

Lab assignment 6: Integration patterns

Exercise 1: Split up components into applications

Take the webshop application you wrote in lab 5, and convert it into 5 separate applications:

- Customers
- Products
- Shopping
- Order
- Webshop client
- (Suppliers: if you have implemented this)

Every application should be implemented in its own project.

You can either copy and paste the webshop project 4 times, and then rename the projects, or you create new projects, and copy and paste the classes into these new projects. Make sure you have the required libraries in the POM file.

The goal of this exercise is to make sure all 5 applications are completely separated from each other. It is not allowed to import a class that exist in another application. It is also not allowed to place one project on the classpath of another project.

Start all 4 webshop applications and then run the webshop client which should then call the individual applications with REST calls.

Exercise 2: JMS

For this exercise we will use ActiveMQ as JMS middleware so we first have to start ActiveMQ.

First edit the file **C:\springtraining\apache-activemq-5.15.3\bin\activemq.bat**

Make the following changes:

```
activemq - Notepad
File Edit Format View Help
REM See the License for the specific language governing permissions and
REM limitations under the License.
REM -----

if exist "%HOME%\activemqrc_pre.bat" call "%HOME%\activemqrc_pre.bat"

if "%OS%"=="Windows_NT" @setlocal

set ACTIVEMQ_HOME=C:\architecturetraining\apache-activemq-5.15.3

rem %~dp0 is expanded pathname of the current script under NT
set DEFAULT_ACTIVEMQ_HOME=%~dp0..

if "%ACTIVEMQ_HOME%"==" " set ACTIVEMQ_HOME=%DEFAULT_ACTIVEMQ_HOME%
set DEFAULT_ACTIVEMQ_HOME=

:doneStart
rem find ACTIVEMQ_HOME if it does not exist due to either an invalid value passed
rem by the user or the %0 problem on Windows 9x
if exist "%ACTIVEMQ_HOME%\README.txt" goto checkJava

rem check for activemq in Program Files on system drive
if not exist "%SystemDrive%\Program Files\activemq" goto checkSystemDrive
set ACTIVEMQ_HOME=%SystemDrive%\Program Files\activemq
goto checkJava

:checkSystemDrive
rem check for activemq in root directory of system drive
if not exist %SystemDrive%\activemq\README.txt goto checkCDrive
set ACTIVEMQ_HOME=%SystemDrive%\activemq
goto checkJava

:checkCDrive
rem check for activemq in C:\activemq for Win9X users
if not exist C:\activemq\README.txt goto noAnthome
set ACTIVEMQ_HOME=C:\activemq
goto checkJava

:noAnthome
echo ACTIVEMQ_HOME is set incorrectly or activemq could not be located. Please set ACTIVEMQ_HOME.
goto end

:checkJava
set JAVACMD=%JAVACMD%

set JAVA_HOME=C:\architecturetraining\jdk1.8.0\jre

if "%JAVA_HOME%" == "" goto noJavaHome
if not exist "%JAVA_HOME%\bin\java.exe" goto noJavaHome
```

Then save **activemq.bat**

Then run the file **startactivemq.bat** and activemq will start:

```
C:\WINDOWS\system32\cmd.exe

C:\springtraining\apache-activemq-5.15.3\bin>activemq start
Java Runtime: Oracle Corporation 1.8.0_45 C:\springtraining\jdk1.8.0\jre
Heap sizes: current=1013632k free=991262k max=1013632k
JVM args: -Dcom.sun.management.jmxremote -Xms1G -Xmx1G -Djava.util.logging.c
onfig.file=logging.properties -Djava.security.auth.login.config=C:\springtrainin
g\apache-activemq-5.15.3\conf\login.config -Dactivemq.classpath=C:\springtrainin
g\apache-activemq-5.15.3\conf;C:\springtraining\apache-activemq-5.15.3\conf;C:\s
pringtraining\apache-activemq-5.15.3\conf; -Dactivemq.home=C:\springtraining\apa
che-activemq-5.15.3 -Dactivemq.base=C:\springtraining\apache-activemq-5.15.3 -Da
ctivemq.conf=C:\springtraining\apache-activemq-5.15.3\conf -Dactivemq.data=C:\sp
ringtraining\apache-activemq-5.15.3\data -Djava.io.tmpdir=C:\springtraining\apac
he-activemq-5.15.3\data\tmp
Extensions classpath:
[C:\springtraining\apache-activemq-5.15.3\lib,C:\springtraining\apache-activem
q-5.15.3\lib\camel,C:\springtraining\apache-activemq-5.15.3\lib\optional,C:\sprin
gtraining\apache-activemq-5.15.3\lib\web,C:\springtraining\apache-activemq-5.15
.3\lib\extra]
ACTIVEMQ_HOME: C:\springtraining\apache-activemq-5.15.3
ACTIVEMQ_BASE: C:\springtraining\apache-activemq-5.15.3
ACTIVEMQ_CONF: C:\springtraining\apache-activemq-5.15.3\conf
ACTIVEMQ_DATA: C:\springtraining\apache-activemq-5.15.3\data
Loading message broker from: xbean:activemq.xml
INFO : Refreshing org.apache.activemq.xbean.XBeanBrokerFactory$1@2f8f11: startu
p date [Tue Mar 20 16:55:57 BOT 2018]; root of context hierarchy
INFO : Using Persistence Adapter: KahaDBPersistenceAdapter[C:\springtraining\ap
ache-activemq-5.15.3\data\kahadb]
INFO : KahaDB is version 6
INFO : PLISTore[C:\springtraining\apache-activemq-5.15.3\data\localhost\tmp_
storage] started
INFO : Apache ActiveMQ 5.15.3 (localhost, ID:Rene-64057-1521579361488-0:1) is s
tarting
INFO : Listening for connections at: tcp://Rene:61616?maximumConnections=1000&w
ireFormat.maxFrameSize=104857600
INFO : Connector openwire started
INFO : Listening for connections at: amqp://Rene:5672?maximumConnections=1000&w
ireFormat.maxFrameSize=104857600
INFO : Connector amqp started
INFO : Listening for connections at: stomp://Rene:61613?maximumConnections=1000
&wireFormat.maxFrameSize=104857600
INFO : Connector stomp started
INFO : Listening for connections at: mqtt://Rene:1883?maximumConnections=1000&w
ireFormat.maxFrameSize=104857600
INFO : Connector mqtt started
WARN : ServletContext@o.e.j.s.ServletContextHandler@e3540e{/,null,STARTING} has
uncovered http methods for path: /
INFO : Listening for connections at ws://Rene:61614?maximumConnections=1000&wir
eFormat.maxFrameSize=104857600
INFO : Connector ws started
INFO : Apache ActiveMQ 5.15.3 (localhost, ID:Rene-64057-1521579361488-0:1) star
ted
INFO : For help or more information please see: http://activemq.apache.org
WARN : Store limit is 102400 mb (current store usage is 4 mb). The data directo
ry: C:\springtraining\apache-activemq-5.15.3\data\kahadb only has 10068 mb of us
able space. - resetting to maximum available disk space: 10068 mb
WARN : Temporary Store limit is 51200 mb (current store usage is 0 mb). The dat
a directory: C:\springtraining\apache-activemq-5.15.3\data only has 10063 mb of
usable space. - resetting to maximum available disk space: 10063 mb
INFO : No Spring WebApplicationInitializer types detected on classpath
INFO : ActiveMQ WebConsole available at http://0.0.0.0:8161/
INFO : ActiveMQ Jolokia REST API available at http://0.0.0.0:8161/api/jolokia/
INFO : Initializing Spring FrameworkServlet 'dispatcher'
INFO : No Spring WebApplicationInitializer types detected on classpath
INFO : jolokia-agent: Using policy access restrictor classpath:/jolokia-access.
xml
```

Now that we have started ActiveMQ, we can continue by importing the projects.

Import the given **Exercise13-Sender** project and the **Exercise13-Receiver** project.
First run the `SpringJmsReceiverApplication.java` in the **Exercise13-Receiver** project.
Then run the `SpringJmsPersonSenderApplication` in the **Exercise13-Sender** project.

You should now see the following in the sender console:

```
Sending a JMS message.  
Sending a JMS message.  
.....
```

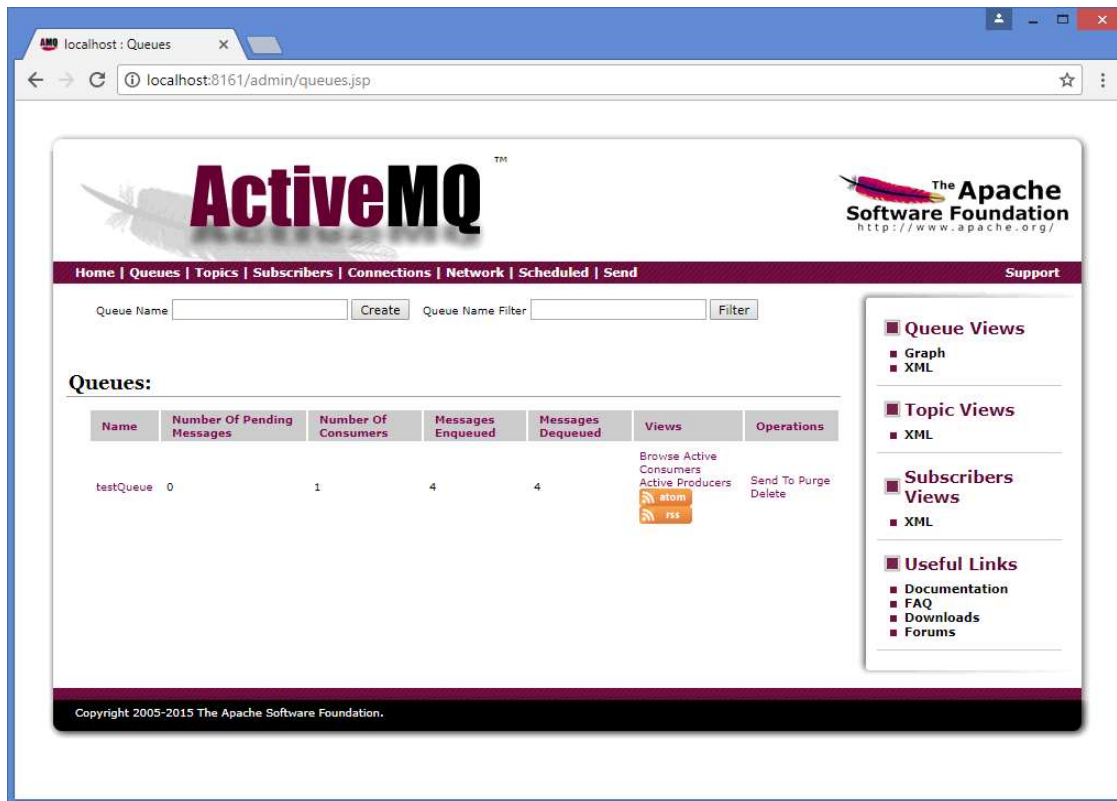
And you should see the following in the receiver console:

```
Receiver has started ...  
JMS receiver received message:Frank Brown  
JMS receiver received message:Mary Smith
```

Then open the ActiveMQ console at <http://localhost:8161/admin>.

You can login with username **admin** and password **admin**

Select the Queues page from the menu:

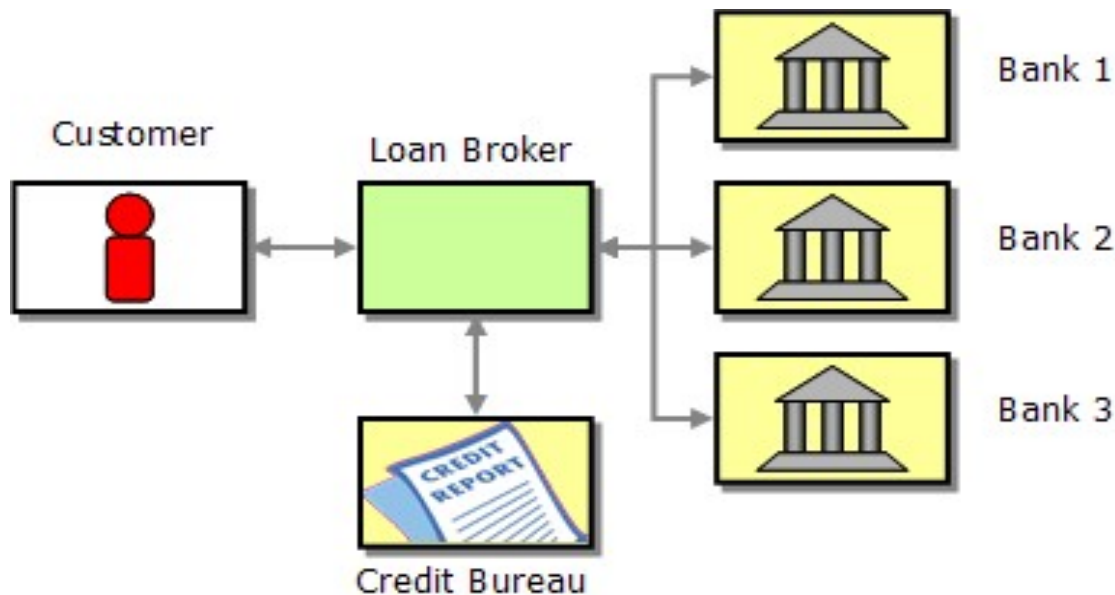


You see that the queue with name **testqueue** has one consumer, and 4 message have been received and consumed.

Now implement the following with JMS in your webshop of exercise 1:

When the user confirms the Order we need to update the stock quantity in the ProductCatalog. Implement this with JMS. Make sure you add the required dependencies in the POM file.

Exercise 3: Integration patterns



We have to design the integration of a loan broker between the following entities:

- The *Test Client* makes requests for loan quotes.
- The *Loan Broker* acts as the central process manager and coordinates the communication between the credit bureau and the banks.
- The *Credit Bureau* provides a service to the Loan Broker, computing customer's credit scores.
- Each *Bank* receives a quote request from the Loan Broker and submits an interest rate quote according to the loan parameters.

We have the following requirements:

- **Management Console:**
 - We want a single front-end that displays the health of all components and allows us to take compensating actions if something goes wrong.
- **Loan Broker Quality of Service:**
 - We want to monitor the loan broker's response times between quote request and response and send it to the management console.
- **Verify the Credit Bureau Operation:**
 - The Credit Bureau is an external service provided by a third party. We want to ensure the correct operation of this service by periodically sending test messages.
- **Credit Bureau Failover:**
 - If the Credit Bureau malfunctions we want to temporarily redirect the credit request messages to another service provider.

Draw the design of your integration solution using the integration patterns.