

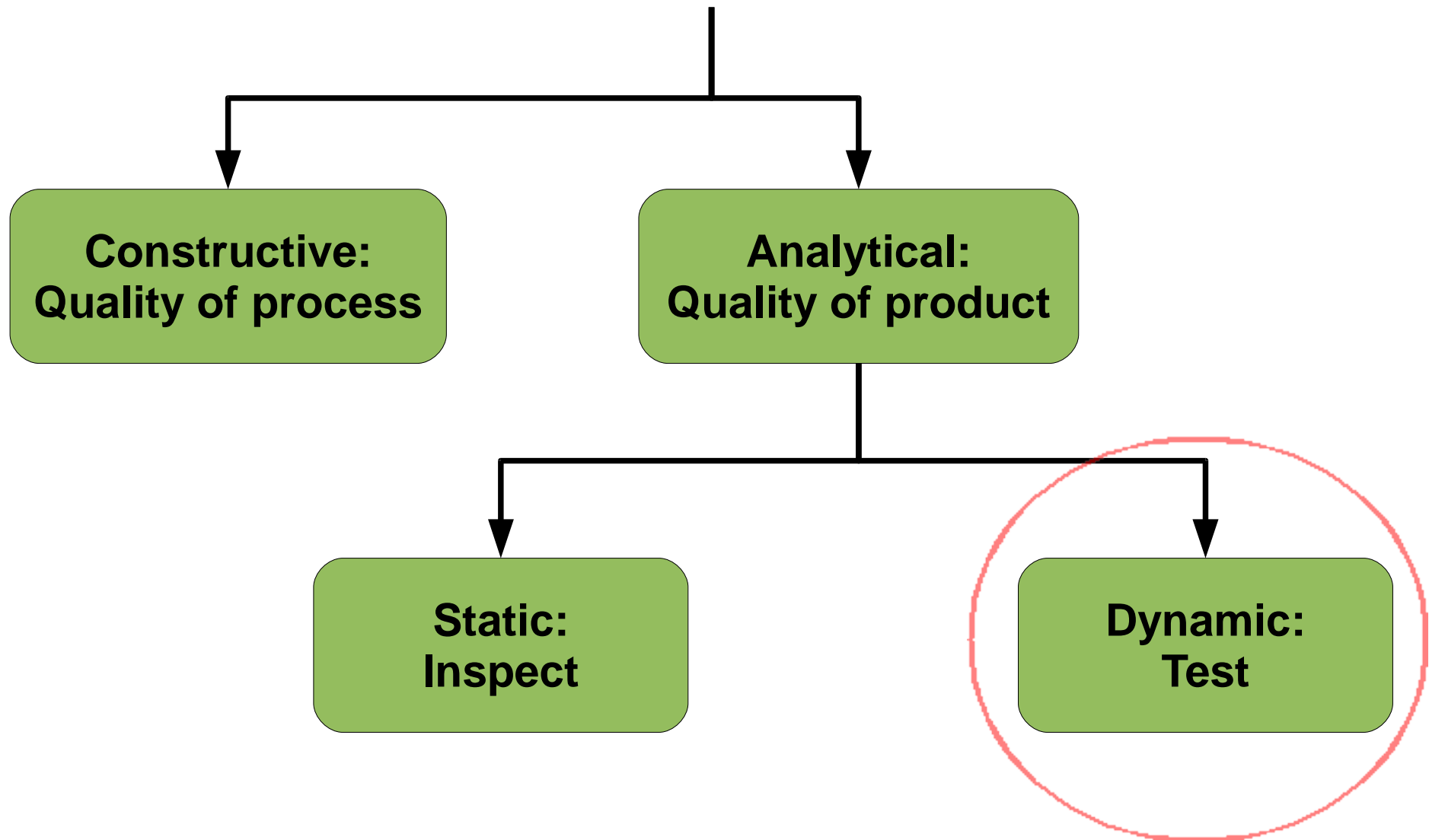


2009

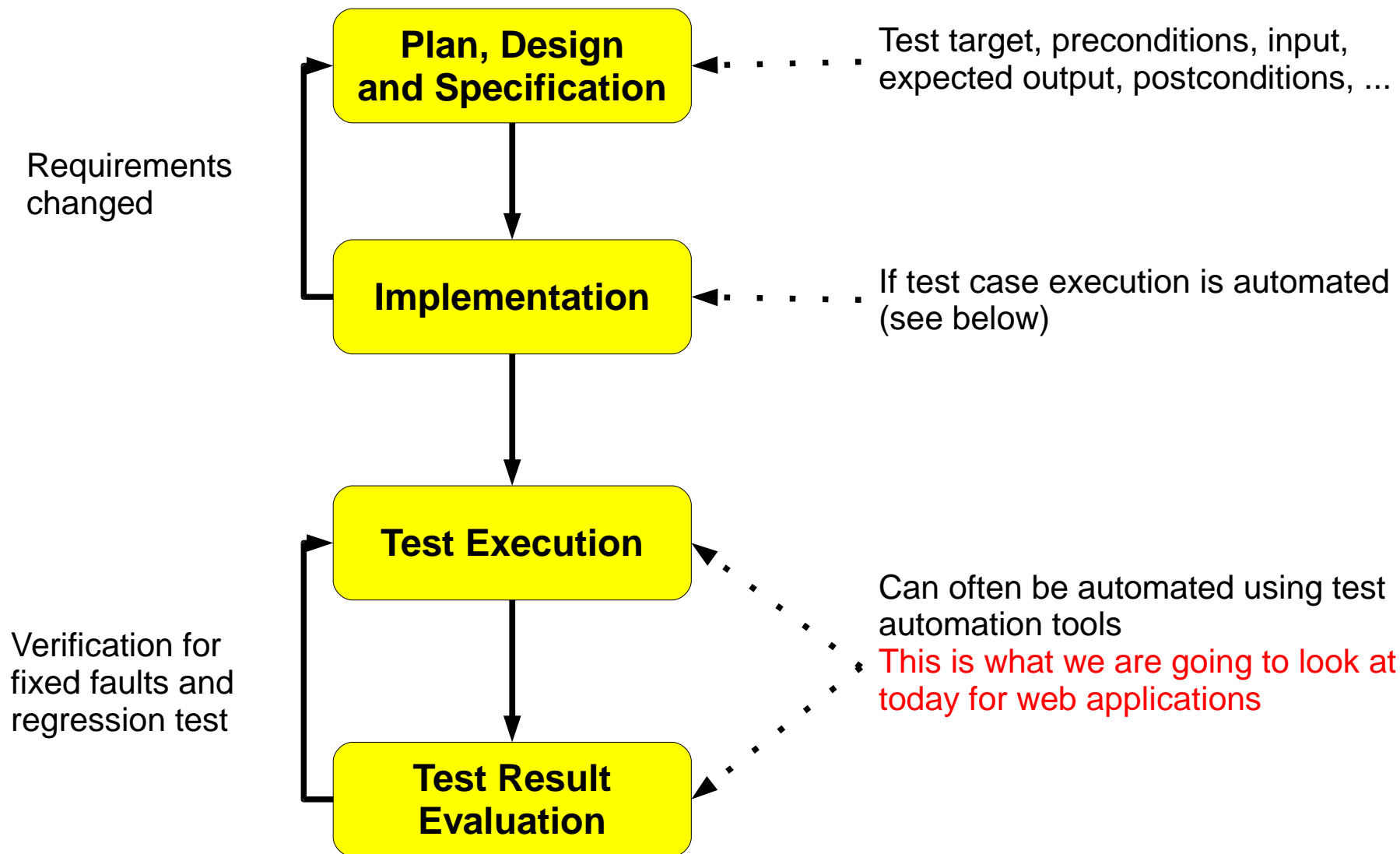
# An introduction to test automation for web applications

Arne-Michael Törsel

# Software quality assurance



# Lifecycle of a testcase

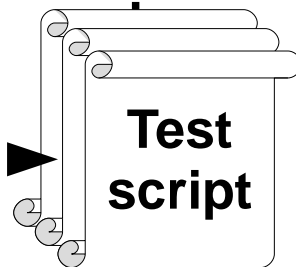
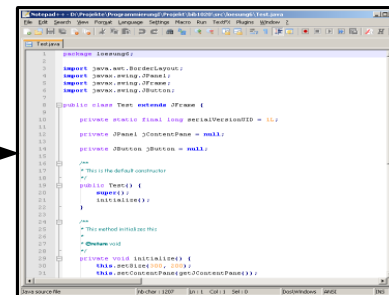


# Test Automation Principles

**Simulate user interaction and verify implementation behaviour  
(requires observable output)**

- **Steered by test case programs: *test scripts***
- **Test cases are grouped into *test suites***
- **Requires known “clean” pre-execution state for either all test cases or the first test case (when test cases build on each other)**

# Selenium: Web Application Test Automation Tool



Record with Selenium IDE in Firefox

Use supported language as Java, Ruby, PHP, ... to build and modify test scripts

Replay and test with Selenium IDE in Firefox

Use Selenium Remote Control to test in Firefox, Internet Explorer and Safari



Open Source project – <http://seleniumhq.org>

# Selenium Command Types

- **Simulate user input**  
Fill input fields, click buttons, ...
- **Check for presence of HTML-Elements and text fragments**  
Used to verify expected application behaviour
- **Extract and store values from the user interface for later use**  
Stored in variables and accessed using `${var_name}` syntax

**Commands are available as API (“client driver”) for:  
Java/JUnit, Java/TestNG, Python, PHP, Ruby, ...**

# Live Demonstration – Selenium

- 1) Recording test cases with Selenium IDE
- 2) Executing recorded test cases with Selenium IDE
- 3) Export recorded test cases to the Java programming language

Tests are done on the  
JForum web forum system

Forum Index -> TODO / New Features

**New Topic**

Subject:

Message body

Emoticons

Options

List topic viewers

User	Views	Date
juhani	42	28/01/2005 16:13:41

Go to Page: 1, 2, 3, 4, 5, 6 Next



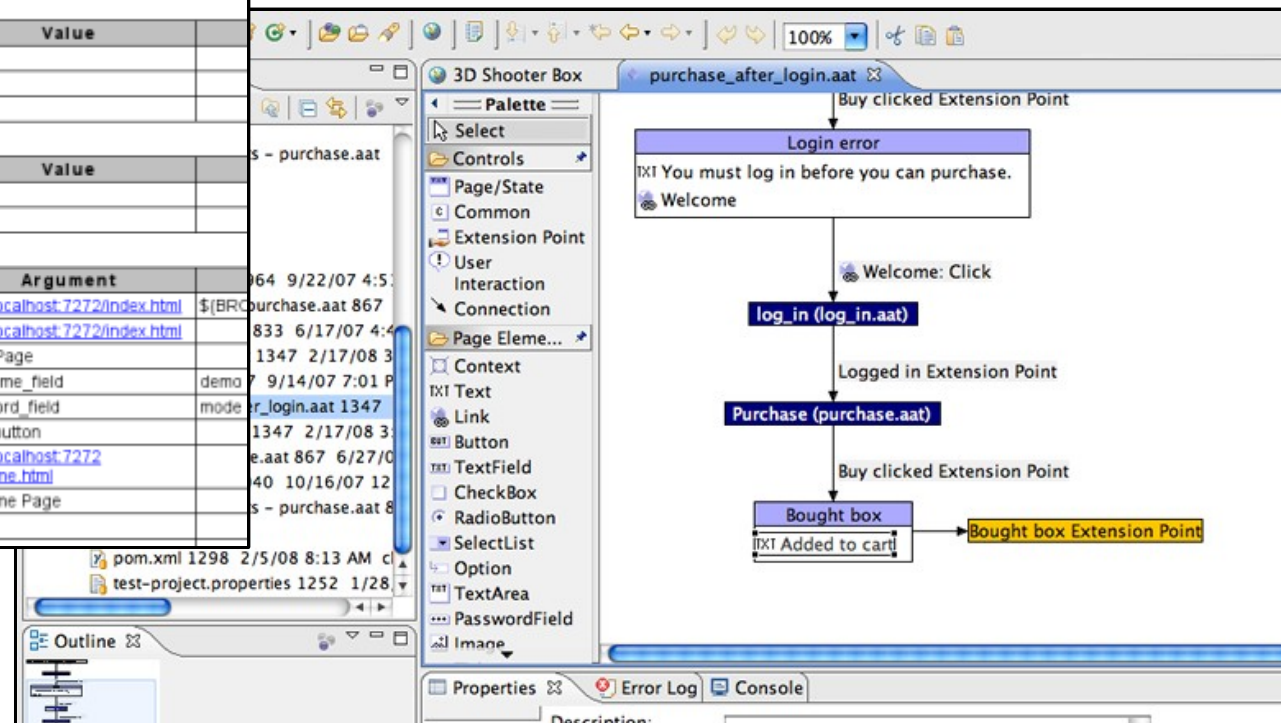
# Abstraction layers for Selenium

Lower maintenance effort by encapsulation of test script code in reusable, “parameterizable” building blocks

Robot Framework + Selenium Library  
(Open Source on Google Code)

Simple Login			
Setting	Value	Value	
Force Tags	regression	smoke	
Library	SeleniumLibrary		
Variable	Value	Value	
\$(BROWSER)	Firefox		
Test Case	Action	Argument	
Valid Login	Open Browser	<a href="http://localhost:7272/index.html">http://localhost:7272/index.html</a>	\$(BROU
	Location Should Be	<a href="http://localhost:7272/index.html">http://localhost:7272/index.html</a>	833 6/17/07 4:4
	Title Should Be	Login Page	1347 2/17/08 3
	Input Text	username_field	demo 7 9/14/07 7:01 P
	Input Text	password_field	mode_r_login.aat 1347
	Click Button	login_button	1347 2/17/08 3
	Location Should Be	<a href="http://localhost:7272/welcome.html">http://localhost:7272/welcome.html</a>	e.aat 867 6/27/0
	Title Should Be	Welcome Page	40 10/16/07 12
	Close Browser		s - purchase.aat 8

CubicTest Eclipse plugin (Selenium subproject)



<http://cubictest.seleniumhq.org/>  
<http://robotframework.org/>



## Another tool: Canoo WebTest

```

<webtest name="register">
  <invoke url="http://www.fh-stralsund.de/jforum"/>
  <clickLink href="user/insert"/>

  <verifyText text="Registration agreement terms"/>
  <clickButton name="submit" />

  <verifyText text="Registration Information" />
  <setInputField name="username" value="john" />
  <setInputField name="email" value="jdoe@email.com" />
  <setInputField name="password" value="abc123" />
  <setInputField name="password_confirm" value="abc123" />
  <clickButton name="submit" />





  <verifyText text="Registration Complete" />
  <clickLink href="user/edit"/>
  <verifyText text="General Information about yourself" />
  <verifyText text="john" />
  <verifyText text="jdoe@email.com" />
</webtest>

```

Example test script

Tool download: <http://webtest.canoo.com>

- XML-based test scripts
- “Headless” execution – no browser required
- Open Source project

webtest				WebTest Test Report Tests started at Thu Sep 06 15:48:04 CEST 2007			
Result Summary				Server Roundtrip Timing Profile			
WebTests	#	%	Graph	Secs	#	%	Histogram
✓	1	100		0 - 1	1	50	
✗	0	0		1 - 3	1	50	
<b>Sum</b>	<b>1</b>	<b>100</b>		3 - 5	0	0	
Steps	#	%	Graph				
✓	5	100		5 - 10	0	0	
✗	0	0		> 30	0	0	
○	0	0		<b>Sum</b>	<b>2</b>	<b>100</b>	
<b>Sum</b>	<b>5</b>	<b>100</b>		<b>Avg</b>		<b>711 ms</b>	

# Summary: Pros and Cons of test automation

## Pro

- Test execution is cheap and quick, enables frequent regression testing
- Enables large test suites
- Quick feedback for developers
- Explicit documentation of test cases

## Contra

- Maintenance effort
- “Path through minefield” analogy – no new errors can be found
- Dependency of test cases may block execution
- Lower fault sensitivity than human tester (“common sense”)