COSBench: A benchmark Tool for Cloud Object Storage Services

Jiangang.Duan@intel.com

2012.10

Agenda

- Self introduction
- COSBench Introduction
- Case Study to evaluate OpenStack* swift performance with COSBench
- Next Step plan and Summary

Self introduction

- Jiangang Duan
- Working in **Cloud Infrastructure Technology Team (CITT)** of Intel APAC R&D Ltd. (shanghai)
- We are software team, Experienced at performance
- To understand how to build an efficient/scale Cloud Solution with Open Source software (OpenStack*, Xen*, KVM*)
- All of work will be contributed back to Community
- Today we will talk about some efforts we try to measure OpenStack* Swift performance

COSBench Introduction

- COSBench is an Intel developed **Benchmark** to measure **Cloud Object Storage Service** performance
  - For S3, OpenStack Swift like Object Storage
  - Not for File system (NFS e.g) and Block Device system (EBS e.g.)
- Requirement of a benchmark to measure Object Store performance
- Cloud solution provider can use it to
  - Compare different Hardware/Software Stacks
  - Identify bottleneck and make optimization
- We are in progress to Open Source COSBench

COSBench is the IOMeter for Cloud Object Storage service

COSBench Key Component

Config.xml:

- define workload with flexibility.

Controller:

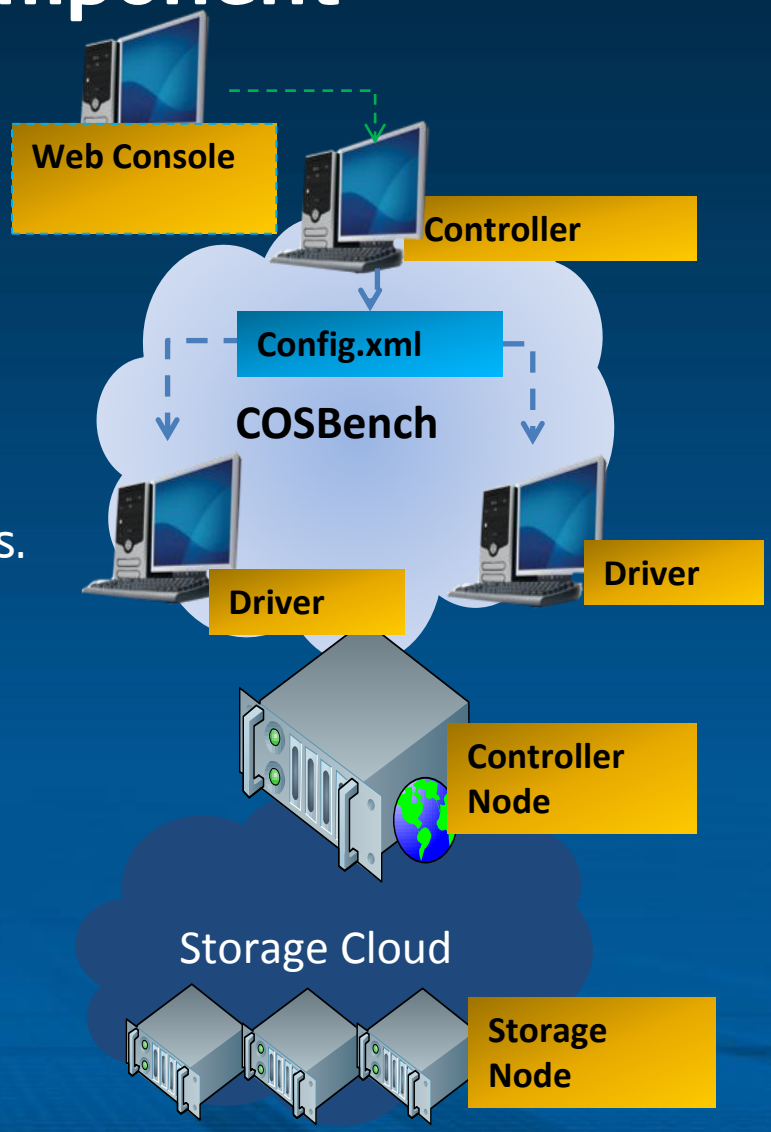
- Control all drivers
- Collect and aggregate stats.

Driver:

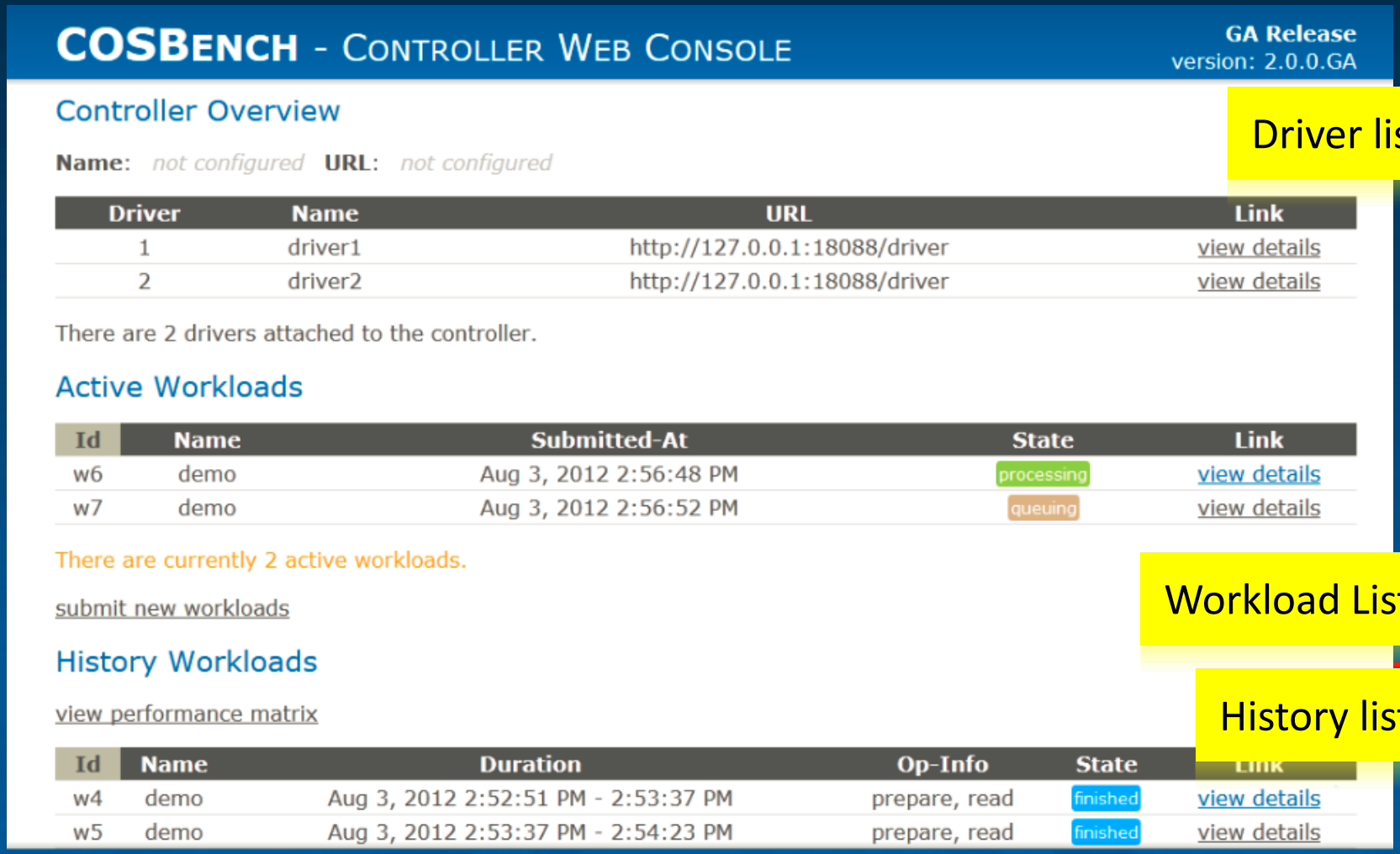
- generate load w/ config.xml parameters.
- can run tests w/o controller.

Web Console:

- Manage controller
- Browse real-time stats
- Communication is based on HTTP (RESTful style)



Web Console



Driver list

Workload List

History list

Intuitive UI to get Overview.

Workload Configuration

Flexible load control

```
- <workflow>
  - <workstage name="init">
      <work type="init" workers="8" config="containers=r(1,32)" />
    </workstage>
  - <workstage name="prepare">
      <work type="prepare" workers="8" config="containers=r(1,32);objects=r(1,50);sizes=c(64)KB" />
    </workstage>
  - <workstage name="main">
    - <work name="main" workers="8" rampup="100" runtime="300">
        <operation type="read" ratio="80" config="containers=u(1,32);objects=u(1,50)" />
        <operation type="write" ratio="20" config="containers=u(1,32);objects=u(51,100);sizes=c(64)KB" />
      </work>
    </workstage>
  - <workstage name="cleanup">
      <work type="cleanup" workers="8" config="containers=r(1,32);objects=r(1,50)" />
    </workstage>
  - <workstage name="dispose">
      <work type="dispose" workers="8" config="containers=r(1,32)" />
    </workstage>
  </workflow>
</workload>
```

object size distribution

Read/Write Operations

Workflow for complex stages

Flexible configuration parameters is capable of complex Cases

Performance Metrics

General Report

Op-Type	Op-Count	Byte-Count	Avg-ResTime	Throughput	Bandwidth	Succ-Ratio
read	12.58 kops	12.28 MiB	10.14 ms	628.84 op/s	628.84 KiB/S	100%
write	3.21 kops	200.88 MiB	10.09 ms	160.71 op/s	10.04 MiB/S	100%

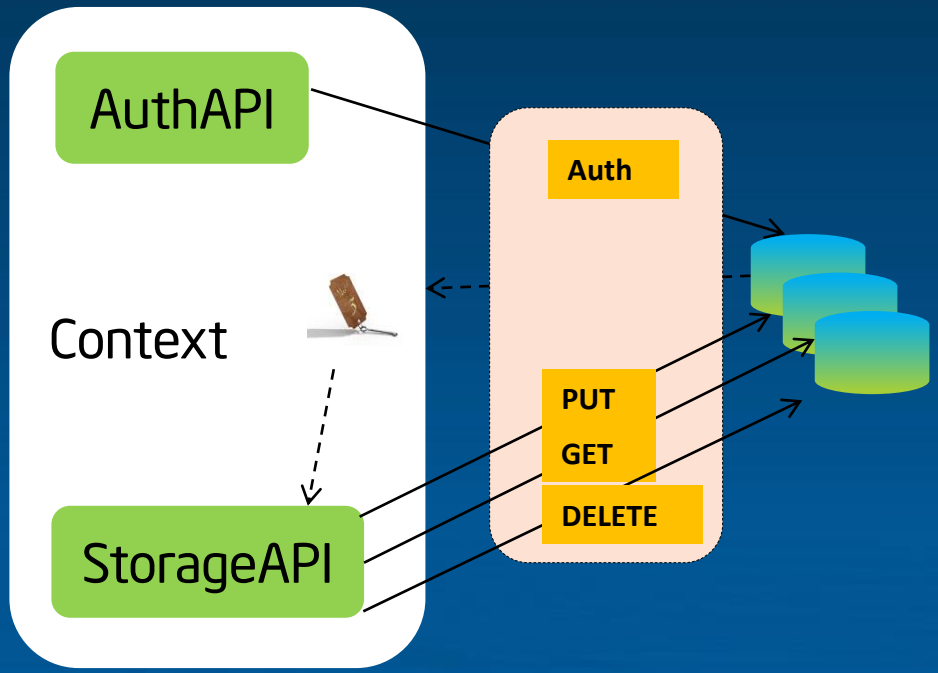
ResTime (RT) Details

Op-Type	60%-RT	80%-RT	90%-RT	95%-RT	99%-RT	100%-RT
read	< 20 ms	< 20 ms	< 20 ms	< 20 ms	< 20 ms	< 60 ms
write	< 20 ms	< 20 ms	< 20 ms	< 20 ms	< 20 ms	< 50 ms

- **Throughput (Operations/s):** the operations completed in one second
- **Response Time (in ms):** the duration between operation initiation and completion.
- **Bandwidth (KB/s):** the total data in KiB transferred in one second
- **Success Ratio (%):** the ratio of successful operations

Easy to be Extended

- Easily extend for new storage system:
- Support
  - OpenStack Swift
  - Amplistor*
  - Adding More

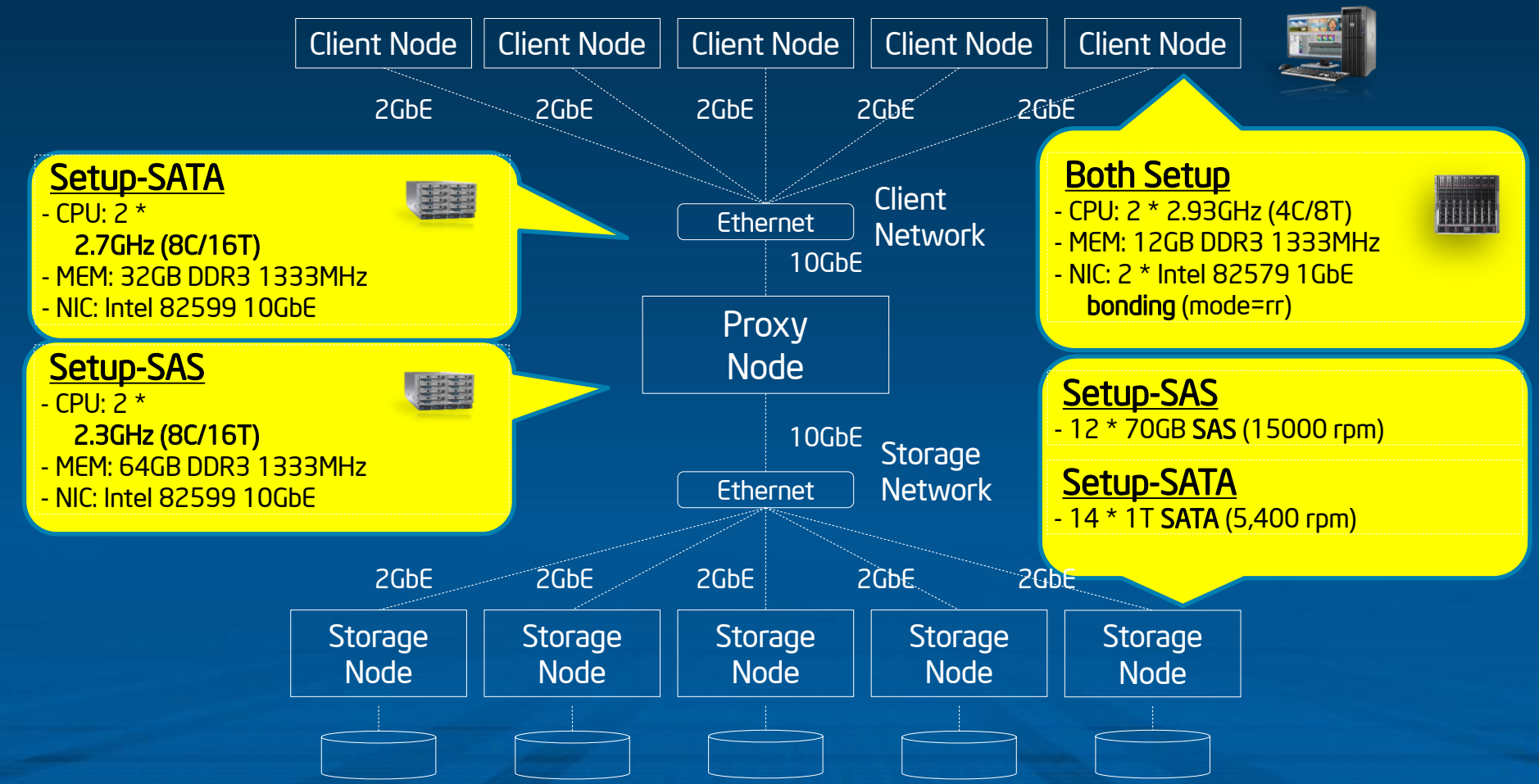


Easy execution engine is capable of complex cases.

System Configuration

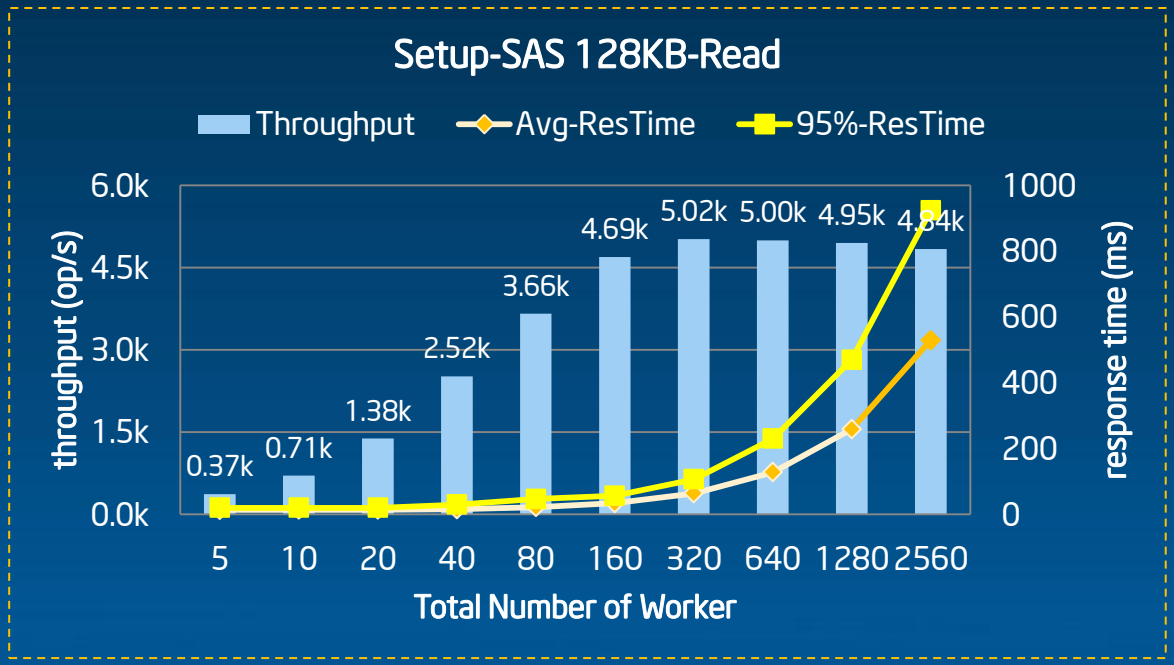
Setup-SATA has higher CPU power

Setup-SAS has faster disks ★



128KB-Read

- SLA: 200ms + 128KB/1MBps = <u>325ms</u>



# Worker	95%-ResTime	Throughput
	ms	op/s
5	20.00	369.49
10	20.00	711.24
20	20.00	1383.30
40	30.00	2517.94
80	46.67	3662.71
160	56.67	4693.97
320	106.67	5019.85
640	230.00	4998.13
1280	470.00	4947.15
2560	923.33	4840.19

The **bottleneck** was identified to be the **proxy's CPU**

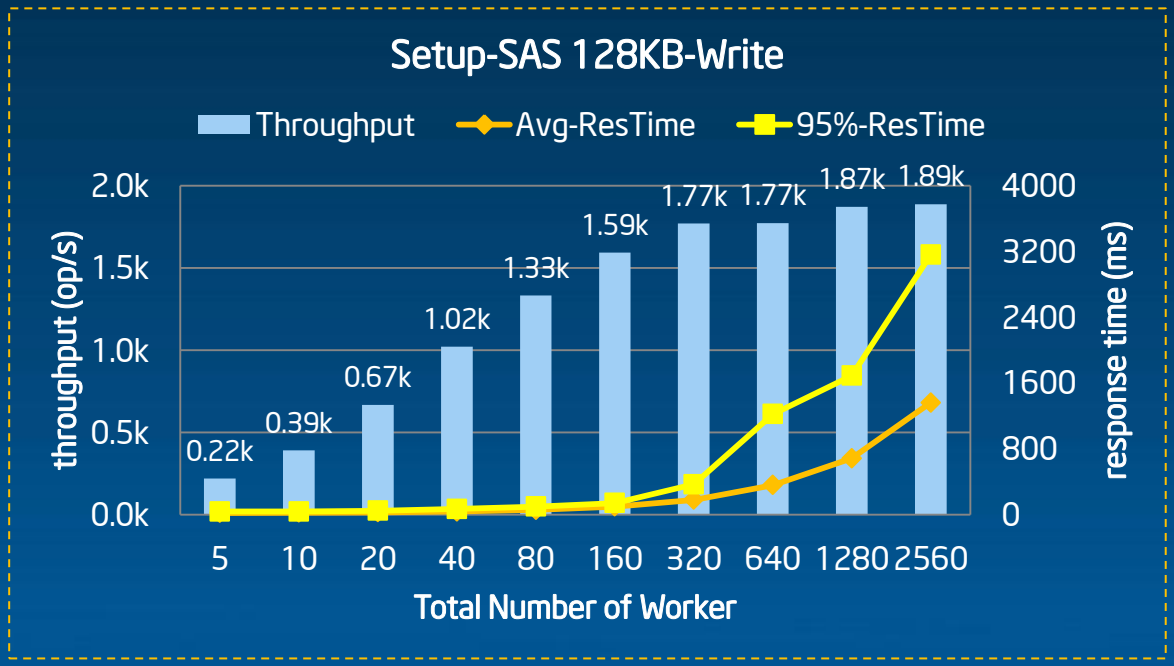
-- The CPU utilization at that node was ~100%!

-- The peak throughput for setup-SATA was 5576 op/s (640 workers)

Better CPU results in higher throughput

128KB-Write

- SLA: 200ms + 128KB/1MBps = 325ms



# Worker	95%-ResTime	Throughput
	ms	op/s
5	40.00	219.73
10	40.00	391.14
20	50.00	668.19
40	70.00	1022.07
80	100.00	1333.34
160	143.33	1594.12
320	370.00	1769.55
640	1223.33	1773.12
1280	1690.00	1871.58
2560	3160.00	1886.81

The **Disks** at storage nodes had significant impact on overall throughput

-- The peak throughput for setup-SATA was only **155** op/s (20 Workers)

-- even after we had put account/container DB files on SSD disks!

For more complete information about performance and benchmark results, visit Performance Test Disclosure

128KB-Write

To fully understand the write performance ...

-- From the **Disk** side: were storage disk a bottleneck?

-- in setup-SATA, all SSD → **1621** op/s compared with 155 op/s

-- Need do more tests to understand the disk impact

-- From the **NIC** side: were storage network a bottleneck?

-- in setup-SAS, two set of object daemon, → **NO** performance change (why? *TODO)*

-- in setup-SATA, all SSD + 32/64KB objects → **Storage node CPU** bottleneck (why? *TODO*)

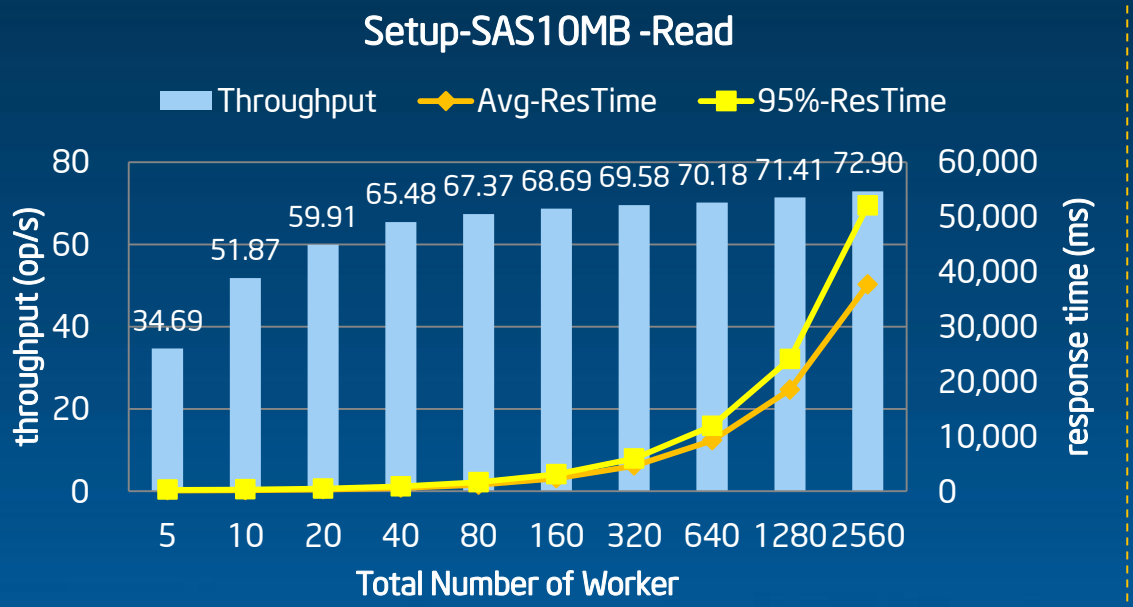
-- From the **SW** side: were container updating a bottleneck?

-- in setup-SATA, 1 container, put account/container from HDD to SSD → **316**% ↑

-- in setup-SATA, 100 containers, put account/container from HDD to SSD → **119**% ↑

10MB-Read

- SLA: 200ms + 10MB/1MBps = 1200ms



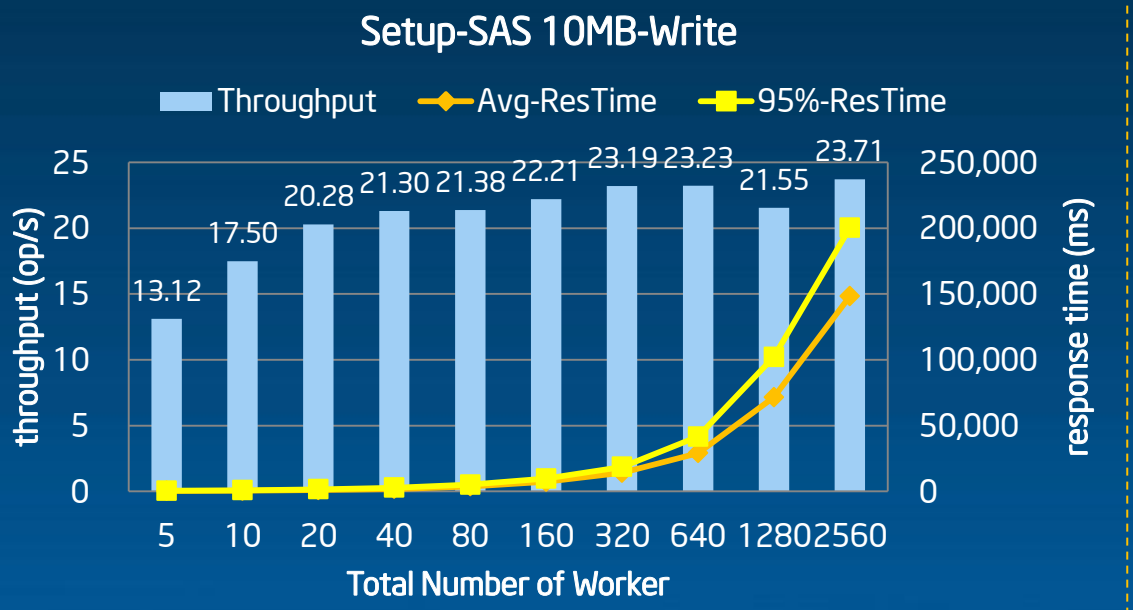
# Worker	95%-ResTime	Throughput
	ms	op/s
5	270.00	34.69
10	320.00	51.87
20	480.00	59.91
40	900.00	65.48
80	1636.67	67.37
160	3093.33	68.69
320	5950.00	69.58
640	11906.67	70.18
1280	24090.00	71.41
2560	52090.00	72.90

The **bottleneck** was identified to be the **clients' NIC BandWidth**

Double client receive bandwidth can double the throughput

10MB-Write

- SLA: 200ms + 10MB/1MBps = 1200ms



# Worker	95%-ResTime	Throughput
	ms	op/s
5	536.67	13.12
10	936.67	17.50
20	1596.67	20.28
40	2786.67	21.30
80	5133.33	21.38
160	9800.00	22.21
320	18623.33	23.19
640	41576.67	23.23
1280	102090.00	21.55
2560	200306.67	23.71

The **bottleneck** might be the **storage nodes' NICs**

-- in setup-SATA, the peak throughput was 15.74 op/s (10 clients)

-- in both setups, the write performance was 1/3 of the read performance

Next Step and call for action

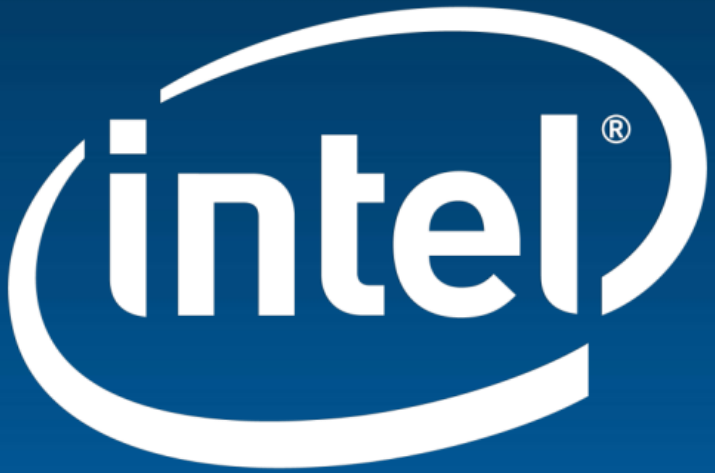
- Open source COSBench (WIP) –
- Keep develop Cosbench to support more Object Storage Software
- Take COSBench as a tool to analyze Cloud Object Service performance (swift and Ceph)
- Contact me (jiangang.duan@intel.com) if you want to evaluate COSBench – and we are glad to hear your feedback to make it better

Summary

- New storage Usage model rises for Cloud Computing age, which need new benchmark
- COSBench is a new benchmark developed by Intel to measure Cloud Object Storage service performance
- OpenStack Swift is stable/high open source performance object Storage implementation, still need improvement

Disclaimers

- INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.
- A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.
- Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.
- The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.
- Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.
- Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm%20
- This document contains information on products in the design phase of development.
- Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.
- *Other names and brands may be claimed as the property of others.
-


intel