What is a JBI SA and how do I create one?

What is a JBI SA and how do I create one?

The JBI spec also includes a deployment unit known as a service assembly. SA is an abbreviation for a JBI service assembly. A JBI SA is used to package up JBI SUs in order to be deployed to a container. A SA can include one or more SUs. The concept of the JBI SA is similar to a JavaEE EAR file that is used to package up one or more WAR files. The SA is really just an archive to bundle up SUs for deployment to the JBI container.

Creating a SA

Creating a SA is rather simple once you have created at least one SU (see What is a JBI SU and how do I create one? for information on creating a SU). This example is generic because it works with any SUs. But the first task is to create a Maven project to house the SA configuration. To create a JBI SA, you will use the Maven archetypes that are provided with ServiceMix. Maven archetypes are used to create a Maven project skeleton to jumpstart project creation via the automation of repetitive tasks by following standard conventions. The result of using a Maven archetype to create a Maven project is a directory structure, a Maven POM file (`pom.xml`) and, depending on the archetype being used, sometimes Java objects and JUnit tests.

Creating a SA Maven Project

Using a Maven archetype, we will create a simple project whose job is to simply bundle up other project. Below is the command to create the SA Maven project. You may replace the -DarchetypeVersion with whatever ServiceMix version you are using:

```
$ [SMX_HOME]/bin/smx-arch sa
  -DgroupId=com.mycompany \
  -DartifactId=my-sa
```

The command above utilizes the servicemix-service-assembly Maven archetype to create a Maven project named my-sa. This command creates a directory named `my-sa` that contains a Maven project skeleton meaning all the necessary files are in place, you simply need to enter the correct values for the configuration. Below is the directory structure created by this archetype:

```
./pom.xml
```

The only thing that is necessary in a SA Maven project is a the Maven `pom.xml` file. The `pom.xml` contains the necessary Maven JBI plugin configuration. This is a plugin that automatically generates the `META-INF/jbi.xml` file based on other information in the `pom.xml` file. The only thing you need to do is specify which SUs will be included in this SA.

Including SUs Inside the SA

In order to actually include SUs inside of the SA, all you need to do is list the SUs as dependencies in the `pom.xml` file. Below is an example of including the my-consumer-su SU inside this SA:

```
<dependencies>
  <dependency>
    <groupId>com.mycompany</groupId>
    <artifactId>my-consumer-su</artifactId>
    <version>1.0-SNAPSHOT</version>
  </dependency>
</dependencies>
```

This information was gleaned from the my-consumer-su `pom.xml` file in the What is a JBI SU and how do I create one? FAQ entry. Now that we have a SU listed as a dependency, it's time to build and package the SA. Below is the Maven command to do this:

```
$ mvn install
```

You will need to have Maven 2.0.4 or higher installed in order to run this command. This packages up the my-consumer-su SU inside of the my-sa SA and places the my-sa build artifact in the target directory.

JBI SA Requirements

There are only two requirements of a JBI SA - that it contain a JBI deployment descriptor located in `META-INF/jbi.xml` and that it contain at least one SU. Below is the hierarchy of the contents of the SU archive that is created when running the Maven install goal as noted above:

```
$ jar tvf ./target/my-sa-1.0-SNAPSHOT.jar
     0 Wed Mar 28 21:10:02 MDT 2007 META-INF/
   126 Wed Mar 28 21:10:00 MDT 2007 META-INF/MANIFEST.MF
   596 Wed Mar 28 21:10:00 MDT 2007 META-INF/jbi.xml
  3391 Wed Mar 28 21:10:00 MDT 2007 my-consumer-su-1.0-SNAPSHOT.zip
     0 Wed Mar 28 21:10:02 MDT 2007 META-INF/maven/
     0 Wed Mar 28 21:10:02 MDT 2007 META-INF/maven/com.mycompany/
     0 Wed Mar 28 21:10:02 MDT 2007 META-INF/maven/com.mycompany/my-sa/
  2461 Wed Mar 28 21:09:44 MDT 2007 META-INF/maven/com.mycompany/my-sa/pom.xml
   110 Wed Mar 28 21:10:00 MDT 2007 META-INF/maven/com.mycompany/my-sa/pom.properties
```

Notice that the SA archive contains not only the required `META-INF/jbi.xml` file, but also the included `my-consumer-su-1.0-SNAPSHOT.zip` file. The `META-INF/jbi.xml` file was generated by the JBI Maven plugin. Below is the contents of this file:

```

$ cat ./target/my-sa-1.0-SNAPSHOT-installer/META-INF/jbi.xml
<?xml version="1.0" encoding="UTF-8"?>
<jbi xmlns="http://java.sun.com/xml/ns/jbi" version="1.0">
  <service-assembly>
    <identification>
      <name>my-sa</name>
      <description>A custom project</description>
    </identification>
    <service-unit>
      <identification>
        <name>my-consumer-su</name>
        <description>my-consumer-su</description>
      </identification>
      <target>
        <artifacts-zip>my-consumer-su-1.0-SNAPSHOT.zip</artifacts-zip>
        <component-name>servicemix-http</component-name>
      </target>
    </service-unit>
  </service-assembly>
```

This file is a deployment descriptor for the JBI container when the SA is deployed. Notice that it contains information about the SA in general but also information about the included SUs, including the JBI component on which the SU depends (the servicemix-http component).

This is all that is needed to create the SA. The SA can now be deployed by copying the my-sa-1.0-SNAPSHOT.jar file into the ServiceMix deploy directory, deployed via a JMX console or via the ServiceMix Ant Tasks. Happy deploying 🙂!

Deployment

To deploy the SA to ServiceMix, simply copy the SA JAR/ZIP file from the target directory to the ServiceMix deploy directory. The deployer should automatically pick up the SA and output some logging to both the console and the `data/log/servicemix.log` file as the deployment takes place to let you know what is happening. You can also get more detail about the deployment by increasing the log level in the `conf/log4j.xml` file from INFO to DEBUG. For more information on ServiceMix logging, see How do I change the logging.

Additional Information

Have you walked through the Tutorials yet? This is a great place to start if you're new to JBI.