CHOOSING THE RIGHT STORAGE FOR YOUR OPENSTACK CLOUD

Federico Lucifredi
Product Management Director, Red Hat
federico@redhat.com @0xF2
CLOUD & STORAGE
BUSINESS NEEDS CLOUD STORAGE

- Illusion of Infinite Capacity
- On Demand Scalability
- Pay As You Go
- Self Service

- Massive scalability
- Easy to expand
- Elasticity
- No more guessing about future.

- API driven
- On demand rapid provisioning and operations.
- Speed and agility

- Unified Management
- Effective Monitoring and Metering.
- Deeper Integration.

- Robust User Interface
- Simplified API
- Multi-tenancy
STORAGE STRATEGIES
STORAGE IS ALL ABOUT WORKLOADS!

&

IT COMES IN ALL SHAPES AND SIZES!
STORAGE DESIGN

1. Qualify need for scale-out storage
2. Design for target workload IO profile(s)
3. Choose storage access method(s)
4. Identify Capacity
5. Determine fault-domain risk tolerance
6. Select data protection method

TARGET STORAGE ARCHITECTURE
UNDERSTANDING YOUR WORKLOADS

Latency
IO Size
Protocol
IO Pattern
IOPS
Capacity
Throughput
R/W %
OTHER FACTORS

- Reduced Costs
- BCP
- Hyper-convergence
- x-86 Architecture
- Management
- Monitoring
- Operations
- Advanced features
- Disaster Recovery
- Backup / Replication
- Thin Provisioning
- Dedup/ compression
- Erasure coding

STORAGE
OPENSTACK STORAGE
OPENSTACK NEEDS STORAGE
HYBRID STORAGE?

**SDS** is well aligned with translating illusion of infinite capacity into reality.

- Is SDS the right option for all storage requirements?
- Should I go with SAN/NAS storage backend?
- Should I have a hybrid storage strategy?
- An answer for this highly depends on how predictable the workload is and if the environment is
  - An on-premise private cloud
  - Or Public cloud.
SHARED STORAGE

Connecting each component to the same shared storage is ideal.

- Should we use different storage backends for each component?
  - Object Storage/NFS for Glance.
  - Local storage for Nova ephemeral
  - Block storage for Cinder.

- Should we use the same storage back end for each component?

- This is a better approach than connecting each component to different storage.
INTEGRATED STORAGE

How deep is the storage integration with openstack?

- Integration between Nova, Glance and Cinder when provisioning instance.
- Create a volume from image.
- Create image from volume.
- Managing Snapshots
- Backup

Does the storage vendor provide a driver to integrate OpenStack with Storage? If yes, is it tested and certified? To what extent it’s integrated?
INTEGRATED STORAGE

- **Ask Your Storage Vendor for a POC**
  - Technical explanation of how each functions are handled by the driver.
  - Showcase how much time it takes for large scale storage tasks to finish.
    - Booting 100+ instances via boot from volumes.
  - Is the driver certified by OpenStack vendor (Integration testing) and how is it distributed?
  - Can I integrate your storage using vendor deployment tools?

- **Benefits Of Strong Integration**
  - Saves disk space on compute nodes and storage nodes.
  - Saves network bandwidth.
  - Reduce the time required for the operations, especially at scale.
    - Rapid provisioning of storage for workload requirements.
**FUTURE-PROOF**

Is the selected storage going to allow you to meet your future storage requirements for IaaS+ and PaaS use cases? BigData as a Service, DBaaS, Manila, etc.

- Initiate a discussion with storage vendor to what extent it supports your future storage requirements.
  - Support for PaaS
  - Support for containers
  - DBaaS
  - Big Data as a Service
  - Manila - File sharing as a service
IN A NUTSHELL

1. Workload requirements and storage features

2. Hybrid strategy?

3. Does the vendor provide a driver to integrate the storage with OpenStack?

4. Is the driver tested and certified?

5. How does the driver handle various storage functions?

6. Are both vendors members of TSANet?

7. Is the deployment tool capable to deploy and integrate OpenStack to the storage?

8. Is it the storage future proof?
RED HAT CEPH STORAGE
ALL IN ONE

FILE

OBJECT

BLOCK

SOFTWARE-DEFINED STORAGE CLUSTER

iSCSI GATEWAY

NFS GATEWAY
# CEPH IS NOT JUST SCALE OUT CAPACITY

<table>
<thead>
<tr>
<th><strong>IOPS Optimized</strong></th>
<th><strong>Throughput Optimized</strong></th>
<th><strong>Cost / Capacity Optimized</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>NVMe SSD in SLED chassis</td>
<td>SSD, HDD in standard / dense chassis</td>
<td>HDD in dense / ultra-dense chassis</td>
</tr>
</tbody>
</table>

## IOPS Optimized
- High IOPS / GB
- Smaller, random IO
- Read / write mix

**Use Case:** MySQL

## Throughput Optimized
- High MB/s throughput
- Large, sequential IO
- Read / write mix

**Use Case:** Rich Media

## Cost / Capacity Optimized
- Low cost / GB
- Sequential IO
- Write mostly

**Use Case:** Active Archives
DATA PROTECTION SCHEMES

REPLICATED POOL
CEPH STORAGE CLUSTER

- FULL COPIES OF STORED OBJECTS
- Very high durability
- Quicker recovery

ERASURE CODED POOL
CEPH STORAGE CLUSTER

- ONE COPY PLUS PARITY
- Cost-effective durability
- Expensive recovery
MULTI-SITE CONFIGURATION

- Configure each Ceph Object Gateway to work in an active-active zone configuration, allowing for writing to non-master zone
- Global object storage clusters with a single namespace
- Enables deployment of clusters across multiple geographic locations
- Clusters synchronize, allowing users to read from or write to the closest one
RBD MIRRORING

- Multi-site replication for block devices
- Replicates virtual block devices across regions
- Designed for disaster recovery and archival
- Integration with Cinder Volume Replication (OSP-10)
BACKUP STRATEGIES

- Volume backup with cinder backup driver
- Backs up volumes of any type to a Ceph back-end store
- Volume snapshot with cinder volume snapshot
- Establish backup policies for datas in the VMs
BlueStore is a new Ceph storage backend optimized for modern media

- Replaces FileStore, which was designed for HDDs
- Supports flexible media topologies (flash, K/V drives, persistent memory)
- Eliminates the need for an underlying filesystem or dedicated journal device
- Provides a 2-3X performance boost
# RED HAT CEPH TECHNICAL REFERENCES

<table>
<thead>
<tr>
<th>Resource</th>
<th>URL</th>
</tr>
</thead>
</table>
BETTER TOGETHER
THANK YOU

Federico Lucifredi
Product Management Director, Red Hat
federico@redhat.com @0xF2