Instruction Sheet  
Feb 21, 2005  
Setting up the example program demonstrating JMS and Message-driven beans (Message Distribution System)

Before starting, make sure you have started up the admin server and have JMC loaded up in your browser. We will create a new application on an existing server, say, the j2eeSamples server. Download the file mds-ear.tar from:
http://www.cse.buffalo.edu/~mhvora/4587/samples/mds-ear.tar
OR
/projects/Spring_2004/cse487/mhvora/public/jms/mds-ear.tar
and place it in the server’s ($JRUN/servers/j2eeSamples) directory.

Creating “Administered Objects” on the Server
1. Go into the JMS Resources section of the j2eeSamples server and delete existing Destinations (testQueue) & Factories.
2. Create aConnectionFactory with the following parameters:
   Name: CallerQueueConnectionFactory
   JNDI Name: jms/factory/CallerQueueConnectionFactory
   Type: Queue Connection Factory
   Transport: RMI
   Username: guest
   Password: guest
3. Create a Destination with the following parameters:
   Name: CallerQueue
   JNDI Name: jms/queue/CallerQueue
   Type: Queue

Deploying application
1. Untar the contents of mds-ear.tar.
   This application contains the following components:
   • mds-ear: An enterprise application (Message Distribution System)
   • secretary-ejb: A message-driven bean receiving messages from a JMS queue
   • boss-ejb: A session bean that the secretary connects to
   • caller-jar: A J2EE (GUI-based) client that sends messages on a JMS queue
2. The three components are precompiled. They will automatically be deployed except for one thing:
   In the file, server-root/mds-ear/caller-jar/mds/Caller.java you need to change the “hostname:99999” to your machine name and JNDI port number respectively.
3. After changing this you must compile the file Caller.java from outside the mds directory.

Executing application
1. Go to the directory server-root/mds-ear/caller-jar.
2. Execute client by typing
   javac mds/Caller
3. This application sends a message onto the JMS queue which is picked up by the MDB.
4. Multiple instances of the application client can send messages at the same time.