SupportPac – FA01 Version 1.0
WebSphere MQ File Transfer Edition WSRR Plugin
Readme

Rich Cumbers (rich.cumbers@uk.ibm.com)

22nd May 2009

Setting up WSRR Discovery Plugin

Add the following jars to WSRR, they can be found in a WMQFTE installation, in the lib directory The above jars need to be added to WSRR. Using the Configuration perspective: Active Configuration Profile -> Plugins -> Jars

- com.ibm.mq.jmqi.jar
- com.ibm.mq.jmqi.local.jar
- com.ibm.mq.jmqi.remote.jar
- com.ibm.mq.jmqi.system.jar
- com.ibm.msg.client.commonservices.jar
- com.ibm.msg.client.commonservices.j2se.jar
- com.ibm.wmqfte.agent.jar
- com.ibm.wmqfte.bootstrap.jar
- com.ibm.wmqfte.cmdline.jar
- com.ibm.wmqfte.common.jar

You must ensure that the WAS installation is using these WebSphere MQ Version 7 jars and not the built-in WebSphere MQ Version 6 jars. For more information on this consult the WAS documentation.

You also need to add the WMQFTE-DiscoveryPlugin.jar that can be found in the SupportPac zip you downloaded.

Add Model Data to WSRR

In order to use the plugin you must add the object definitions that we use. The file can be found in the models directory inside the zip. Consult the WSRR documentation for how to add custom models to a WSRR installation.

Add a ServiceDiscoveryTask to WSRR

The next step is to add the configuration required for the WMQFTE Service Discovery task. Under the *Configuration perspective, Active Configuration profile -> Plug-ins* there is a menu option called *Service Discovery*. Clicking this brings up a pane on the right hand side with an option called: *ServiceDiscoveryConfiguration* . Clicking that will show an xml editor pane, in which you need to paste the following, changing the values in blue as appropriate.

```
<discovery-task name="WMQFTEDiscoveryTask">
  <discoverer type="WMQFTE">
    <discovererClass>com.ibm.wmqfte.wsrr.servicediscovery.WMQFTEServiceDiscovery
</discovererClass>
    <parserClass>com.ibm.wmqfte.wsrr.servicediscovery.WMQFTEConfigParser</parserC
lass>
    <validatorClass>com.ibm.wmqfte.wsrr.servicediscovery.WMQFTEConfigValidator</v
alidatorClass>
  </discoverer>
  <governance>
    <discovered-
transition>http://www.ibm.com/xmlns/prod/serviceregistry/6/0/governance/DefaultL
ifecycle#Discover</discovered-transition>
    <removed-
transition>http://www.ibm.com/xmlns/prod/serviceregistry/6/0/governance/DefaultL
ifecycle#Remove</removed-transition>
  </governance>
  <configuration>
    <qmgr>coordination_qm</qmgr>
    <host>scorpio.hursley.ibm.com</host>
    <port>1414</port>
    <channel>SYSTEM.DEF.SVRCONN</channel>
  </configuration>
</discovery-task>
```

In order to have the above defined task run, you will need to add something to the scheduler. Under the *Configuration perspective, Active Configuration profile -> Plug-ins* there is a menu option called Scheduler. Clicking this brings up a pane on the right hand side with an option called: *ServiceDiscoveryScheduler*.

Clicking that will show an xml editor pane, in which you need to paste the following, again changing the blue values as appropriate:

```
<scheduler-task name="WMQFTEDiscoveryTask">
  <type>Timed</type>
  <interval>2</interval>
  <class>com.ibm.sr.servicediscovery.ServiceDiscoveryTask</class>
  <enabled>true</enabled>
  <startTimeOffset>1</startTimeOffset>
  <intervalGranularity>minutes</intervalGranularity>
  <transactionality>None</transactionality>
</scheduler-task>
```

You should have a working WSRR Service Discovery Plugin, and you will start to see Agent, Queue Manager and Connection objects being created and updated in the Concepts page.

Notes

The plugin does output log information, checking the SystemOut.log file of your WAS installation should yield diagnostic information if the plugin does not appear to be working as it should.