



# Quality of Service in a flash-based OpenStack Environment

Salil Gokhale

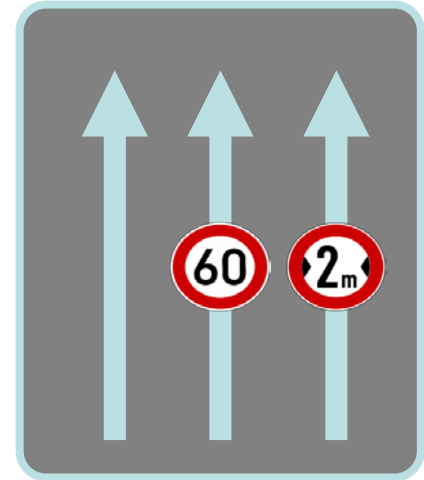
Sr. Product Manager, HP Storage

# Why Quality of Service?

- **Performance provisioning at a granular level.**  
Controls on transaction (IOPS) and throughput (GB per second) performance, first set on a LUN basis and at the single VM level in the future.
- **'Noisy neighbors' disrupt cloud and enterprise storage delivery.**  
When an application or VM spikes up its performance utilization and leaves less controller caching and flash resources available for other applications, customers feel the adverse affects of 'noisy neighbors.'  
This limitation is a major reason why it is so difficult for cloud storage providers to create consistent multi-tenant environments using traditional storage systems which lack Storage QoS.

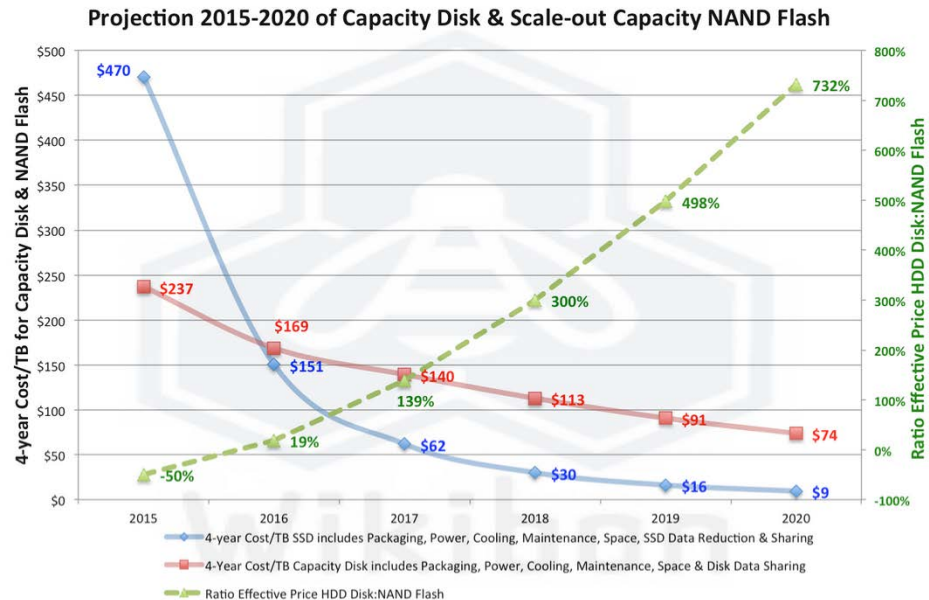
# How can OpenStack QoS with Flash help you?

- Differentiation of workloads on the same Storage array.
- For OpenStack, QoS is part of the Standard interface provided by the Cinder Block Driver
- Take advantage of Data Compaction Technologies available on All Flash Arrays (AFA).
- Satisfy wide range of QoS rules using a single array.
- Decreased storage rack space requirements
- Reduced power and cooling costs



# Cost is a key driver

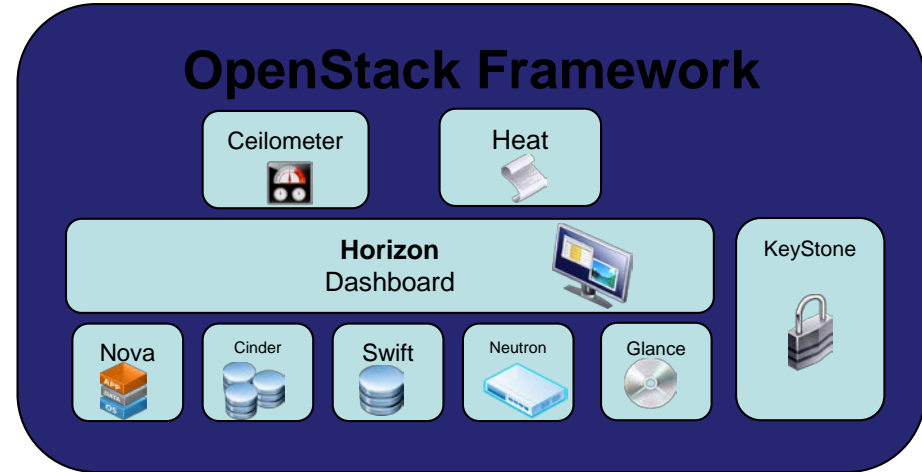
- All-flash solution costs have declined significantly over the past few years
- All-flash systems consistently moving to cheaper NAND



# OpenStack Overview

OpenStack framework includes the following modules:

- **Nova** for Virtual Machines / Compute
- **Cinder** for Block Storage
- **Swift** for Object Storage
- **Neutron** for Networking
- **Horizon** for Dashboard
- **Keystone** for Identity Services
- **Glance** for Image Services
- **Ceilometer** for Telemetry
- **Heat** for Orchestration



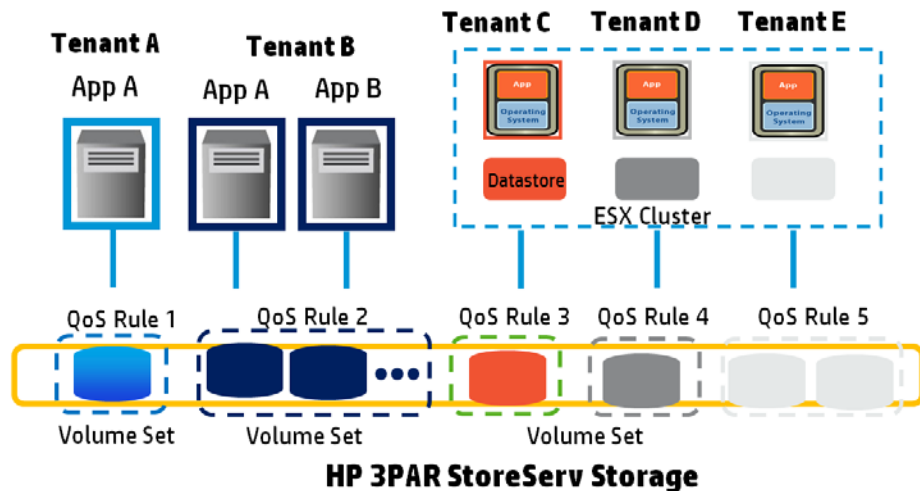
# Volume Type: QoS Specs

- Volume Type defines a collection of criteria/capabilities used to describe a particular service level.
- QoS Specs define Quality-of-Service(QoS) rules for volumes
- QoS Specs can be enforced either at
  - the storage subsystem “backend”
  - the hypervisor “frontend”
  - both
- Backend - Vendor specific rules (examples)
  - HP 3PAR (IOPS and bandwidth: min, max; latency goals, priority)
  - Solidfire (IOPS: min, max, burst)
- Frontend
  - Limit by throughput
    - Total bytes/sec, read bytes/sec, write bytes/sec
  - Limit by IOPS
    - Total IOPS/sec, read IOPS/sec, write IOPS/sec

# HP 3PAR - QoS

The QoS Specification within the 3PAR Cinder Driver supports the following:

- Minimum bandwidth goal
- Maximum bandwidth limit
- Minimum IOPS goal
- Maximum IOPS limit
- **Latency goal**
- **Priority**



# It's how you use it

- All-flash is rapidly becoming mainstream
- Cost and reliability are key in fueling this change
- Moving to newer and cost effective media is investable
- An architecture that can take advantage of this is critical

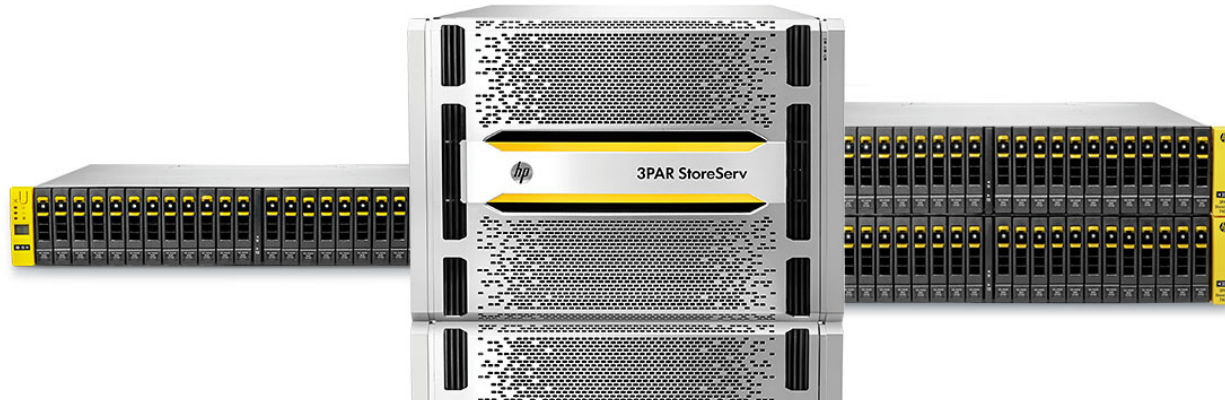
There can be economy only  
where there is efficiency  
- Benjamin Disraeli





Architecture. **Matters**

*HP 3PAR StoreServ Greatest Competitive Advantage*



Architecture Whitepaper : <http://h20195.www2.hp.com/V2/GetPDF.aspx%2F4AA3-3516ENW.pdf>

Flash Optimized Whitepaper : <http://h20195.www2.hp.com/V2/GetPDF.aspx%2F4AA4-7264ENW.pdf>

Thin Technologies Whitepaper: <http://h20195.www2.hp.com/v2/GetPDF.aspx%2F4AA3-8987ENW.pdf>

Priority Optimization (Storage QoS )Whitepaper: <http://h20195.www2.hp.com/V2/GetPDF.aspx%2F4AA4-7604ENW.pdf>

OpenStack Whitepaper: <http://www8.hp.com/h20195/v2/GetPDF.aspx%2F4AA5-1930ENW.pdf>