



# Container-Native Storage for Modern Applications with OpenShift and Red Hat Gluster Storage

Driving the Future of Storage

Sayan Saha  
Director, Product Management, Storage Business, Red Hat

Michael Adam  
Engineering Manager, Container-Native Storage, Red Hat

Annette Clewett  
Senior Storage Architect, Storage Business, Red Hat

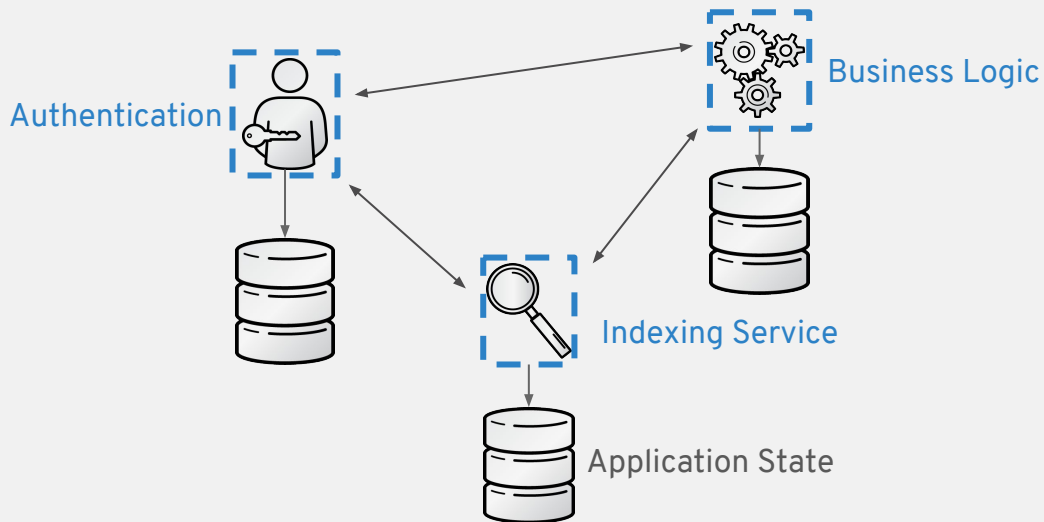
Daniel Messer  
Technical Marketing, Storage Business, Red Hat

# The Journey So Far

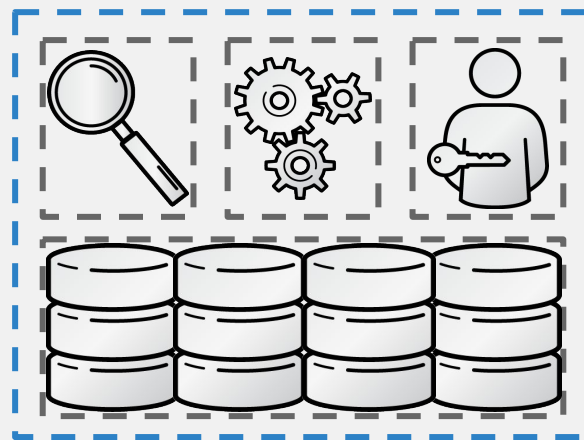
**Containers are for stateless workloads only, right?**

# Containers need Persistence

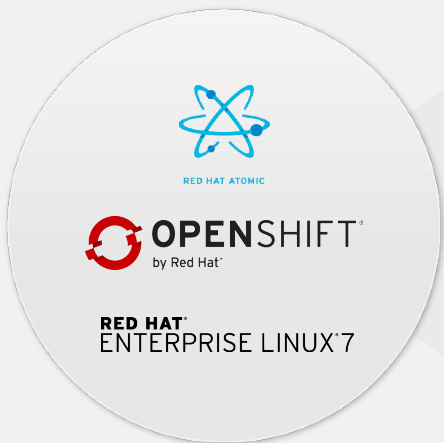
## Microservice Application



## Monolithic Application



# OpenShift supports Persistent Storage



GlusterFS

Amazon EBS

Azure Disk

GCE Disk

iSCSI

NFS

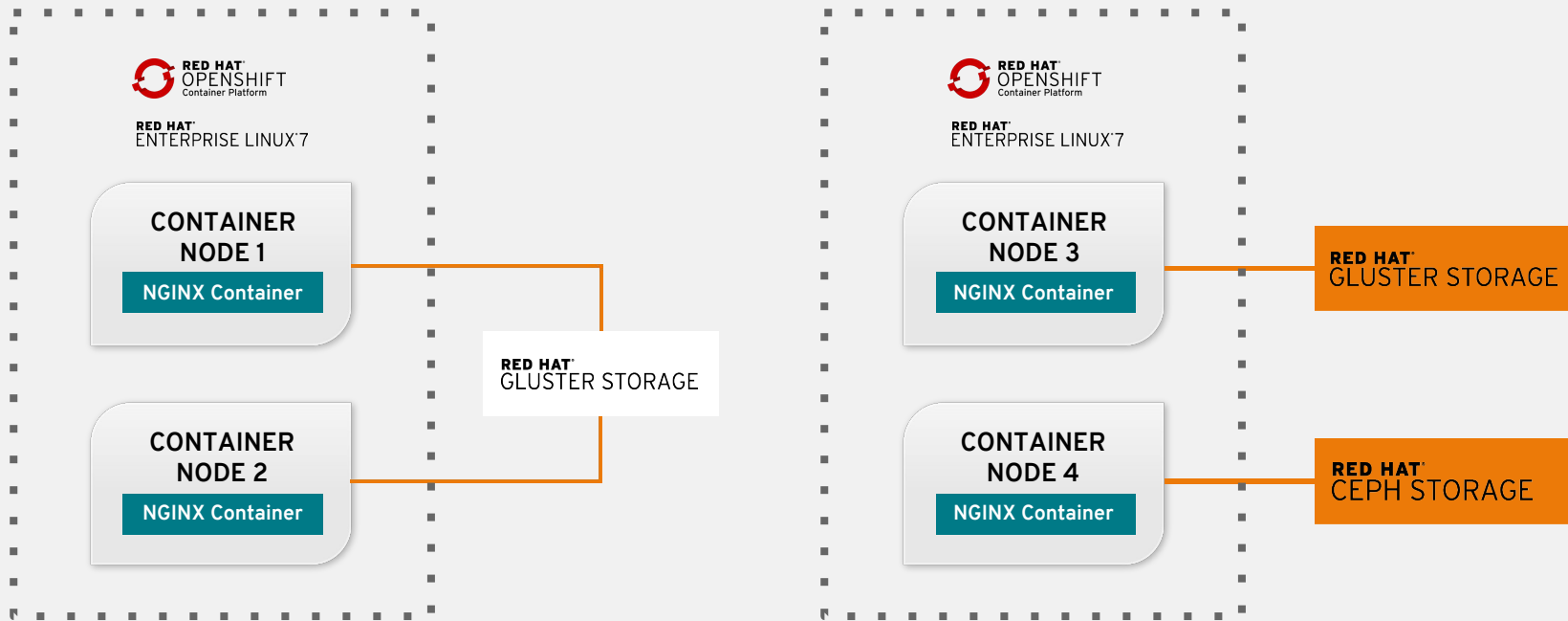
Ceph RBD

AUTOMATED CONFIGURATION

SINGLE CONTROL PANEL

CHOICE OF PERSISTENT STORAGE

# GlusterFS Support in OpenShift



Scalable



Highly-Available



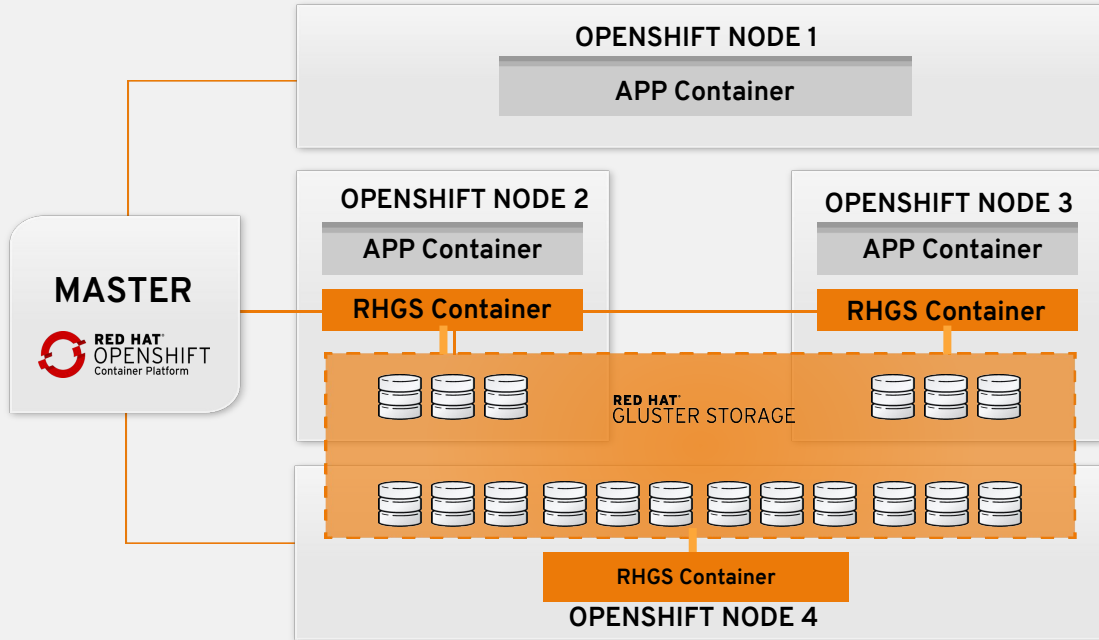
Automated



Elastic

**What if we could provide that out of  
the box?**

# Introducing Container-Native Storage



Co-Locate Storage and Apps

Dynamic Provisioning

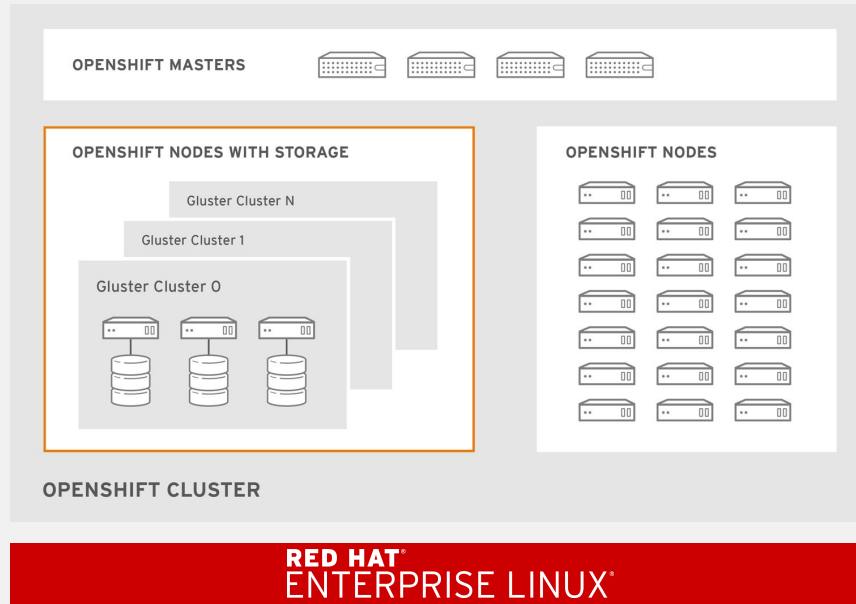
Managed by OpenShift

Infrastructure-Agnostic



# Why would I want this?

# Consistent Storage Experience

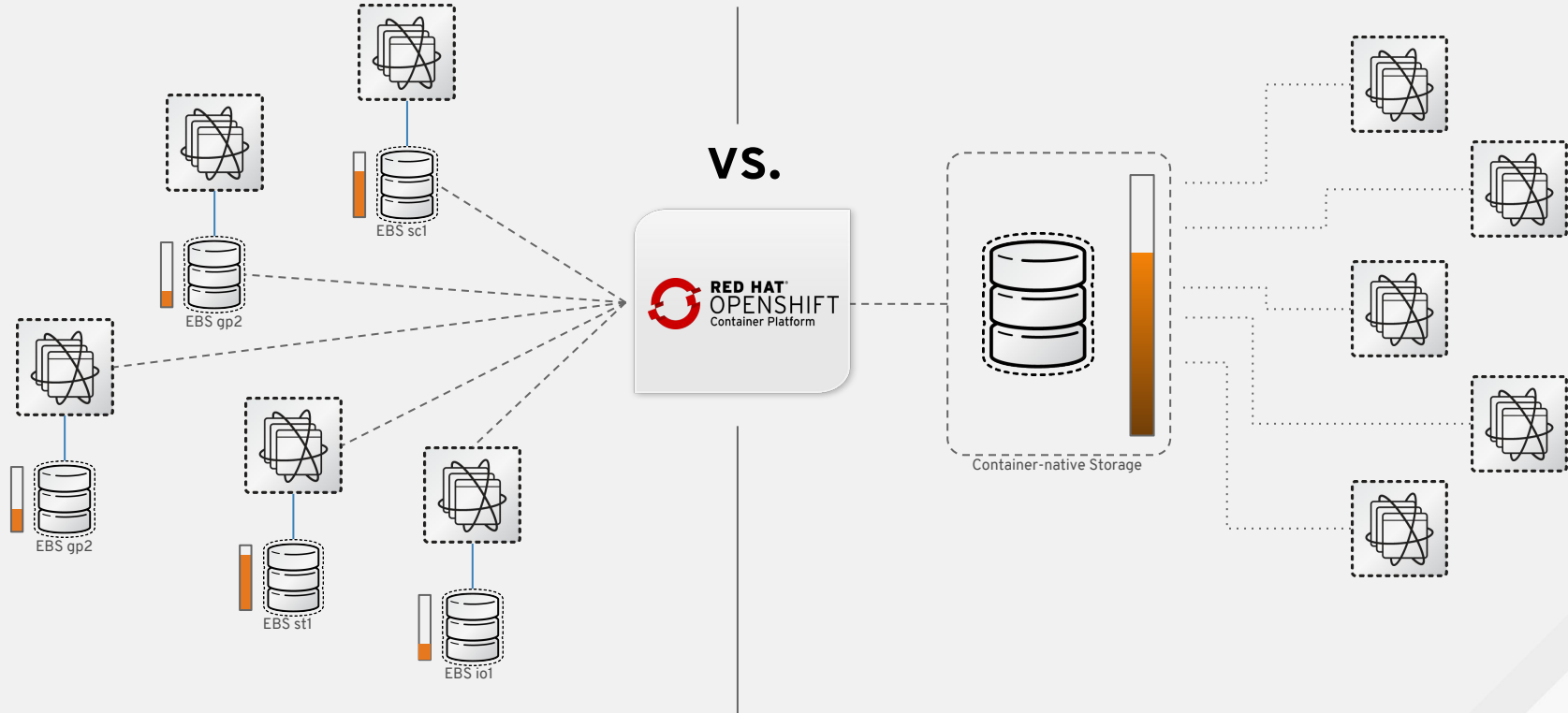


RED HAT®  
VIRTUALIZATION

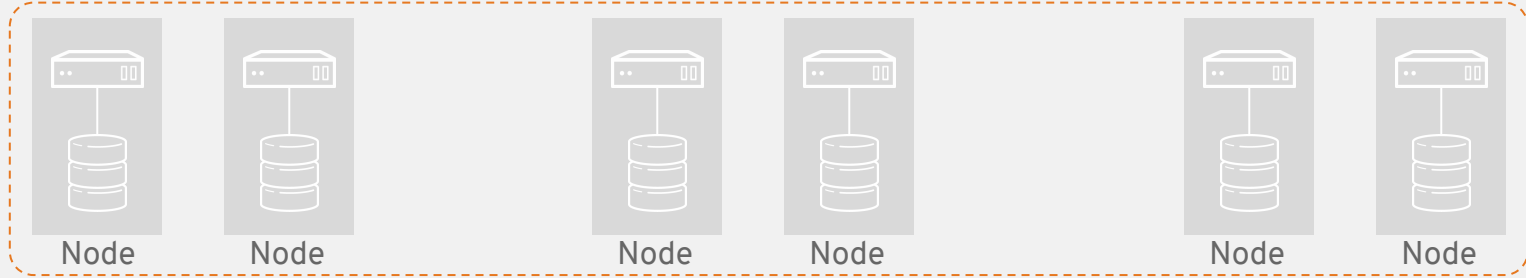
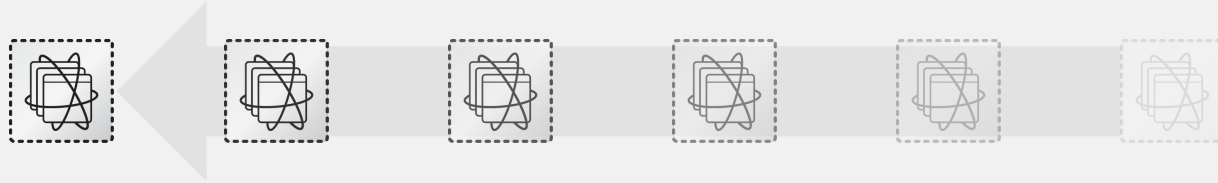
RED HAT®  
OPENSTACK®  
PLATFORM



# Storage Consolidation



# Simplify Container Availability



AVAILABILITY  
ZONE A



AVAILABILITY  
ZONE B



AVAILABILITY  
ZONE C

# Demonstration

# OpenShift + Storage - How it all works!

# So... what is CNS?

- CNS: providing dynamic persistent storage for openshift with Gluster in a **hyper-converged** fashion
- Heketi: the high-level service interface to gluster to manage the lifecycle of volumes in one or more Gluster clusters.

Openshift  $\Leftrightarrow$  Heketi  $\Leftrightarrow$  Gluster

# Components of CNS

- In openshift/kubernetes:
  - dynamic glusterfs provisioner
  - glusterfs plugin
- Heketi:
  - high-level service interface for gluster volume lifecycle management
  - Running as a container in openshift
- glusterfs:
  - one or more glusterfs clusters
  - running as containers in openshift
- cns-deploy:
  - tool to deploy gluster and heketi into an existing openshift cluster



# Open Source! ⇒ Upstream Bits

- <https://github.com/gluster/gluster-kubernetes>
- <https://github.com/heketi/heketi>
- <https://github.com/kubernetes/kubernetes> : pkg/volume/glusterfs/
- <https://github.com/gluster/gluster>

# Glossary: OpenShift/Kube Storage

- **pod**: group of one or more containers that form an entity
- **persistent volume (PV)**: to be mounted by application pod
- **provisioner**: to provide PVs upon request
- **plugin**: mechanism to mount the PV, referenced in PV
- **persistent volume claim (PVC)**: mechanism for a user to request a PV
- Access types for volumes:
  - RWO - read write once (single node)
  - RWX - read write many (multiple nodes)
  - ROX - read only many (multiple nodes)
- flavors of provisioning: dynamic and static

# Dynamic Provisioning (OCP 3.4)



RED HAT  
GLUSTER STORAGE

**1** Submit Persistent Volume Claim

**2** OpenShift requests volume to be created

**4** OpenShift binds persistent volume to persistent volume claim request



**3** Persistent volume is created by storage system and registered with OpenShift



RED HAT  
GLUSTER STORAGE

# PV creation: glusterfs dynamic provisioner

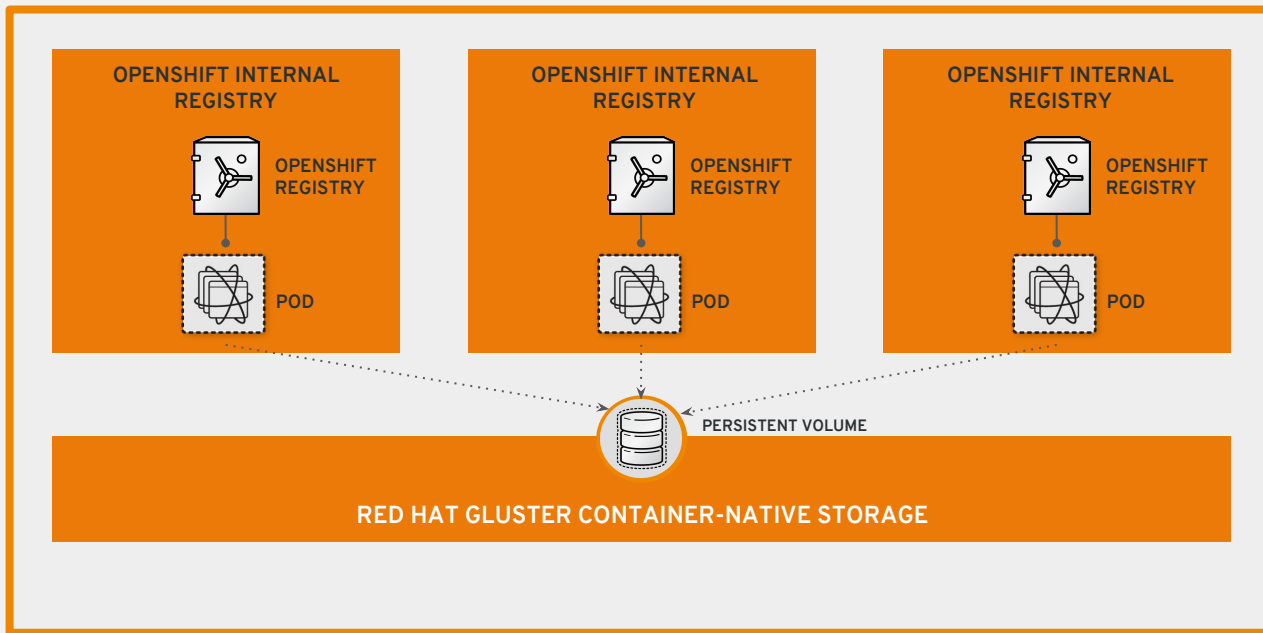
- PVC (created by user) references the glusterfs provisioner
  - glusterfs provisioner extracts details from PVC
  - provisioner tells heketi to create a volume of given size and type
    - heketi looks for a gluster cluster that can satisfy this request
    - if found, heketi tells the gluster instance to create the volume
      - gluster creates a volume
    - Heketi hands volume back to provisioner
  - provisioner creates PV and puts the gluster volume details into it
  - provisioner puts glusterfs as the mount plugin into the PV
  - Provisioner returns PV to the caller
- PVC is bound to the PV and can later be used in a pod by the user

# What's shipping Today

# Container Storage Use Cases

Persistent Storage for Apps  
Storage for OpenShift Infra (Registry)

# OpenShift Registry on CNS



Scalable



Highly-Available



Automated



Integrated

# CNS 3.4 (Jan 2017)

- Set it all up in a single command (cns-deploy)
- Dynamic provisioning
- Deploy Gluster cluster as a DaemonSet
- Basic scalability
  - Serving 100+ volumes from 3-node cluster



# CNS 3.5 (April 2017)

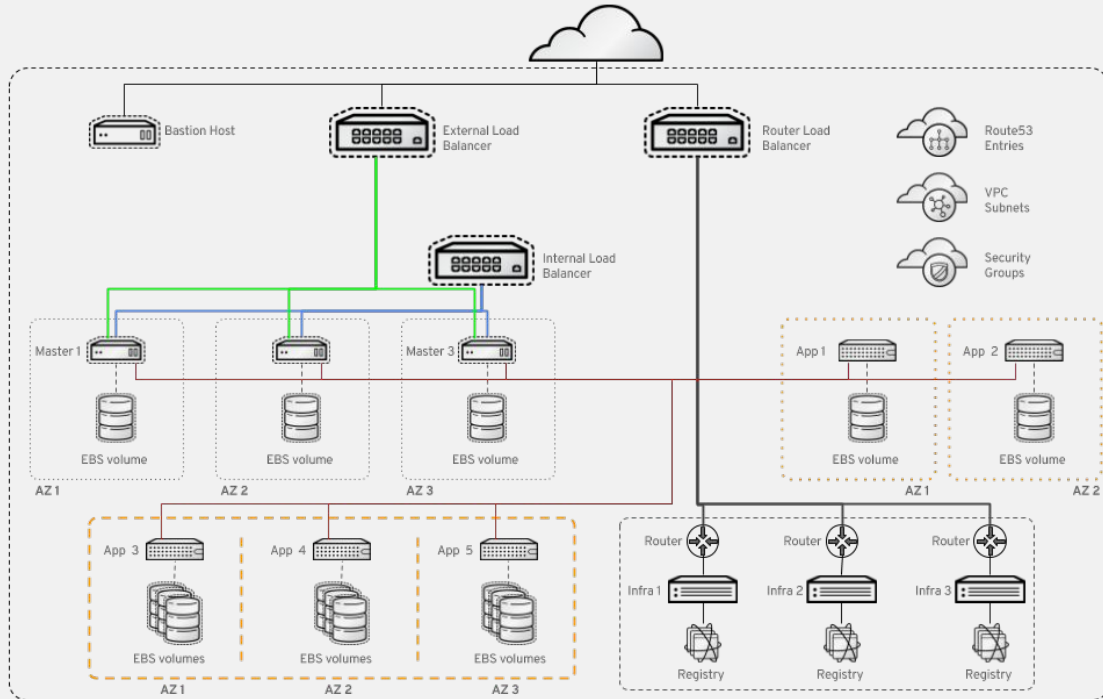
- CNS volumes can now back the OpenShift Registry
- Day-2 maintenance operations: Remove/replace disk
- Enabled policy based workload provisioning
- Basic support for snapshots and replication
- Greatly enhanced scalability (in number of volumes)
  - Can serve 300+ volumes from 3-node cluster instead of 100

# Reference Architecture & Demo

# Reference Architecture Description

OpenShift with Container Native Storage on Amazon Web Services

Reference Architecture: <https://access.redhat.com/articles/3018151>



# Demonstration II

**What does the future look like?**

# Key Asks from Customers

Larger RHGS volume density per 3-node cluster

Proper support for RWO workloads

Lightweight S3 access for developers

Day-2 Maintenance

# Roadmap

- Greatly improved scalability (1000+ ?)
- Proper RWO-support with gluster-block
- Day-2 maintenance continued: remove node, ...
- Gluster as registry backend native choice
- Lightweight S3 support for applications
- CNS deployment integrated with openshift-ansible

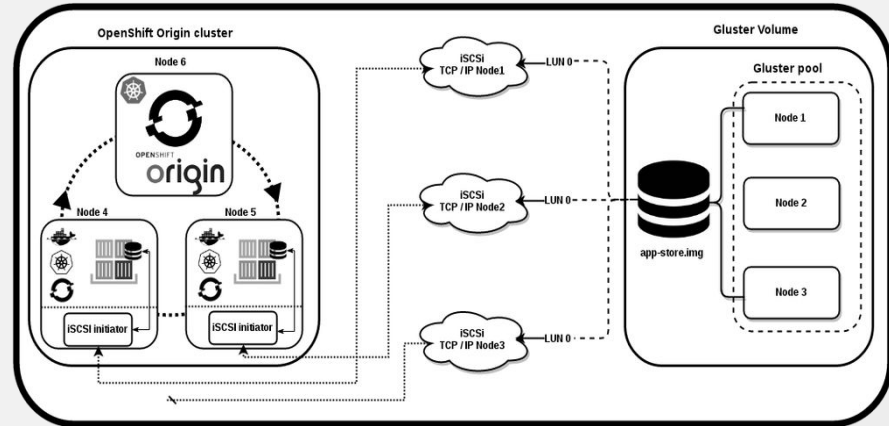
# Gluster-Scaling: Brick Multiplexing

- Why?
  - Radically lower CPU, memory and port consumption/usage
  - Order of magnitude increase in the number of volumes per cluster especially for OpenShift storage
- What it is?
  - Move from 1 brick per process to multiple bricks per process
  - Many bricks can consume **one** port, **one** set of global data structures, and **one** pool of global threads
  - Configurable number of bricks per Server Process
  - Provides more flexibility for I/O scheduling across volumes
- When?
  - Summer 2017 for container-storage use-cases



# Better RWO support for OpenShift

- Why
  - Current Gluster-file volumes do not meet performance requirements of some RWO workloads
- What
  - iSCSI access to volumes with tcmu-runner and libgfiapi
  - Orchestration of block devices managed by gluster-block
  - New gluster-block provisioner for kubernetes
- When
  - Summer CY 2017



# S3 support for OpenShift

- Why?
  - S3 support for building cloud-native apps
- What it is?
  - Gluster-s3 container provides a S3 interface to a gluster volume
  - Embedded as a micro-service inside OpenShift
  - Targeted for binary build archival, backups, and registry storage
- When?
  - 2-H CY 2017

RED HAT  
**SUMMIT**

# THANK YOU



[plus.google.com/+RedHat](https://plus.google.com/+RedHat)



[facebook.com/redhatinc](https://facebook.com/redhatinc)



[linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)



[twitter.com/RedHatNews](https://twitter.com/RedHatNews)



[youtube.com/user/RedHatVideos](https://youtube.com/user/RedHatVideos)

The logo consists of a red speech bubble shape pointing downwards, containing the text "RED HAT" in a smaller font above "SUMMIT" in a larger font, both in white.

**RED HAT**  
**SUMMIT**

**LEARN. NETWORK.  
EXPERIENCE  
OPEN SOURCE.**