GMQ在线扩容测试

针对版本V1.0.0

**©成都基础平台架构部**

2017/3/2

修订记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 修订内容 | 作者 | 审核 | 修订日期 |
| V1.0.0 | 初始版本 | 郜焱磊 | 成都基础平台架构部 | 2017/3/02 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目 录

[1 背景 5](#_Toc476215860)

[2 环境配置 5](#_Toc476215861)

[2.1 原集群节点 5](#_Toc476215862)

[2.2 新增集群节点 5](#_Toc476215863)

[3 方案选择 6](#_Toc476215864)

[4 测试目标 6](#_Toc476215865)

[5 扩容测试 6](#_Toc476215866)

[5.1 集群初始状态 6](#_Toc476215867)

[5.2 测试一 7](#_Toc476215868)

[5.2.1 测试场景 7](#_Toc476215869)

[5.2.2 扩容前 7](#_Toc476215870)

[5.2.3 扩容中 9](#_Toc476215871)

[5.2.4 发送完毕 12](#_Toc476215872)

[5.2.5 本次测试结论 14](#_Toc476215873)

[5.3 测试二 15](#_Toc476215874)

[5.3.1 测试场景 15](#_Toc476215875)

[5.3.2 扩容前 15](#_Toc476215876)

[5.3.3 扩容中 17](#_Toc476215877)

[5.3.4 发送完毕 19](#_Toc476215878)

[5.3.5 本次测试结论 21](#_Toc476215879)

[6 整体结论 21](#_Toc476215880)

# 背景

随着接入的团队项目越来越多，目前在线的gmq吞吐量已到达饱和，需对gmq进行在线水平扩容，提高gmq的吞吐量。

# 环境配置

新增4台，16核，32G的服务器，用于扩容。由于集群为了保证数据可靠性中有一主一从概念，所以新增4台服务器，对整个集群来讲，仅新增了2组节点。

## 原集群节点

|  |  |  |  |
| --- | --- | --- | --- |
| 节点名称 | 主机地址 | 端口 | 备注 |
| Broker- master1 | 10.128.31.104 | 10911 | 主 |
| Broker- master1 | 10.128.31.106 | 10911 | 从 |
| Broker- master2 | 10.128.31.105 | 10911 | 主 |
| Broker- master2 | 10.128.31.107 | 10911 | 从 |

## 新增集群节点

|  |  |  |  |
| --- | --- | --- | --- |
| 主机名 | 主机地址 | 端口 | 备注 |
| Broker- master3 | 10.128.31.223 | 10911 | 主 |
| Broker- master3 | 10.128.31.224 | 10911 | 从 |
| Broker- master4 | 10.128.31.225 | 10911 | 主 |
| Broker- master4 | 10.128.31.226 | 10911 | 从 |

# 方案选择

文档中方案三

# 测试目标

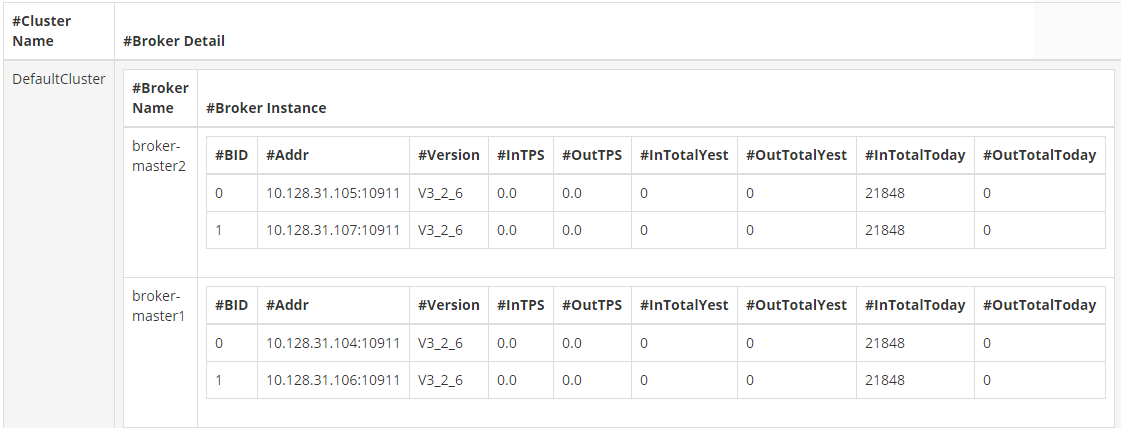
在前期测试中，在集群稳定的情况下，发送消息/消费消息不会出现消息丢失，重复消费的问题。所以本次测试不会针对扩容前/后，集群在稳定情况下进行测试。主要针对在扩容过程中、扩容后集群相对不稳定的情况下进行以下几点测试。

* 扩容后集群是否能正确在线。
* 扩容中新增接口正确性。
* 扩容中Topic队列是否自动更新到新的节点上。
* 扩容后发送msg是否会发送到新加的节点上的队列里。
* 扩容后整个集群队列的负载情况。
* 在发送消息时，同时进行扩容，是否会出现，程序卡死，程序异常，消息丢失的情况。
* 在消费过程时，同时进行扩容，是否会出现，程序卡死，程序异常，消息逐一消费，重复消费的等情况。

# 扩容测试

由于本地与服务器有网络延迟，tps不会很高，如果单独一个线程发送很多消息，耗时相当长。所以暂定每个线程发送少量消息，随着线程数目的增大，数据量增大。

## 集群初始状态



图：5-1-1

## 测试一

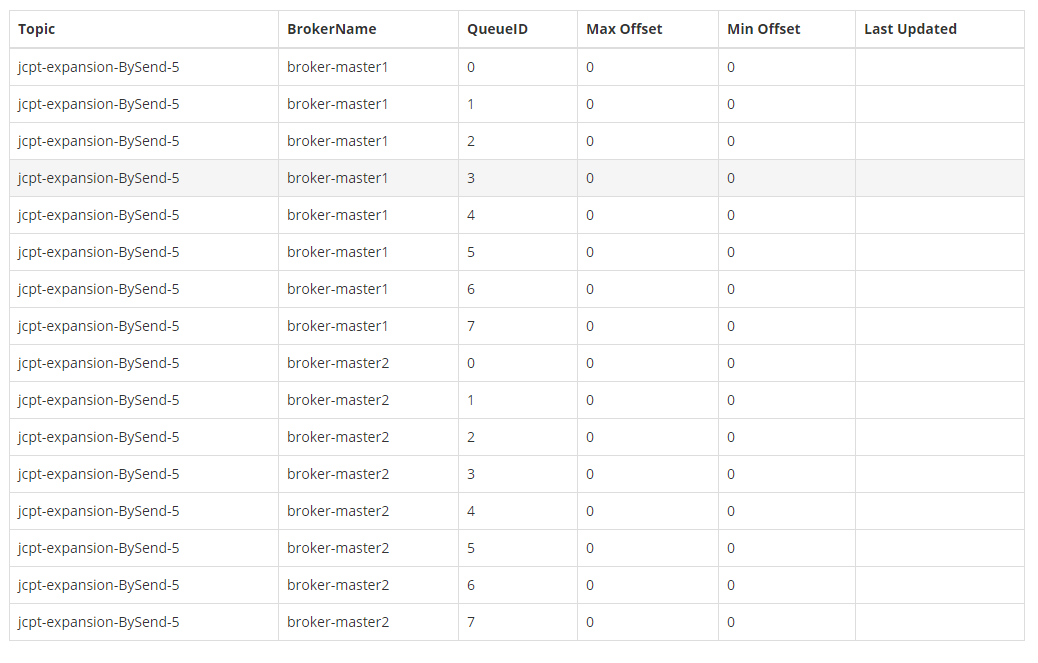
### 测试场景

一个生产者线程，发送2w条消息。

一个消费者线程

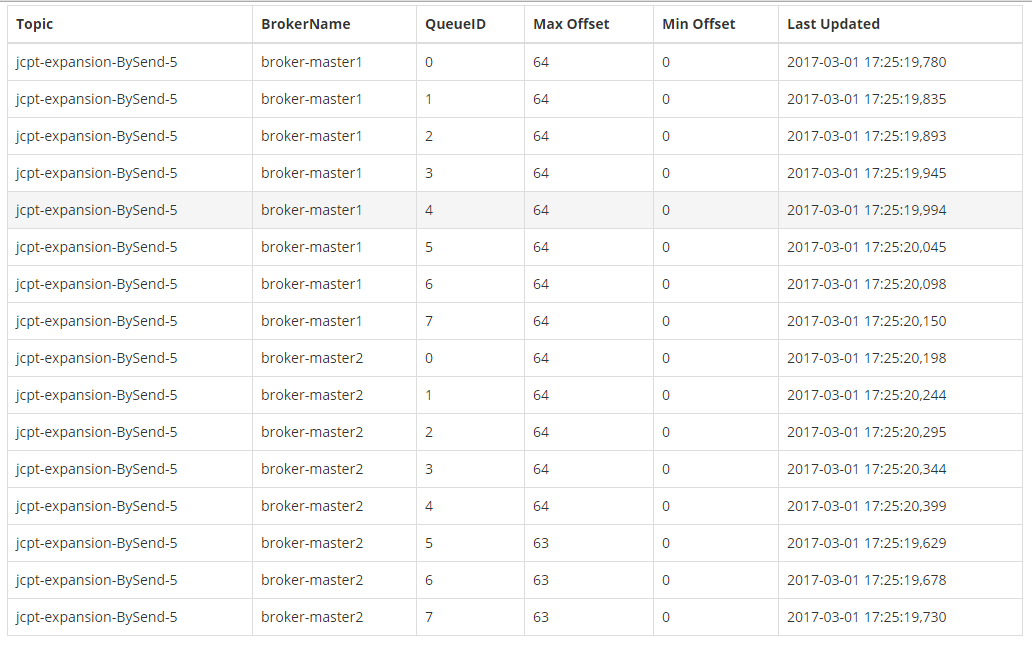
### 扩容前

新建topic jcpt-expansion-BySend-5 状态 。



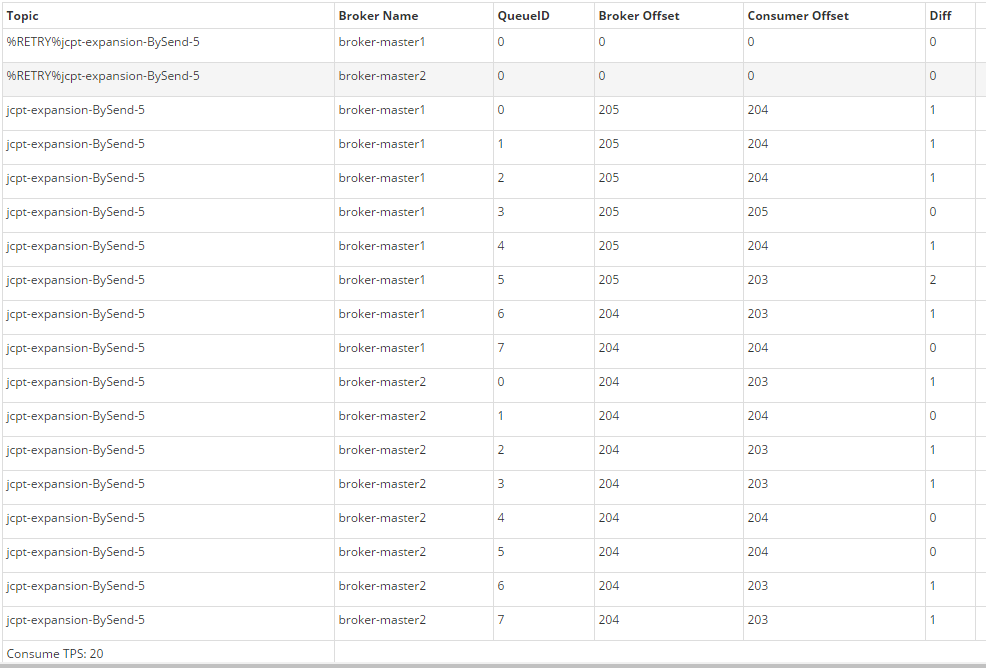
图：5-1-2

Producer发送数据正常



图：5-1-3

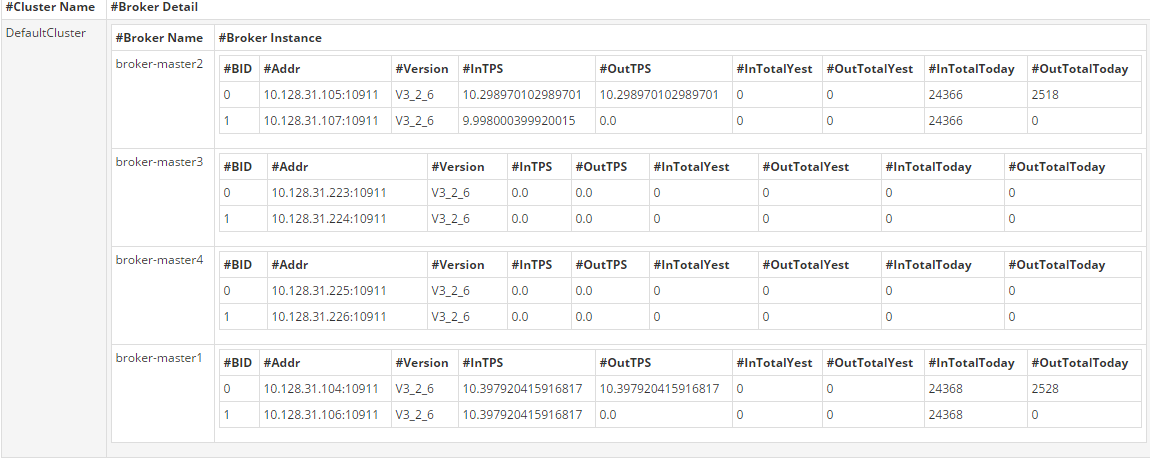
consumer消费数据



图：5-1-4

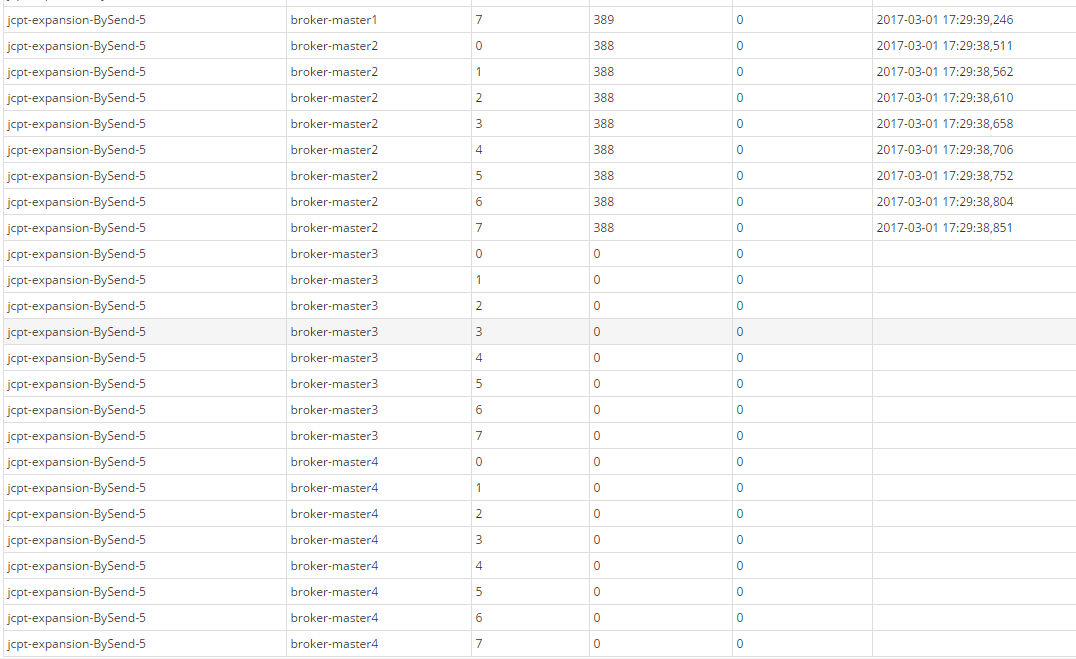
### 扩容中

集群变化



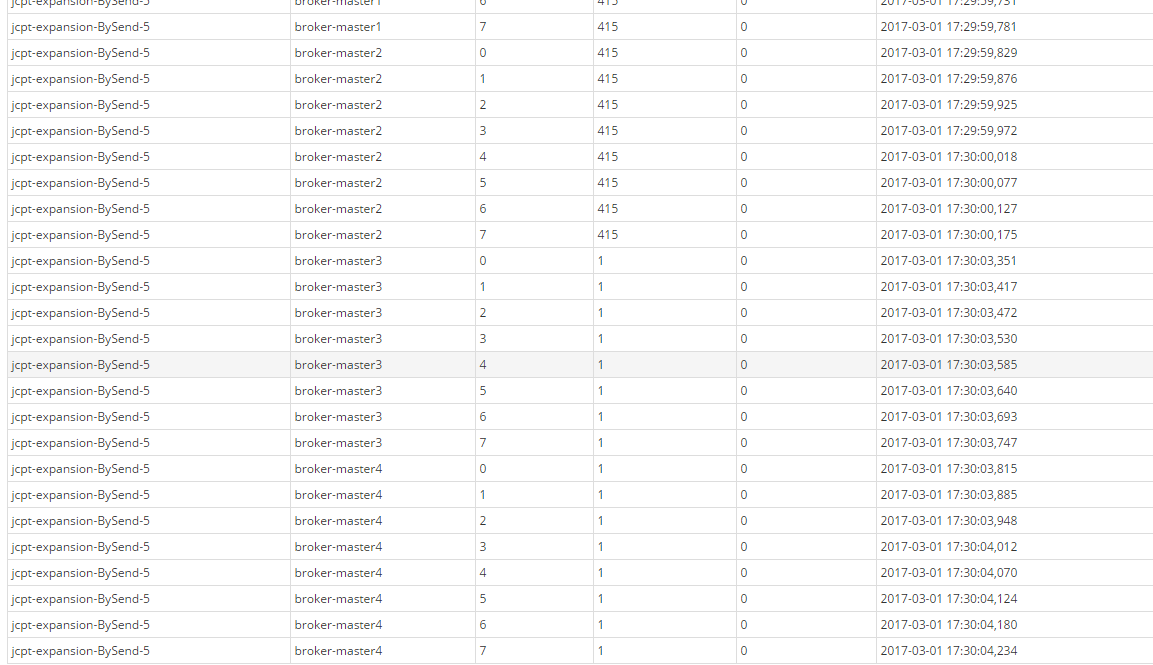
图：5-1-5

topic队列数变化



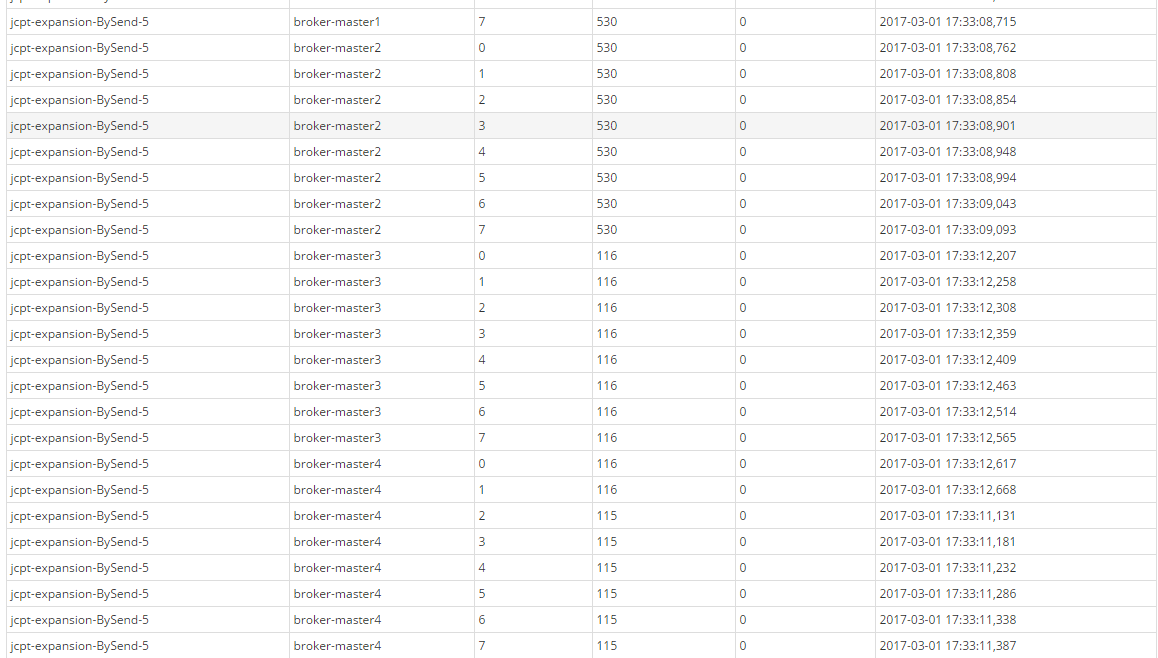
图：5-1-6

扩容后大约1分钟左右，发送消息



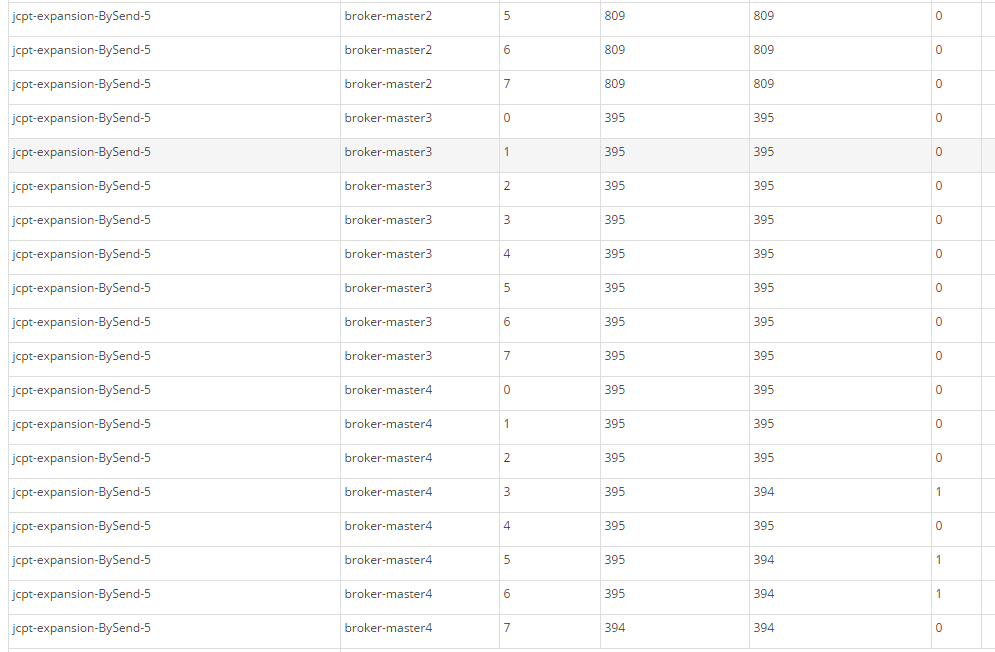
图：5-1-7

扩容后三分钟，发送消息



图：5-1-8

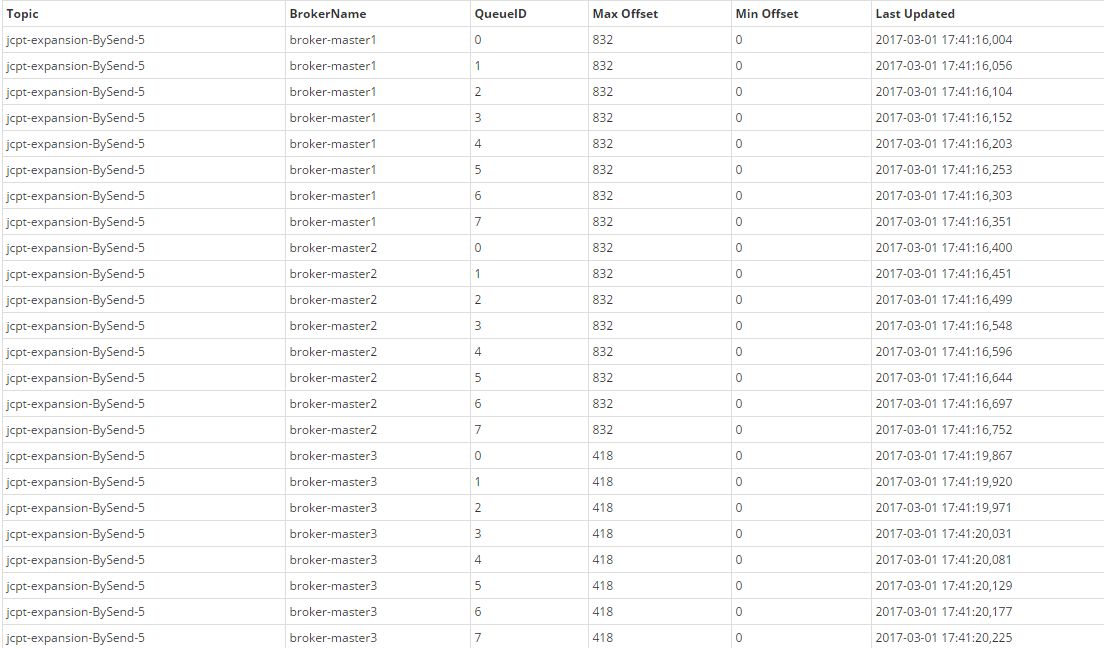
扩容后10分钟，消费情况

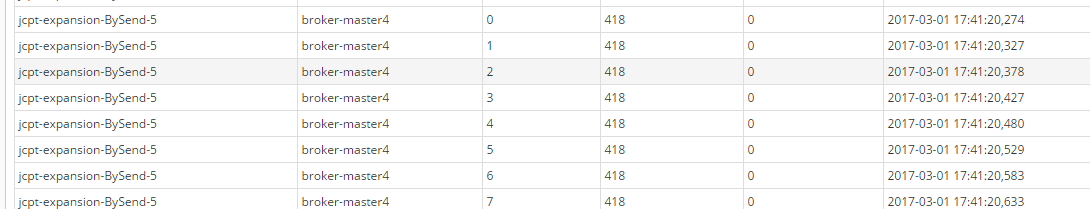


图：5-1-9

### 发送完毕

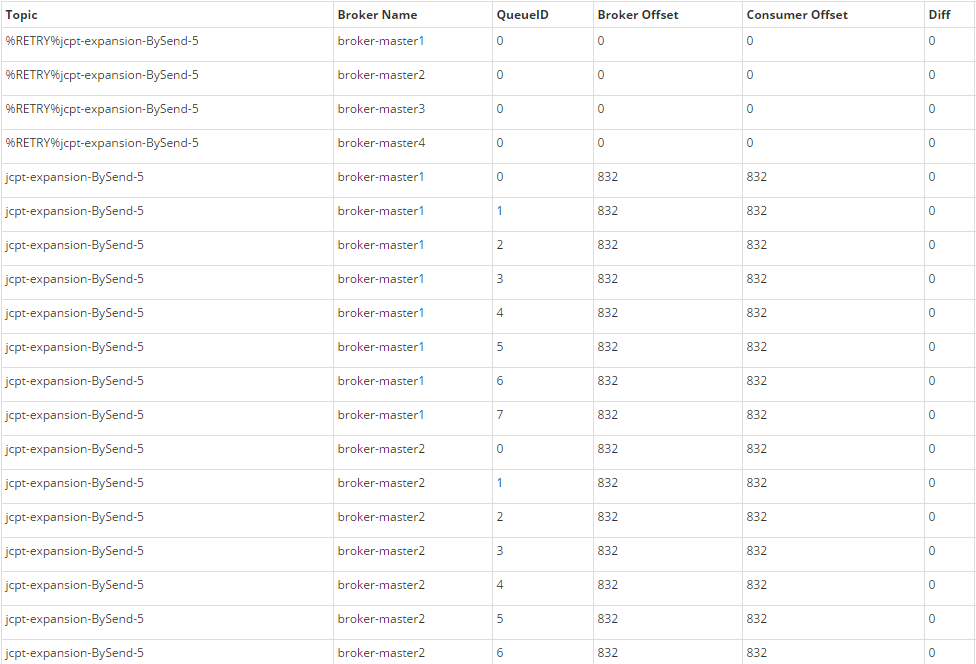
topic队列情况

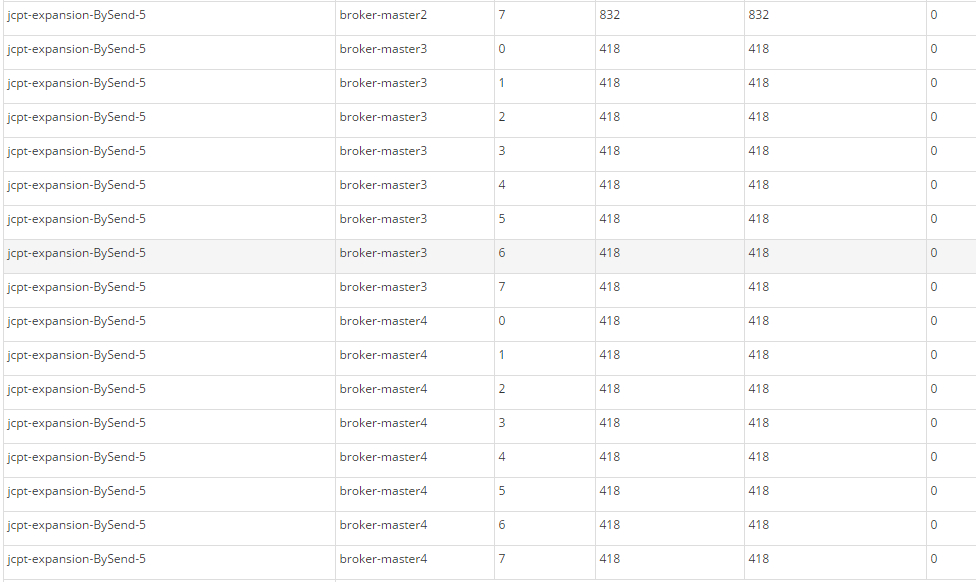




图：5-1-10

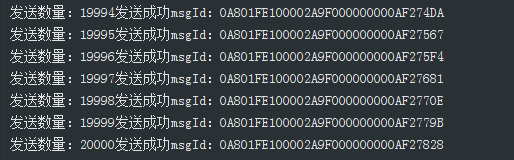
消费情况





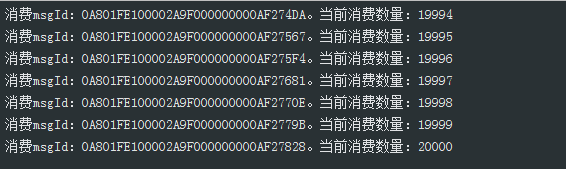
图：5-1-11

程序截图

Producer

图：5-1-12

Consumer



图：5-1-13

### 本次测试结论

* 通过本次测试1个线程发送2w条数据。

图5-1-12，5-1-13显示：看出在程序中发送数量与消费数量均为2w条。

图5-1-10显示：两个老节点信息每个队列832条消息，两个新节点每个队列数为418条消息，消息总数为832\*16+418\*16=13312+6688=20000

图5-1-11显示：消费BrokerOffset与ConsumerOffset一致，与图十中每个队列的消息数也一致。

**结论一：在扩容中，扩容后。发送/消费消息数目完全匹配，没有出现消息丢失，重复消费等情况，生成程序与消费程序都正常跑完，没有出现程序异常，中断等情况。**

* 图5-1-5与图5-1-1对比，可以看出扩容后集群节点新增两个。

**结论二：扩容后，集群数目正常，新加节点正常加入集群。**

* 图5-1-3与图5-1-6对比，可以看出扩容后，通过开发接口，topic队列会更新到新的节点上。

**结论三：开发接口可用于topic更新，topic会自动更新到新的节点上。**

* 图5-1-6中表示：扩容后新节点并不会马上写数据，大约持续1分钟左右的样子会有新数据，如图5-1-7。

图5-1-7中当新节点所有队列有1条消息时，老节点每个队列有415条数据。三分钟后如图5-1-8新节点每个队列为116条消息，老节点每个队列有530条消息。不难看出，新老节点队列所增加消息条数完全一致。

**结论四：扩容后，大约40秒，消息会负载到新的节点上。之后发送消息会负载到topic所有队列中。**

## 测试二

### 测试场景

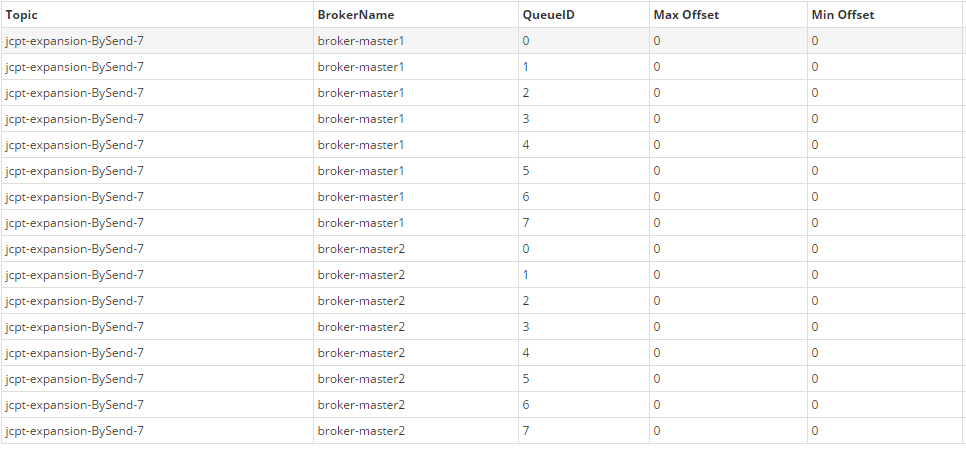
由于测试一数据较少，本次测试开500个线程发送，发送500万数据，进一步验证数据的准确性，扩容topic更新，负载等情况。

500个生成者线程

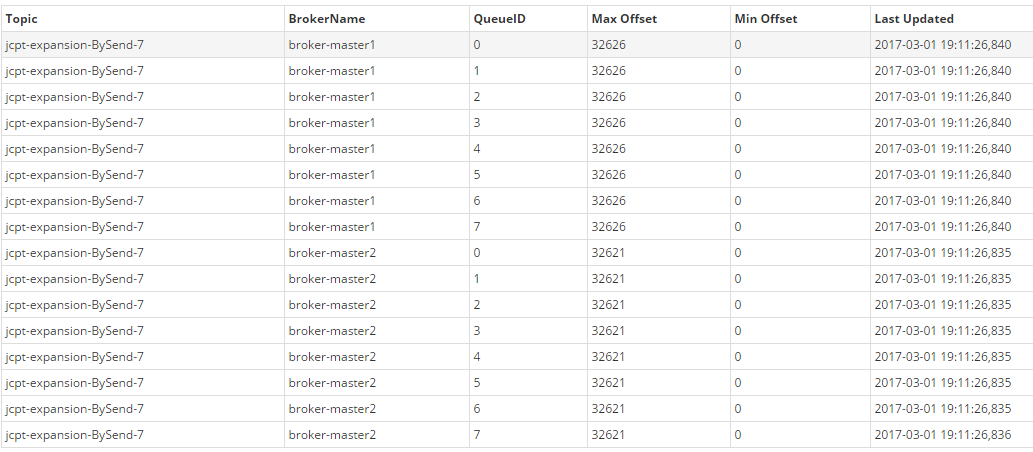
1个消费者线程

### 扩容前

新建topic jcpt-expansion-BySend-7 状态 。

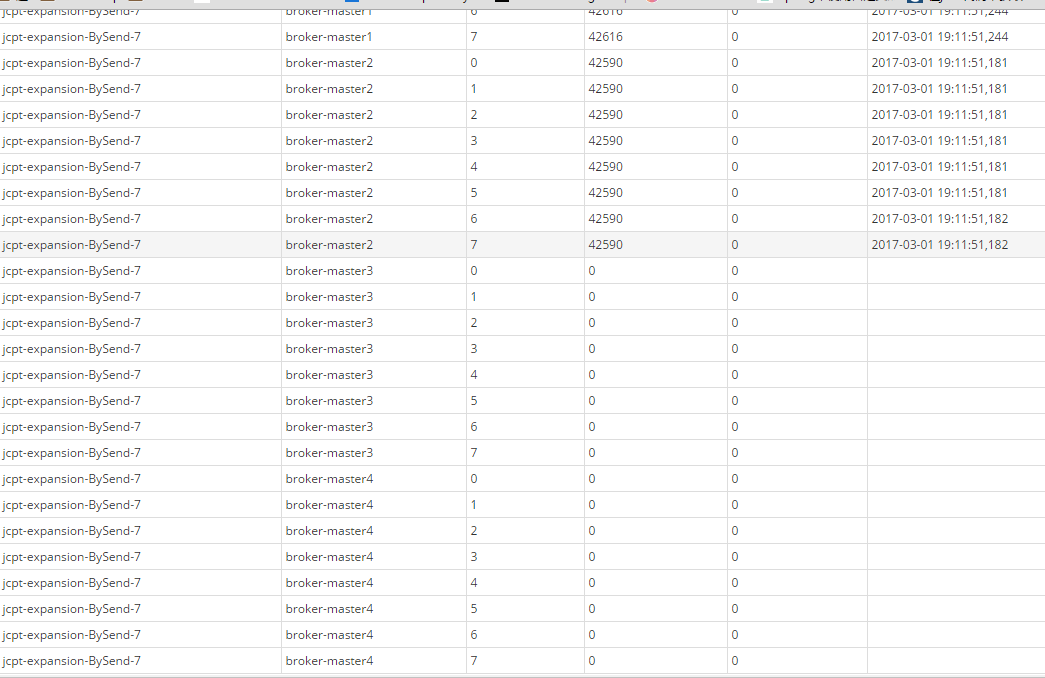
图：5-2-1

topic



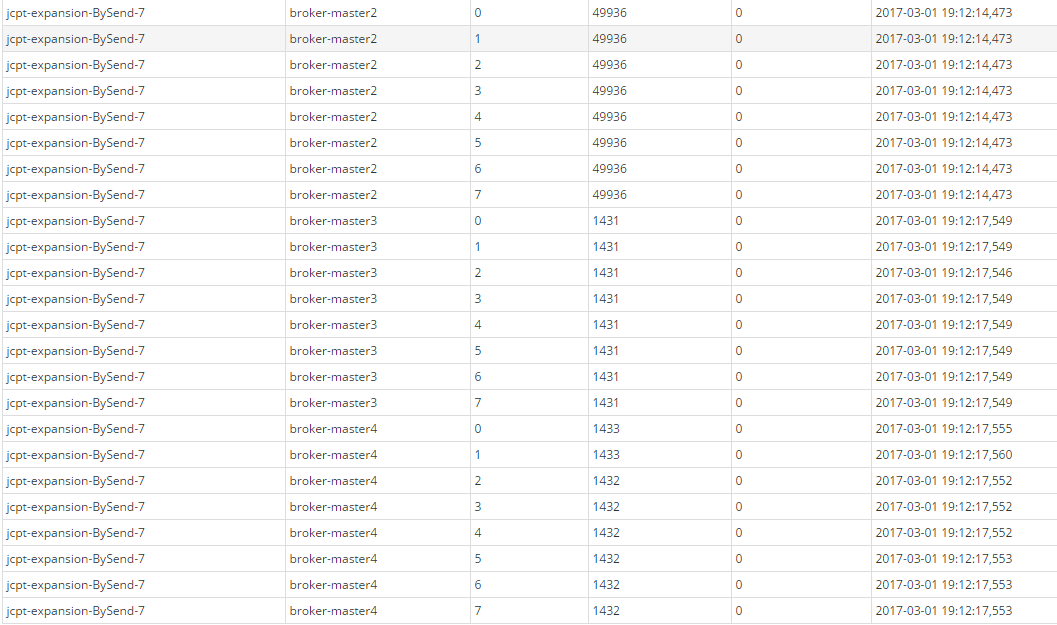
图：5-2-2

### 扩容中



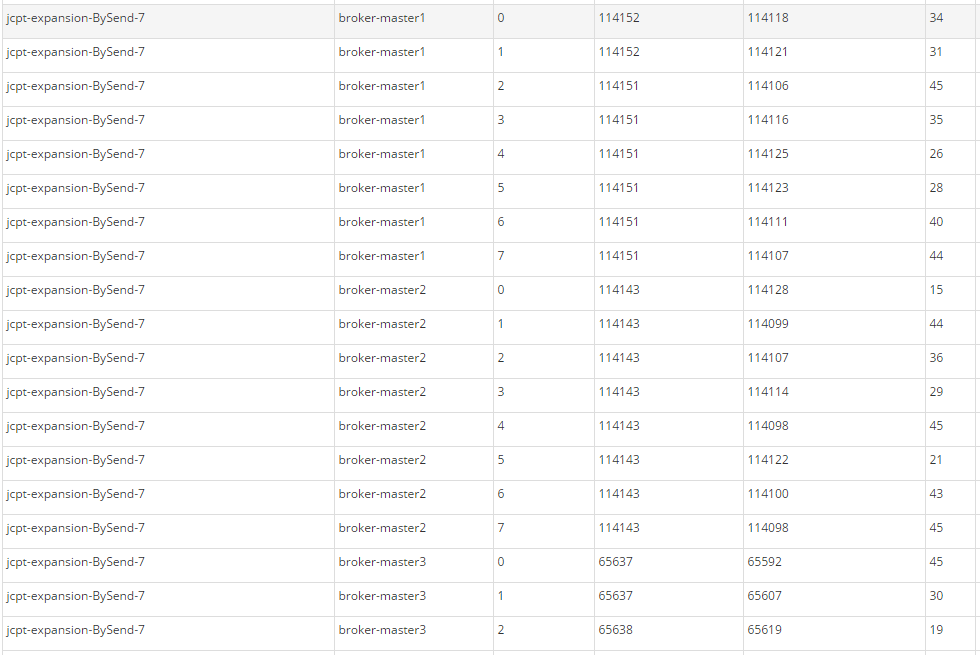
图：5-2-3

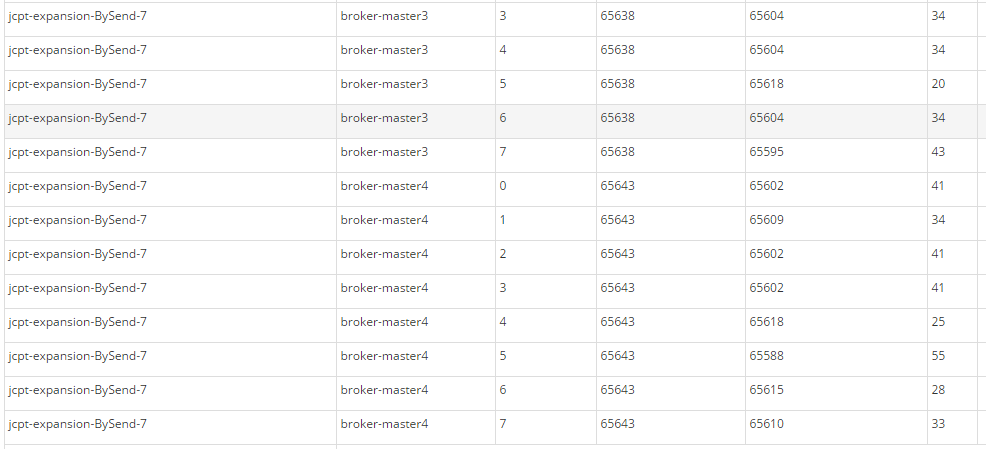
1分钟左右



图：5-2-4

消费情况

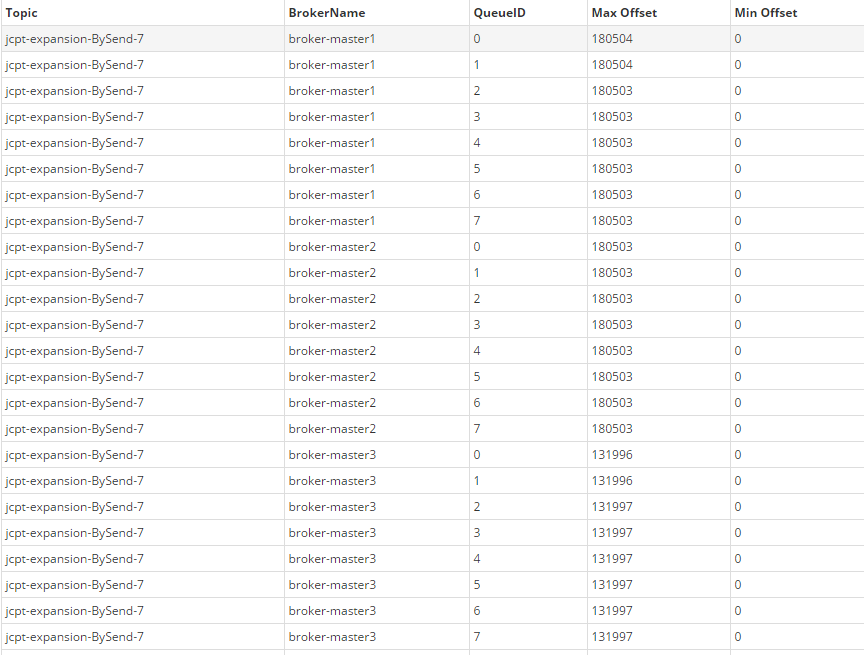


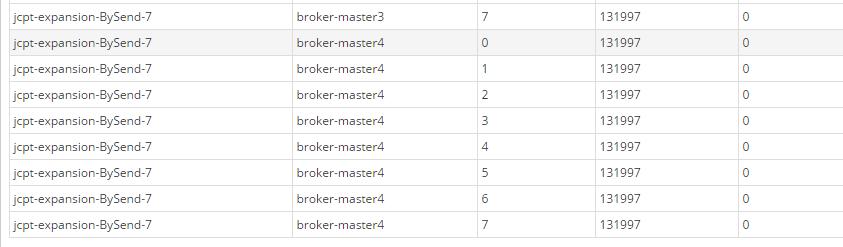


图：5-2-5

### 发送完毕

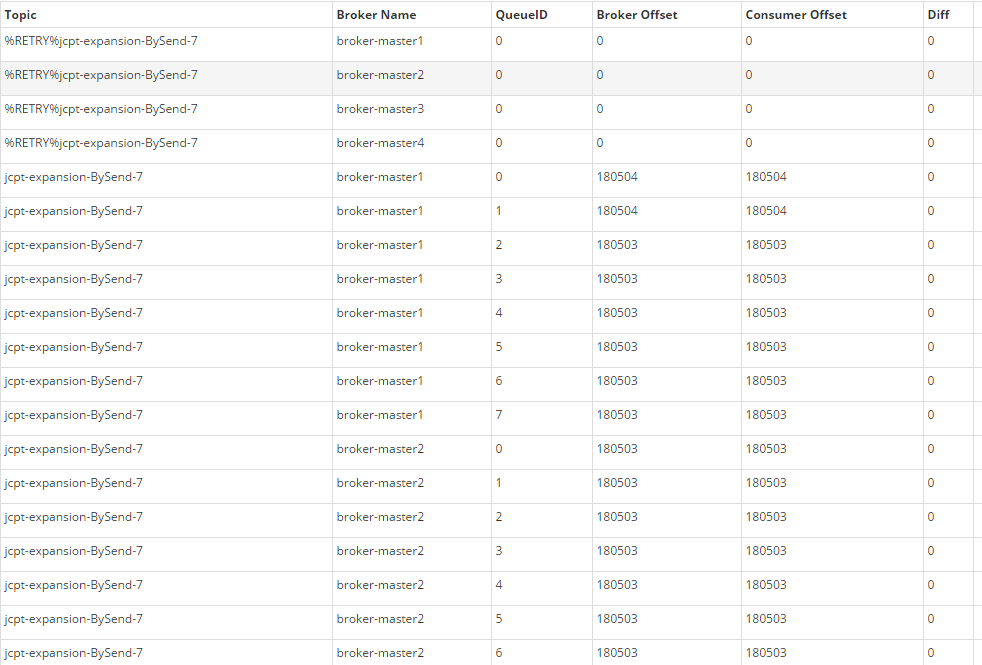
发送数量

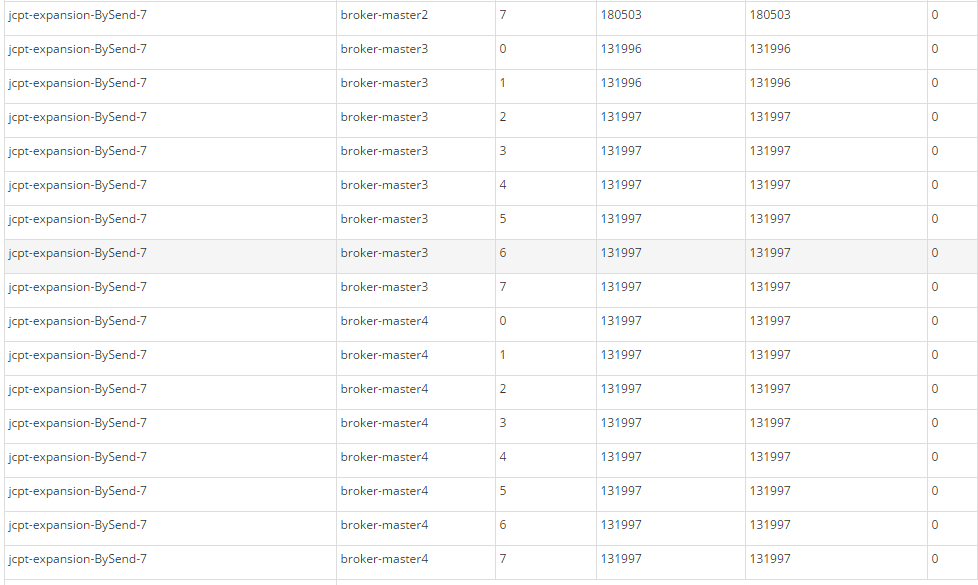




图：5-2-6

消费数量





图：5-2-7

### 本次测试结论

* 图5-2-1与图5-2-3可以看出topic队列的更新。
* 图5-2-5与5-2-7看出消费消息并没有受到扩容影响有大规模的消息堆积情况。
* 图5-2-6可以看出发送消息总数180503\*16+131997\*16= 2888048+2111952=500万，
* 图5-2-6与图5-2-7可以看出 发送时队列中保存消息条数，与消费消息条数一致。
* 所以本次测试结果与上次测试完全相同，

# 整体结论

进过两轮的测试，可以看出两次除了发送消息数量的大小意外，扩容并没有对集群有很大影响。结论如下：

* 集群扩容后新的节点可以自动加入集群，整个集群的运行状态也十分良好。
* 所开发的topic更新接口经过两次测试，均无出现异常，达到理想效果。
* 扩容后Topic会自动更新到新的节点上。
* 扩容后大约1分钟左右，发送消息会发送到新加的节点上的队列里。
* 扩容后发送消息，可负载到整个集群队列上。
* 在发送消息时，同时进行扩容，没有出现，程序卡死，程序异常，消息丢失的等情况。
* 在消费过程时，同时进行扩容，没有出现，程序卡死，程序异常，消息逐一消费，重复消费的等情况。