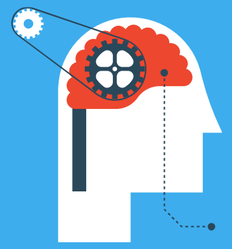


Spectrum Scale Expert Talks

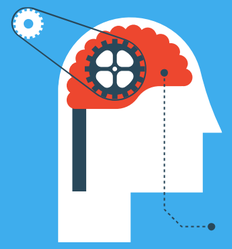
Episode 8:

Multi-node scaling of AI workloads using NVIDIA DGX, OpenShift and Spectrum Scale

IBM Spectrum Scale

Show notes:
www.spectrumscaleug.org/experttalks

Join our conversation:
www.spectrumscaleug.org/join



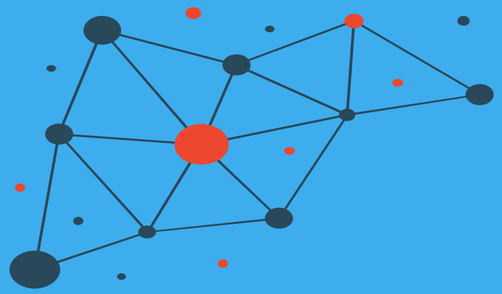
SSUG::Digital

Welcome to digital events!

IBM Spectrum Scale

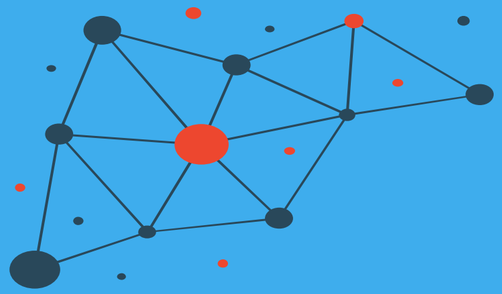
Show notes:
www.spectrumscaleug.org/experttalks

Join our conversation:
www.spectrumscaleug.org/join



About the user group

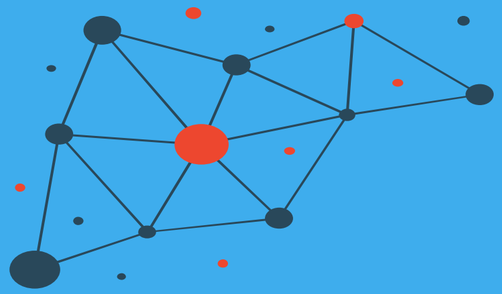
- Independent, work with IBM to develop events
- Not a replacement for PMR!
- Email and Slack community
- https://www.spectrumscaleug.org/join



We are …

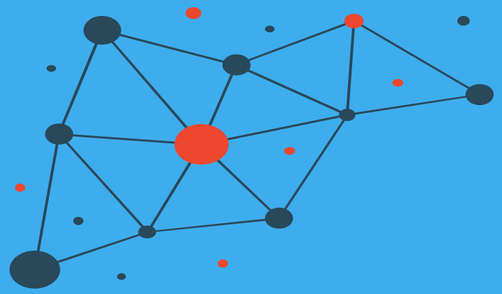
- Simon Thompson (UK)
- Kristy Kallback-Rose (USA)
- Bob Oesterlin (USA)
- Bill Anderson (USA)
- Chris Schipalius (Australia)

IBM CHAMPION



Check https://www.spectrumscaleug.org/experttalks for charts, show notes and upcoming talks

- Past talks:
  - 001: What is new in Spectrum Scale 5.0.5?
  - 002: Best practices for building a stretched cluster
  - 003: Strategy update
  - 004: Update on performance enhancements in Spectrum Scale (file create, MMAP, direct IO, ESS 5000)
  - 005: Update on functional enhancements in Spectrum Scale (inode management, vCPU scaling, NUMA considerations)
  - 006: Persistent Storage for Kubernetes and OpenShift environments
  - 007: Manage the lifecycle of your files using the policy engine
- Today:
  - Nov 4: Multi-node scaling of AI workloads using NVIDIA DGX, OpenShift and Spectrum Scale
- Next:
  - Nov 16: User Meeting at SC20 (Session 1) – Storage for AI https://www.spectrumscaleug.org/event/sc20-meeting-session-1-storage-for-ai/
  - Nov 18: User Meeting at SC20 (Session 2) – What is new in Spectrum Scale 5.1? https://www.spectrumscaleug.org/event/sc20-meeting-session-2-what-is-new-in-spectrum-scale-5-1/



Speakers

- Thomas Schoenemeyer (NVIDIA)
- Gero Schmidt (IBM)
- Simon Lorenz (IBM)

Disclaimer

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

IBM reserves the right to change product specifications and offerings at any time without notice. This publication could include technical inaccuracies or typographical errors. References herein to IBM products and services do not imply that IBM intends to make them available in all countries.

Multi-node scaling of AI workloads using NVIDIA DGX, OpenShift and Spectrum Scale

Thomas Schoenemeyer
NVIDIA Senior Solution Architect

Gero Schmidt
IBM Spectrum Scale BDA Software Engineer

Simon Lorenz
IBM Spectrum Scale BDA Architect

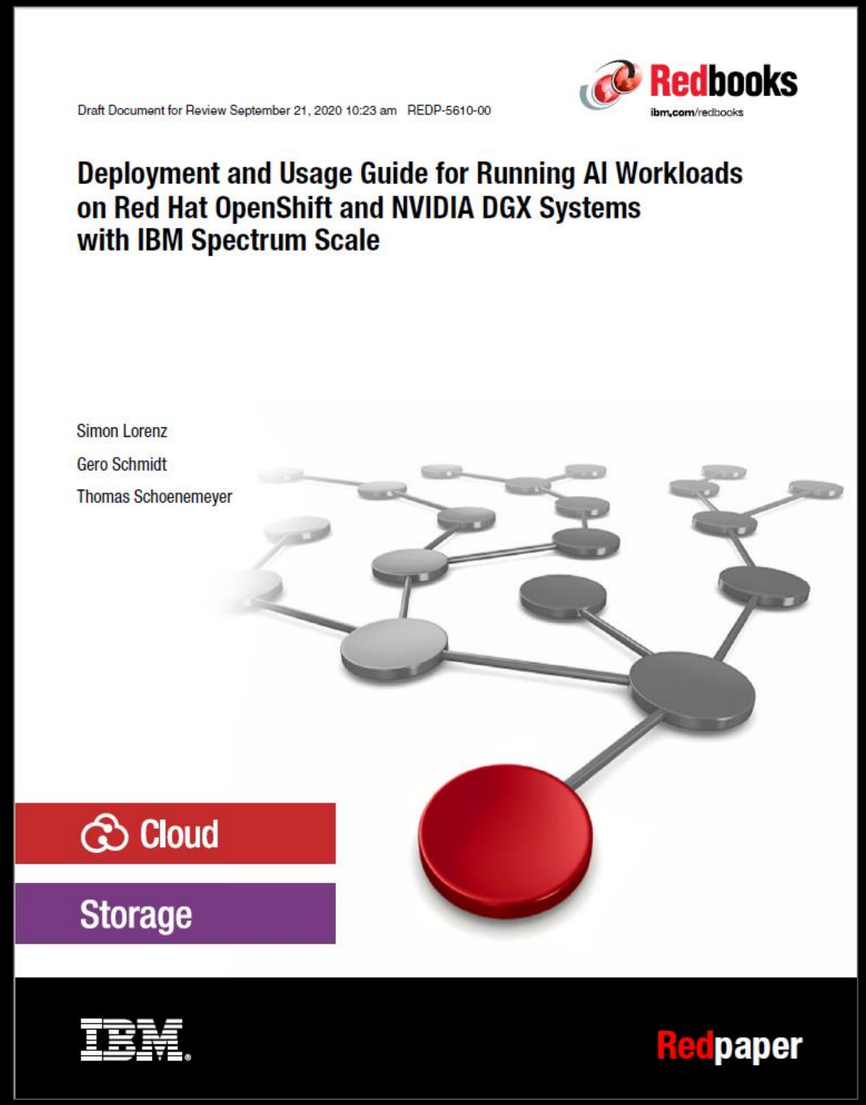
IBM

IBM Redpaper published on the topic:

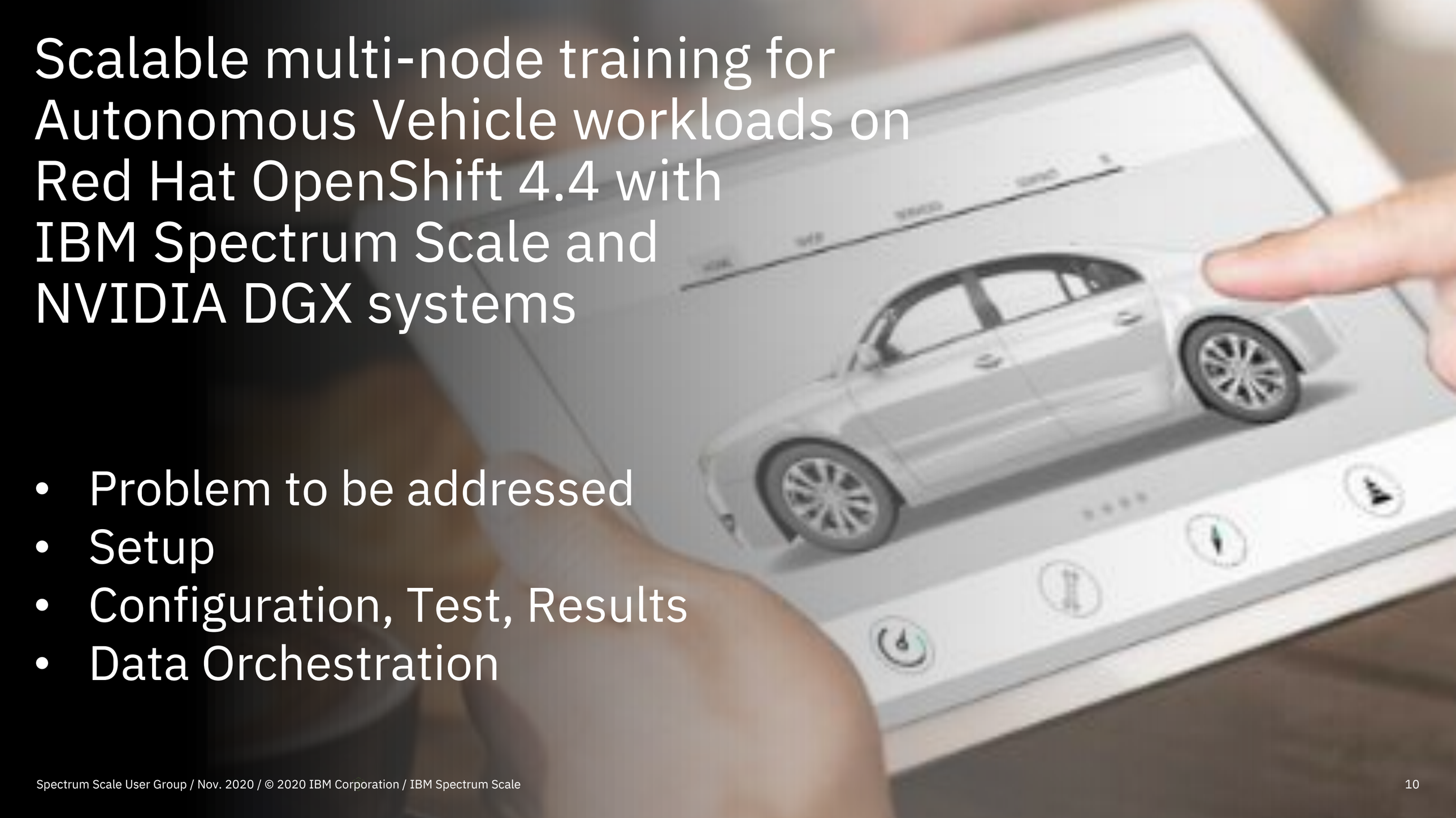
Deployment and Usage Guide for Running AI Workloads on Red Hat OpenShift and NVIDIA DGX Systems with IBM Spectrum Scale

Visit:

http://www.redbooks.ibm.com/redpieces/abstracts/redp5610.html
(published September 21, 2020)



Scalable multi-node training for Autonomous Vehicle workloads on Red Hat OpenShift 4.4 with IBM Spectrum Scale and NVIDIA DGX systems



- Problem to be addressed
- Setup
- Configuration, Test, Results
- Data Orchestration

DEEP LEARNING AT SCALE FOR AV

Safer Driving begins in the DC

Many DNN models are deployed in autonomous vehicles and need to be trained

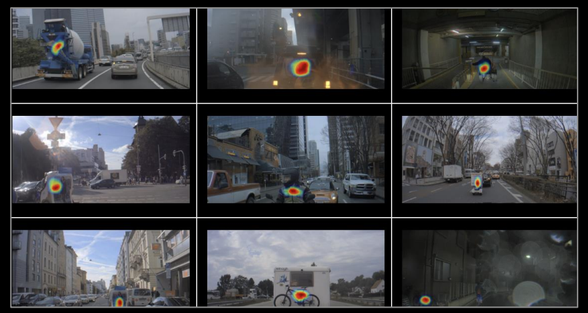
- Many scenarios: highway, urban, countryside, etc.
- Different conventions: traffic signs, road marks, etc.
- Thousands of unexpected conditions/events

Core Challenge: need **high accuracy** and **robust DNNs**

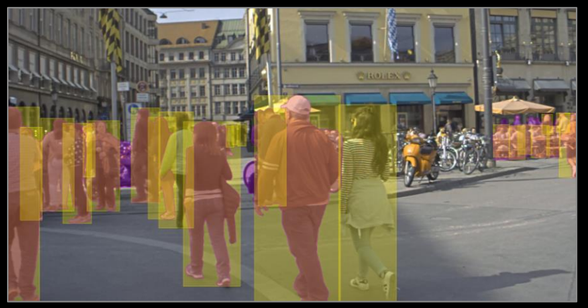
AV WORKFLOW – CHALLENGES & PAIN POINTS

50 Car Fleet, Driving 6 Hours/Day, Generates 1.6PB Each Day

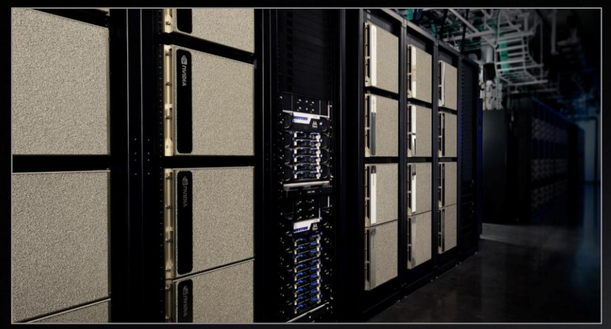
INGESTION
1.6PB/day needs to be transported, encoded, stored



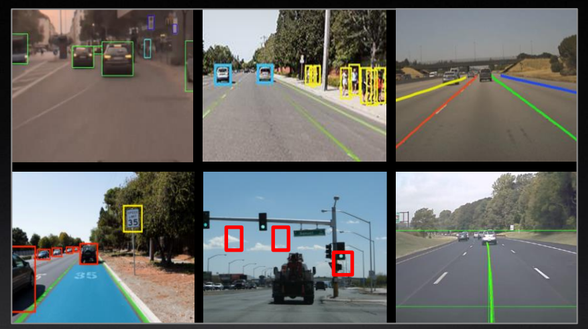
CURATION
Billions of frames.
Find the 10-14% that are useful



LABELING
Manage 1000+ workers with 50+ projects. Ensure quality every frame.



TRAINING
20+ models. 100s Engineers,
Optimize each model w/ 50+ parallel experiments.



REPLAY
Test against 10,000s hours of sensor data.
Repeat Daily

SIMULATION
Drive millions of miles.
Find the most critical scenarios to test.

DEEP LEARNING AT SCALE FOR AV
Common Approach to Improve Accuracy

- **Increase network architecture**
  - Larger networks are slower at training and inference
  - Hyperparameter Tuning
  - **May need several days on one worker**
- **Increase training data**
  - 100 cars with five 2MP cameras => 200 PB / year
  - Improvement due to "just" more data saturates fairly quickly
  - Only a tiny fraction of unlabeled data is and can be used for training
  - find the **most informative** unlabeled data

COMPUTATION SCALE REQUIRED

For AV Training at Scale

Assume you have 2.7 million labeled images

1 DGX-A100 trains 900k labeled images on 1 DNN in 1 day

(ResNet-18)

10 DNNs* required for self-driving

10 parallel experiments at all times (Hyperparameter Tuning)

→ **Work is finished within one day with 30 DGX-A100**

*10 is minimum

SOLVE LARGE SCALE DL PROBLEMS

NVIDIA GPU Worker Nodes

Algorithms for Deep Learning

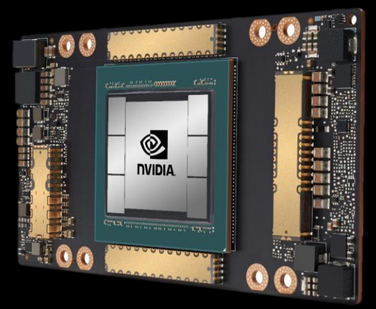
- Matrix-Matrix Operations well-suited for running on NVIDIA GPU workers
- Integrated AI Systems with DGX OS, NVLINK and NVSwitch

NVIDIA DGX-1

Large Scale Training: Remove all performance bottlenecks in a cluster

- Use Optimized Containers for GPU worker nodes → **NVIDIA NGC**
- High-Speed Networking between GPU worker nodes → **NVIDIA NCCL**
- Deploy Data Parallelism with Kubeflow/MPI/Horovod
- Seamless integration of GPU Worker nodes into K8s/OpenShift 4.x

NVIDIA DGX A100



A100 GPU

THE POWER OF SCALABILITY

MLPERF 0.7 Standard AI Benchmark – Lower is better

Benchmark	Time to resolution on single node DGX A100	Large Scale Results (DGX SuperPOD Cluster)	Number of nodes/workers (NVIDIA DGX A100)
NLP (BERT) - PyTorch NVIDIA 20.06	49.01 Min	3.36 Min	32
Reinforcement Learning (MiniGo) - TensorFlow NVIDIA 20.06	299.73 Min	29.7 Min	32
Translation (Non-recurrent) - PyTorch NVIDIA 20.06	7.84 Min	1.02 Min	20
Translation (Recurrent) GNMT - PyTorch NVIDIA 20.06	7.81 Min	0.98 Min	32
Object Detection (Heavy Weight) Mask R-CNN GNMT - PyTorch NVIDIA 20.06	82.16 Min	10.46 Min	32
Object Detection (Light Weight) SSD - PyTorch NVIDIA 20.06	10.21 Min	0.89 Min	64
Image Classification (ResNet-50 v1.5) - MXNet NVIDIA 20.06	39.78 Min	1.06 Min	96

NVIDIA ACCELERATED COMPUTING PRODUCT FAMILY

One architecture – CUDA-accelerated computing

FULLY INTEGRATED AI SYSTEMS + DGXPERTS

Deployed for this study

DGX-1

DGX-2

DGX A100

DESKTOP

TITAN/
GeForce

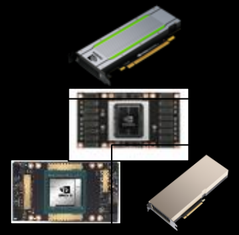
WORKSTATION

DGX Station

VIRTUAL WORKSTATION

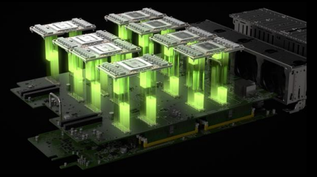
vGPU (e.g.
V100/Quadro)

DATA CENTER



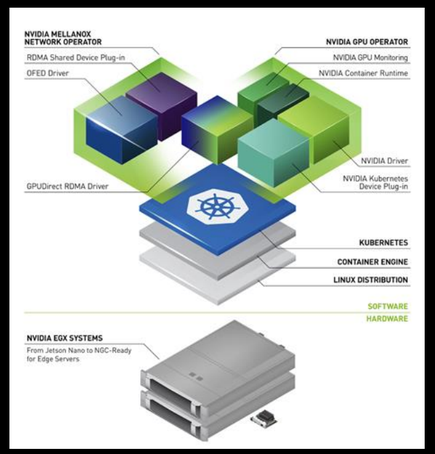
T4 / V100
A100

SERVER PLATFORM



HGX-1/ 2
HGX A100

EDGE



EGX

EGX A100

From Jetson Nano
to Edge Server

NGC

CONTAINERS

150+

TRAINED MODELS

100+

HELM CHARTS

ML, Inference

Monthly Updates

UP TO 4X MORE PERFORMANCE IN 1.5 YEARS

Full Stack Innovation Delivers Continuous Improvements

OPTIMIZED FRAMEWORKS

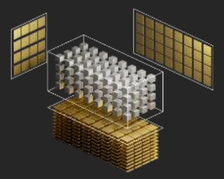
Apex, Horovod, MXNet, PyTorch, TensorFlow
Available through NVIDIA NGC

CUDA-X LIBRARIES

CUDA & Core Libraries: cuBLAS, cuDNN, CUDA graphs, DALI, NCCL, SHARP

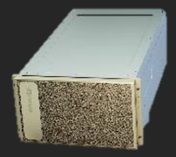
HARDWARE INNOVATIONS

GPU



Tensor Core

NVLink / NVSwitch



DGX

CX-6, HDR IB

DGX SuperPOD

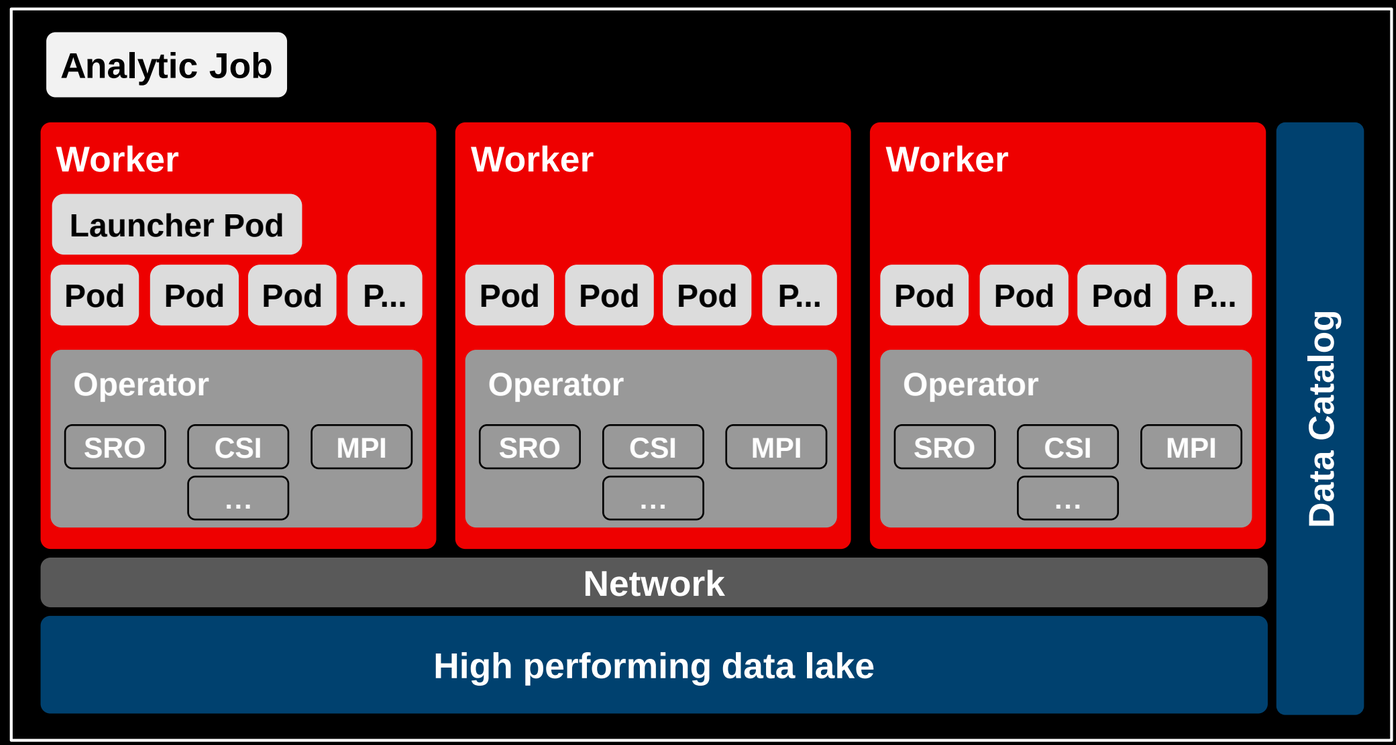
Use NVIDIA and NGC in OpenShift

Scaling large deep learning workloads is challenging

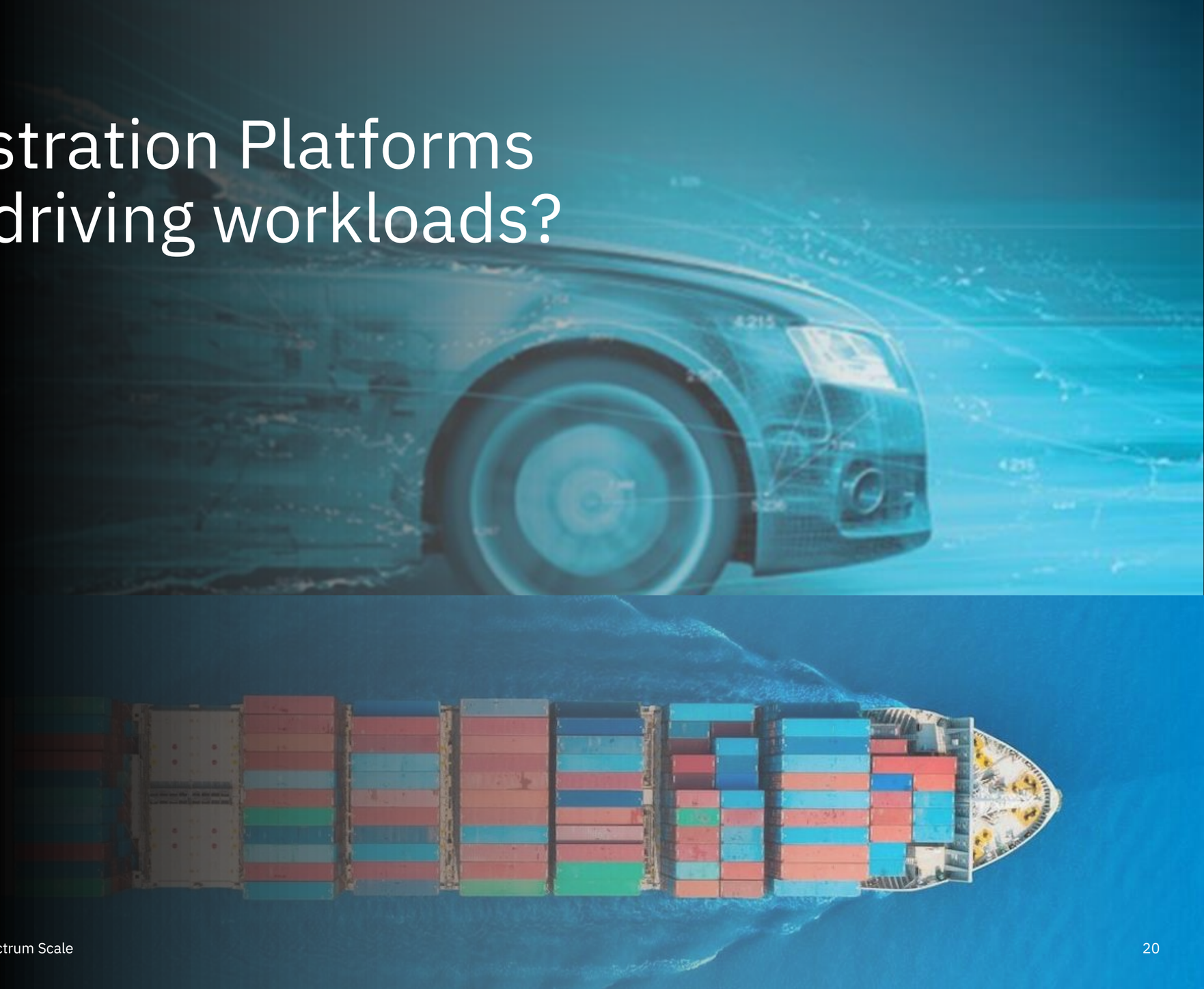
- Horizontal scaling: ensure optimal resource utilization across
  - Nodes, CPU, GPU, Memory, …

- High performing data lake
  - Global namespace
  - Performance
  - No silos
  - No data copies
  - Secure

- Know your data



Why use Container Orchestration Platforms for autonomous driving workloads?

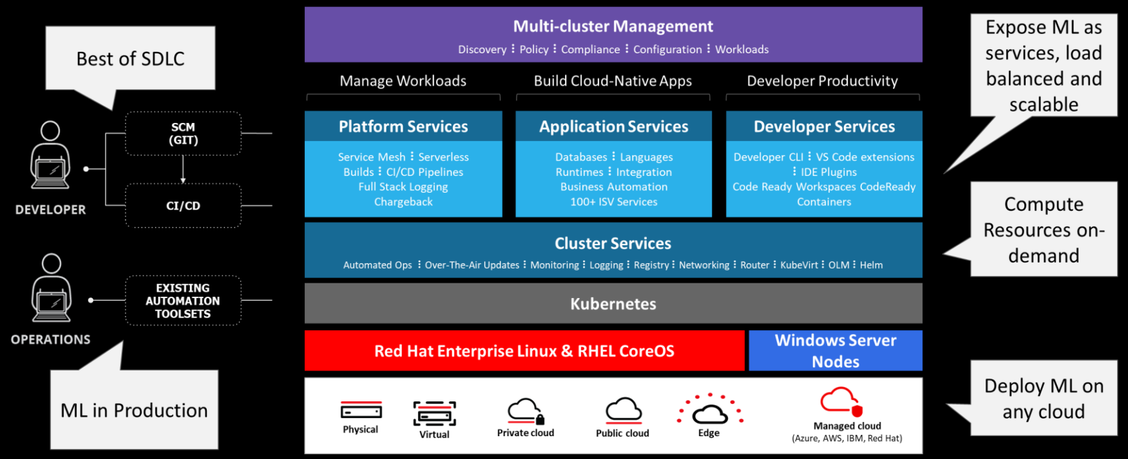


Automates application

- Deployment
- Management
- Scaling
- Networking
- Security

Components used for the Proof of Concept

Why Red Hat OpenShift?



Partners In Delivering AI-Powered Transformation

NVIDIA	IBM Storage
• #1 in GPUs, #1 in AI Supercomputing • AI solution orientation • Expertise with ML/DL developers and data scientists	• #1 in SDS with storage offerings serving end to end data pipelines • IBM Spectrum Scale file storage proven for its performance and scalability with top Supercomputers
AI Server and Storage leadership joining hands	

IBM Spectrum Scale

- Performance: remove data-related bottlenecks
- Ease of management: enable global collaboration
- Economics: optimize cost and performance
- Robust: data availability, integrity and security
- Containerized: Easier access to Kubernetes data

Storage for the world's smartest supercomputers

Summit System

- 4608 nodes each
- 200 petaflops peak performance for modeling and simulation
- 3.3 ExaOps peak performance for data analytics and AI

2.5 TB/sec
throughput to storage architecture

500 PB
HDD storage capacity

NVMe Flash for AI and Big Data Workloads

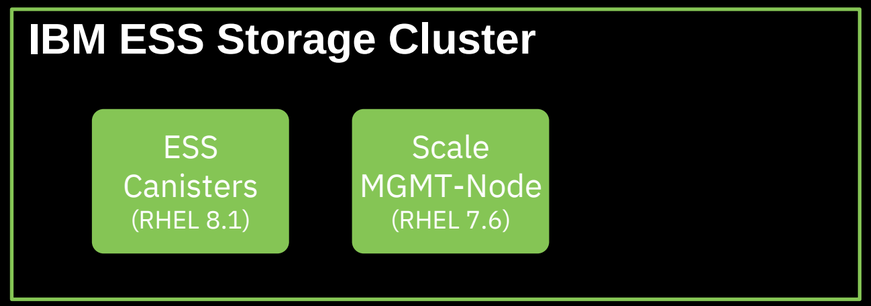
IBM Elastic Storage System 3000

All-new storage solution

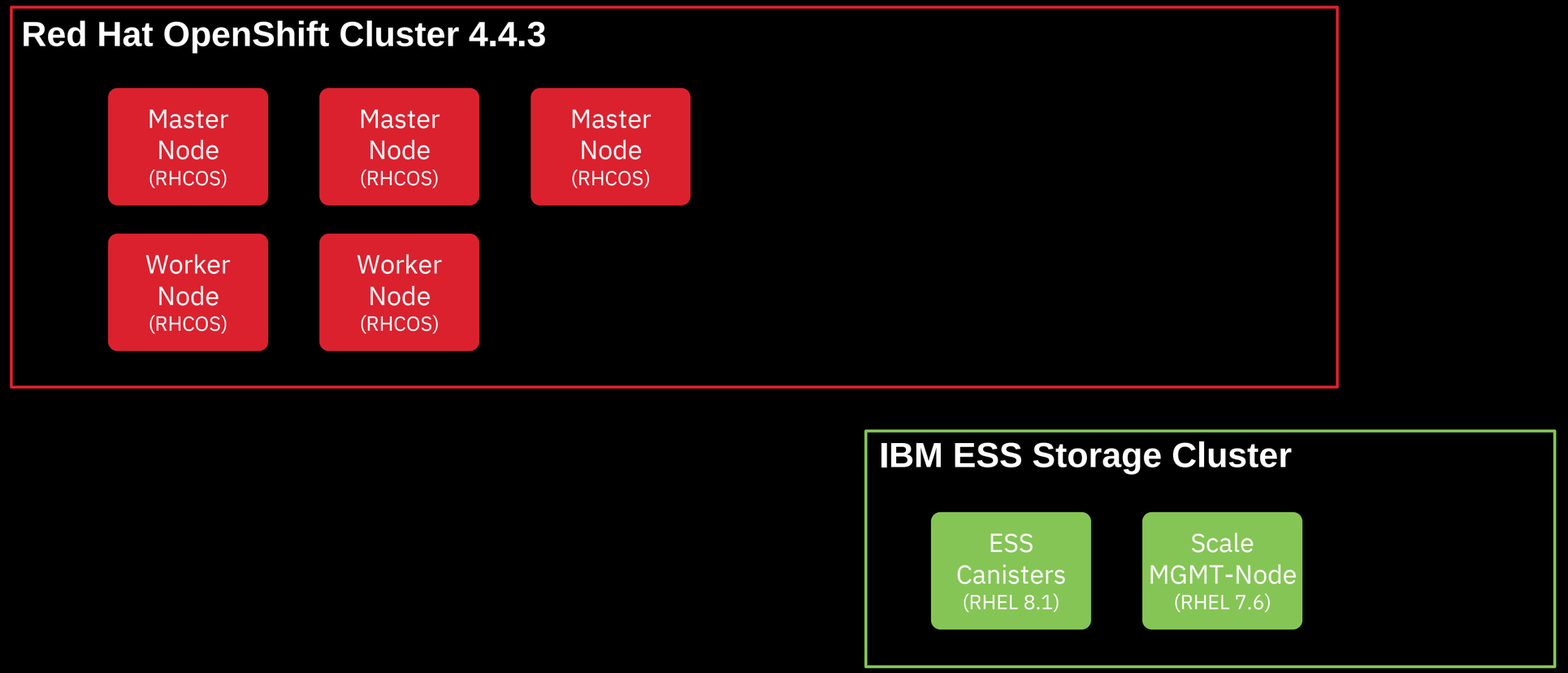
- Leverages proven FS9100 technology
- Integrated scale-out advanced data management with end-to-end NVMe storage
- Containerized software for ease of install and update
- Fast initial configuration, update and scale-out expansion
- Performance, capacity, and ease of integration for AI and Big Data workflows

IBM **Spectrum Scale**

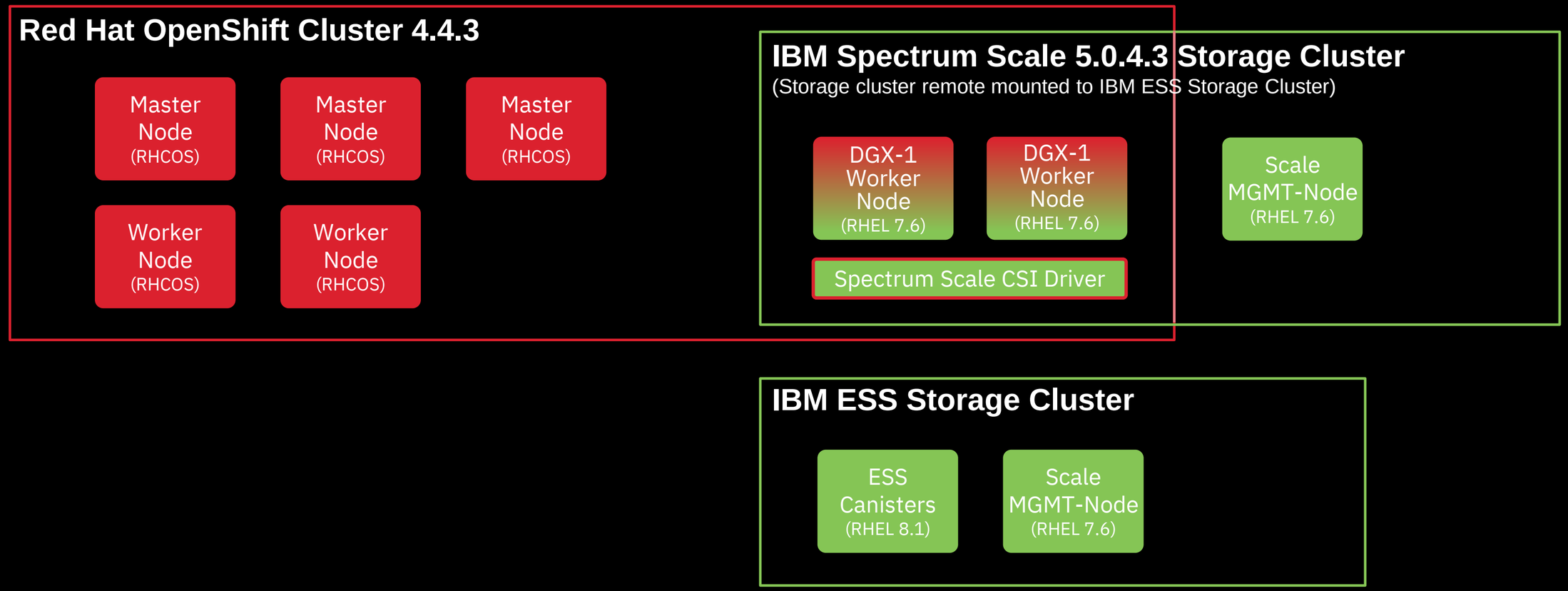
Red Hat OpenShift & IBM Spectrum Scale & CSI



Red Hat OpenShift & IBM Spectrum Scale & CSI



Red Hat OpenShift & IBM Spectrum Scale & CSI



Red Hat OpenShift & IBM Spectrum Scale & CSI

Red Hat OpenShift Cluster 4.4.3

3 Master Nodes
RHCOS

2 Worker Nodes
VM's RHCOS

IBM Spectrum Scale 5.0.4.3 Storage Cluster
(Storage cluster remote mounted to IBM ESS Storage Cluster)

DGX-1
Worker 1

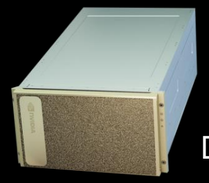
DGX-1
Worker 2

VM

- RH 7.6
  Scale GUI, Quorum, Manager

IBM ESS Storage Cluster

Node, Scale GUI, Quorum
ESS3000



DGX A100 in process of being tested and validated by NVIDIA and IBM

Infiniband Network

Red Hat OpenShift Cluster 4.4.3

3 Master Nodes
RHCOS

2 Worker Nodes
VM's RHCOS

IBM Spectrum Scale 5.0.4.3 Storage Cluster
(Storage cluster remote mounted to IBM ESS Storage Cluster)

DGX-1
Worker 1

DGX-1
Worker 2

VM

- RH 7.6
  Scale GUI, Quorum, Manager

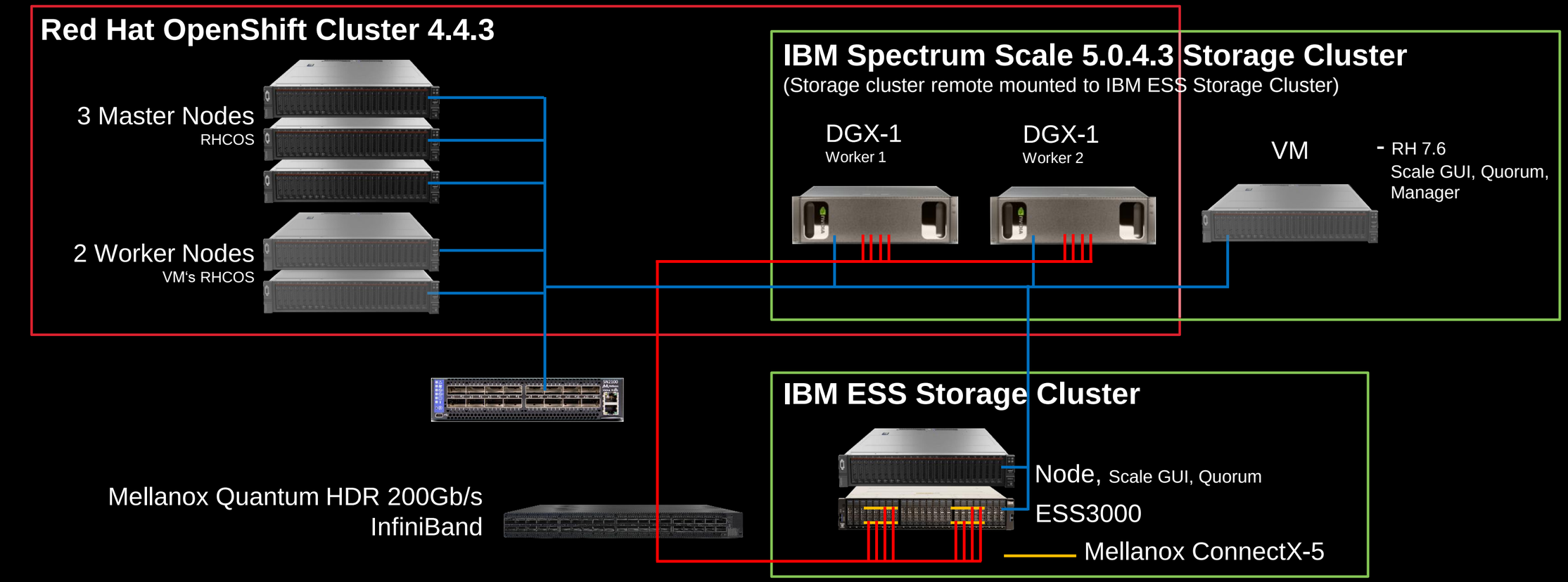
Mellanox Quantum HDR 200Gb/s InfiniBand

IBM ESS Storage Cluster

Node, Scale GUI, Quorum
ESS3000
Mellanox ConnectX-5

100Gb IB EDR Data Network 10.10.10.x/24

Network options, Infiniband



Installation Steps

- Integrating DGX-1 Systems as Worker Nodes into the Red Hat OpenShift Cluster
- Adding DGX-1 Systems as Client Nodes to the IBM Spectrum Scale Cluster
- Configure Red Hat OpenShift Stack: GPU, RDMA, Spectrum Scale CSI, MPI

Installation Steps: Integrate DGX-1 Systems as Worker Nodes

- Installing Red Hat Enterprise Linux 7.6 Base OS and DGX Software
  https://docs.nvidia.com/dgx/dgx-rhel-install-guide/installing-rhel7.html#installing-rhel7https://docs.nvidia.com/dgx/dgx-rhel-install-guide/installing-dgx-sw.html#installing-required-components
  Note: Stop right before step to install the NVIDIA® CUDA® driver.
- Installing NVIDIA Mellanox InfiniBand Drivers (MLNX_OFED)
  https://docs.nvidia.com/dgx/dgx-rhel-install-guide/installing-ib-drivers.html#installing-ib-drivers
  Note: Stop right before step to install the NVIDIA peer memory module (nvidia-peer-memory-dkms).
- Installing GPUDirect RDMA Kernel Module
  Manually build nvidia-peer-memory kernel module from https://github.com/Mellanox/nv_peer_memory
- Installing NVIDIA Mellanox SELinux Module
  Download and apply: `# semodule -i infiniband.pp`
  https://docs.mellanox.com/download/attachments/19804150/infiniband.zip?version=1&modificationDate=1575464686823&api=v2&download=true
- Adding DGX-1 systems as Worker Nodes to the Red Hat OpenShift Cluster
  https://docs.openshift.com/container-platform/4.4/machine_management/user_infra/adding-rhel-compute.html

Installation Steps: Add DGX-1 Nodes to Spectrum Scale Cluster

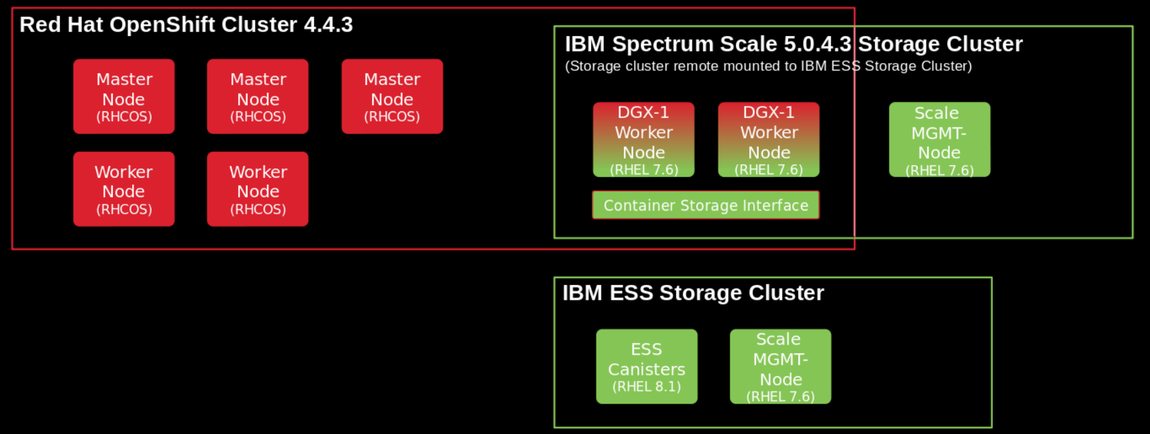
Add DGX-1 worker nodes as IBM Spectrum Scale *Client nodes* to the local Spectrum Scale cluster:

```
# ./spectrumscale node add dgx01.ocp4.scale.ibm.com
# ./spectrumscale node add dgx02.ocp4.scale.ibm.com
# ./spectrumscale install [--precheck]
# ./spectrumscale deploy [--precheck]
```

See: https://www.ibm.com/support/knowledgecenter/en/STXKQY_5.0.4/com.ibm.spectrum.scale.v5r04.doc/bl1ins_addingtoaninstallation.htm

Configure InfiniBand RDMA

```
[dgx]
verbsRdma enable
verbsRdmaSend yes
verbsPorts mlx5_0/1 mlx5_1/1 mlx5_2/1 mlx5_3/1
```



Remote Mount of ESS filesystem

```
# mmremotecluster show
Cluster name:    ess3000.bda.scale.ibm.com
Contact nodes:   fscc-fab3-3-a-priv.bda.scale.com,fscc-fab3-3-b-priv.bda.scale.com
File systems:    ess3000_4M (ess3000_4M)
```

See: https://www.ibm.com/support/knowledgecenter/STXKQY_5.0.4/com.ibm.spectrum.scale.v5r04.doc/bl1adv_admrmsec.htm

Installation Steps: Configure Red Hat OpenShift Stack

Add-Ons:

- Special Resource Operator (SRO)
- NVIDIA Mellanox RDMA Shared Device Plugin
- MPI Operator
- IBM Spectrum Scale CSI Driver

Additional Steps:

- Enabling `IPC_LOCK` in User Namespace for RDMA Shared Device Plugin

OpenShift: *Special Resource Operator* (GPU support)

https://github.com/openshift-psap/special-resource-operator

- Provides driver support (e.g. CUDA for NVIDIA GPUs)
- Adds **`nvidia.com/gpu`** as a new resource for K8s scheduler

(1) Install from OperatorHub in OpenShift Web Console:

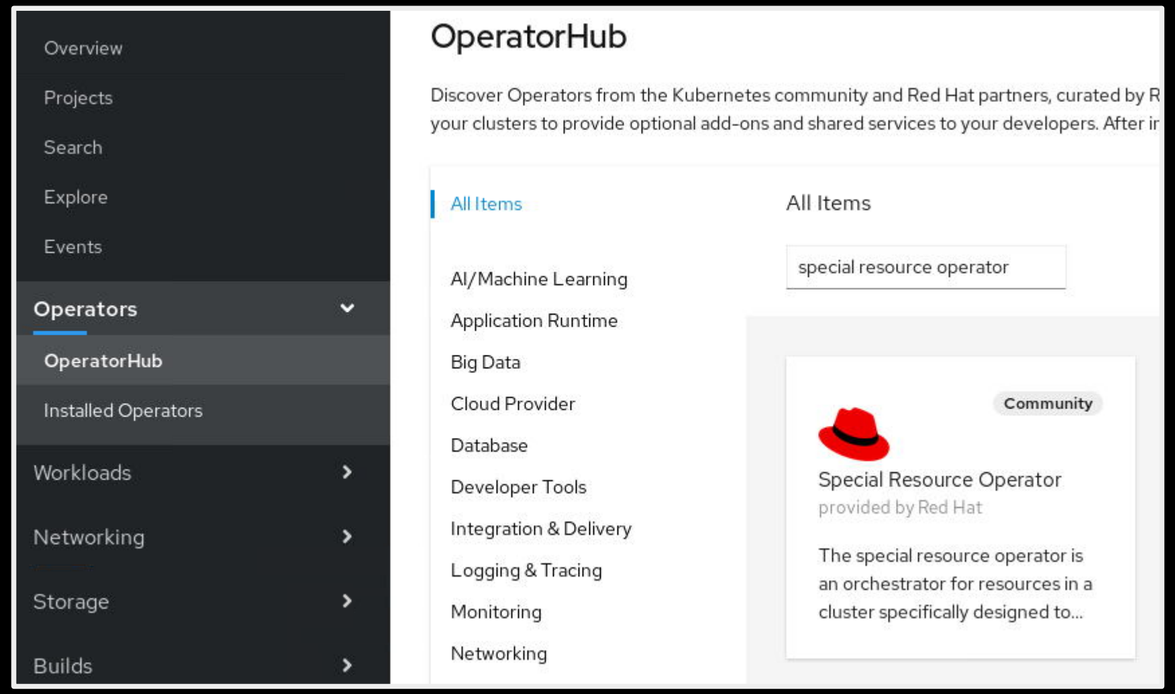
(2) Install manually:

1. Installation of the *Node Feature Discovery* (NFD) operator

```
# git clone https://github.com/openshift/cluster-nfd-operator
# cd cluster-nfd-operator
# make deploy
```

2. Installation of SRO (requires NFD operator as dependency)

```
# git clone https://github.com/openshift-psap/special-resource-operator
# cd special-resource-operator
# PULLPOLICY=Always make deploy
```




```
# oc describe node dgx01.ocp4.scale.com
Allocatable:
  nvidia.com/gpu:     8
```

OpenShift: *NVIDIA Mellanox RDMA Shared Device Plugin*

https://github.com/Mellanox/k8s-rdma-shared-dev-plugin

- Provides shared access to InfiniBand (IB) ports for non-privileged pods
- Adds **`rdma/[my-name-ibX]`** as a new resource for K8s scheduler

IB enables high performance communication between GPUs with NVIDIA Collective Communications Library (NCCL) via RDMA so that multi-node workloads can scale out seamlessly across worker nodes.

Install manually:

```
# git clone https://github.com/Mellanox/k8s-rdma-shared-dev-plugin.git
# oc apply -f images/k8s-rdma-shared-dev-plugin-config-map.yaml
# oc apply -f images/k8s-rdma-shared-dev-plugin-ds.yaml
```

```
# oc describe node dgx01.ocp4.scale.com
Allocatable:
  rdma/shared_ib0:    100
  rdma/shared_ib1:    100
  rdma/shared_ib2:    100
  rdma/shared_ib3:    100
```

Configured via *ConfigMap*:

```
apiVersion: v1
kind: ConfigMap
metadata:
 name: rdma-devices
data:
 config.json: |
  {
    "configList": [{
      "resourceName": "shared_ib0",
      "rdmaHcaMax": 100,
      "devices": ["ib0"]
     }, ...
    ]
  }
```

OpenShift: *NVIDIA Mellanox RDMA Shared Device Plugin*

https://github.com/Mellanox/k8s-rdma-shared-dev-plugin

- Provides shared access to InfiniBand (IB) ports for non-privileged pods
- Adds **`rdma/[my-name-ibX]`** as a new resource for K8s scheduler

IB enables high performance communication between GPUs with NVIDIA Collective Communications Library (NCCL) via RDMA so that multi-node workloads can scale out seamlessly across worker nodes.

Install manually:

```
# git clone https://github.com/Mellanox/k8s-rdma-shared-dev-plugin.git
# oc apply -f images/k8s-rdma-shared-dev-plugin-config-map.yaml
# oc apply -f images/k8s-rdma-shared-dev-plugin-ds.yaml
```

```
# oc describe node dgx01.ocp4.scale.com
Allocatable:
  rdma/shared_ib0:    100
  rdma/shared_ib1:    100
  rdma/shared_ib2:    100
  rdma/shared_ib3:    100
```

```
# cat test-hca-pod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: mofed-test-pod
spec:
  restartPolicy: OnFailure
  containers:
  - image: mellanox/centos_7_4_mofed_4_[...]
    name: mofed-test-ctr
    securityContext:
      capabilities:
        add: [ "IPC_LOCK" ]
    resources:
      limits:
        rdma/shared_ib0: 1
        rdma/shared_ib1: 1
        rdma/shared_ib2: 1
        rdma/shared_ib3: 1
    command:
    - sh
    - -c
    - |
      ls -l /dev/infiniband /sys/class/net
      sleep 1000000
```

OpenShift: *MPI Operator*

https://github.com/kubeflow/mpi-operator

- Provides custom resource definition **kind: MPIJob**
- Allows to conveniently schedule distributed multi-GPU and multi-node training jobs using *Message Passing Interface* (MPI)

Install manually:

```
# git clone https://github.com/kubeflow/mpi-operator.git
# cd mpi-operator/
# oc apply -f deploy/v1alpha2/mpi-operator.yaml
```

```
apiVersion: kubeflow.org/v1alpha2
kind: MPIJob
metadata:
  name: my-mpi-job
spec:
  slotsPerWorker: 1
  mpiReplicaSpecs:
    Launcher:
      replicas: 1
      template:
        spec:
          containers:
          - image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
            command:
            - mpirun
            - -np
            - "16"
            [...]
    Worker:
      replicas: 16
      template:
        spec:
          containers:
          - image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
            resources:
              limits:
                nvidia.com/gpu: 1
            [...]
```

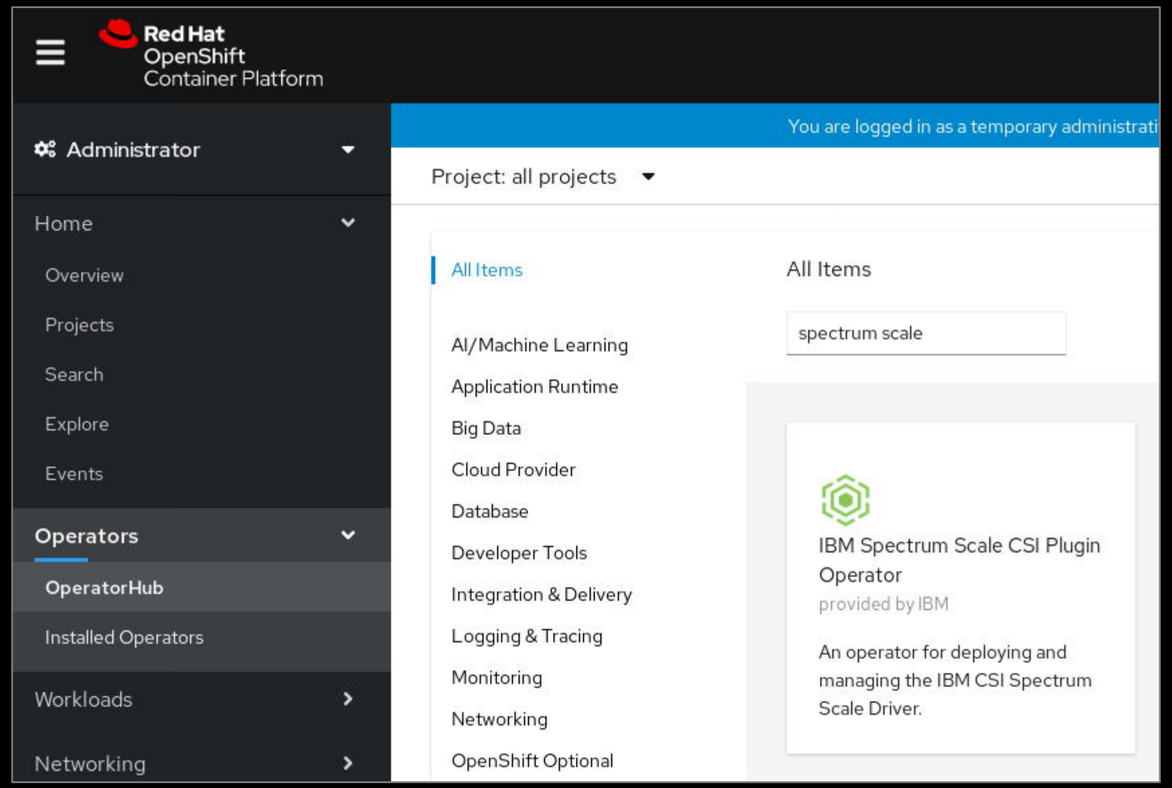
OpenShift: *IBM Spectrum Scale CSI* (storage driver)

https://github.com/IBM/ibm-spectrum-scale-csi

https://www.ibm.com/support/knowledgecenter/STXKQY_CSI_SHR/ibmspectrumscalecsi_welcome.html

- Operator deploys and manages CSI (Container Storage Interface) plugin for IBM Spectrum Scale
- Provides *persistent storage* to containers as Persistent Volumes (PVs) with IBM Spectrum Scale as storage backend

http://www.redbooks.ibm.com/redpieces/abstracts/redp5589.html

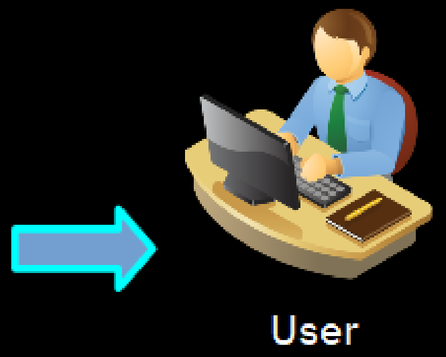


OpenShift: *IBM Spectrum Scale CSI* – Storage Provisioning

- ***Dynamic provisioning***

Admin

Admin creates *StorageClass* in OpenShift/Kubernetes



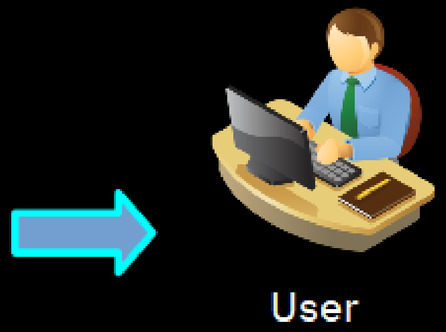
User

User claims volume (PV) from *StorageClass* through ***Persistent Volume Claim*** (PVC) (*self-service provisioning*)

- ***Static provisioning***

Admin

Admin provisions static *PVs* in OpenShift/Kubernetes



User

User claims volume (PV) from pool of pre-provisioned PVs through ***Persistent Volume Claim*** (PVC)

OpenShift: Dynamic Provisioning with *IBM Spectrum Scale CSI*

OpenShift: System admin

```
# oc apply -f scale-stgclass.yaml
# cat scale-stgclass.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: remotefs-m4
provisioner: spectrumscale.csi.ibm.com
parameters:
  volBackendFs: ess3000_4M
  clusterId: "21505721748717715"
reclaimPolicy: Delete
```

OpenShift: User namespace

```
# oc apply -f nv-priv-pvc.yaml
# cat nv-priv-pvc.yaml
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: "adas-priv-pvc"
spec:
  storageClassName: remotefs-m4
  accessModes:
  - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
```

```
spec:
  containers:
  - name: a2d2-prep
    image: tf2:20.03
    [...]
    volumeMounts:
    - name: a2d2-priv
      mountPath: /workspace
  volumes:
  - name: a2d2-priv
    persistentVolumeClaim:
      claimName: adas-priv-pvc
      readOnly: false
```

OpenShift: Static Provisioning with *IBM Spectrum Scale CSI*

OpenShift: System admin

```
# oc apply -f nv-pv01.yaml
# cat nv-pv01.yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: "adas-data-pv01"
spec:
  capacity:
    storage: 100Gi
  accessModes:
    - ReadWriteMany
  csi:
    driver: spectrumscale.csi.ibm.com
    volumeHandle: "16217308676014575381;099B6A7A:5EB99743;
                   path=/gpfs/ess3000_4M/adas"
```

OpenShift: User namespace

```
# oc apply -f nv-data-pvc.yaml
# cat nv-data-pvc.yaml
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: "adas-data-pvc"
spec:
  accessModes:
  - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
```

```
spec:
  containers:
  - name: a2d2-train
    image: tf2:20.03
    [...]
    volumeMounts:
    - name: a2d2-data
      mountPath: /data
      readOnly: false
  volumes:
  - name: a2d2-data
    persistentVolumeClaim:
      claimName: adas-data-pvc
```

No default storage class allowed! Match based on capacity & access mode!

OpenShift: Static Provisioning with *IBM Spectrum Scale CSI*

OpenShift: System admin

```
# oc apply -f nv-pv01.yaml
# cat nv-pv01.yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: "adas-data-pv01"
  labels:
    type: data
    dept: adas
spec:
  storageClassName: static
  capacity:
    storage: 100Gi
  accessModes:
    - ReadWriteMany
  csi:
    driver: spectrumscale.csi.ibm.com
    volumeHandle: "16217308676014575381;099B6A7A:5EB99743;
                   path=/gpfs/ess3000_4M/adas"
```

OpenShift: User namespace

```
# oc apply -f nv-data-pvc.yaml
# cat nv-data-pvc.yaml
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: "adas-data-pvc"
spec:
  storageClassName: static
  accessModes:
  - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
  selector:
    matchLabels:
      type: data
      dept: adas
```

```
spec:
  containers:
  - name: a2d2-train
    image: tf2:20.03
    [...]
    volumeMounts:
    - name: a2d2-data
      mountPath: /data
      readOnly: false
  volumes:
  - name: a2d2-data
    persistentVolumeClaim:
      claimName: adas-data-pvc
```

*Use of **labels & storage class annotation** allows perfect match: PV ↔ PVC*

OpenShift: Enabling **IPC_LOCK** in User Namespace

- **NVIDIA Mellanox RDMA device plugin** requires **IPC_LOCK** capability in OpenShift 4 security context.

- A ***regular user*** normally runs under the **restricted** ***Security Context Constraints (SCC)*** in OpenShift 4 so **IPC_LOCK** is not generally available to regular users.

- *A **system admin*** has access to the **privileged** ***Security Context Constraints (SCC)*** and can request the **IPC_LOCK** capability when running pods anytime.

Allowing a regular user to run MPI jobs with RDMA resources for multi-GPU training across nodes requires to grant a user access to the **IPC_LOCK** capability in the user's namespace.
The *system admin* can create a new *SCC* derived from the **restricted** SCC, extend it by the **IPC_LOCK** capability and make the new SCC available to the user namespace by creating a *service account* , a *role binding* and a *role* referencing this new SCC.

```
spec:
  serviceAccount: mpi
  serviceAccountName: mpi
  containers:
  - name: your-container-name
    image: your-container-image:tag
    securityContext:
      capabilities:
        add: [ "IPC_LOCK" ]
```

OpenShift: Enabling `IPC_LOCK` in User Namespace

(1) Derive a new *SCC* from `restricted` SCC and add `IPC_LOCK` capability

(2) Create a new *service account* in the user namespace

(3) Create a new *role* in the user namespace

(4) Create a new *role binding* in the user namespace

```
# oc get scc restricted -o yaml > mpi-scc.yaml
# vi mpi-scc.yaml
[...]
defaultAddCapabilities:
- IPC_LOCK
metadata:
  name: scc-for-mpi
users:
- system:serviceaccount:[user-namespace]:mpi
[...]
# oc apply -f mpi-scc
```

```
# oc apply -f mpi-sa.ya
# cat mpi-sa.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: mpi
  namespace: [user-name
```

```
# oc apply -f mpi-role.yaml
# cat mpi-role.yaml
apiVersion: rbac.authorization.k8s.
kind: Role
metadata:
  name: mpi
  namespace: [user-namespace]
rules:
- apiGroups:
  - security.openshift.io
  resources:
  - securitycontextconstraints
  verbs:
  - use
  resourceNames:
  - scc-for-mpi
```

```
# oc apply -f mpi-rolebinding.yaml
# cat mpi-rolebinding.yaml
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: mpi
  namespace: [user-namespace]
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: mpi
  namespace: [user-namespace]
subjects:
- kind: ServiceAccount
  name: mpi
  namespace: [user-namespace]
userNames:
- system:serviceaccount:[user-namespace]:mpi
```

Multi-GPU, multi-Node MPI Jobs with TensorFlow

```
apiVersion: kubeflow.org/v1alpha2
kind: MPIJob
metadata:
  name: tf2-a2d2-16x01x02-gpu
spec:
  slotsPerWorker: 1
  mpiReplicaSpecs:
    Launcher:
      replicas: 1
      template:
        spec:
          containers:
          - image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
            command:
            - mpirun
            - -np
            - "16"
            [...]
    Worker:
      replicas: 16
      template:
        spec:
          containers:
          - image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
            resources:
              limits:
                nvidia.com/gpu: 1
            [...]
```

```
Launcher:
  replicas: 1
  template:
    spec:
      containers:
      - name: tf2-a2d2-16x01x02-gpu
        image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
        command:
        - mpirun
        - -np
        - "16"
        - -wdir
        - "/workspace/scripts/tf2_comparison/hvd"
        [...]
        - -x
        - NCCL_DEBUG=INFO
        - -x
        - NCCL_IB_DISABLE=0
        - -x
        - NCCL_NET_GDR_LEVEL=1
        [...]
        - python
        - main.py
        - --model_dir=checkpt
        - --batch_size=16
        - --exec_mode=train
        - --max_steps=16000
```

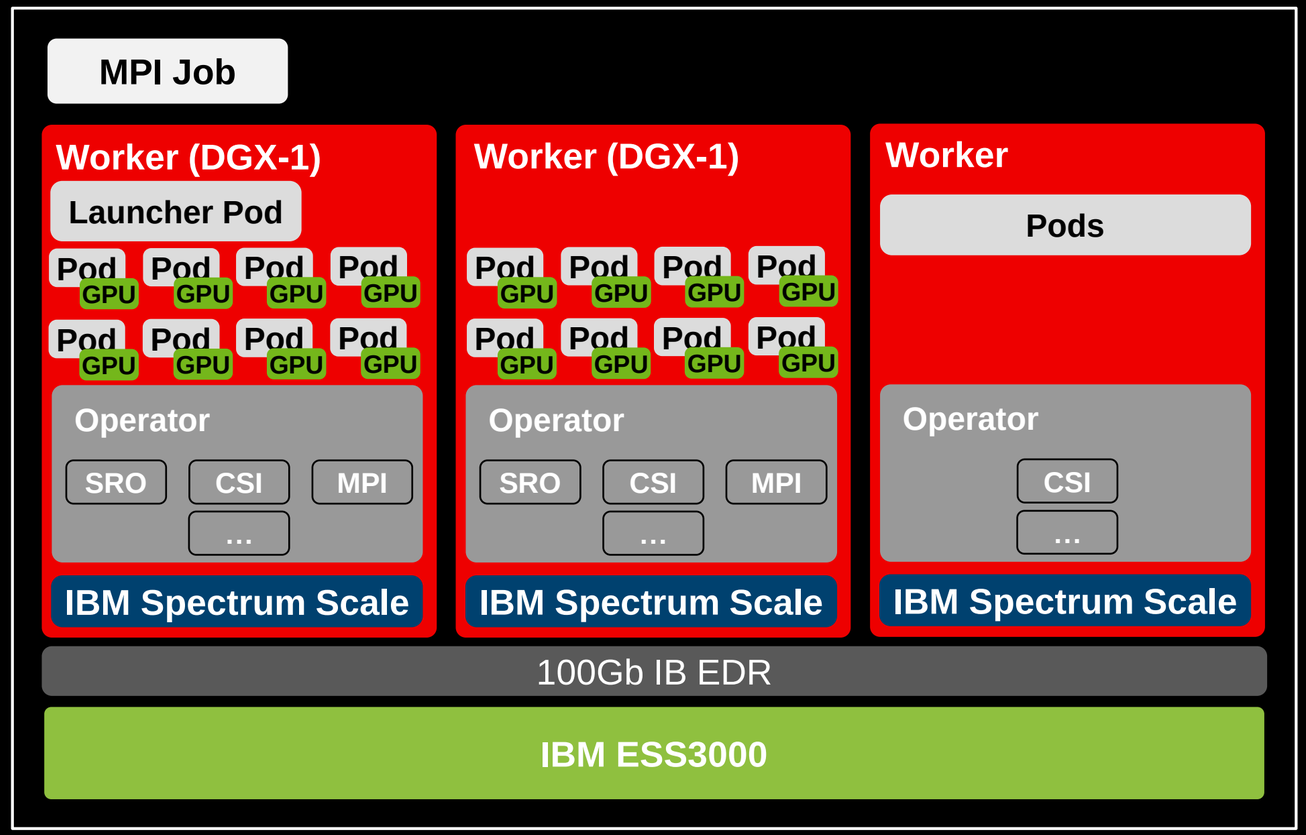
Multi-GPU, multi-Node MPI Jobs with TensorFlow

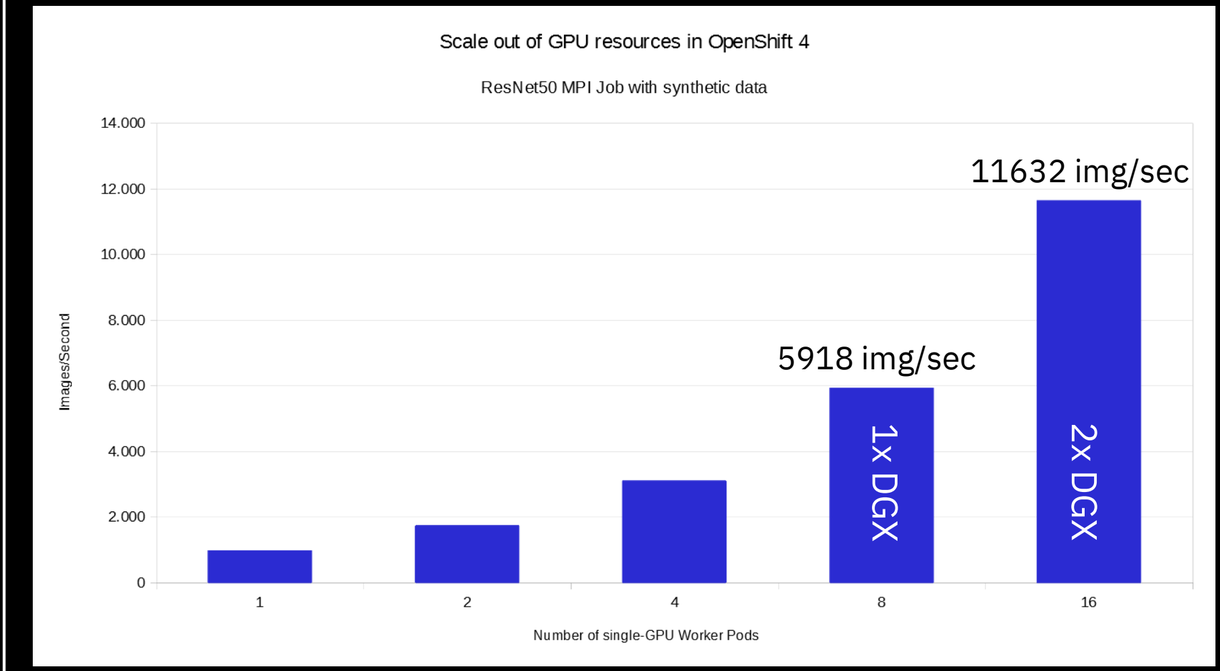
```
apiVersion: kubeflow.org/v1alpha2
kind: MPIJob
metadata:
  name: tf2-a2d2-16x01x02-gpu
spec:
  slotsPerWorker: 1
  mpiReplicaSpecs:
    Launcher:
      replicas: 1
      template:
        spec:
          containers:
          - image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
            command:
            - mpirun
            - -np
            - "16"
            [...]
    Worker:
      replicas: 16
      template:
        spec:
          containers:
          - image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
            resources:
              limits:
                nvidia.com/gpu: 1
            [...]
```

```
Worker:
  replicas: 16
  template:
    spec:
      serviceAccount: mpi
      serviceAccountName: mpi
      containers:
      - name: tf2-a2d2-16x01x02-gpu
        image: nvcr.io/nvidia/tensorflow:20.03-tf2-py3
        securityContext:
          capabilities:
            add: [ "IPC_LOCK" ]
        resources:
          limits:
            nvidia.com/gpu: 1
            rdma/shared_ib0: 1
            rdma/shared_ib1: 1
            rdma/shared_ib2: 1
            rdma/shared_ib3: 1
        volumeMounts:
        - name: a2d2-data
          mountPath: /workspace
          readOnly: false
      volumes:
      - name: a2d2-data
        persistentVolumeClaim:
          claimName: adas-data-pvc
```

Multi-GPU, multi-Node GPU Scaling with Tensorflow ResNet50

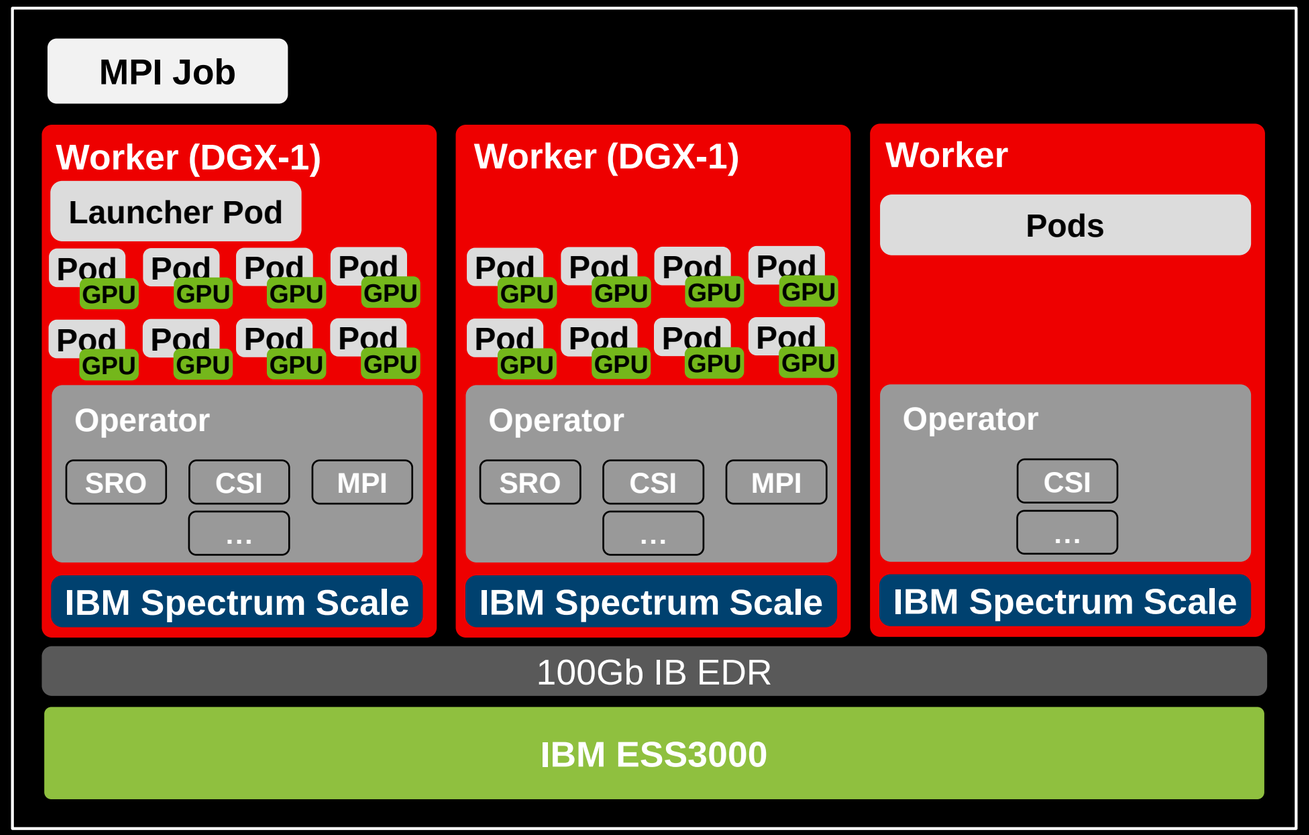
1x Launcher Pod starting
16x Worker Pods with 1 GPU each (i.e. all Pods evenly scaled across all GPUs and DGX Worker Nodes)

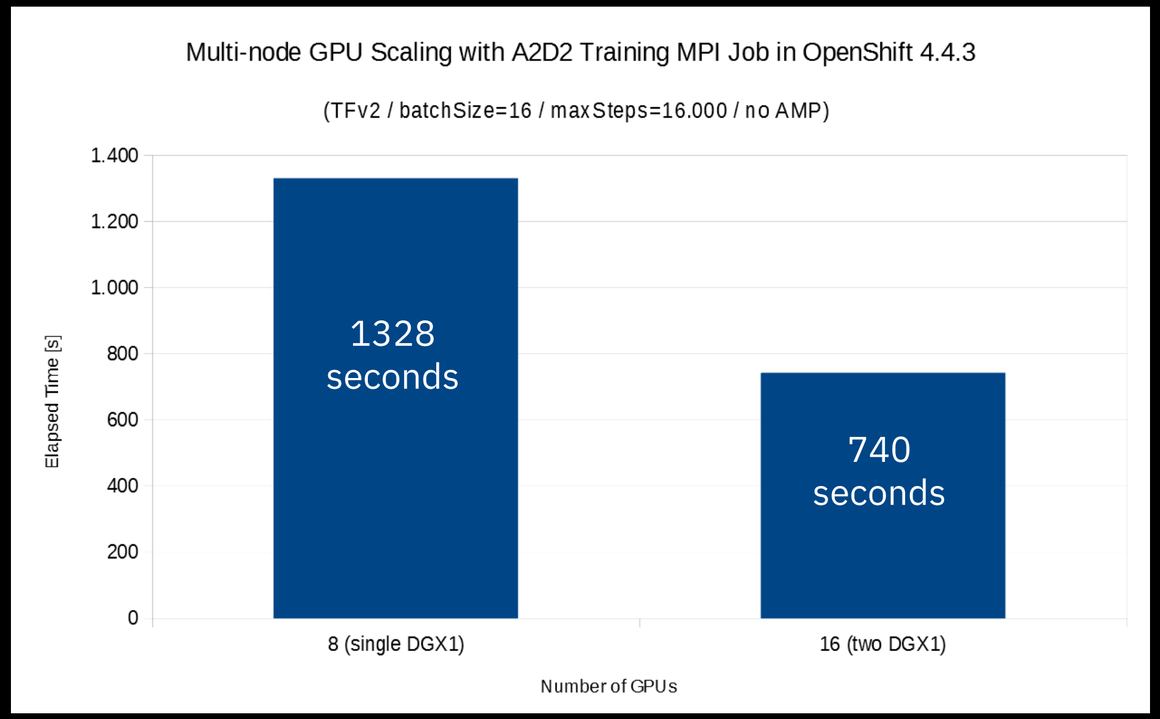




Multi-Node GPU Scaling with A2D2 Training

1x Launcher Pod,
starting 16x Worker Pods
each accessing 1 GPU, scale across all nodes





There is no AI without IA

(information architecture)

Know your data

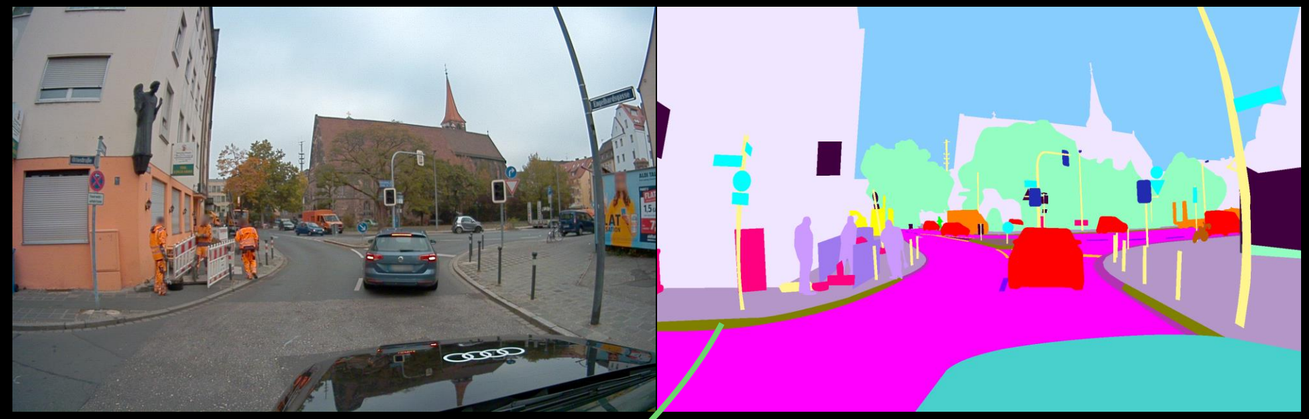
What?
Where?
When?
…

Use your data

at the right time

Autonomous Driving Dataset used

- Audi Autonomous Driving Dataset (A2D2) published by Audi https://www.a2d2.audi
- Six cameras and five Li-DAR units, providing full 360° coverage
- Data is time synchronized and mutually registered
- 41,277 frames with semantic segmentation image and point cloud labels
- Semantic segmentation dataset features 38 categories

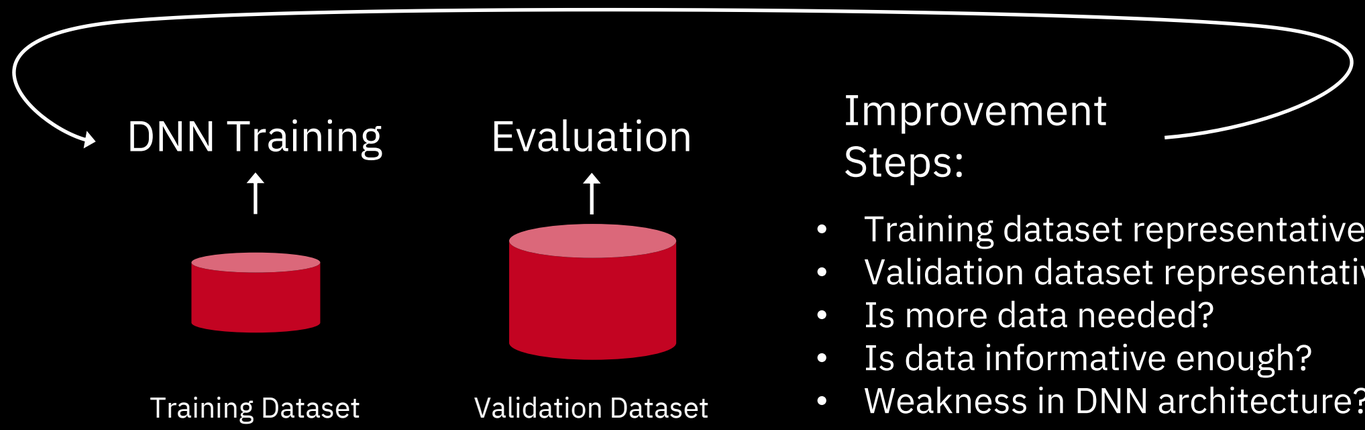


(A2D2- Jacob Geyer et al, 2020, https://arxiv.org/abs/2004.06320)

```
26      "#ffffc8": "Utility vehicle 2",
27      "#e96400": "Sidebars",
28      "#6e6e00": "Speed bumper",
29      "#808000": "Curbstone",
30      "#ff[illegible]125": "Solid line",
31      "#40[illegible]040": "Irrelevant signs",
32      "#b9[illegible]a57": "Road blocks",
33      "#000[illegible]64": "Tractor",
34      "#8b6363": "Non-drivable street",
35      "#d23273": "Zebra crossing",
36      "#ff0080": "Obstacles / trash",
37      "#fff68f": "Poles",
38      "#960096": "RD restricted area",
39      "#ccff99": "Animals",
```

	A	H	I	J	K	L	M	N	O	P
1	Filename	Car	Bicycle	Pedestrian	Truck	Small vehicles	Traffic signal	Traffic sign	Utility vehicle	Sidebars
50	62316.png	39564	3155	2302	0	0	0	7626	0	0
51	78793.png	48538	0	0	7671	0	113	465	0	0
52	12593.png	238144	1511	0	0	1342	0	8956	0	0
53	20783.png	31558	1525	4213	0	0	3337	20014	3917	0
54	27813.png	10728	14712	585	3514	0	5184	3814	1762	0
55	70939.png	31037	1889	187	0	0	0	18397	0	0
56	00488.png	11482	0	0	0	0	130	180	0	0
57	42035.png	299263	4371	0	0	0	4149	3565	0	0
58	73496.png	15546	0	0	0	0	0	7331	0	1955
59	07232.png	25964	4476	5941	0	0	0	17881	0	0
60	74320.png	80	0	0	0	0	0	233	0	4041
61	75272.png	2796	0	0	0	0	0	835	0	137
62	21604.png	172247	3282	0	0	0	180	1413	0	0
63	41344.png	32295	14794	243	13485	0	2348	2346	0	0
64	73456.png	43828	0	0	0	0	0	6814	0	5598
65	68156.png	108909	2071	0	498	0	0	3995	0	0
66	46816.png	213287	0	103	197	0	332	5287	0	0
67	56672.png	102111	0	0	1794	0	0	25615	34367	0
68	22033.png	3783	169	16855	0	585	688	1846	0	0
69	06963.png	28567	0	210	9933	0	0	3346	0	0
70	50297.png	232642	0	8496	0	0	0	7660	0	0
71	62375.png	39551	3061	3878	0	0	0	7880	0	0
72	71954.png	156984	0	26788	41861	0	0	80	0	0

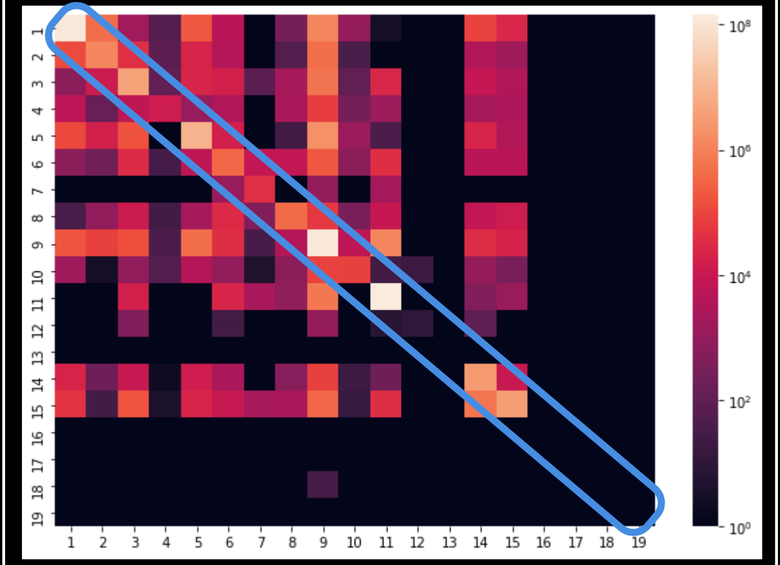
Building the right training and validation dataset



Improvement Steps:

- Training dataset representative?
- Validation dataset representative?
- Is more data needed?
- Is data informative enough?
- Weakness in DNN architecture?

- Building a representative validation dataset is challenging
- Validation dataset is significantly larger than the training dataset
- Validating the trained DNN against a large validation dataset is critical to understand its weaknesses



Confusion matrix presenting the training evaluation

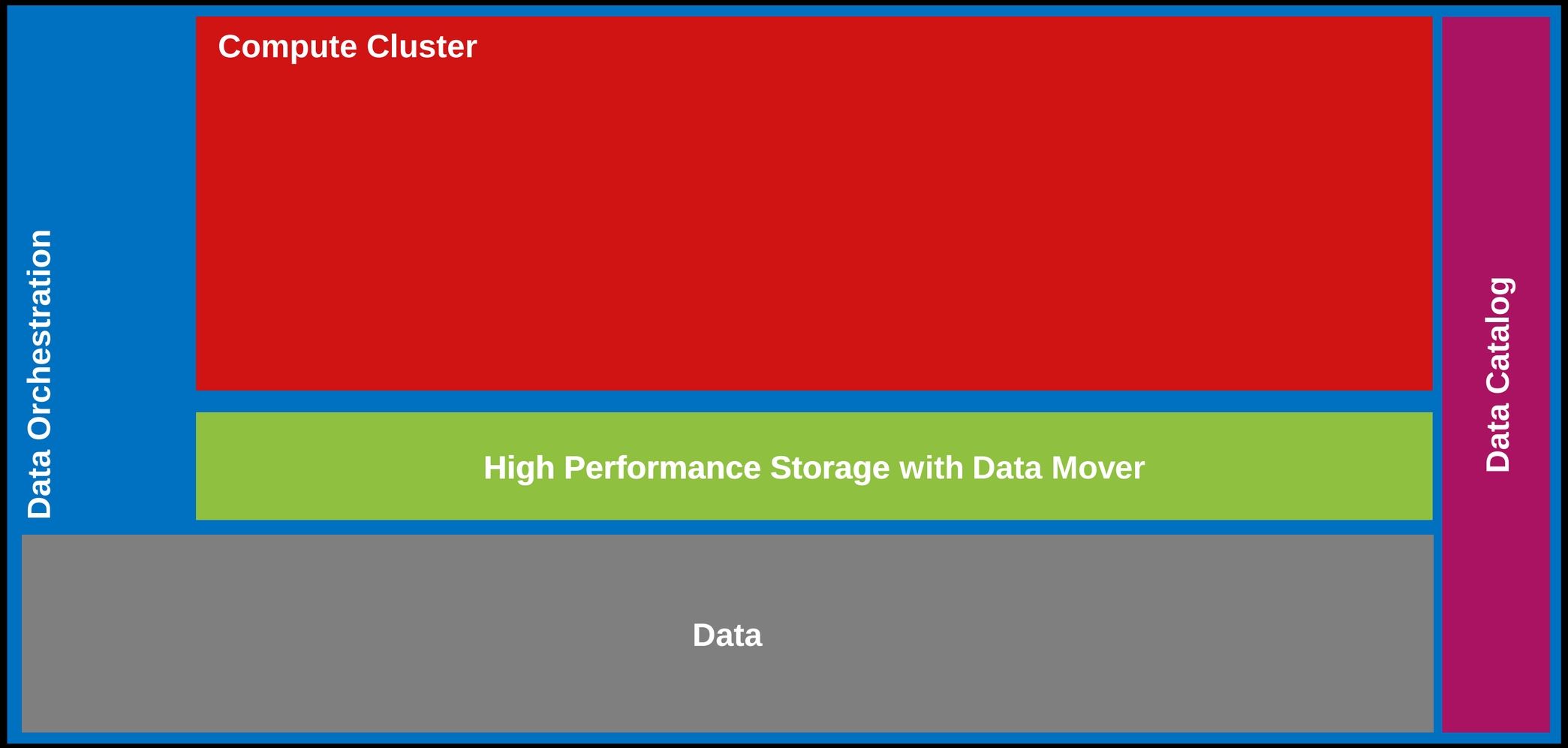
For a perfect predictor:

Diagram would only have a diagonal line from top left to bottom right.

That would read as the network would have classified all pixels of a certain class right.

Data Orchestration

- Abstract data access across storage systems
- Present data with a global namespace
- Train with the right data
- Ensure data is available at the right time

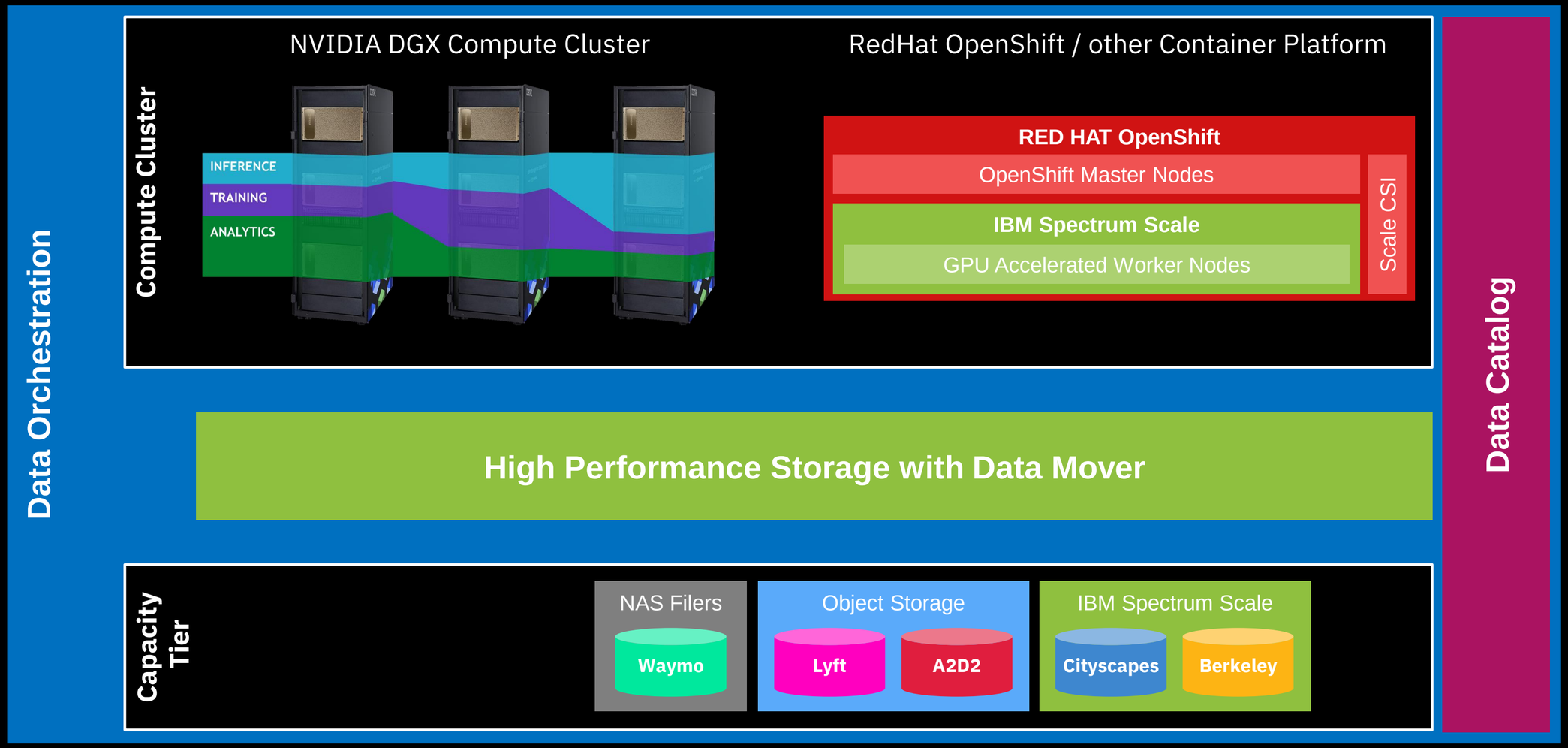


Data Orchestration

- Abstract data access across storage systems
- Present data with a global namespace
- Train with the right data
- Ensure data is available at the right time

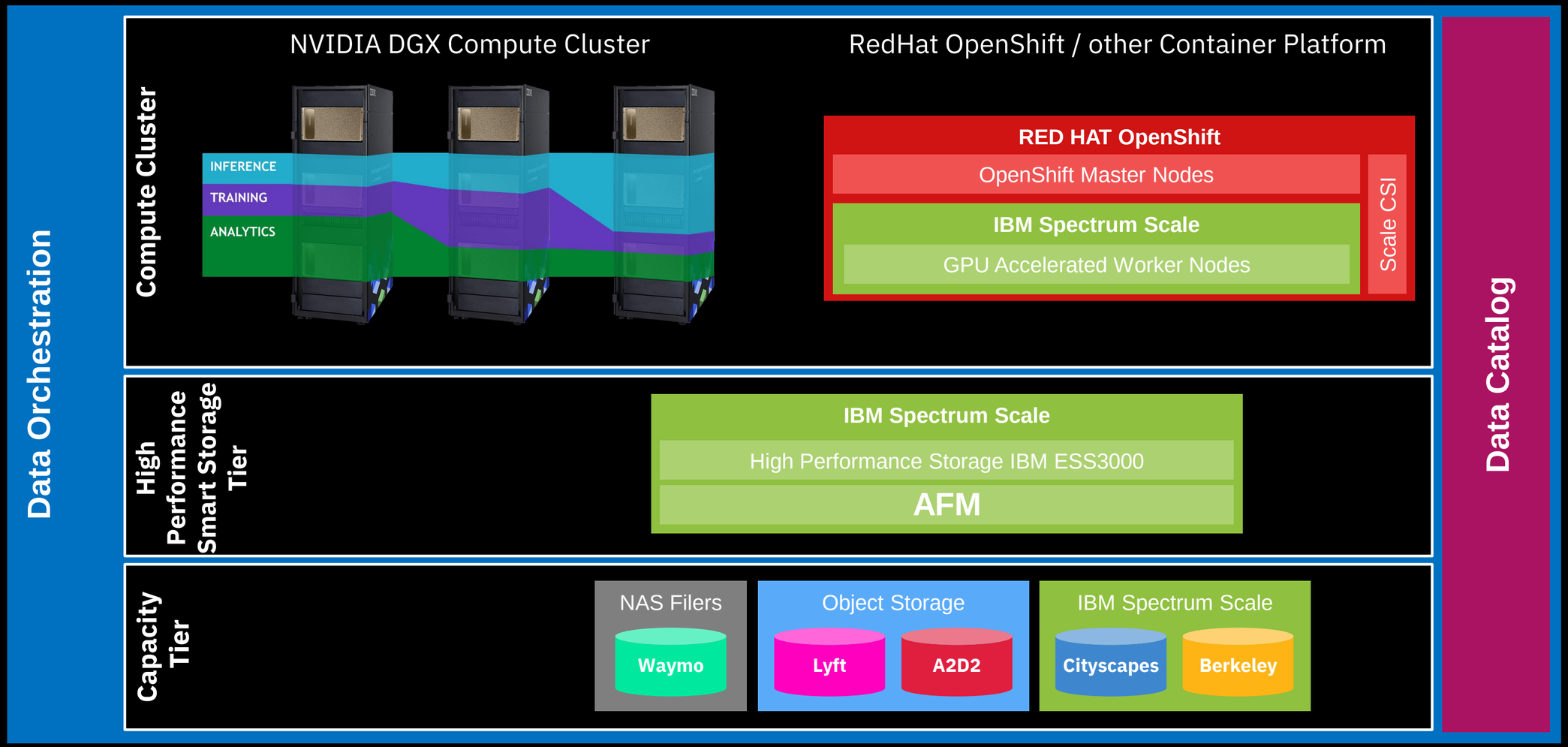
Data Orchestration

- Abstract data access across storage systems
- Present data with a global namespace
- Train with the right data
- Ensure data is available at the right time



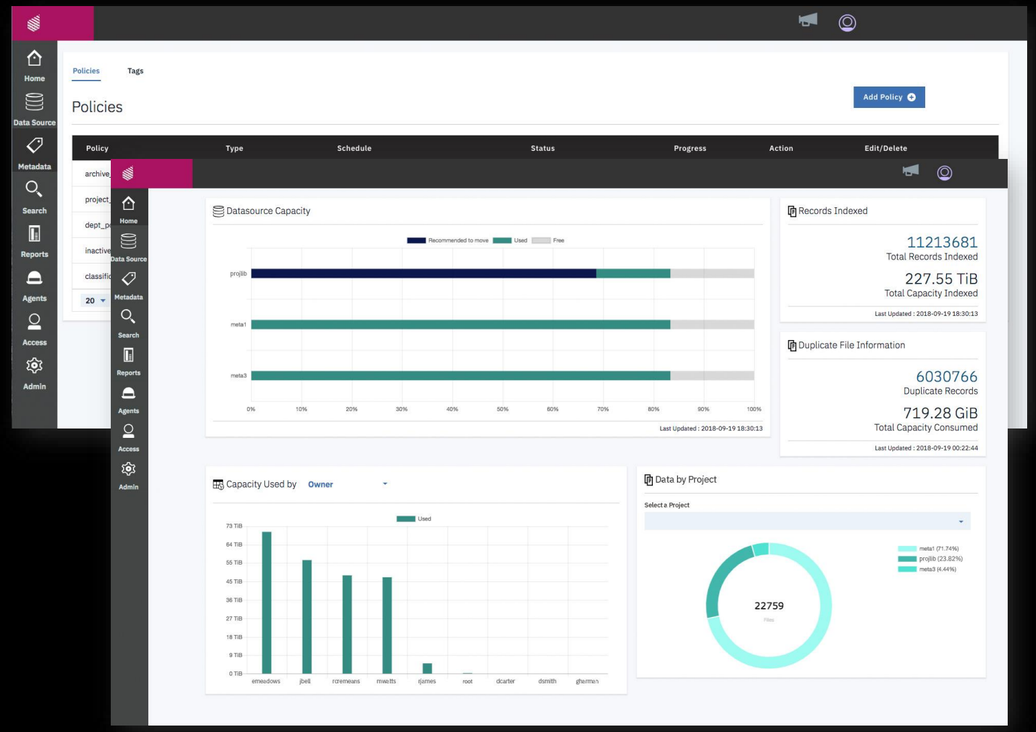
Data Orchestration

- Abstract data access across storage systems
- Present data with a global namespace
- Train with the right data
- Ensure data is available at the right time



IBM Spectrum Discover

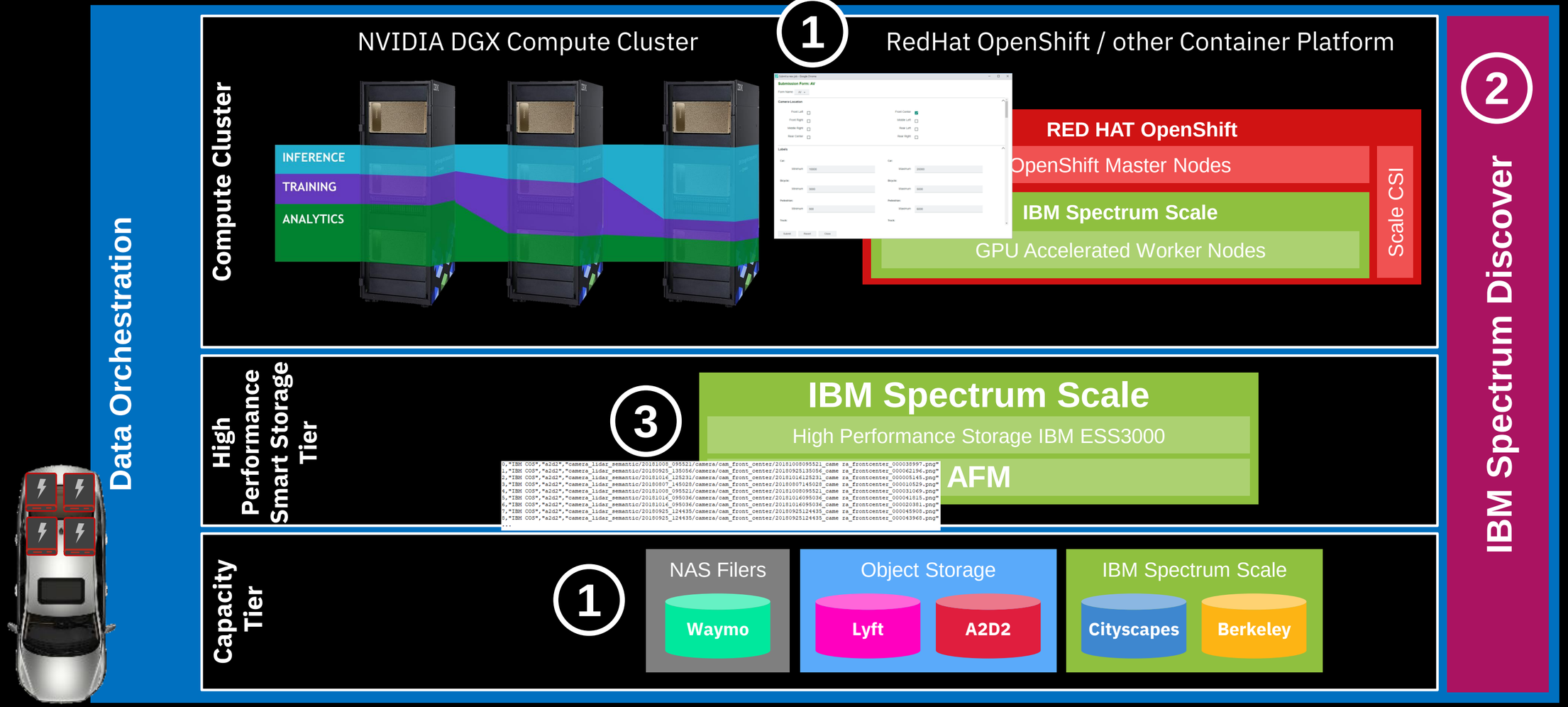
- Automate cataloging/indexing
- Locate and identify for data users or IT Admin
- Manage data governance or analyze for security
- Enable comprehensive insight
- Create custom action agents
- Enable security analysis and data governance



Search billions of files/objects in 0.5 sec and manage AI workflows, data security analysis and data governance

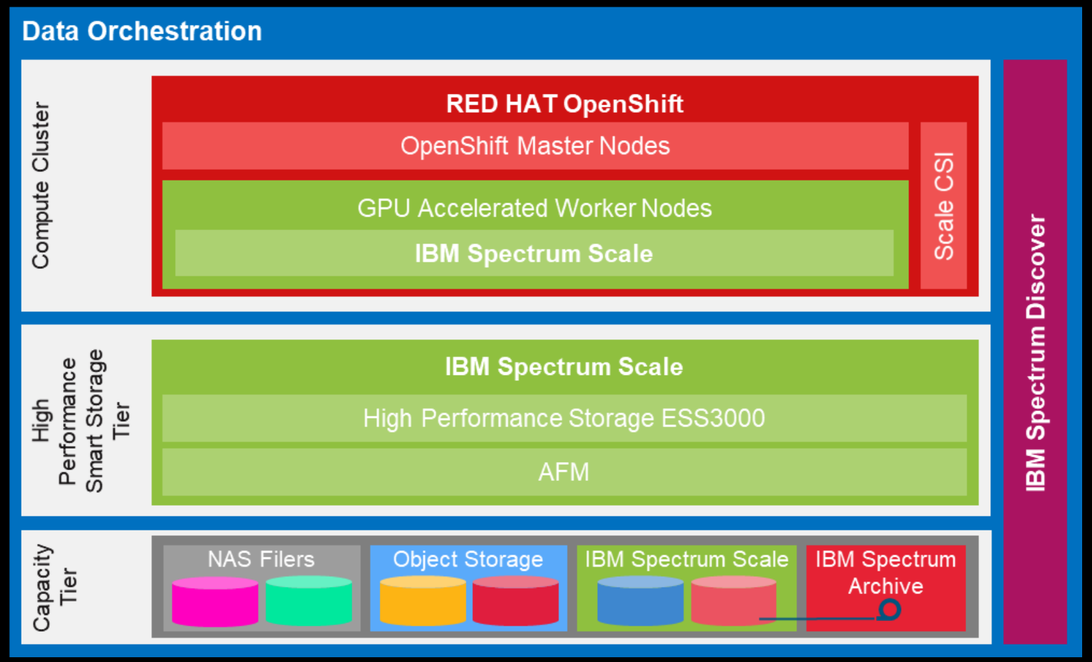
Data Orchestration

- Abstract data access across storage systems
- Present data with a global namespace
- Train with the right data
- Ensure data is available at the right time



IBM Redpaper soon to be published on the topic:

Data Accelerator for AI and Analytics

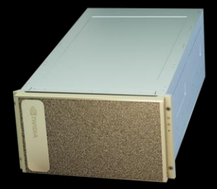


LEARN MORE

www.ibm.com/it-infrastructure/storage/spectrum

www.openshift.com/

www.nvidia.com/dgx-pod

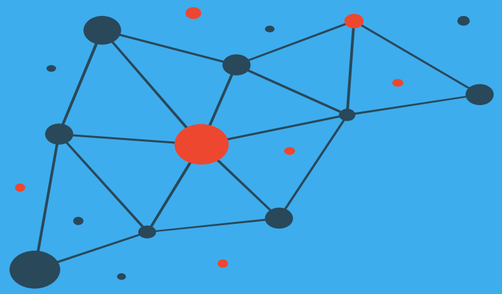


NVIDIA DGX A100

IBM Spectrum Scale

IBM Spectrum Discover

Red Hat OpenShift



Check https://www.spectrumscaleug.org/experttalks for charts, show notes and upcoming talks

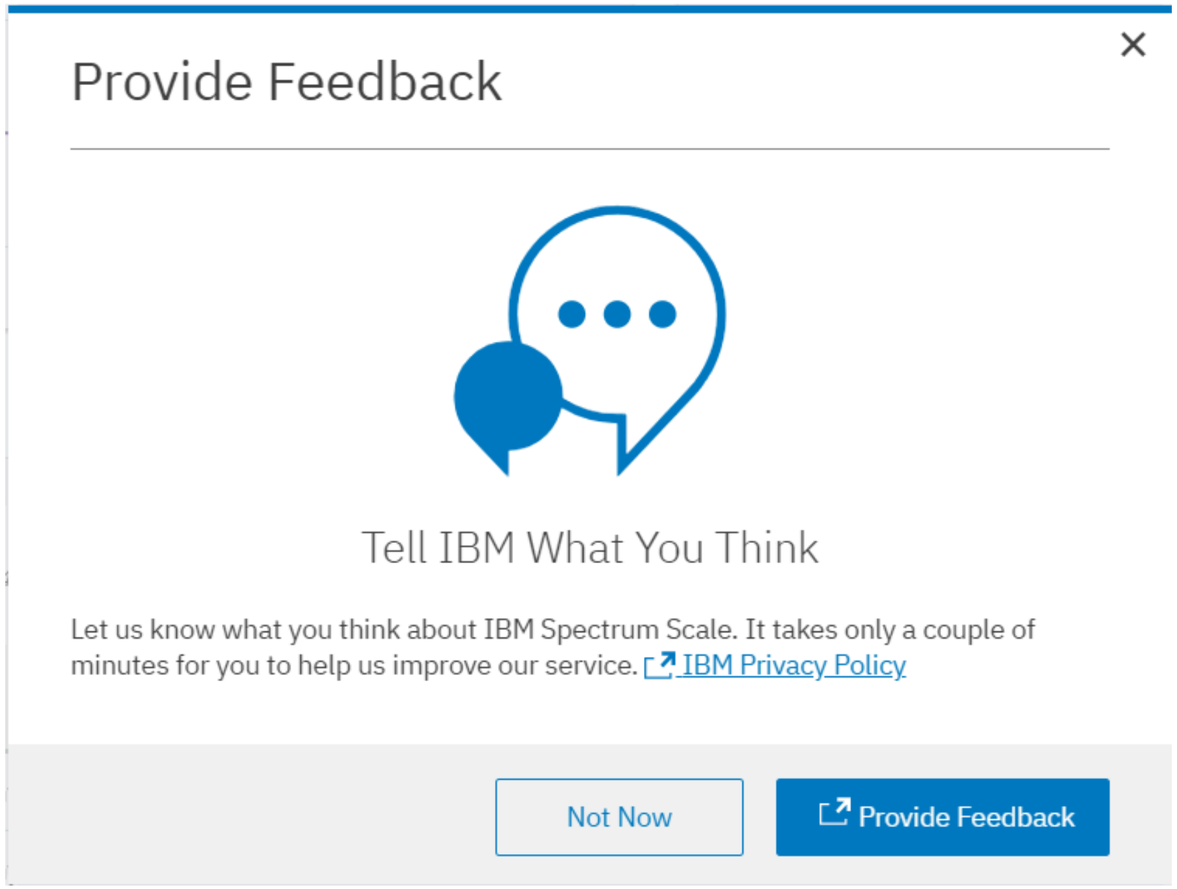
- Past talks:
  - 001: What is new in Spectrum Scale 5.0.5?
  - 002: Best practices for building a stretched cluster
  - 003: Strategy update
  - 004: Update on performance enhancements in Spectrum Scale (file create, MMAP, direct IO, ESS 5000)
  - 005: Update on functional enhancements in Spectrum Scale (inode management, vCPU scaling, NUMA considerations)
  - 006: Persistent Storage for Kubernetes and OpenShift environments
  - 007: Manage the lifecycle of your files using the policy engine
- Today:
  - Nov 4: Multi-node scaling of AI workloads using NVIDIA DGX, OpenShift and Spectrum Scale
- Next:
  - Nov 16: User Meeting at SC20 (Session 1) – Storage for AI https://www.spectrumscaleug.org/event/sc20-meeting-session-1-storage-for-ai/
  - Nov 18: User Meeting at SC20 (Session 2) – What is new in Spectrum Scale 5.1? https://www.spectrumscaleug.org/event/sc20-meeting-session-2-what-is-new-in-spectrum-scale-5-1/

Thank you!

IBM Spectrum Scale

Please help us to improve Spectrum Scale with your feedback

- If you get a survey in email or a popup from the GUI, please respond
- We read every single reply



Spectrum Scale User Group

The Spectrum Scale (GPFS) User Group is free to join and open to all using, interested in using or integrating IBM Spectrum Scale.

The format of the group is as a web community with events held during the year, hosted by our members or by IBM.

See our web page for upcoming events and presentations of past events. Join our conversation via mail and Slack.

www.spectrumscaleug.org