Functional testing

Functional Testing with Cargo

This page explains how to do functional testing with the Cargo Java API and JUnit.

Are you using ANT or Maven2/Maven3?

If you are relying on ANT or Maven2/Maven3 for your build, it is probably easier for you to use the associated tasks/plugin directly:

- The documentation for the CARGO ANT tasks can be found here: Ant support.
- The documentation for the CARGO Maven2/Maven3 plugin can be found here: Maven2 plugin.

Note that the Maven2/Maven3 plugin also has archetypes you can directly use as a skeleton for integration.

Example

The best is to create a JUnit `TestSetup` extension so that you start and stop your container once for the whole test suite. For example:

```
import junit.extensions.TestSetup;
import junit.framework.Test;
[...]

public class CargoTestSetup extends
TestSetup
{
    InstalledLocalContainer container;

    public CargoTestSetup(Test test)
    {
        super(test);
    }
```

```
    protected void setUp() throws Exception
    {
        // (1) Optional step to install the
container from a URL pointing to its
distribution
        Installer installer = new
ZipURLInstaller(
            new
URL("http://archive.apache.org/dist/tomcat/t
omcat-6/v6.0.32/bin/apache-tomcat-6.0.32.zip
"));
        installer.install();

        // (2) Create the Cargo Container
instance wrapping our physical container
        LocalConfiguration configuration =
(LocalConfiguration) new
DefaultConfigurationFactory().createConfigur
ation(
            "tomcat6x",
ContainerType.INSTALLED,
ConfigurationType.STANDALONE);
        InstalledLocalContainer container =
            (InstalledLocalContainer) new
DefaultContainerFactory().createContainer(
                "tomcat6x",
ContainerType.INSTALLED, configuration);

container.setHome(installer.getHome());

        // (3) Statically deploy some WAR
```

```
(optional)
        configuration.addDeployable(new
WAR("cargo.war"));

        // (4) Start the container
        container.start();
    }

    protected void tearDown() throws
Exception
    {
        // (6) Stop the container
```

```
        container.stop();
    }
}
```

Then write your test case. For example:

```
import junit.framework.TestCase;
import junit.framework.Test;
import junit.framework.TestSuite;
[...]

public class MyTest extends TestCase
{
    public static Test suite()
    {
        TestSuite suite = new
TestSuite(MyTest.class);
        return new CargoTestSetup(suite);
    }

    public void testSomething()
    {
        // (5) Perform any test you wish
here
        [...]
    }
}
```

References

- Using Cargo 0.4 for functional testing