JSR88

Definition

JSR88-compliant containers support

Explanation

Cargo supports JSR 88: J2EE Application Deployment API, allowing it to be used with any JSR88-compliant container.

The core functionality is implemented by the `o.c.c.container.spi.deployer.AbstractJsr88Deployer` class (a Deployer implementation), which acts as a proxy to the JSR88 DeploymentManager.

GlassFish

The GlassFish 3.x remote container uses the JSR-88 API. For the connection to succeed, the following JARs need to be in the container classpath or in the current Java Thread's context classloader:

- deployment-client.jar and its dependencies; which seem to be:
  - admin-cli.jar
  - auto-depends.jar
  - common-util.jar
  - glassfish-api.jar
  - deployment-common.jar
  - hk2-core.jar

Here is an example code for the users of the Java API:

```
List<URL> urls = new ArrayList<URL>();

// Add many libraries from GlassFish
for (File jar : new File(this.localContainer.getHome(),
"glassfish/modules").listFiles())
{
    if (jar.isFile())
    {
        urls.add(jar.toURI().toURL());
    }
}

// Create a ClassLoader contaning all these JARs
URL[] urlsArray = new URL[urls.size()];
urlsArray = urls.toArray(urlsArray);
URLClassLoader classLoader = new URLClassLoader(urlsArray,
Thread.currentThread().getContextClassLoader());
Thread.currentThread().setContextClassLoader(classLoader);

// Now, create the GlassFish Remote container
...
```

Here is an example Maven2 plugin configuration:

```
<pluginRepositories>
  <pluginRepository>
     <id>glassfish</id>
     <url>http://download.java.net/maven/glassfish</url>
  </pluginRepository>
</pluginRepositories>

<repositories>
  <repository>
     <id>glassfish</id>
     <url>http://download.java.net/maven/glassfish</url>
  </repository>
</repositories>

...

<plugin>
  <groupId>org.codehaus.cargo</groupId>
  <artifactId>cargo-maven2-plugin</artifactId>
  <version>${cargo.plugin.version}</version>
  <configuration>
    <container>
      <containerId>glassfish3x</containerId>
      <type>remote</type>
    </container>
    <configuration>
      <type>runtime</type>
      <properties>
        <cargo.hostname>virtualbox-xp</cargo.hostname>
        <cargo.rmi.port>4848</cargo.rmi.port>
      </properties>
    </configuration>
  </configuration>
  <dependencies>
    <dependency>
      <groupId>org.glassfish.deployment</groupId>
      <artifactId>deployment-client</artifactId>
      <version>3.0.1</version>
    </dependency>
  </dependencies>
</plugin>
```

When adding the `org.glassfish.deployment:deployment-client` dependency, please make sure that the version matches **exactly** the targeted GlassFish server's version (beware the updates that you install on GlassFish, these indeed change the version of your server); else you might run into unexpected surprises like:

```
error submitting remote command:
com.sun.enterprise.admin.cli.CommandException:
   CLI001 Invalid Command: list: InvalidCommandException ->
[Help 1]
```

or

```
GlassFish list help command report

Exit Code : SUCCESS
list - null

monitor
GeneratedHelp true
pattern_operand
SYNOPSIS Usage: list [--monitor=false] pattern %%%EOL%%%
```

The list of versions you can choose from can be found on: http://download.java.net/maven/glassfish/org/glassfish/deployment/deployment-client/