

Automation of itests in web applications

Automation of itests in web applications

In our WAR packaging scheme, we package SCA artifacts with Tuscany runtime jars in a WAR so that it can be deployed to a web container such as Tomcat, Jetty or Geronimo. In release 1.1, we already have 5-6 samples as web applications.

- 1) Most of the validation of web applications is done manually before a release. This is time-consuming and error prone.
- 2) Our integration tests are not performed in the WAR packaging scheme. Trying samples is a smoke test. If we can run the itests in a web application, we'll get much more coverage.

To address the above issues, I propose that we add the following capabilities:

- 1) Provide an option to wrap itests into web applications with little effort to convert the test cases
- 2) Automate the steps to start/stop a web container and deploy web applications to the container
- 3) Automate the testing of our JUnit-based test cases in the web application

I did some prototyping and here is what I have achieved so far:

- 1) Add some code to "org.apache.tuscany.sca.host.webapp.TuscanyServletFilter" (tuscany-host-webapp) to intercept the HTTP requests to a special path "/junit". The control is then delegated to a new class WebTestRunner that knows how to find and run the JUNIT test cases and produce the results of the test (for example, number of runs, failures and errors, similar with what maven surefire plugin does). The report is sent back to the web client.

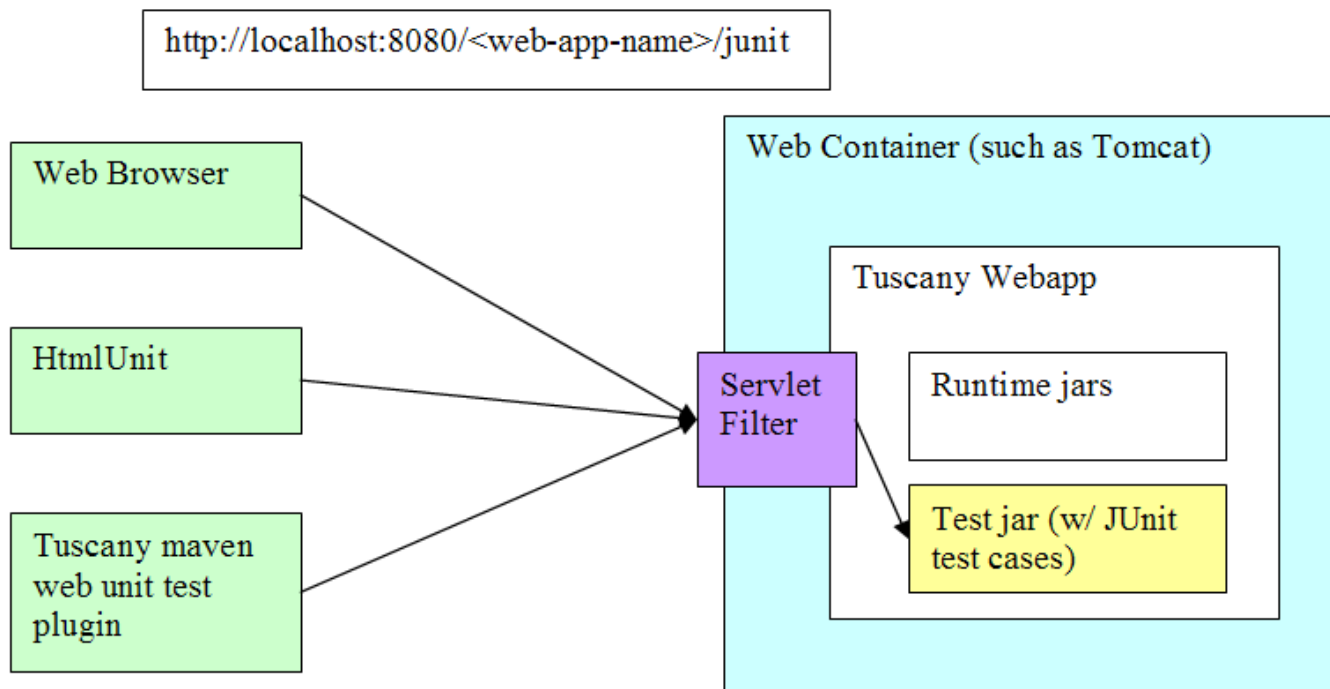
With this, we can run the plain JUNIT test cases in the web application by connecting to a URL like <http://localhost:8080/<app-name>/junit>.

- 2) Adjust the sample-calculator-webapp to validate the idea

- Add a test case calculator.CalculatorTestCase
- Configure the maven-war-plugin in the pom.xml to include the test classes as a jar into the WAR (WEB-INF/test-lib/)
- Change the junit dependency from test to runtime so it will be packaged into the WAR

- 3) Run the maven build as usual, produce a WAR, deploy to Tomcat, browse to <http://localhost:8080/sample-calculator-webapp/junit>, and see the test results in the browser. To run a selected list of test cases, <http://localhost:8080/sample-calculator-webapp/junit?test1,test2>.

This seems to be non-invasive as we already configure the servlet filter for web applications. The regular URLs to JSPs or web services stay the same. We don't have to change the test case code too much.



I also went a bit further:

- 1) Developed a maven plugin to trigger the web-based junit tests and analyze the results
- 2) Configure the pom.xml with cargo plugin to start an embedded Jetty server, deploy the WAR, trigger the web-based junit tests, and stop the server.

Now I have a complete automation. To avoid repeating the pom configuration for each itest, we can add a profile in the parent pom to support this scheme.

Appendix

Sample results

```
<?xml version="1.0" encoding="UTF-8" ?>
<report>
  <testsuites>
    <testsuite name="null" tests="1" failures="0" errors="0" time="0.094">
      <testcase name="testTry(calculator.CalculatorTestCase)" time="0" />
    </testsuite>
  </testsuites>
</report>
```

Updated pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * software distributed under the License is distributed on an
 * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
 * KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations
 * under the License.
-->
<project>
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.apache.tuscany.sca</groupId>
    <artifactId>tuscany-sca</artifactId>
    <version>1.2-incubating-SNAPSHOT</version>
    <relativePath>../../pom.xml</relativePath>
  </parent>
  <artifactId>sample-calculator-webapp</artifactId>
  <packaging>war</packaging>
  <name>Apache Tuscany SCA Calculator Sample in a WebApp</name>

  <repositories>
    <repository>
      <id>apache.incubator</id>
      <url>http://people.apache.org/repo/m2-incubating-repository</url>
    </repository>
  </repositories>

  <dependencies>
    <dependency>
      <groupId>org.apache.tuscany.sca</groupId>
      <artifactId>tuscany-host-webapp</artifactId>
      <version>1.2-incubating-SNAPSHOT</version>
    </dependency>

    <dependency>
      <groupId>org.apache.tuscany.sca</groupId>
      <artifactId>tuscany-implementation-java-runtime</artifactId>
      <version>1.2-incubating-SNAPSHOT</version>
    </dependency>
  </dependencies>
</project>
```

```

        <scope>runtime</scope>
    </dependency>

    <!-- marking dependency as provided to exclude from war file -->
    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>servlet-api</artifactId>
        <version>2.3</version>
        <scope>provided</scope>
    </dependency>

    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.2</version>
        <scope>compile</scope>
    </dependency>

</dependencies>

<build>
    <finalName>${artifactId}</finalName>

    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-jar-plugin</artifactId>
            <version>2.2</version>
            <executions>
                <execution>
                    <configuration>
                        <finalName>web</finalName>
                    </configuration>
                    <phase>test</phase>
                    <goals>
                        <goal>test-jar</goal>
                    </goals>
                </execution>
            </executions>
        </plugin>

        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-war-plugin</artifactId>
            <version>2.0.2</version>
            <configuration>
                <webResources>
                    <resource>
                        <!-- this is relative to the pom.xml directory -->
                        <directory>target</directory>
                        <includes>
                            <include>*-tests.jar</include>
                        </includes>
                        <targetPath>WEB-INF/test-lib</targetPath>
                    </resource>
                </webResources>
            </configuration>
        </plugin>

        <plugin>
            <groupId>org.apache.tuscany.sca</groupId>
            <artifactId>tuscany-maven-web-unittest</artifactId>
            <version>1.2-incubating-SNAPSHOT</version>
            <executions>
                <execution>
                    <configuration></configuration>
                    <phase>integration-test</phase>
                    <goals>
                        <goal>test</goal>
                    </goals>
                </execution>
            </executions>
        </plugin>
    </plugins>
</build>

```

```

        </executions>
    </plugin>
    <plugin>
        <groupId>org.apache.tuscany.sca</groupId>
        <artifactId>tuscany-maven-ant-generator</artifactId>
        <version>1.2-incubating-SNAPSHOT</version>
        <executions>
            <execution>
                <goals>
                    <goal>generate</goal>
                </goals>
            </execution>
        </executions>
    </plugin>
</plugins>
</build>

<profiles>
    <profile>
        <id>jetty</id>
        <activation>
            <activeByDefault />
        </activation>
        <build>
            <plugins>
                <plugin>
                    <groupId>org.codehaus.cargo</groupId>
                    <artifactId>cargo-maven2-plugin</artifactId>
                    <executions>
                        <execution>
                            <id>start-container</id>
                            <phase>pre-integration-test</phase>
                            <goals>
                                <goal>start</goal>
                            </goals>
                        </execution>
                        <execution>
                            <id>stop-container</id>
                            <phase>post-integration-test</phase>
                            <goals>
                                <goal>stop</goal>
                            </goals>
                        </execution>
                    </executions>
                    <configuration>
                        <container>
                            <containerId>jetty6x</containerId>
                            <type>embedded</type>
                            <systemProperties>
                                <org.apache.commons.logging.Log>
                                    org.apache.commons.logging.impl.SimpleLog
                                </org.apache.commons.logging.Log>
                            </systemProperties>
                        </container>
                        <wait>false</wait>
                        <configuration>
                            <properties>
                                <cargo.servlet.port>8080</cargo.servlet.port>
                            </properties>
                            <deployables>
                                <deployable>
                                    <location>
                                        ${project.build.directory}/${project.build.finalName}.${project.
packaging}

                                        </location>
                                        <pingURL>http://localhost:8080/${project.build.finalName}/junit<
/pingURL>

                                    </deployable>
                                </deployables>
                            </configuration>
                        </configuration>
                    </configuration>
                </plugin>
            </plugins>
        </build>
    </profile>
</profiles>

```

```

        </plugin>
      </plugins>
    </build>
  </profile>
  <profile>
    <id>geronimo</id>
    <properties>
      <geronimo.home>C:\Apache\geronimo-tomcat6-jee5-2.0.2</geronimo.home>
    </properties>

    <build>
      <plugins>
        <plugin>
          <groupId>org.apache.geronimo.plugins</groupId>
          <artifactId>geronimo-maven-plugin</artifactId>
          <version>2.0.2</version>
          <configuration>
            <geronimoHome>${geronimo.home}</geronimoHome>
          </configuration>
          <executions>
            <execution>
              <id>start-container</id>
              <phase>pre-integration-test</phase>
              <goals>
                <goal>start</goal>
              </goals>
              <configuration>
                <background>true</background>
              </configuration>
            </execution>
            <execution>
              <id>deploy-war</id>
              <phase>pre-integration-test</phase>
              <goals>
                <goal>deploy</goal>
              </goals>
              <configuration>
                <moduleArchive>
                  ${project.build.directory}/${project.build.finalName}.${project.
packaging}

                  </moduleArchive>
                </configuration>
              </execution>
              <execution>
                <id>undeploy-war</id>
                <phase>post-integration-test</phase>
                <goals>
                  <goal>undeploy</goal>
                </goals>
                <configuration>
                  <moduleId>org.apache.tuscany.sca/sample-calculator-webapp/1.2-incubating-
SNAPSHOT/war</moduleId>

                  </configuration>
                </execution>
              <execution>
                <id>stop-container</id>
                <phase>post-integration-test</phase>
                <goals>
                  <goal>stop</goal>
                </goals>
                <configuration>
                  <username>system</username>
                  <password>manager</password>
                </configuration>
              </execution>
            </executions>
          </plugin>
        </plugins>
      </build>
    </profile>

```

```
</profiles>  
</project>
```