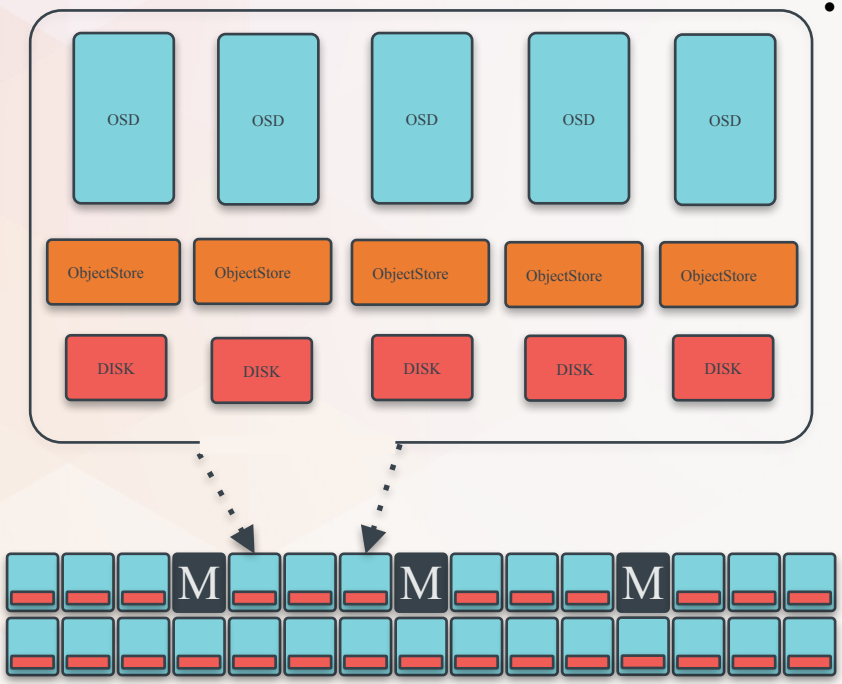
CEPHALOCON APAC 2018
THE FUTURE OF STORAGE
22-23 March 2018 | BEIJING

Performance tuning in BlueStore&RocksDB

Li Xiaoyan (Lisa), Intel

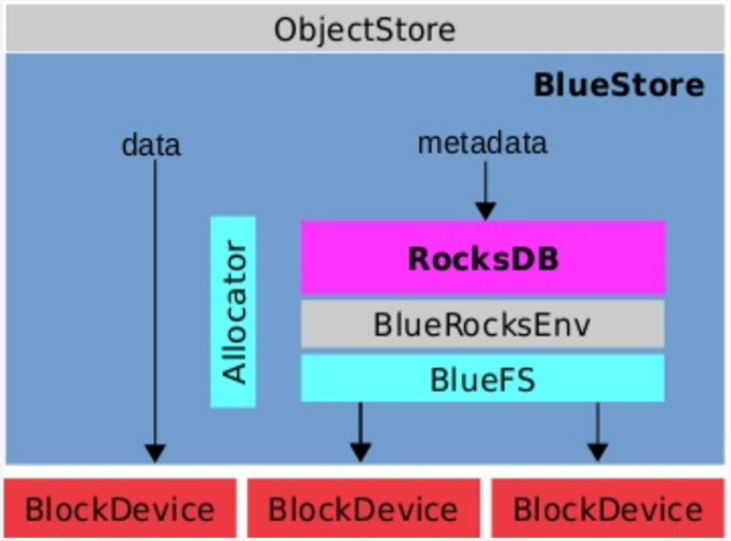
Overview

- ObjectStore is to store data in local nodes.
- BlueStore is default object store.



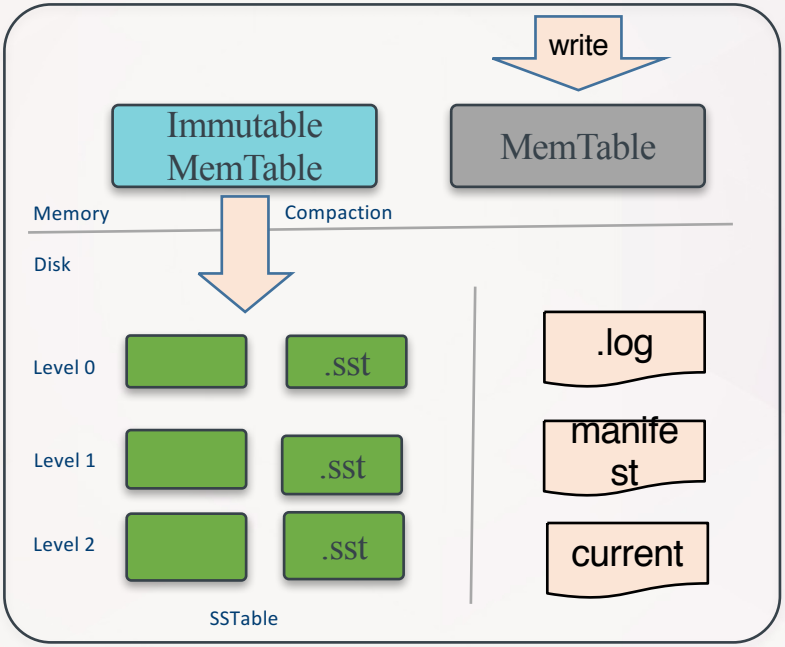
BlueStore

- BlueStore = Block + NewStore
- Data write directly to raw block device.
- Metadata write to Key/value database (Rocksdb).
  - Object data
  - Omap
  - Deferred logs
  - others
- RocksDB is above light weight file system BlueFS.



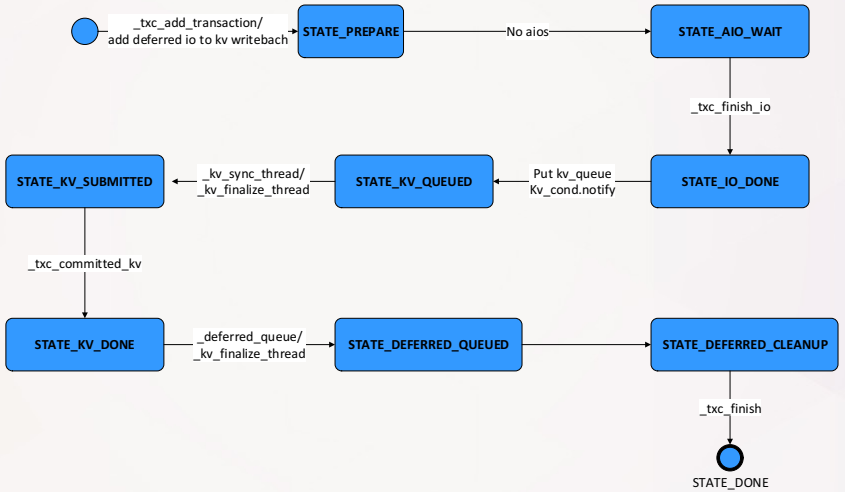
RocksDB

- A key-value database, originated by Google, improved by Facebook.
- Based on LSM (Log-Structure merge Tree).
- Key words:
  - Active MemTable
  - Immutable MemTable
  - SST file
  - LOG



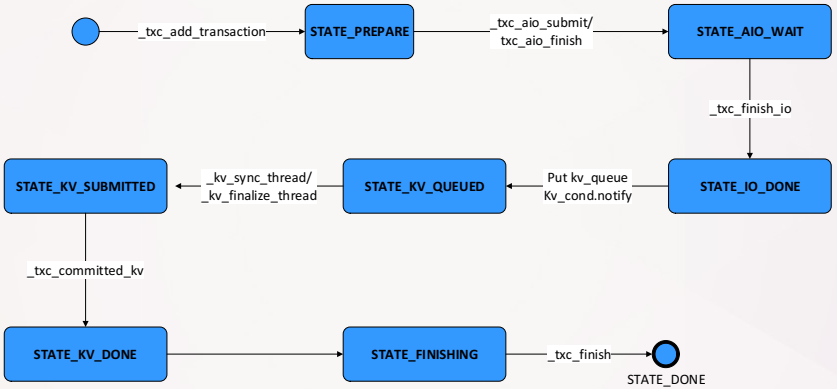
BlueStore – small write

- RocksDB acts as journal (deferred IO).
- Customer data is written to RocksDB, and return to OSD.
- Later customer data is written into block device.
- Deferred IO entry is deleted from RocksDB.



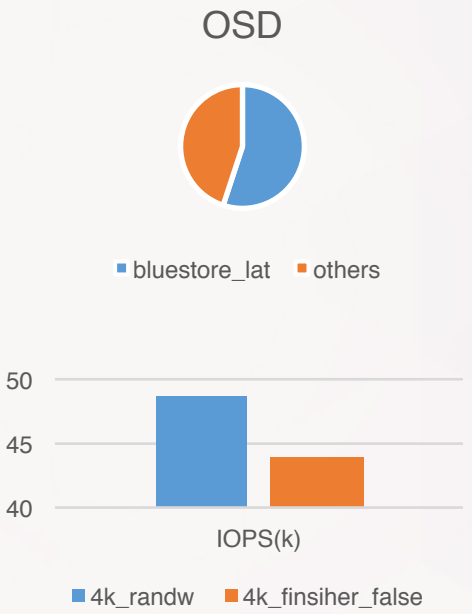
BlueStore – big write

- No journal is needed.
- Customer data is written into a new space.
- Return to OSD when metadata is written into RocksDB.
- The old space is released.



OSD tuning 1

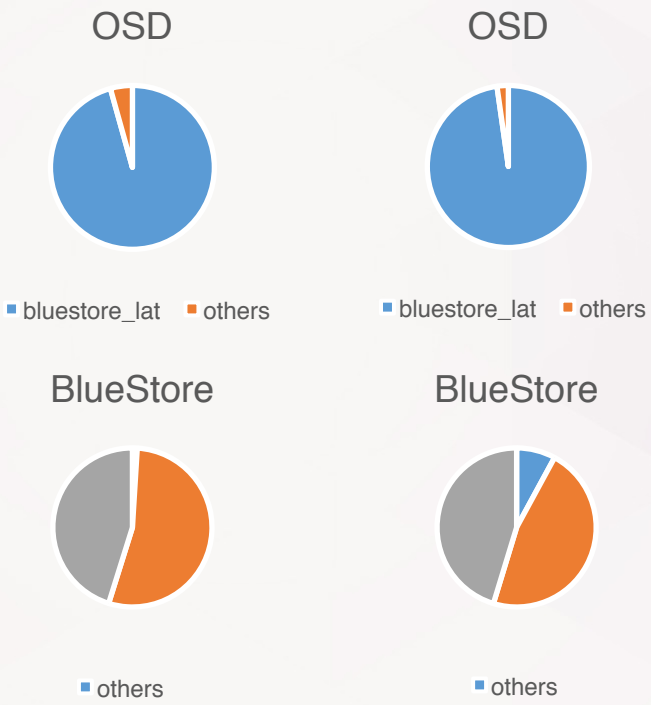
- 4k random writes.
- With default config (Perf dump data): top chart.
- BlueStore finisher is single-thread by default.
- After setting bluestore_shard_finishers: bottom charts.



Write IO time span

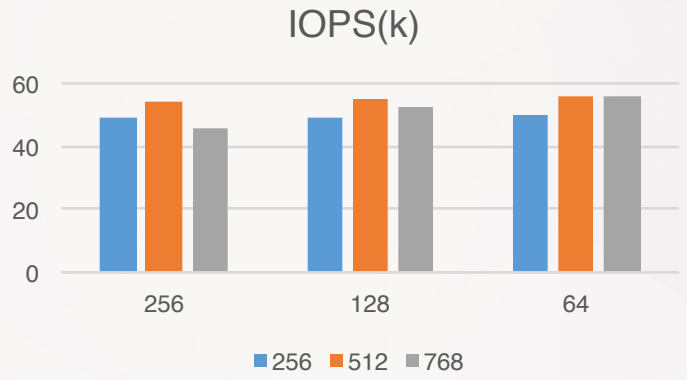
- Get time span from perf dump.
- OSD total latency: from OSD handles a IO in Messengers to commit the IO.
- BlueStore latency: from BlueStore gets a IO to commit te IO to OSD.

- Note: Left 4k random writes, right 16k random writes



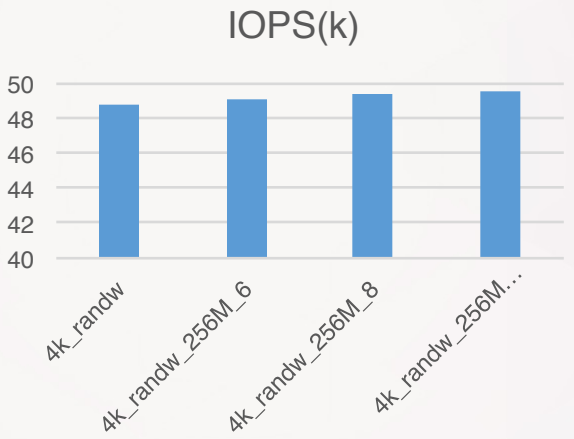
Random write tuning

- Keep total memory usage consistent.
- RocksDB options:
  - min_write_buffer_number_to_merge (default 1)
  - write_buffer_size (default 256MB), changed to 128, 64.
- 4k random writes.



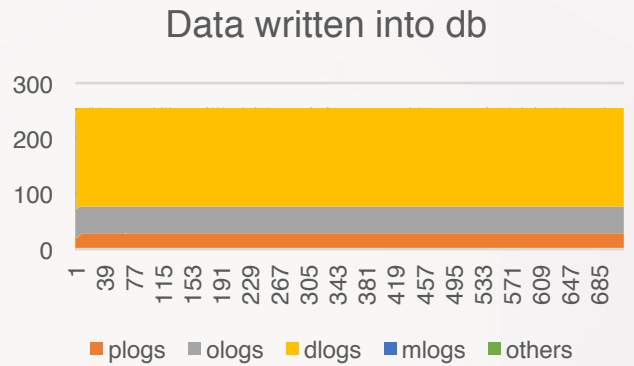
Random write tuning – cont.

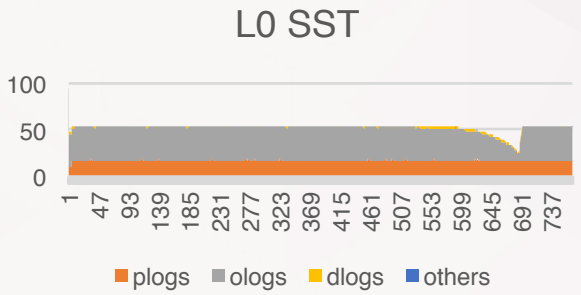
- Increase total memory usage consistent.
- RocksDB options:
  - max_write_buffer_number (default 8)
- The improvement is little.
- 4k random writes.



Random write – 4k

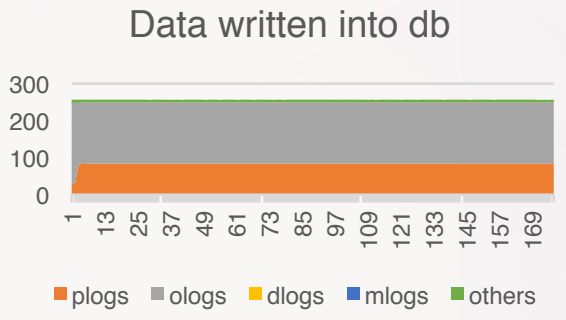
- When data is written into RocksDB, it is added into memTables.
- Once flush condition is triggered, data in memTables are flushed into L0 SST files.
- Deferred logs are main data in every memTable while object data are main data in L0 files.

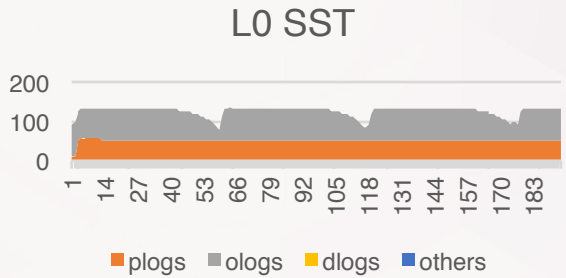




Random write – 16k

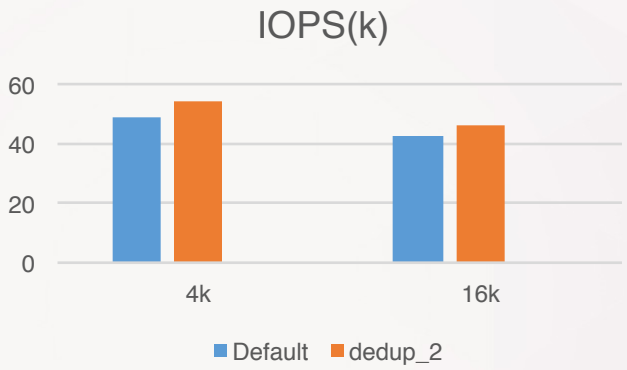
- Similar data as 4k random writes.
- Object data are main data both in every memTable and every L0 SST file.

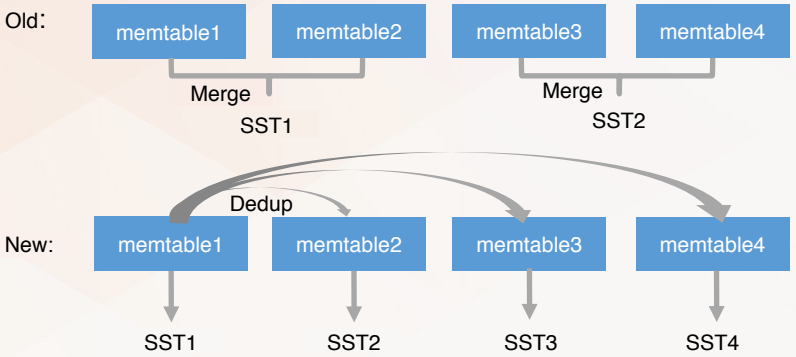




RocksDB - Flush data recursively.

- Random writes 4k/16k.
- Add a flush style: to delete duplicated entries recursively.
- Performance is similar as merge num = 2, but data written into L0 is decreased. (5G per 10mins)





Future work

- RocksDB is still heavy for pg logs, object data.
  - For pg logs data, they are written once, and read when the OSD node gets recovery.
  - For object data, one-time journal may be enough.

Thanks & QA

Backup – Test config

- Hardware
  - Memory: 128G
  - CPU: Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20GHz
  - Disk: Intel P3700 400G
- Software
  - Ceph master branch @f584df78c294b11baa7527d8eab0874ae6a2b809
- Config
  - A OSD, a monitor, and a manager.