Spectrum Scale
On, In, and Under the Cloud

Dean Hildebrand – Principal Research Staff Member
IBM Almaden Research Center

IBM Spectrum Scale (nee' GPFS)

- Proven, scalable, high-performance **software defined storage** solution
  - Simplified data management and automated information lifecycle tools
  - Capable of managing petabytes of data and billions of files
- 15 years in the marketplace
- Broad adoption across many industries
  - Technical computing
    - Electronic Design Automation, Life Sciences, Oil & Gas Exploration, Government, Media, Financial Services, HPC
  - Enterprise computing
    - Business Analytics, OLTP
- Over 3,000 customers with 100K+ systems world-wide
- Research driven!

Continual Drive to Avoid Islands of Storage

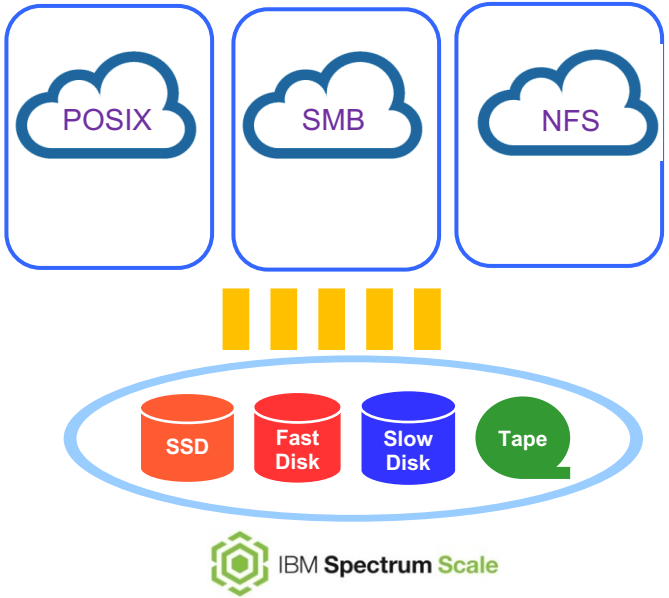
Spectrum Scale and the Cloud

On the Cloud

In the Cloud

Under the Cloud

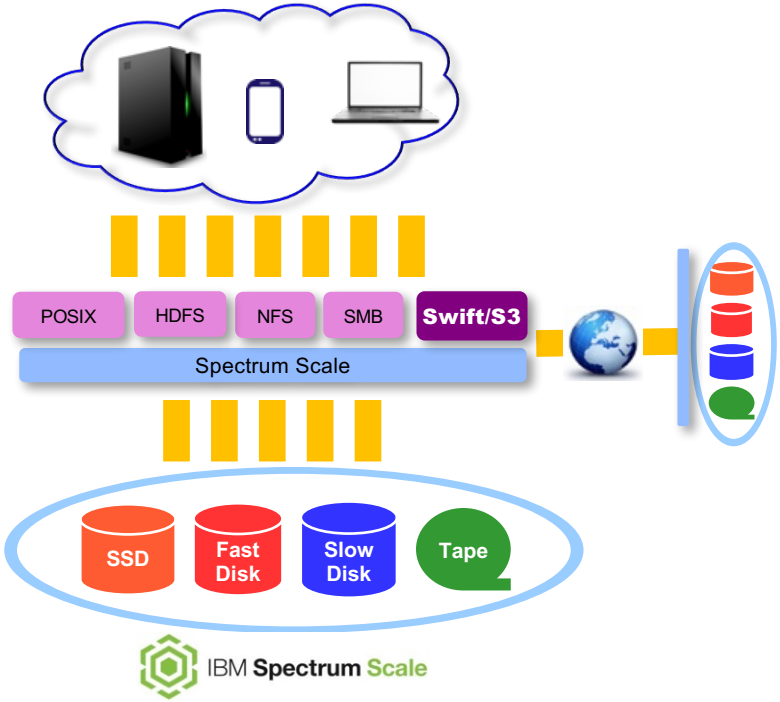
IN THE CLOUD



SUPPORTING NUMEROUS PROTOCOLS AND APIS TO ACCESS DATA

Object Storage

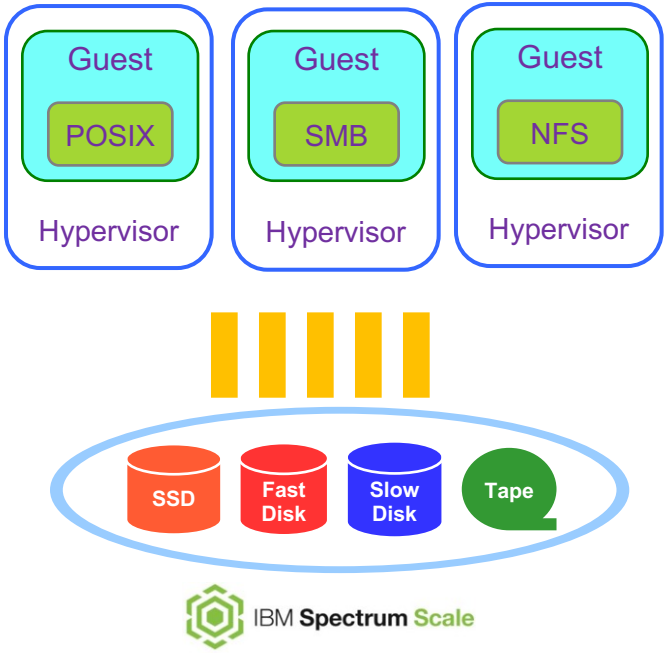
- Unified File and Object
  - POSIX, SMB, NFS, Swift, S3
- Integrated analytics
  - HDFS
- Secure and web-friendly for sharing data
- Distribute data across multiple sites
- Fast metadata indexing and search capabilities



Access Data using File Protocols

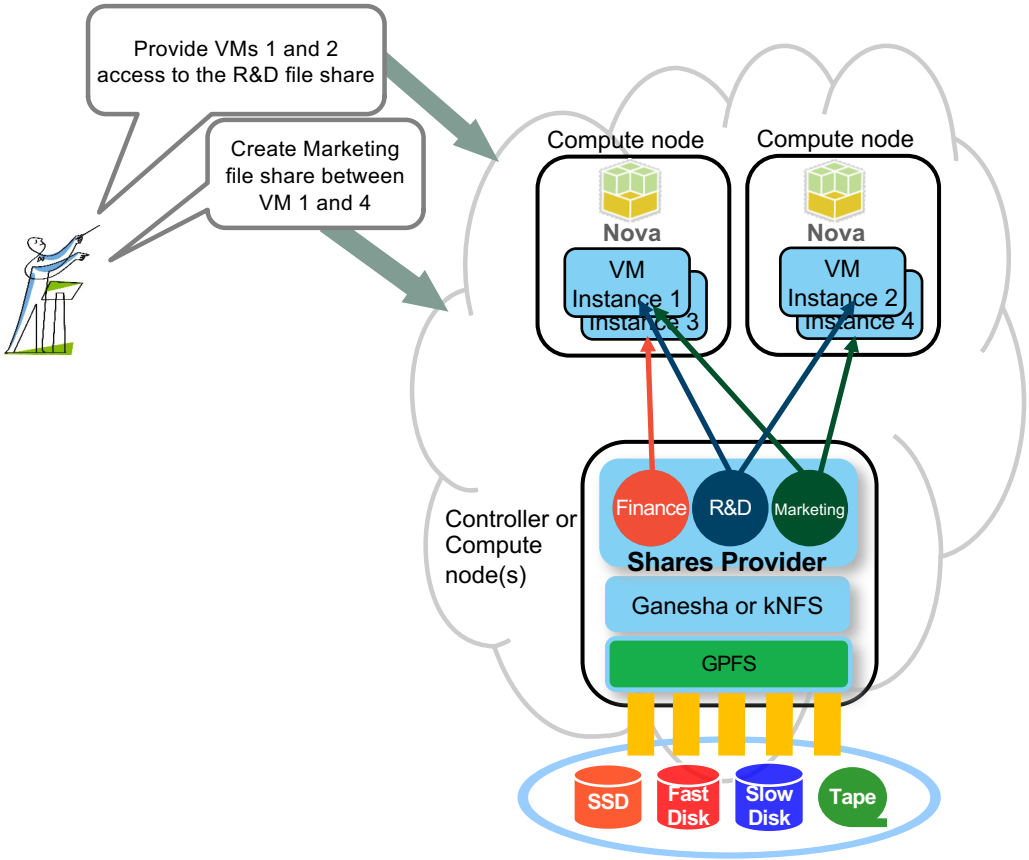
Same great data access…

…now from the Cloud



OpenStack Manila

- Provides shared file system services for VMs
- Vendor neutral API
- Supports cNFS and CES NFS
- Supported Operations by driver:
  - ✓ Create/Delete/List shares
  - ✓ Allow/Deny access to shares (Create NFS exports and allow/deny access)
  - ✓ List share access rules
  - ✓ Create share from snapshot
  - ✓ Extend share
  - ✓ Create/Delete/List share snapshots
  - ✓ Use driver to setup Compression by using share type
  - ✓ Use driver to Manage, un-manage shares (including existing filesets)



Containers

Standard way to package an application and all its dependencies

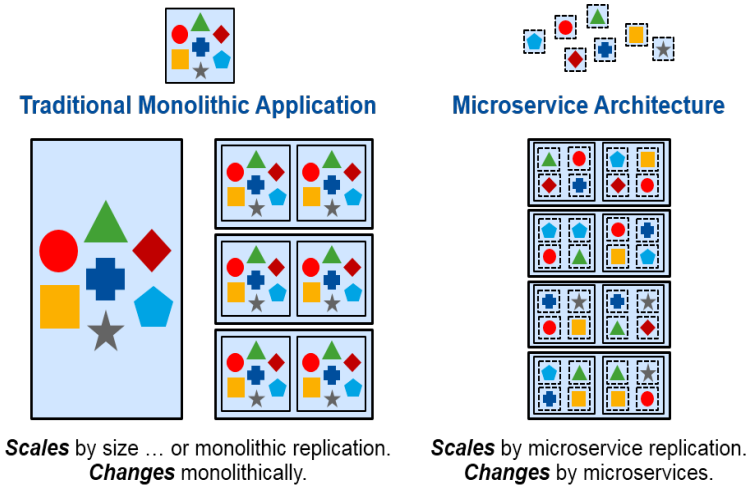
Portable between environments and run without changes

Isolate unique elements to enable a standardize infrastructure

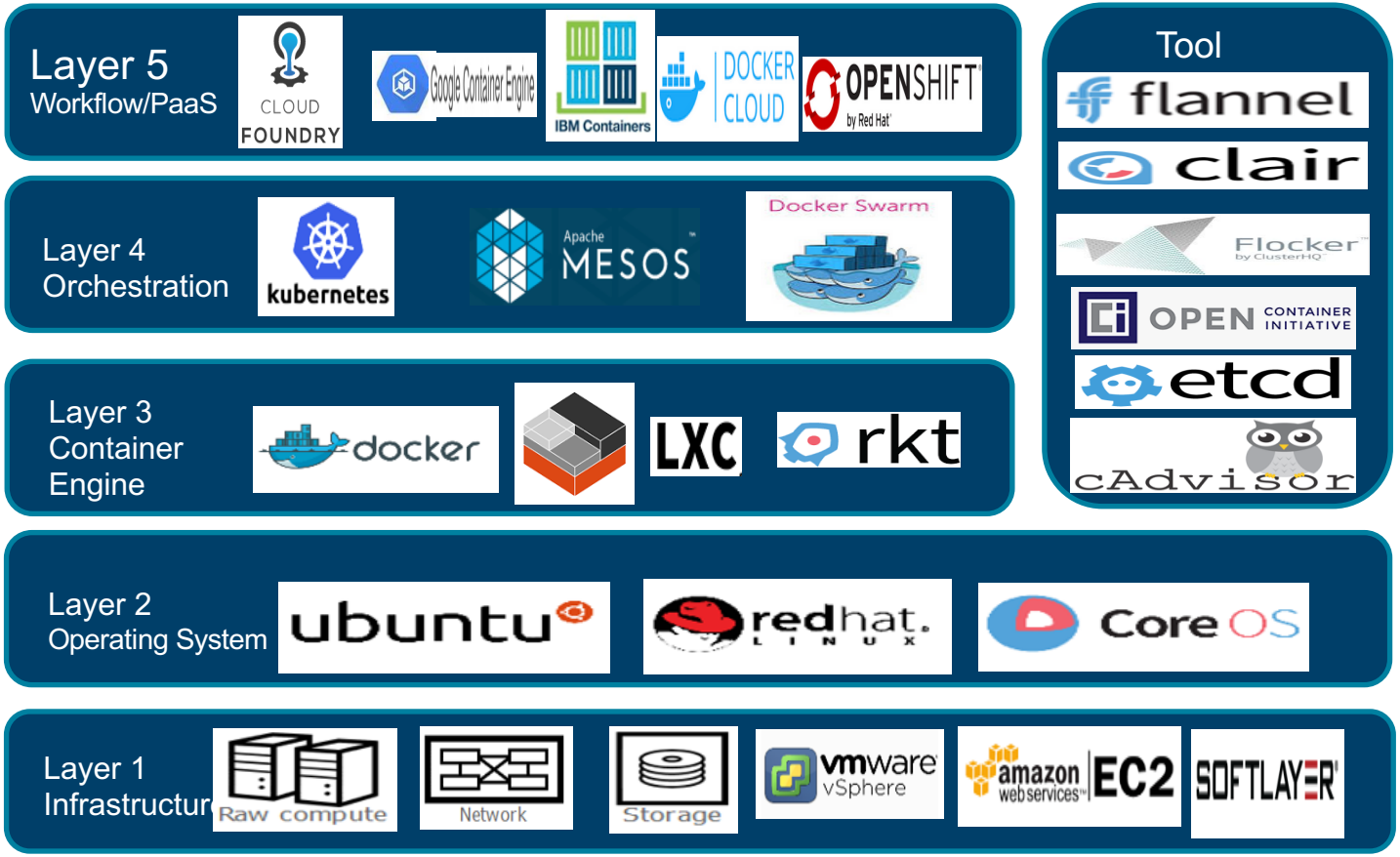
Fast and lightweight

Containers, Containers, Containers

- HPC and Scientific Computing
  - Portable and reproducible science
- On-premise Clouds
- DevOps and continuous integration
- Simplify and speed application development
- Load balancing across cloud VM services (e.g., EC2)

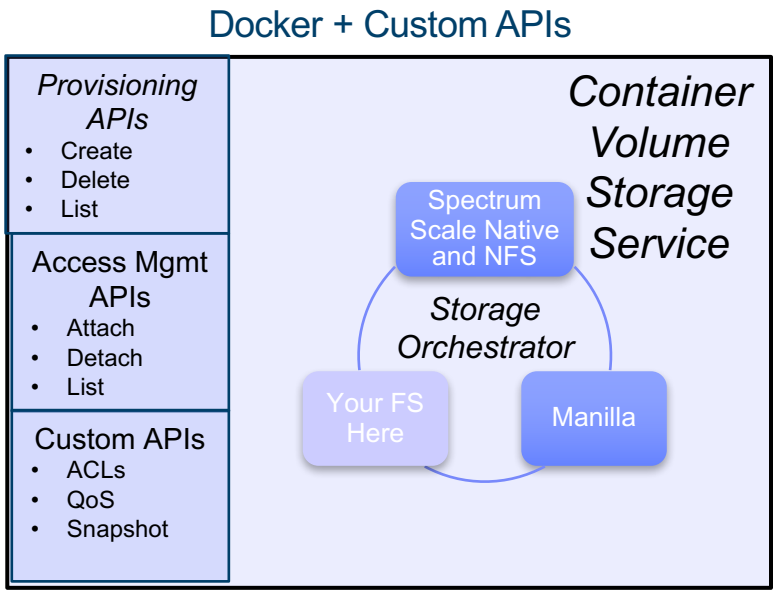


Container EcoSystem

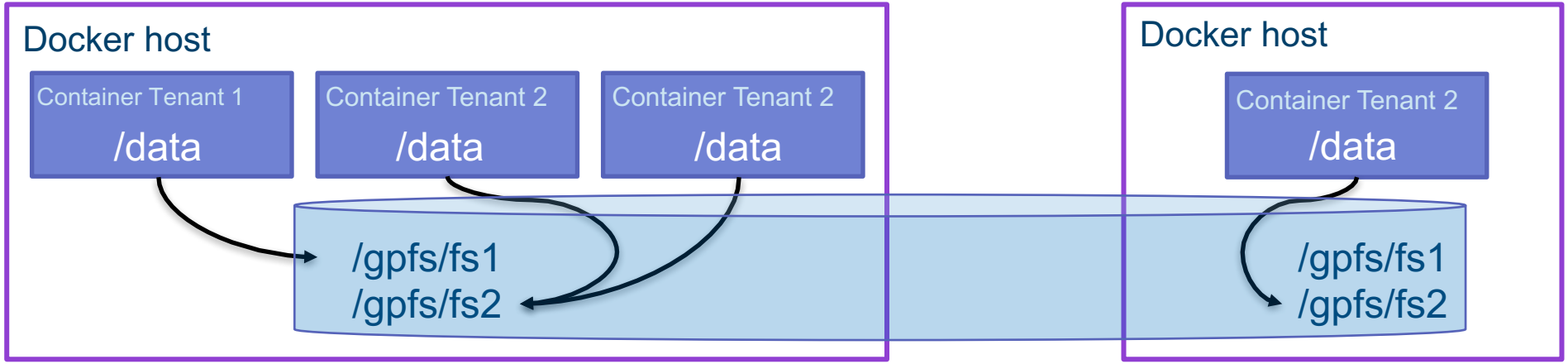


Container Volume Service Architecture

- Storage volume drivers abstract storage complexities while exposing key capabilities
- Each container framework has different mechanisms for provisioning storage
  - Build a single driver that can 'link' to each framework allowing access to single dataset across all frameworks
- General architecture to support…
  - A variety of storage services
    - Initially supporting shared file services
    - Could expand to support shared block storage as well
  - A variety of container frameworks
- Support Docker/Kubernetes REST-APIs
  - Plus possible additional value add APIs
- Support existing or new volumes
- All open-source



Spectrum Scale and Docker



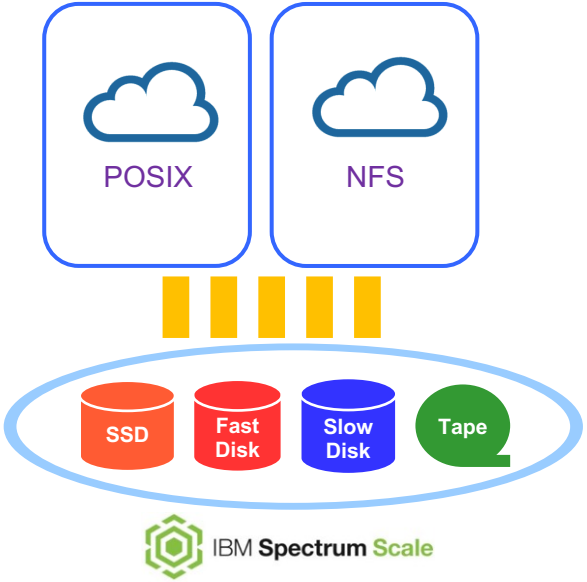
Details

- Single or separate userid namespaces between containers and hosts
- Data sharing across containers and hosts
- All POSIX commands supported from container

A Few Benefits

- Multi-tenant access
  - Container can only access its volumes
  - Allow root access in container without allowing root access to file system
  - ACLs can add an extra level of security
- Native client performance

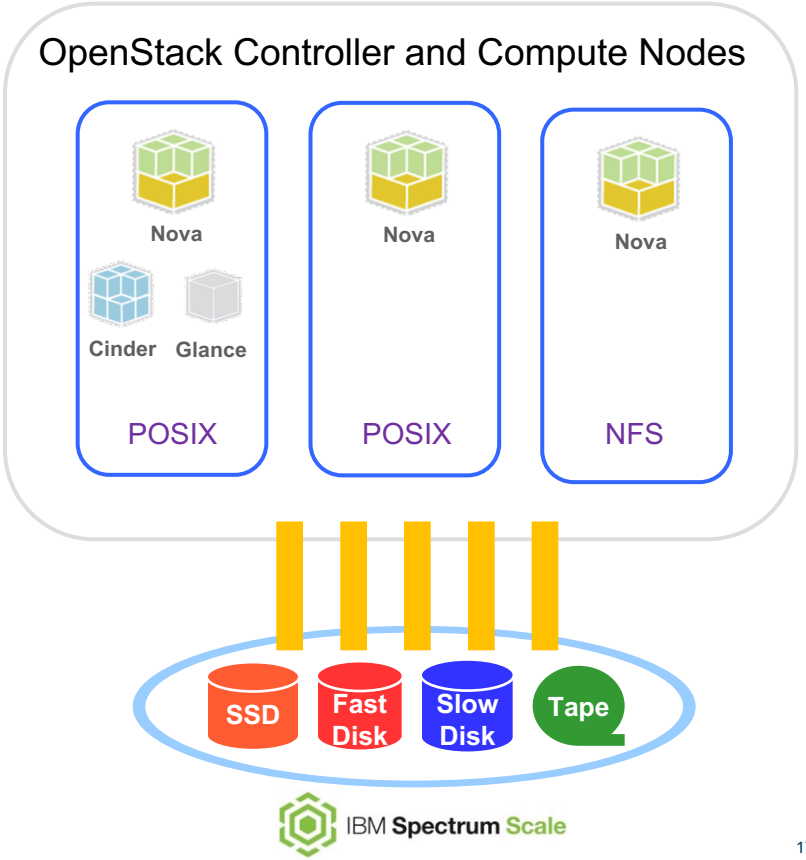
UNDER THE CLOUD



SUPPORTING CLOUD TECHNOLOGIES

OpenStack Cinder: virtual block storage

- Provision and launch VMs across cluster
- Simply files in the unified namespace
- Localizes & optimizes all Volume operations
- Optionally formats and labels Volumes
- VM instantiation using Copy-on-Write
  - Integration with OpenStack Glance
- Enables advanced Volume provisioning
- Supports high-performing VM live migration
- Supports POSIX/NFS based deployments
- Supports all enterprise Spectrum Scale features



Future – Running Docker Container Images

- Container images are created out of immutable layers
  - All changes saved using copy-on-write
- Containers images currently run on local server
  - No shared access
  - Images must be copied from registry to each server
- As always, local access has scalability drawbacks
- Benefits of shared storage
  - No need to copy data and manage local disk space
  - Instantaneous container launch directly out of registry
  - Single copy of images across entire cluster
    - No need to re-download image for every node
- Still investigating
  - optimizing copy-on-write
  - performance

ON THE CLOUD

SUPPORTING VIRTUAL CLOUD INFRASTRUCTURES

SOFTLAYER® an IBM Company

amazon web services

Microsoft Azure

Google Cloud Platform

ACCEPTING LOSS OF CONTROL OVER HARDWARE
(BUT IT WAS ALREADY LOST ANYWAY)

Storage is now squares and boxes

ACCEPTING LOSS OF TIME

DEPLOYMENT TIME < EXECUTION TIME

For Example

How long should it take to deploy a 1000+ node cluster with 10PB that will run for 1 hour?