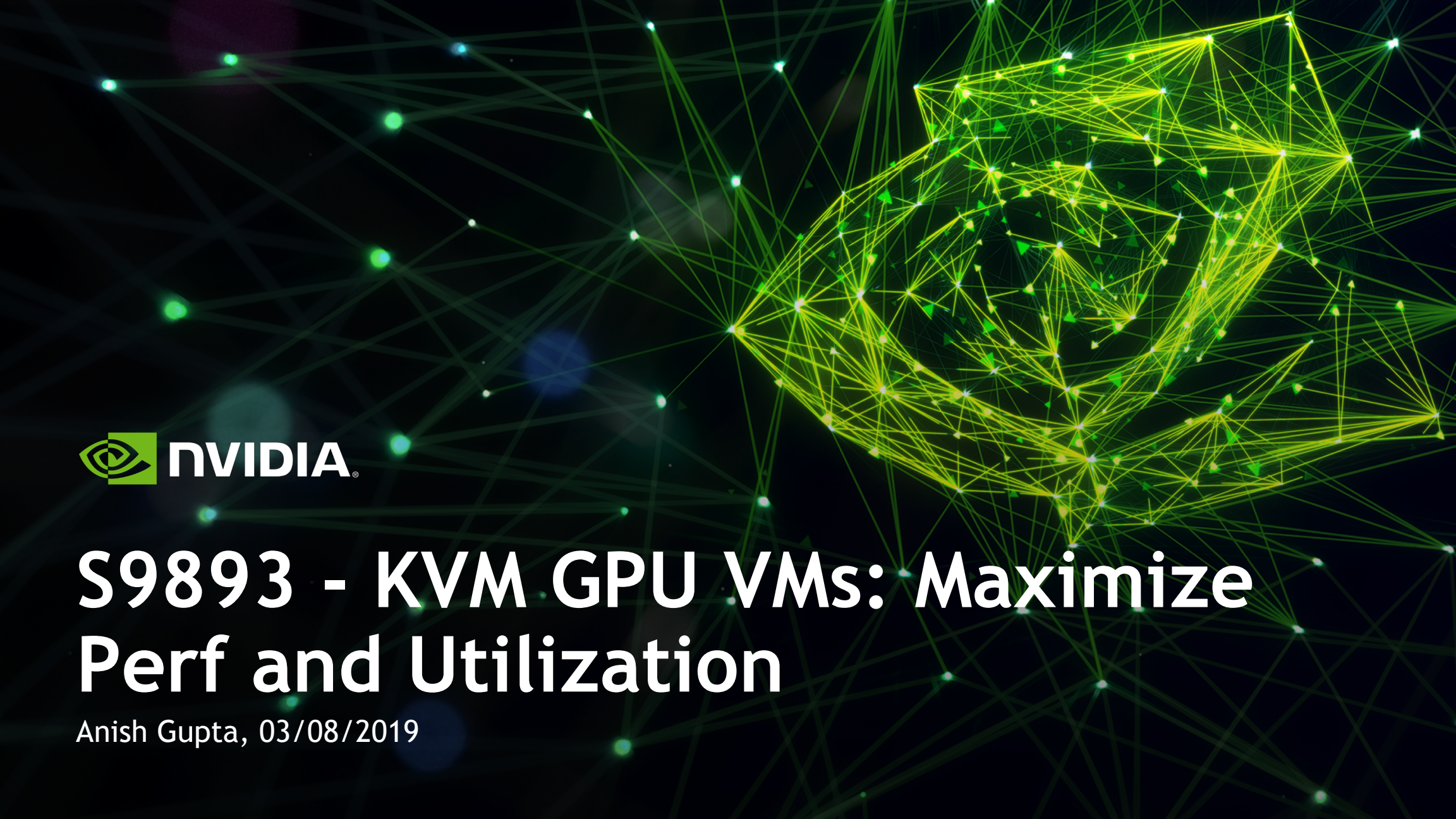


S9893 - KVM GPU VMs: Maximize Perf and Utilization

Anish Gupta, 03/08/2019

Agenda

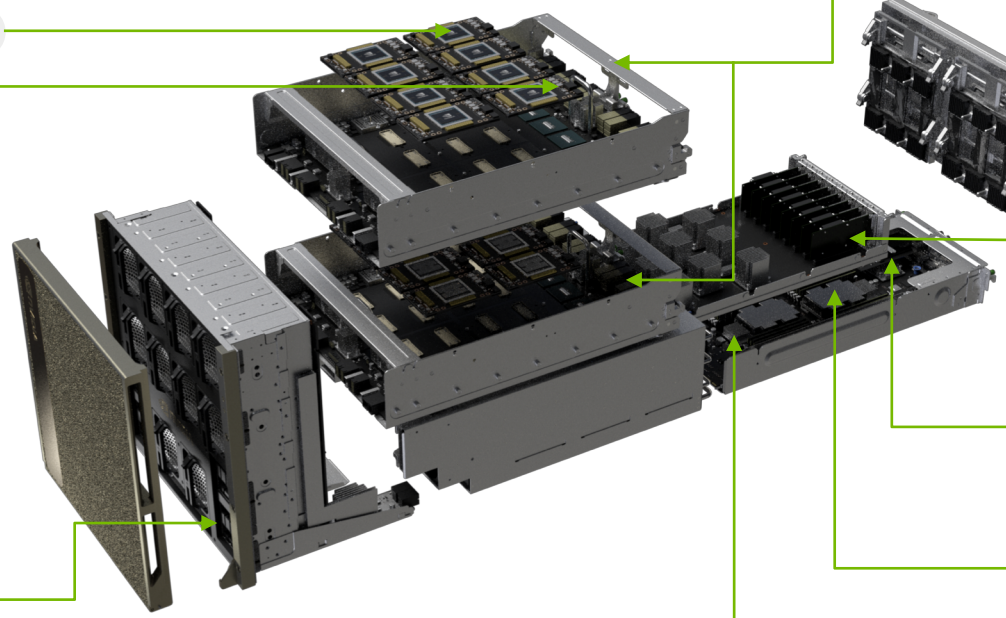
INTRODUCING KVM VIRTUALIZATION ON DGX-2

KVM BENEFITS

- SECURE MULTI-TENANCY
- IMPROVE SYSTEM AVAILABILITY
- EASY TO USE

HOW WE ENABLE KVM

DGX-2 has Immense compute



NVIDIA Tesla V100 32GB 1

Twelve NVSwitches 3
2.4 TB/sec bi-section bandwidth

30 TB NVME SSDs 7
Internal Storage
25GB/s seq read (fio)

2 **Two GPU Boards**
8 V100 32GB GPUs per board
6 NVSwitches per board
512GB Total HBM2 Memory
interconnected by
Plane Card

4 **Eight EDR Infiniband/100 GigE**
200 GB/sec Total
Bi-directional Bandwidth

8 **Two High-Speed Ethernet**

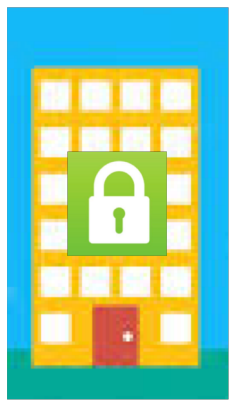
5 **Two Intel Xeon Platinum CPUs**

6 **1.5 TB System Memory**

3

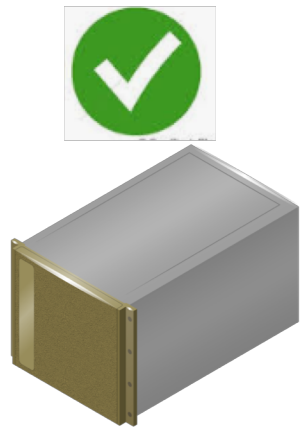
PROBLEM STATEMENT WITH GPU VIRTUALIZATION

SECURE MULTI-TENANCY



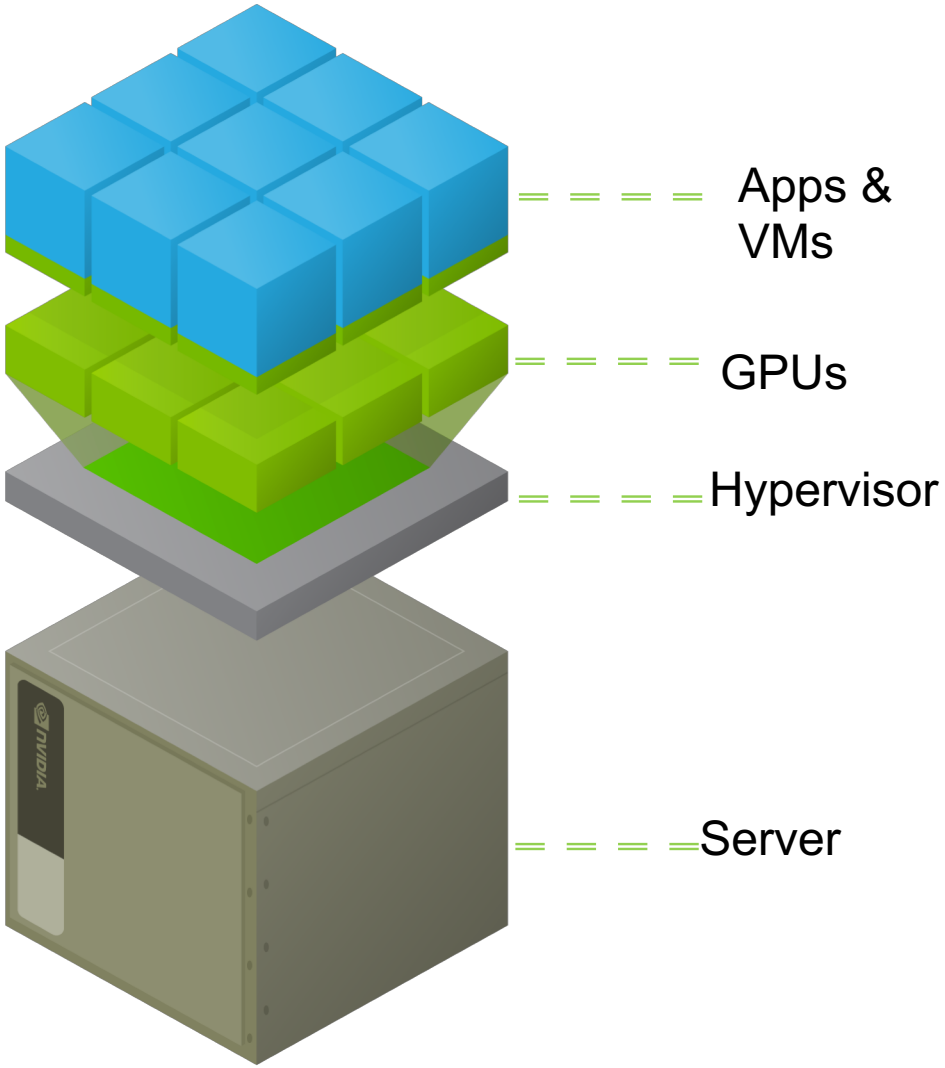
“ Launch secure tenants with different number of GPUs targeting Healthcare, CSPs, Higher Education use cases

IMPROVED SYSTEM AVAILABILITY



“ Hypervisor assisted Hardware health management with no down time

Secure Multi-Tenancy



DGX-2 KVM VIRTUALIZATION

Enable your own private DL Training Cloud for your Enterprise

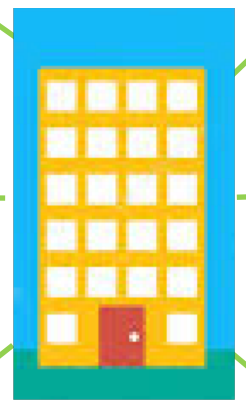
- KVM hypervisor for Ubuntu Linux
- Enable teams of developers to simultaneously access DGX-2
- Flexibly allocate GPU resources to each user and their experiments
- Full GPU's and NVSwitch access within VMs — either all GPU's or as few as 1

DGX-2 KVM provides secure Multi-tenancy

Run **many** popular deep-learning frameworks - GPU-tuned by NVIDIA

Pre-configured templates for various sized GPU tenants

Optimized drivers and libraries for maximized multi-GPU performance

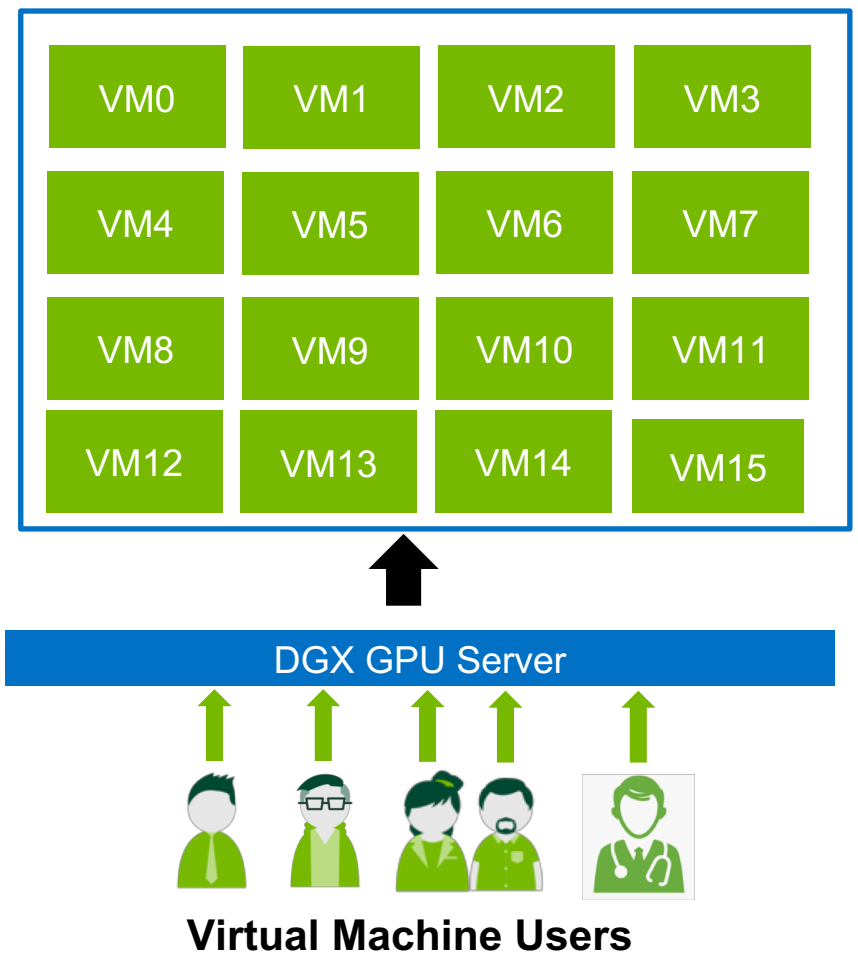


Easy VM lifecycle management with NVIDIA tools

Clean, minimal O/S Guest OS image

Non-disruptive updates for software and security

Many Users For Improved Utilization



Multiple Users on **one** DGX Server

- 16 x1-GPU VMs
- Many non-GPU VMs

Schedule Multi-DL apps on available HW

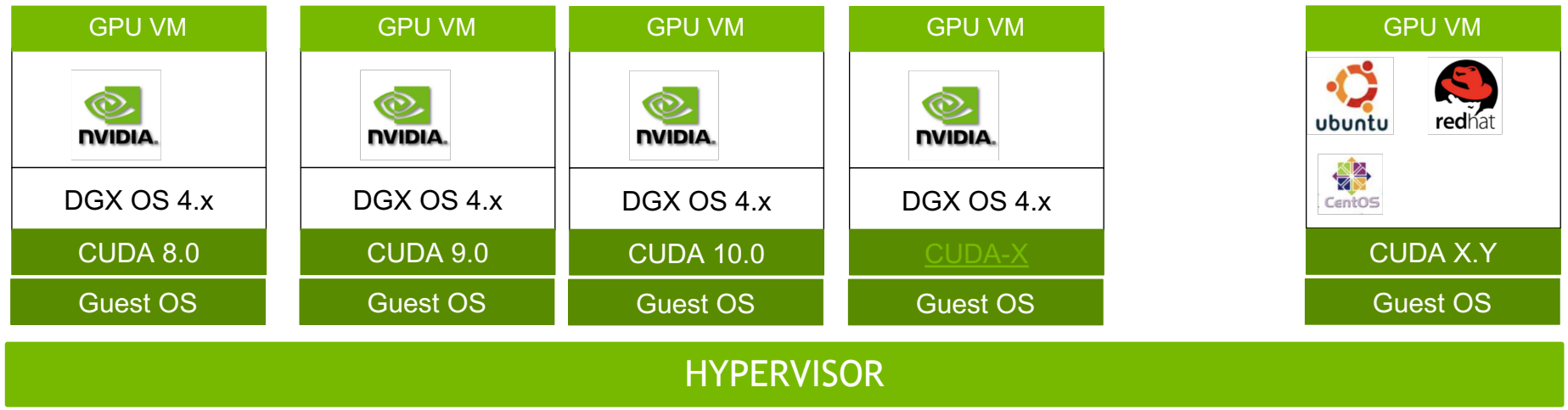
ML/DL training, NGC containers, RAPIDS

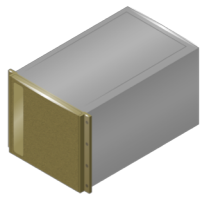
Healthcare, Higher Education

Cloud / Datacenters deployment

Ex: Data Science team within Fortune 500 organization

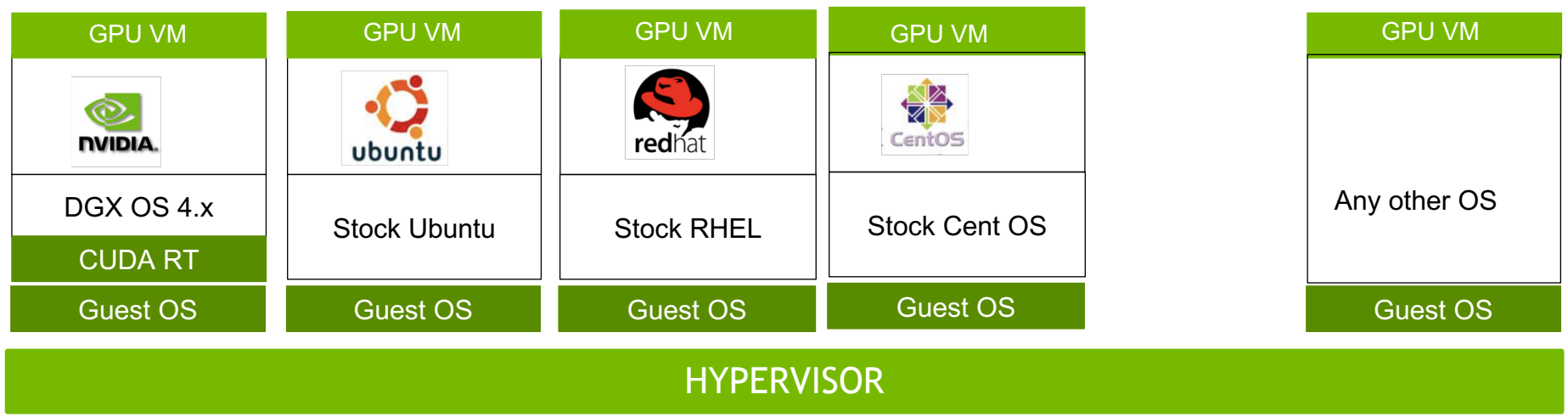
GPU VMs Can Run Different CUDA & Drivers

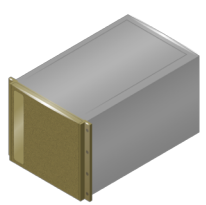




Good for: Developing apps across CUDA

GPU VM Can Have Different Guest OS



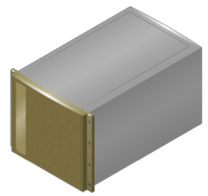


Good for: Developing apps across OSes

GPU VMs Can Run Different Apps : DL & Data Analytics

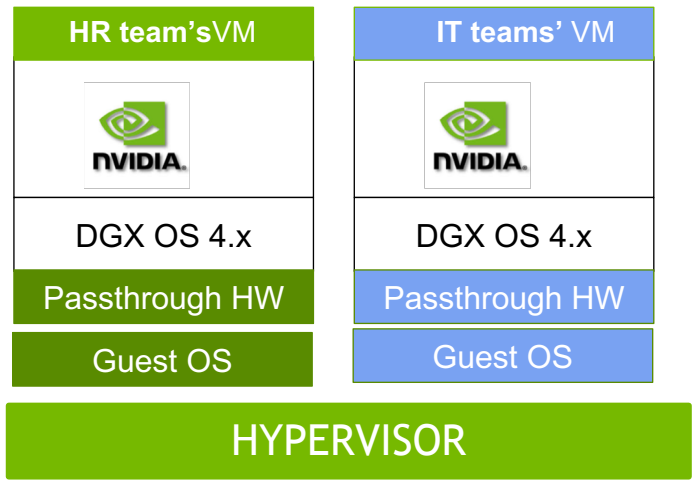
GPU VM	GPU VM	GPU VM	GPU VM	GPU VM
TensorFlow	Microsoft Cognitive Toolkit	Caffe2	mxnet	RAPIDS, .edu Healthcare, Other apps
TF Tuned SW	CNTK Tuned SW	Caffe2 Tuned SW	Pytorch Tuned SW	Tuned SW
CUDA RT	CUDA RT	**CUDA RT**	**CUDA RT**	**CUDA RT**
Guest OS	Guest OS	Guest OS	Guest OS	Guest OS

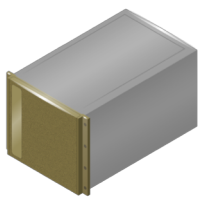
HYPERVISOR



Ex: Demo RAPIDs in a 16-GPU VM

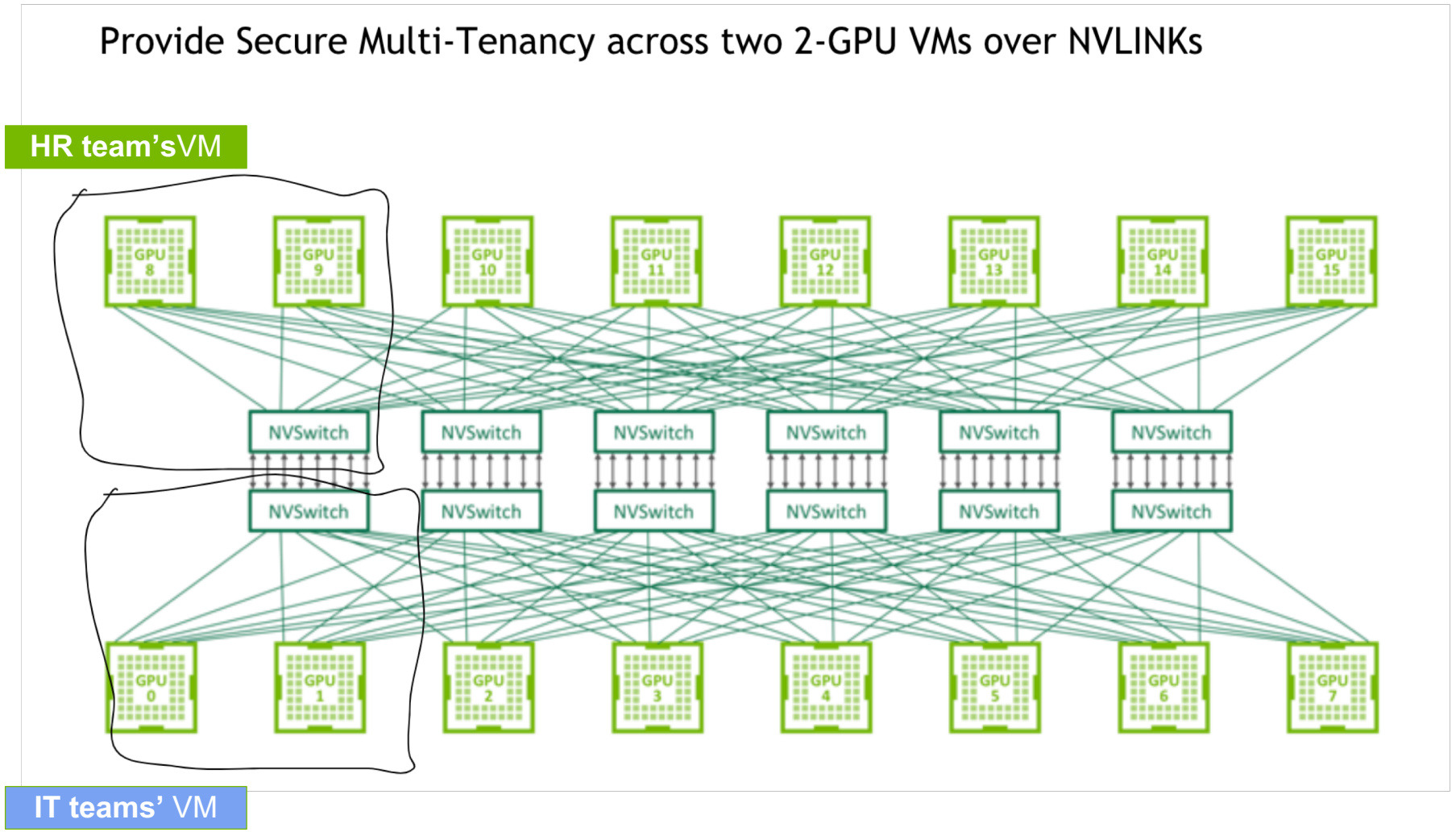
Secure Multi-tenancy Considerations





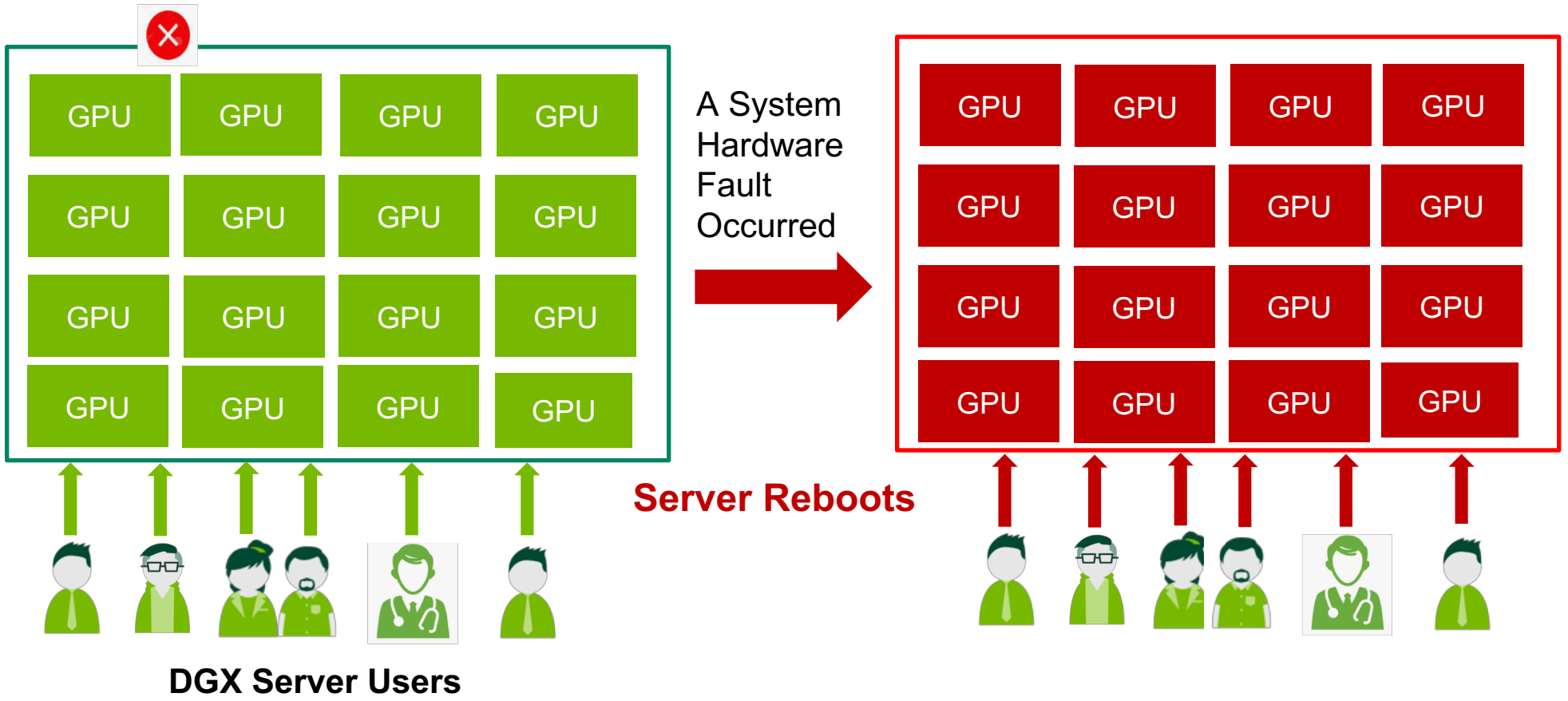
- Each tenant gets dedicated HW i.e. CPUs, Mem, GPUs etc.
- HW is isolated across tenants
- GPUs, NVSwitches are passthrough
- NVLINKs are isolated
- Cannot "reset" PCI HW
- Cannot download malicious "code" to HW

VMs Get Complete Isolation Of GPUs & NVLINKs

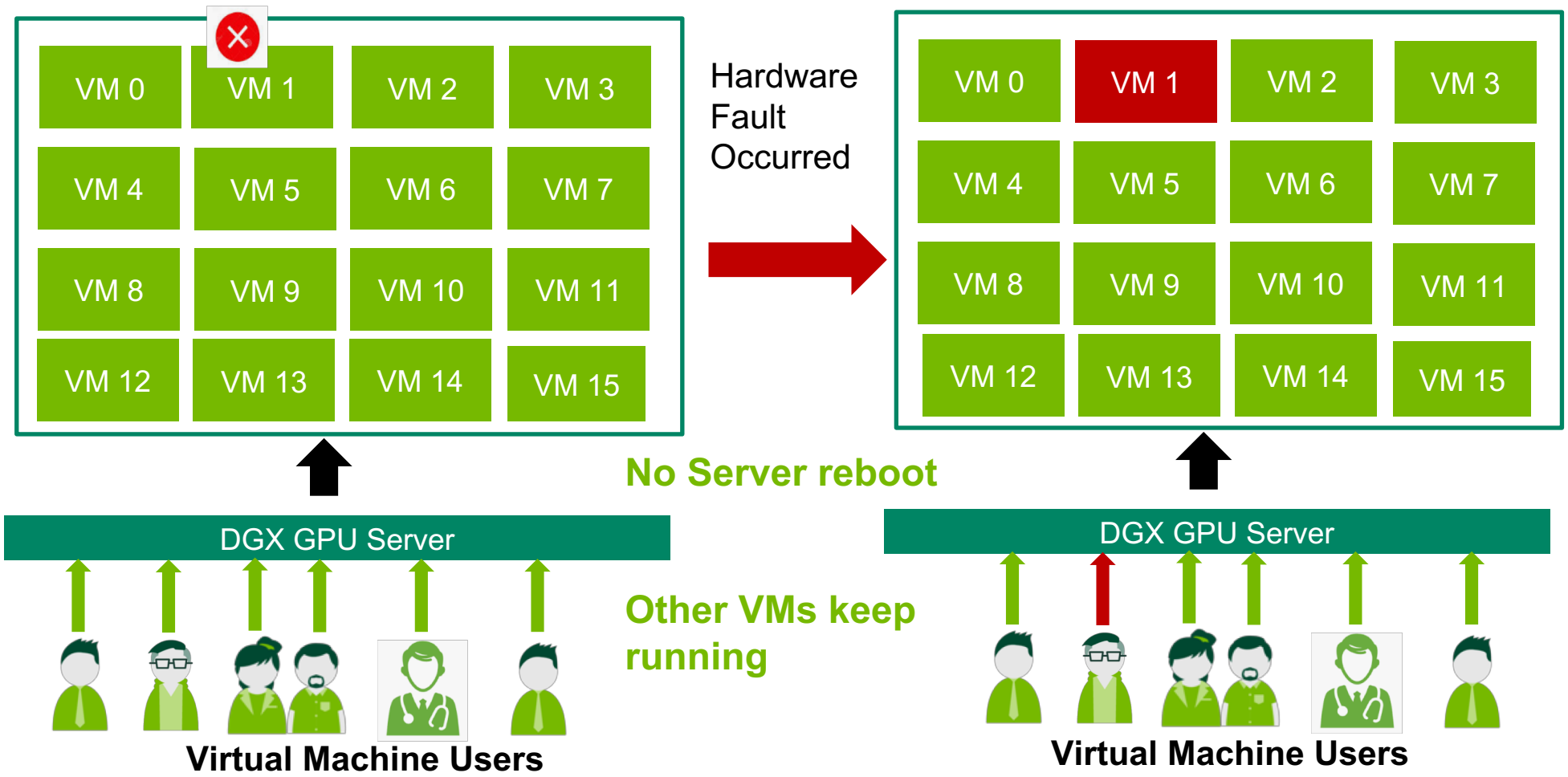


Improved System Availability

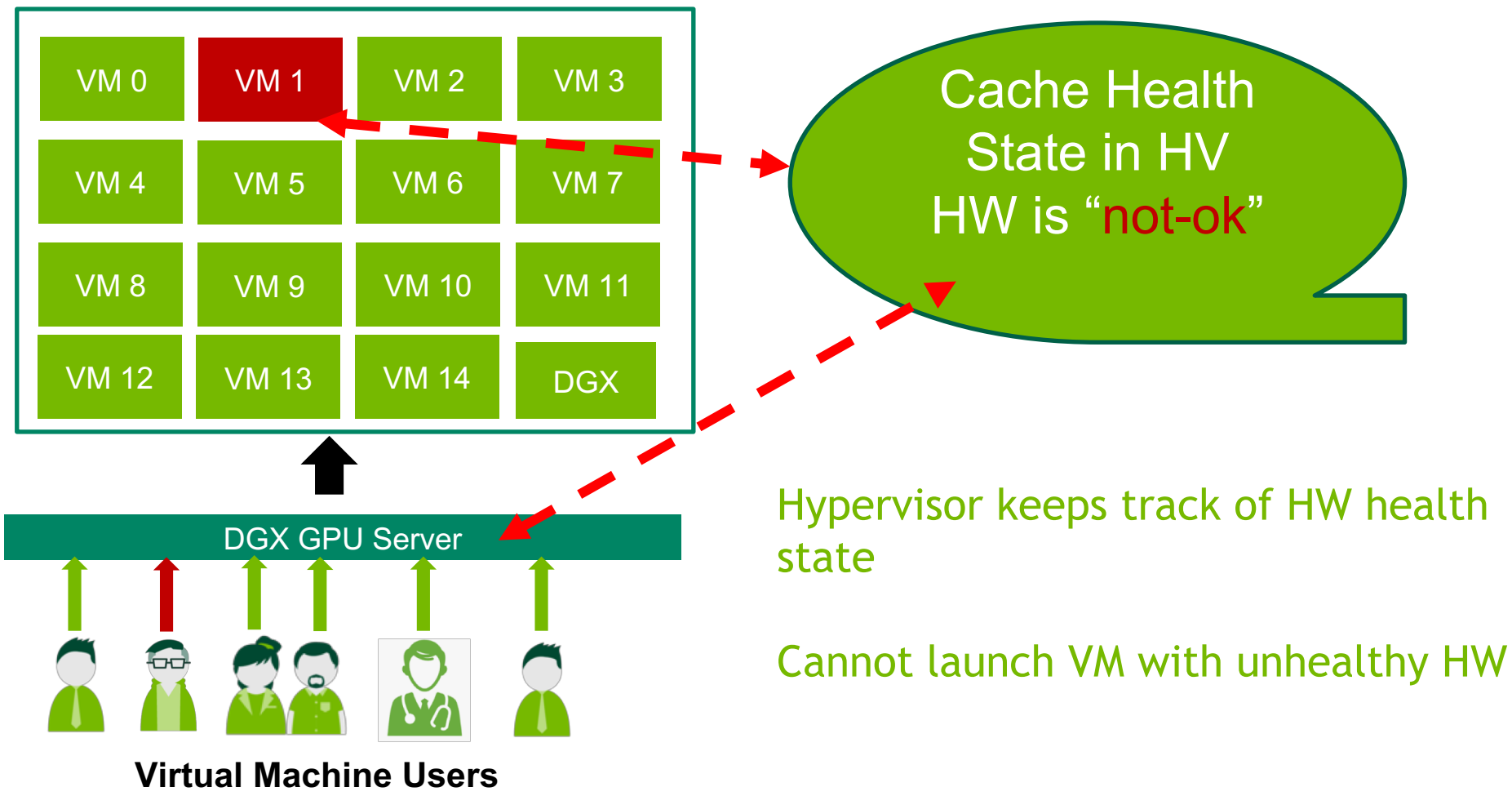
Baremetal : HW Fault Can Impact Entire Server



Fault Inside A VM Doesn't Impact Entire Server



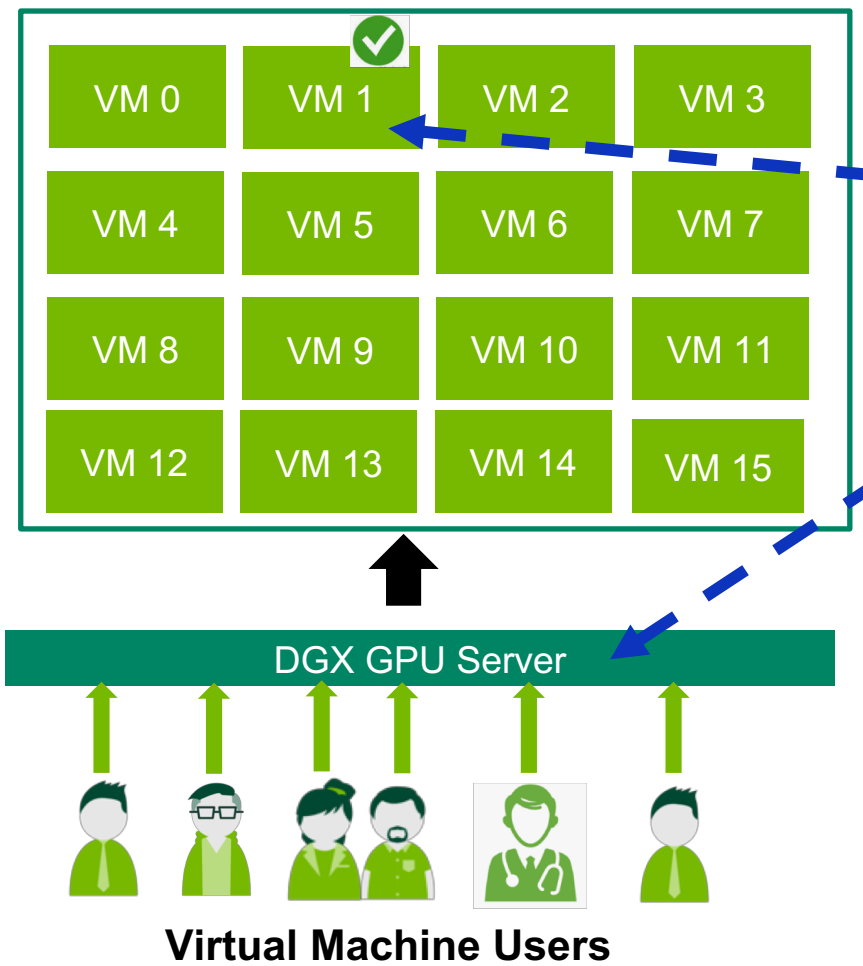
Track Unhealthy HW In Hypervisor



Hypervisor keeps track of HW health state

Cannot launch VM with unhealthy HW

One Faulted HW Is RMAed - We Recover



Cache Health State in HV HW is "**ok**"

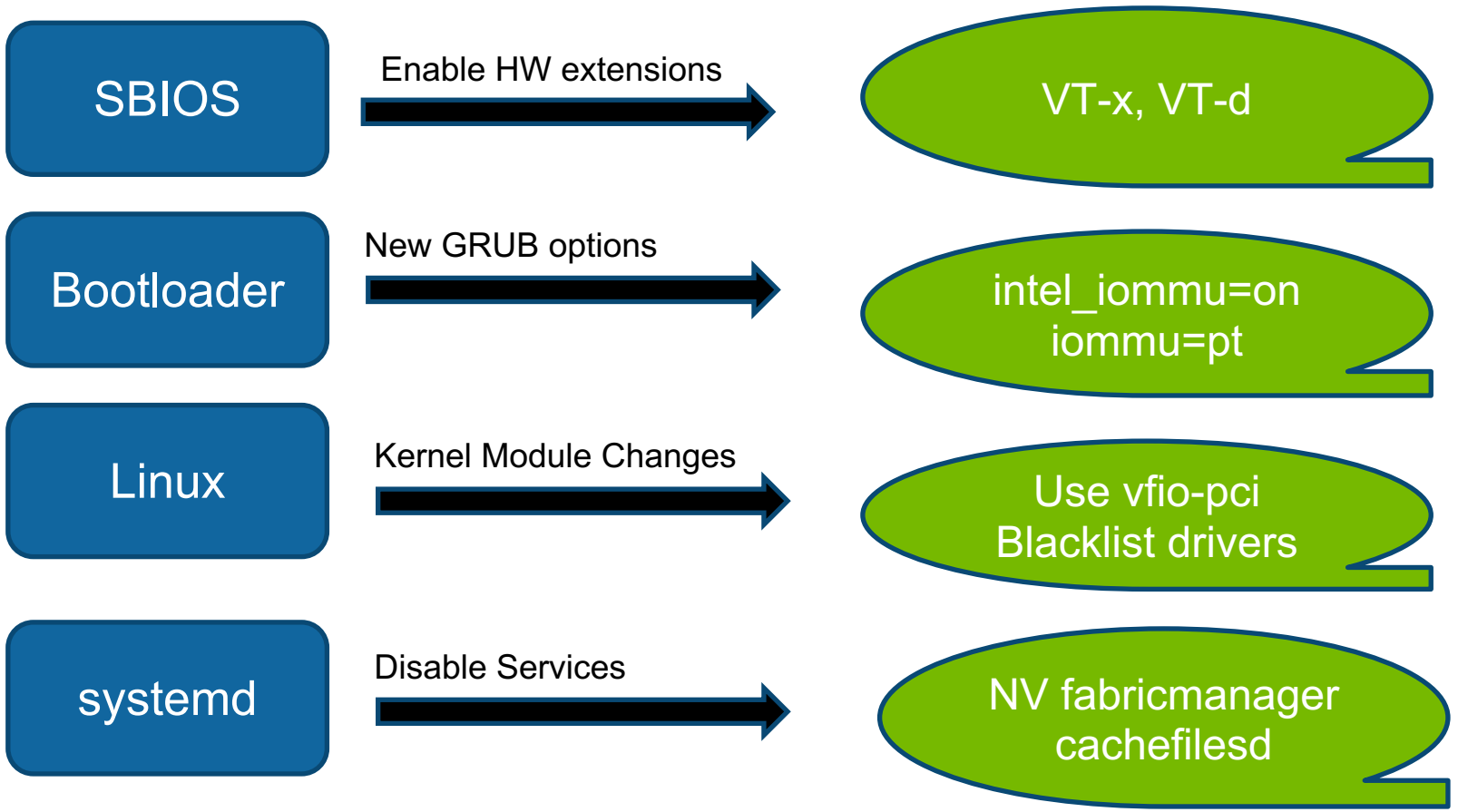
Once bad HW is replaced, our SW auto-assimilates

Hardware state is reset to "ok"

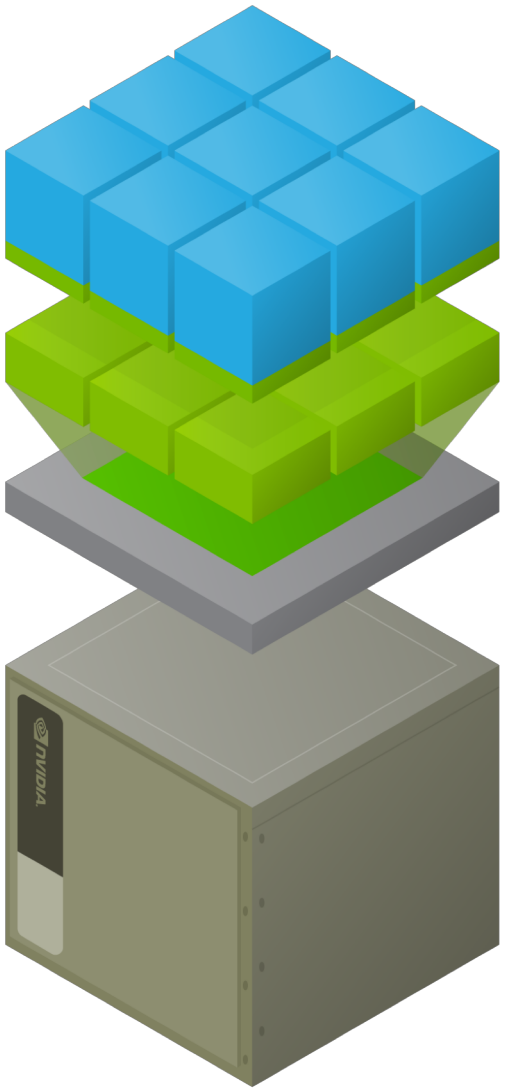
VM can be relaunched safely

What We Did To Enable KVM

Modified System SW For KVM



Modified “qemu” & Guest OS



Apps and VMs

GPUs

Hypervisor

QEMU on Server

Guest OS Support

NVDA SW stack for DL apps

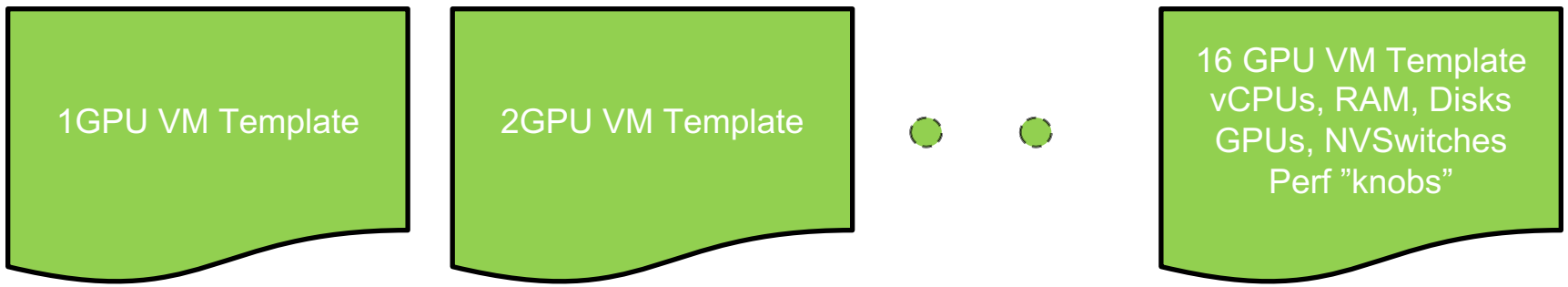
EOU tools, Telemetry, Health Mgmt, Perf

Perf knobs
VM lifecycle tool
Collect telemetry
Manage GPU Health

Make GPU VMs Secure

MMIO filtering
Isolate NVLINKs
No FW download

Pre-Configured Templates For Optimal Perf



Pre-Configure Templates for Optimal Performance:

- Only support power-of-2 GPU VMs
- Select IO devices Using PCIe and NVLINK topology
- Select CPU cores and Memory based on NUMA
- Pin vCPU cores with Core-affinity for Hyper Threads

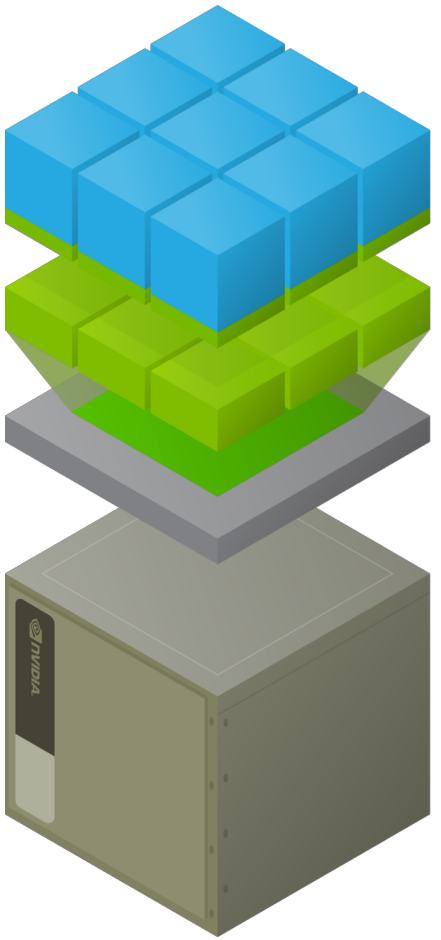
NOTE : Pre-Configured Templates can be easily modified

How To Use KVM

One Command To Change to KVM “Mode”

Bare-metal Mode

apt install dgx-kvm-sw



KVM Mode

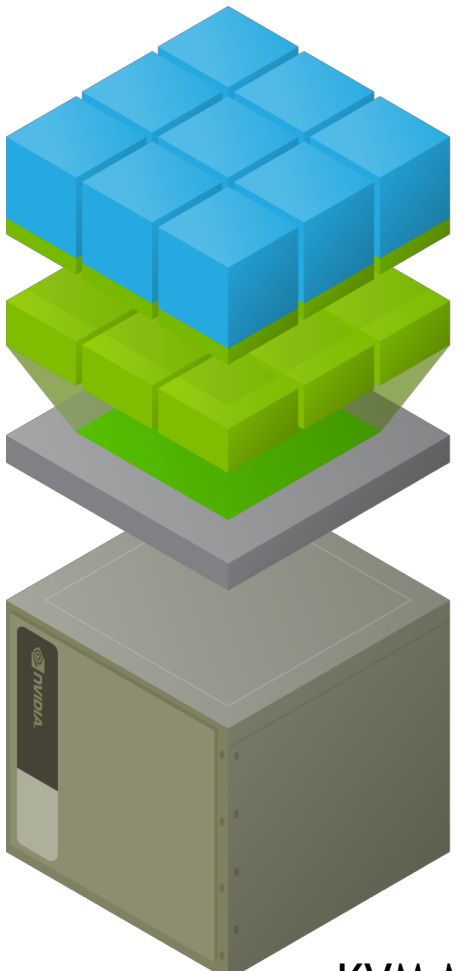
Apps & VMs

GPUs

Hypervisor

Server

Creating & Deleting GPU VMs Is Easy



KVM Mode

nvidia-vm create –gpu-count 1

nvidia-vm delete <vm>

VM0	VM1	VM2	VM3
VM4	VM5	VM6	VM7
VM8	VM9	VM10	VM11
VM12	VM13	VM14	VM15

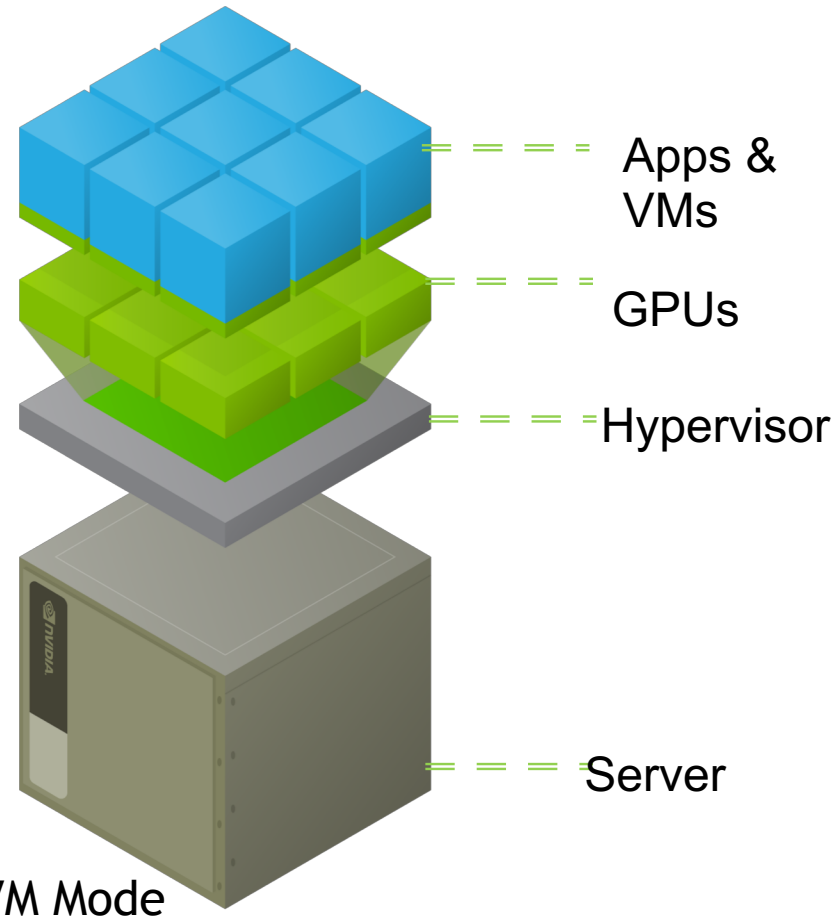
DGX GPU Server

Virtual Machine Users

One Command To Revert To Bare-metal Mode

Bare-metal Mode

apt remove dgx-kvm-sw



KVM Mode

Day Use as KVM Server, Night for HPC workloads

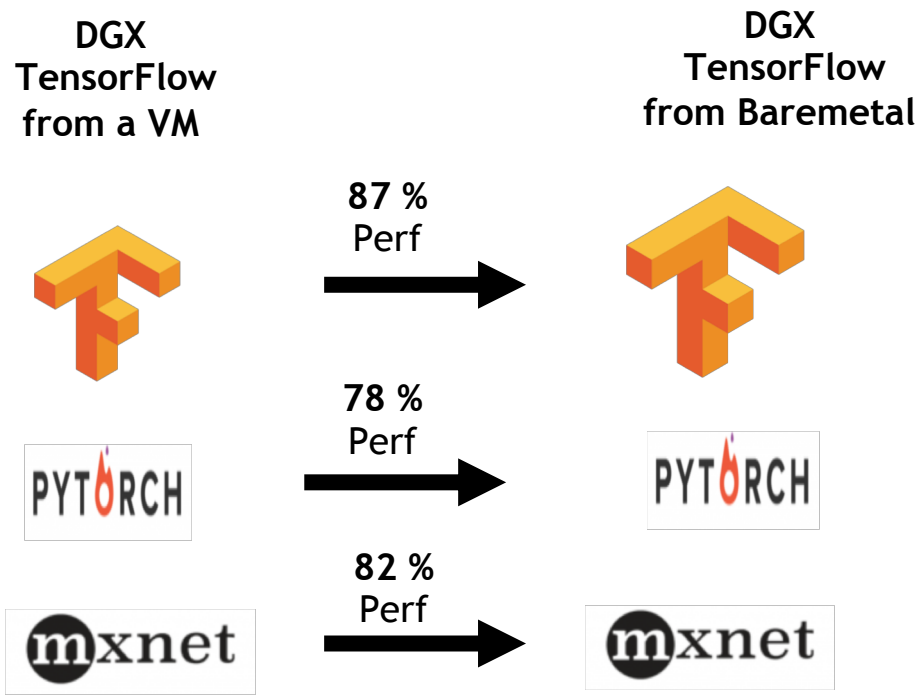
Near Bare-Metal Performance

Apply Performance “Knobs”

1. Leverage HW topology to get near Bare-metal Perf
   - NVLINK & PCI topology to select GPUs & NVSwitches
   - NUMA to select Memory and CPU cores
2. Multiple queues for Network and Block Devices

Before Performance “Knobs”

Typically Virtualization has ~ 20-30% performance overhead*

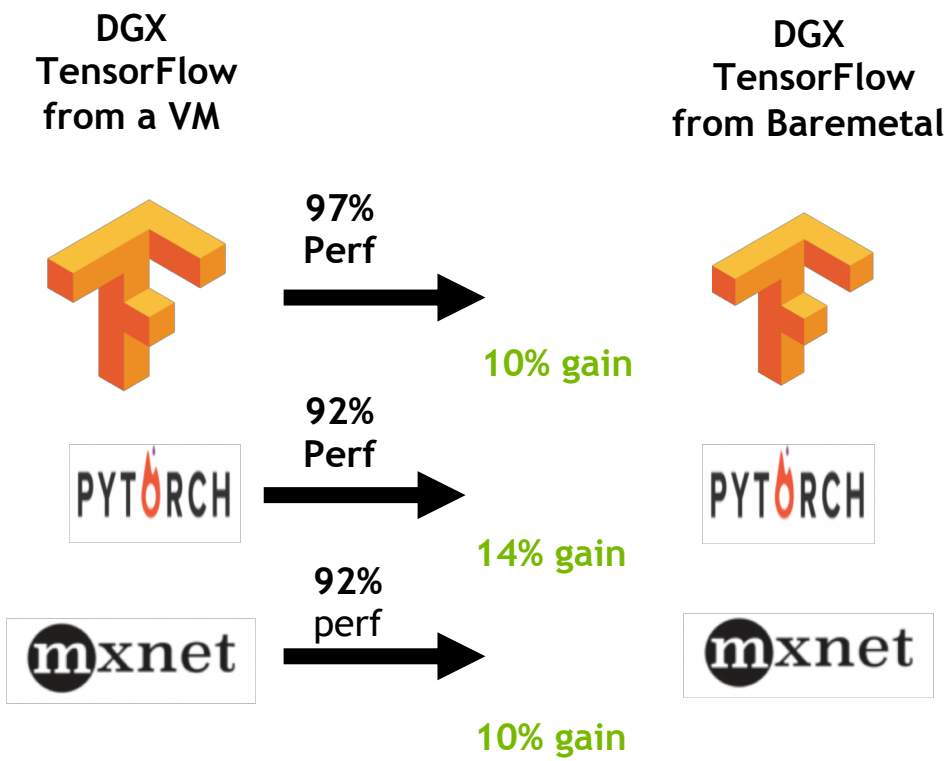


ResNet152, Batch 16 Training

* See OpenBenchMarking.org: KVM vs Baremetal Benchmarks

After Performance “Knobs”

DGX performance tuned Virtual Machines show minimal impact



ResNet152, Batch 16 Training

Where To Get KVM Software

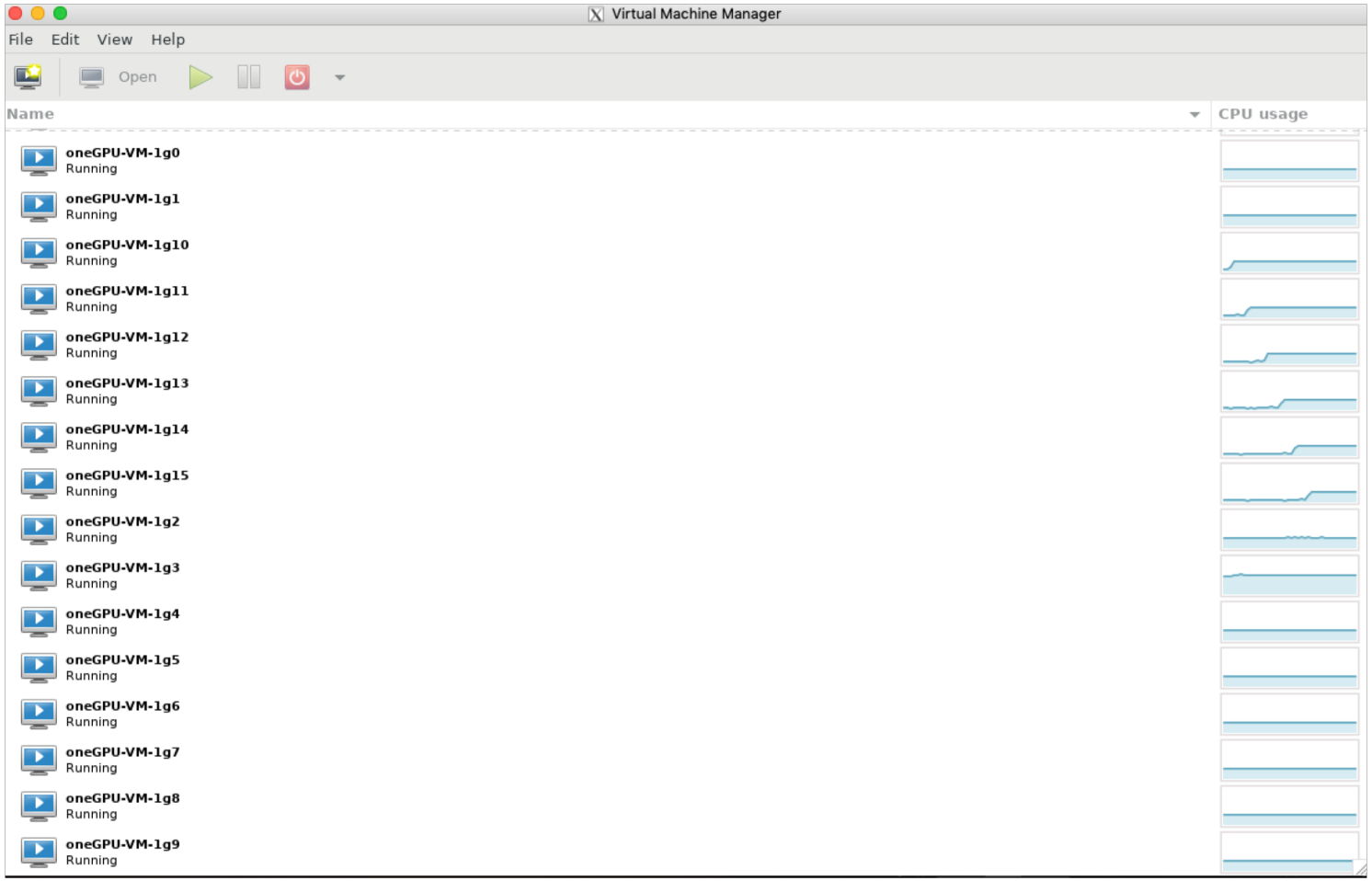
1. By default, not installed
2. Download repository pre-configured on DGX
3. Need to explicitly install
    1. dgx-kvm-sw package
    2. Guest OS images available as separate package
    3. Multiple Guest OS versions available

4. Air Gapped Customers : Reach NVIDIA Enterprise Experience

DGX-2 KVM Demo

- See DGX2 KVM Demo Page

Can Use OSS Tools To Monitor VMs

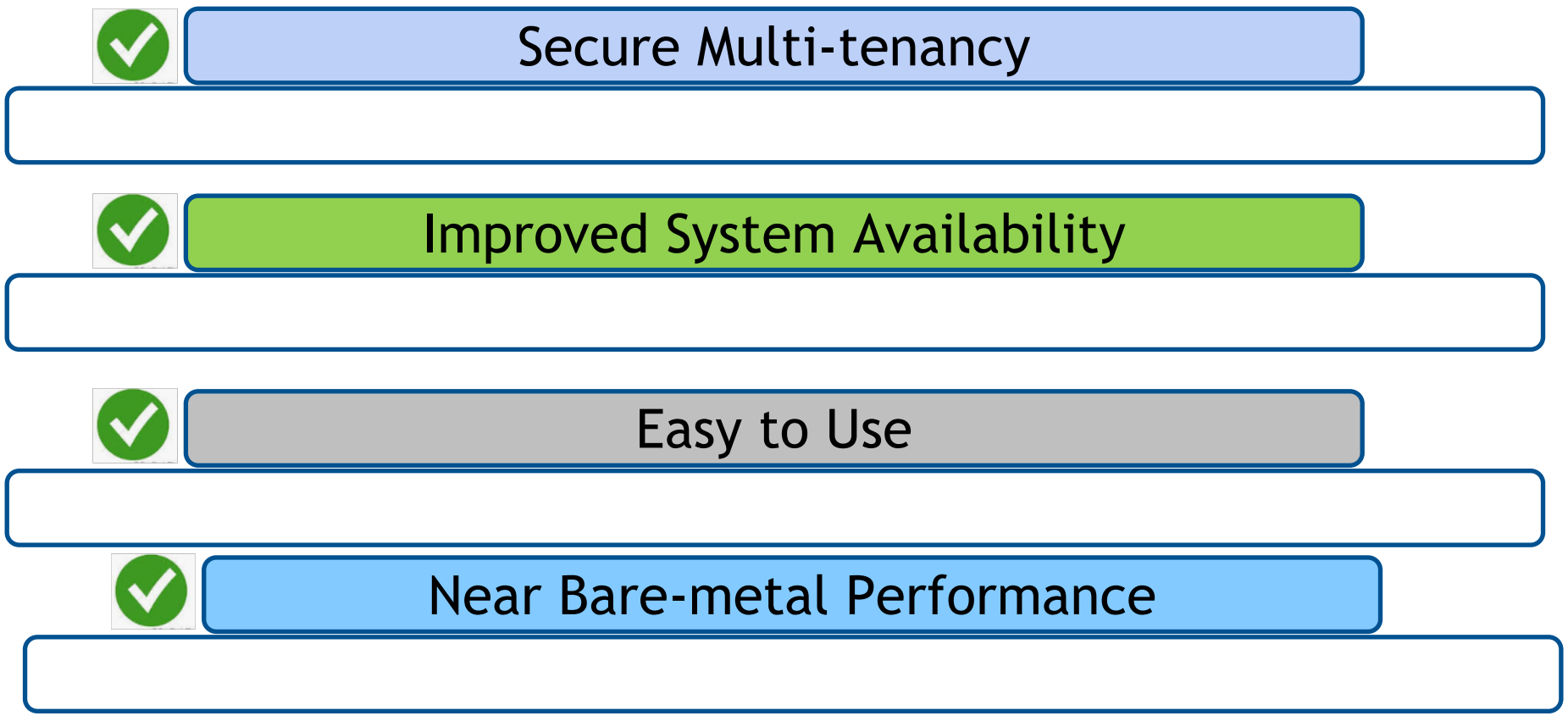


https://github.com/dholt/openstack-lab

References

- DGX2 User Guide
  - https://docs.nvidia.com/dgx/pdf/dgx2-user-guide.pdf
  - See chapter 11
- DGX Best Practices
  - https://docs.nvidia.com/dgx/bp-dgx/index.html#topic_2
  - See Chapters 10, 11
- KVM Questionnaire: http://tinyurl.com/y4d3y6xm

SUMMARY


Secure Multi-tenancy
Improved System Availability
Easy to Use
Near Bare-metal Performance

DGX-2 KVM Team

Engineering

Anish Gupta

Govinda Tatti

Peter Bradstreet

Ranen Chatterjee

Management
Prod Mktg

Varinder Singh

Raaghav Hebbar

Newton Liu

Haiduong Vo

NVIDIA LED DGX SESSIONS AT GTC 2019

NOTE: For details on all DGX-related sessions, visit: GTC site and search for "DGX" or look-up session ID

NVIDIA LED SESSIONS			
Session #, Date/Time	**Location**	**Session Name**	**Product Featured**
S91003 Wed 3/20, 2pm	Convention Center Room 210A	**MXNet Computer Vision and Natural Language Processing Models Accelerated with NVIDA TensorCores** - Przemyslaw Tredak (DevTech Engineer) and Cyrus Vahid (Principle Evangelist AWS Deep Engine)	
S9417 Wed 3/20, 3pm	SJ Convention Center Room 211B	**Molecular Generative VAEs: Parallelization, Optimization, and Latent Space Analysis on DGX-1** - Ellen Du and Joey Storer, Research Scientists, Dow Chemical Company	
S9469 Wed 3/20, 4pm	SJ Convention Center Room 231	**MATLAB and NVIDIA Docker: A Complete AI Solution, Where You Need It, in an Instant** - Jos Martin and Joss Knight, Engineering, MathWorks	
S9892 Wed 3/20, 4pm	SJ Convention Center Room 220A	**Deep Learning for Autonomous Driving at BMW** **-** Alexander Frickenstein, PhD Candidate, BMW	
CE9153 Wed 3/20, 4pm	SJ Convention Center Hall 3 Pod D	**Connect with Experts: How to effectively use GPGU VMs on DGX-2** - Anish Gupta, Varinder Singh, Raaghav Hebbar, Ranen C and Chris Zankel (Nvidia)	

Thank You !!