GMQ—事务场景生产TPS测试

# 一  机器部署

## 1、机器组成

共4台机器，均为16G内存，每台服务器均有4个CPU，2核

|  |  |  |
| --- | --- | --- |
| 机器名 | IP地址 | 用途 |
| xxdlyc02/node 104 | 10.128.31.104 | 消息队列压力测试机 |
| xxdlyc03/node 105 | 10.128.31.105 | 消息队列压力测试机 |
| xxdlyc03/node 106 | 10.128.31.106 | 消息队列压力测试机 |
| xxdlyc03/node 107 | 10.128.31.107 | 消息队列压力测试机 |

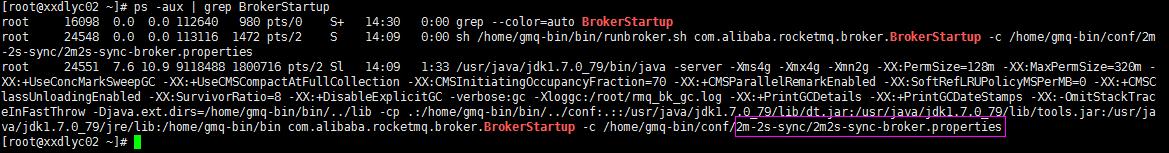
## 2、运行环境配置

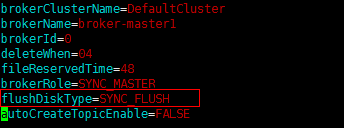


上图中NameSrv使用四台机器与Broker集群共用，Broker使用双Master/Slave、同步双写刷盘模式。Master与Slave通过制定相同的brokerName配对，其中Master的BrokerId必须为0，Slave的BrokerId必须大于0.

## 3、刷盘方式

每台机器master机器均采用同步刷盘方式





# 二 性能评测

## 1、评测目的

多线程事务环境下，测试producer端的TPS 。

## 2、评测指标

生产者producer

事务消息长度与TPS的关系、多线程TPS

## 3、评测逻辑

（1）固定消息长度，producer端发送消息body大小为128字节。

（2）输入不同的线程数，产生不同组的producer，记录发送消息的TPS、发送成功数等等。

（3）根据多组测试数据，分析平均的生产TPS。

## 4、评测步骤

（1）创建性能测试的topic，名称为TransactionTopic， 队列个数默认8个。

（2）输入线程数、消息长度，并做表格记录TPS。

（3）针对特定场景，单线程下增加消息message的size，记录producer端，并做表格记录。

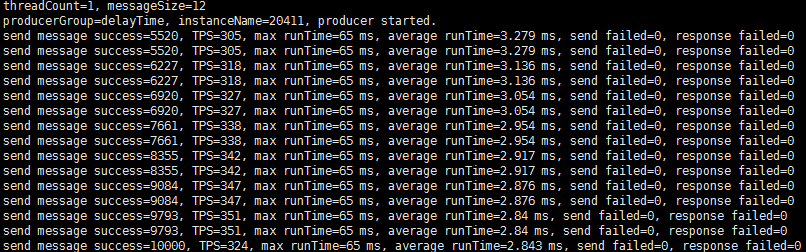
（4）固定消息message的size，逐次增加线程数量，记录producer端的TPS，并做表格记录。

## 5、评测过程

### 5.1 消息大小与事务生产者TPS关系测试

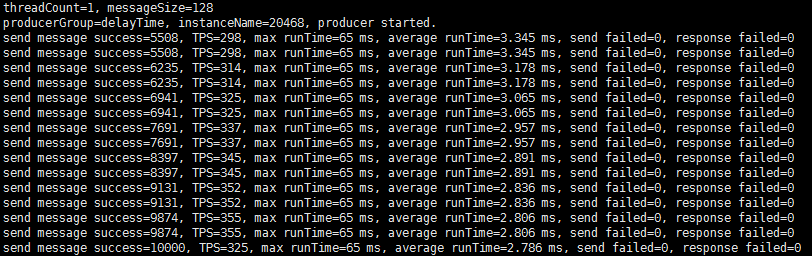
1. 第一组，单线程，消息长度12字节

Producer端发送事务消息记录



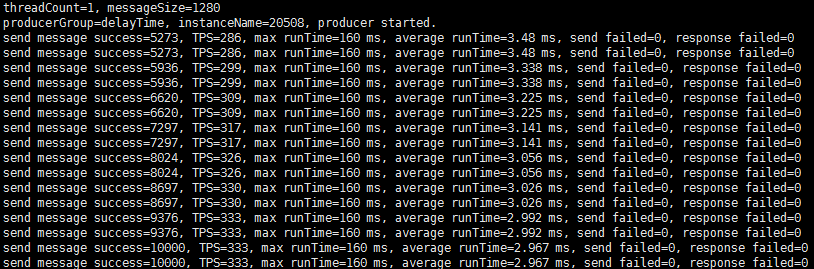
1. 第二组，单线程，消息长度128字节

Producer端发送事务消息记录



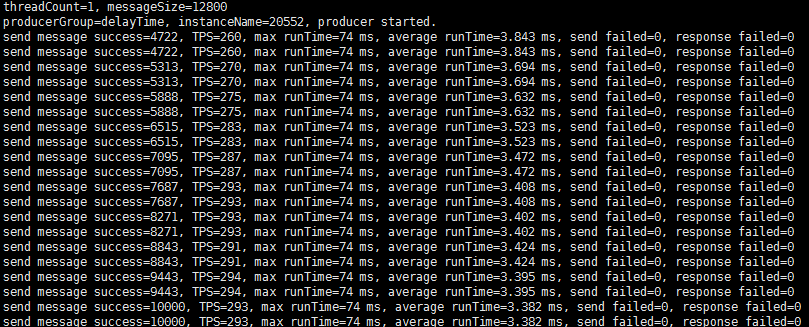
1. 第三组，单线程，消息长度1280字节

Producer端发送事务消息记录



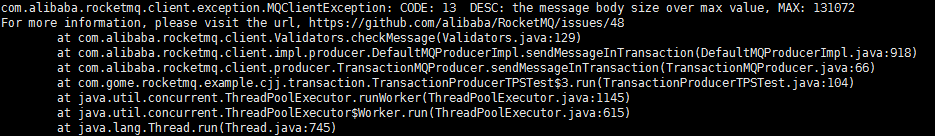
1. 第四组，单线程，消息长度12800字节

Producer端发送事务消息记录



1. 第五组，单线程，消息长度128000字节

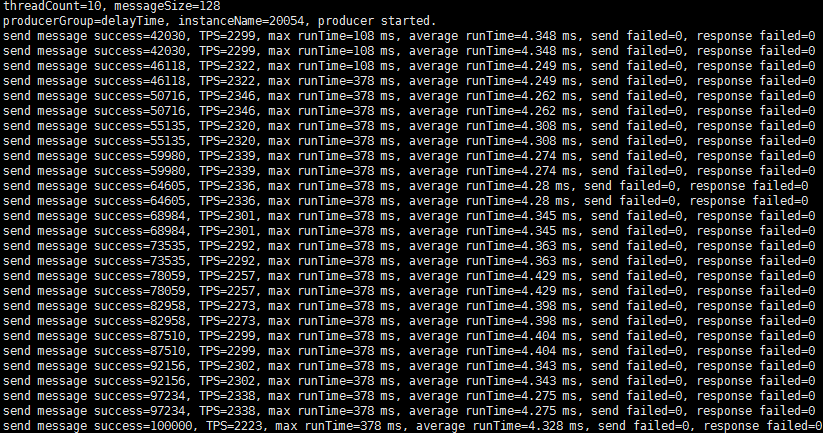
Producer端发送事务消息记录，超过消息最大长度，抛出异常。



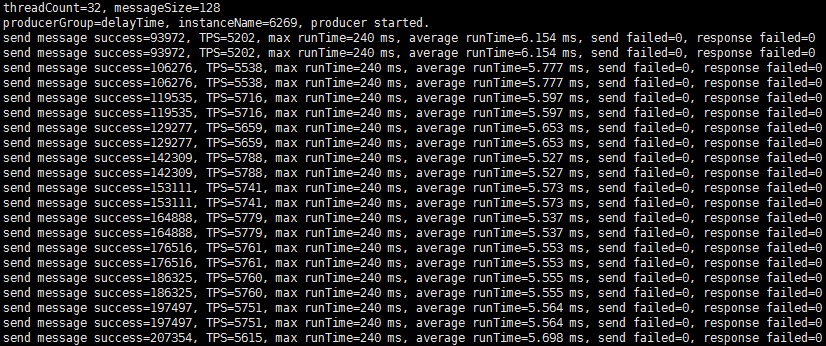
### 5.2多线程TPS测试

1. 第一组 线程个数10，消息长度固定128字节

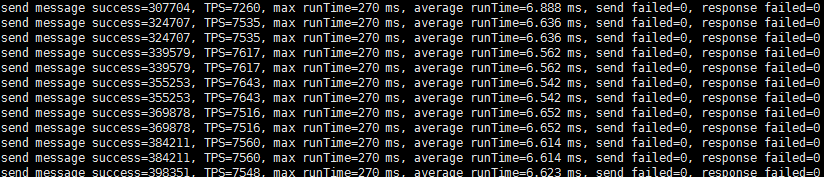
Producer端发送事务消息记录



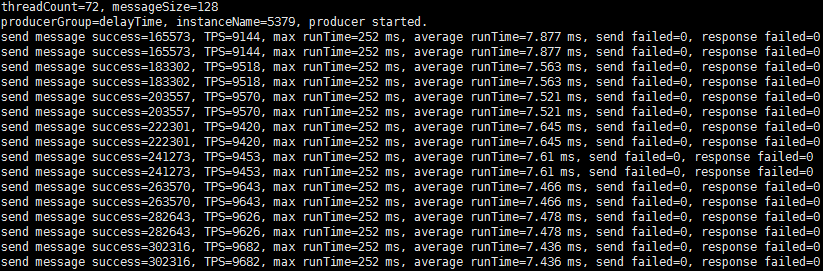
1. 第二组 线程个数32，消息长度固定128字节



1. 第三组 线程个数50，消息长度固定128 字节

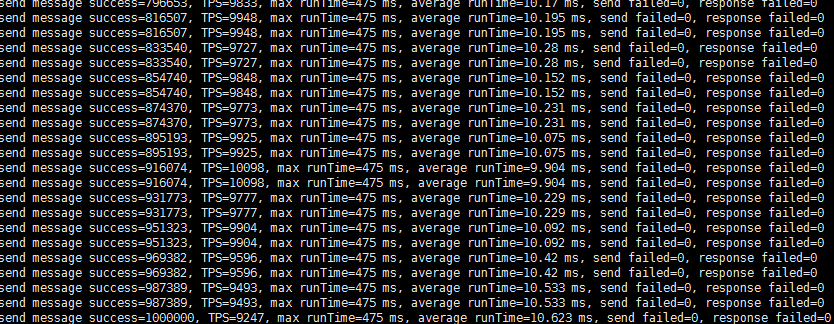


1. 第四组 线程个数72，消息长度固定128字节

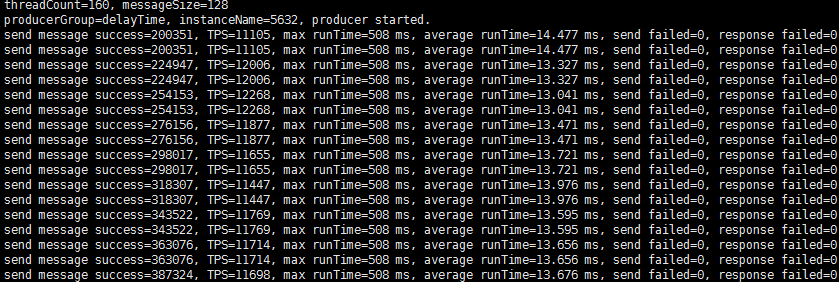


1. 第五组 线程个数100，消息长度固定128字节

Producer端发送事务消息记录

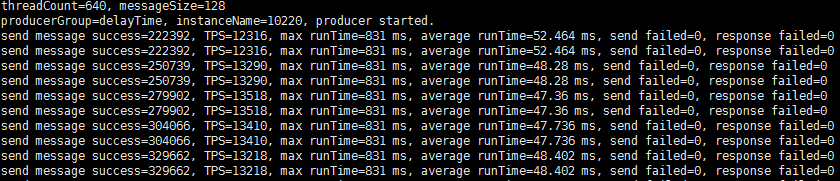


1. 第六组 线程个数160，消息固定长度128字节



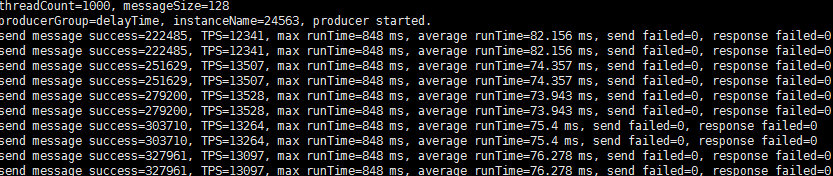
1. 第七组 线程个数640，消息长度固定128字节

Producer端发送事务消息记录



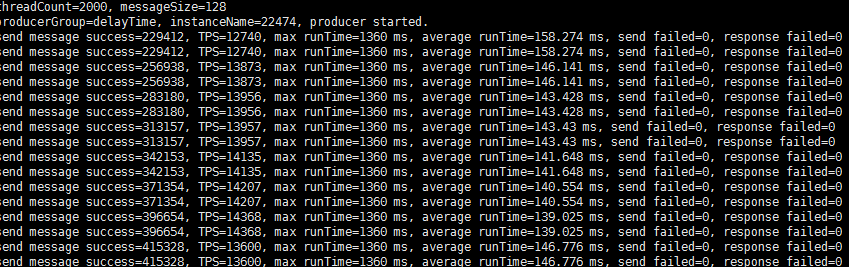
1. 第八组 线程个数1000，消息长度固定128字节

Producer端发送事务消息记录



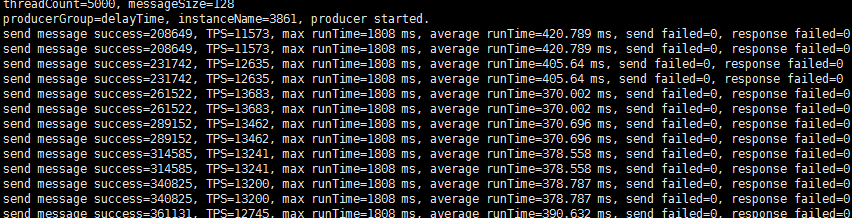
1. 第九组 线程个数2000，消息长度固定128字节

Producer端发送事务消息记录



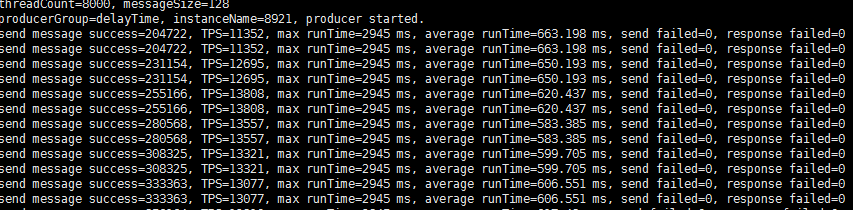
1. 第十组 线程个数5000，消息长度固定128 字节

Producer端发送事务消息记录



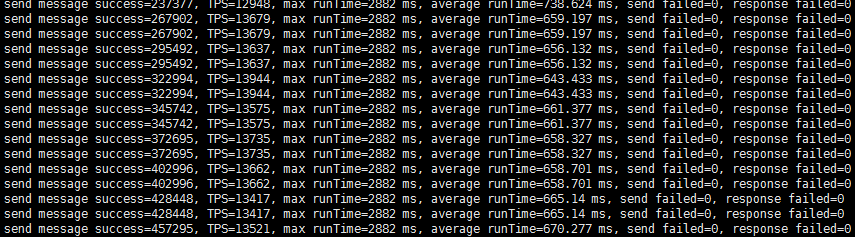
1. 第十一组 线程个数8000，消息长度固定128 字节

Producer端发送事务消息记录



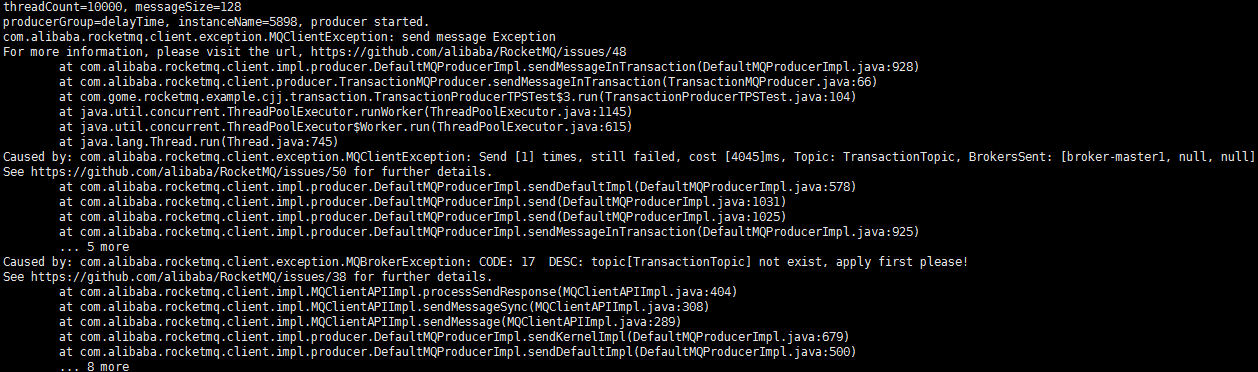
1. 第十二组 线程个数9000，消息长度固定 128字节

Producer端发送事务消息记录



1. 第十二组 线程个数10000，消息长度固定128字节

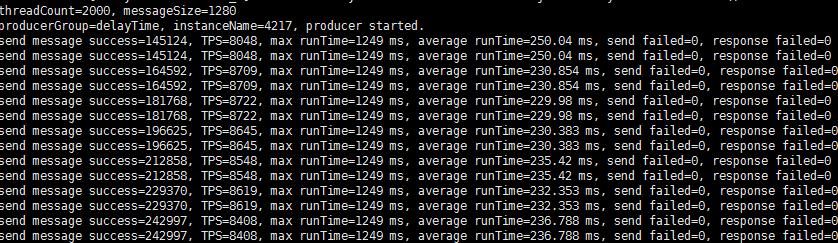
Producer端发送事务消息记录



### 5.3固定线程数为2000时

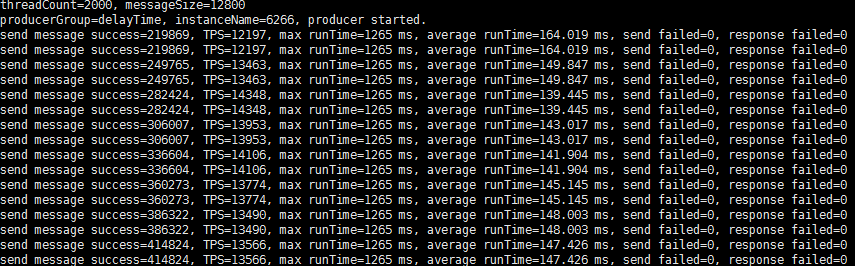
1. 第一组 线程个数2000，消息长度固定1280字节

Producer端发送事务消息记录



1. 第一组 线程个数2000，消息长度固定12800字节

Producer端发送事务消息记录



6、分析结果如下：

1. 单线程事务消息生产：

保持线程数为1,，逐步增加消息大小，TPS取平均值统计值，结果如下

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | TPS(Producer) | 线程数 | 消息长度 |
| 1 | 330 | 1 | 12 |
| 2 | 350 | 1 | 128 |
| 3 | 330 | 1 | 1280 |
| 4 | 290 | 1 | 12800 |
| 5 | 异常 | 1 | 128000 |

1. 多线程事务消息生产

保持消息长度为128字节，逐步增加线程数，不配置message消息的key值，TPS取平均值统计值，结果如下

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | TPS(Producer) | 线程数 | 消息长度  （28字节） | 是否配置 message的key值 |
| 1 | 300 | 1 | 128 | FALSE |
| 2 | 2300 | 10 | 128 | FALSE |
| 3 | 5600 | 32 | 128 | FALSE |
| 4 | 7600 | 50 | 128 | FALSE |
| 5 | 9600 | 72 | 128 | FALSE |
| 6 | 10500 | 100 | 128 | FALSE |
| 7 | 12000 | 160 | 128 | FALSE |
| 8 | 13400 | 640 | 128 | FALSE |
| 9 | 13500 | 1000 | 128 | FALSE |
| 10 | 14000 | 2000 | 128 | FALSE |
| 11 | 13500 | 5000 | 128 | FALSE |
| 12 | 13500 | 8000 | 128 | FALSE |
| 13 | 13600 | 9000 | 128 | FALSE |
| 14 | 异常 | 10000 | 128 | FALSE |

（3）当固定线程数为2000时，改变消息大小，TPS取平均值统计值，结果如下

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | TPS(Producer) | 线程数 | 消息长度 |
| 1 | 14000 | 2000 | 128 |
| 2 | 8600 | 2000 | 1280 |
| 3 | 13500 | 2000 | 12800 |

# 三 评测结果

1、消息大小与事务生产者TPS关系测试方面：消息大小达到阀值128时，TPS为最大，当消息大小大于128时，TPS呈下降趋势。但是消息大小达到128000时，程序抛出异常。

2、多线程TPS测试方面：固定消息大小为128，随着线程数量增加， producer端的TPS均保持逐步增加的趋势，当线程数达到2000时，TPS 达到最大值；当线程数超过2000时，TPS下降，但稳定在13500左右；当线程数超过10000时，抛出异常。

3、当线程数为2000时，改变消息的大小，当消息大小为1280时，测得的TPS与消息大小为128时测得的TPS相比下降的幅度很大，当消息大小为12800时，TPS 反而上升了，即呈现一个“U”形状。但是测试普通多线程时，TPS是随着消息大小的增加而下降的。