

分 数:	
评卷人:	

华中科技大学

研究生（数据中心技术）课程 实验报告

学 号 M202173487

姓 名 廖子逸

专 业 电子信息

课程指导教师 施展 童薇

院（系、所） 武汉光电国家研究中心

2021 年 1 月 6 日

一 Minio

实验一：系统搭建

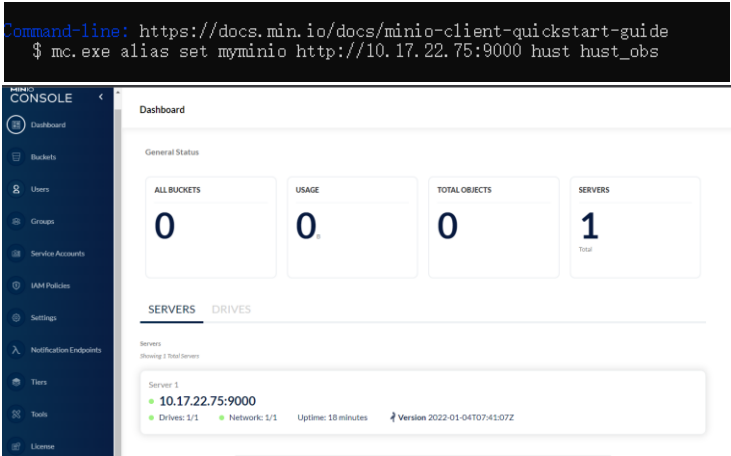


图 1 Minio 搭建 Dashboard 示例图

实验二：性能观测

步骤一：新建桶

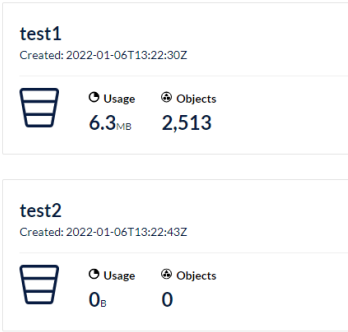


图 2 新建 Bucket 示例图

步骤二：修改参数，观测结果

Config1:

```
s3bench.exe ^
-accessKey=hust ^
-accessSecret=hust_obs ^
-bucket=test2 ^
-endpoint=http://127.0.0.1:9000 ^
-numClients=8 ^
-numSamples=100 ^
-objectNamePrefix=loadgen ^
-objectSize=1024
pause
```

```
Results Summary for Write Operation(s)
Total Transferred: 0.098 MB
Total Throughput: 0.02 MB/s
Total Duration: 4.723 s
Number of Errors: 0

-----
Write times Max: 0.779 s
Write times 99th %ile: 0.779 s
Write times 90th %ile: 0.551 s
Write times 75th %ile: 0.448 s
Write times 50th %ile: 0.371 s
Write times 25th %ile: 0.260 s
Write times Min: 0.105 s

Results Summary for Read Operation(s)
Total Transferred: 0.098 MB
Total Throughput: 1.84 MB/s
Total Duration: 0.053 s
Number of Errors: 0

-----
Read times Max: 0.047 s
Read times 99th %ile: 0.047 s
Read times 90th %ile: 0.002 s
Read times 75th %ile: 0.001 s
Read times 50th %ile: 0.001 s
Read times 25th %ile: 0.001 s
Read times Min: 0.000 s
```

图 3 Config1 下的性能观测

Config2:

```
s3bench.exe ^
  -accessKey=hust ^
  -accessSecret=hust_obs ^
  -bucket=test2 ^
  -endpoint=http://127.0.0.1:9000 ^
  -numClients=8 ^
  -numSamples=100 ^
  -objectNamePrefix=loadgen ^
  -objectSize=10240
pause
```

```
Results Summary for Write Operation(s)
Total Transferred: 0.977 MB
Total Throughput: 0.17 MB/s
Total Duration: 5.742 s
Number of Errors: 0
-----
Write times Max: 0.750 s
Write times 99th %ile: 0.750 s
Write times 90th %ile: 0.628 s
Write times 75th %ile: 0.530 s
Write times 50th %ile: 0.437 s
Write times 25th %ile: 0.360 s
Write times Min: 0.142 s

Results Summary for Read Operation(s)
Total Transferred: 0.977 MB
Total Throughput: 94.05 MB/s
Total Duration: 0.010 s
Number of Errors: 0
-----
Read times Max: 0.002 s
Read times 99th %ile: 0.002 s
Read times 90th %ile: 0.001 s
Read times 75th %ile: 0.001 s
Read times 50th %ile: 0.001 s
Read times 25th %ile: 0.001 s
Read times Min: 0.001 s
```

图 4 Config2 下的性能观测

Config3:

```
s3bench.exe ^
  -accessKey=hust ^
  -accessSecret=hust_obs ^
  -bucket=test2 ^
  -endpoint=http://127.0.0.1:9000 ^
  -numClients=30 ^
  -numSamples=100 ^
  -objectNamePrefix=loadgen ^
  -objectSize=1024
pause
```

```
Results Summary for Write Operation(s)
Total Transferred: 0.098 MB
Total Throughput: 0.01 MB/s
Total Duration: 8.555 s
Number of Errors: 0
-----
Write times Max: 4.979 s
Write times 99th %ile: 4.979 s
Write times 90th %ile: 4.413 s
Write times 75th %ile: 3.240 s
Write times 50th %ile: 1.724 s
Write times 25th %ile: 1.591 s
Write times Min: 0.771 s

Results Summary for Read Operation(s)
Total Transferred: 0.098 MB
Total Throughput: 6.10 MB/s
Total Duration: 0.016 s
Number of Errors: 0
-----
Read times Max: 0.009 s
Read times 99th %ile: 0.009 s
Read times 90th %ile: 0.008 s
Read times 75th %ile: 0.005 s
Read times 50th %ile: 0.004 s
Read times 25th %ile: 0.003 s
Read times Min: 0.001 s
```

图 5 Config3 下的性能观测

总结:

对比图 3 和图 4, 配置 2 相比配置 1, 扩大了 10 倍块大小, 由结果得出: 扩大块大小, 在写测试性能上变化不大, 但在读测试上, 带宽明显变大, 延迟明显变小。

对比图 3 和图 5, 配置 3 相比配置 1, 增加了约 4 倍数量的 clients, 导致: 写测试中, 带宽降低 1 倍, 延迟大幅度变大 (增加后的最小延迟等于增加前的最大延迟); 读测试中, 带宽提高 4 倍, 延迟大幅度降低。

实验三: 尾延迟

步骤一: 尾延迟观测

尾部延迟 (也称为高百分比延迟) 是指客户端很少看到的高延迟。例如: “我的服务通常在 10 毫秒左右响应, 但有时需要 100 毫秒左右”。世界上有很多导致尾部等待时间的原因, 包括争用, 垃圾回收, 数据包丢失, 主机故障以及操作系统在后台执行的奇怪操作。

为什么要关注尾延迟? 木桶效应, 短板效应。

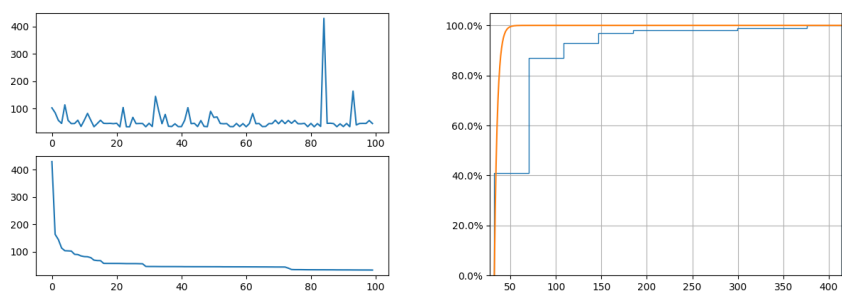


图 6 尾延迟观测

步骤二：对冲请求

原理：当请求在指定的时间间隔后没有返回时，会发起对冲请求。通常对于 95%尾延迟作为阈值，认为 95%尾延迟以上的请求出现问题，然后重新发送请求。

操作：修改 `request_timing()` 函数，当 `system_time` 超过阈值的时候，递归调用一次 `request_timing()` 函数并且返回较小一个 `system_time`。由图 6 观察得到，阈值选取 50ms。

结果：如图 7 所示，尾延迟降低至 110ms（原来为 400ms）。实验验证了对冲请求的有效性。

不足：在 `request_timing()` 函数中穿插时间判断，然后递归调用 `request_timing()` 是一种比较简单的实现方式，但是有可能影响原有进程效率，且不符合实际的多客户访问情况。应该在 `executor.submit()` 函数中设置计时器，超时则重新请求，才是完整端到端测试，但这样需要修改的代码较多，留给以后尝试。

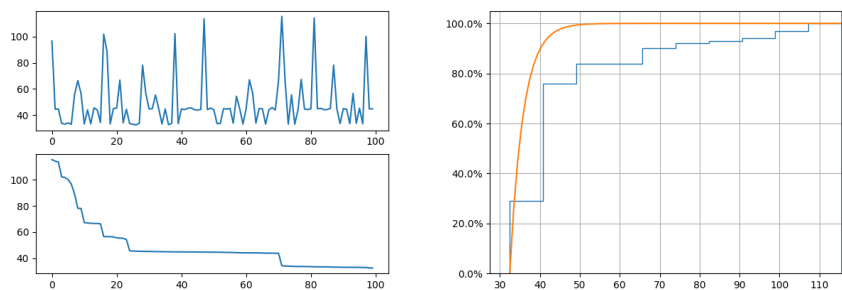


图 7 对冲请求后的尾延迟观测

二 Ceph

实验一：系统搭建

系统使用 VirtualBox 建立三台 Ubuntu 虚拟机搭建 Ceph 集群。

具体步骤为：

- 1 使用 VirtualBox 安装三台虚拟机，系统版本为：Ubuntu20.04，每台虚拟机添加一个虚拟硬盘做 OSD。
- 2 分配 IP 段，连接 Xshell 操作虚拟机，主机和 IP 分配如表 1 所示。
- 3 安装 Cephadm，使用 Cephadm 容器方法部署 Ceph 集群。
- 4 Ceph 搭建完成，如图 8 所示。
- 5 配置 Ceph 对象存储网关，如图 9 所示。

详细步骤见博客：<https://blog.csdn.net/HzauTriste/article/details/122480450>

```
root@ceph111:~# ceph -s
cluster:
  id:          5688d2f8-944b-11ec-941d-a39f4739400c
  health:      HEALTH_OK

services:
  mon: 3 daemons, quorum ceph111,ceph113,ceph112 (age 12m)
  mgr: ceph111.lxcvtj(active, since 14m), standbys: ceph112.imkpbo
  osd: 3 osds: 3 up (since 12m), 3 in (since 5d)
  rgw: 3 daemons active (rgw01.zone01.ceph111.fhvjkb, rgw01.zone01.ceph112.ijcysd, rgw01.zone01.ceph113.xwlchi)

task status:

data:
  pools:   7 pools, 145 pgs
  objects: 287 objects, 12 KiB
  usage:   3.2 GiB used, 57 GiB / 60 GiB avail
  pgs:    145 active+clean
```

图 8 Ceph 示意图

表 1 Ceph Host IP

Host	IP
Ceph111	10.0.0.111
Ceph112	10.0.0.112
Ceph113	10.0.0.113

```
root@ceph111:~# ceph orch ps --daemon-type rgw
NAME                                HOST      STATUS    REFRESHED  AGE  VERSION  IMAGE NAME                                IMAGE ID    CONTAINER ID
rgw.rgw01.zone01.ceph111.fhvjkb    ceph111  running  (17m)     5m ago    5d    15.2.13  docker.io/ceph/ceph:v15                2cf504fded39  5615d04a0e3a
rgw.rgw01.zone01.ceph112.ijcysd    ceph112  running  (15m)     5m ago    5d    15.2.13  docker.io/ceph/ceph:v15                2cf504fded39  5c2ac90fbe2f
rgw.rgw01.zone01.ceph113.xwlchi    ceph113  running  (17m)     5m ago    5d    15.2.13  docker.io/ceph/ceph:v15                2cf504fded39  69ff5cbe988a
```

图 9 Ceph RGW 进程

实验二：性能观测

使用 S3 bench 进行测试

1 初始配置。

```
run-s3bench.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://10.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIW2Z8T357PS4QKF4Y ^
-accessSecret=LZabxBBjxG6rw2LQeQLyKHlQbdgM7vILMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=8 ^
-numSamples=256 ^
-objectNamePrefix=loadgen ^
-objectSize=1024
pause
```

```
Running Write test...

Running Read test...

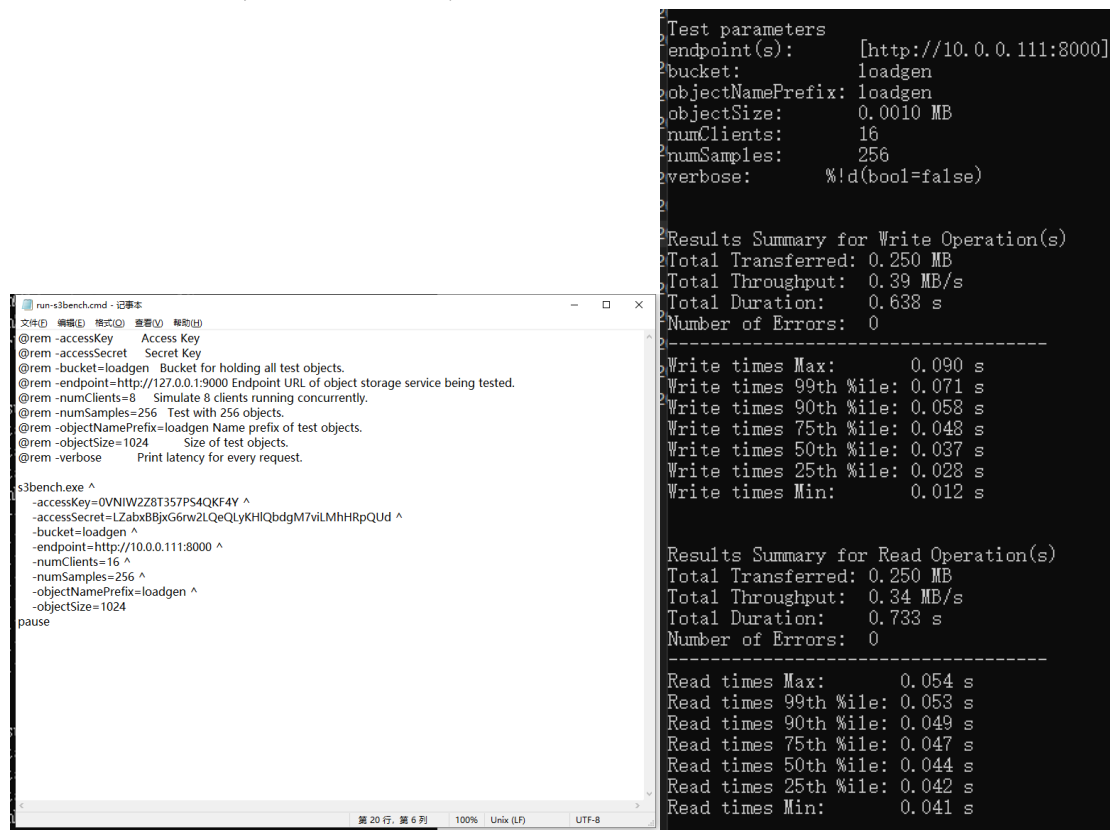
Test parameters
endpoint(s):      [http://10.0.0.111:8000]
bucket:           loadgen
objectNamePrefix: loadgen
objectSize:       0.0010 MB
numClients:       8
numSamples:       256
verbose:          %!d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 0.250 MB
Total Throughput:  0.35 MB/s
Total Duration:    0.717 s
Number of Errors:  0
-----
Write times Max:   0.053 s
Write times 99th %ile: 0.047 s
Write times 90th %ile: 0.032 s
Write times 75th %ile: 0.024 s
Write times 50th %ile: 0.020 s
Write times 25th %ile: 0.017 s
Write times Min:   0.010 s

Results Summary for Read Operation(s)
Total Transferred: 0.250 MB
Total Throughput:  0.18 MB/s
Total Duration:    1.409 s
Number of Errors:  0
-----
Read times Max:    0.066 s
Read times 99th %ile: 0.062 s
Read times 90th %ile: 0.047 s
Read times 75th %ile: 0.044 s
Read times 50th %ile: 0.042 s
Read times 25th %ile: 0.042 s
```

图 10 使用 S3 bench 进行 Ceph 性能观测 1

2 增加 clients 至 16, 可见延迟增加, 带宽增加。



```
run-s3bench.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIWZ28T357PS4QKF4Y ^
-accessSecret=LZabx8BjxG6rw2LQeQLyKHIQbdgM7vILMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=16 ^
-numSamples=256 ^
-objectNamePrefix=loadgen ^
-objectSize=1024
pause

Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.0010 MB
numClients: 16
numSamples: 256
verbose: %!d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 0.250 MB
Total Throughput: 0.39 MB/s
Total Duration: 0.638 s
Number of Errors: 0

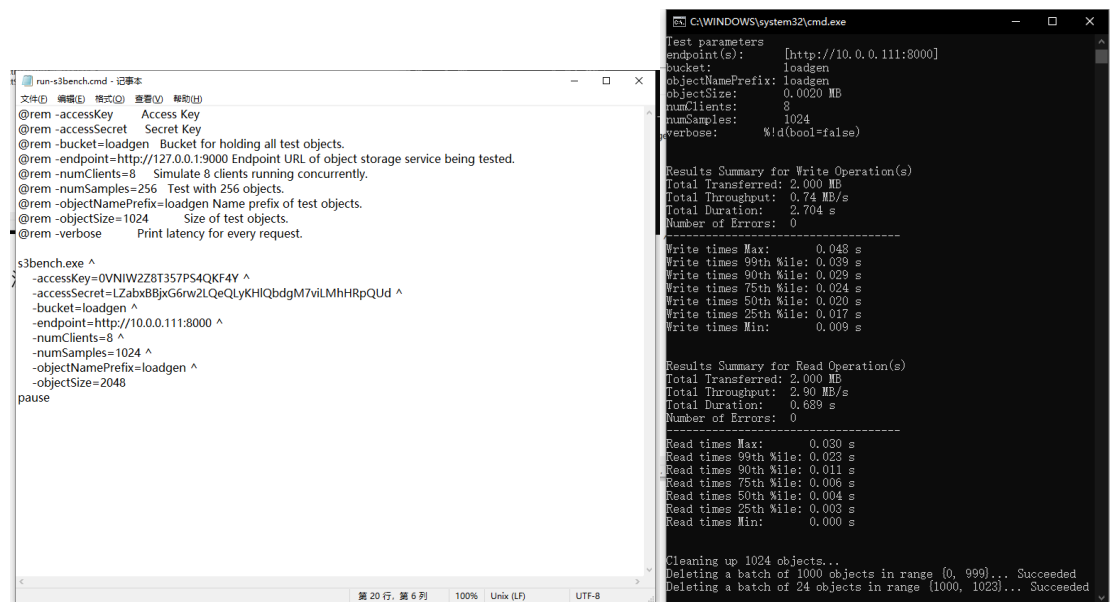
Write times Max: 0.090 s
Write times 99th %ile: 0.071 s
Write times 90th %ile: 0.058 s
Write times 75th %ile: 0.048 s
Write times 50th %ile: 0.037 s
Write times 25th %ile: 0.028 s
Write times Min: 0.012 s

Results Summary for Read Operation(s)
Total Transferred: 0.250 MB
Total Throughput: 0.34 MB/s
Total Duration: 0.733 s
Number of Errors: 0

Read times Max: 0.054 s
Read times 99th %ile: 0.053 s
Read times 90th %ile: 0.049 s
Read times 75th %ile: 0.047 s
Read times 50th %ile: 0.044 s
Read times 25th %ile: 0.042 s
Read times Min: 0.041 s
```

图 11 使用 S3 bench 进行 Ceph 性能观测 2

3 加大测试数据量, 可见 object 大小增大一倍带宽提升, 延迟降低



```
run-s3bench.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIWZ28T357PS4QKF4Y ^
-accessSecret=LZabx8BjxG6rw2LQeQLyKHIQbdgM7vILMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=8 ^
-numSamples=256 ^
-objectNamePrefix=loadgen ^
-objectSize=2048
pause

CA\WINDOWS\system32\cmd.exe
Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.0020 MB
numClients: 8
numSamples: 1024
verbose: %!d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 2.000 MB
Total Throughput: 0.74 MB/s
Total Duration: 2.704 s
Number of Errors: 0

Write times Max: 0.048 s
Write times 99th %ile: 0.039 s
Write times 90th %ile: 0.029 s
Write times 75th %ile: 0.024 s
Write times 50th %ile: 0.020 s
Write times 25th %ile: 0.017 s
Write times Min: 0.009 s

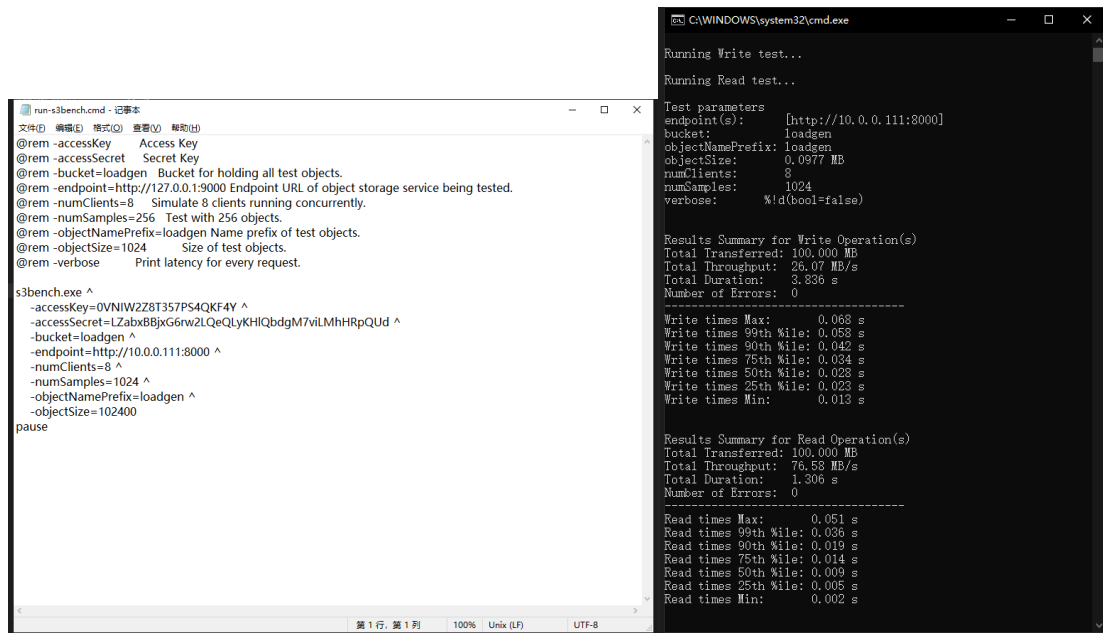
Results Summary for Read Operation(s)
Total Transferred: 2.000 MB
Total Throughput: 2.90 MB/s
Total Duration: 0.689 s
Number of Errors: 0

Read times Max: 0.030 s
Read times 99th %ile: 0.023 s
Read times 90th %ile: 0.011 s
Read times 75th %ile: 0.006 s
Read times 50th %ile: 0.004 s
Read times 25th %ile: 0.003 s
Read times Min: 0.000 s

Cleaning up 1024 objects...
Deleting a batch of 1000 objects in range [0, 999]... Succeeded
Deleting a batch of 24 objects in range [1000, 1023]... Succeeded
```

图 12 使用 S3 bench 进行 Ceph 性能观测 3

4.增大 100 倍 objectsize 大小, 带宽显著提高, 延迟基本不变, 说明还没到带宽瓶颈。



The screenshot shows a Windows command prompt window titled 'run-s3bench.cmd - 记事本' and a terminal window titled 'C:\WINDOWS\system32\cmd.exe'. The command prompt contains the following commands:

```
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIW2Z8T357PS4QKF4Y ^
-accessSecret=LZabxBBjxG6rw2LQeQLyKHlQbdgM7vILMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=8 ^
-numSamples=1024 ^
-objectNamePrefix=loadgen ^
-objectSize=102400
pause
```

The terminal output shows the results of the write and read tests:

```
Running Write test...
Running Read test...

Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.0977 MB
numClients: 8
numSamples: 1024
verbose: %!d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 100.000 MB
Total Throughput: 26.07 MB/s
Total Duration: 3.836 s
Number of Errors: 0

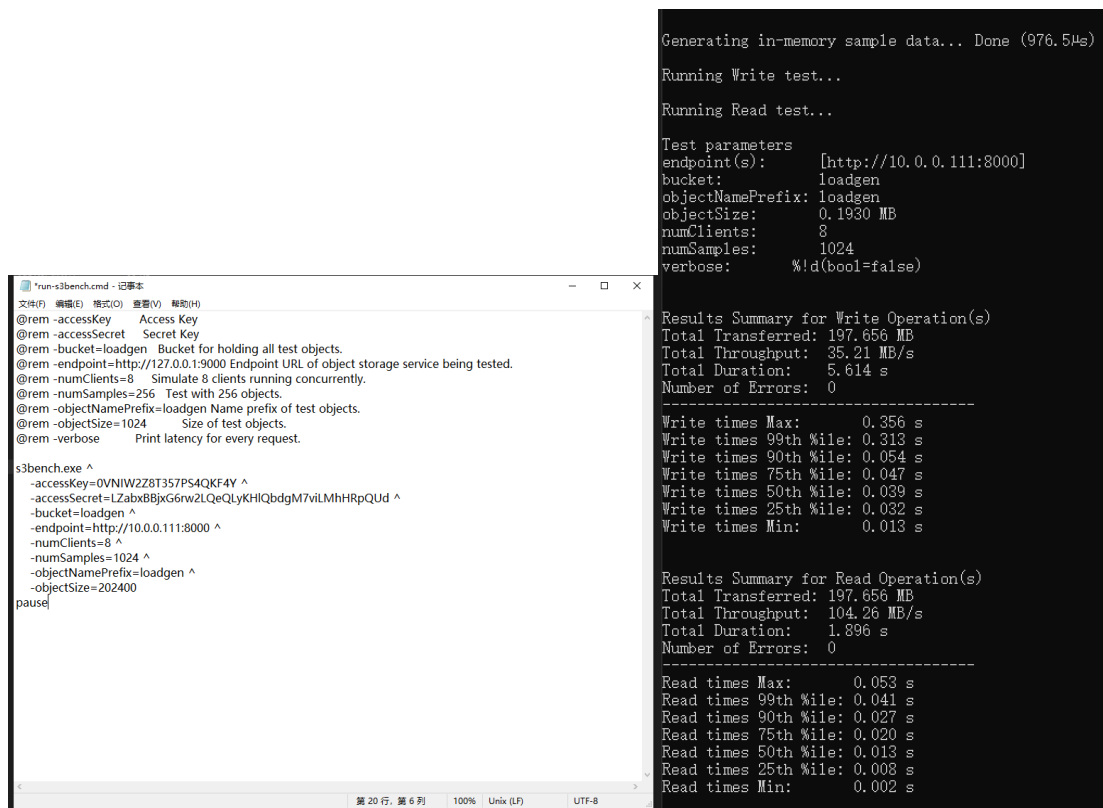
Write times Max: 0.068 s
Write times 99th %ile: 0.058 s
Write times 90th %ile: 0.042 s
Write times 75th %ile: 0.034 s
Write times 50th %ile: 0.028 s
Write times 25th %ile: 0.023 s
Write times Min: 0.013 s

Results Summary for Read Operation(s)
Total Transferred: 100.000 MB
Total Throughput: 76.58 MB/s
Total Duration: 1.306 s
Number of Errors: 0

Read times Max: 0.051 s
Read times 99th %ile: 0.036 s
Read times 90th %ile: 0.019 s
Read times 75th %ile: 0.014 s
Read times 50th %ile: 0.009 s
Read times 25th %ile: 0.005 s
Read times Min: 0.002 s
```

图 13 使用 S3 bench 进行 Ceph 性能观测 4

5 相较 4 增大一倍 objectsize，写的尾延迟明显增大，带宽增大。



The screenshot shows a Windows command prompt window titled 'run-s3bench.cmd - 记事本' and a terminal window titled 'C:\WINDOWS\system32\cmd.exe'. The command prompt contains the following commands:

```
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIW2Z8T357PS4QKF4Y ^
-accessSecret=LZabxBBjxG6rw2LQeQLyKHlQbdgM7vILMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=8 ^
-numSamples=1024 ^
-objectNamePrefix=loadgen ^
-objectSize=202400
pause
```

The terminal output shows the results of the write and read tests:

```
Generating in-memory sample data... Done (976.5µs)
Running Write test...
Running Read test...

Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.1930 MB
numClients: 8
numSamples: 1024
verbose: %!d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 197.656 MB
Total Throughput: 35.21 MB/s
Total Duration: 5.614 s
Number of Errors: 0

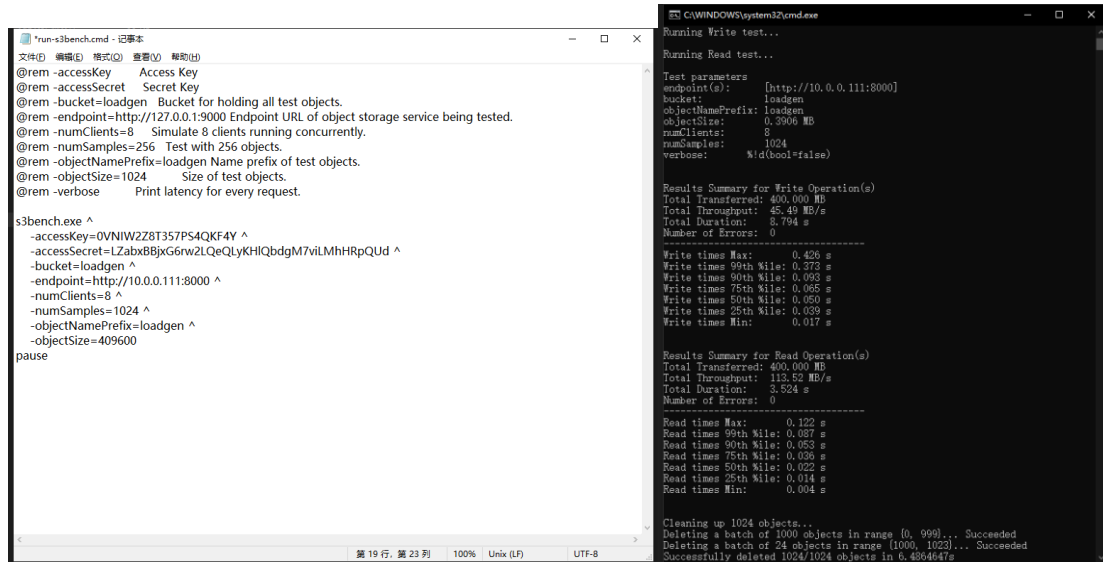
Write times Max: 0.356 s
Write times 99th %ile: 0.313 s
Write times 90th %ile: 0.054 s
Write times 75th %ile: 0.047 s
Write times 50th %ile: 0.039 s
Write times 25th %ile: 0.032 s
Write times Min: 0.013 s

Results Summary for Read Operation(s)
Total Transferred: 197.656 MB
Total Throughput: 104.26 MB/s
Total Duration: 1.896 s
Number of Errors: 0

Read times Max: 0.053 s
Read times 99th %ile: 0.041 s
Read times 90th %ile: 0.027 s
Read times 75th %ile: 0.020 s
Read times 50th %ile: 0.013 s
Read times 25th %ile: 0.008 s
Read times Min: 0.002 s
```

图 14 使用 S3 bench 进行 Ceph 性能观测 5

6 相较 5 增大一倍 objectsize，读带宽增加不大，且读尾延迟增加较多，写性能上带宽增大，延迟变化不大。



The screenshot shows a Windows command prompt window titled "run-s3bench.cmd - 记事本" on the left and a terminal window titled "C:\WINDOWS\system32\cmd.exe" on the right. The command prompt contains the following commands:

```
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIWZ28T357PS4QKF4Y ^
-accessSecret=LZabxBjxG6rw2LQeQLyKHlQbdgM7vILMhHrpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=8 ^
-numSamples=256 ^
-objectNamePrefix=loadgen ^
-objectSize=409600
pause
```

The terminal window shows the output of the S3 bench test. It includes test parameters, results for write and read operations, and cleanup status.

```
Running Write test...
Running Read test...

Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.3906 MB
numClients: 8
numSamples: 1024
verbose: %d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 400.000 MB
Total Throughput: 45.49 MB/s
Total Duration: 8.794 s
Number of Errors: 0

Write times Max: 0.426 s
Write times 99th Ntile: 0.372 s
Write times 90th Ntile: 0.063 s
Write times 75th Ntile: 0.065 s
Write times 50th Ntile: 0.050 s
Write times 25th Ntile: 0.039 s
Write times Min: 0.017 s

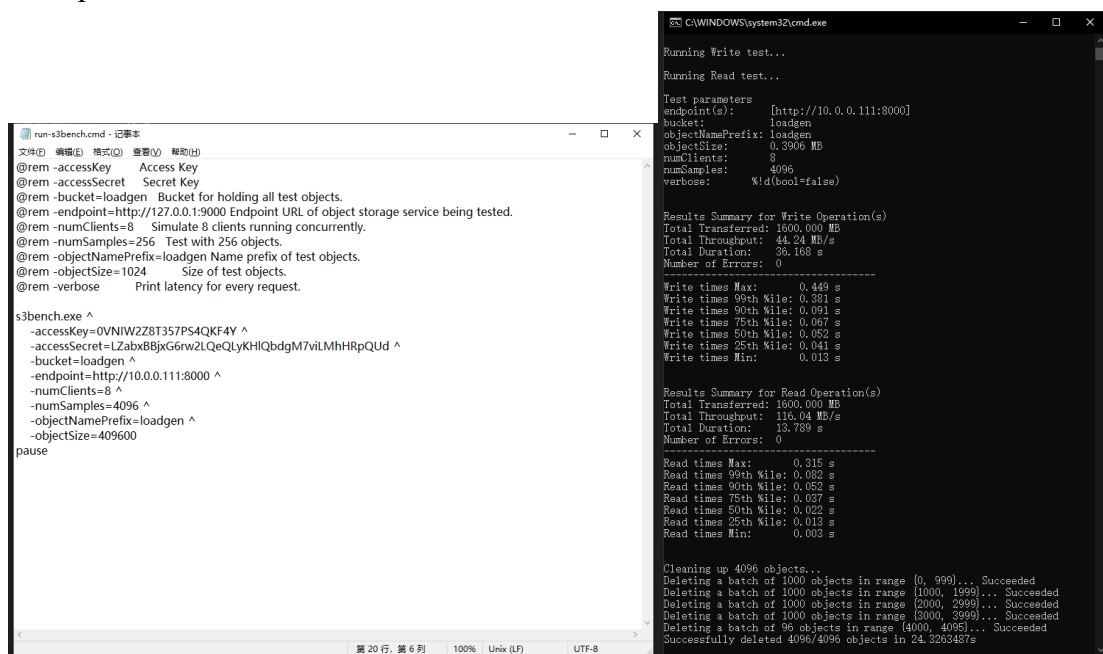
Results Summary for Read Operation(s)
Total Transferred: 400.000 MB
Total Throughput: 113.52 MB/s
Total Duration: 3.524 s
Number of Errors: 0

Read times Max: 0.122 s
Read times 99th Ntile: 0.087 s
Read times 90th Ntile: 0.053 s
Read times 75th Ntile: 0.028 s
Read times 50th Ntile: 0.022 s
Read times 25th Ntile: 0.014 s
Read times Min: 0.004 s

Cleaning up 1024 objects...
Deleting a batch of 1000 objects in range [0, 999]... Succeeded
Deleting a batch of 24 objects in range (1000, 1023)... Succeeded
Successfully deleted 1024/1024 objects in 6.4364647s
```

图 15 使用 S3 bench 进行 Ceph 性能观测 6

7 增大 Samples 数量，对性能影响不大。



The screenshot shows a Windows command prompt window titled "run-s3bench.cmd - 记事本" on the left and a terminal window titled "C:\WINDOWS\system32\cmd.exe" on the right. The command prompt contains the following commands:

```
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIWZ28T357PS4QKF4Y ^
-accessSecret=LZabxBjxG6rw2LQeQLyKHlQbdgM7vILMhHrpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=8 ^
-numSamples=4096 ^
-objectNamePrefix=loadgen ^
-objectSize=409600
pause
```

The terminal window shows the output of the S3 bench test. It includes test parameters, results for write and read operations, and cleanup status.

```
Running Write test...
Running Read test...

Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.3906 MB
numClients: 8
numSamples: 4096
verbose: %d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 1600.000 MB
Total Throughput: 44.24 MB/s
Total Duration: 36.168 s
Number of Errors: 0

Write times Max: 0.449 s
Write times 99th Ntile: 0.381 s
Write times 90th Ntile: 0.091 s
Write times 75th Ntile: 0.067 s
Write times 50th Ntile: 0.052 s
Write times 25th Ntile: 0.041 s
Write times Min: 0.013 s

Results Summary for Read Operation(s)
Total Transferred: 1600.000 MB
Total Throughput: 116.04 MB/s
Total Duration: 13.789 s
Number of Errors: 0

Read times Max: 0.315 s
Read times 99th Ntile: 0.082 s
Read times 90th Ntile: 0.052 s
Read times 75th Ntile: 0.037 s
Read times 50th Ntile: 0.022 s
Read times 25th Ntile: 0.013 s
Read times Min: 0.003 s

Cleaning up 4096 objects...
Deleting a batch of 1000 objects in range [0, 999]... Succeeded
Deleting a batch of 1000 objects in range (1000, 1999)... Succeeded
Deleting a batch of 1000 objects in range (2000, 2999)... Succeeded
Deleting a batch of 1000 objects in range (3000, 3999)... Succeeded
Deleting a batch of 96 objects in range (4000, 4095)... Succeeded
Successfully deleted 4096/4096 objects in 24.3263487s
```

图 16 使用 S3 bench 进行 Ceph 性能观测 7

8 增多 client 数量，带宽增加，延迟增加。


```
run-s3bench.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIWZ28T357PS4QKF4Y ^
-accessSecret=LZabxBBjxG6rw2LQeQLyKHlQbdgM7viLMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=16 ^
-numSamples=1024 ^
-objectNamePrefix=loadgen ^
-objectSize=409600
pause

C:\WINDOWS\system32\cmd.exe
objectNamePrefix: loadgen
objectSize: 0.3906 MB
numClients: 16
numSamples: 1024
verbose: %d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 400.000 MB
Total Throughput: 54.25 MB/s
Total Duration: 7.374 s
Number of Errors: 0
-----
Write times Max: 0.449 s
Write times 99th %ile: 0.212 s
Write times 90th %ile: 0.157 s
Write times 75th %ile: 0.131 s
Write times 50th %ile: 0.110 s
Write times 25th %ile: 0.093 s
Write times Min: 0.036 s

Results Summary for Read Operation(s)
Total Transferred: 400.000 MB
Total Throughput: 127.14 MB/s
Total Duration: 3.146 s
Number of Errors: 0
-----
Read times Max: 0.169 s
Read times 99th %ile: 0.149 s
Read times 90th %ile: 0.102 s
Read times 75th %ile: 0.070 s
Read times 50th %ile: 0.039 s
Read times 25th %ile: 0.020 s
Read times Min: 0.005 s

Cleaning up 1024 objects...
Deleting a batch of 1000 objects in range (0, 999)... Succeeded
Deleting a batch of 24 objects in range (1000, 1023)... Succeeded
Successfully deleted 1024/1024 objects in 6.0487249s
```

图 17 使用 S3 bench 进行 Ceph 性能观测 8

9 client 数量比 8 增加一倍，尾延迟显著增加，带宽几乎不变。

```
run-s3bench.cmd - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
@rem -accessKey Access Key
@rem -accessSecret Secret Key
@rem -bucket=loadgen Bucket for holding all test objects.
@rem -endpoint=http://127.0.0.1:9000 Endpoint URL of object storage service being tested.
@rem -numClients=8 Simulate 8 clients running concurrently.
@rem -numSamples=256 Test with 256 objects.
@rem -objectNamePrefix=loadgen Name prefix of test objects.
@rem -objectSize=1024 Size of test objects.
@rem -verbose Print latency for every request.

s3bench.exe ^
-accessKey=OVNIWZ28T357PS4QKF4Y ^
-accessSecret=LZabxBBjxG6rw2LQeQLyKHlQbdgM7viLMhHRpQUd ^
-bucket=loadgen ^
-endpoint=http://10.0.0.111:8000 ^
-numClients=32 ^
-numSamples=1024 ^
-objectNamePrefix=loadgen ^
-objectSize=409600
pause

C:\WINDOWS\system32\cmd.exe
numSamples: 1024
verbose: %d(bool=false)

Generating in-memory sample data... Done (949,24s)
Running Write test...
Running Read test...

Test parameters
endpoint(s): [http://10.0.0.111:8000]
bucket: loadgen
objectNamePrefix: loadgen
objectSize: 0.3906 MB
numClients: 32
numSamples: 1024
verbose: %d(bool=false)

Results Summary for Write Operation(s)
Total Transferred: 400.000 MB
Total Throughput: 54.03 MB/s
Total Duration: 7.403 s
Number of Errors: 0
-----
Write times Max: 1.160 s
Write times 99th %ile: 0.535 s
Write times 90th %ile: 0.329 s
Write times 75th %ile: 0.269 s
Write times 50th %ile: 0.218 s
Write times 25th %ile: 0.171 s
Write times Min: 0.051 s

Results Summary for Read Operation(s)
Total Transferred: 400.000 MB
Total Throughput: 129.63 MB/s
Total Duration: 3.086 s
Number of Errors: 0
-----
Read times Max: 0.509 s
Read times 99th %ile: 0.269 s
Read times 90th %ile: 0.217 s
Read times 75th %ile: 0.163 s
Read times 50th %ile: 0.064 s
Read times 25th %ile: 0.031 s
Read times Min: 0.006 s

Cleaning up 1024 objects...
Deleting a batch of 1000 objects in range (0, 999)... Succeeded
Deleting a batch of 24 objects in range (1000, 1023)... Succeeded
Successfully deleted 1024/1024 objects in 6.0680681s
```

图 18 使用 S3 bench 进行 Ceph 性能观测 9

实验三：尾延迟

使用 Amazon S3 python 版本连接 Ceph 对象存储网关进行延迟搜集。
测试 1 结果如图 19 所示。

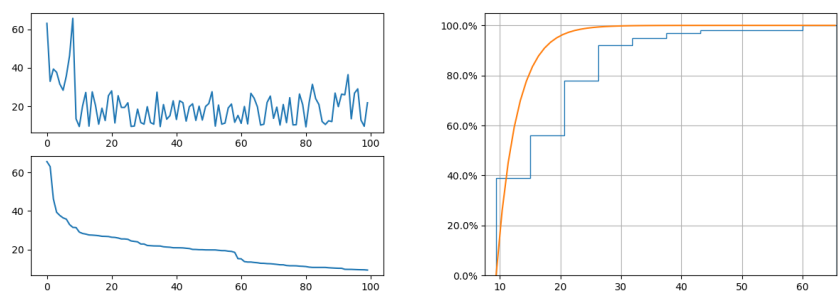


图 19 使用 Amazon S3 python 版本连接 Ceph 对象存储网关进行延迟搜集结果 1
测试 2 结果如图 20 所示。

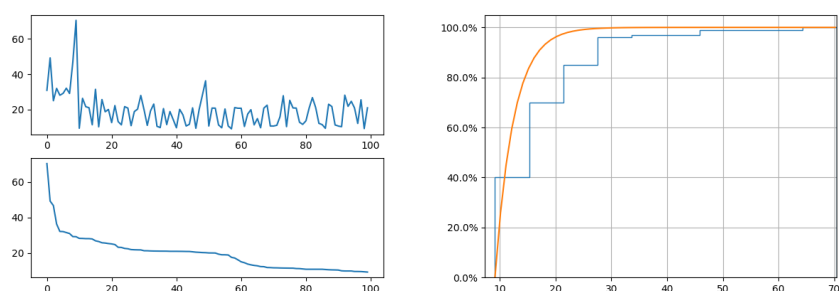


图 20 使用 Amazon S3 python 版本连接 Ceph 对象存储网关进行延迟搜集结果 2

使用与在 Minio 系统中相同的对冲请求策略，设置阈值为 45ms，结果如图 21 和 22 所示。

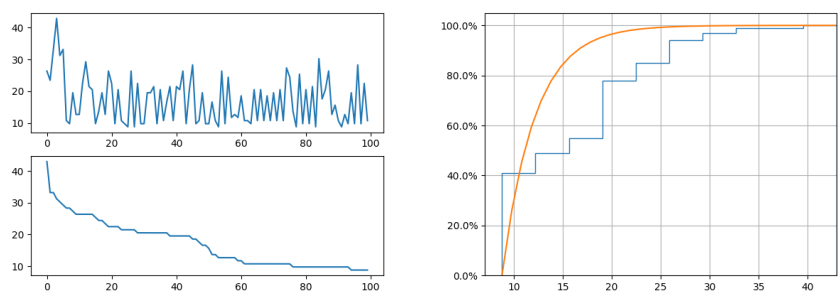


图 21 使用 Amazon S3 python 版本连接 Ceph 对象存储网关使用对冲请求后的延迟结果 1

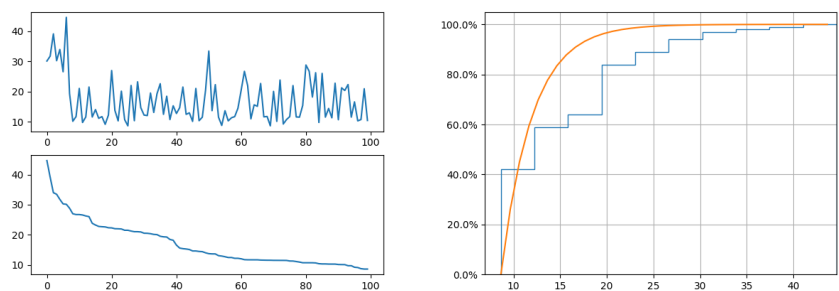


图 22 使用 Amazon S3 python 版本连接 Ceph 对象存储网关使用对冲请求后的延迟结果 2

从两次实验结果来看，尾延迟明显降低，从之前的 60ms-70ms 之前降低至 40ms-45ms 之间。