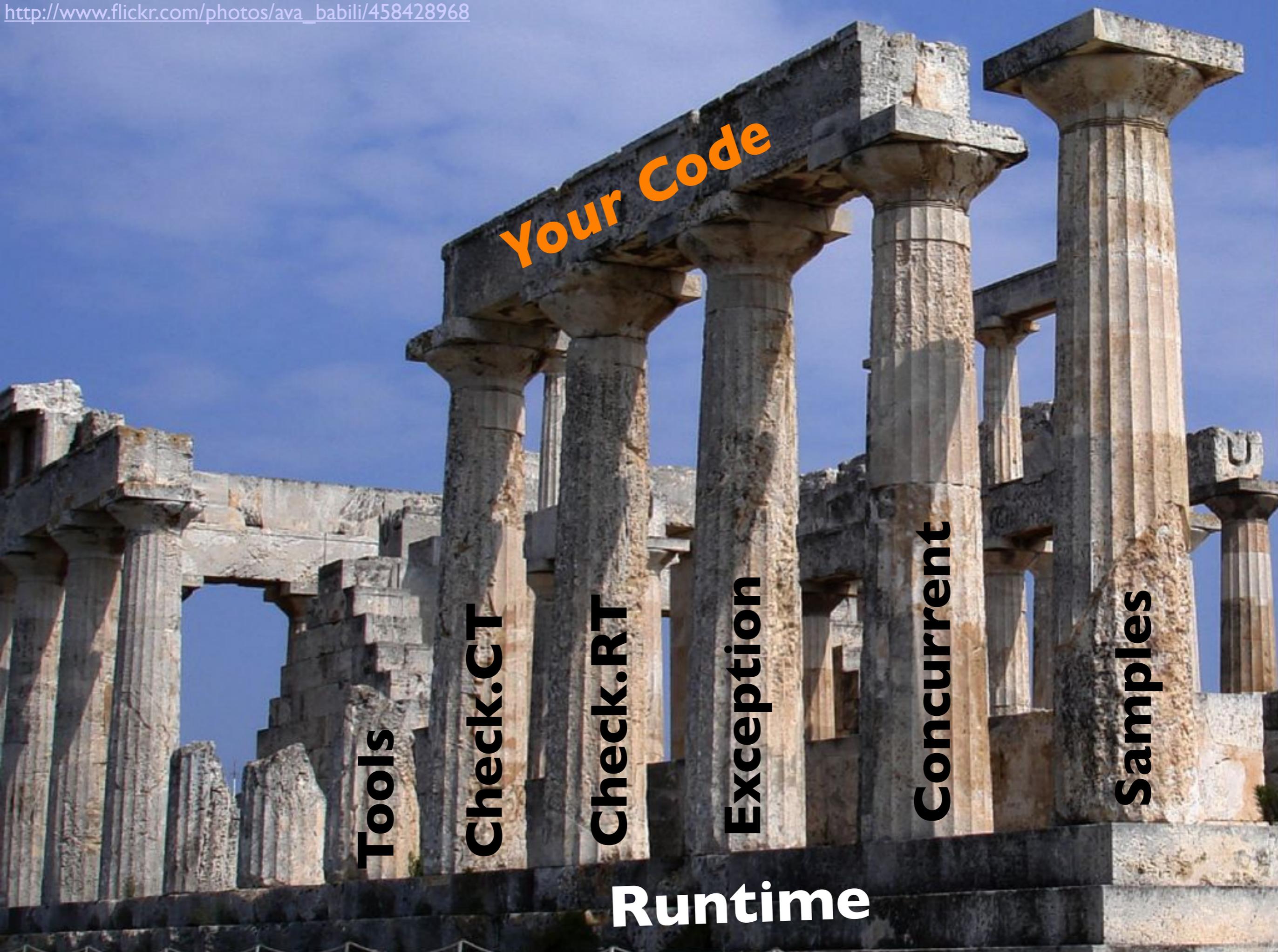
```
private pointcut nonVoidMethods() :  
    execution(public Object+ *.*(..))  
    && applicationCode()  
    && !mayReturnNull();
```



Advice

- Code, der eingewebt wird
 - before()
 - after()
 - around()
- Ähnlichkeit mit Methoden

```
after() returning(Object returned) : nonVoidMethods() {  
    assert returned != null : thisJoinPoint.getSignature()  
        + " returns null!";  
}
```



Your Code

Tools

Check.CT

Check.RT

Exception

Concurrent

Samples

Runtime



**Dritter Akt:
Kein Anschluss
unter dieser Nummer!**

```
int rc = HttpURLConnection.getResponseCode();
```

**NoRouteToHostException:
No route to host**



```
int rc = HttpURLConnection.getResponseCode();
```

NoRouteToHostException:
No route to 216.34.181.96



```
socket = new Socket(host, port);
```

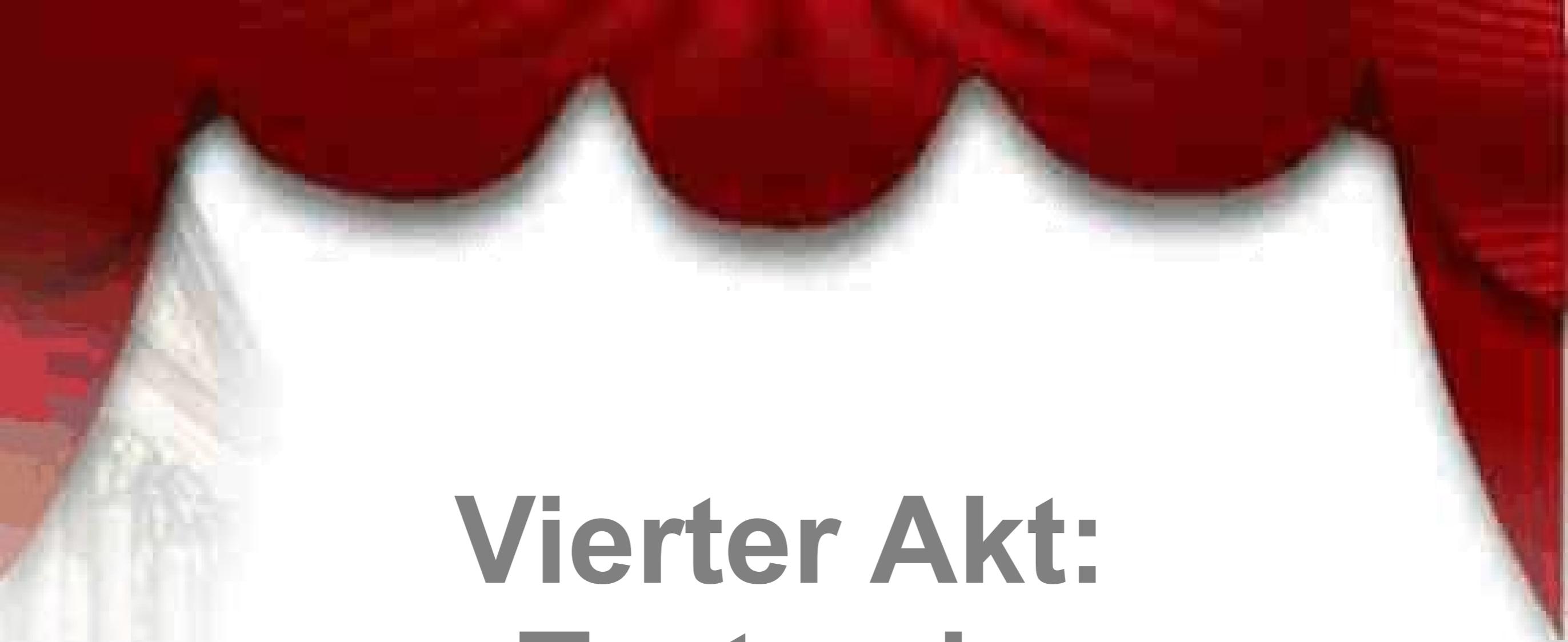


**ConnectException:
Connection refused**

```
socket = new Socket(host, port);
```

**ConnectException:
Connection to bnd:815 refused**





**Vierter Akt:
Testen in
vollen Zügen**

```
@RunWith(SmokeRunner.class)  
public final class LoginServletTest {
```

```
    ...
```

```
    @IntegrationTest("needs LoginService")
```

```
    @Test
```

```
    public void testDoPost() throws IOException {
```

```
        ...
```

```
    }
```

```
}
```

Run Configurations

Create, manage, and run configurations

Create a configuration that will launch a JUnit test.



type filter text

- JUnit4RunnerTest
- LocalhostTest
- LoginServletTest
- ParallelRunnerTest
- ParallelRunnerTest-integrationT
- patterntesting-check-ct
- patterntesting-check-rt
- patterntesting-check-rt-SmokeT
- patterntesting-concurrent
- patterntesting-concurrent-integ
- patterntesting-exception
- patterntesting-rt
- patterntesting-rt-it
- patterntesting-rt-SmokeTest
- patterntesting-samples-intro
- patterntesting-samples-jfs2010
- patterntesting-samples-jfs2010**
- patterntesting-tools
- patterntesting-tools-integration

Filter matched 122 of 131 items

Name: patterntesting-samples-jfs2010-integrationTest

Test (x)= Arguments Classpath JRE Source »2

Program arguments:

Variables...

VM arguments:

Variables...

Working directory:
 Default: \${workspace_loc:patterntesting-samples-jfs2010}
 Other:
Workspace... File System... Variables...

Apply Revert

Close Run

@SmokeTest





@Broken

```
@RunWithOn("Mac OS X")
```

```
@SkipTestOn("Linux")
```



@Deprecated



**Fünfter Akt:
Die Ausnahme von
der Ausnahme**

Internal Error



An internal error occurred while showing an internal error.
You are recommended to exit the workbench.
Subsequent errors may happen and may terminate the workbench without
warning.
See the .log file for more details.

Do you want to exit the workbench?

Yes

No

@TestException

Exception

Factory

A woman with long dark hair is sitting in a meditative pose on a bed. She is facing away from the camera, with her back to it. The room is very dark, with a soft light source from the side illuminating her hair and the contours of her body. The text "Schlussakt: mach('s) mit..." is overlaid in the center of the image in a white, sans-serif font.

**Schlussakt:
mach('s) mit...**

```
<dependency>  
  <groupId>org.patterntesting</groupId>  
  <artifactId>patterntesting-rt</artifactId>  
  <version>1.0.2</version>  
</dependency>
```

1.0.x noch nicht im Maven Central Repository (30.9.)

```
<repositories>  
  <repository>  
    <id>agentes</id>  
    <name>agentes Repository</name>  
    <url>http://labs.agentes.de/repository</url>  
  </repository>  
</repositories>
```

Install

Available Software

Check the items that you wish to install.

Work with:

Find more software by working with the ['Available Software Sites'](#) preferences.

type filter text

Name	Version
<input checked="" type="checkbox"/> AspectJ Development Tools (Required)	
<input checked="" type="checkbox"/> AspectJ Development Tools	2.0.1.e35x-20091001-1600
<input type="checkbox"/> AspectJ Development Tools Source (Optional)	
<input type="checkbox"/> Other AJDT Tools (Optional)	

Details

Show only the latest versions of available software Hide items that are already installed
 Group items by category What is [already installed?](#)
 Contact all update sites during install to find required software

Einbindung Light



Java

patterntesting-rt.jar

ASPECTJRT_LIB
(aspectjrt.jar)

@RunWith(**SmokeRunner.class**)

@Broken

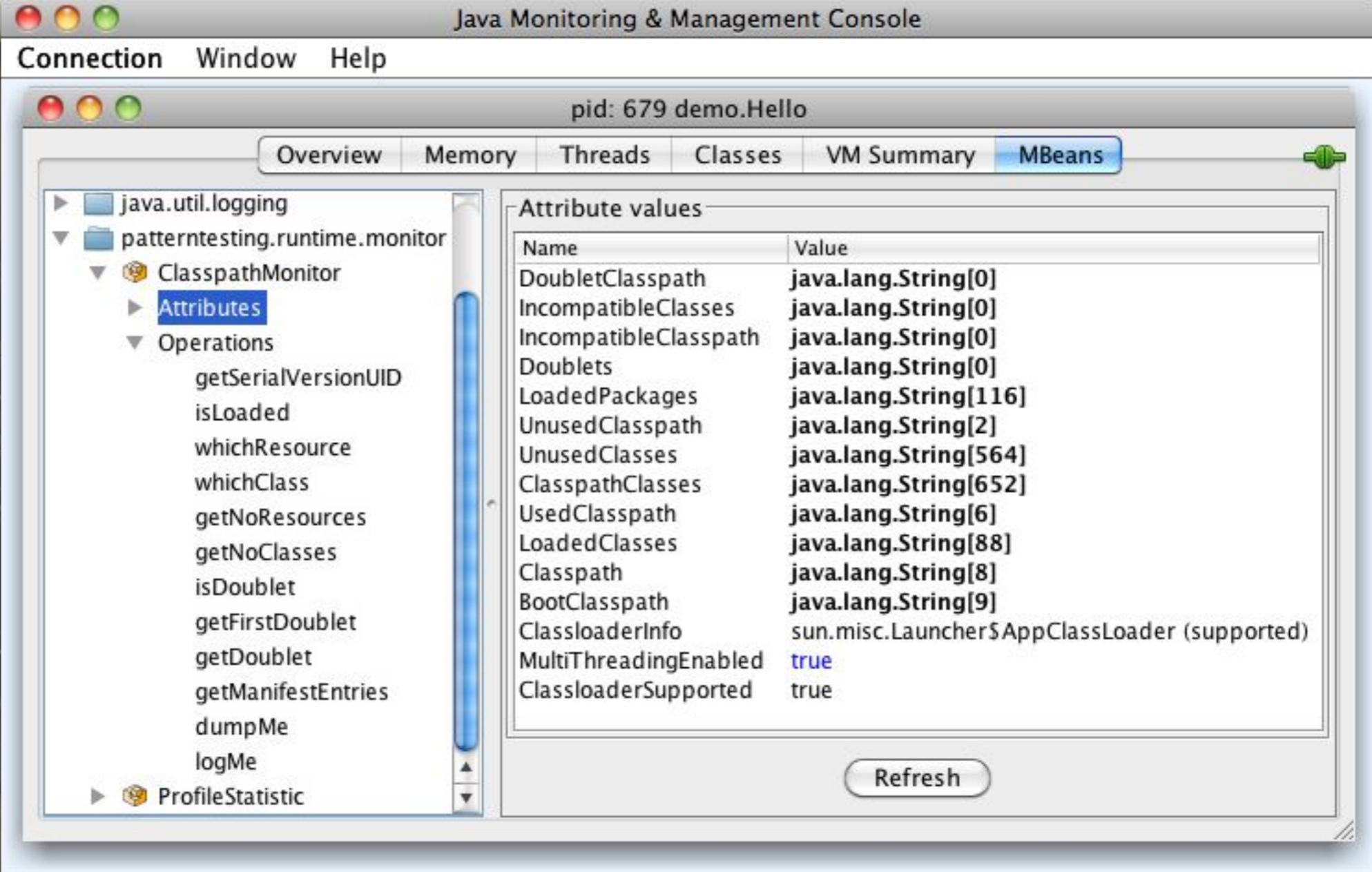
@SkipTestOn

@RunTestOn

@SmokeTest

@IntegrationTest

Pattern Testing Runtime



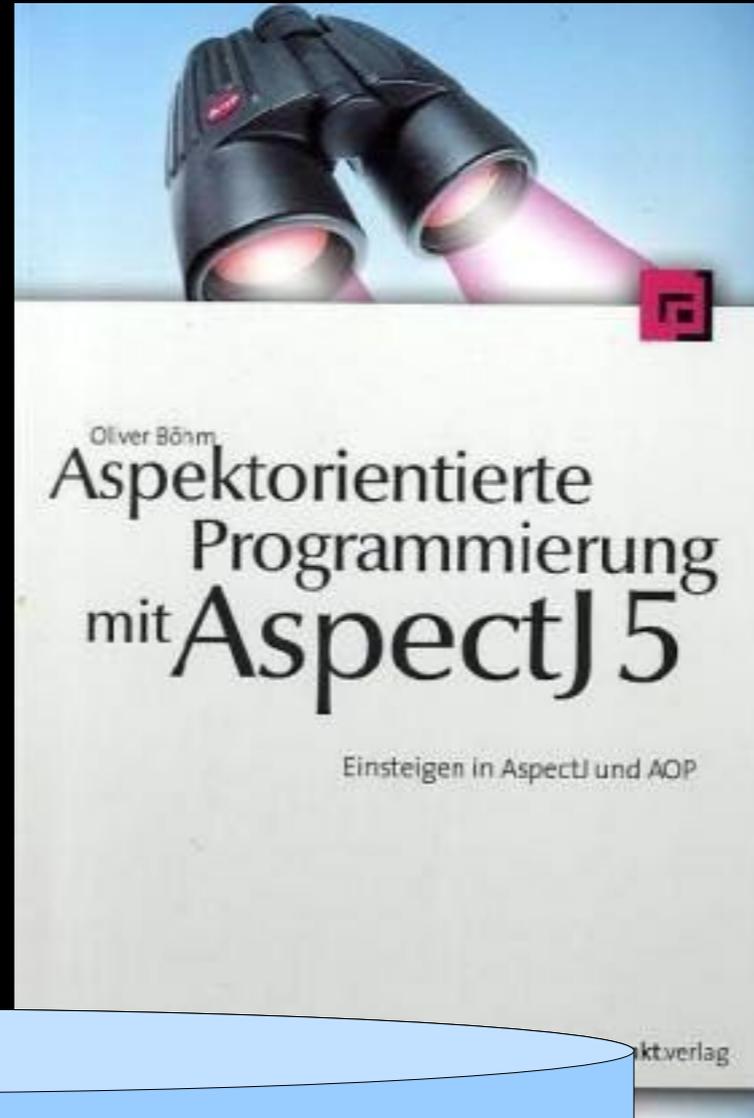
The screenshot shows the Java Monitoring & Management Console window. The title bar reads "Java Monitoring & Management Console". Below the title bar, there are tabs for "Overview", "Memory", "Threads", "Classes", "VM Summary", and "MBeans". The "MBeans" tab is selected. The left pane shows a tree view of MBeans under the "patternesting.runtime.monitor" package, with "ClasspathMonitor" expanded to show its "Attributes" and "Operations". The right pane displays the "Attribute values" for the selected MBean, listing various attributes and their values.

Name	Value
DoubletClasspath	java.lang.String[0]
IncompatibleClasses	java.lang.String[0]
IncompatibleClasspath	java.lang.String[0]
Doublets	java.lang.String[0]
LoadedPackages	java.lang.String[1 16]
UnusedClasspath	java.lang.String[2]
UnusedClasses	java.lang.String[564]
ClasspathClasses	java.lang.String[652]
UsedClasspath	java.lang.String[6]
LoadedClasses	java.lang.String[88]
Classpath	java.lang.String[8]
BootClasspath	java.lang.String[9]
ClassLoaderInfo	sun.misc.Launcher\$AppClassLoader (supported)
MultiThreadingEnabled	true
ClassLoaderSupported	true

Refresh

```
ClasspathMonitor.registerAsMBean();
```

Normal



patterntesting-rt.jar

+ weitere...

Aspect Path



Test-Version

- AspectJ
- patterntesting-rt.jar
- patterntesting-check-rt.jar
- u.a.
- andere Libs

★ Release-Version

- Pure Java
- patterntesting-rt.jar
- andere Libs



@NullArgsAllowed

@MayReturnNull

@LogRuntimeException

@NotYetImplemented

@PublicForTesting

@TestThread

@TestException

@ProfileMe

Zugabe?



@RunWith(ParallelRunner.class)

1.0.2

1.1.0



- ▶ **Dokumentation**
- ▶ **Wiki**
- ▶ **PatternTesting Samples**
- ▶ **patterntesting-all**
- ▶ **Multi-Threading**

WANTED

Java-Entwickler
AspectJ-Entwickler
Maven-Experte

Sachdienliche Hinweise bitte an:
jobs@patterntesting.de

The Vision

BombSquad

		1	2	2	1
1	1	1			
					
					1



Zeit 7.3
Bomben 5
Raster 6 x 6

Raster

To be the Minesweeper
for Your Java Code!

Tragedy?

in den Hauptrollen:

PatternTesting
Christian Heise
Oli B.

Regie:

Rainer Zufall

Drehbuch:

Ausde M. Alltag
J244

Pleiten, Pech und PatternTesting

wurde ihnen präsentiert von

Chris & Oli



weitere Infos:



oliver.boehm@agentes.de

christian.heise@agentes.de

http://oli.blogger.de

http://patterntesting.org