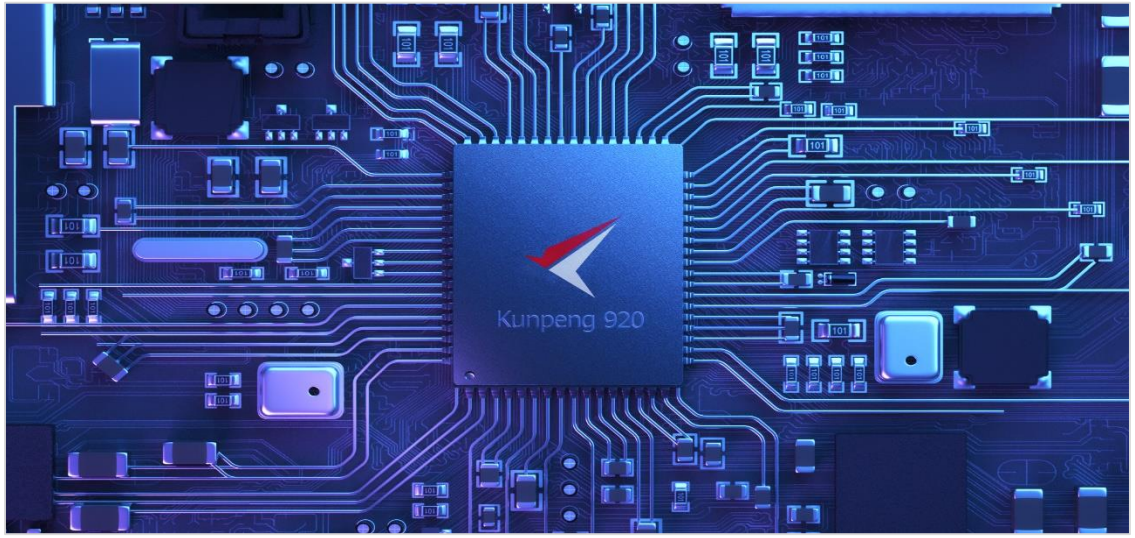
Performance Optimization for All Flash based on aarch64

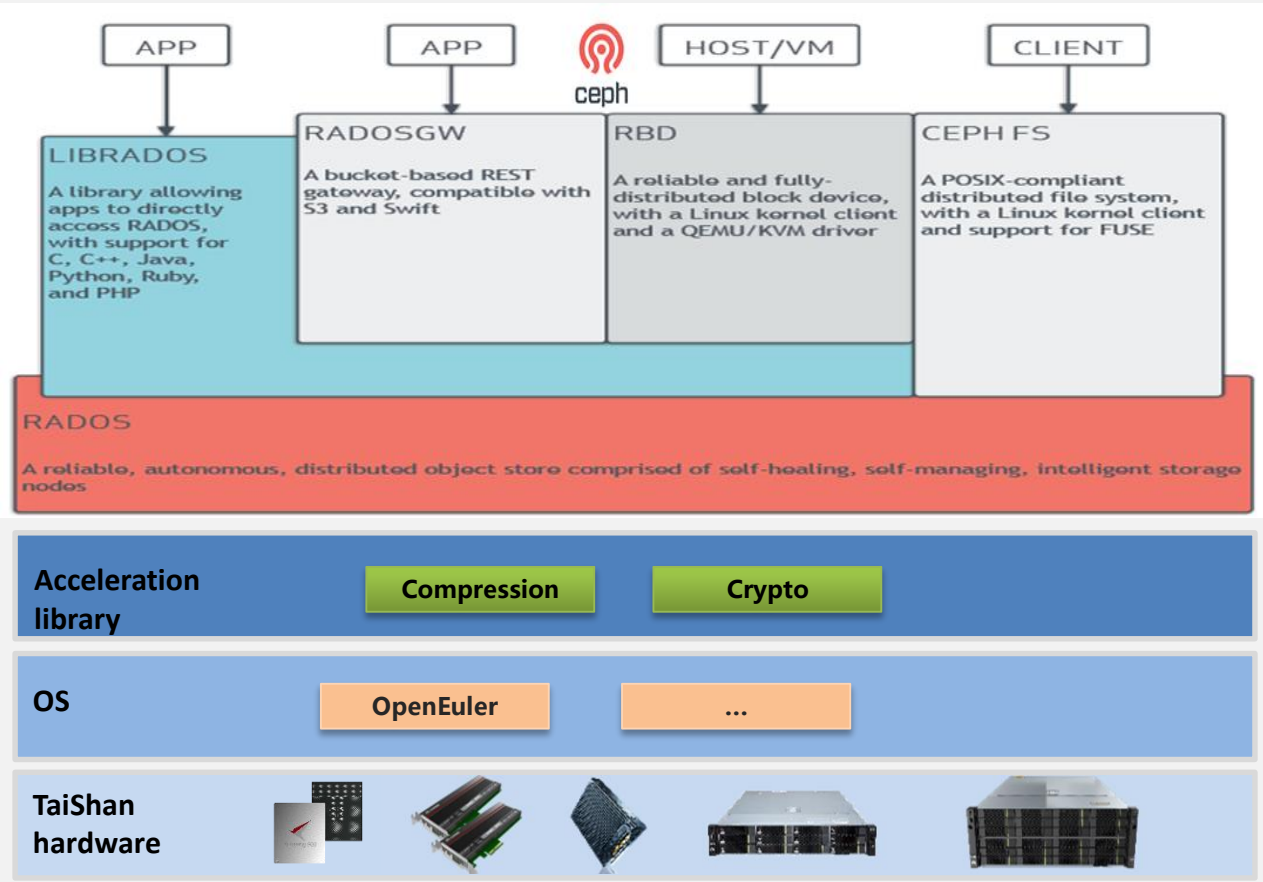
Huawei 2021/06/11

Ceph solution based on Kunpeng920



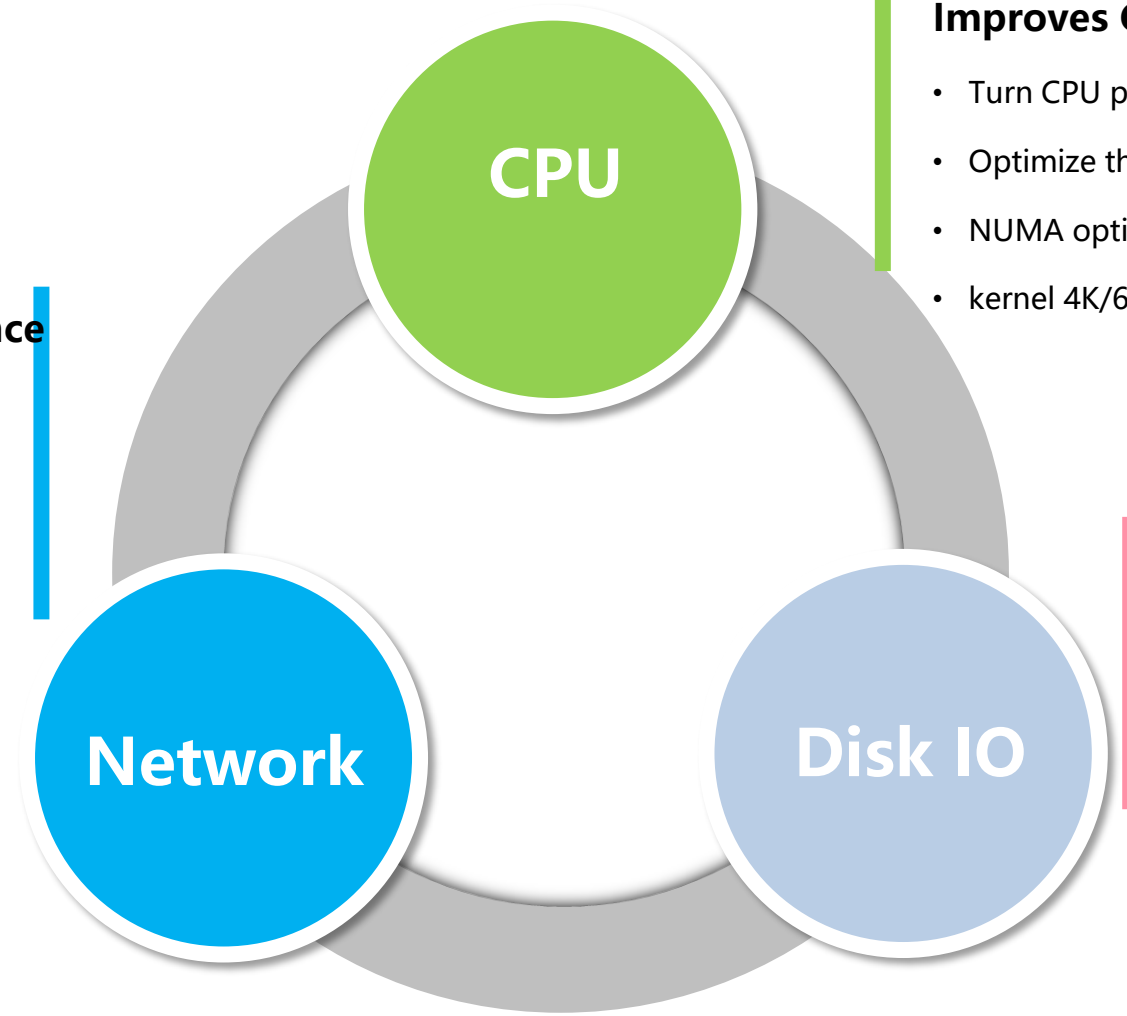
Kunpeng920 ARM-Based CPU
Industry's Most Powerful ARM-Based CPU

- Core Count:32/48/64 cores
- Frequency:2.6 /3.0GHz
- Memory controller : 8 DDR4 controllers
- Interface: PCIe 4.0, CCIX, 100G RoCE, SAS/SATA
- Process： 7nm



Ceph technical architecture based on Kunpeng chips

Optimizes performance through software and hardware collaboration



Improves CPU usage

- Turn CPU prefetching on or off
- Optimize the number of concurrent threads
- NUMA optimization
- kernel 4K/64K pagesize

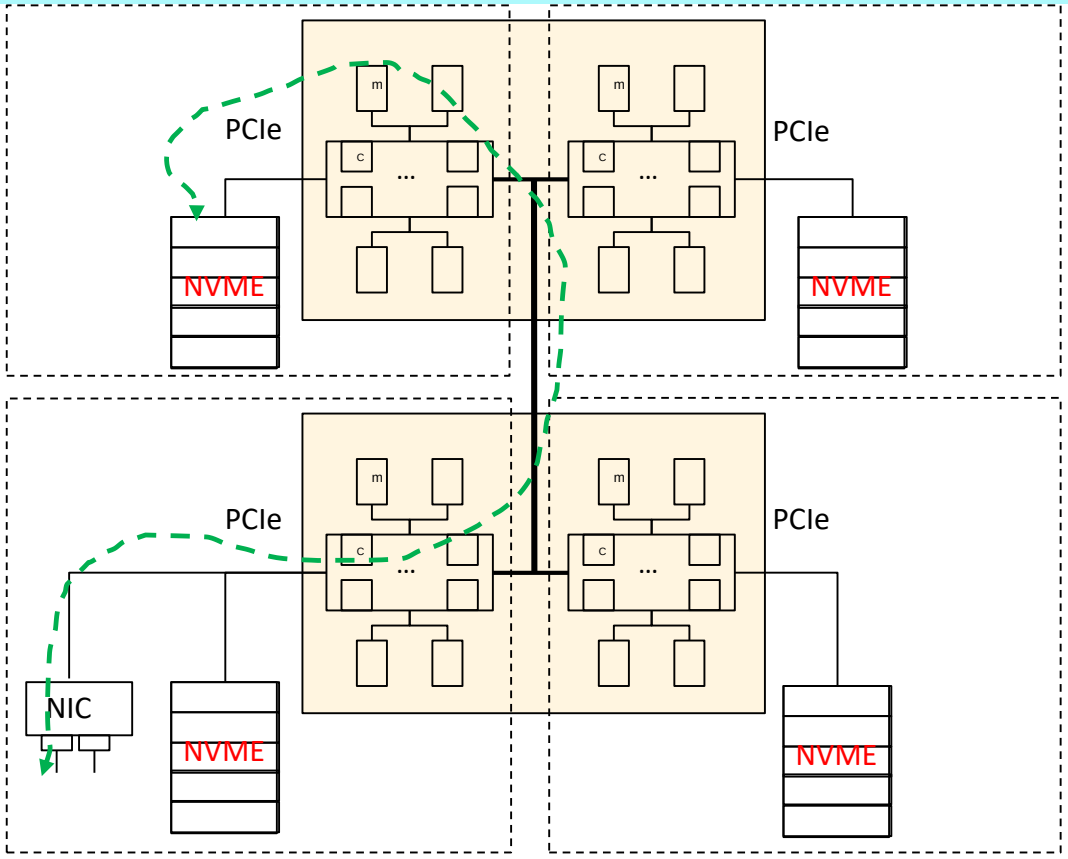
Optimizes NIC performance

- Interrupt core binding
- MTU adjustment
- TCP parameter adjustment

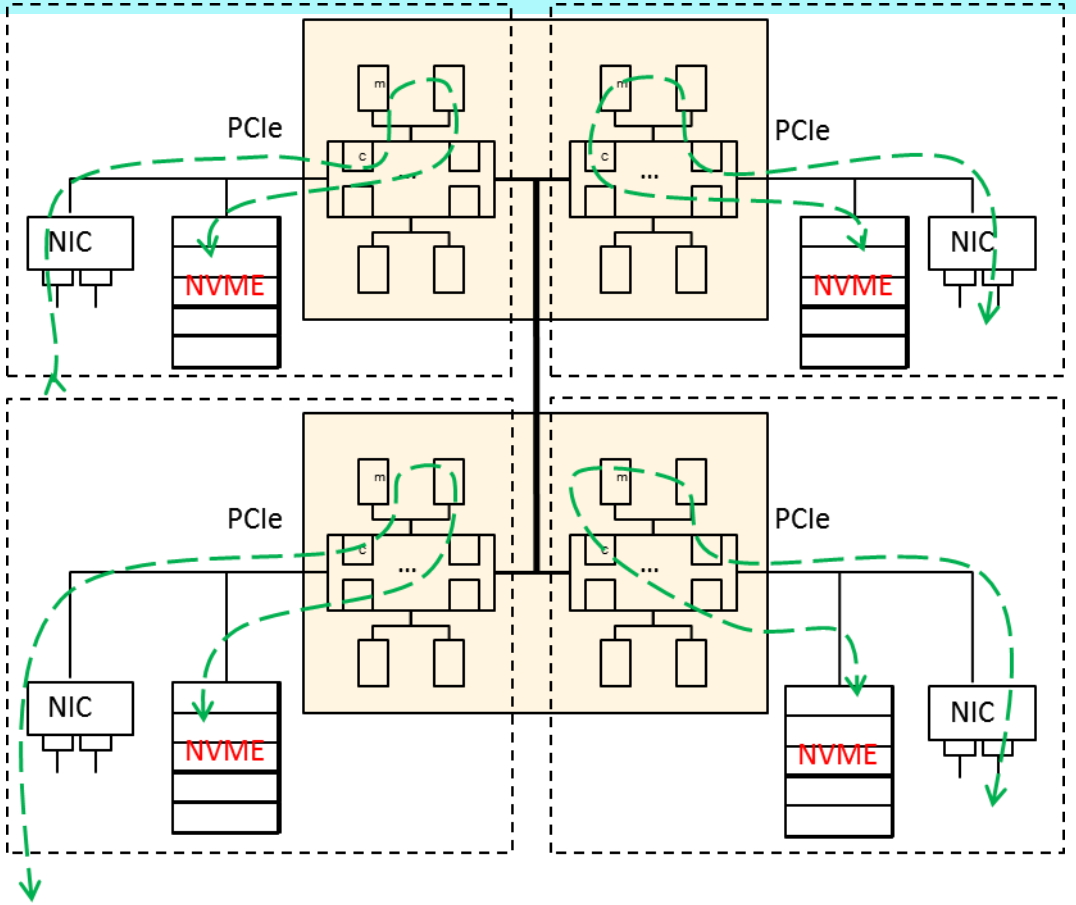
I/O performance optimization

Table of Contents

①Data access across NUMA



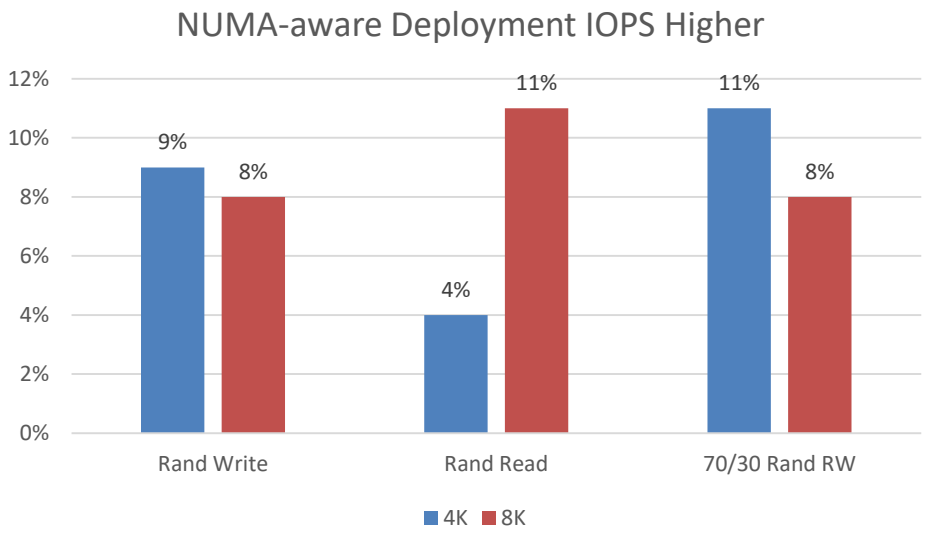
Data flows across NUMA

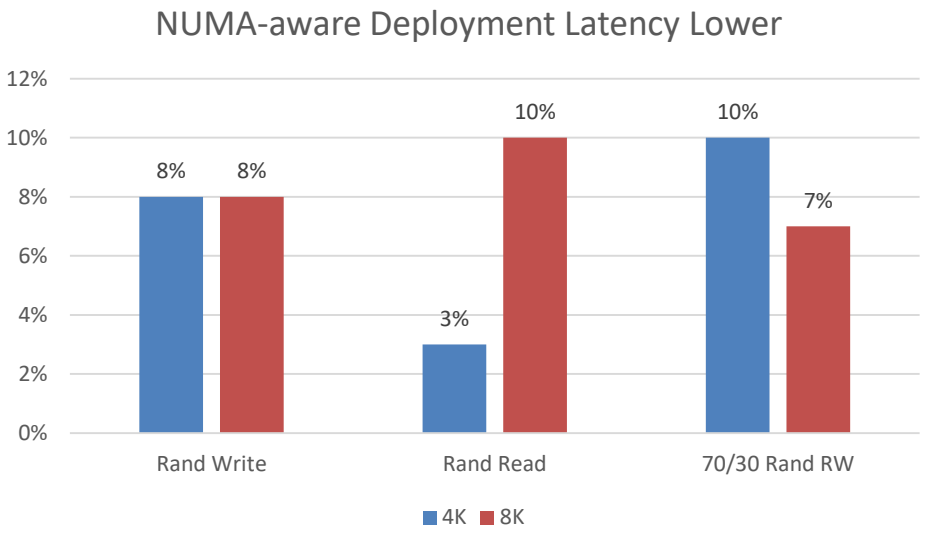


The data flow is completed within NUMA.

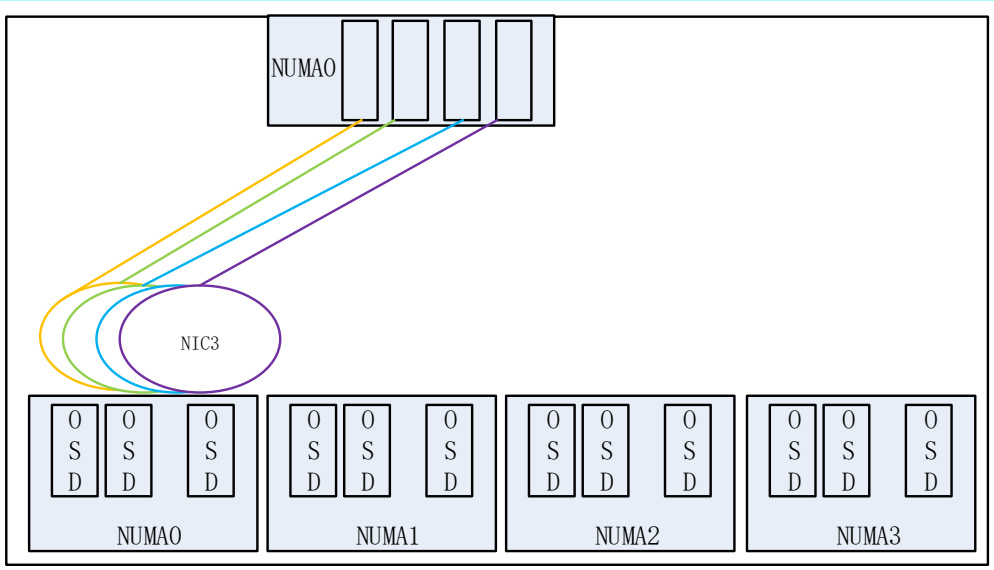
OSD NUMA-aware Deployment

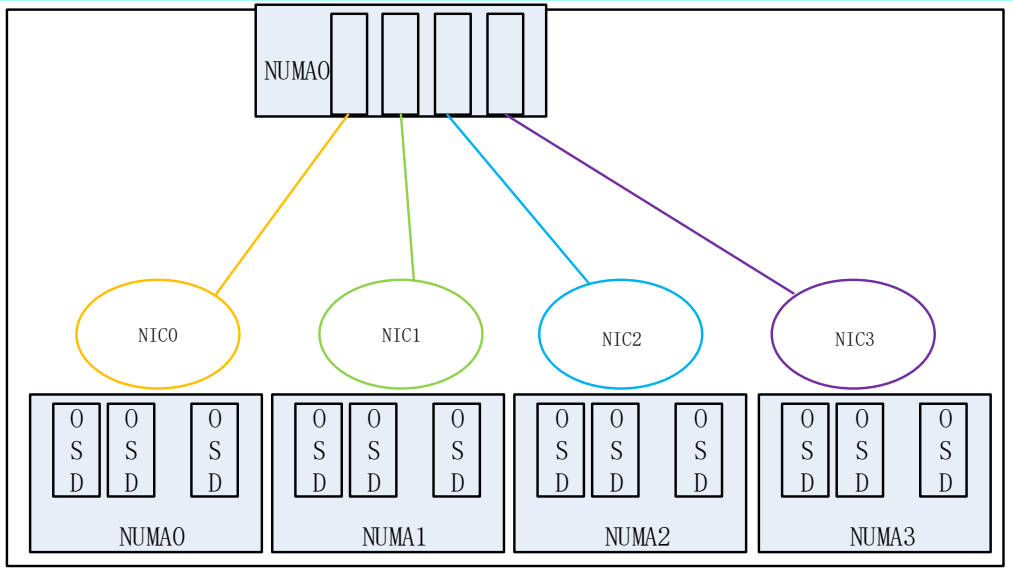
- OSDs are evenly deployed in NUMA nodes to avoid cross-NUMA scheduling and memory access overheads.

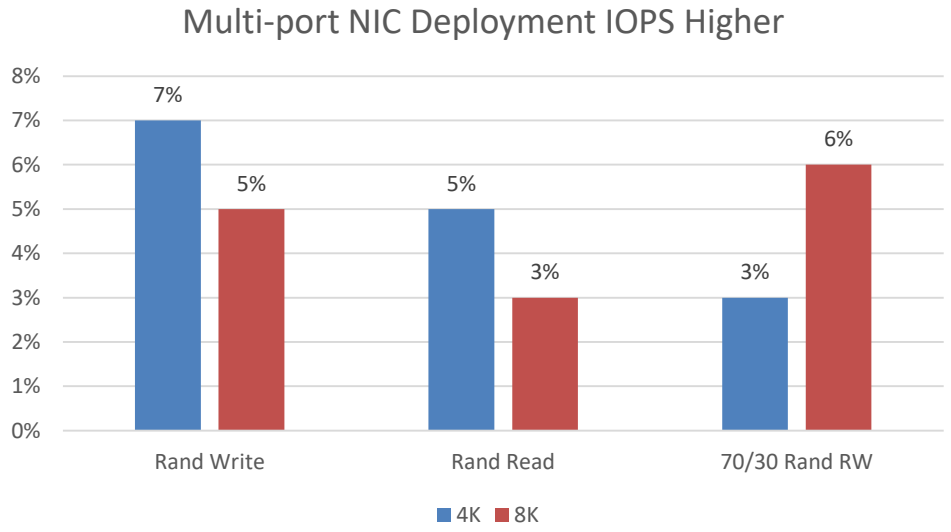


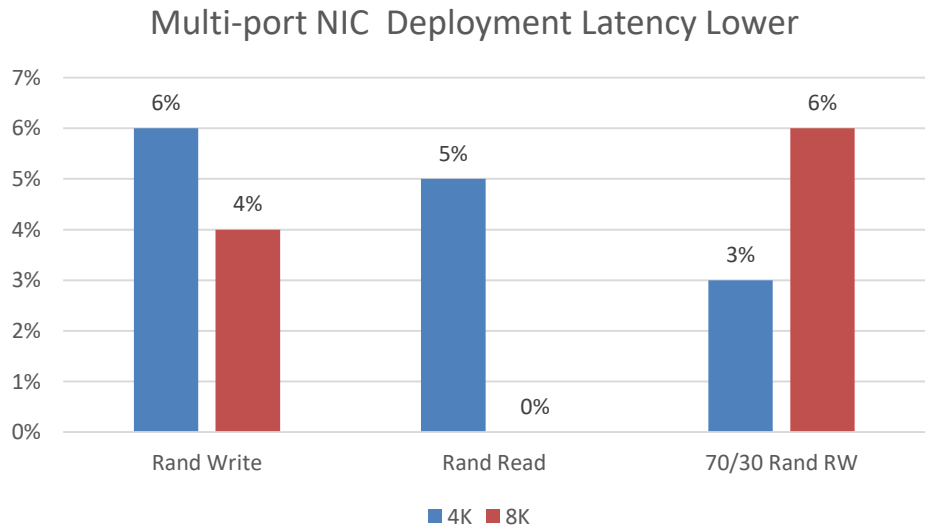


② Multi-port NIC deployment



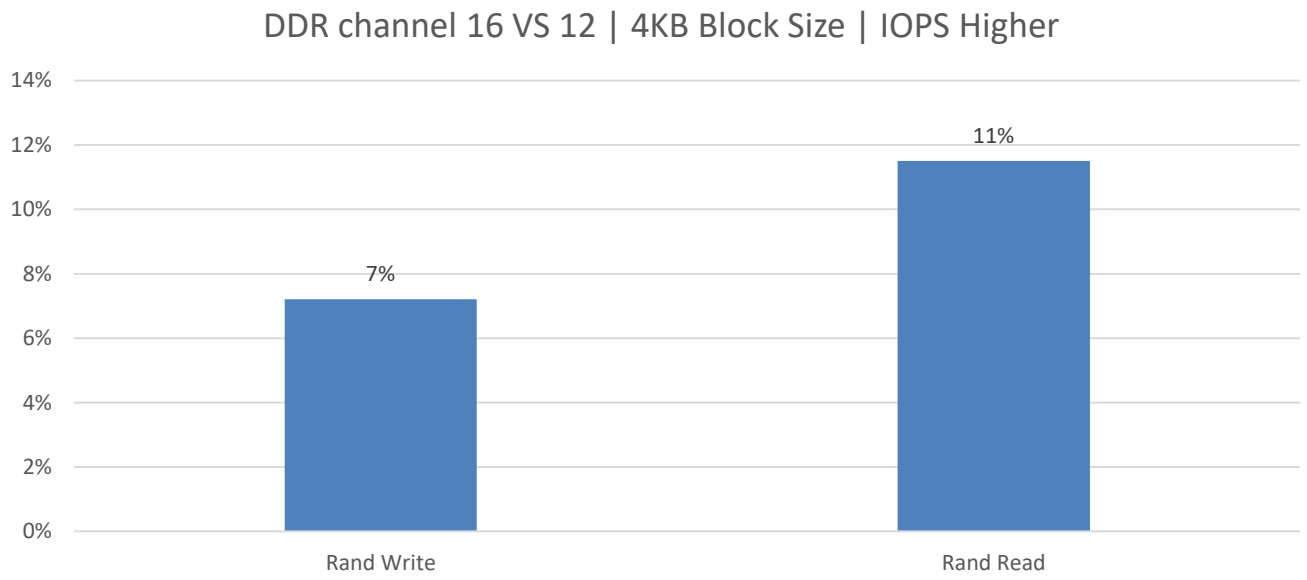






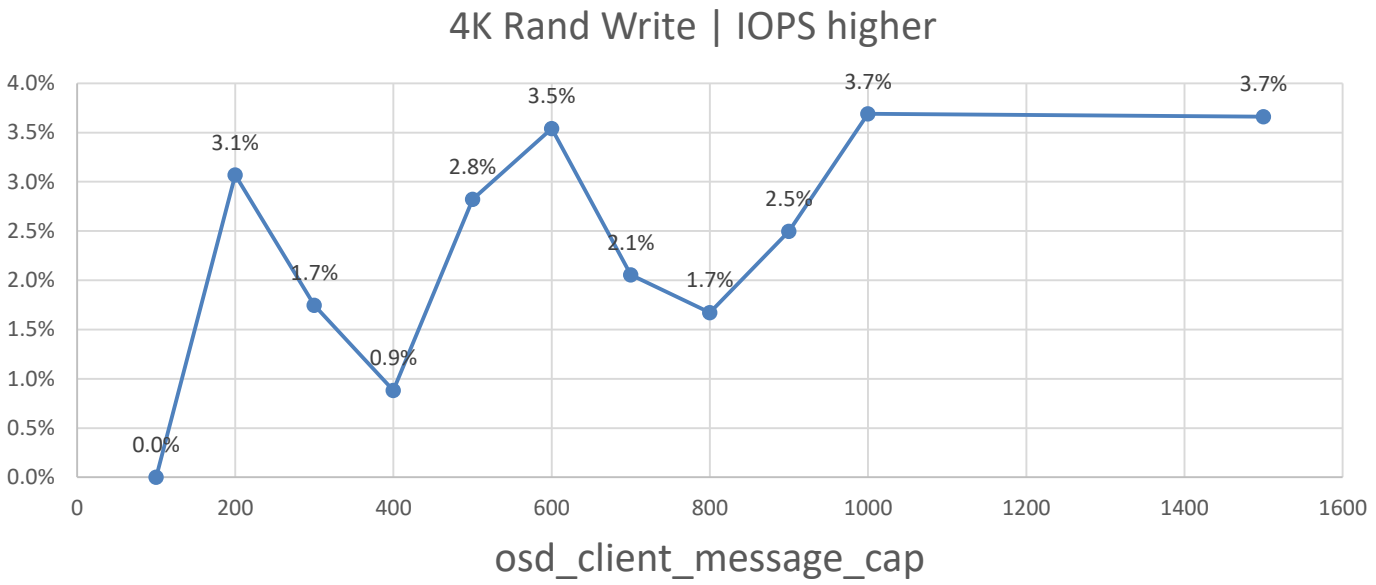
- NIC interrupts and OSD processes are bind to the same NUMA.
- NICs receive packets and OSD recvmsg in the same NUMA.

③ DDR Multi-channel connection



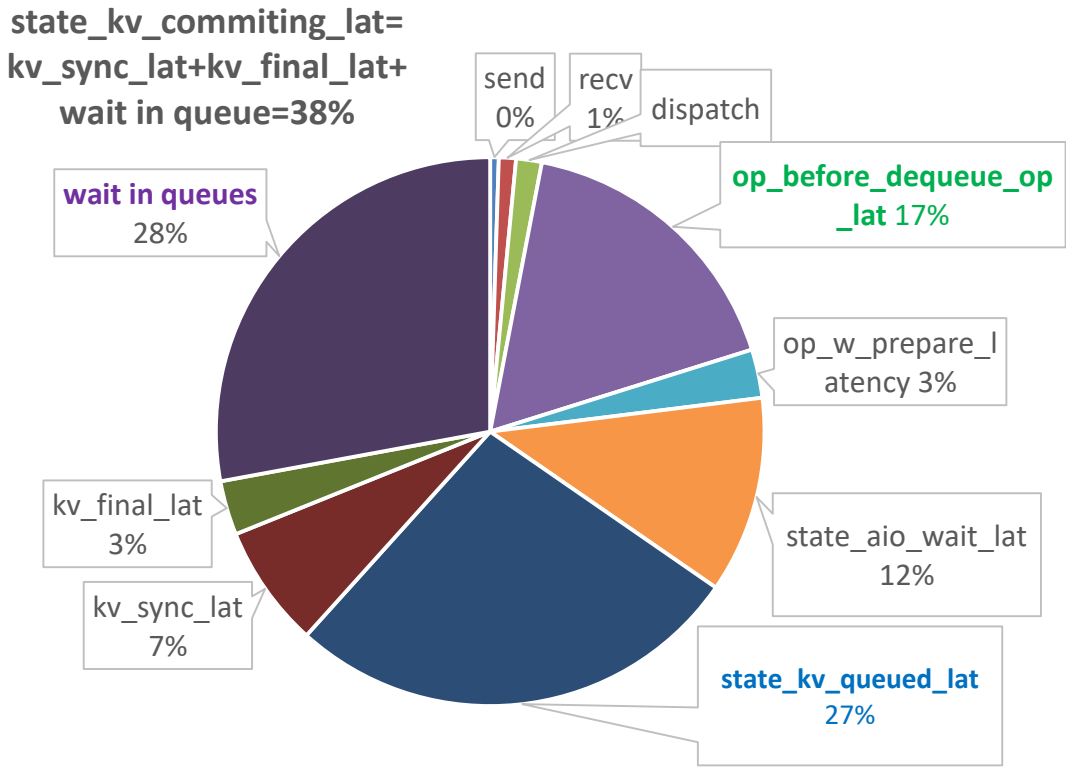
- 16-channels DDR is 7% higher than that of 12-channels DDR.

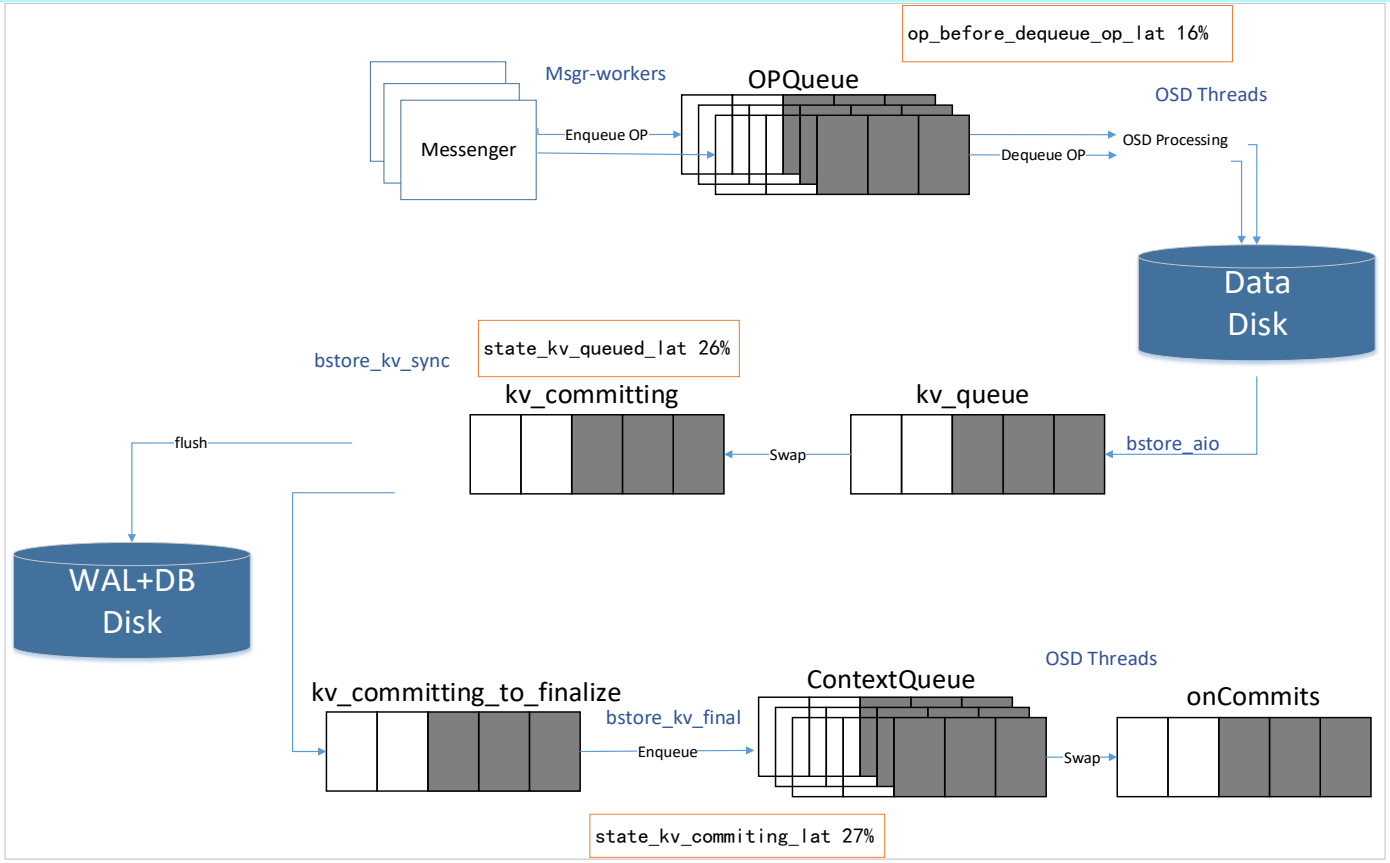
④Messenger throttler



- Gradually increase the osd_client_message_cap. When the value is 1000, the IOPS increases by 3.7%.

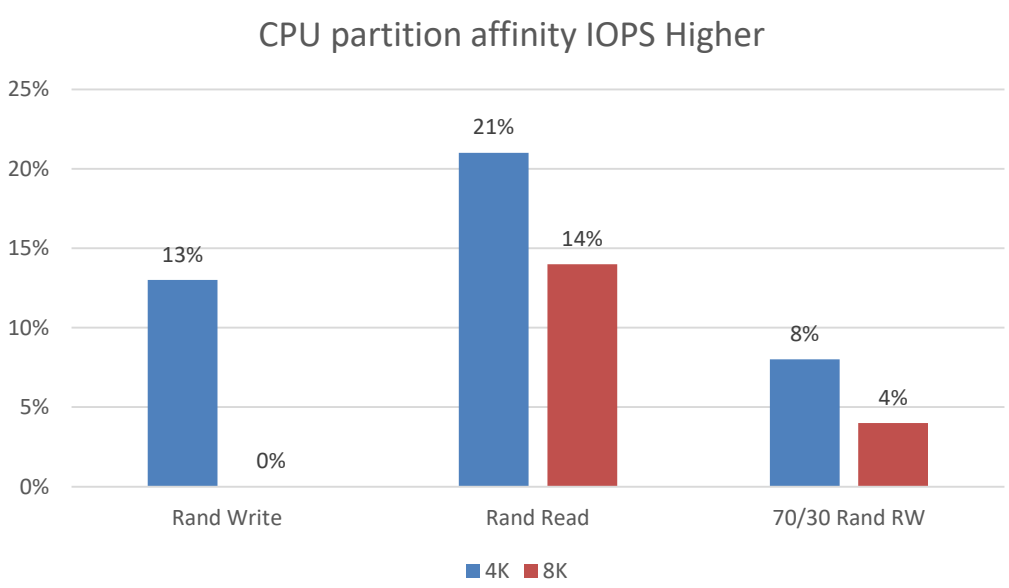
⑤ Long queue waiting time in the OSD

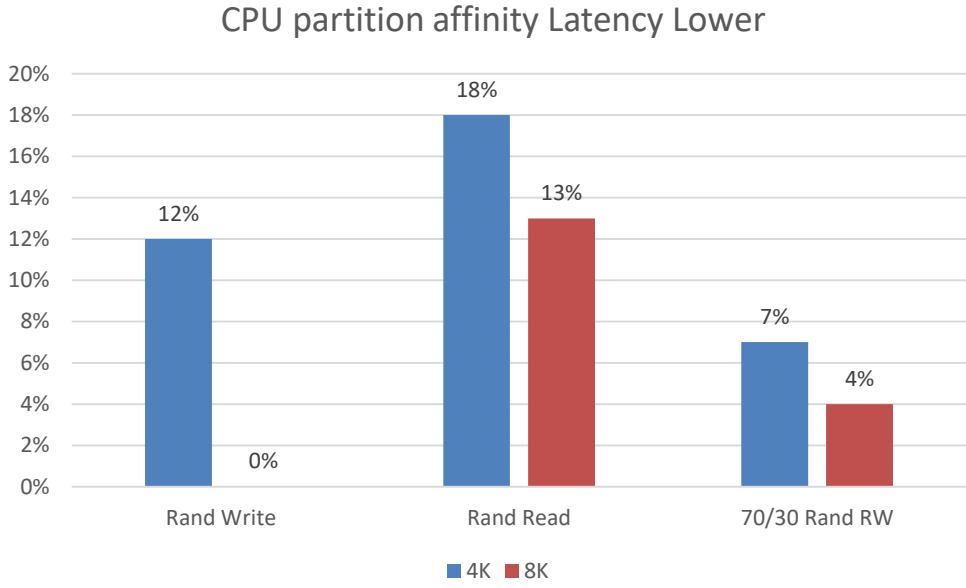




- The average length of the OPQueue queue is less than 10 and the queuing time is short. The length of kv_queue and kv_committing queues is greater than 30, and the queuing duration is long.

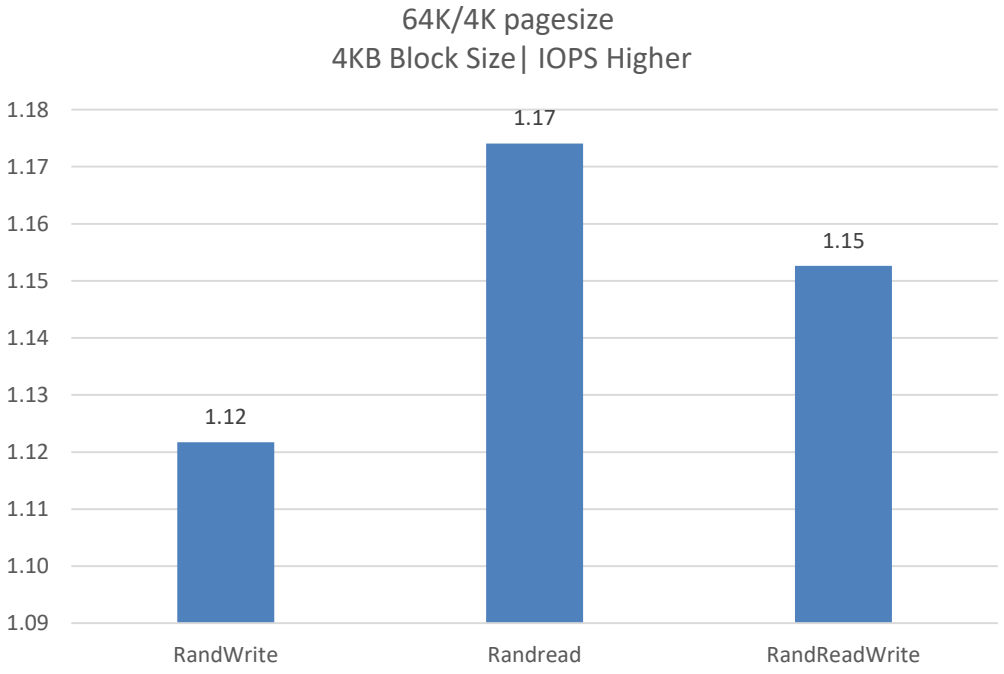
CPU partition affinity





- The CPU partition is used to separate the msgr-worker/tp_osd_tp and bstore threads to achieve fair scheduling.

⑥Use 64K PageSize



- Compared with 4K pages, 64K pages reduce TLB miss and improve performance by 10%+.

Tips:

- Use small page alignment to reduce memory waste in bufferlist::reserve

```
diff --git a/src/common/buffer.cc b/src/common/buffer.cc
index d9b32163db..8d7583e7ad 100644
--- a/src/common/buffer.cc
+++ b/src/common/buffer.cc
@@ -1269,7 +1269,7 @@ static ceph::spinlock debug_lock;
   void buffer::list::reserve(size_t prealloc)
   {
     if (get_append_buffer_unused_tail_length() < prealloc) {
-      auto ptr = ptr_node::create(buffer::create_page_aligned(prealloc));
+      auto ptr = ptr_node::create(buffer::create_small_page_aligned(prealloc));
       ptr->set_length(0);   // unused, so far.
       _carriage = ptr.get();
       _buffers.push_back(*ptr.release());
```

- Use CEPH_PAGE_SHIFT for compatibility with various page sizes

```
diff --git a/src/include/mempool.h b/src/include/mempool.h
index c03aa175cf..fe84f3b8f0 100644
--- a/src/include/mempool.h
+++ b/src/include/mempool.h
@@ -259,7 +259,7 @@ public:
     // Dirt cheap, see:
     //   https://fossies.org/dox/glibc-2.32/pthread__self_8c_source.html
     size_t me = (size_t)pthread_self();
-    size_t i = (me >> 12) & ((1 << num_shard_bits) - 1);
+    size_t i = (me >> CEPH_PAGE_SHIFT) & ((1 << num_shard_bits) - 1);
     return i;
   }
```

64K Pagesize Issues

■ Write Amplification Issue

When bluefs_buffered_io is set to true, metadata is written using buffer I/O, and sync_file_range is called to write data to disks by page in the kernel. The magnification factor is 2.46 for 4K pages and 5.46 for 64K pages.

When bluefs_buffered_io is set to false, metadata is written using direct I/O, sync_file_range is not called. The magnification factor is 2.29. Too many writes affect the disk life cycle. Therefore, set bluefs_buffered_io to false.

■ Tcmalloc and kernel page size issue

When the tcmalloc page size is smaller than the kernel page size, the memory keeps increasing until it approaches osd_memory_target, and the performance deteriorates significantly. Ensure that the page size of tcmalloc is greater than the kernel page size.

⑦ Optimized crc32c in rocksdb

- Rocksdb's crc32c_arm64 is supported since Ceph pacific. Backport this feature to earlier version, the performance is improved by about 3%.

```
Samples: 671K of event 'cycles', 4000 Hz, Event count (approx.): 39486291182 lost: 0/0 drop: 0/197
Overhead  Shared Object          Symbol
   8.83%  ceph-osd              [.] std::atomic<rocksdb::InlineSkipList<rocksdb::MemTableRep::KeyComp
   8.05%  ceph-osd              [.] rocksdb::ExtractUserKey
   7.51%  ceph-osd              [.] rocksdb::GetVarint32Ptr
   6.52%  ceph-osd              [.] rocksdb::Slice::compare
   6.21%  ceph-osd              [.] rocksdb::GetLengthPrefixedSlice
   5.39%  ceph-osd              [.] rocksdb::Slice::Slice
   3.52%  ceph-osd              [.] rocksdb::InternalKeyComparator::CompareKeySeq
   3.49%  libc-2.27.so          [.] memcmp
   3.28%  ceph-osd              [.] rocksdb::crc32c::Slow_CRC32
   3.04%  ceph-osd              [.] rocksdb::Slice::size
   2.75%  ceph-osd              [.] rocksdb::UserComparatorWrapper::Compare
```

Questions?

THANK YOU