

RocketMQ实战（一）



张丰哲 (/u/cb569cce501b) ✓已关注

2017.04.08 21:37 字数 1321 阅读 18267 评论 48 喜欢 110 赞赏 4

(/u/cb569cce501b)

阿里巴巴有2大核心的分布式技术，一个是OceanBase，另一个就是RocketMQ。在实际项目中已经领教过RocketMQ的强大，本人计划写一个RocketMQ实战系列，将涵盖RocketMQ的简介，环境搭建，初步使用、API详解、架构分析、管理员集群操作等知识。

What is RocketMQ?

RocketMQ作为一款分布式的消息中间件（阿里的说法是不遵循任何规范的，所以不能完全用JMS的那一套东西来看它），经历了Metaq1.x、Metaq2.x的发展和淘宝双十一的洗礼，在功能和性能上远超ActiveMQ。

1.要知道RocketMQ原生就是支持分布式的，而ActiveMQ原生存在单点性。

2.RocketMQ可以保证严格的消息顺序，而ActiveMQ无法保证！

3.RocketMQ提供亿级消息的堆积能力，这不是重点，重点是堆积了亿级的消息后，依然保持写入低延迟！

4.丰富的消息拉取模式（Push or Pull）

Push好理解，比如在消费者端设置Listener回调；而Pull，控制权在于应用，即应用需要主动的调用拉消息方法从Broker获取消息，这里面存在一个消费位置记录的问题（如果不记录，会导致消息重复消费）。

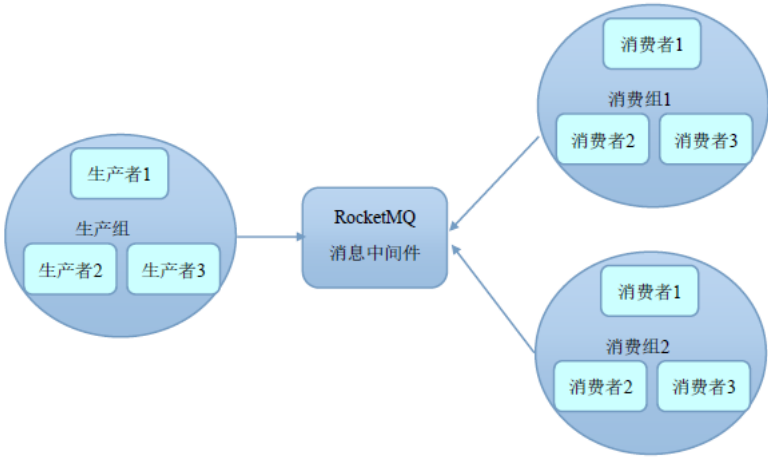
5.在Metaq1.x/2.x的版本中，分布式协调采用的是Zookeeper，而RocketMQ自己实现了一个NameServer，更加轻量级，性能更好！

6.消息失败重试机制、高效的订阅者水平扩展能力、强大的API、事务机制等等（后续详细介绍）

初步理解Producer/Consumer Group

ActiveMQ中并没有Group这个概念，而在RocketMQ中理解Group的机制很重要。





Group机制

想过没有，通过Group机制，让RocketMQ天然的支持消息负载均衡！

比如某个Topic有9条消息，其中一个Consumer Group有3个实例（3个进程 OR 3台机器），那么每个实例将均摊3条消息！（注意RocketMQ只有一种模式，即发布订阅模式。）

install RocketMQ

RocketMQ的Broker集群部署模式还挺多的，比如单Master模式、多Master模式、多Master多Slave模式（异步复制）、多Master多Slave模式（同步双写）等。明确个概念，RocketMQ Slave不可以写，可以读，类似于MySQL的主从机制。

单Master模式：

无需多言，一旦单个broker重启或宕机，一切都结束了！很显然，线上不可以使用。

多Master模式：

全是Master，没有Slave。当然，一个broker宕机了，应用是无影响的，缺点在于宕机的Master上未被消费的消息在Master没有恢复之前不可以订阅。

多Master多Slave模式（异步复制）：

多对Master-Slave，高可用！采用异步复制的方式，主备之间短暂延迟，MS级别。Master宕机，消费者可以从Slave上进行消费，不受影响，但是Master的宕机，会导致丢失掉极少量的消息。

多Master多Slave模式（同步双写）：

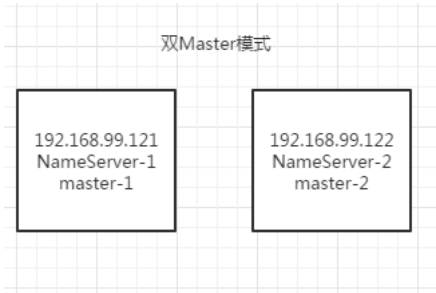
和上面的区别点在于采用的是同步方式，也就是在Master/Slave都写成功的前提下，向应用返回成功，可见不论是数据，还是服务都没有单点，都非常可靠！缺点在于同步的性能比异步稍低。

+

🔖

🔗

这里我将采用2个Master的方式进行搭建演示，会了双Master，其他的将很简单。（多Master在实际中也是非常常用的，如果并发非常大，考虑多Master多Slave模式）



双Master模式架构

在192.168.99.121/122机器上各一个NameServer、Master进程。

以192.168.99.121为例：

第一步，修改/etc/hosts文件

```
1 CentOS-192.168.99.121 x
2 CentOS-192.168.99.122 x

127.0.0.1 localhost localhost.localdomain localhost4 localhost6
::1 localhost localhost.localdomain localhost6
127.0.0.1 mydream121

#rocketmq配置
192.168.99.121 rocketmq-nameserver-1
192.168.99.121 rocketmq-master-1
192.168.99.122 rocketmq-nameserver-2
192.168.99.122 rocketmq-master-2
```

hosts配置

```
[root@mydream121 ~]# ping rocketmq-nameserver-2
PING rocketmq-nameserver-2 (192.168.99.122) 56(84) bytes of data.
64 bytes from rocketmq-nameserver-2 (192.168.99.122): icmp_seq=1 ttl=64 time=3.21 ms
64 bytes from rocketmq-nameserver-2 (192.168.99.122): icmp_seq=2 ttl=64 time=0.897 ms
64 bytes from rocketmq-nameserver-2 (192.168.99.122): icmp_seq=3 ttl=64 time=0.330 ms
^C
--- rocketmq-nameserver-2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2023ms
rtt min/avg/max/mdev = 0.330/1.482/3.219/1.240 ms
[root@mydream121 ~]# ping rocketmq-master-2
PING rocketmq-master-2 (192.168.99.122) 56(84) bytes of data.
64 bytes from rocketmq-nameserver-2 (192.168.99.122): icmp_seq=1 ttl=64 time=0.631 ms
64 bytes from rocketmq-nameserver-2 (192.168.99.122): icmp_seq=2 ttl=64 time=0.360 ms
^C
--- rocketmq-master-2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1967ms
rtt min/avg/max/mdev = 0.360/0.495/0.631/0.137 ms
[root@mydream121 ~]#
```

确保相互之间可以ping通

第二步，解压并创建存储路径

```
tar -xvf alibaba-rocketmq-3.2.6.tar.gz

mkdir -p alibaba-rocketmq/store/{commitlog,consumequeue,index}
```

第三步，配置文件

+

🔖

🔗

```
[root@mydream121 conf]# cd 2m-noslave/
[root@mydream121 2m-noslave]# ll
total 8
-rw-r--r--. 1 52583 users 148 Mar 28 2015 broker-a.properties
-rw-r--r--. 1 52583 users 148 Mar 28 2015 broker-b.properties
[root@mydream121 2m-noslave]# vim broker-a.properties
[root@mydream121 2m-noslave]# vim broker-b.properties
[root@mydream121 2m-noslave]#
```

broker-x.properties

```

#所属集群名字
brokerClusterName=rocketmq-cluster
#broker名字，注意此处不同的配置文件填写的不一样
brokerName=broker-a
#0 表示 Master, >0 表示 Slave
brokerId=0
#nameServer地址，分号分割
namesrvAddr=rocketmq-nameserver1:9876;rocketmq-nameserver2:9876
#在发送消息时，自动创建服务器不存在的topic，默认创建的队列数
defaultTopicQueueNums=4
#是否允许 Broker 自动创建Topic，建议线下开启，线上关闭
autoCreateTopicEnable=true
#是否允许 Broker 自动创建订阅组，建议线下开启，线上关闭
autoCreateSubscriptionGroup=true
#Broker 对外服务的监听端口
listenPort=10911
#删除文件时间点，默认凌晨 4点
deleteWhen=04
#文件保留时间，默认 48 小时
fileReservedTime=120
#commitLog每个文件的大小默认1G
mappedFileSizeCommitLog=1073741824
#ConsumeQueue每个文件默认存30W条，根据业务情况调整
mappedFileSizeConsumeQueue=300000
#destroyMappedFileIntervalForcibly=120000
#redeleteHangedFileInterval=120000
#检测物理文件磁盘空间
diskMaxUsedSpaceRatio=88
#存储路径
storePathRootDir=/software/alibaba-rocketmq/store
#commitLog 存储路径
storePathCommitLog=/software/alibaba-rocketmq/store/commitLog
#消费队列存储路径存储路径
storePathConsumeQueue=/software/alibaba-rocketmq/store/consumequeue
#消息索引存储路径
storePathIndex=/software/alibaba-rocketmq/store/index
#checkpoint 文件存储路径
storeCheckpoint=/software/alibaba-rocketmq/store/checkpoint
#abort 文件存储路径
abortFile=/software/alibaba-rocketmq/store/abort

```

配置项

上面已经将实际中常用的配置项给出来了！

第四步，修改日志配置文件

```
[root@mydream122 alibaba-rocketmq]# cd conf
[root@mydream122 conf]# ll
total 32
drwxr-xr-x. 2 52583 users 4096 Mar 28 2015 2m-2s-async
drwxr-xr-x. 2 52583 users 4096 Mar 28 2015 2m-2s-sync
drwxr-xr-x. 2 52583 users 4096 Apr 8 05:15 2m-noslave
-rw-r--r--. 1 52583 users 7786 Mar 28 2015 logback_broker.xml
-rw-r--r--. 1 52583 users 2331 Mar 28 2015 logback_filtersrv.xml
-rw-r--r--. 1 52583 users 2313 Mar 28 2015 logback_namesrv.xml
-rw-r--r--. 1 52583 users 2435 Mar 28 2015 logback_tools.xml
[root@mydream122 conf]#
```

注意到logback*.xml配置文件中：

+

□

⌘

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <appender name="DefaultAppender"
    class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${user.home}/logs/rocketmqlogs/broker_default.log</file>
    <append>true</append>
    <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
      <fileNamePattern>${user.home}/logs/rocketmqlogs/otherdays/broker_default.%i.log
    </fileNamePattern>
    <minIndex>1</minIndex>
    <maxIndex>5</maxIndex>
    </rollingPolicy>
    <triggeringPolicy
      class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
```

需要替换\${user.name}

可以使用sed进行替换：

```
sed -i 's/${user.home}/software/alibaba-rocketmq#g' *.xml
```

第五步，修改启动脚本中的JVM参数

```
#=====
# JVM Configuration
#=====
JAVA_OPT="${JAVA_OPT} -server -Xms1g -Xmx1g -Xmn512m -XX:PermSize=128m -XX:MaxPermSize=320m"
JAVA_OPT="${JAVA_OPT} -XX:+UseConcMarkSweepGC -XX:+UseCMSCompactAtFullCollection -XX:CMSInitiatingOccupancy
sUnloadingEnabled -XX:SurvivorRatio=8 -XX:+DisableExplicitGC"
JAVA_OPT="${JAVA_OPT} -verbose:gc -Xloggc:${HOME}/rmq_bk_gc.log -XX:+PrintGCDetails -XX:+PrintGCDateStamps"
JAVA_OPT="${JAVA_OPT} -XX:-OmitStackTraceInFastThrow"
JAVA_OPT="${JAVA_OPT} -Djava.ext.dirs=${BASE_DIR}/lib"
```

vim runbroker.sh/runserver.sh

注意，在这里我将JVM的堆的初始化和最大大小统一设置为1G，并将新生代大小设置为512M。主要是考虑到我的虚拟机内存，实际上在线上是可以走默认的4G堆内存的。

第六步，启动NameServer

```
nohup sh mqnamesrv &
```

```
[root@mydream122 ~]# jps
6482 NamesrvStartup
6513 Jps
[root@mydream122 ~]#
```

NameServer

```
[root@mydream121 rocketmqlogs]# pwd
/software/alibaba-rocketmq/logs/rocketmqlogs
[root@mydream121 rocketmqlogs]# more namesrv.log
2017-04-08 20:31:14 INFO main - rocketmqHome=/software/alibaba-rocketmq
2017-04-08 20:31:14 INFO main - kvConfigPath=/root/namesrv/kvConfig.json
2017-04-08 20:31:14 INFO main - listenPort=9876
2017-04-08 20:31:14 INFO main - serverWorkerThreads=8
2017-04-08 20:31:14 INFO main - serverCallbackExecutorThreads=0
2017-04-08 20:31:14 INFO main - serverSelectorThreads=3
2017-04-08 20:31:14 INFO main - serverOnewaySemaphoreValue=256
2017-04-08 20:31:14 INFO main - serverAsyncSemaphoreValue=64
2017-04-08 20:31:14 INFO main - serverChannelMaxIdleTimeSeconds=120
2017-04-08 20:31:14 INFO main - serverSocketSndBufSize=2048
2017-04-08 20:31:14 INFO main - serverSocketRcvBufSize=1024
2017-04-08 20:31:14 INFO main - serverPooledByteBufAllocatorEnable=false
2017-04-08 20:31:17 INFO main - The Name Server boot success.
2017-04-08 20:31:17 INFO NettyEventExecutor - NettyEventExecutor service started
2017-04-08 20:32:16 INFO NSScheduledThread1 - .....
2017-04-08 20:32:16 INFO NSScheduledThread1 - configTable SIZE: 0
[root@mydream121 rocketmqlogs]#
```

nameserver启动日志



第七步，启动broker-X

```
[root@mydream121 bin]# nohup sh mqbroker -c /software/alibaba-rocketmq/conf/2m-noslave/broker-a.properties >/dev/null 2>&1 &
[2] 6827
[root@mydream121 bin]# jps
6775 NamesrvStartup
6863 Jps
6834 BrokerStartup
[root@mydream121 bin]#
```

启动broker

注意观察日志：

```
2017-04-08 20:49:13 INFO main - rocketmqHome=/software/alibaba-rocketmq
2017-04-08 20:49:13 INFO main - namesrvAddr=rocketmq-nameserver-1:9876;rocketmq-nameserver-2:9876
2017-04-08 20:49:13 INFO main - brokerIP1=192.168.99.121
2017-04-08 20:49:13 INFO main - brokerIP2=192.168.99.121
2017-04-08 20:49:13 INFO main - brokerName=broker-a
2017-04-08 20:49:13 INFO main - brokerClusterName=rocketmq-cluster
2017-04-08 20:49:13 INFO main - brokerId=0
2017-04-08 20:49:13 INFO main - brokerPermission=6
2017-04-08 20:49:13 INFO main - defaultTopicQueueNums=4
2017-04-08 20:49:13 INFO main - autoCreateTopicEnable=true
2017-04-08 20:49:13 INFO main - clusterTopicEnable=true
2017-04-08 20:49:13 INFO main - brokerTopicEnable=true
2017-04-08 20:49:13 INFO main - autoCreateSubscriptionGroup=true
2017-04-08 20:49:13 INFO main - sendMessageThreadPoolNums=20
2017-04-08 20:49:13 INFO main - pullMessageThreadPoolNums=18
2017-04-08 20:49:13 INFO main - adminBrokerThreadPoolNums=16
2017-04-08 20:49:13 INFO main - clientManageThreadPoolNums=16
2017-04-08 20:49:13 INFO main - flushConsumerOffsetInterval=5000
2017-04-08 20:49:13 INFO main - flushConsumerOffsetHistoryInterval=60000
2017-04-08 20:49:13 INFO main - rejectTransactionMessage=false
2017-04-08 20:49:13 INFO main - fetchNamesrvAddrByAddressServer=false
2017-04-08 20:49:13 INFO main - sendThreadPoolQueueCapacity=100000
2017-04-08 20:49:13 INFO main - pullThreadPoolQueueCapacity=100000
2017-04-08 20:49:13 INFO main - filterServerNums=0
2017-04-08 20:49:13 INFO main - longPollingEnable=true
2017-04-08 20:49:13 INFO main - shortPollingTimeMills=1000
2017-04-08 20:49:13 INFO main - notifyConsumerIdsChangedEnable=true
```

broker.log

```
[root@mydream121 rocketmqlogs]# jps
6775 NamesrvStartup
6913 Jps
6834 BrokerStartup
[root@mydream121 rocketmqlogs]# netstat -tnlup | grep -E '6775|6834'
tcp        0      0 0.0.0.0:9876          0.0.0.0:*             LISTEN      6775/java
tcp        0      0 0.0.0.0:10911        0.0.0.0:*             LISTEN      6834/java
tcp        0      0 0.0.0.0:10912        0.0.0.0:*             LISTEN      6834/java
[root@mydream121 rocketmqlogs]# ^C
```

进程与端口

第八步：RocketMQ Console

把rocketmq-console.war部署到Tomcat下即可。

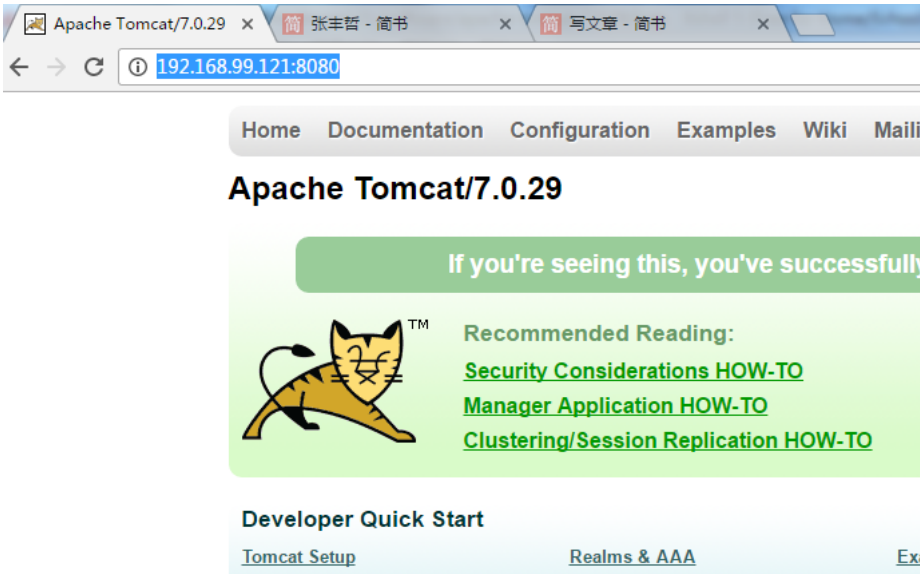
```
[root@mydream121 webapps]# pwd
/software/apache-tomcat-7.0.29/webapps
[root@mydream121 webapps]# unzip rocketmq-console.war -d rocketmq-console
```

解压WAR包

```
rocketmq.namesrv.addr=192.168.99.121:9876;192.168.99.122:9876
throwDone=true
```

在解压WAR包后的CLASS下更改config.properties





好久不见，TOM猫

192.168.99.121:8080/rocketmq-console/cluster/list.do									
RocketMQ Console Cluster Topic Connection Nameserv Message Broker Offset Consumer									
#Cluster Name	#Broker Detail								
rocketmq-cluster	#Broker Name	#Broker Instance							
	broker-a	#BID	#Addr	#Version	#InTPS	#OutTPS	#InTotalYest	#OutTotalYest	#InTotalToday
		0	192.168.99.121:10911	V3_2_6	0.0	0.0	0	0	0
	broker-b	#BID	#Addr	#Version	#InTPS	#OutTPS	#InTotalYest	#OutTotalYest	#InTotalToday
		0	192.168.99.122:10911	V3_2_6	0.0	0.0	0	0	0

rocketmq管控台

这个管控台实际上还是比较简陋的，我们使用比较多的是mqadmin操作命令，后续会介绍。

OK，到这里，双Master的搭建已经完成了！

本篇博客到此为止，下期再见，晚安！

📖 日记本 (/nb/10261827)

📄 举报文章 © 著作权归作者所有



张丰哲 (/u/cb569cce501b) ♂

写了 63893 字，被 2144 人关注，获得了 1674 个喜欢

(/u/cb569cce501b)

✓ 已关注

资深Java工程师 51CTO博客【2014-2016】：http://zhangfengzhe.blog.51cto.com/

👍 喜欢 | 110

🗨️

🐱

🖼️

更多分享

+








📄

🔗

(http://cwb.assets.jianshu.io/notes/images/1110137)

被以下专题收入，发现更多相似内容

+ 收入我的专题

-  程序员 (/c/NEt52a?utm_source=desktop&utm_medium=notes-included-collection)
-  今日看点 (/c/3sT4qY?utm_source=desktop&utm_medium=notes-included-collection)
-  javaWeb学习 (/c/73d938175cf5?utm_source=desktop&utm_medium=notes-included-collection)
-  简书收藏 -... (/c/4cd4d54b005a?utm_source=desktop&utm_medium=notes-included-collection)
-  Java学习笔记 (/c/04cb7410c597?utm_source=desktop&utm_medium=notes-included-collection)
-  Java 杂谈 (/c/0b39448c4e08?utm_source=desktop&utm_medium=notes-included-collection)
-  RocketMQ (/c/8cfe32491344?utm_source=desktop&utm_medium=notes-included-collection)
- 展开更多 ▾

+

🔖

🔗