# ActiveMQ分享7--ActiveMQ Master Slave集群配置

# 1.前言

     ActiveMQ Master/Slave集群可以提高ActiveMQ的高可用性，一旦一个Broker被killed，另一个Broker可以迅速代替。Master/Slave集群不支持负载均衡，仅能解决单点故障。

    ActiveMQ Master/Slave有三种配置方案：

    基于共享文件系统（KahaDB）

    基于JDBC

     基于可复制的Level DB

    本文演示的ActiveMQ Master/Slave采用的ActiveMQ5.10版本，JDK1.7，在一台Windows主机环境下部署。

    假设AMQ集群节点分别部署在下面的目录中：

|  |  |
| --- | --- |
| 1  2  3 | D:\MQ\apache-activemq\cluster\amq1  D:\MQ\apache-activemq\cluster\amq2  D:\MQ\apache-activemq\cluster\amq3 （基于LeveDB的时候用了3个节点演示） |

# 2.基于共享文件系统

      我们以在同一台机器上部署集群为例，如果部署在不同的物理主机，需要共享文件系统。

     修改的配置文件是conf目录下的activemq.xml。

## 2.1修改持久化适配器

     在amq1和amq2的activemq.xml中，找到persistenceAdapter，修改持久化适配器，指定共享文件的目录。这一点是最重要

|  |  |
| --- | --- |
| 1  2  3 | < persistenceAdapter >       < kahaDB  directory = "D:/MQ/apache-activemq/cluster/kahadb" />  </ persistenceAdapter > |

## 2.2修改端口号

   找到下面的行，修改amq2的端口号，默认的61616修改成了61617（模拟第二台物理机）

|  |  |
| --- | --- |
| 1  2  3 | < transportConnectors >      < transportConnector  name = "openwire"  uri = "[tcp://0.0.0.0:61617?maximumConnections=1000&amp;wireFormat.maxFrameSize=104857600](tcp://0.0.0.0:61617?maximumConnections=1000&wireFormat.maxFrameSize=104857600)" />  </ transportConnectors > |

## 2.3小结

我们为了模拟第二台物理机，修改了amq2的端口号。

事实上，如果是同一台物理机下，只需要各个AMQ节点指定相同的共享文件目录即可，端口不修改也可以。

集群启动后，只有率先启动的节点（Master）一直锁定这lock文件，其它节点（Slave）由于不能访问该文件而等待启动。一旦Master节点挂掉，lock文件锁被释放；其它Slave节点中率先锁定lock文件的，成为Master节点。

       Lock文件的位置是D:\MQ\apache-activemq\cluster\kahadb\lock。

# 3.基于JDBC

      这里我们以MySQL举例。

## 3.1添加数据源

     在amq1和amq2的activemq.xml中增加下面的内容，可以放在</broker>标签的后面。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | < bean  id = "mysql-ds"  class = "org.apache.commons.dbcp.BasicDataSource"  destroy-method = "close" >       < property  name = "driverClassName"  value = "com.mysql.jdbc.Driver" />       < property  name = "url"  value = "jdbc:mysql://localhost:3306/amq?relaxAutoCommit=true" />       < property  name = "username"  value = "root" />       < property  name = "password"  value = "root" />       < property  name = "maxActive"  value = "200" />       < property  name = "poolPreparedStatements"  value = "true" />  </ bean > |

3.2修改持久化适配器

     在amq1和amq2的activemq.xml中，找到persistenceAdapter，修改持久化适配器，如下所示。

|  |  |
| --- | --- |
| 1  2  3 | < persistenceAdapter >      < jdbcPersistenceAdapter  dataDirectory = "${activemq.data}"  dataSource = "#mysql-ds"  createTablesOnStartup = "false" />  </ persistenceAdapter > |

    createTablesOnStartup="false" 在启动前不要配置，等集群启动过了，再配置。createTablesOnStartup是启动时创建表，只创建一次就够了。

## 3.3添加JDBC驱动

    把mysql的驱动jar包放在各节点的lib目录下。

## 3.4小结

     基于JDBC和基于共享文件系统的集群方案原理是一致的，只是把共享文件系统换成了共享数据库。

     这种集群方式比共享文件系统方案更简单，更利于分布式部署，但是如果数据库失效，那么整个集群的节点随之全部失效。换言之，需要保证数据库的可靠性，比如采用数据库集群。

# 4.基于可复制LevelDB

    Leveldb是一个google实现的非常高效的kv数据库，是单进程的服务，能够处理十亿级别规模Key-Value型数据，占用内存小。

     基于可复制LevelDB的集群方案，需要引入ZooKeeper。根据ZooKeeper的使用方式可以分为单节点的ZooKeeper和Zookeeper集群。这里我们只讲述ZooKeeper集群，单节点不是一个可靠的选择。

## 4.1 Zookeeper集群配置

     ZooKeeper可以在网站<http://zookeeper.apache.org/> 下载。我们使用的是zookeeper-3.4.6，假设ZooKeeper分部署在下面三个目录中。

|  |  |
| --- | --- |
| 1  2  3 | D:\MQ\apache-activemq\cluster\zookeeper1  D:\MQ\apache-activemq\cluster\zookeeper2  D:\MQ\apache-activemq\cluster\zookeeper3 |

### 4.1.1 zoo.cfg

      把3个节点中config目录下的zoo\_sample.cfg复制一份，改成zoo.cfg。

      因为是在一台主机上部署，所以每个zoo.cfg中的clientPort不能重复；如果有三台主机，那么采用默认的clientPort就行，同理server.1、server.2、server.3可以写作<各自ip>:2888:3888。

       其中zookeeper1中的配置如下

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #默认配置  tickTime=2000  initLimit=10  syncLimit=5  #需要修改的配置  dataDir=D: /MQ/apache-activemq/cluster/zkdata/z1/  clientPort=2181  server.1=[localhost:2888:3888](http://localhost:2888:3888)  server.2=[localhost:2889:3889](http://localhost:2889:3889)  server.3=[localhost:2890:3890](http://localhost:2890:3890) |

      zookeeper2中的配置如下

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #默认配置  tickTime=2000  initLimit=10  syncLimit=5  #需要修改的配置  dataDir=D: /MQ/apache-activemq/cluster/zkdata/z2/  clientPort=2182  server.1=[localhost:2888:3888](http://localhost:2888:3888)  server.2=[localhost:2889:3889](http://localhost:2889:3889)  server.3=[localhost:2890:3890](http://localhost:2890:3890) |

      zookeeper3中的配置如下 

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #默认配置  tickTime=2000  initLimit=10  syncLimit=5  #需要修改的配置  dataDir=D: /MQ/apache-activemq/cluster/zkdata/z3/  clientPort=2183  server.1=[localhost:2888:3888](http://localhost:2888:3888)  server.2=[localhost:2889:3889](http://localhost:2889:3889)  server.3=[localhost:2890:3890](http://localhost:2890:3890) |

### 4.1.2 myid

创建三个目录

|  |  |
| --- | --- |
| 1  2  3 | D:\MQ\apache-activemq\cluster\zkdata\z1  D:\MQ\apache-activemq\cluster\zkdata\z2  D:\MQ\apache-activemq\cluster\zkdata\z3 |

每个目录中都创建一个名为myid的文件（文本文件，删掉txt后缀）,3个文件的内容分别写1、2、3。

|  |  |
| --- | --- |
| 1  2  3 | D:\MQ\apache-activemq\cluster\zkdata\z1\myid   1  D:\MQ\apache-activemq\cluster\zkdata\z2\myid   2  D:\MQ\apache-activemq\cluster\zkdata\z3\myid   3 |

## 4.2 ActiveMQ配置

    在3个amq节点中配置activemq.xml中的持久化适配器。根据以下配置示例，修改其中bind、zkAddress和hostname。如果你是在三台主机上部署，那么bind项可以写成bind="tcp://0.0.0.0:0"，默认采用61619端口。我们这里在一台主机上演示，因此需要保证bind端口不冲突。

     amq1的配置

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | < persistenceAdapter >         < replicatedLevelDB            directory = "${activemq.data}/leveldb"            replicas = "3"            bind = "<tcp://0.0.0.0:62618>"            zkAddress = "[localhost:2181,localhost:2182,localhost:2183](http://localhost:2181,localhost:2182,localhost:2183)"            hostname = "localhost"            zkPath = "/activemq/leveldb-stores"         />    </ persistenceAdapter > |

     amq2的配置

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | < persistenceAdapter >         < replicatedLevelDB            directory = "${activemq.data}/leveldb"            replicas = "3"            bind = "<tcp://0.0.0.0:62619>"            zkAddress = "[localhost:2181,localhost:2182,localhost:2183](http://localhost:2181,localhost:2182,localhost:2183)"            hostname = "localhost"            zkPath = "/activemq/leveldb-stores"         />    </ persistenceAdapter > |

        amq3的配置

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | < persistenceAdapter >         < replicatedLevelDB            directory = "${activemq.data}/leveldb"            replicas = "3"            bind = "<tcp://0.0.0.0:62620>"            zkAddress = "[localhost:2181,localhost:2182,localhost:2183](http://localhost:2181,localhost:2182,localhost:2183)"            hostname = "localhost"            zkPath = "/activemq/leveldb-stores"         />    </ persistenceAdapter > |

## 4.3 集群启动和错误排除

     先依次启动ZooKeeper，再依次启动ActiveMQ。启动第1个Zookeeper时，控制台会报连接错误，因为其它的ZooKeeper节点还没启动。

     如果出现了"activemq LevelDB IOException handler"错误，针对于apache-activemq-5.10.1，需要删除各个ActiveMQ节点下的pax-url-aether-1.5.2.jar包，并注释掉activemq.xml中的下述日志配置。

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | <!-- Allows accessing the server log       <bean id="logQuery" class="org.fusesource.insight.log.log4j.Log4jLogQuery"             lazy-init="false" scope="singleton"             init-method="start" destroy-method="stop">       </bean>     --> |

## 4.4 小结

     每个ActiveMQ的BrokerName必须相同，否则不能加入集群。

     经过验证，当一个ActiveMQ节点挂掉，或者一个ZooKeeper节点挂掉，ActiveMQ服务依然正常运转。如果仅剩一个ActiveMQ节点，因为不能选举Master，ActiveMQ不能正常运转；同样的，如果ZooKeeper仅剩一个节点活动，不管ActiveMQ各节点是否存活，ActiveMQ也不能正常提供服务。

# 5.客户端访问方式

## 5.1 Failover使用格式

|  |  |
| --- | --- |
| 1 | failover:([tcp://localhost:61616,tcp://localhost:61626](tcp://localhost:61616,tcp:))?randomize=false |

## 5.2 updateURIsURL

updateURIsURL，通过URL（或者本地路径）获取重连的url，这样做具有良好的扩展性，因为客户端每次连接都是从URL（或文件）中加载一次，所以可以随时从文件中更新url列表，做到动态添加MQ的备点。

|  |  |
| --- | --- |
| 1 | failover:()?randomize= false &updateURIsURL=[file:/d:/urllist.txt](http://file/d:/urllist.txt) |

urllist.txt中的地址通过英文逗号分隔，示例：

|  |  |
| --- | --- |
| 1 | tcp: //localhost :61616,tcp: //localhost :61617, tcp: //localhost :61618 |