Distributed Testing with SmartFrog

Julio Guijarro
Steve Loughran

HP Laboratories, Bristol, UK

julio.guijarro at hpl.hp.com
steve.loughran at hpl.hp.com

2006-09-07
About Us

**Julio Guijarro**
Research scientist at HP Laboratories on
*Grid-Scale Deployment*

*Leads the SmartFrog open source effort*

**Steve Loughran**
Research scientist at HP Laboratories on
*Grid-Scale Deployment*
Apache Ant committer
Co-author of
*Java Development with Ant*
2nd edition due by Xmas (as is Ant1.7)
Two different distributed systems

CERN Large Hadron Collider

Multi-tier webapp
How do you show it works?

• Europe's high-end server farms
• Years of simulations
• Nobel Prize winners, Computer Scientists and physics PhD students

• An old laptop nobody wants
• Any spare time before the go-live date
• You
Any application without adequate system tests doesn’t exist
Challenges in System Testing

0. Developer education & motivation
1. Writing the tests
2. Running the tests
3. Analysing the results

If we can make system testing compelling, the tests will follow
Writing the tests

No single test framework will suffice
Unit-test derived test frameworks

- JUnit 3+extensions
- JUnit 4
- TestNG
- Apache Cactus
- HttpUnit
- Selenium
- jsUnit
- ...

Axis Happiness Page

Examining webapp configuration

Needed Components

- Axis 2.8
- ApacheJakarta-tomcat-5.5.20
- ApacheJakarta-tomcat-5.5.20-api
- ApacheJakarta-tomcat-5.5.20-osgi
- ApacheJakarta-tomcat-5.5.20-servlet
- ApacheJakarta-tomcat-5.5.20-util
- ApacheJakarta-tomcat-5.5.20-webapp
- commons-discovery (Apache Commons Discovery)
- commons-logging (Apache Commons Logging)
- commons-logging-api (Apache Commons Logging API)
- commons-logging-impl (Apache Commons Logging Implementations)
- commons-logging-api (Apache Commons Logging API)
- commons-logging-impl (Apache Commons Logging Implementations)
- commons-logging-api (Apache Commons Logging API)
- commons-logging-impl (Apache Commons Logging Implementations)

Optional Components

- Found Mail API (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)
- Found JavaMail (JavaMail)

<Cactus>/IDE

Application/Web Server

CactusServlet

CactusTest extends ServletTestCase

setUp testXXX tearDown

NormalTest extends TestCase

Server-side classes
Running the tests
Today

- IDE/Ant/Maven hosted test runners
- Continuous Integration servers
- JUnit is primary test platform
- Bridges to other processes (Cactus, Selenium Server)
- Local system
- Local network

*System testing is tacked on to unit testing*
• Applications & tests run on multiple different hosts
• Hosts (real and virtual) are dynamically allocated
• Host and networks configured as part of the test
• Reports include output from all hosts and network
• Result presentation scales up.
• Data mining of multiple tests runs
• Runs as a background task under the CI server.
Transforming how mainstream developers deploy applications

- HPLabs research
- Java Based distributed deployment system
- LGPL licensed
- http://smartfrog.org/
Configuration is Deployment

System {
    appServer { ... }
    database { ... }
    logging { ... }
}

appServer extends Jetty {
    db LAZY database;
    log LAZY logging;
}

database extends MySql {
    log LAZY logging;
    users [
        ["web",""]
        ["admin","secReT"]
    ];
}

Logging extends Log {
    level=3;
    dir="/logs";
}
SmartFrog for testing

– set up application with different options
– preconfigure the database
– bring up Xen/VMware images for hosting
– deploy the tests and collect the results
– clean up

Tests are just something else to deploy
HttpUnitTests extends JUnitTestSuite {
    package "d1.webapp.test";
    name "HttpUnitTests";
    server.url TBD;
    sfProcessHost "client";
    properties [
        ["server.url", server.url],
    ];
    classes [
        "EventFeedTest",
        "HappyTagTest",
        "IndexTest"
    ];
}
Test methodology

- Allocate & configure test systems
- Deploy application across nodes
- Deploy tests on other nodes
- Collect and correlate results
- Try to understand what went wrong
## Test scenarios

<table>
<thead>
<tr>
<th>Test</th>
<th>Single Node</th>
<th>Multiple Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Traditional testing</td>
<td>client/server</td>
</tr>
<tr>
<td>Single Node</td>
<td>Write once, test everywhere -client side apps</td>
<td>fully distributed application and tests</td>
</tr>
</tbody>
</table>
Demo: “Enterprise” Web Server

- SmartFrog daemon
- Database VM: MySQL, Test Database
- App Server VM: Web App, App Server
- Test Master: Test Runner, Test Listener
- Client: HttpUnit, JUnit, Cactus, JUnit

Diagram showing the integration of various components in an enterprise web server environment.
Demo: Interop testing of SOAP Systems

- SmartFrog daemon
How we test SmartFrog itself

One ring of SmartFrog daemons deploys a new set

Security: x x ✓ x ✓ ✓
Data Mining of Results

Please, solve this
Ant's HTML report of JUnit is ubiquitous.
But the underlying XML is limited

summary at the top
<testsuite errors="0" failures="0" hostname="Zermatt"
name="api.Api_05_system_create_destroy_Test" tests="1" time="2.623"
timestamp="2006-06-29T11:09:48">
<properties>
  <property name="epr.hp"
    value="http://deployapi.iseran.com/alpine/portal/" />
</properties>
<testcase classname="api.Api_05_system_create_destroy_Test"
name="testCreateCreatedTimeDestroySystem" time=" 2.623"/>
<system-out><![CDATA[
0    DEBUG org.smartfrog.projects.alpine.transport.http.HttpTransmitter -
     Posting to http://deployapi.iseran.com:8080/alpine/portal/
141  DEBUG httpclient.wire.header - >> "POST /alpine/portal/
                        HTTP/1.1[
                        ]"
151  DEBUG httpclient.wire.header - >> "SOAPAction: [\r][\n"
151  DEBUG httpclient.wire.header - >> "User-Agent: Alpine [\r][\n"
]]></system-out>
</testsuite>

Minimal context

single-host log loses information
We need a unified format of test results

- real-time and post-run display/analysis
- extensible metadata (dublin core, context)
- integrate logs from multiple machines
- streamable to disk/over the network
- cross-language stack traces

A new XML format, or XHTML?
SmartFrog XHTML output

~live output log capture
• Swing GUI for testing on OurGrid
• Aggregate view of results
• “partial” success

http://gridunit.sourceforge.net/
Call to Action

• Focus on system tests
• Automate deployment
• Automate system testing

• Work on an XML/XHTML test result format
• Test framework authors: help us run your frameworks.

If you have big deploy/test problems, come join the SmartFrog community
Any W3C/OASIS/WSI/GGF/IETF standard without tests doesn’t exist