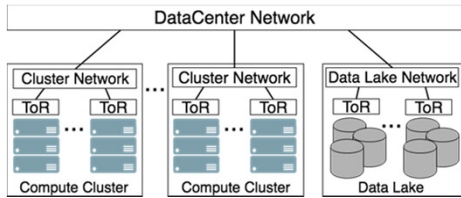


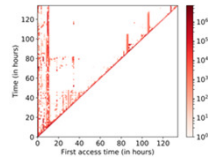
Problem

- In a multi-tenant data center, network may become a bottleneck.
 - Network over-subscription
 - Shared Data lake

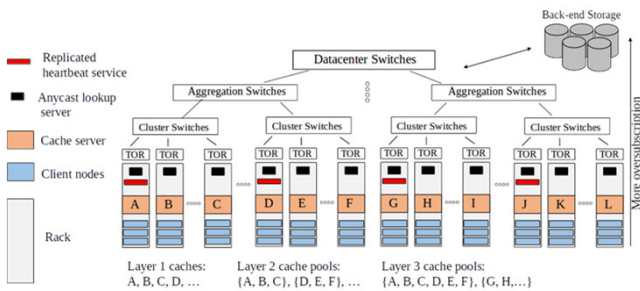


Solution

- Observation:
 - High Data Reuse
 - Uneven Data Popularity
- Caching
- Multi-layered caching: data on access side of the network bottleneck
 - Read cache
 - Data analytics
 - Write cache
 - Pipelined batch jobs

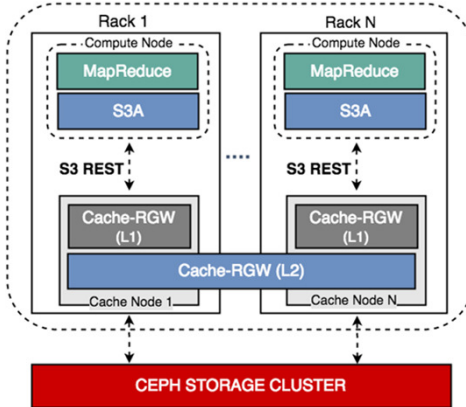


Our Architecture



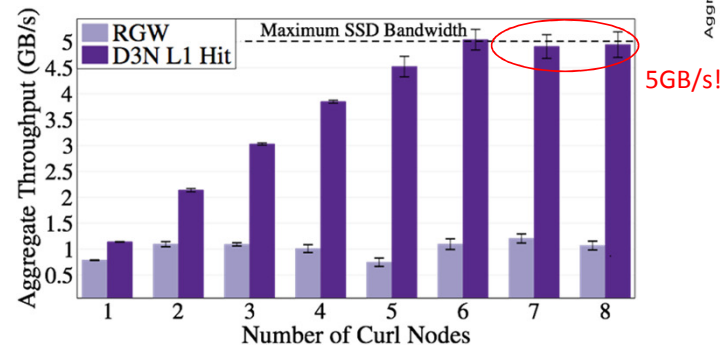
Implementation

- Two level caching mechanism implemented by modifying the original **CEPH Rados Gateway**.
- L1-Cache and L2-cache are **logically separated**, they physically share the same physical cache infrastructure.
- "D3N" caches read/write traffics.
- uses LRU replacement.



Evaluation

L1 Cache Hit Performance



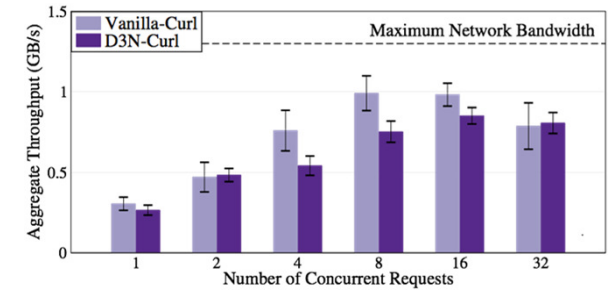
Project Webpage:

<http://info.massopencloud.org/blog/bigdata-research-at-moc>

Github Repo for Cache-RGW Code: <https://github.com/maniabd/engage1>

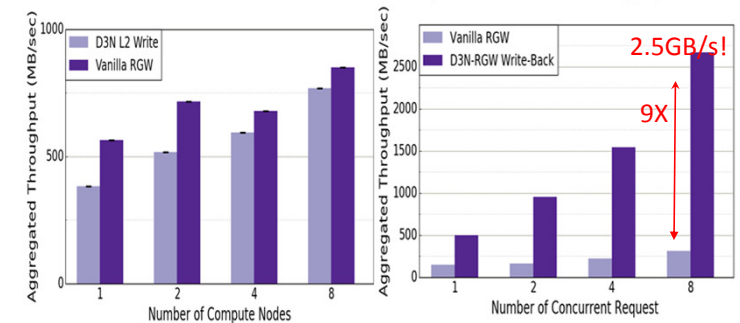
Evaluation

D3N adds low overhead (at read misses)



D3N impose overheads a small

D3N adds low overhead (for write-through)



Write-through policy imposes small overhead

D3N adds low overhead (for write-through)

