

JMS Resources deployment with deploy tool

You can deploy jms resources with deploy command (without web console). When you want to create a group of resources consist of ConnectionFactory, topics and queues you only have to deploy a geronimo-activemq-ra connector (RA meaning "ResourceAdapter") with an deploy plan like this...



```

<connector xmlns="http://geronimo.apache.org/xml/ns/j2ee/connector-1.2">
  <dep:environment xmlns:dep="http://geronimo.apache.org/xml/ns/deployment-1.2">
    <dep:moduleId>
      <dep:groupId>console.jms</dep:groupId>
      <dep:artifactId>jmsResources</dep:artifactId>
      <dep:version>1.0</dep:version>
    <dep:type>rar</dep:type>
  </dep:moduleId>
  <dep:dependencies>
    <dep:dependency>
      <dep:groupId>org.apache.geronimo.configs</dep:groupId>
      <dep:artifactId>activemq-broker</dep:artifactId>
      <dep:type>car</dep:type>
    </dep:dependency>
  </dep:dependencies>
</dep:environment>
<resourceadapter>
  <resourceadapter-instance>
    <resourceadapter-name>jmsResources</resourceadapter-name>
    <workmanager xmlns="http://geronimo.apache.org/xml/ns/naming-1.2">
      <gbean-link>DefaultWorkManager</gbean-link>
    </workmanager>
  </resourceadapter-instance>
  <outbound-resourceadapter>
    <connection-definition>
      <connectionfactory-interface>javax.jms.ConnectionFactory</connectionfactory-interface>
      <connectiondefinition-instance>
        <name>IHCICConnectionFactory</name>
        <implemented-interface>javax.jms.QueueConnectionFactory</implemented-interface>
        <implemented-interface>javax.jms.TopicConnectionFactory</implemented-interface>
        <connectionmanager>
          <xa-transaction>
            <transaction-caching />
          </xa-transaction>
          <single-pool>
            <max-size>10</max-size>
            <min-size>0</min-size>
            <blocking-timeout-milliseconds>5000</blocking-timeout-milliseconds>
            <idle-timeout-minutes>0</idle-timeout-minutes>
            <match-one />
          </single-pool>
        </connectionmanager>
      </connectiondefinition-instance>
    </connection-definition>
  </outbound-resourceadapter>
</resourceadapter>

<!-- ElementSupportData topic-->
<adminobject>
  <adminobject-interface>javax.jms.Topic</adminobject-interface>
  <adminobject-class>org.apache.activemq.command.ActiveMQTopic</adminobject-class>
  <adminobject-instance>
    <message-destination-name>ElementSupportData</message-destination-name>
    <config-property-setting name="PhysicalName">ElementSupportData</config-property-setting>
  </adminobject-instance>
</adminobject>

<!-- Sample Queue -->

  <adminobject>
    <adminobject-interface>javax.jms.Queue</adminobject-interface>
    <adminobject-class>org.apache.activemq.command.ActiveMQQueue</adminobject-class>
    <adminobject-instance>
      <message-destination-name>SampleQueue</message-destination-name>
      <config-property-setting name="PhysicalName">SampleQueue</config-property-setting>
    </adminobject-instance>
  </adminobject>
</connector>

```

Now you only have to deploy it with this command (I named deploy plan as JmsResources.xml) ussing for this example geronimo 2.0.2:

```
java -jar %GERONIMO_HOME%/bin/deployer.jar --user system --password manager --host localhost deploy JmsResources.xml %GERONIMO_HOME%/repository/org/apache/geronimo/modules/geronimo-activemq-ra/2.0.2/geronimo-activemq-ra-2.0.2.rar
```

Remember that geronimo-activemq-ra-2.0.2.rar path or name can change according to the geronimo 2.x installation that you have. If deploy works well, in Geronimo Web Console-> JMS Resources will appear a new JMS Resource group named jmsResources (console.jms/jmsResources/1.0/rar). console.jms/jmsResources/1.0/rar is the component name created ussing data defined in moduleId tag.

If you want to undeploy it, you can do it from Geronimo Web Console -> J2EE connectors or with deploy command as this manner...

```
java -jar %GERONIMO_HOME%/bin/deployer.jar --user system --password manager --host localhost undeploy console.jms/jmsResources/1.0/rar
```

Remember set RemoteDeployHostname into %GERONIMO_HOME%\var\config\config-substitutions.properties if you want to do remote deploy (when --host parameter is not the localhost).