Automatisiertes Testen von Web-Applikationen

mit dem Open Source Tool WebTest

Marc Guillemot
Independent Consultant
Germany
mg@internetzky.de

WebTest & HtmlUnit lead developer

Dierk König
Canoo Engineering AG
Switzerland
dierk.koenig@canoo.com

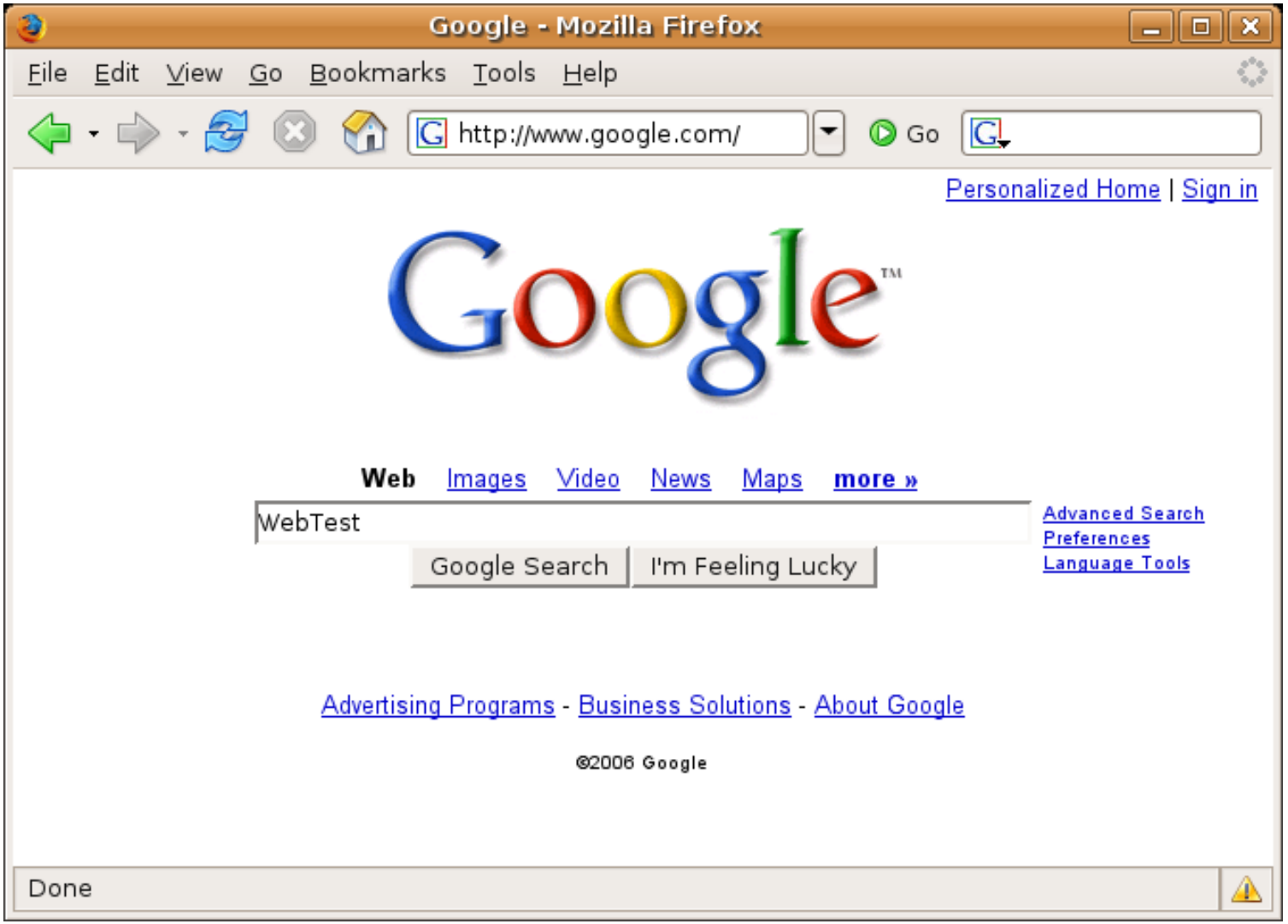
WebTest founder

Free open source tool for automated testing of web applications.

What is WebTest?

- automated Web functional testing tool
- open source (Apache license)
- founded in 2001
- currently 4 active committers (Switzerland, Germany, Australia)
- used in banks, assurance, institutions, solution providers, ...
- adapted to a wide range of projects: from small to huge

Testing Google



WebTest Starter

google.xml

```
<project name="example" default="test">
  <target name="test">

    <webtest name="check that WebTest is Google's top 'WebTest' result">
      <steps>
        <invoke url="http://www.google.com"/>
        <verifyTitle text="Google"/>
        <setInputField name="q" value="WebTest"/>
        <clickButton label="I'm Feeling Lucky"/>
        <verifyTitle text="Canoo WebTest Homepage"/>
      </steps>
    </webtest>

  </target>
</project>
```

```
# runWebtest google.xml
```

canoo

AllTests Tests started at Mon Oct 23 18:49:48 CEST 2006

Result Summary

Result	#	%	Graph
✓	5	100	
✗	0	0	
▫	0	0	
Sum	**5**	**100**	

Server Roundtrip Timing Profile

Secs	#	%	Histogram
0 - 1	1	50	
1 - 5	1	50	
5 - 10	0	0	
10 - 30	0	0	
> 30	0	0	
Sum	**2**	**100**	
Avg		**1**	

Test Scenario Overview (1332)

No	Result	Name	# Steps	Timing profile			Failing step
				Duration	%	Graph	
1	✓	check that WebTest is Google's top 'WebTest' result	5 / 5	1332	100		

✓ check that WebTest is Google's top 'WebTest' result

Test started at Mon Oct 23 18:49:48 CEST 2006, lasting 1332 ms.
Source: `/home/marc/fmr/OOPSLA2006/build.xml:4:`
Base URL (used by **invoke** steps with a relative URL): http://localhost/

No	Result	Name	Parameter	Duration
1	✓	**invoke** Resulting page	**method** GET **url** http://www.google.com	1138
2	✓	**verifyTitle**	**text** Google	8
3	✓	**setInputField**	**name** q **value** WebTest	3
4	✓	**clickButton** Resulting page	**label** I'm Feeling Lucky	183
5	✓	**verifyTitle**	**text** Canoo WebTest Homepage	0

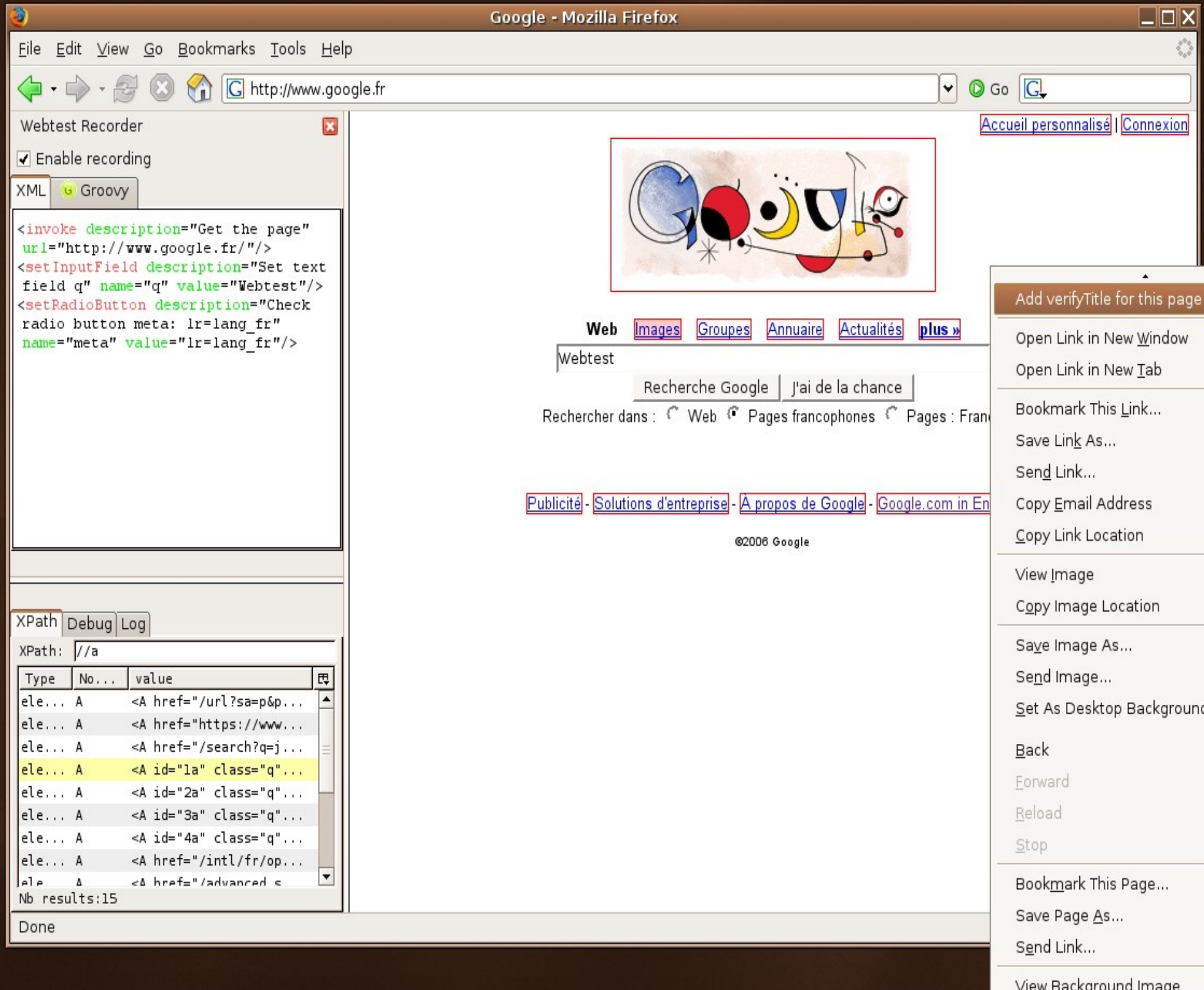
Back to Test Report Overview

WebTest results (zoomed)

✓ **check that WebTest is Google's top 'WebTest' result**

Test started at Mon Oct 23 18:49:48 CEST 2006, lasting 1332 ms.
Source: `/home/marc/fmr/OOPSLA2006/build.xml:4:`
Base URL (used by **invoke** steps with a relative URL): http://localhost/

No	Result	Name	
1	✓	**invoke** Resulting page	**method** GET **url** http://www.google.com
2	✓	**verifyTitle**	**text** Google
3	✓	**setInputField**	**name** q **value** WebTest
4	✓	**clickButton** Resulting page	**label** I'm Feeling Lucky
5	✓	**verifyTitle**	**text** Canoo WebTest Homepage


Google - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.google.fr
Go
Webtest Recorder
Enable recording
XML
Groovy
<invoke description="Get the page" url="http://www.google.fr/"/>
<setInputField description="Set text field q" name="q" value="Webtest"/>
<setRadioButton description="Check radio button meta: lr=lang_fr" name="meta" value="lr=lang_fr"/>
XPath
Debug
Log
XPath: //a
Type No... value
ele... A <A href="/url?sa=p&p...
ele... A <A href="https://www...
ele... A <A href="/search?q=j...
ele... A <A id="1a" class="q"...
ele... A <A id="2a" class="q"...
ele... A <A id="3a" class="q"...
ele... A <A id="4a" class="q"...
ele... A <A href="/intl/fr/op...
Nb results:15
Done
Accueil personnalisé | Connexion
Web Images Groupes Annuaire Actualités plus »
Webtest
Recherche Google J'ai de la chance
Rechercher dans : Web Pages francophones Pages : Franc
Publicité - Solutions d'entreprise - A propos de Google - Google.com in En
©2006 Google
Add verifyTitle for this page: "Google"
Open Link in New Window
Open Link in New Tab
Bookmark This Link...
Save Link As...
Send Link...
Copy Email Address
Copy Link Location
View Image
Copy Image Location
Save Image As...
Send Image...
Set As Desktop Background...
Back
Forward
Reload
Stop
Bookmark This Page...
Save Page As...
Send Link...
View Background Image

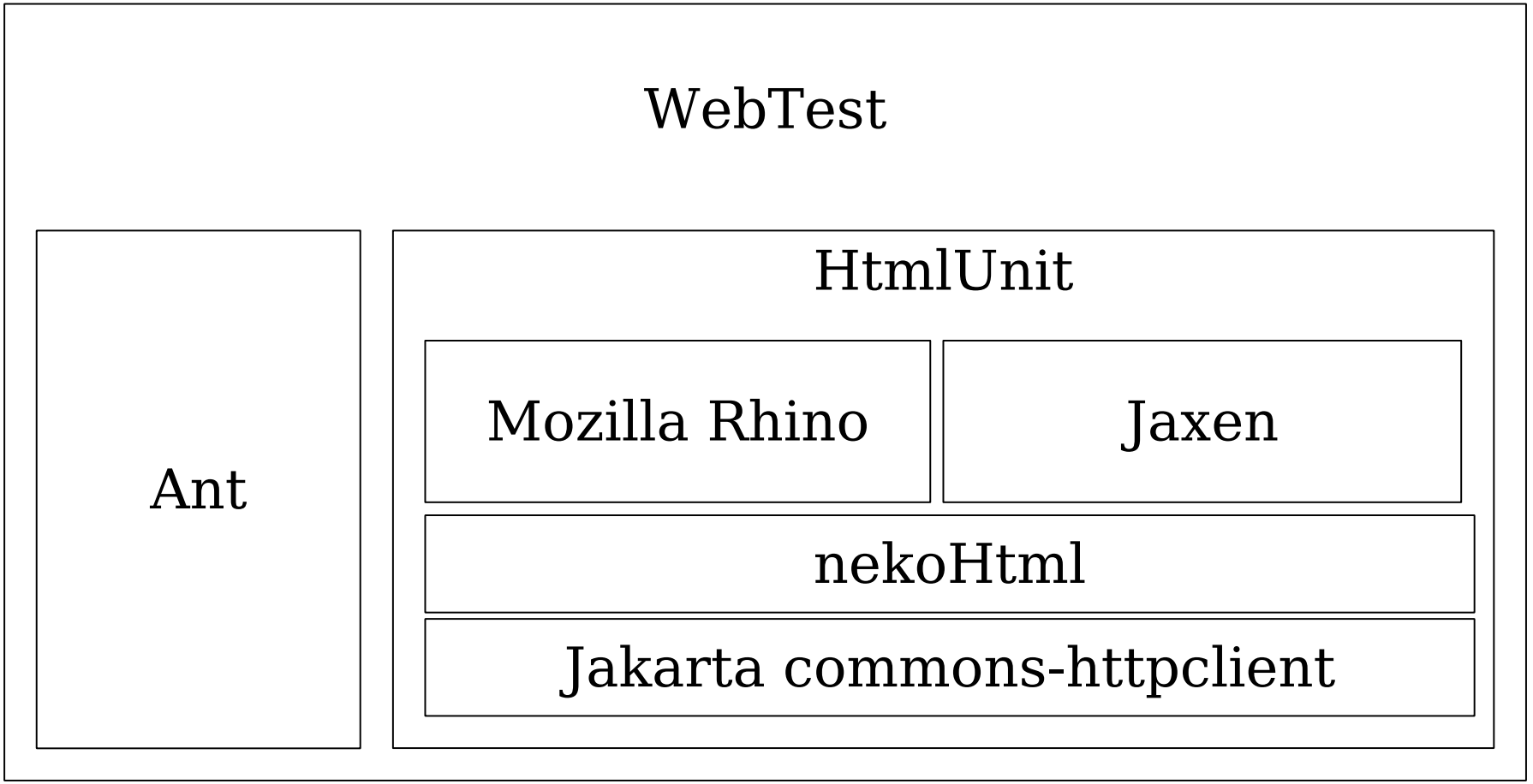
Over 100 WebTest Steps

- General:
  - <invoke .../>
  - <clickLink.../>
  - ...
- Forms
  - <setInputField.../>
  - <setRadioButton.../>
  - <setCheckBox.../>
  - <clickButton.../>
  - ...
- Verification
  - <verifyTitle.../>
  - <verifyXPath.../>
  - <verifyInputField>
  - ...
- PDF
  - <pdfDecryptDocument.../>
  - <pdfVerifyField.../>
  - <pdfVerifyText.../>
  - ...
- Excel Documents
  - <excelFindRow.../>
  - <excelVerifyCellValue.../>
  - ...
- ...

Comprehensive online documentation:

http://webtest.canoo.com/

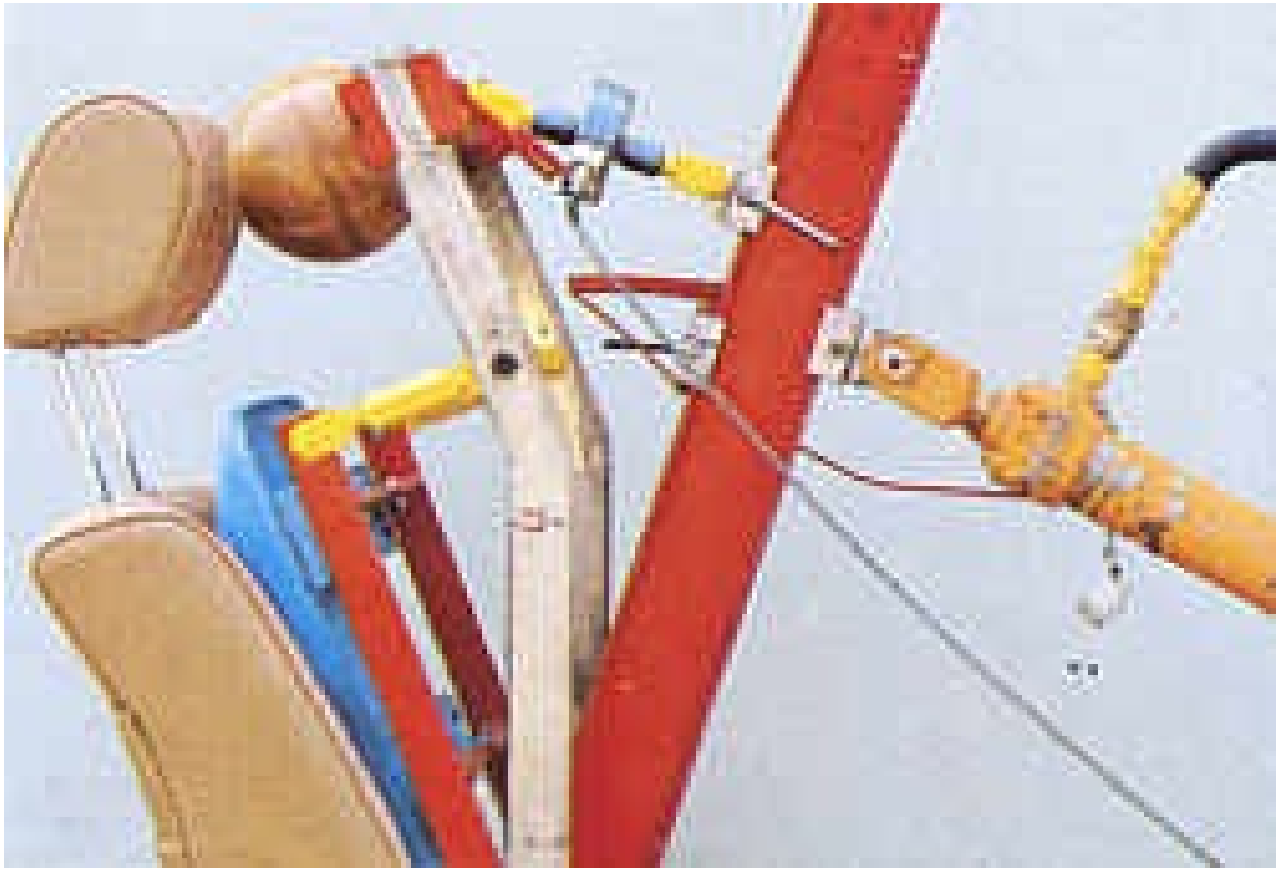
Building blocks



The 4 testing paradigms

- Capture / replay
  - "the least cost-effective way of test automation"
  - Use capturing wisely
- Model based testing
  - Specification: modelling expected behaviour
  - Fault model: modelling unexpected behaviour
  - Be tolerant to accidental behaviour
- Data driven testing
  - Narrow scope for data variations on same workflow
- Scripted automation
  - Maximum flexibility and power
  - Maximum responsibilty

A software engineering activity



Lessons learned

- use recorder only to give you a jump start
- specify expectations
- distinguish guaranteed from accidental behaviour
- apply rules of software engineering
  - remove duplications, extract modules
  - use source control
- consider modifications and extensions
  - Ant, Java, Groovy
- design for testability
  - valid html code, test early

Determine what you want to test

How to test addition into basket:

- <clickLink label=”Einkaufskorb”/> ?
- <clickLink label=”Ajax”/> ?
- <clickLink xpath=”//*[text() = 'Top Angebot']//a”/> ?
- <clickLink xpath=”//*[text() = '6.60']/following-sibling::a]”/> ?

=> it depends: these 4 examples don't test the same thing!

Sustainable Tests

- exploit XPath wisely
  - good: //*[@id='total']

    bad: /html/body/div[2]/table[3]/tbody/tr[6]/td[4]
- testable HTML code
- structured tests
  - property files
  - XML entities
  - Ant macros

Wisely use properties

```
<project name="example" default="test">
  <target name="test">

    <webtest name="check that WebTest is Google's
     top 'WebTest' result">
      <steps>
        <invoke url="${startUrl}"/>
        <verifyTitle text="Google"/>
        <setInputField name="q" value="WebTest"/>
        <clickButton label="${luckyButton}"/>
        <verifyTitle text="Canoo WebTest
     Homepage"/>
      </steps>
    </webtest>

  </target>
</project>
```

de.properties

```
startUrl=http://www.google.de/de
luckyButton=Auf gut Glück!
```

fr.properties

```
startUrl=http://www.google.fr/fr
luckyButton=J'ai de la chance
```

en.properties

```
startUrl=http://www.google.com/ncr
luckyButton=I'm Feeling Lucky
```

```
...
<ant antfile="google.xml">
  <property file="de.properties">
</ant>
....
```

Use entities for common blocks

myTest.xml

```
<!DOCTYPE project SYSTEM ”myDtd.dtd”>
<project>
      ....
     <steps>
       &goToLoginPage;
       ...
    </steps>
...
```

myDtd.dtd

```
<!ENTITY goToLoginPage SYSTEM
    "includes/goToLoginPage.xml">
```

goToLoginPage.xml

```
<group description=”go to login”>
  <invoke url=”http://mysite.com”/>
  <verifyTitle text=”My great Web Site”/>
  <clickLink label=”login”/>
  <verifyText text=”Restricted area”/>
</group>
```

... or macros

```
...
<macrodef name="doLogin”/>
  <attribute name="login”/>
  <attribute name="password”/>
  <sequential>
    &goToLoginPage;
    <setInputField forLabel=”Login” value=”@{login}”/>
    <setInputField forLabel=”Password” value=”@{password}”/>
    <verifyText text=”Hello .*\. Welcome to the restricted area” regex=”true”/>
  <sequential>
</macrodef>
...
```

myTest.xml

```
<project>
  ....
  <steps>
    <doLogin login=“john” password=“john”/>
    ...
  </steps>
...
```

Fine control (1)

```
...
    <steps>
     <groovy description="configure HTTP 1.0 as default protocol version">
          import org.apache.commons.httpclient.*
          import org.apache.commons.httpclient.params.*

          DefaultHttpParams.defaultParams.version = HttpVersion.HTTP_1_0
     </groovy>
     <invoke url="http://myHost"/>
     .....
    </steps>
...
```

Fine control (2)

First name	Last name	Location	Since
Dierk	König	Switzerland	2001
Denis	Antonioli	Switzerland	2002
Marc	Guillemot	Germany	2003
Paul	King	Australia	2004

```
...
<steps>
 .....
 <groovy description="test table sorted by last name">
import com.canoo.webtest.engine.StepFailedException as SFE

def table = step.context.currentResponse.getHtmlElementById('theTable')
def tds = table.getByXPath('tbody/tr/td[2]')
def texts = tds*.asText()
def sorted = new ArrayList(texts).sort()
if (sorted != texts)
   throw new SFE("Not correctly sorted", "
 </groovy>
 .....
   </steps>
...
```

Error

Message

Not correctly sorted

Location

(line: 0)

Details

expected value `["Antonioli", "Guillemot", "King", "König"]`
actual value `["König", "Antonioli", "Guillemot", "King"]`

Simply extend WebTest

```
<groovyScript>
 class MyExtension extends com.canoo.webtest.steps.Steps
 {
        String myProp
        void doExecute()
        {
              // do something
        }
  }
  project.taskDefinitions['myNewStep'] = MyExtension
 </groovyScript>
  ....
 <steps>
    ...
   <myNewStep myProp="whatever"/>
    ...
</steps>
```

AJAX

```
...
<webtest name="Test DWR 2.0 RC3 demo App">
  <steps>
    <invoke url="http://localhost:8080/dwr/simpletext"/>
    <verifyXPath xpath="//*[@id= 'demoReply']" text=""/>
    <clickButton label="Send"/>
    <sleep seconds="2"
description="Wait for completion of async call"/>
    <verifyXPath xpath="//*[@id= 'demoReply']"
            text="Hello, Joe"/>
  </steps>
</webtest>
...
```

#	Result	Name	Parameter
1	✓	**invoke** Resulting page	**url** http://localhost:8080/dwr/simpletext
2	✓	**verifyXPath**	**text** **xpath** //*[@id= 'demoReply']
3	✓	**clickButton**	**label** Send
4	✓	**sleep** Wait for completion of async call	**seconds** 2
5	✓	**verifyXPath**	**text** Hello, Joe **xpath** //*[@id= 'demoReply']

(currently) no dedicated support

- need to add <sleep.../>
- nothing to “see” in report

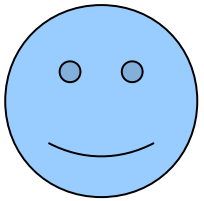
Groovy WebTest

```
ant.webtest(name: 'Test Google with Groovy, AntBuilder and WebTest')
{
   steps()
   {
      invoke(url: 'http://www.google.com')
      verifyTitle(text: 'Google')
      setInputField(name: 'q', value: 'Groovy')
      clickButton(name: 'btnG')
      verifyXPath(xpath: "//a[@href='http://groovy.codehaus.org/']")
   }
}
```

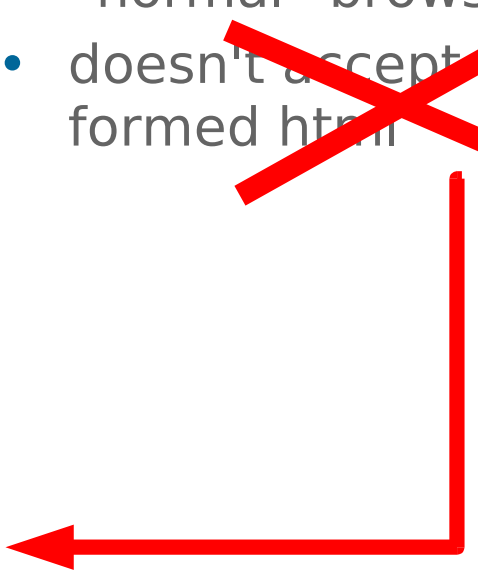
Grails automagically generates WebTests for the generated CRUD operations

Key properties

- simple
- fast
- excellent reporting
- very low TCO
- runs everywhere
- no display needed
- easy to extend
- straightforward integration
- doesn't accept (too) badly formed html



- js support not as good as “normal” browser
- doesn't accept (too) badly formed html



WebTest (possible) future(s)

- dedicated AJAX support (coming)
- WebTestClipse
- (even) better reports
- load testing
- ...

Free open source tool for automated testing of web applications.

Happy Testing!

http://webtest.canoo.com/

Marc Guillemot & Dierk König
JAX 07, 26.04.2007

webtest

Bonus

User testimonials

(from WebTest mailing list)

- the ROI on WebTest is many orders of magnitude higher than any tool I've used
- Support = good and $0
- It still Just Works. It has been remarkably robust, mature, and bug-free
- It has scaled well
- tests are much quicker to write and cheap to maintain
- Excellent reporting. Forget all the other reasons!

Customize XPath

```
<groovyScript>
import org.jaxen.*
import org.jaxen.function.*
import com.canoo.webtest.engine.xpath.XPathHelper

class ReverseFunction implements Function {
    Object call(Context context, List _args) {
        def input = StringFunction.evaluate(_args[0], context.navigator);
        return input.reverse()
}   }
XPathHelper.registerGlobalFunction("http://webtest.canoo.com",
                                   "reverse", new ReverseFunction())
</groovyScript>
...
<verifyXPath xpath="wt:reverse('food')" text="doof"/>
```

Customize WebTest

Home > Functional testing > WebTest

```
...
<verifyXPath xpath="//*[@id='navPath1']” text="Home”/>
<verifyXPath xpath="//*[@id='navPath2']” text="Functional testing”/>
<verifyXPath xpath="//*[@id='navPath3']” text="WebTest”/>
...
```

Works but:

- too low level
- not reusable

Customize WebTest

Home > Functional testing > WebTest

```
...
<macrodef name="verifyNavPath”/>
  <attribute name="level1”/>
  <attribute name="level2”/>
  <attribute name="level3”/>
  <sequential>
    <verifyXPath xpath="//*[@id='navPath1']” text="@{level1}”/>
    <verifyXPath xpath="//*[@id='navPath2']” text="@{level2}”/>
    <verifyXPath xpath="//*[@id='navPath3']” text="@{level3}”/>
</sequential>
</macrodef>
...
<verifyNavPath level1="Home” level2="Functional testing”
     level3="WebTest”/>
...
```

Customize WebTest

```
<groovyScript name="verifyNavPath”><![CDATA[
  class VerifyNavigationPath extends com.canoo.webtest.steps.Step {
    String level1, level2, level3, level4
    void doExecute() {
      def ant = new AntBuilder(project)
      def levels = [0, level1, level2, level3, level4]
      for (i in 1..<levels.size()) {
        if (levels[i])
          ant.verifyXPath(xpath: "//a[@id='navPath${i}']/text()", text: levels[i],
              description: "Verify level ${i}")
  } } }
  project.addTaskDefinition('verifyNavPath', VerifyNavigationPath)
]]></groovyScript>
...
<verifyNavPath level1="Home”/>
<verifyNavPath level1="Home” level2="Functional testing”
     level3="WebTest”/>
```

webtest