# 简介

[JMS](http://www.baike.com/wiki/JMS)（[Java Message Service](http://www.baike.com/wiki/Java+Message+Service)，[Java消息服务](http://www.baike.com/wiki/Java%E6%B6%88%E6%81%AF%E6%9C%8D%E5%8A%A1" \t "_blank" \o "Java消息服务)）是一组Java应用[程序](http://www.baike.com/wiki/%E7%A8%8B%E5%BA%8F" \t "_blank" \o "程序)接口(Java [API](http://www.baike.com/wiki/API))，它提供创建、发送、接收、读取[消息](http://www.baike.com/wiki/%E6%B6%88%E6%81%AF" \t "_new" \o "消息)的服务。由Sun公司和它的合作伙伴设计的JMS API定义了一组公共的应用程序接口和相应语法，使得Java程序能够和其他消息[组件](http://www.baike.com/wiki/%E7%BB%84%E4%BB%B6" \t "_new" \o "组件)进行通信。

  JMS是一种与厂商无关的 API，用来访问消息收发系统。它[类](http://www.baike.com/wiki/%E7%B1%BB" \t "_new" \o "类)似于 [JDBC](http://www.baike.com/wiki/JDBC) (Java [Database](http://www.baike.com/wiki/Database) Connectivity)：这里，JDBC 是可以用来访问许多不同关系[数据库](http://www.baike.com/wiki/%E6%95%B0%E6%8D%AE%E5%BA%93" \t "_new" \o "数据库)的 API，而 JMS 则提供同样与厂商无关的访问方法，以访问消息收发服务。许多厂商目前都支持 JMS，包括 IBM 的 MQSeries、BEA的 Weblogic JMS service和 Progress 的 [SonicMQ](http://www.baike.com/wiki/SonicMQ" \t "_blank" \o "SonicMQ)，这只是几个例子。

JMS 使您能够通过消息收发服务（有时称为消息中介程序或路由器）从一个 JMS 客户机向另一个 JML 客户机发送消息。消息是 JMS 中的一种[类型](http://www.baike.com/wiki/%E7%B1%BB%E5%9E%8B" \t "_new" \o "类型)[对象](http://www.baike.com/wiki/%E5%AF%B9%E8%B1%A1)，由两部分组成：报头和消息主体。报头由路由信息以及有关该消息的[元数据](http://www.baike.com/wiki/%E5%85%83%E6%95%B0%E6%8D%AE)组成。消息主体则携带着应用程序的数据或有效负载。根据有效负载的类型来划分，可以将消息分为几种类型，它们分别携带：简单文本 (TextMessage)、可序列化的对象 (ObjectMessage)、属性集合 (MapMessage)、字节流 (BytesMessage)、原始值流 (StreamMessage)，还有无有效负载的消息 (Message)。

# 模型：

|  |  |  |
| --- | --- | --- |
| 发送者 | MQServer | 接收者 |

P2P，一对一模型

发布订阅， 一对多模型

# 数据持久化：

<http://blog.csdn.net/xyw_blog/article/details/9128219>

* 可以使用远程数据库，配置jdbc

一是直接的JDBC操作，二是journal的jdbc操作。Journal是优化过的，先存文件，然后再批量写数据库。

* 文件存储KahaDB， 具体查看activemq.xml

# 协议：

* 发送者和接收者都长连接到MQServer。
* [Protocols](http://activemq.apache.org/protocols.html)
  + [AMQP](http://activemq.apache.org/amqp.html)
  + [MQTT](http://activemq.apache.org/mqtt.html)
  + [OpenWire](http://activemq.apache.org/openwire.html)
  + [REST](http://activemq.apache.org/rest.html)
  + [RSS and Atom](http://activemq.apache.org/rss-and-atom.html)
  + [Stomp](http://activemq.apache.org/stomp.html)
  + [WSIF](http://activemq.apache.org/wsif.html)
  + [WS Notification](http://activemq.apache.org/ws-notification.html)
  + [XMPP](http://activemq.apache.org/xmpp.html)

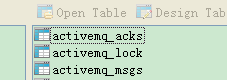
支持的数据类型有：JMS 定义了5中消息类型： TextMessage、MapMessage、BytesMessage、

StreamMessage和ObjectMessage

选择最合适的消息类型可以使JMS最有效 的处理消息。

# 配置：

配置mysql持久化

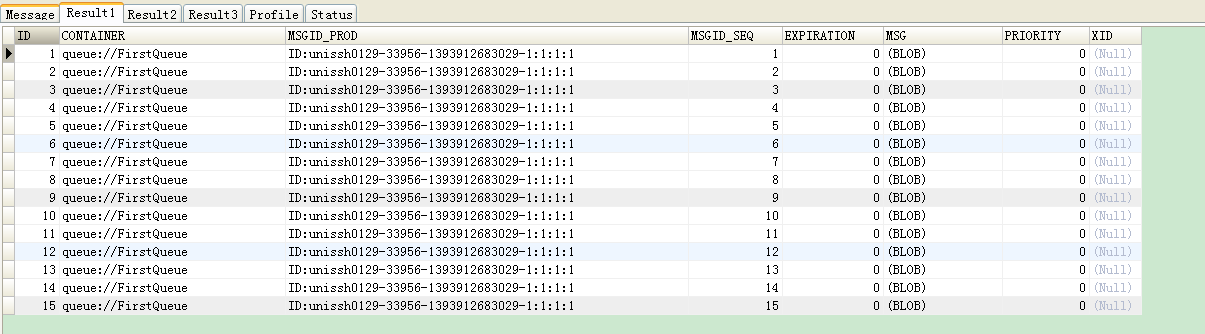


配置成功后查看数据库会发现多了3张表。

具体配置查看

**activemq.xml 和 jdbc.properties**

# 测试：



* producer.setDeliveryMode(DeliveryMode.PERSISTENT); // 消息设置持久化

发送15条数据后发现这15条数据已经作为离线消息存入数据库了。

* 当开始一个receiver后，成功接收到了消息，然后再查数据库就发现数据已经删了。

# 远程管理：

<!--

The managementContext is used to configure how ActiveMQ is exposed in

JMX. By default, ActiveMQ uses the MBean server that is started by

the JVM. For more information, see:

参考 http://activemq.apache.org/jmx.html

-->

<broker useJmx="true" brokerName="BROKER1">

</broker>

# 远程管理- activemqbrowser

在http://sourceforge.net/projects/activemqbrowser/上下载了个 activemqbrowser，远程管理工具。按照<http://activemq.apache.org/jmx.html> 配置后，死活连接不上，而用java自带的jConsole却可以连上去的。 叫我不得怀疑activemqbrowser这工具是不是有问题。

经过不懈的折腾发现：run\_activemq\_browser.bat 使用的5.6的jar。

|  |
| --- |
| **echo off**  **start "ActiveMQBrowser" javaw -Xms128m -Xmx512m -splash:cube.png -cp .\QBrowserV2\_Neo.jar;.\activemq-web-5.6.0.jar;.\activemq-all-5.6.0.jar;.\jide-oss-2.6.2.jar;.\imq.jar;.\jms.jar;.\imqadmin\_ja.jar;.\imqadmin.jar;.\imqutil\_ja.jar;.\imqutil.jar;.\imqjmx.jar;.\imqjmx\_ja.jar com.qbrowser.ActiveMQBrowser** |

而我安装的Activemq版本是5.9.0， 尝试将activemq-all-5.6.0.jar和activemq-web-5.6.0.jar替换为activemq-all-5.9.0.jar， activemq-web-5.9.0.jar。OK，居然OK了。

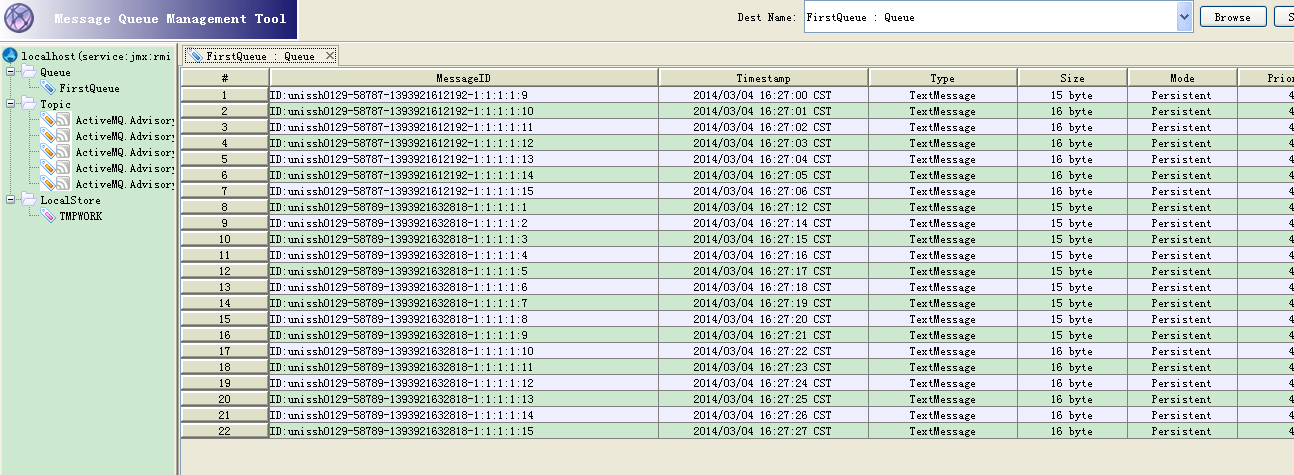
这是版本问题，搞定！！！！！

最后提醒一点，如果启动activemq过程中，出现

Error: Password file read access must be restricted: D:\apache-activemq-5.0.0\bin\../conf/  
jmx.password

参考 <http://eric-gcm.iteye.com/blog/1399500>， 修改文件访问属性即可解决！

附上连上成功截图：



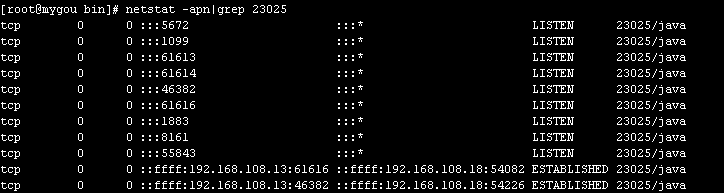
# 认证管理：

参考activemq.xml。

# JMX无法远程连接？

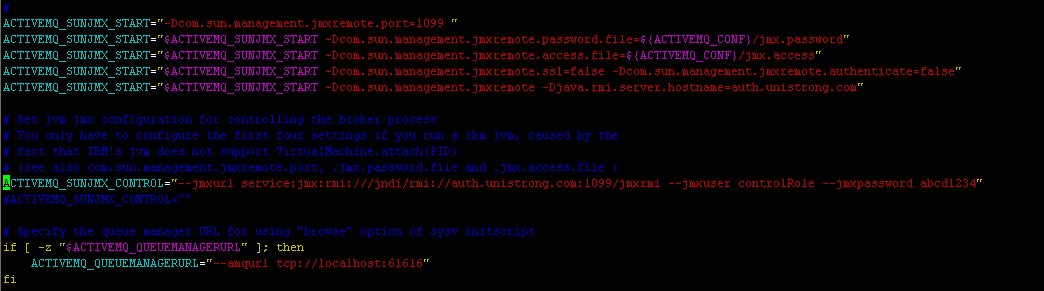
在redhat上安装了activemq， 启动都正常，查看端口也正常。测试程序sender，recver都OK。但是window上通过jconsole 连接时却私活连接不上。telnet 1099端口也OK. 到底是什么问题呢？

-端口都OK-



仔细检查配置，上网查资料，一遍又一遍的尝试。终于搞定了。

贴一下activemq配置。



关键就是加了 -Djava.rmi.server.hostname=auth.unistrong.com。

ACTIVEMQ\_SUNJMX\_CONTROL="--jmxurl service:jmx:rmi:///jndi/rmi://auth.unistrong.com:1099/jmxrmi --jmxuser controlRole --jmxpassword abcd1234"

都是使用的域名，对，这就是关键。<http://yinwei.163.org.blog.163.com/blog/static/86282092201081423231260/>

|  |
| --- |
| Jconsole连不上远程Jmx Server 2010-09-14 14:36:07|  分类： [默认分类](http://yinwei.163.org.blog.163.com/blog/#m=0&t=1&c=fks_087067092095084067084095084095092080083075086086095065) |  标签：[jmx](http://yinwei.163.org.blog.163.com/blog/" \l "m=0&t=3&c=jmx)  |举报|字号 订阅  具体问题就描述了可参看下面几个连接 http://shallon.javaeye.com/blog/129750 http://blog.csdn.net/cyq1984/archive/2010/07/26/5766776.aspx  皆因下面这个bug所致 http://bugs.sun.com/bugdatabase/view\_bug.do?bug\_id=6209663  解决办法无外乎两个，修改/etc/hosts这个文件。另外暴露jmx server时使用ip而非hostname。 若还是不行，则在jmx server启动时加上如下系统参数  -Djava.rmi.server.hostname=<host ip>  host ip与你jmx url中的ip一致即可。  写此博客帮助其他倒霉蛋。 |

# Topic和Queue模型

* Queue是P2P模型，多个Receiver同时访问Queue，Queue里面的每个msg不确定被谁收到，也就是说多个Receiver是抢占式的。
* Topic是发布订阅模型，多个Receiver同时访问Topic，每个Receiver都会收到msg，但没有对消息进行持久化。也就是说如果Receiver都不在线的话，sender发的消息都是没有持久化的。

**举例：**

**3台mdm服务器，集群，web页面选中10台设备推送策略，这10台设备不确定是连接的那台mdm服务器。3台mdm服务器都必须收到消息，解析消息判断当前连上设备中是否包含这10台设备，如果有才推送消息。**

**独立queue模型**

* **1个mdm创建1个queue，mdm1\_queue, mdm2\_queue, mdm3\_queue, Web和activemq建立3个session，发送3次。**

**发布订阅模型**

**Web建立和activemq建立1个session，创建mdmTopic， 3台mdm作为订阅者。**

**但是需要为这3台mdm持久化消息。**

|  |
| --- |
| **经测试是可以的。重启后receiver再连接上来也能收到。但是需要为每个receiver指定唯一的clientID。** |