IBM Spectrum Storage

IBM Spectrum Storage

**Spectrum Scale Cloud Team Initiatives**
**May 16 2018**
**Spectrum Scale User Group Meeting**

**John Lewars and Gautam Shah – Spectrum Scale Development**
**jlewars@us.ibm.com and gautam@us.ibm.com**

IBM

- The information presented on these charts does not represent any official commitment in terms of Spectrum Scale future direction or deliverables.

- IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

- Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

- The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

- The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

# Agenda

- Work Items for Spectrum Scale Cloud and Object Team

- Public and Private Cloud Enablement in Spectrum Scale

- Details on Spectrum Scale on AWS
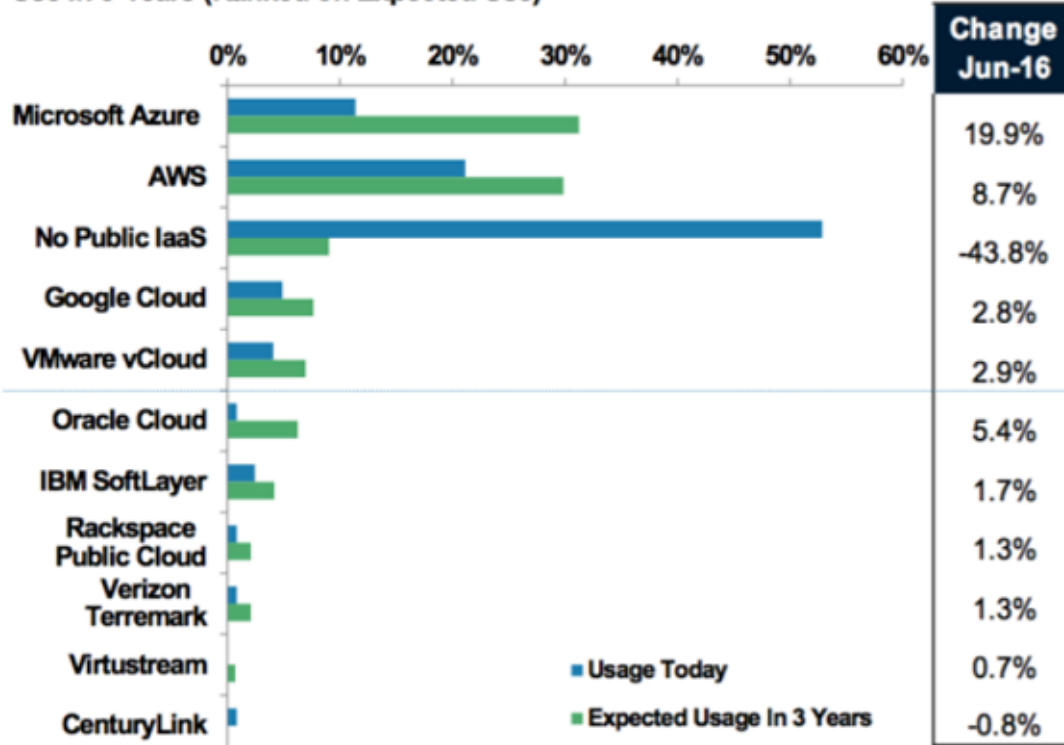
- Spectrum Scale Container Support

# Spectrum Scale Cloud And Object Team Work Items

**IBM®**

1. Enable Spectrum Scale on the following Cloud Environments and expand the flexibility of our offerings to better support hybrid environments.

    a) Delivered a Trial Offering support of Spectrum Scale on Amazon (AWS) on Sept. 13 2017, have made updates (in November, February, and April), and intend to make a production offering available in 2018

    b) Worked on Scale +ICP Use cases (e.g. DB2 Warehouse running with Spectrum Scale and the hostPath deiver – see a blog on ICP + Scale + hostPath <u>here</u>) and we have prototyped using CAM to automate Scale deployment with ICP

    c) Support available Spectrum Scale offerings on SoftLayer and support/collaborate on these offerings

    d) We've prototyped Spectrum Scale support on Azure and are doing further investigations (seeking customer input on Azure and have started engaging in one collaborative effort)

    e) We've recently started investigating running Spectrum Scale on Google Cloud

    f) Support OpenStack Cloud environments via deliverables such as Cinder & Manila driver support

2. Support and enhance our OpenStack Swift based Object Storage solution which expands the methods by which Spectrum Scale data can be accessed. For example we have updated our solution to run on the Pike Open Stack release support for our object solution in Spectrum Scale 5.0.1

3. Investigate better integration, and more flexible support of storage for, virtualized environments (focused on container support)

4. Working on providing automation for application and verification of performance related tunables/configuration options and working with other teams on how to better automate Scale deployment

# Public and Private Cloud Enablement

# Where Market Data Suggests Cloud Dollars Are Heading

Exhibit 15: Cloud-based IaaS Use Today and Expected Use in Three Years

**% of Respondents Using Cloud-based IaaS Today and Expected Use In 3-Years (Ranked on Expected Use)**

| | Change Jun-16 |
|---|---|
| Microsoft Azure | 19.9% |
| AWS | 8.7% |
| No Public IaaS | -43.8% |
| Google Cloud | 2.8% |
| VMware vCloud | 2.9% |
| Oracle Cloud | 5.4% |
| IBM SoftLayer | 1.7% |
| Rackspace Public Cloud | 1.3% |
| Verizon Terremark | 1.3% |
| Virtustream | 0.7% |
| CenturyLink | -0.8% |

Legend: ■ Usage Today ■ Expected Usage In 3 Years

Source: n=100 (US and EU data), Source: AlphaWise, Morgan Stanley CIO Survey

Morgan Stanley

http://www.networkworld.com/article/3091083/cloud-computing/microsoft-could-overtake-amazon-in-the-cloud-morgan-stanley-survey-finds.html#tk.rss_cloudcomputing

# Drivers for Spectrum Scale in Cloud Environments

- An increasing number of customers are seeing pressure to move workloads to public clouds (e.g. government mandates requiring public cloud consideration)

- Offers a way to try things without infrastructure investments (POCs, code evaluation, etc.)

- Provides operational expense model focus (over capital expense model)

- On demand provisioning and rapid deployment

- In an agile manner, can more efficiently adapt deployment configuration to meet the specific needs of the use case (particularly for transient/temporary use cases)

- Lower cost for some workloads (particularly 'bursty' workloads)

- **Other observations:**
    - Several workloads are moving to the cloud (private and public)
    - Some (new) companies have only deployments on the cloud (born in the cloud)

# Spectrum Scale on AWS

# AWS Spectrum Scale Quick Start

- IBM has collaborated with Amazon to release a Spectrum Scale Quick Start (launched Sept. 13/17)
https://aws.amazon.com/quickstart/architecture/ibm-spectrum-scale/

- Quick Start provides a 90 trial license and is supported via Spectrum Scale mailing list and forum (not for production use)

- Follows Best Practices for AWS deployments, e.g., security and high availability. We provide an HA multi-AZ solution by default but (as of Feb. 2018) allow for configuring single AZ replication for performance and cost savings reasons

- Support for creation of new VPCs or launching instances into existing VPCs

- Leverages Amazon's security best practices.  Access to the Spectrum Scale cluster is through a Bastion host created by Amazon developed template

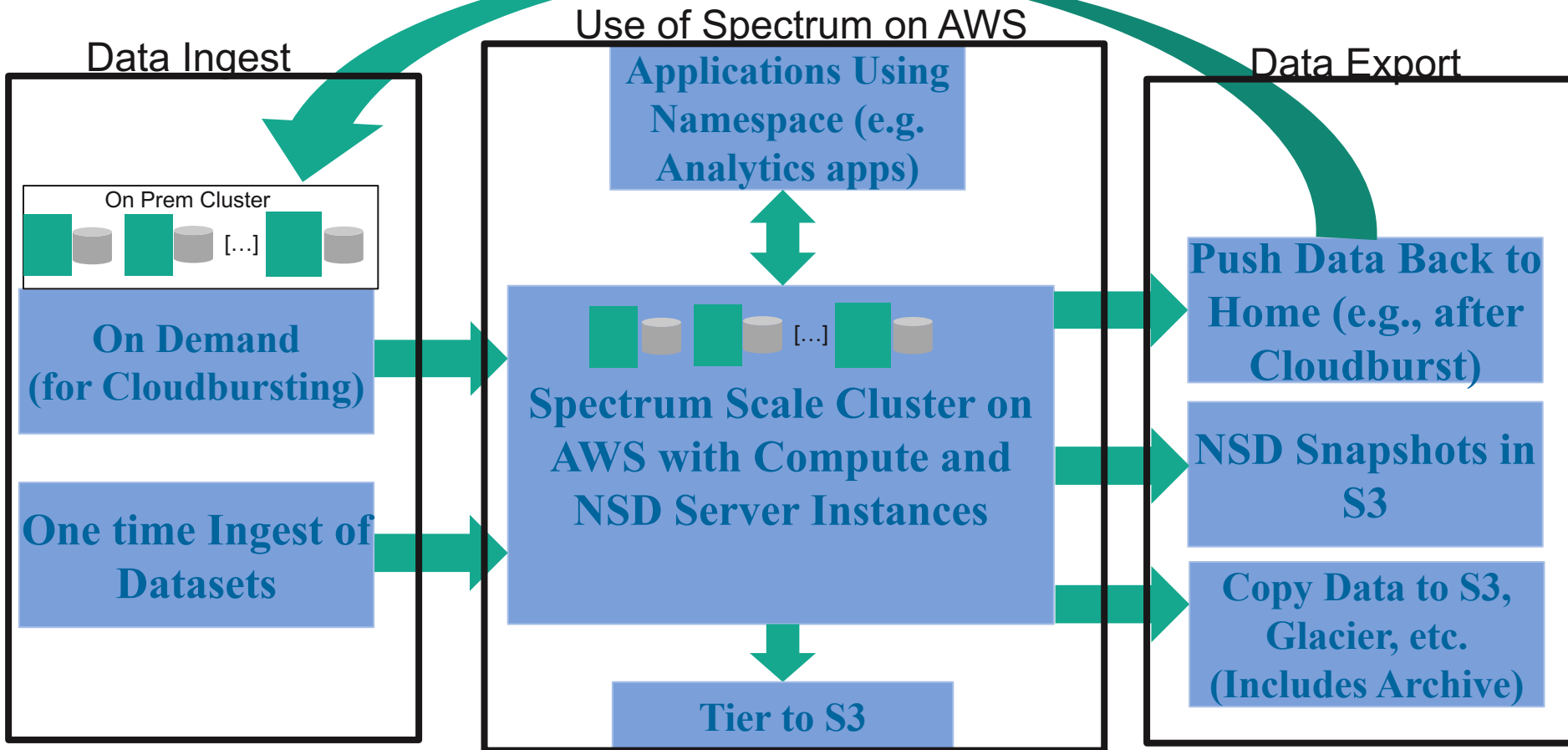# AWS Spectrum Scale Quick Start (cont)

- Focuses on simplicity/usability of deployment (e.g. reduced config options) and leverages Amazon features such as Cloud Formation templates and AMIs (Amazon Machine Image)

- We deploy using auto-scaling (for efficiency) then disable auto-scaling in favor of auto-recovery (to allow failed instances to be relaunched with same root device image, same IP address, etc.)

- Clusters are created with two Availability Zones (AZ's) in a single region and 2X Spectrum Scale replication is used by default (we deploy the Spectrum Scale cluster into a number of AZs according to the data replication value that is selected by the customer)

- Limited number of configuration choices (e.g. EBS volume types, EC2 instances types, networking options, etc) allows for better customization, more expansive testing, performance measurements, etc

IBM Spectrum Scale on AWS Architecture Diagram
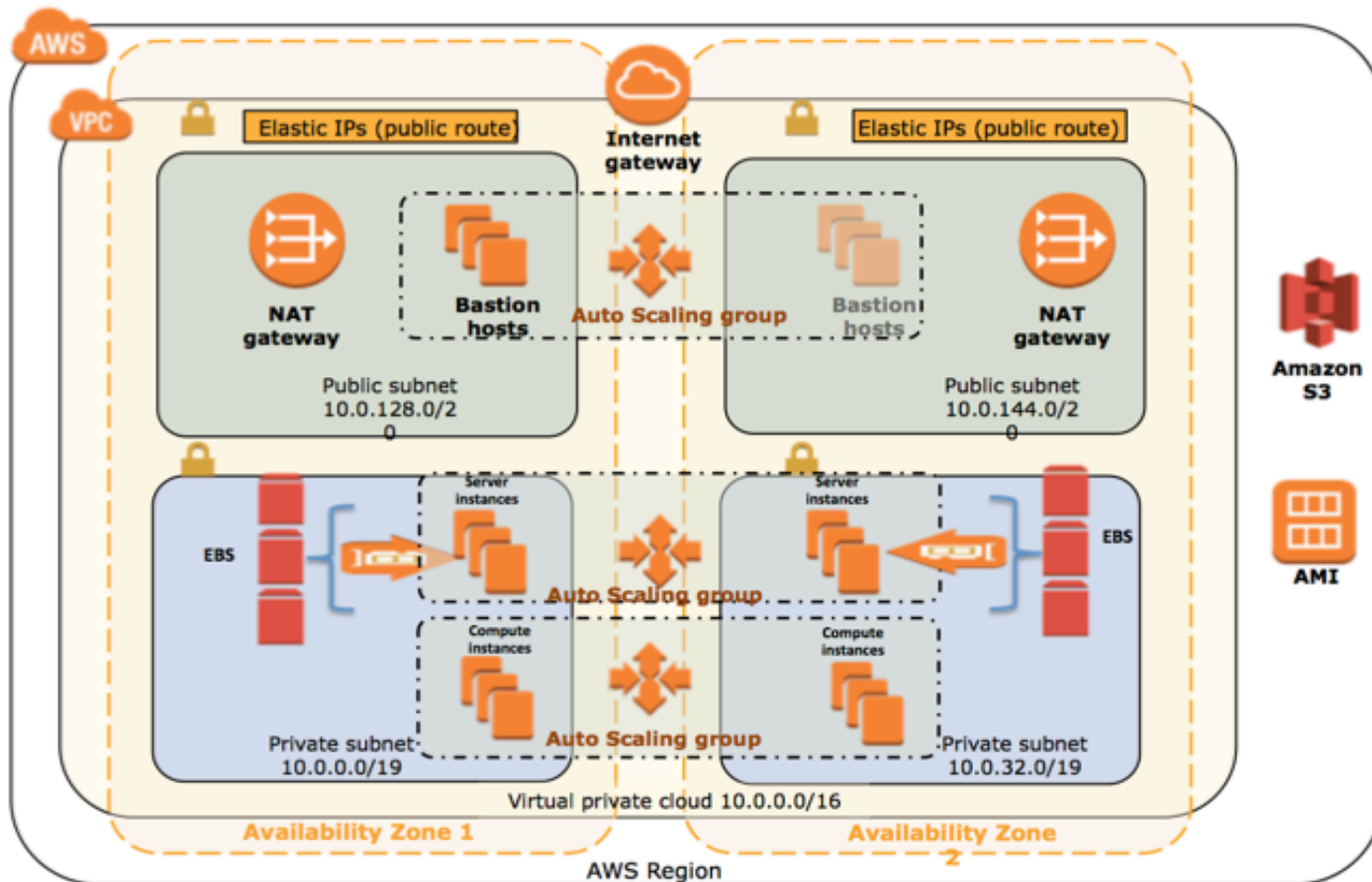
# Past AWS Updates and Intended Plans Going Forward

- AWS2 Update (completed Nov. 2017)
  - Cleaned up AMI (removed unneeded elements)
  - Decoupled metadata replication from data replication (2 metadata replicas and 1 or 2 data replicas)
  - Fixed size for root volume (100 GiB)
- AWS3 Update (released Feb. 2018)
  - Single data replica and Availability Zone support for customers desiring:
    o Reduced costs (no cross AZ data movement charges)
    o Improved write performance (only one copy of data is written)
  - Code refactoring to improve scaling (deployment efficiency improvements based on input from Amazon)
- AWS4 Update (released Apr. 2018)
  - Spectre/Meltdown vulnerability fixes
  - Support for automating the expansion/contraction of clusters (delivered as samples)
- Next
  - We intend to make a supported G/A version (with restrictions) available for production usage and, based on customer feedback, continue to enhance this offering (e.g. remove restrictions and integrate with AWS ecosystem)

# Data Flow Options on AWS

## Data Ingest

### Use of Spectrum on AWS

## Data Export

**Applications Using Namespace (e.g. Analytics apps)**

On Prem Cluster

[…]

**On Demand (for Cloudbursting)**

**One time Ingest of Datasets**

**Spectrum Scale Cluster on AWS with Compute and NSD Server Instances**

[…]

**Tier to S3**

**Push Data Back to Home (e.g., after Cloudburst)**

**NSD Snapshots in S3**

**Copy Data to S3, Glacier, etc. (Includes Archive)**

# Spectrum Scale Cloud Related Guides/Publications

- Spectrum Scale Amazon AWS Quick Start, Deployment Guide, and demo videos:
  https://aws.amazon.com/quickstart/architecture/ibm-spectrum-scale/
  https://s3.amazonaws.com/quickstart-reference/ibm/spectrum/scale/latest/doc/ibm-spectrum-scale-on-the-aws-cloud.pdf

- https://youtu.be/kc8JPAEzkPE

- https://www.youtube.com/watch?v=6j5Xj_d0bh4&t=21s


- Guide for running Spectrum Scale on SoftLayer systems:
  http://www.redbooks.ibm.com/redpapers/pdfs/redp5410.pdf


- OpenStack Redpaper:
  http://www.redbooks.ibm.com/redpapers/pdfs/redp5331.pdf


- Redpaper for Spectrum Scale Unified Object Access :
  http://www.redbooks.ibm.com/abstracts/redp5113.html


- Knowledge center OpenStack documentation for Spectrum Scale:
  https://www.ibm.com/support/knowledgecenter/en/STXKQY_4.2.1/com.ibm.spectrum.scale.v4r21.doc/bl1ins_openstackusecase.htm

# Questions On Spectrum Scale in the Public Cloud:

Would you considering running Spectrum Scale on AWS as part of a trial (if the software was made available for free)?

Would you pay for Spectrum Scale software on AWS?

Would you considering running Spectrum Scale on Azure as part of a trial (if the software was made available for free)?

Would you pay for Spectrum Scale software on Azure?
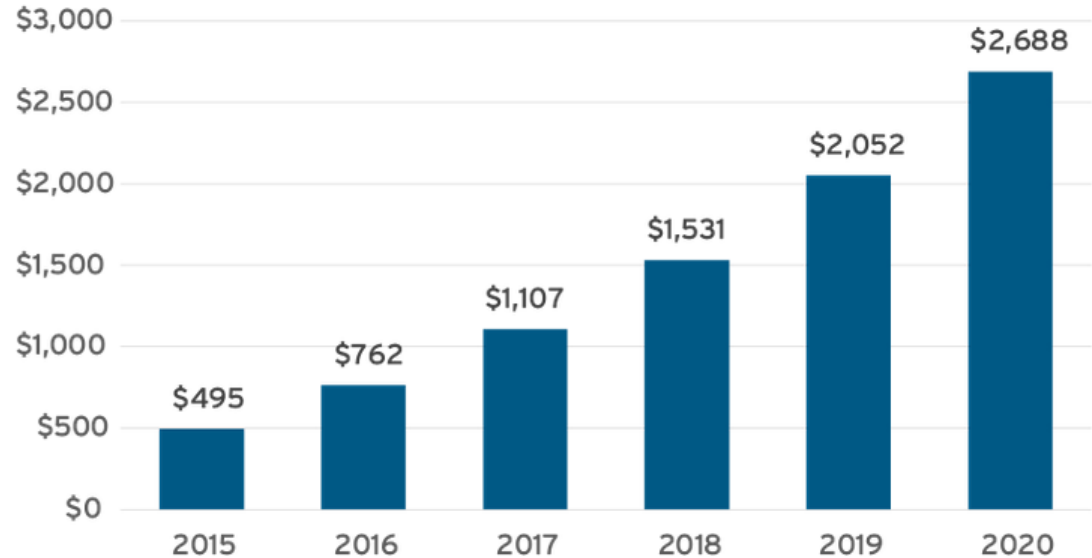
What about any other public cloud offerings?

# Spectrum Scale Container Support

# Container Projected Growth Rate per 451 Research

- Containers growth predicted to be at high rate:
  - http://www.itworldcanada.com/article/application-container-market-small-but-mighty-451-research-says/389916



Container revenue between 2015-2020, according to 451 Research.

# Containers

Self Contained (Portable)

Data Access Isolation (Multi-Tenancy)

Data Management

High Performance Data Access

Fast and lightweight

Resource Utilization

Open-Source

Global Repositories

# High Level Strategy for Containers

1. Team is has verified the use of GPFS and NFS (exported GPFS file systems) in a variety of environments, including Kubernetes and Docker Swarm.
   These investigations have included running/deploying workloads in containers.

2. Our product team has re-engaged in the Open Source Ubiquity project: https://github.com/IBM/ubiquity and we are addressing regressions that were introduced into the Ubiquity driver's support for Spectrum Scale.

   Ubiquity is not part of the Spectrum Scale product but we intend to support Spectrum Scale and Ubiquity (support is tentatively scheduled for 3Q 2018 but we are starting to engage in POCs now)
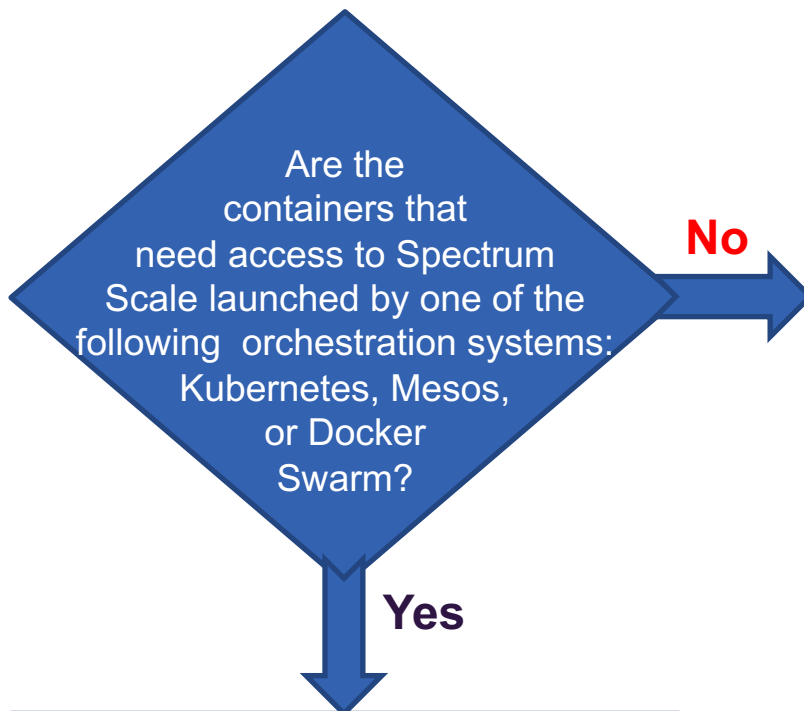
   We're looking to understand workload use cases to determine gaps in solutions based on the hostPath Kubernetes driver, and what solutions may require Ubiquity driver support

3. We are focusing on meeting the requirements of Spectrum Compute's IBM Cloud Private product (which uses Kubernetes), and investigating options for new integration.

4. We're interested in customer feedback on the use case for Spectrum Scale in containers.

# What is Possible with Spectrum Scale and Containers?

- Can we use Spectrum Scale Storage within Docker containers?
  - Yes - Any Spectrum Scale storage (be it native protocol or a Spectrum Scale filesystem that has been exported via NFS) can be made available inside Docker containers
  - We can manually provision, mount, and attach (via the docker '-v' run flag) Spectrum Scale volumes to Docker containers, e.g.'-v /gpfs/projects/project1:/gpfs'  or '-v /gpfs/projects/project2:/gpfs'

/gpfs/projects/project1 ➡ /gpfs  (Container 1)
/gpfs/projects/project2 ➡ /gpfs  (Container 2)

Host mounted fs pathname          Bind mount in Container

- Can we automate this (bind mount) process via a plug-in Docker driver support?
  - Yes--hostPath driver supports automating bindmount function for Kubernetes
  - Docker/Mesos/Kubernetes driver plug-ins are available via the Ubiquity Open Source product

- Can we currently containerize Spectrum Scale (i.e., deliver Scale via a container)?
  - No –this is currently not supported.  Note the current kernel extension design of Spectrum Scale would require running it in a privileged container.

# Accessing a Scale file system from a container- What do I need?   Step 1

Are the containers that need access to Spectrum Scale launched by one of the following  orchestration systems: Kubernetes, Mesos, or Docker Swarm?

**No**

**Yes**

If you're using Kubernetes, Mesos, or Docker Swarm orchestration software, you may access Scale via a Ubiquity service plug-in, or (only on Kubernetes) the hostPath driver.

Proceed to step 2 for details.

If there is no orchestration system in place, you should not need a storage driver/plug-in for Spectrum Scale access.  Some possible cases:

1) If you're launching containers directly via a "docker run" command line statements, then you may use the docker  '-v' flag (or, as of Docker 17.06, the '--mount' flag) to bind mount a Spectrum Scale file system into your container, e.g. the Scale file system '/scale/scratch' mounted on the host could be made available at the mount point '/scratch' in a newly launched container as follows:
docker run –v /scale/scratch:/scratch [. . .]
or
docker run --mount type=bind,source=/scale/scratch,\
target=/scratch [. . .]

2) If you are running a workload management platform, or High Performance job scheduler that has support for running containers (for example IBM Spectrum Compute LSF), it may already create any bind mounts needed by jobs that launch containers.  For example, IBM LSF can run workloads in containers started via 'docker run' command with the '-v' flag set to automatically mount any Spectrum Scale directory required, including the job's current working, spool, temporary, checkpoint, and $LSF_TOP directories.

IBM Spectrum Storage

IBM

I'm using Kubernetes, Mesos, or Docker Swarm

Are you using Kubernetes?

**No**

**Yes**
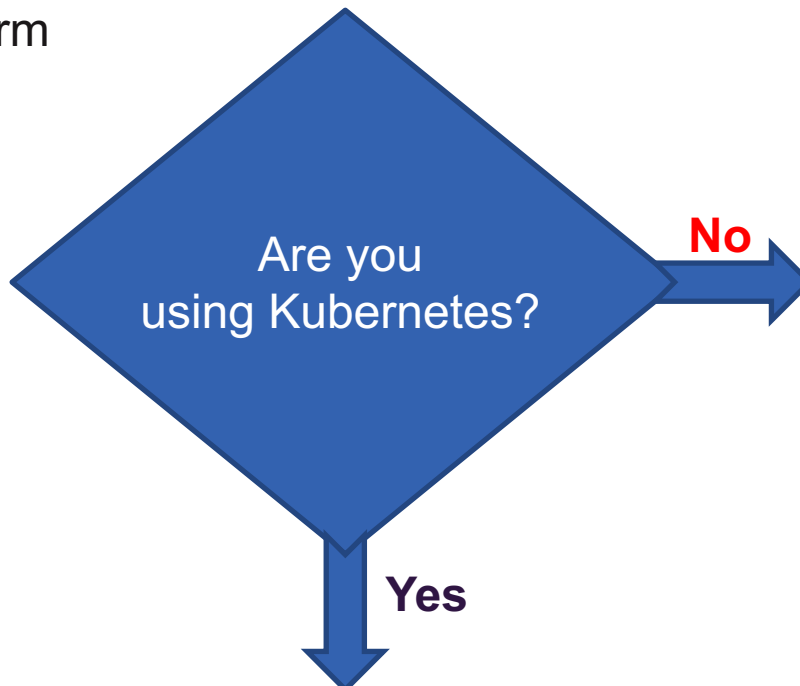
If you're using Kubernetes then, depending on your use case, you may choose to use the hostPath driver or the Ubiquity plugin for Kubernetes.
Proceed to step 3 for details.

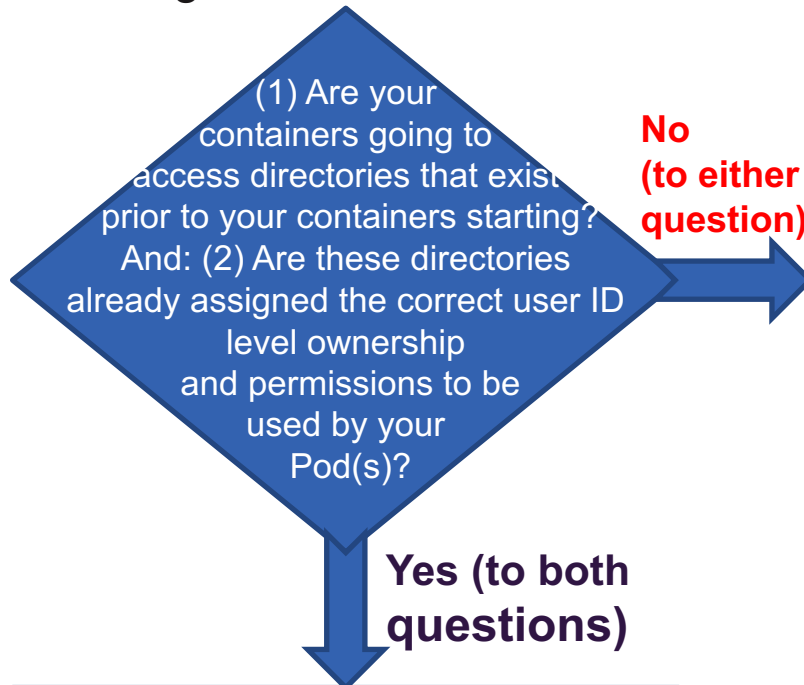1) IBM has tested the Ubiquity docker volume plug-in to provide Spectrum Scale access to containers launched from both Docker Swarm and Mesos.

2) The hostPath driver is available for Kubernetes but, with docker swarm, we can bind mount the volume using a "docker service create" command when starting the service being launched, as per this example:

```
docker service create --replicas 1 --mount
type=bind,source=/gpfs/gpfs_device,destination=
/gpfs/gpfs_device --name helloworld1 alpine
ping docker.com
```

Using the command from this example, if the service fails, it will get restarted on different node with the same bind mount in effect, provided the same path exists (/gpfs/gpfs_device) on the new service node.

IBM Spectrum Storage

IBM

## I'm using Kubernetes Orchestration

(1) Are your containers going to access directories that exist prior to your containers starting? And: (2) Are these directories already assigned the correct user ID level ownership and permissions to be used by your Pod(s)?

**No (to either question)**

**Yes (to both questions)**

If you're using Kubernetes and all of the directories your Pods want to access:
(1) exist already and
(2) have the ownership and permissions set to allow your Pods to access them,
then the the hostPath driver should meet your use case.

If you answered 'No' to either question, the Ubiquity plugin for Kubernetes may be the appropriate choice for your workload but read the following details to verify:

(1) The ability for a Pod to request the creation of new a Kubernetes volume as it launches refers to a concept known as Dynamic Provisioning in Kubernetes.  The Spectrum Scale Ubiquity project plans to continue support two types volumes, based on the two types of Spectrum Scale filesets:
(i)  Spectrum Scale independent filesets                (1000   supported per file system),
(ii) Spectrum Scale dependent filesets                (10000 supported per file system).
If you require higher volume scaling (from a single file system) your use case may not be appropriate for Spectrum Scale.

Note that Dynamic provisioning in the context of Spectrum Scale refers to the dynamic creation of Spectrum Scale directories and filesets, and does not imply the automation of Spectrum Scale installation, cluster and file system creation/configuration, etc.

You should also consider some characteristics of snapshots if you plan to use them with Spectrum Scale filesets.  Note that each independent fileset supports up to 256 snapshots, and every dependent fileset belongs to one independent fileset (all dependent filesets share their parent's snapshot limit).

There may be a performance degradation observed with using large number of filesets and/or snapshots. Note that independent filesets have their own dedicated inode space which enables more efficient per-fileset functions, such as fileset snapshots.

(2) In a statically provisioned use case, all directories are created up front, and the Kubernetes hostPath driver should meet this use case if  these storage volumes are created with the ownership and permissions that match the user ID of the pod that tries to use them (the hostPath driver allows for group ID permission changes at run time but not user ID permission changes).

Therefore the Ubiquity plugin for Kubernetes is needed for workloads that don't have all their volumes created in advance with the correct directory ownership and permissions assigned.

# Ubiquity Github

Ubiquity is not part of the Spectrum Scale product but available via github:

**Ubiquity Service**           https://github.com/IBM/ubiquity

**Ubiquity Docker Plugin**    https://github.com/IBM/ubiquity-docker-plugin

**Ubiquity K8s DP and FV**   https://github.com/IBM/ubiquity-k8s

Note that there are issues with Spectrum Scale support after V0.4.  Until our Development team provides a new version of the code (tentatively in 3Q 2018), any customers seeking to use Spectrum Scale with Ubiquity should reach out to our development team to engage in a POC.

# Questions Regarding Container Usage

- Are you using containers?   Any comments on workloads that are applicable to containers? Are there aspects of the value proposition of containers that we've missed discussing?

- What orchestration method(s) (e.g. Kubernetes, Docker Swarm) are you looking at?

- Do you see containers and cloud coming together?  Does anyone have opinions on what else the Spectrum Scale development team should be doing in this area?

# Backup

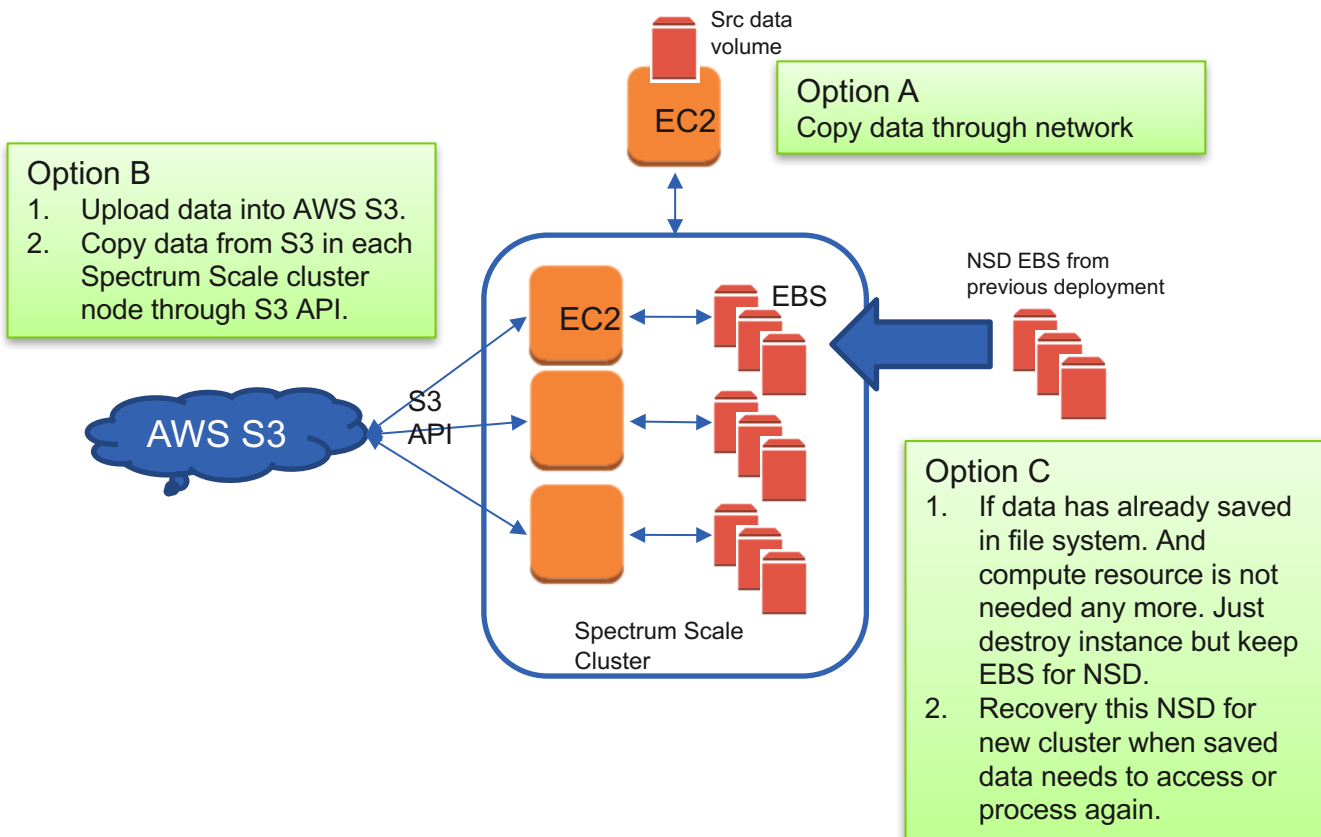# High Level Strategy for OpenStack

Support OpenStack Cloud environments with Cinder & Manila drivers and investigation of best practices (e.g. multi-tenancy recommendations and accessing Spectrum Scale data from VMs)
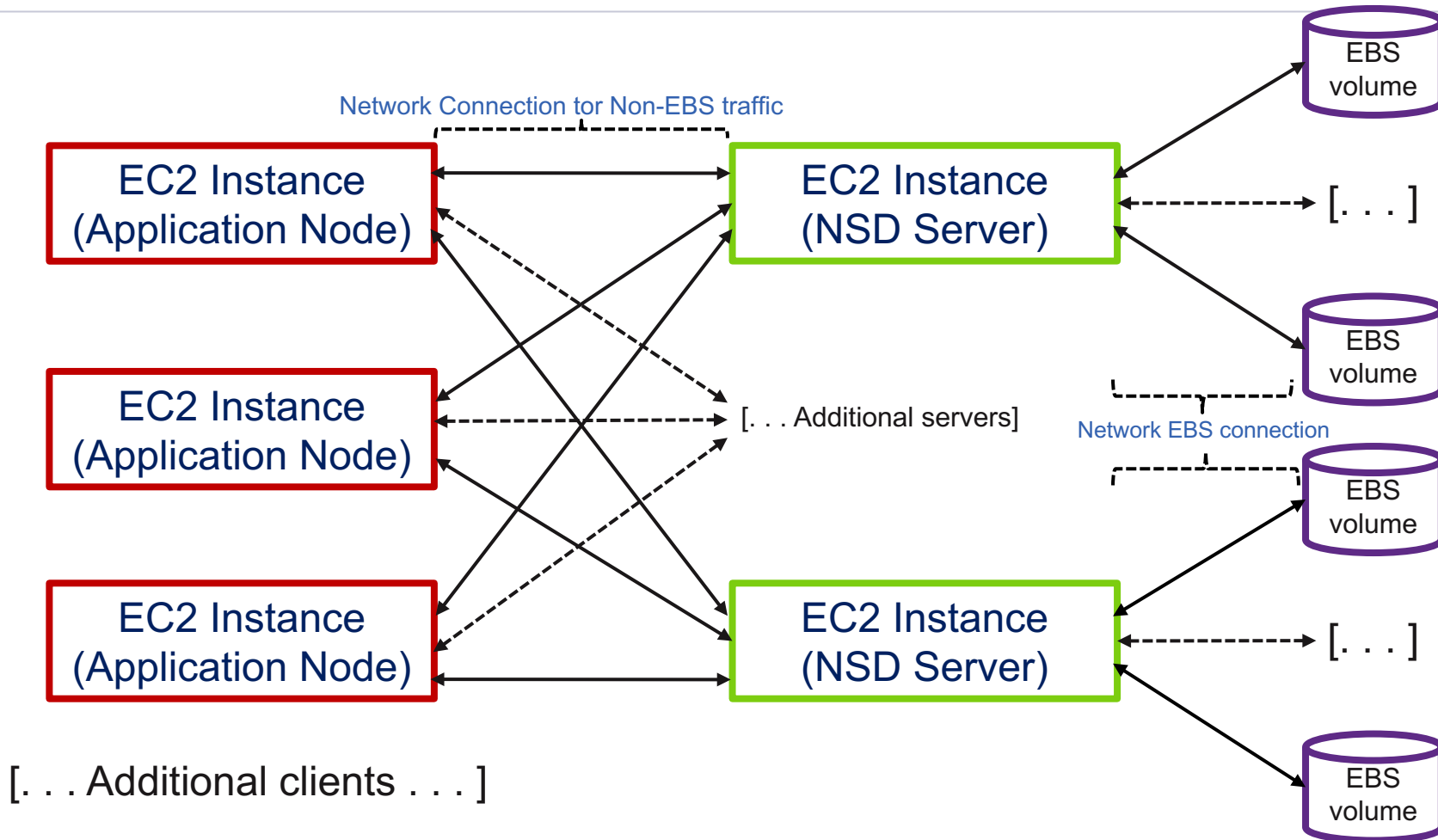
**Questions:**

1) Who is using OpenStack now?

2) Why are you using it or why are you not using it?

3) Are you thinking about using it in the future?

4) What else would you like to see from Spectrum Scale in terms of OpenStack support?

5) Do you agree that OpenStack has become a private cloud play?
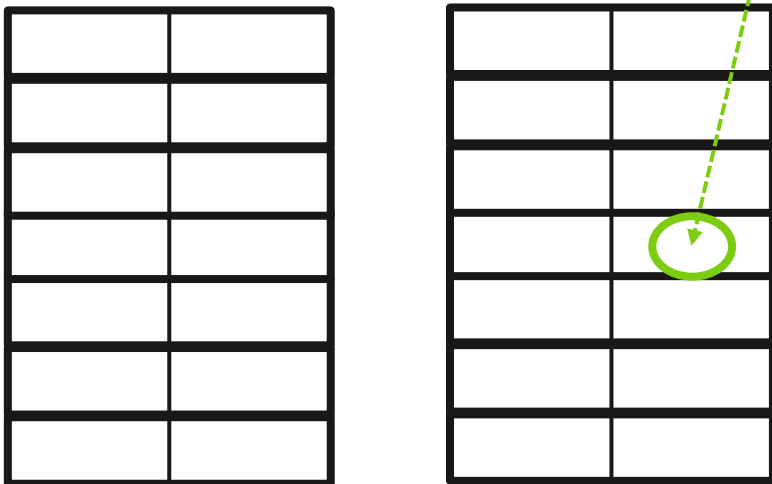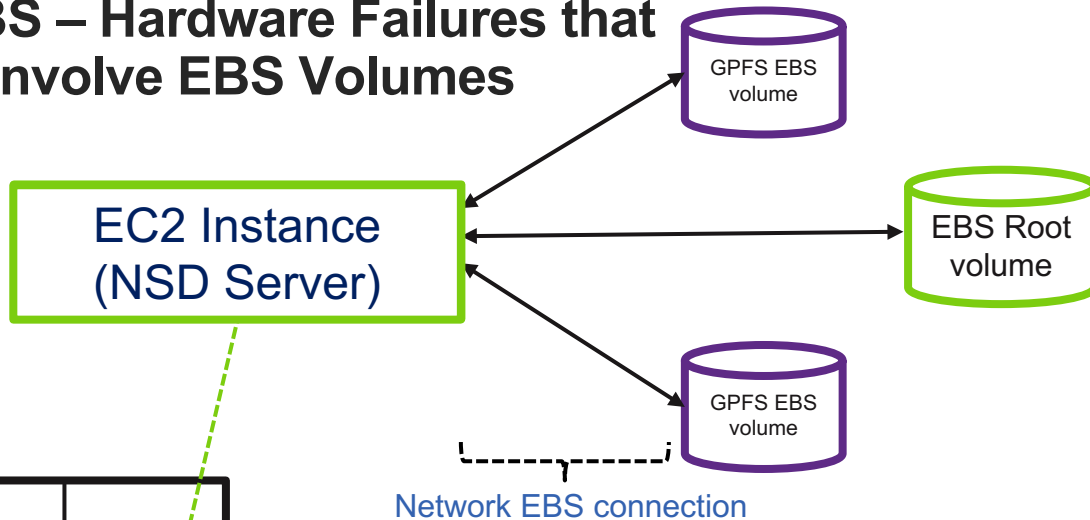
# **OpenStack**

# Details on AWS Solution

# Data Ingest Options for Spectrum Scale Clusters on AWS

IBM Spectrum Storage

IBM

Src data volume

EC2

**Option A**
Copy data through network

**Option B**
1. Upload data into AWS S3.
2. Copy data from S3 in each Spectrum Scale cluster node through S3 API.

AWS S3

S3 API

EC2

EBS

NSD EBS from previous deployment

Spectrum Scale Cluster

**Option C**
1. If data has already saved in file system. And compute resource is not needed any more. Just destroy instance but keep EBS for NSD.
2. Recovery this NSD for new cluster when saved data needs to access or process again.

# AWS Spectrum Scale Using EC2 Instances Backed by EBS

# AWS Spectrum Scale Using EC2 Instances Backed by EBS – Hardware Failures that Do Not Involve EBS Volumes

# AWS Spectrum Scale Using EC2 Instances
## Backed by EBS – Hardware Failures that
## Do Not Involve EBS Volumes

GPFS EBS volume

EC2 Instance
(NSD Server)

EBS Root volume

GPFS EBS volume

Network EBS connection

In the case of a hardware failure leading to termination of an EC2 instance, if the hardware failure is not tied to an EBS volume used by the instance, simply restarting the instance should allow it to be launched on new hardware, using the same EBS root volume and GPFS data disks.

# Approaches to Handling Failures on EBS Volumes Used by Spectrum Scale

1. Run without any Spectrum Scale replication but this assumes a willingness to accept potential data loss if an EBS volume used for GPFS data/metadata fails.  (This approach assumes that snapshots/backups of the Scale data are taken to allow recovery from such failures).  The SLA Amazon provides for EBS volumes should be carefully reviewed before choosing this option.
From https://aws.amazon.com/ebs/details/#elasticvolumes:

> Amazon EBS volumes are designed for an annual failure rate (AFR) of between 0.1% - 0.2%, where failure refers to a complete or partial loss of the volume, depending on the size and performance of the volume. This makes EBS volumes 20 times more reliable than typical commodity disk drives, which fail with an AFR of around 4%. For example, if you have 1,000 EBS volumes running for 1 year, you should expect 1 to 2 will have a failure.

See Amazon EC2 and EBS Service Level Agreement. for full details


2. Use Spectrum Scale replication to replicate all GPFS EBS volumes used by GPFS.  Two-way data and metadata replication is suggested, but other options could be considered, depending on data protection requirements (e.g. 2 way data replication and 3-way metadata replication will allow the filesystem to stay mounted, should two separate NSD servers experience failures related to GPFS EBS volumes).
In the case of an EBS volume used by GPFS failing, the 'mmrestripefs –r' command can be used to re-replicate data across the disks and restore the cluster replication levels.

# AWS Spectrum Scale Using EC2 Instances Backed by EBS
## EBS Root Volume Hardware Failures



EC2 Instance (NSD Server)

EC2 Instance (NSD Server)

GPFS EBS volume

EBS Root volume

GPFS EBS volume

Network EBS connection
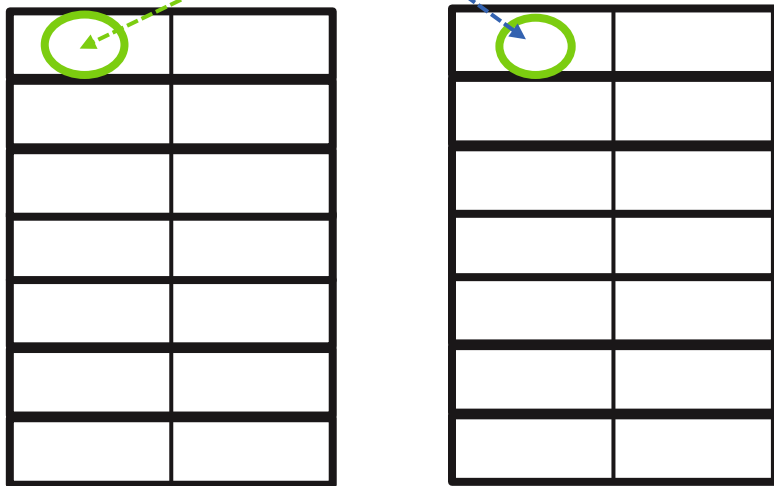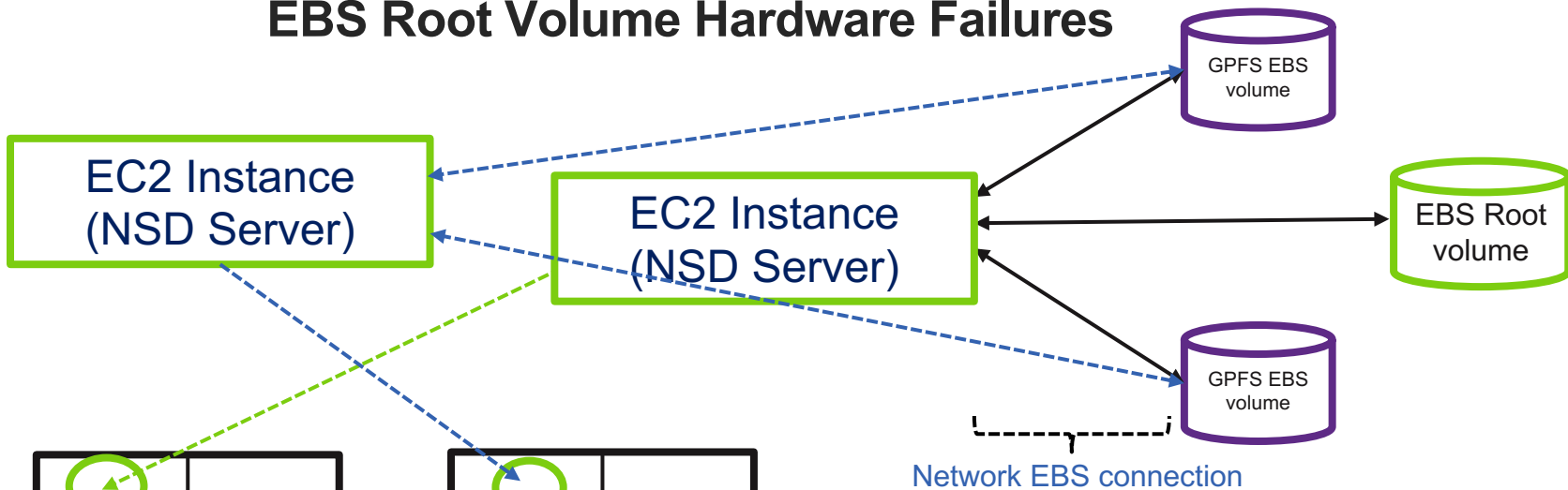
A hardware failure causing the loss of an EBS root volume will result in the termination of an EC2 instance, thereby preventing that instance from being reactivated.  The loss of any GPFS data volumes in such a failure case can be handled as per the previous slide's recommendations. Alternatively, (but this requires unmounting the filesystem), any EBS volumes assigned to the failing instance can be reassigned via mmchnsd to existing instances in the cluster or a new instance can be activated to replace the failed instance.

# Thanks!