STORAGE DEVELOPER CONFERENCE
SDC 22
Fremont, CA
September 12-15, 2022
BY Developers FOR Developers
A SNIA Event

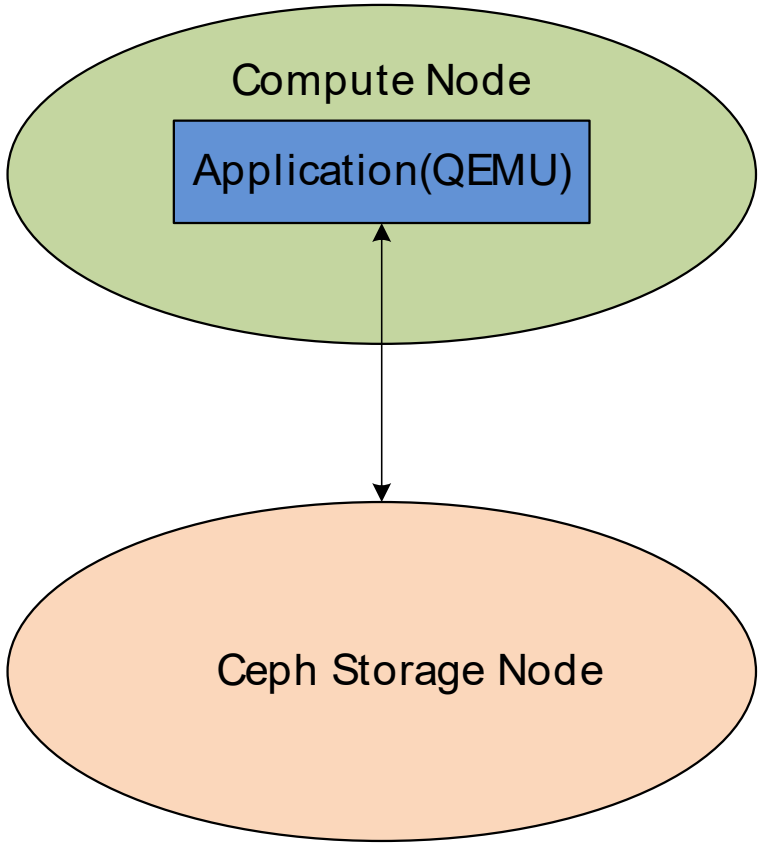
Implementation of Persistent Write Log Cache with Replication in Ceph

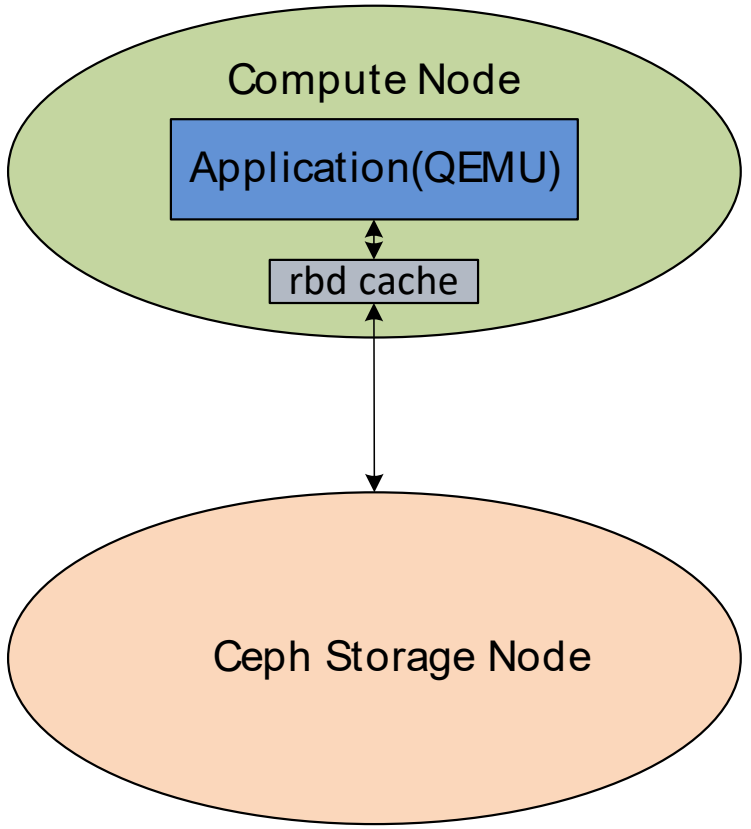
Presented by Feng, Hualong <hualong.feng@intel.com>
Speaker by Liu, Chunmei <chunmei.liu@intel.com>

Agenda

- Overview
- Local Persistent Memory Mode
- Remote Replicated Mode
- Q&A

A workload as example



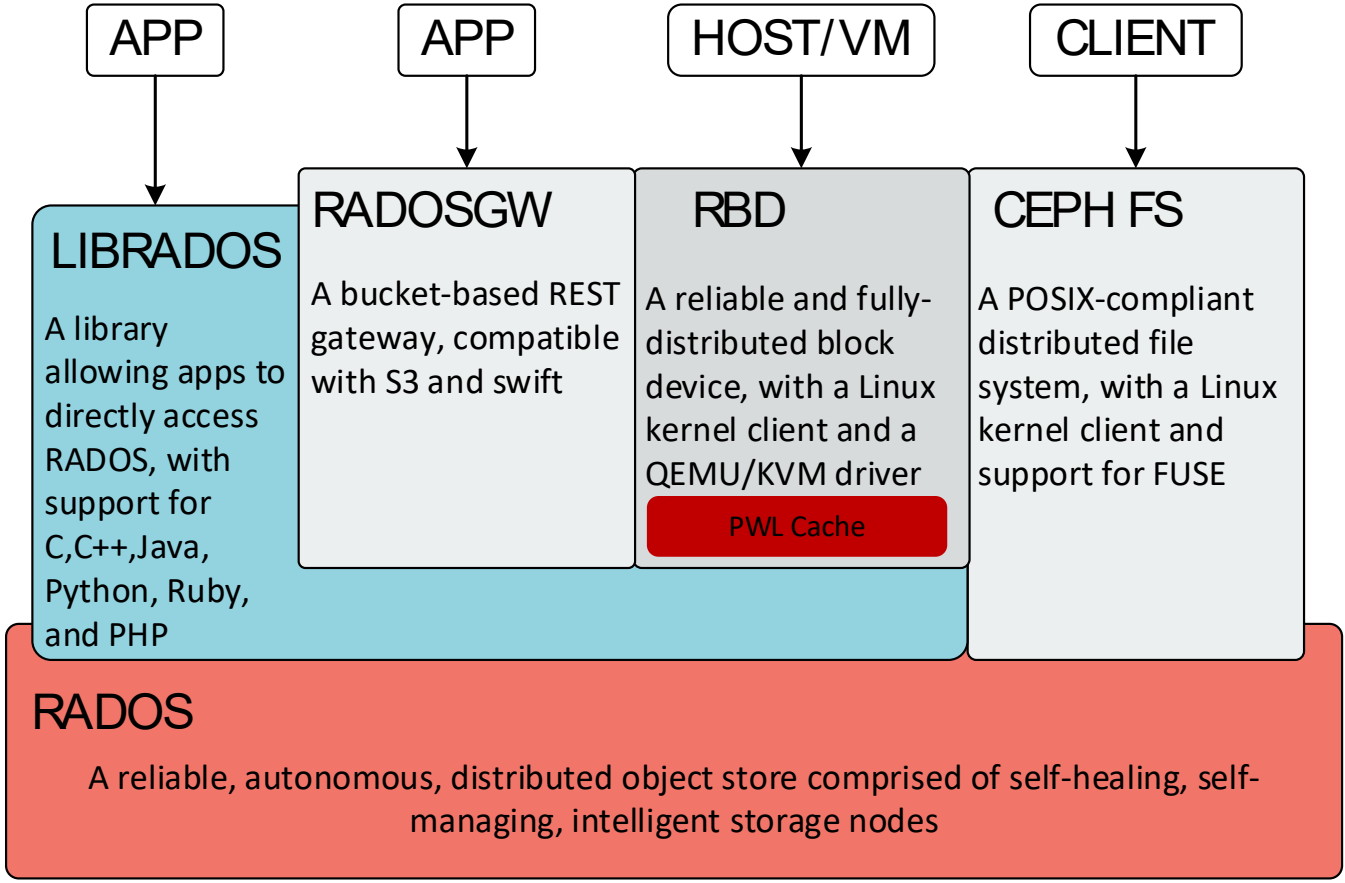


Application write directly to storage node, burst write will increase backend pressure

Using rbd cache(memory cache) maybe loss the data

Persistent Write Log Cache

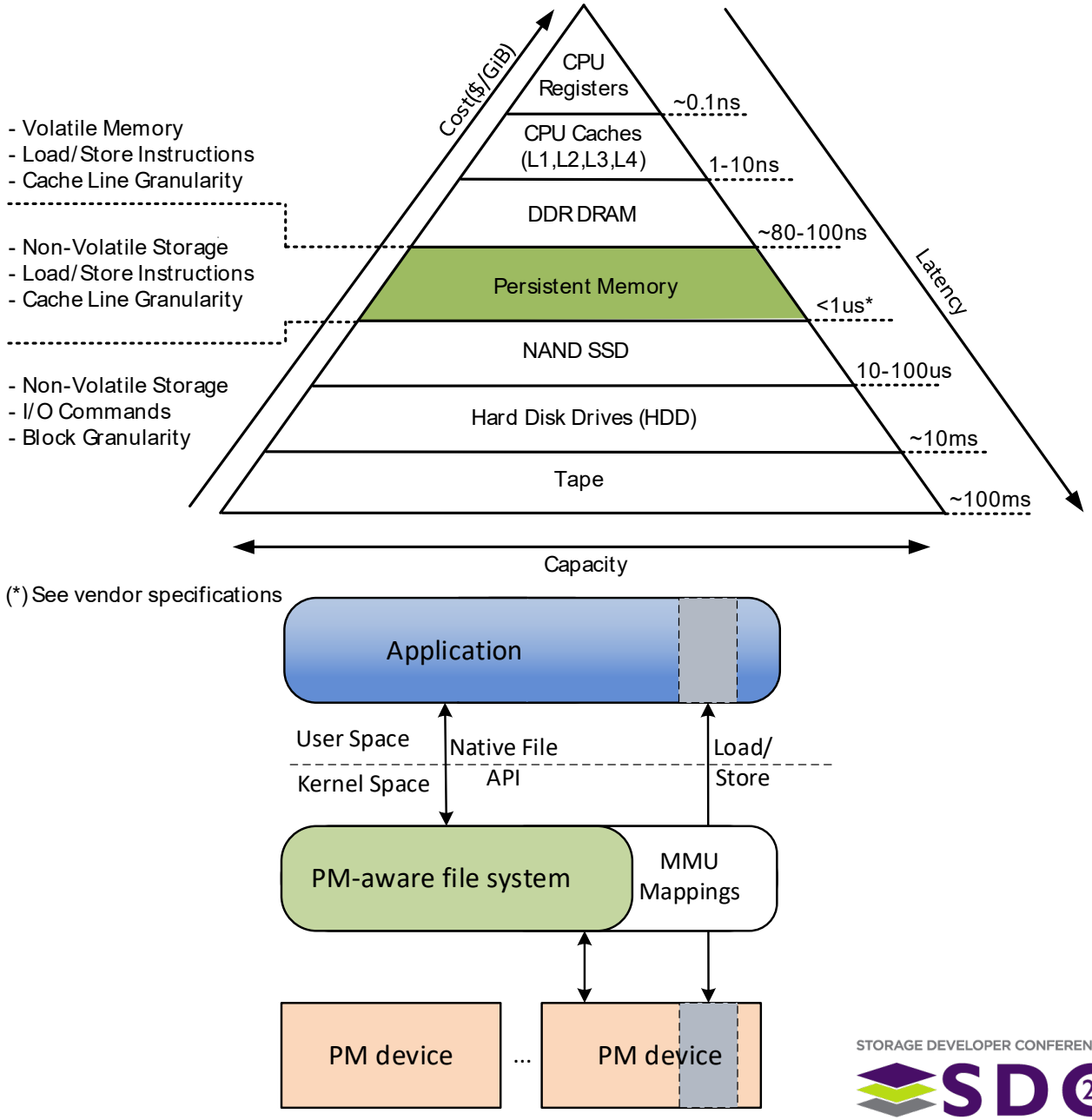
- Goals
  - Improve Ceph RBD write performance
  - Mask RADOS tail latency, keep 99.99% of user request tail latency in 1ms
  - Guarantee write durability through any single failure compared with RBD cache(memory cache)
- Use Case
  - Workloads require for low latency (average and tail latency), like DB, QEMU.
  - Need persistent feature and high performance
- Two Backend
  - **PMEM(local mode & replicated mode)**
  - NVME SSD(only local mode)



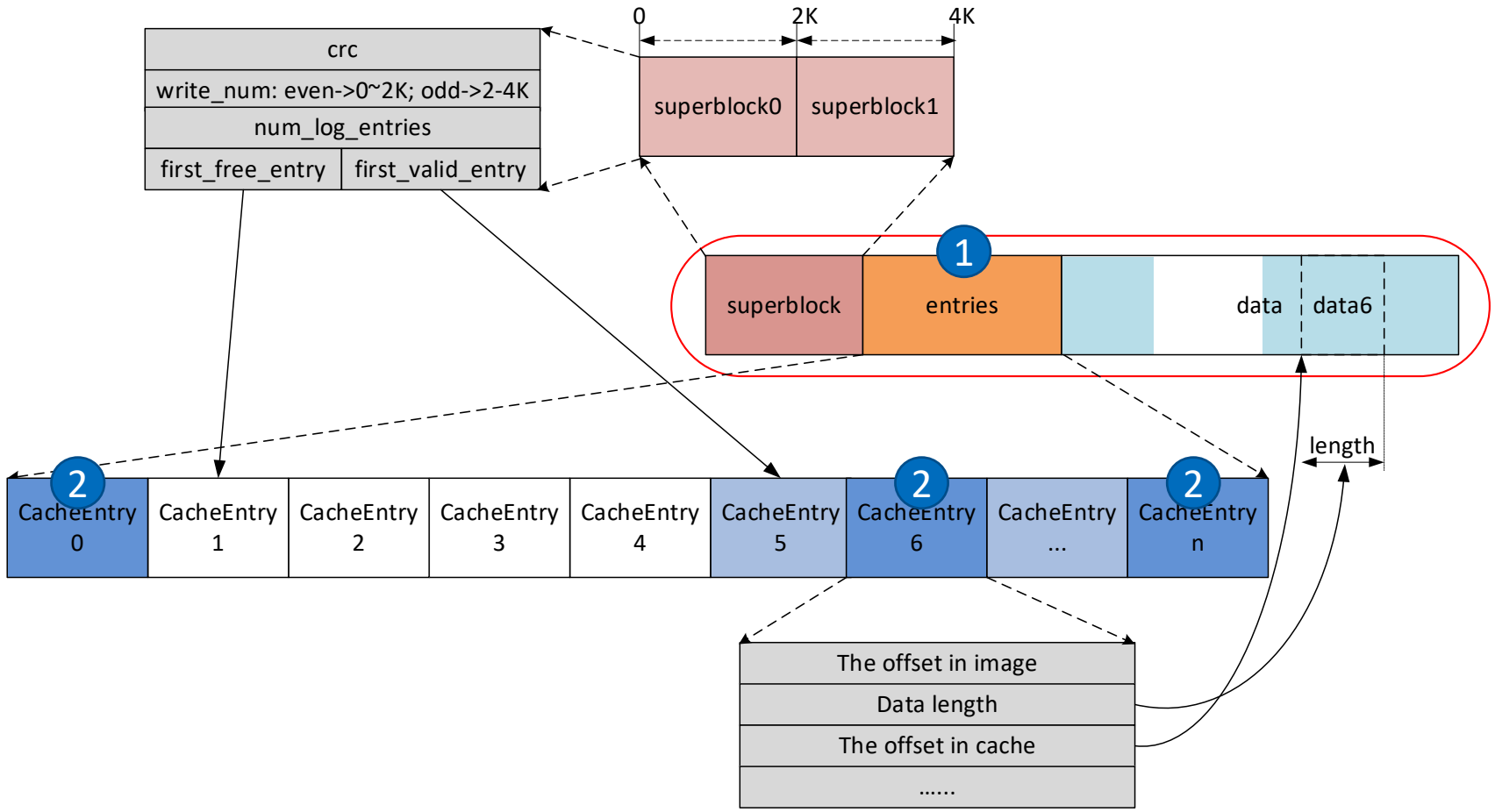
Local Persistent Memory Mode

Persistent memory

- The persistent memory tier offers **greater capacity** than DRAM and significantly **faster performance** than storage
- Applications can access persistent memory resident data structures in-place, **like** they do with **traditional memory**, eliminating the need to page caches of data back and forth between memory and storage.
- The persistent memory can **be accessed** directly **by RDMA**
- PMDK/**libpmem** library simplify persistent memory programming.



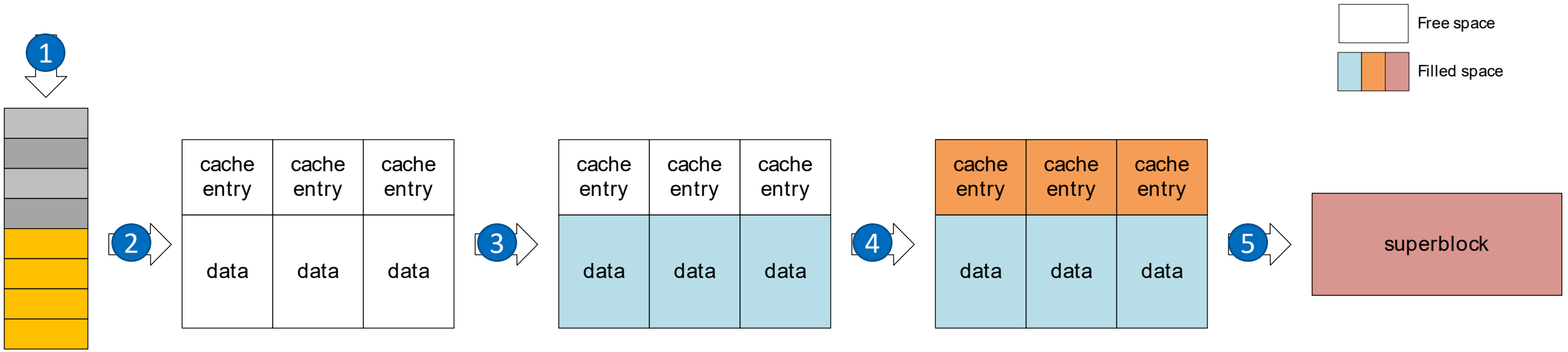
PWL Local PMEM Layout



1 A Ring composed of log entries, with pointers to allocated data buffers

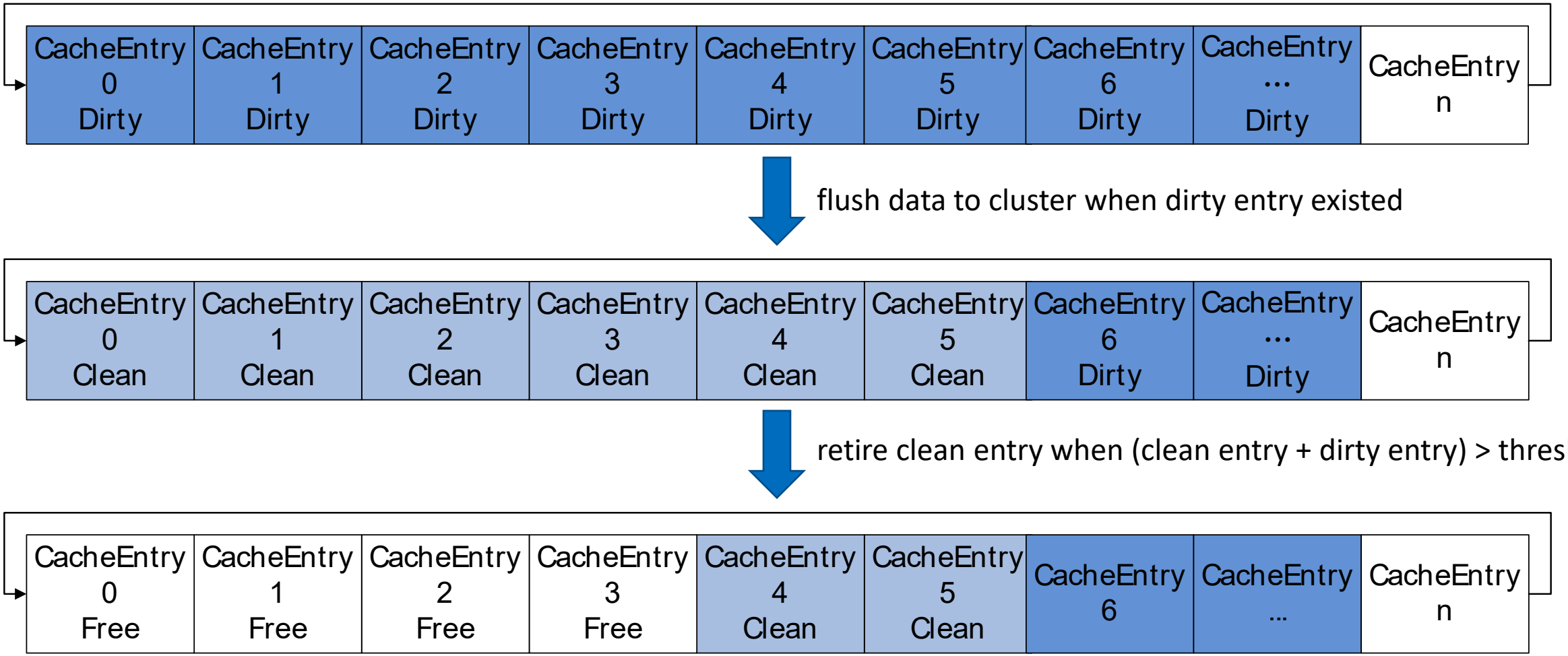
2 User flush request will insert a sync point entry to guarantee order partially

PWL write operation flow



1. I/O request is dispatched to PWL layer
2. Check/reserve free resource(cache entry and data space), if no free resource, put it in defer queue, if yes, allocate space
3. PWL write and flush data to persistent memory
4. PWL write and flush entry to persistent memory
5. Update superblock, persist superblock

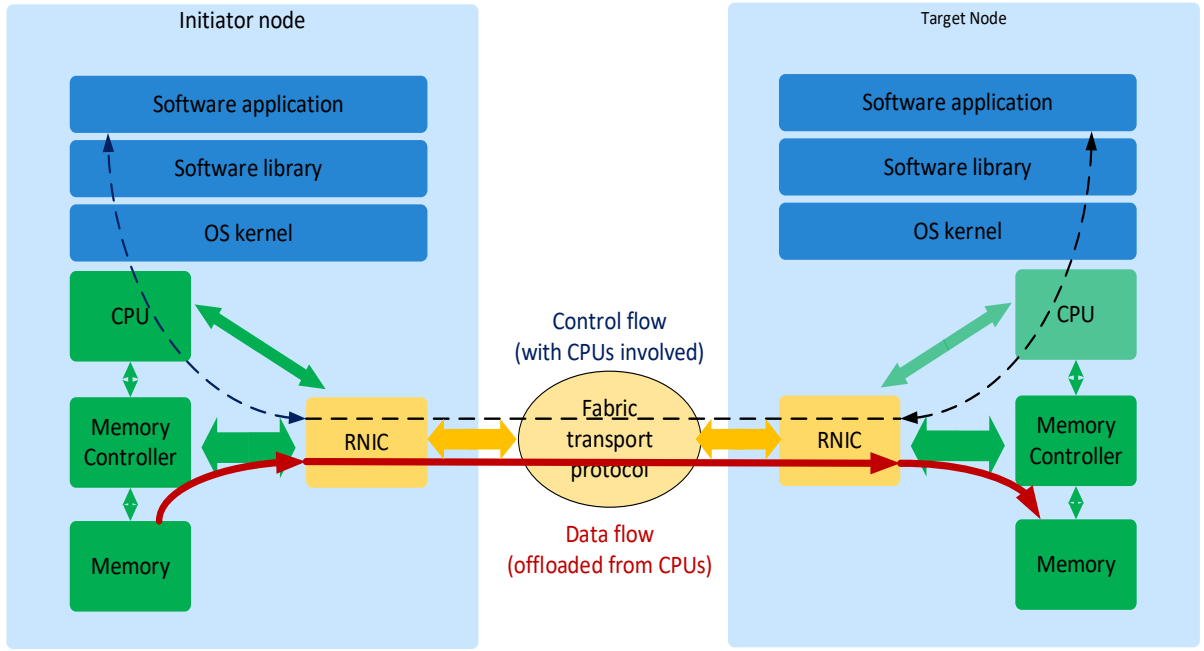
PWL Local retire flow



Remote Replicated Mode

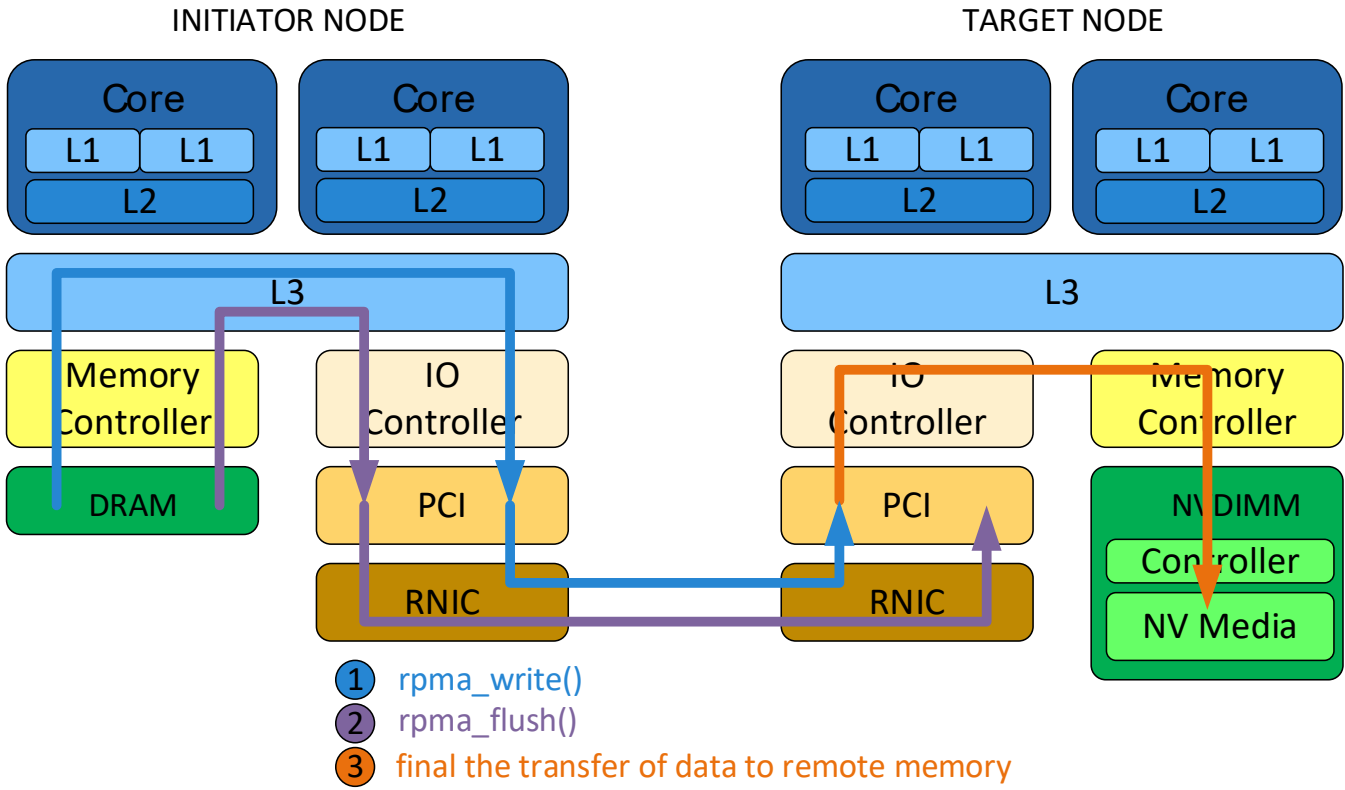
RDMA Introduction

- enabling a **user-level program** to directly access the memory of another computer, without interrupting the processors.
- **one-sided primitives(READ/WRITE)**, RDMA-capable NIC(RNIC) can directly read/write the server memory bypassing the server CPU.
- **two-sided primitives(SEND/RECV)** are similar to message passing.
- RDMA writes provide the **lowest latency** and **highest throughput**.

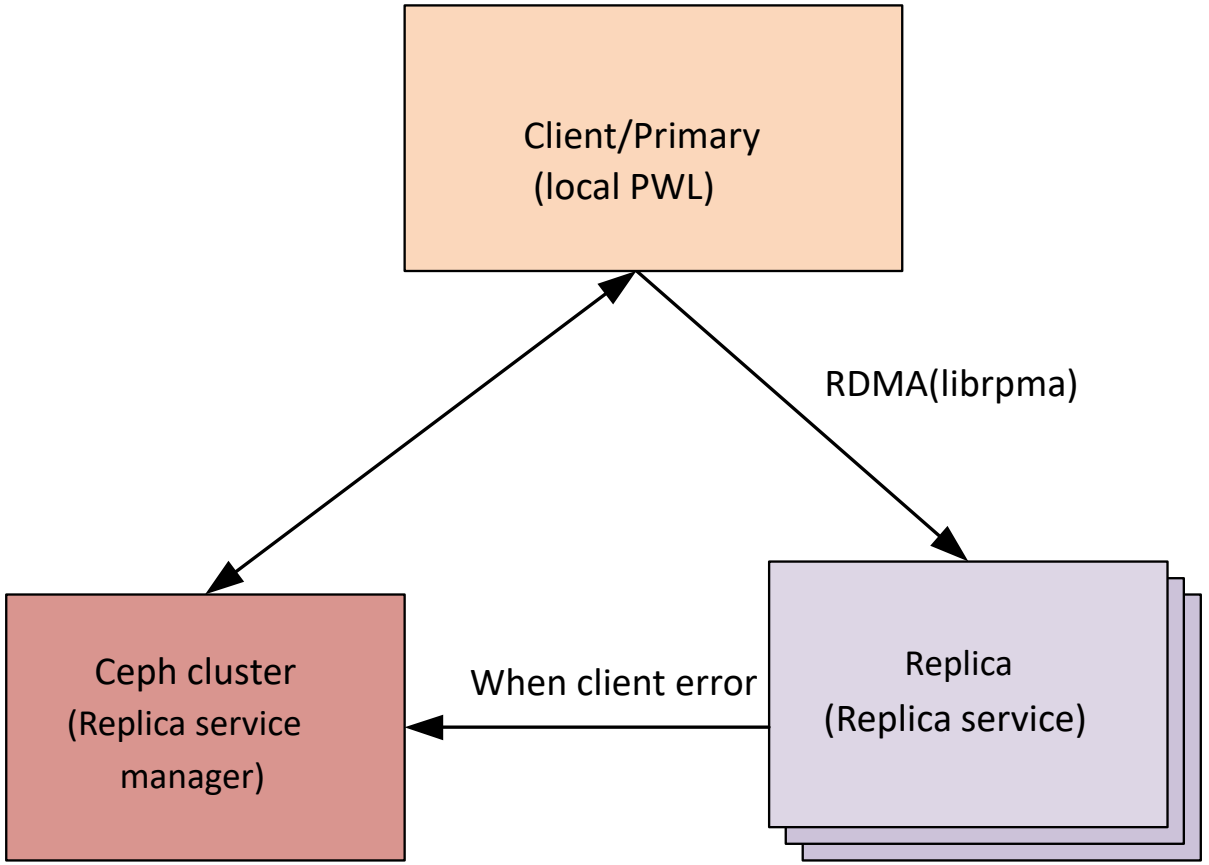


Direct Write to Persistent Memory

- In order to enable Direct Write to Persistent Memory is turning off Intel Direct Data I/O (DDIO) on target node.
- Write to persistent memory is a feature of a platform and its configuration which allows an RDMA-capable network interface to write data to platform's persistent memory in a persistent way.
- PWL Remote replicate includes two steps:
  - a sequence of RDMA Write operations
  - then followed by one RDMA Flush (Read) operation.
- **librpma** library simplify accessing persistent memory on remote hosts over Remote Direct Memory Access (RDMA).

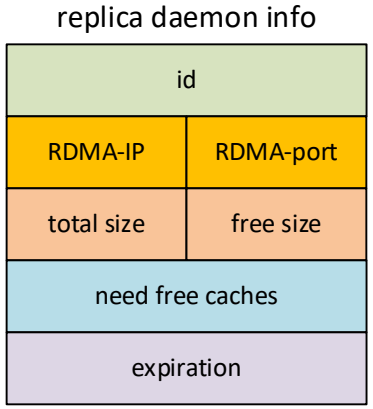


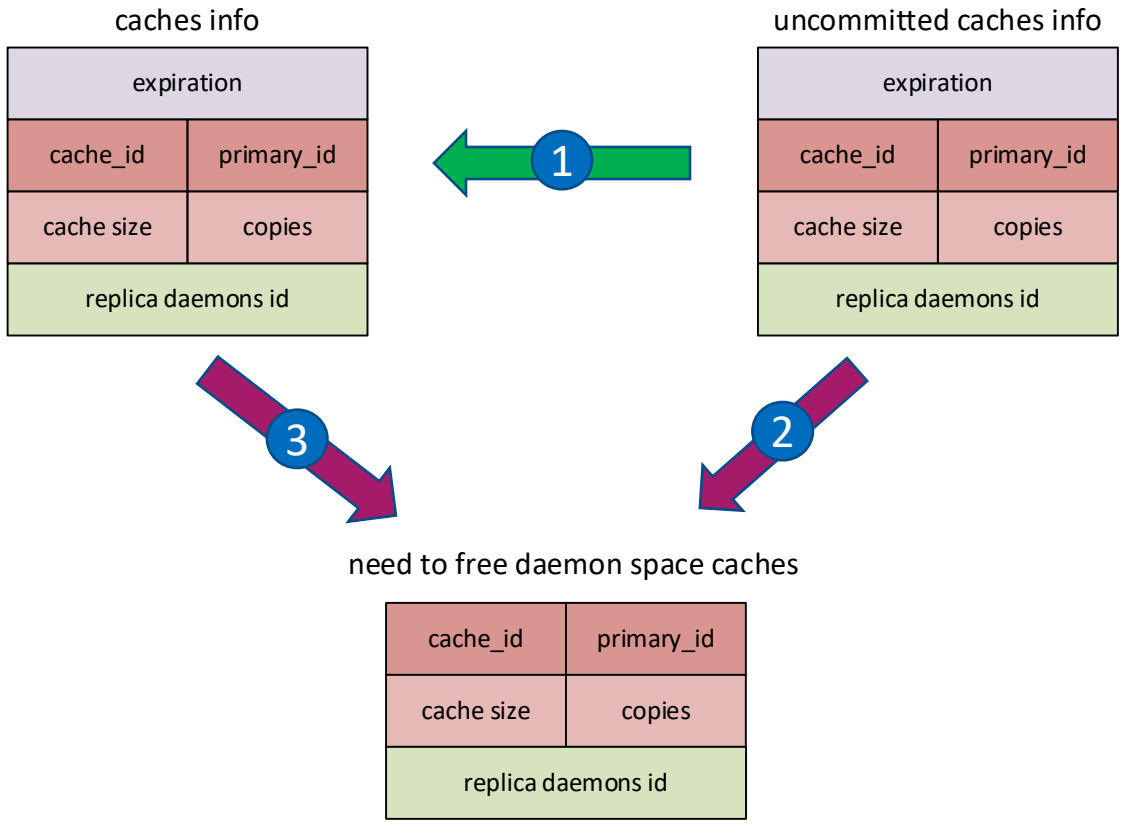
Replica overview



1. Replica service is a daemon process
2. Replica Service Manager manage replica service in Ceph cluster
3. Using RDMA to copy data to replica
4. Replica write data to Ceph cluster only on client error

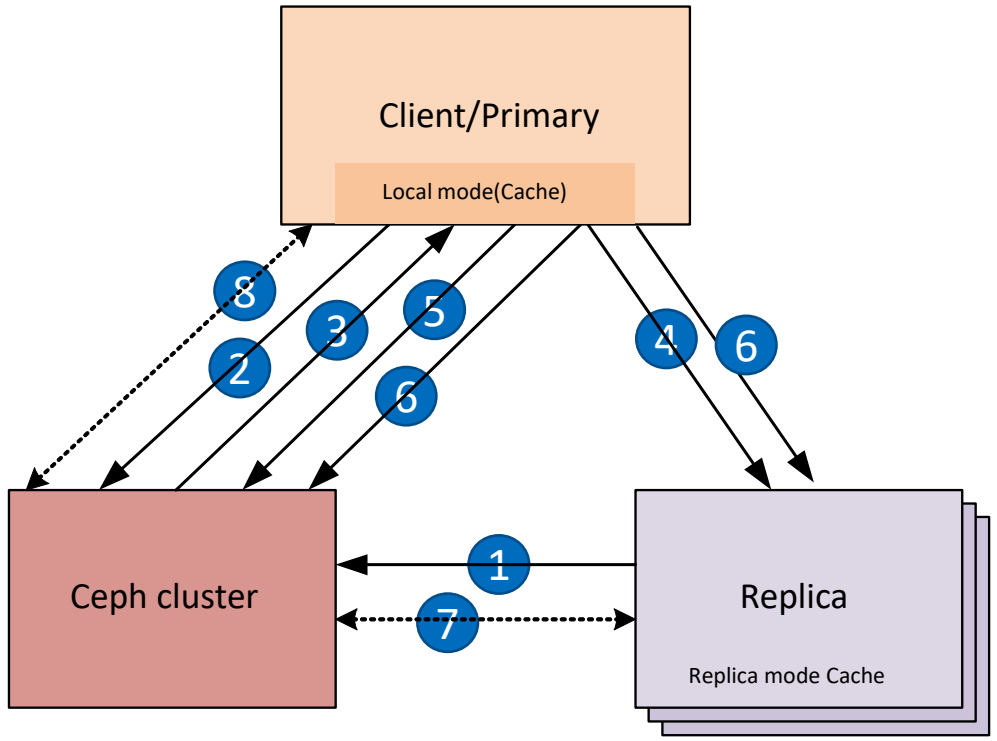
Data Structure about Replicated Implementation





1. Client succeed to connect replicated daemon
2. Client failed to connect replicated daemon
3. Client occurs error which lead to expired or RDMA disconnected

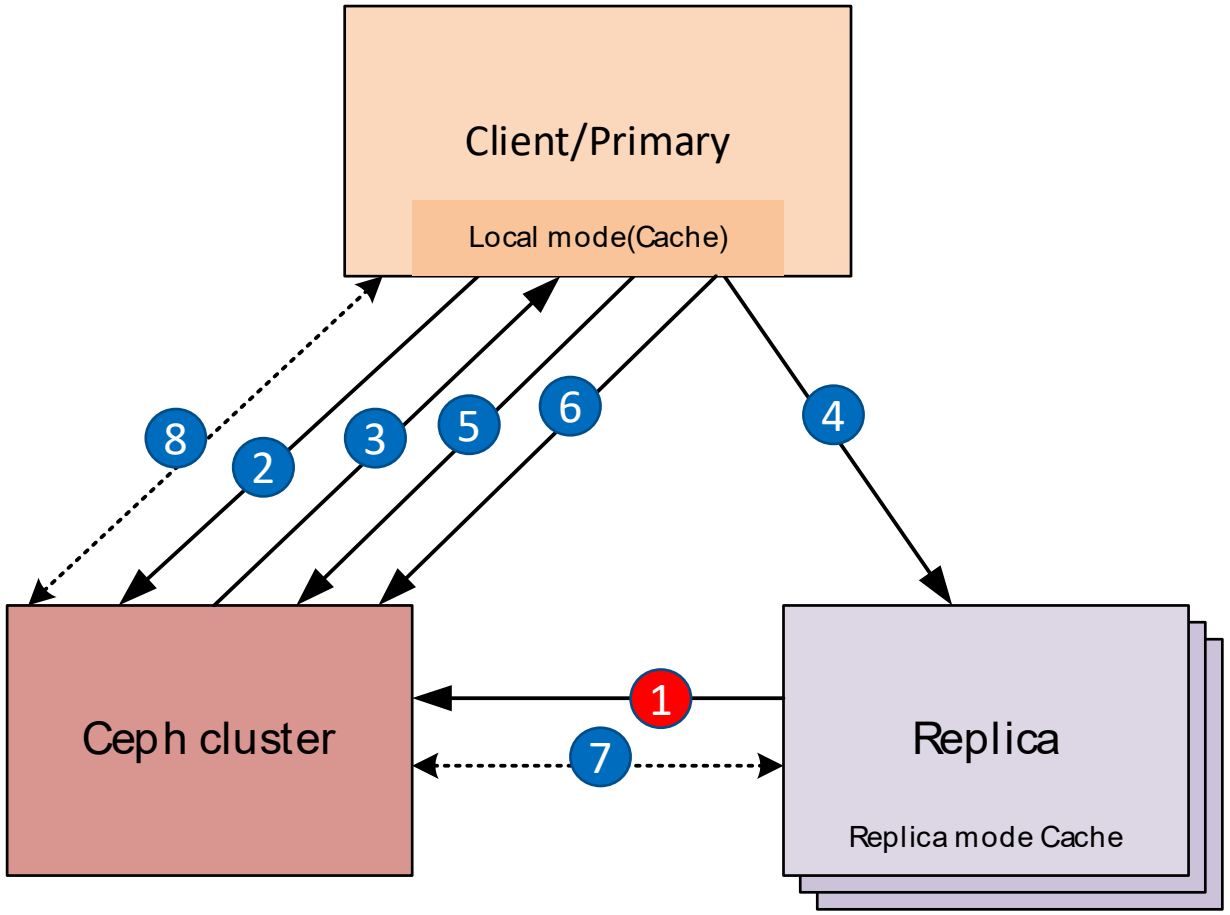
Replica Control Process

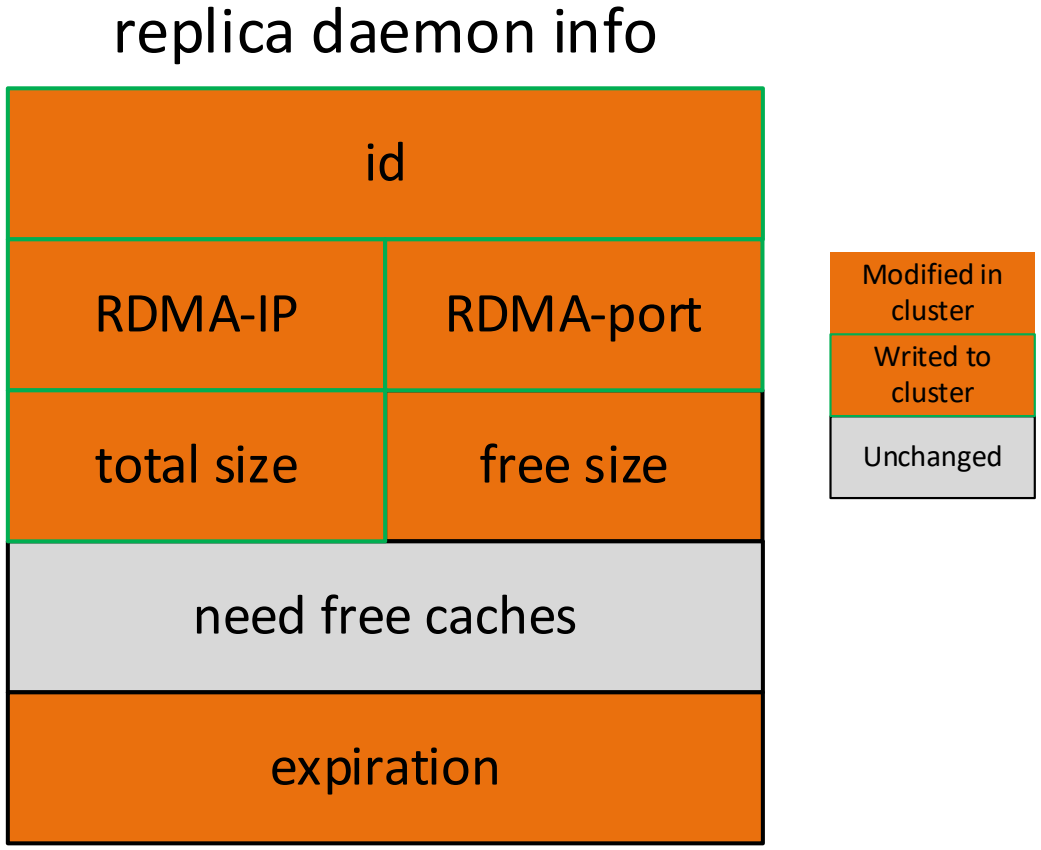


1. When replica daemon started up, it reported to Ceph cluster
2. Client requests replicated daemons information to replica
3. Ceph cluster replies replicated daemons information to client or no space
4. Client connects replicated daemons
5. Client acknowledges this connection success or failure
6. Free space from Client
7. Replica sets up heartbeats to make Ceph cluster know the replicated daemon status and free unused cache
8. Client sets up heartbeats to make Ceph cluster know its status

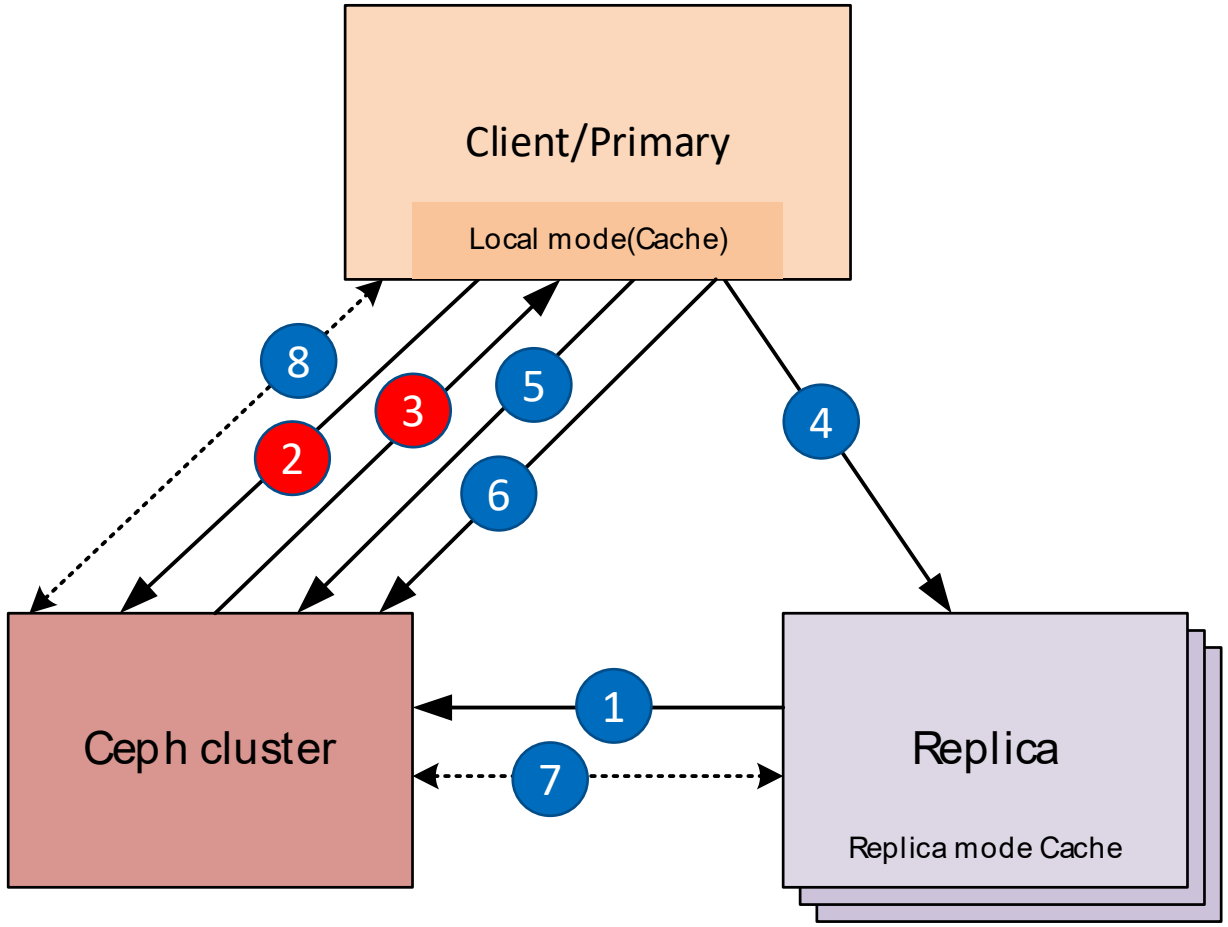
Replica Control Process

1 When replica daemon started up, it reported to Ceph cluster





Replica Control Process

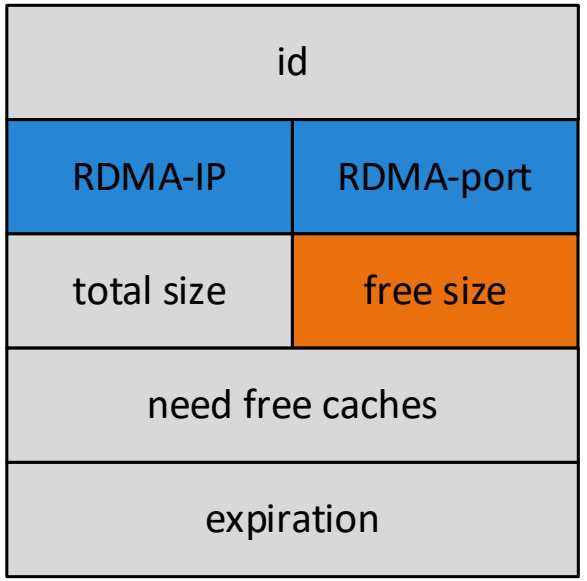


2 Client requests replicated daemons information to replica

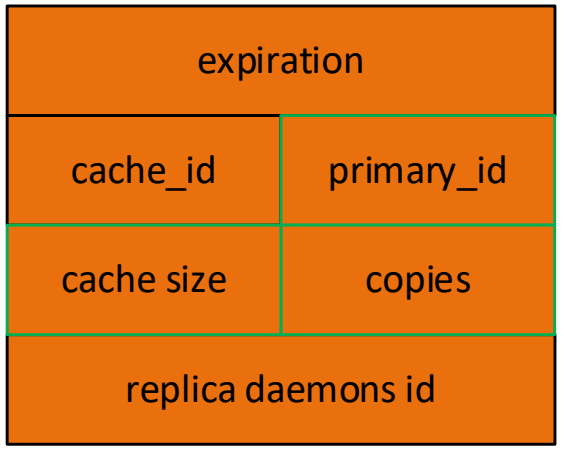
3 Ceph cluster replies replicated daemons information to client or no space

Return to client

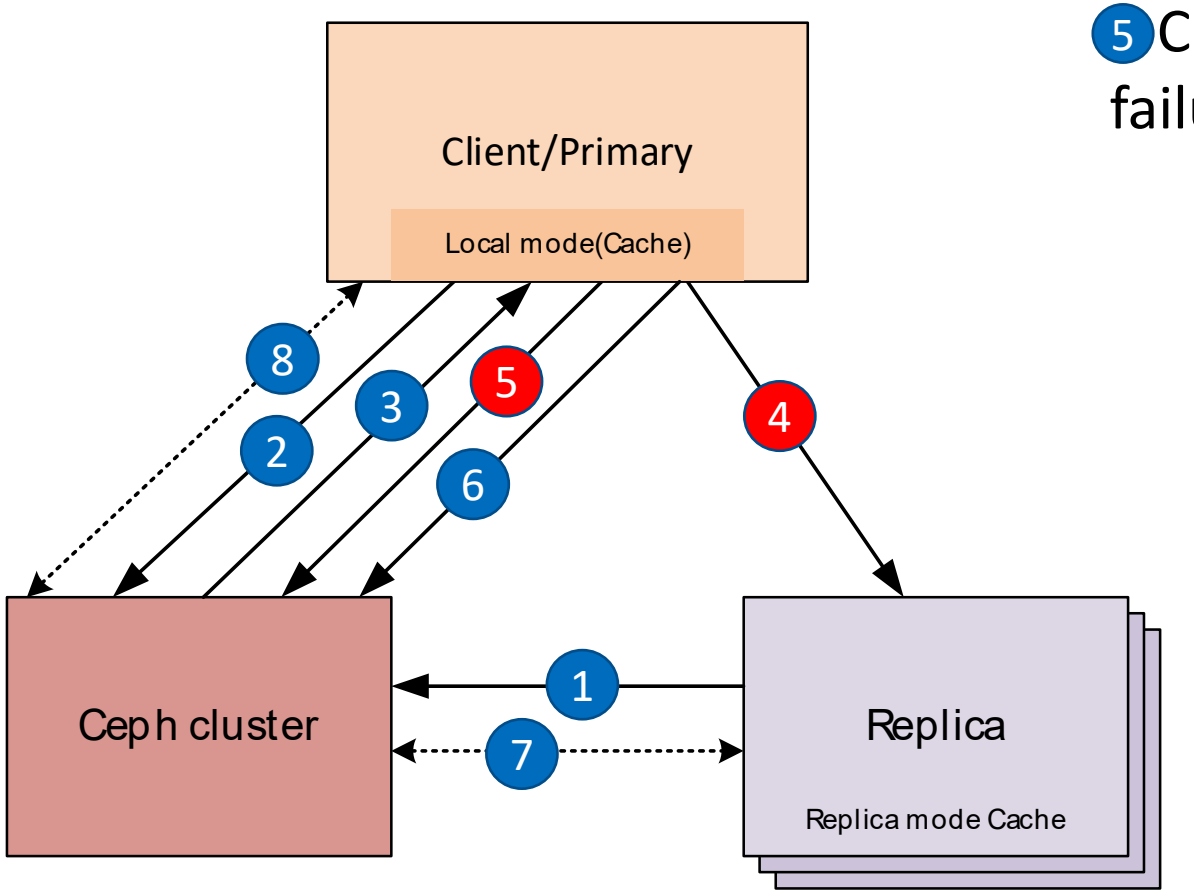
replica daemon info



uncommitted caches info

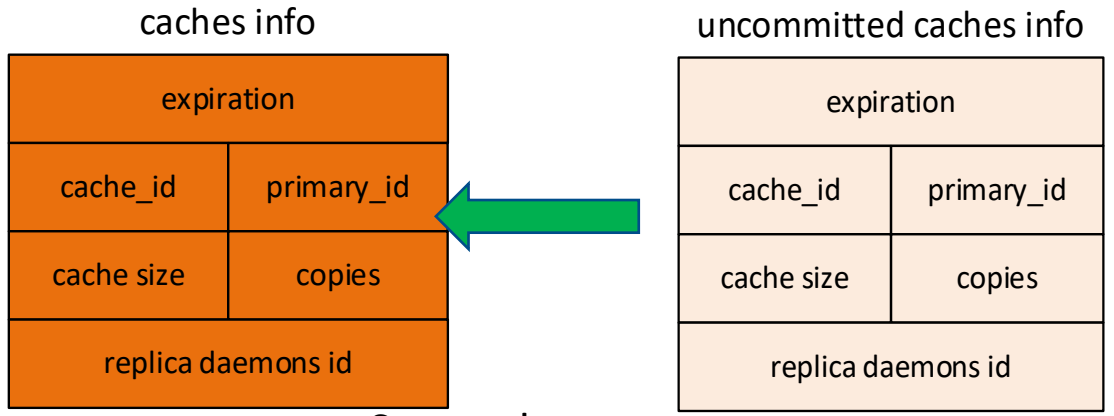


Replica Control Process

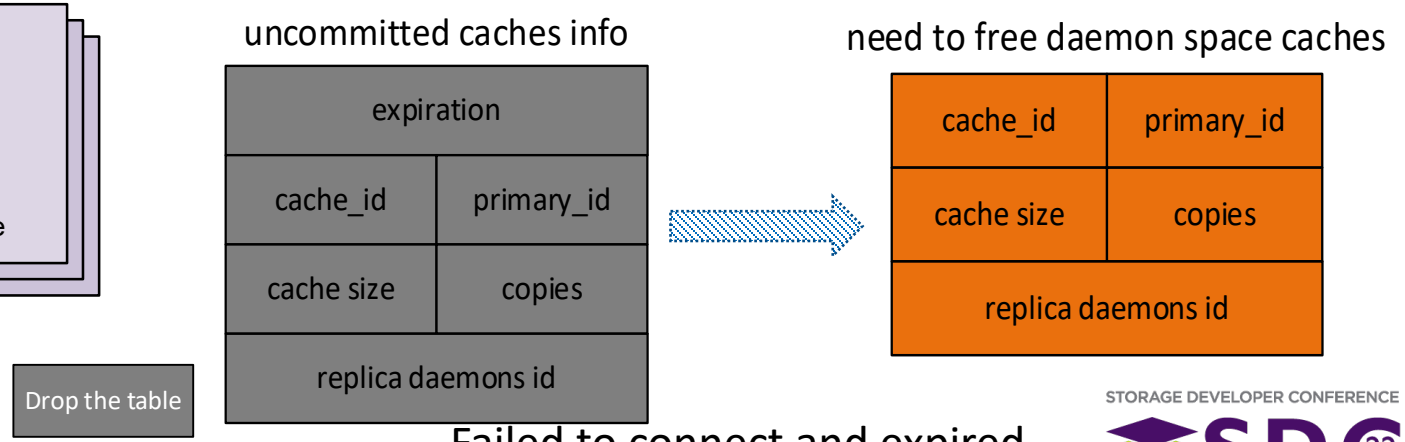


4 Client connects replicated daemons

5 Client acknowledges this connection success or failure

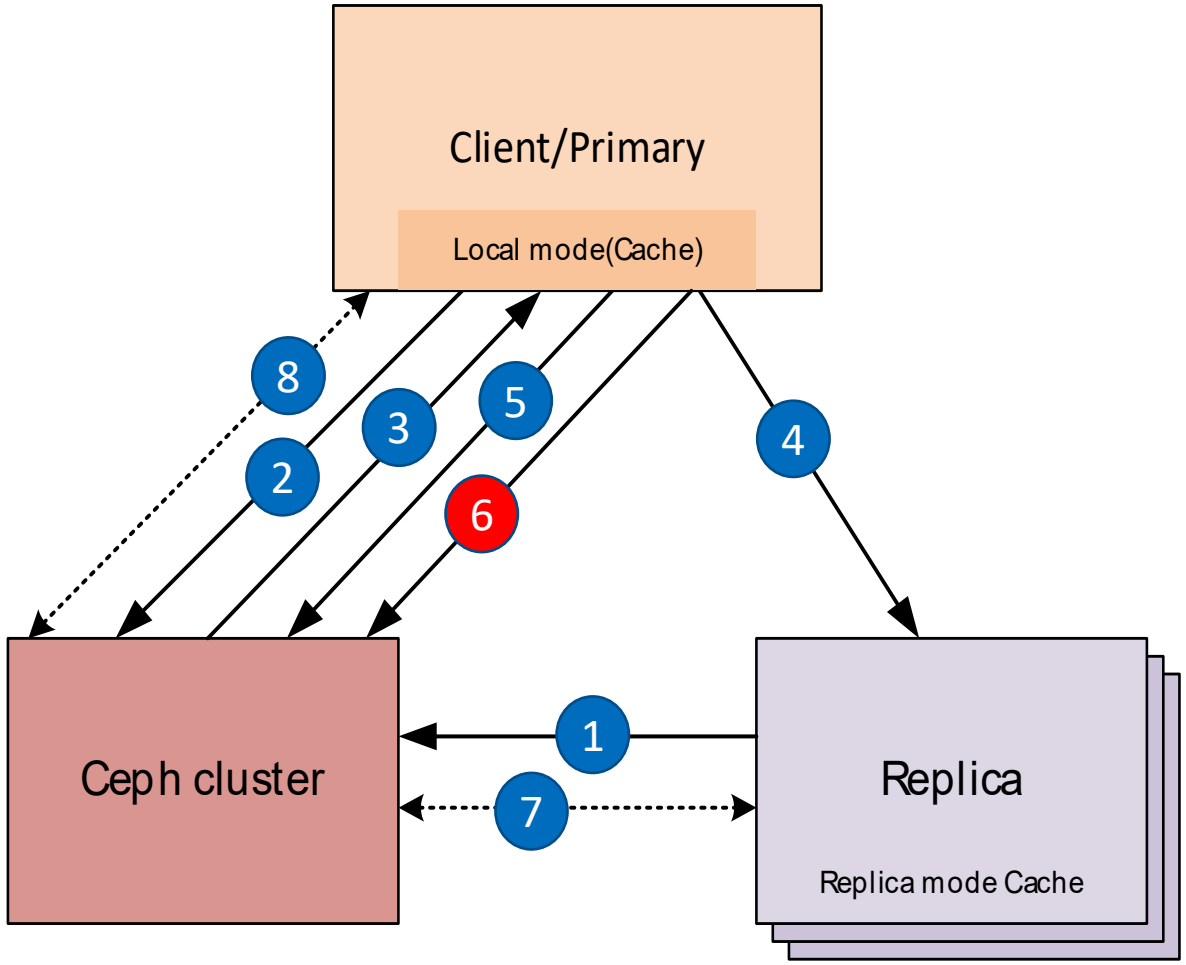


Succeed to connect



Failed to connect and expired

Replica Control Process



6. Free space from Client

caches info

expiration	
cache_id	primary_id
cache size	copies
replica daemons id	

Drop the table

Client succeed to free daemon space

caches info

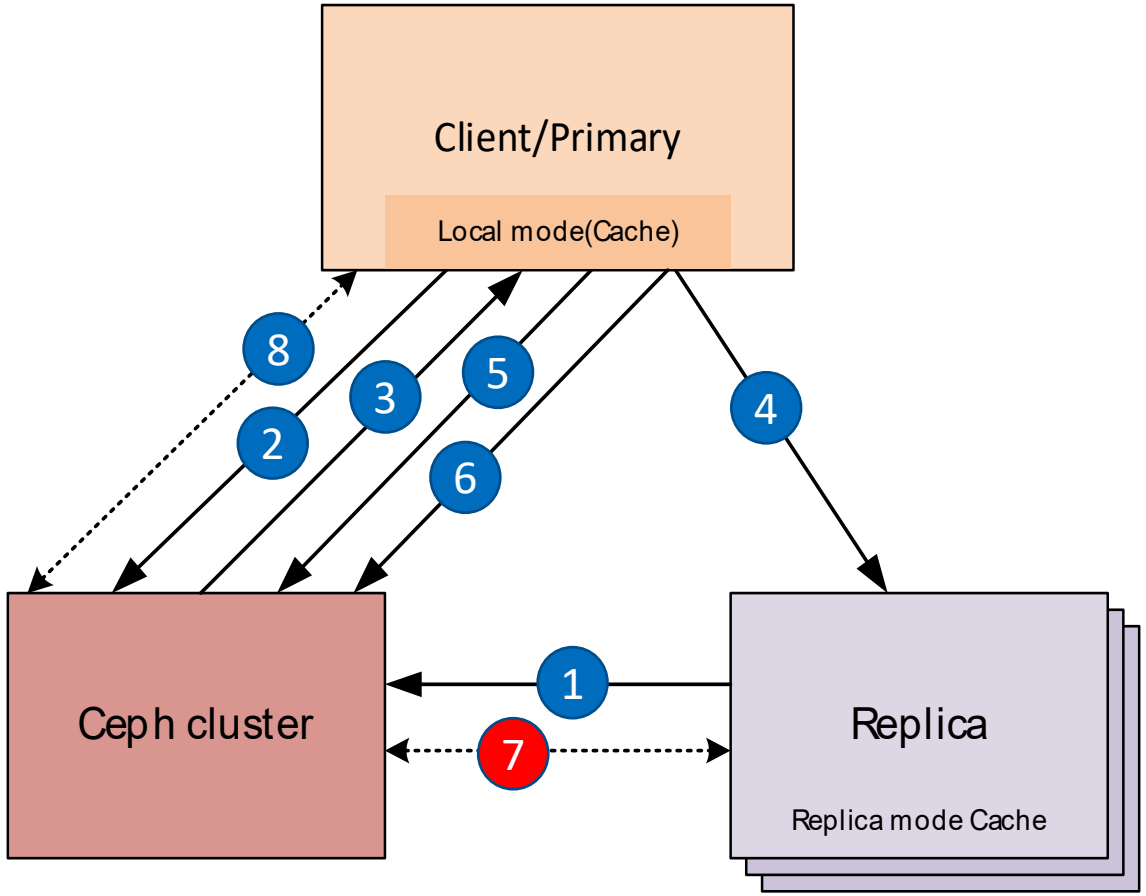
expiration	
cache_id	primary_id
cache size	copies
replica daemons id	

need to free daemon space caches

cache_id	primary_id
cache size	copies
replica daemons id	

Client failed to free daemon space

Replica Control Process



7 Replica sets up heartbeats to make Ceph cluster know the replicated daemon status and free unused cache

replica daemon info

id	
RDMA-IP	RDMA-port
total size	free size
need free caches Cache id	
expiration	

need to free daemon space caches

cache_id	primary_id
cache size	copies
replica daemons id	

Ping succeed

Find replica id tell it to free space

Deleted Replica telled
Drop the table on daemons empty
Drop the table

replica daemon info

id	
RDMA-IP	RDMA-port
total size	free size
need free caches	
expiration	

caches info

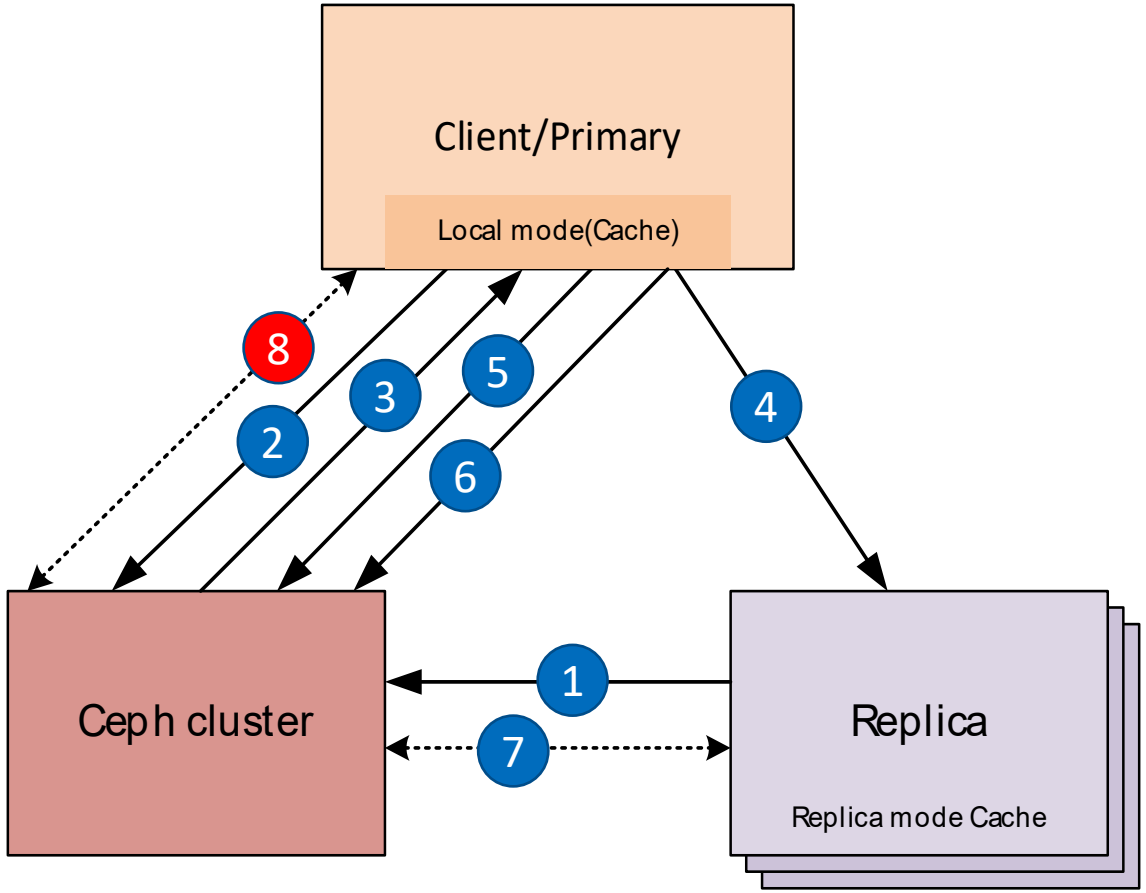
expiration	
cache_id	primary_id
cache size	copies
replica daemons id	

Ping expired

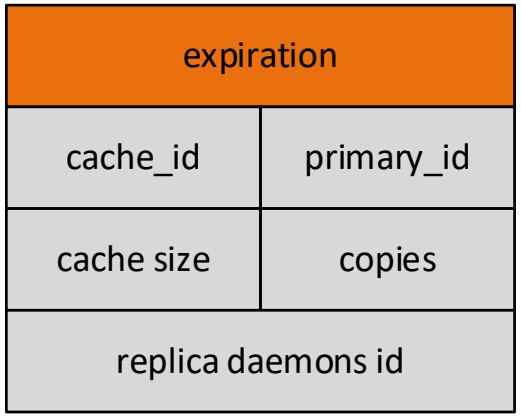
Delete dead replica id

Replica Control Process

8 Client sets up heartbeats to make Ceph cluster know its status

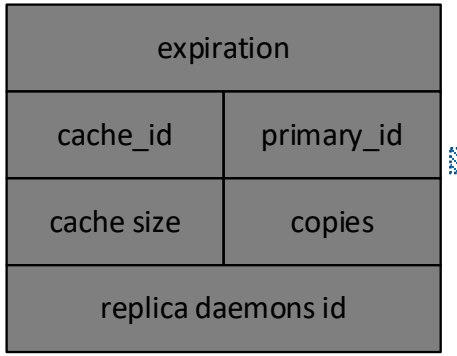


caches info

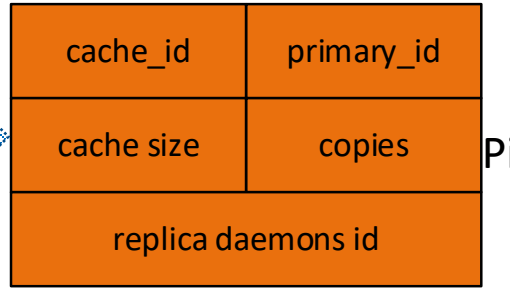


Ping succeed

caches info

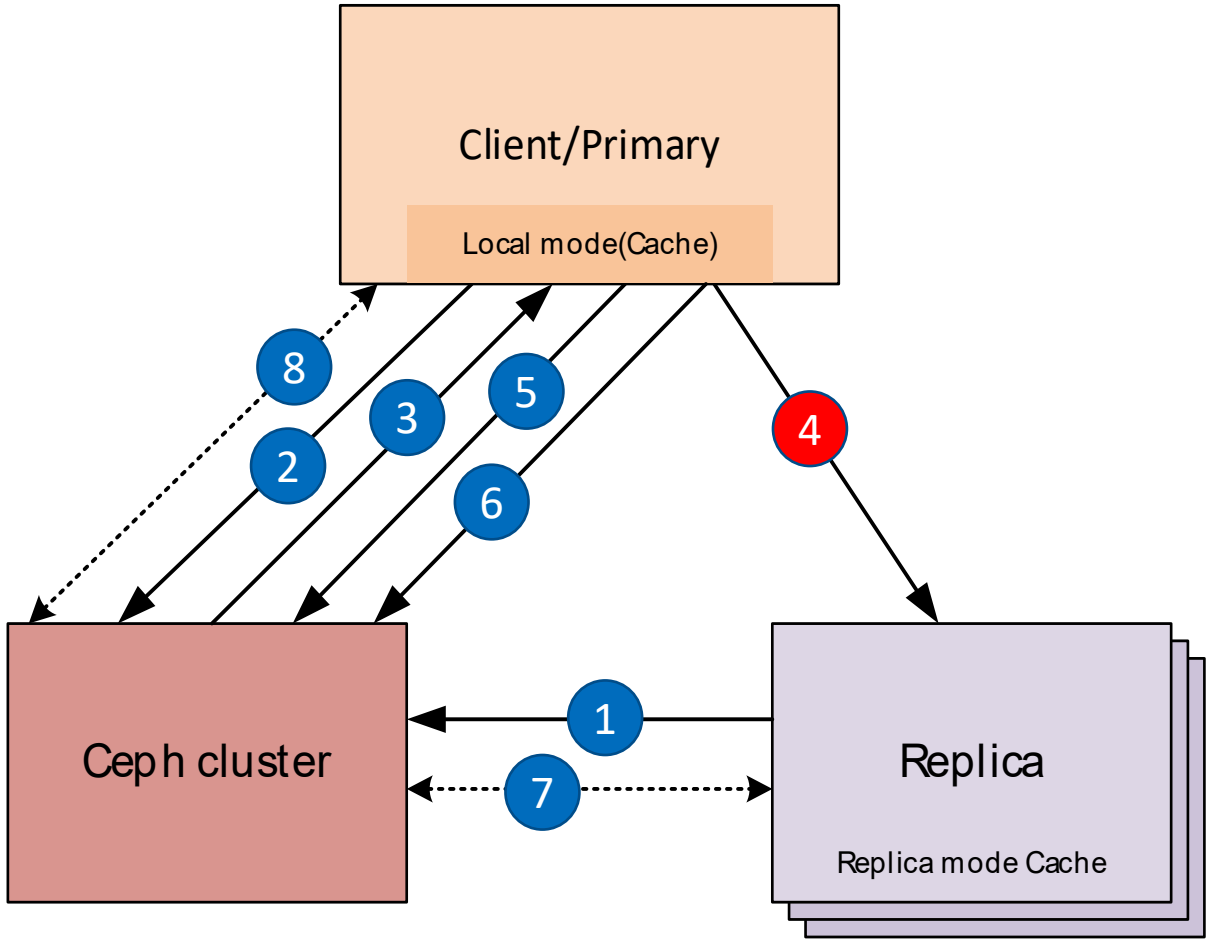


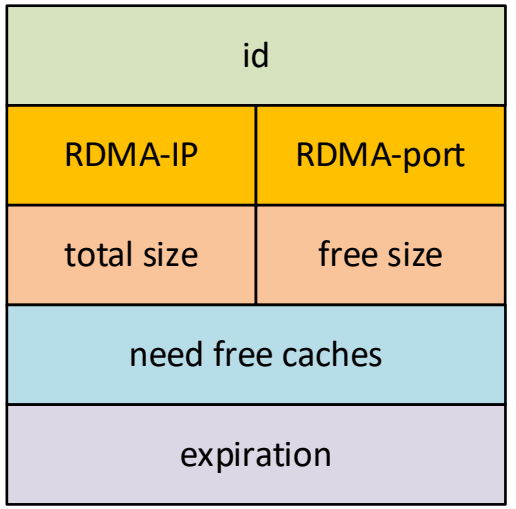
need to free daemon space caches

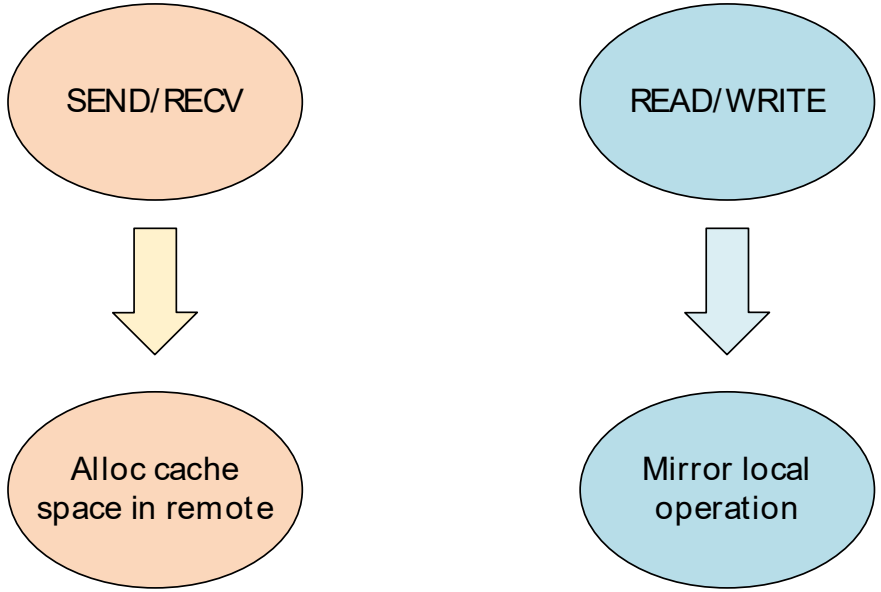


Ping expired

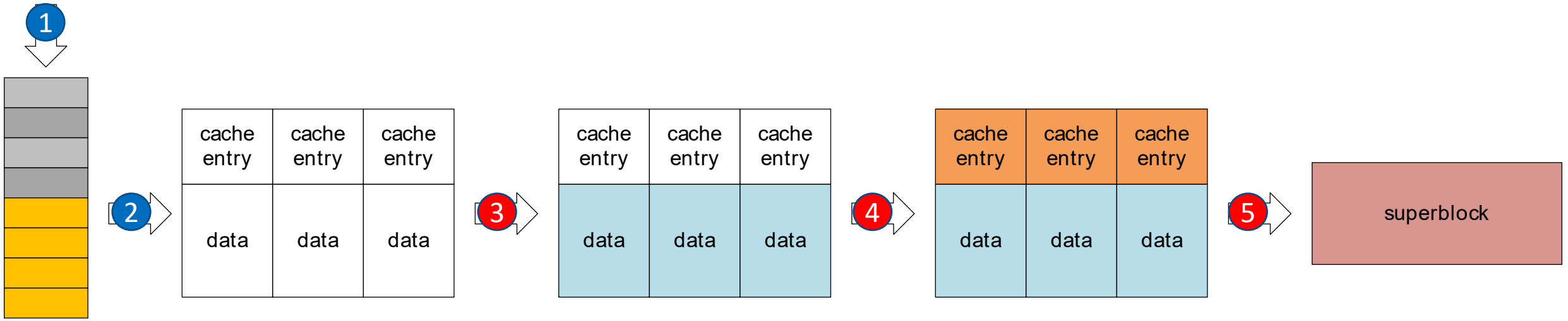
RDMA Usage in Remote Replicated Mode





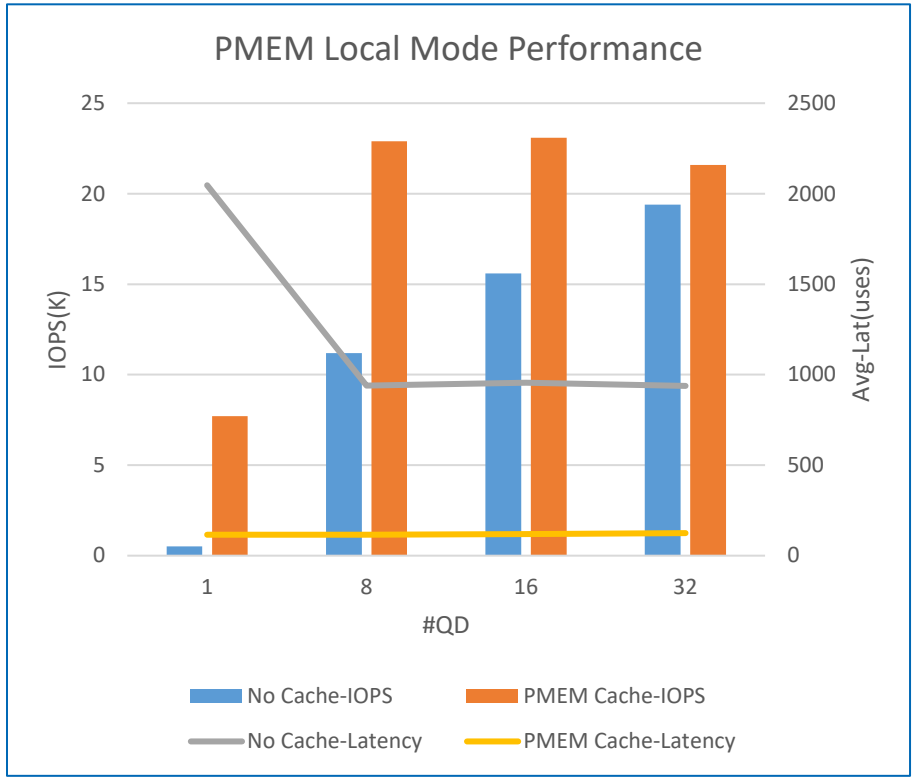


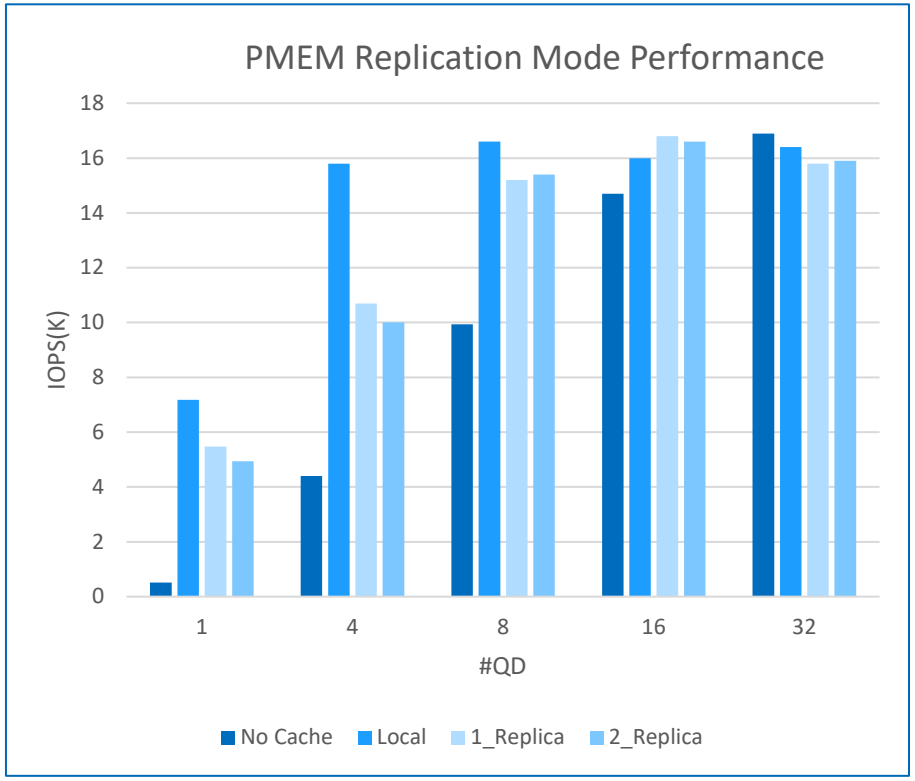
PWL Replicated Mirror IO process



1. I/O request is dispatched to PWL defer queue
2. Check free resource(cache entry and data space), if no, stay in defer queue, if yes, allocate space
3. PWL write and flush data to local persistent memory (and to remote persistent memory)
4. PWL write and flush entry to local persistent memory (and to remote persistent memory)
5. Update and persist superblock on local persistent memory (and to remote persistent memory)

Ceph Client Write-log Cache – Performance (cache is full state)





Notices and Disclaimers

- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. **No computer system can be absolutely secure.**
- Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. For more complete information about performance and benchmark results, visit http://www.intel.com/benchmarks .
- Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/benchmarks .
- Intel® Advanced Vector Extensions (Intel® AVX)* provides higher throughput to certain processor operations. Due to varying processor power characteristics, utilizing AVX instructions may cause a) some parts to operate at less than the rated frequency and b) some parts with Intel® Turbo Boost Technology 2.0 to not achieve any or maximum turbo frequencies. Performance varies depending on hardware, software, and system configuration and you can learn more at http://www.intel.com/go/turbo.
- Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.
- Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.
- The benchmark results may need to be revised as additional testing is conducted. The results depend on the specific platform configurations and workloads utilized in the testing, and may not be applicable to any particular user's components, computer system or workloads. The results are not necessarily representative of other benchmarks and other benchmark results may show greater or lesser impact from mitigations.
- Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.
-

That’s all.

Your feedback is important to us.

Q&A

Backup

Intel Data Direct I/O Technology(Intel DDIO)

- allowing RNIC to read and write directly to the CPU cache
- reducing the overhead of DMA controller invalidations.
- helping to deliver increased bandwidth, lower latency, and reduced power consumption.