



链滴

rocketmq 服务部署

作者: [gitsilence](#)

原文链接: <https://ld246.com/article/1632123166220>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

摘自: <https://zhuanlan.zhihu.com/p/226173170>

rocketmq源码构建

系统: Mac OS 12.0 --- arm64

jdk版本: 1.8

maven版本: 3.6.3

```
git clone https://github.com/apache/rocketmq.git
```

```
cd rocketmq
```

执行mvn构建命令, 这里的命令含义是: 清理之前编译、打包的文件, 激活profile 名为 release-al

然后打包, 不编译单元测试, 也不运行单元测试

```
mvn clean -P release-all package -Dmaven.test.skip=true
```

编译后进入到 **distribution/target** 目录, **rocket-mq-xxx.tar.gz** 和 **rocket-mq-xxx.zip** 两种, zip 是windows下的, tar.gz为Linux下的;

tar.gz解压后的目录结构:

```
.
├── LICENSE
├── NOTICE
├── README.md
├── benchmark
│   ├── batchproducer.sh
│   ├── consumer.sh
│   ├── producer.sh
│   ├── runclass.sh
│   ├── shutdown.sh
│   └── tproducer.sh
├── bin
│   ├── README.md
│   ├── cachedog.sh
│   ├── cleancache.sh
│   ├── cleancache.v1.sh
│   ├── dledger
│   │   └── fast-try.sh
│   ├── mqadmin
│   ├── mqadmin.cmd
│   ├── mqbroker
│   ├── mqbroker.cmd
│   ├── mqbroker.numanode0
│   ├── mqbroker.numanode1
│   ├── mqbroker.numanode2
│   ├── mqbroker.numanode3
│   ├── mqnamesrv
│   ├── mqnamesrv.cmd
│   ├── mqshutdown
│   └── mqshutdown.cmd
```

```
|— os.sh
|— play.cmd
|— play.sh
|— runbroker.cmd
|— runbroker.sh
|— runserver.cmd
|— runserver.sh
|— setcache.sh
|— startfsrv.sh
|— tools.cmd
|— tools.sh
|— conf
|   |— 2m-2s-async
|   |   |— broker-a-s.properties
|   |   |— broker-a.properties
|   |   |— broker-b-s.properties
|   |   |— broker-b.properties
|   |— 2m-2s-sync
|   |   |— broker-a-s.properties
|   |   |— broker-a.properties
|   |   |— broker-b-s.properties
|   |   |— broker-b.properties
|   |— 2m-noslave
|   |   |— broker-a.properties
|   |   |— broker-b.properties
|   |   |— broker-trace.properties
|   |— broker.conf
|   |— dledger
|   |   |— broker-n0.conf
|   |   |— broker-n1.conf
|   |   |— broker-n2.conf
|   |— logback_broker.xml
|   |— logback_namesrv.xml
|   |— logback_tools.xml
|   |— plain_acl.yml
|   |— tools.yml
|— lib
|   |— annotations-13.0.jar
|   |— commons-beanutils-1.9.4.jar
|   |— commons-cli-1.2.jar
|   |— commons-codec-1.9.jar
|   |— commons-collections-3.2.2.jar
|   |— commons-digester-2.1.jar
|   |— commons-lang3-3.4.jar
|   |— commons-logging-1.2.jar
|   |— commons-validator-1.7.jar
|   |— dledger-0.2.2.jar
|   |— fastjson-1.2.76.jar
|   |— gson-2.8.6.jar
|   |— guava-19.0.jar
|   |— jaeger-thrift-1.6.0.jar
|   |— jaeger-tracerresolver-1.6.0.jar
|   |— javassist-3.20.0-GA.jar
|   |— javax.annotation-api-1.3.2.jar
```

- jcommander-1.72.jar
- jna-4.2.2.jar
- kotlin-stdlib-1.4.10.jar
- kotlin-stdlib-common-1.4.0.jar
- libthrift-0.14.1.jar
- logback-classic-1.0.13.jar
- logback-core-1.0.13.jar
- netty-all-4.1.65.Final.jar
- okhttp-4.9.0.jar
- okio-2.8.0.jar
- openmessaging-api-0.3.1-alpha.jar
- opentracing-noop-0.33.0.jar
- opentracing-tracerresolver-0.1.8.jar
- opentracing-util-0.33.0.jar
- rocketmq-acl-4.9.2-SNAPSHOT.jar
- rocketmq-broker-4.9.2-SNAPSHOT.jar
- rocketmq-client-4.9.2-SNAPSHOT.jar
- rocketmq-common-4.9.2-SNAPSHOT.jar
- rocketmq-example-4.9.2-SNAPSHOT.jar
- rocketmq-filter-4.9.2-SNAPSHOT.jar
- rocketmq-logging-4.9.2-SNAPSHOT.jar
- rocketmq-namesrv-4.9.2-SNAPSHOT.jar
- rocketmq-openmessaging-4.9.2-SNAPSHOT.jar
- rocketmq-remoting-4.9.2-SNAPSHOT.jar
- rocketmq-srvutil-4.9.2-SNAPSHOT.jar
- rocketmq-store-4.9.2-SNAPSHOT.jar
- rocketmq-tools-4.9.2-SNAPSHOT.jar
- slf4j-api-1.7.7.jar
- snakeyaml-1.19.jar
- tomcat-annotations-api-8.5.46.jar
- tomcat-embed-core-8.5.46.jar

- bin目录下放的是脚本文件
- conf目录，配置文件

2m-2s-async 双主双从异步复制模式

2m-2s-sync 双主双从同步双写模式

2m-noslave 双主模式

- lib 存放的依赖jar包

测试namesrv的启动

启动命令：

`nohup /Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqnamesrv &`

使用 `jps` 查看是否启动成功

关闭命令

`/Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqshutdown namesrv`

集群部署

双主双从异步复制模式

两个台电脑

- Mac OS
 - 地址: 192.168.166.20
- Cent OS
 - 地址: 192.168.166.206

保证两台电脑在同一网段, 端口是通的, 可以关闭防火墙

需要开放的端口: NameServer的9876端口、Broker的10911、11011、10909、11009端口

Mac OS rocketmq配置文件

1、编辑 `conf/2m-2s-async/broker-a.properties`

```
# 集群名称
brokerClusterName=mq-broker-cluster
# broker名字, 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
# 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=48

# Broker的角色, ASYNC_MASTER=异步复制master, SYNC_MASTER=同步双写master, SLAVE=slave节点
brokerRole=ASYNC_MASTER

# 刷盘方式, ASYNC_FLUSH=异步刷盘, SYNC_FLUSH=同步刷盘
flushDiskType=ASYNC_FLUSH
# broker对外服务的监听端口
listenPort=10911
# nameServer地址, 如果name server是多台集群的话, 就用分号分隔
namesrvAddr=192.168.166.20:9876;192.168.166.206:9876

# 每个topic对应队列的数量, 默认为4, 实际参考consumer实例的数量, 值过小不利于consumer负载均衡
defaultTopicQueueNums=8
# 是否允许broker自动创建Topic, 生产建议关闭
autoCreateTopicEnable=true
# 是否允许broker自动创建订阅组, 生产建议关闭
autoCreateSubscriptionGroup=true
# 设置brokerIp
brokerIp=192.168.166.20
```

```
# 存储路径
storePathRootDir=/Users/gitsilence/JavaTools/rocketmq/data/store-a
# commitLog 存储路径
storePathCommitLog=/Users/gitsilence/JavaTools/rocketmq/data/store-a/commitlog
# 消费队列存储路径
storePathConsumerQueue=/Users/gitsilence/JavaTools/rocketmq/data/store-a/consumequeue
# 消息索引存储路径
storePathIndex=/Users/gitsilence/JavaTools/rocketmq/data/store-a/index
# checkpoint 文件存储路径
storeCheckpoint=/Users/gitsilence/JavaTools/rocketmq/data/store-a/checkpoint
# abort 存储路径
abortFile=/Users/gitsilence/JavaTools/rocketmq/data/store-a/abort
# 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
```

2、配置slave节点 [conf/2m-2s-async/broker-b-s.properties](#)

```
# 集群名称
brokerClusterName=mq-broker-cluster
# broker名字，不同的配置文件填写的不一样
brokerName=broker-b
# 0 表示master， >0 表示slave
brokerId=1
# 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=48
# Broker的角色，AYNSC_MASTER=异步复制master，SYNC_MASTER=同步双写master，SLAVE=slave节点
brokerRole=SLAVE
# 刷盘方式，ASYNC_FLUSH=异步刷盘，SYNC_FLUSH=同步刷盘
flushDiskType=ASYNC_FLUSH
# broker对外服务的监听端口
listenPort=11011
# nameServer地址，如果name server是多台集群的话，就用分号分隔
namesrvAddr=192.168.166.20:9876;192.168.166.206:9876

# 每个topic对应队列的数量，默认为4，实际参考consumer实例的数量，值过小不利于consumer负载均衡
defaultTopicQueueNums=8
# 是否允许broker自动创建Topic，生产建议关闭
autoCreateTopicEnable=true
# 是否允许broker自动创建订阅组，生产建议关闭
autoCreateSubscriptionGroup=true
# 设置brokerIp
brokerIp=192.168.166.20
# 存储路径
storePathRootDir=/Users/gitsilence/JavaTools/rocketmq/data/store-b
# commitLog 存储路径
storePathCommitLog=/Users/gitsilence/JavaTools/rocketmq/data/store-b/commitlog
# 消费队列存储路径
```

```
storePathConsumerQueue=/Users/gitsilence/JavaTools/rocketmq/data/store-b/consumequeue
# 消息索引存储路径
storePathIndex=/Users/gitsilence/JavaTools/rocketmq/data/store-b/index
# checkpoint 文件存储路径
storeCheckpoint=/Users/gitsilence/JavaTools/rocketmq/data/store-b/checkpoint
# abort 存储路径
abortFile=/Users/gitsilence/JavaTools/rocketmq/data/store-b/abort
# 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
```

先把两个节点 name server 启动

```
nohup /Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqnamesrv >>
/Users/gitsilence/JavaTools/rocketmq/logs/namesrv/namesrv.log &
```

linux节点启动

```
nohup /root/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqnamesrv >> /root/rocketmq/logs
namesrv/namesrv.log &
```

Linux 环境下遇到的问题：

- 启动的时候设置内存太大，只是用来测试的话，我们可以改下启动脚本的内存配置， [runserver.sh](#)
- 还有一个就是脚本格式问题，会导致脚本执行不了，解决办法： https://blog.lacknb.cn/articles/019/10/10/1577974148936.html#toc_h2_18

nameserver启动完毕后，可以执行 `jps`，查看是否起来

Cent OS rocketmq配置文件

1、配置master节点 `conf/2m-2s-async/broker-b.properties`

```
# 集群名称
brokerClusterName=mq-broker-cluster
# broker名字，不同的配置文件填写的不一样
brokerName=broker-b
# 0 表示master， >0 表示slave
brokerId=0
# 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=48
# Broker的角色，ASYNC_MASTER=异步复制master，SYNC_MASTER=同步双写master，SLAVE=
lave节点
brokerRole=ASYNC_MASTER
# 刷盘方式，ASYNC_FLUSH=异步刷盘，SYNC_FLUSH=同步刷盘
flushDiskType=ASYNC_FLUSH
# broker对外服务的监听端口
listenPort=10911
# nameServer地址，如果name server是多台集群的话，就用分号分隔
```

```
namesrvAddr=192.168.166.20:9876;192.168.166.206:9876
```

```
# 每个topic对应队列的数量，默认为4，实际参考consumer实例的数量，值过小不利于consumer负载均衡
```

```
defaultTopicQueueNums=8
```

```
# 是否允许broker自动创建Topic，生产建议关闭
```

```
autoCreateTopicEnable=true
```

```
# 是否允许broker自动创建订阅组，生产建议关闭
```

```
autoCreateSubscriptionGroup=true
```

```
# 设置brokerIp
```

```
brokerIP1=192.168.166.206
```

```
# 存储路径
```

```
storePathRootDir=/root/rocketmq/data/store-a
```

```
# commitLog 存储路径
```

```
storePathCommitLog=/root/rocketmq/data/store-a/commitlog
```

```
# 消费队列存储路径
```

```
storePathConsumerQueue=/root/rocketmq/data/store-a/consumequeue
```

```
# 消息索引存储路径
```

```
storePathIndex=/root/rocketmq/data/store-a/index
```

```
# checkpoint 文件存储路径
```

```
storeCheckpoint=/root/rocketmq/data/store-a/checkpoint
```

```
# abort 存储路径
```

```
abortFile=/root/rocketmq/data/store-a/abort
```

```
# 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
```

2、配置 slave 节点, [conf/2m-2s-async/broker-a-s.properties](#)

```
# 集群名称
```

```
brokerClusterName=mq-broker-cluster
```

```
# broker名字，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=1
```

```
# 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=48
```

```
# Broker的角色, AYNCS_MASTER=异步复制master, SYNC_MASTER=同步双写master, SLAVE=slave节点
```

```
brokerRole=SLAVE
```

```
# 刷盘方式, ASYNC_FLUSH=异步刷盘, SYNC_FLUSH=同步刷盘
```

```
flushDiskType=ASYNC_FLUSH
```

```
# broker对外服务的监听端口
```

```
listenPort=11011
```

```
# nameServer地址，如果name server是多台集群的话，就用分号分隔
```

```
namesrvAddr=192.168.166.20:9876;192.168.166.206:9876
```

```
# 每个topic对应队列的数量，默认为4，实际参考consumer实例的数量，值过小不利于consumer负载均衡
```

```
defaultTopicQueueNums=8
```

```
# 是否允许broker自动创建Topic，生产建议关闭
```



```
autoCreateTopicEnable=true
# 是否允许broker自动创建订阅组，生产建议关闭
autoCreateSubscriptionGroup=true
# 设置brokerIp
brokerIP1=192.168.166.206
# 存储路径
storePathRootDir=/root/rocketmq/data/store-b
# commitLog 存储路径
storePathCommitLog=/root/rocketmq/data/store-b/commitlog
# 消费队列存储路径
storePathConsumerQueue=/root/rocketmq/data/store-b/consumequeue
# 消息索引存储路径
storePathIndex=/root/rocketmq/data/store-b/index
# checkpoint 文件存储路径
storeCheckpoint=/root/rocketmq/data/store-b/checkpoint
# abort 存储路径
abortFile=/root/rocketmq/data/store-b/abort
# 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
```

启动broker

1、先启动mac os master节点 broker-a

```
nohup /Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqbroker -c /Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/conf/2m-2s-async/broker-a.properties >> /Users/gitsilence/JavaTools/rocketmq/logs/broker/master-broker-a.log &
```

2、启动 Linux 的 master节点 broker-b

```
nohup sh /root/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqbroker -c /root/rocketmq/rocketmq-4.9.2-SNAPSHOT/conf/2m-2s-async/broker-b.properties >> /root/rocketmq/logs/broker/master-broker-b.log &
```

3、启动linux的slave节点 broker-a

```
nohup sh /root/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqbroker -c /root/rocketmq/rocketmq-4.9.2-SNAPSHOT/conf/2m-2s-async/broker-a-s.properties >> /root/rocketmq/logs/broker/slave-broker-a.log &
```

4、启动mac os 的slave节点 broker-b

```
nohup /Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqbroker -c /Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/conf/2m-2s-async/broker-b-s.properties >> /Users/gitsilence/JavaTools/rocketmq/logs/broker/slave-broker-b.log &
```

启动完成后可以使用mqadmin检查 集群状态

```
/Users/gitsilence/JavaTools/rocketmq/rocketmq-4.9.2-SNAPSHOT/bin/mqadmin clusterList -n "192.168.166.20:9876;192.168.166.206:9876"
```

RocketMQLog:WARN No appenders could be found for logger (io.netty.util.internal.InternalT

readLocalMap).

RocketMQLog:WARN Please initialize the logger system properly.

#Cluster Name	#Broker Name	#BID	#Addr	#Version	#InTPS(LO
D)	#OutTPS(LOAD)	#PCWait(ms)	#Hour	#SPACE	
mq-broker-cluster	broker-a	0	192.168.166.20:10911	V4_9_1	0.00(0,0
s)	0.00(0,0ms)	0	453367.41	-1.0000	
mq-broker-cluster	broker-a	1	192.168.166.206:11011	V4_9_1	0.00(0,
ms)	0.00(0,0ms)	0	453367.41	-1.0000	
mq-broker-cluster	broker-b	0	192.168.166.206:10911	V4_9_1	0.00(0,
ms)	0.00(0,0ms)	0	453367.41	-1.0000	
mq-broker-cluster	broker-b	1	192.168.166.20:11011	V4_9_1	0.00(0,0
s)	0.00(0,0ms)	0	453367.41	-1.0000	

单机部署

部署rocketmq-console可视化监控

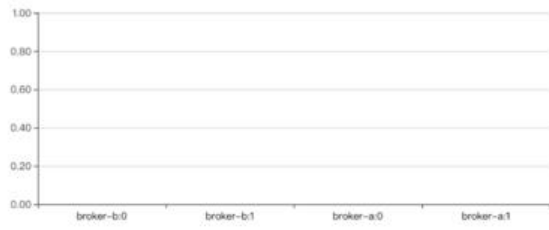
- 1、`git clone git@github.com:apache/rocketmq-externals.git`
- 2、`cd rocketmq-externals`
- 3、`git checkout release-rocketmq-console-1.0.0`
- 4、`cd rocketmq-console`
- 5、`mvn clean package -Dmaven.test.skip=true`
- 6、构建成功后，执行启动命令

```
nohup java -jar target/rocketmq-console-ng-1.0.0.jar --rocketmq.config.namesrvAddr='192.168.166.20:9876;192.168.166.206:9876' >> /Users/gitsilence/JavaTools/rocketmq/logs/mq-console/mq-console.log &
```

浏览器访问：<http://localhost:8080>

RocketMQ控制台											更换语言
集群											
mq-broker-cluster											
分片	编号	地址	版本	生产消息TPS	消费消息TPS	昨日生产总数	昨日消费总数	今天生产总数	今天消费总数	操作	
broker-b	0(master)	192.168.166.206:10911	V4_9_1	0.00	0.00	0	0	0	0	状态	配置
broker-b	1(slave)	192.168.166.20:11011	V4_9_1	0.00	0.00	0	0	0	0	状态	配置
broker-a	0(master)	192.168.166.20:10911	V4_9_1	0.00	0.00	0	0	0	0	状态	配置
broker-a	1(slave)	192.168.166.206:11011	V4_9_1	0.00	0.00	0	0	0	0	状态	配置

Broker TOP 10



日期:

Broker 5min trend



主题:

SCHEDULE_TOPIC_XXXX