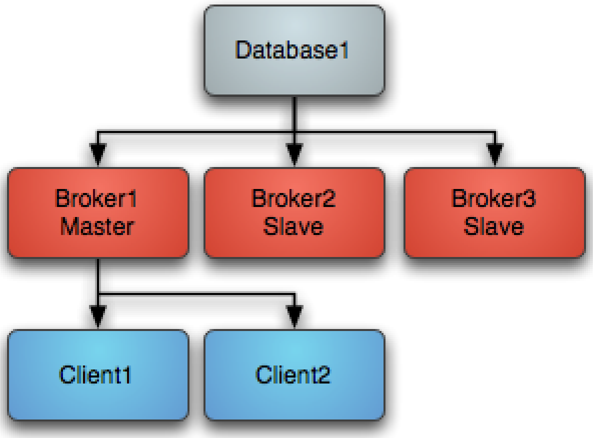
ActiveMQ高可用集群

1 Master Slave

https://activemq.apache.org/masterslave

一个单一的MQ实例，如果机器故障了，系统就不可用了。

ActiveMQ提供了主从集群机制来实现高可用。

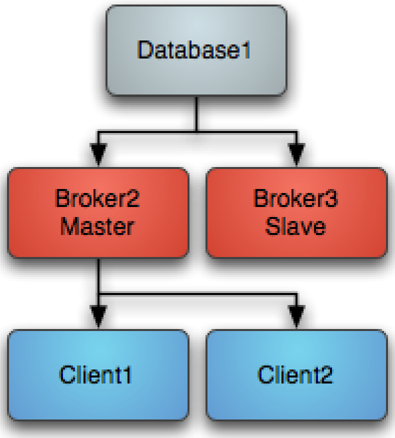


- 多个Broker实例共享存储
- 多个Broker通过抢独占锁来成为Master
- Master节点对外提供服务，Slave节点暂停等待独占锁
- Master节点故障，非持久化消息将丢失，所以一般要用持久化消息。
- 客户端以多Broker故障恢复方式进行连接

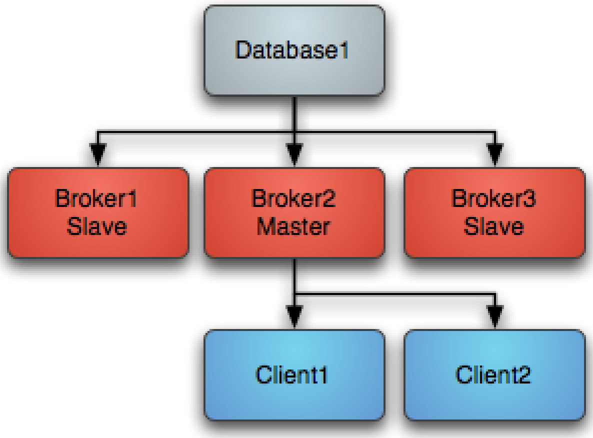
  ```
  failover:(tcp://broker1:61616,tcp://broker2:61616,tcp://broker3:61616)
  ```

  failover详细说明：http://activemq.apache.org/failover-transport-reference.html

Broke1 Master 故障



Broke1 重启



ActiveMQ中支持如下两种Master Slave实现方式：

- Shared File System Master Slave 独占锁 是共享存储目录下的lock文件

  https://activemq.apache.org/shared-file-system-master-slave
- JDBC Master Slave 独占锁为 activemq_lock表记录

  https://activemq.apache.org/jdbc-master-slave

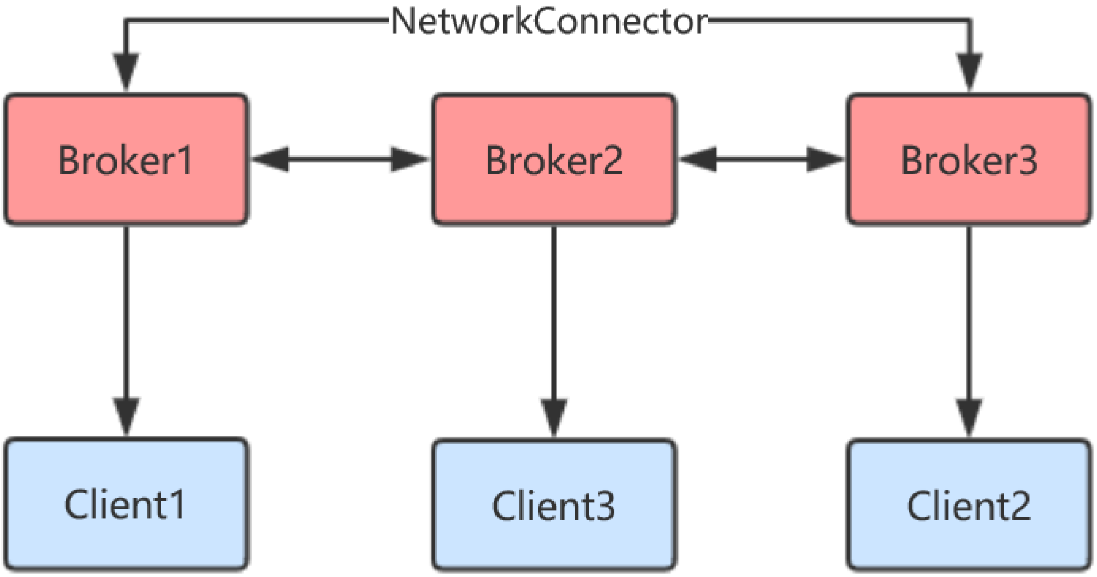
2 分布式队列和主题

2.1 Networks of Brokers

https://activemq.apache.org/networks-of-brokers

From 1.1 onwards of ActiveMQ supports *networks of brokers* which allows us to support distributed queues and topics across a network of brokers.

This allows a client to connect to any broker in the network - and fail over to another broker if there is a failure - providing from the clients perspective a HA cluster of brokers.



原理说明：

- 独立的Broker彼此相连；
- 客户端采用Failover方式连接任意一个Broker；
- 客户端生产的消息发送到它连接的broker，并存储在该broker上；
- 消费者客户端可以连接任意Broker来消费目标的消息。

https://activemq.apache.org/how-do-distributed-queues-work

Distributed Queues in Store/Forward

When we publish a message on a queue, it is stored in the persistent store of the broker that the publisher is communicating. Then if that broker is configured to store/foward to other brokers and clients, the broker will send it to *one* of these clients (which could be a node or a broker depending on the dispatch algorithm). This dispatch algorithm continues until the message is finally dispatched and consumed by a client.

At any point in time the message will only exist in one broker's store until its consumed. Note that messages are only distributed onto other brokers if there is a consumer on those brokers.

e.g. if we had broker A, B, C and a publisher on a queue on A. If we have consumers on the queue on A and B then messages for the queue will be spread across both brokers A and B; some messages going to B, some being consumed on A, none going to C. If a consumer on the queue starts on C, then messages will flow there too. If the consumer stops then no more messages will be dispatched to C.

配置：

- **静态IP配置方式：**

```
  <broker brokerName="receiver" persistent="false" useJmx="false">
    <networkConnectors>
      <!-- 配置要网络连接的其他Broker -->
      <networkConnector uri="static:(tcp://host1:61616,tcp://host2:61616,tcp://..)"/>
    </networkConnectors>
    ...
  </broker>
```

There are some useful properties you can set on a static network connector for retries: 一些有用的重试参数:

property	default	description
initialReconnectDelay	1000	time(ms) to wait before attempting a reconnect (if useExponentialBackOff is false)
maxReconnectDelay	30000	time(ms) to wait before attempting to re-connect
useExponentialBackOff	true	increases time between reconnect for every failure in a reconnect sequence
backOffMultiplier	2	multipler used to increase the wait time if using exponential back off

e.g.

```
uri="static:(tcp://host1:61616,tcp://host2:61616)?maxReconnectDelay=5000&useExponentialBackOff=false"
```

- **multicast 动态发现方式:**

```
<networkConnectors>
   <networkConnector uri="multicast://default"/>
</networkConnectors>
```

所有broker开启multicast:

```
<!-- transportConnector 中通过属性discoveryUri="multicast://default" 开启mulcast -->
<transportConnector name="openwire" uri="tcp://0.0.0.0:61616?maximumConnections=1000&amp;wireFormat.maxFrameSize=104857600" discoveryUri="multicast://default">
    <publishedAddressPolicy>
        <!--并添加publishedAddressPolicy 发布地址时发布IP，防止主机名不可访问的情况 -->
        <publishedAddressPolicy publishedHostStrategy="IPADDRESS" />
    </publishedAddressPolicy>
</transportConnector>
```

客户端连接:

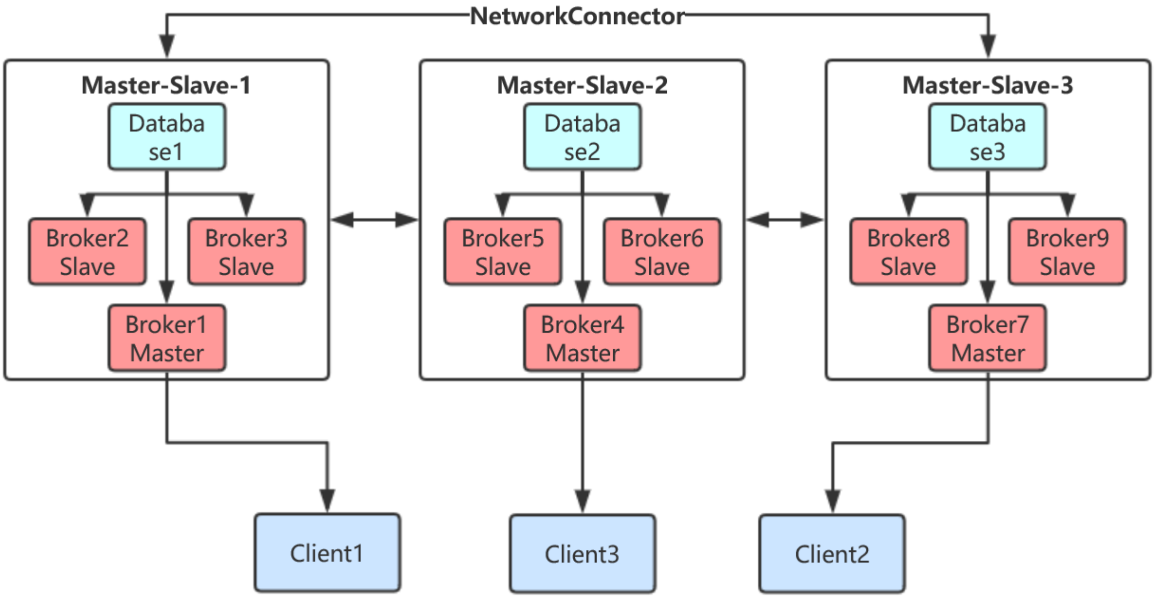
```
failover:(tcp://broker1:61616,tcp://broker2:61616,tcp://broker3:61616)
```

【注意】如果要实际使用，请详细了解networkConnector的配置属性：

https://activemq.apache.org/networks-of-brokers

Broker网络连接的不足：缺乏高可用

2.2 Networks + Master-Slave



networkConnectors 配置：

```
<networkConnectors>
   <networkConnector uri="masterslave:(tcp://other-cluster-1-master:61616,tcp://cluster-1-slave:61616)" />
    <networkConnector uri="masterslave:(tcp://other-cluster-2-master:61616,tcp://cluster-2-slave:61616)" />
    ………
</networkConnectors>
```

客户端连接：

```
failover:(tcp://broker1:61616,tcp://broker2:61616,tcp://broker3:61616,...)?randomize=true
```

可用配置参数请从官网文档详细了解：

https://activemq.apache.org/networks-of-brokers

http://activemq.apache.org/failover-transport-reference.html