cloud STACK

INDIA USER GROUP MEETUP

CLOUDSTACK GPU INTEGRATION

Bringing AI, Graphics & VDI to IaaS Clouds

ROHIT YADAV

CTO, SHAPEBLUE

JULY 11, 2025 | GREATER NOIDA, INDIA

$ whoami

- Committer, PMC Member & Ex-VP Apache CloudStack
- Chair, CloudStack India User Group
- ASF Member & Kubernetes SIG Member
- CTO, ShapeBlue
- Tinkerer, Homelabber, Renewables Champion

CloudStack & Legacy GPU Support

- ExtraConfig
- KVM Groovy-based Agent Hooks
- Out-of-band configuration (hacking hypervisors)
- Old GPU/vGPU support with XenServer
  https://cwiki.apache.org/confluence/display/CLOUDSTACK/GPU+and+vGPU+support+for+CloudStack+Guest+VMs

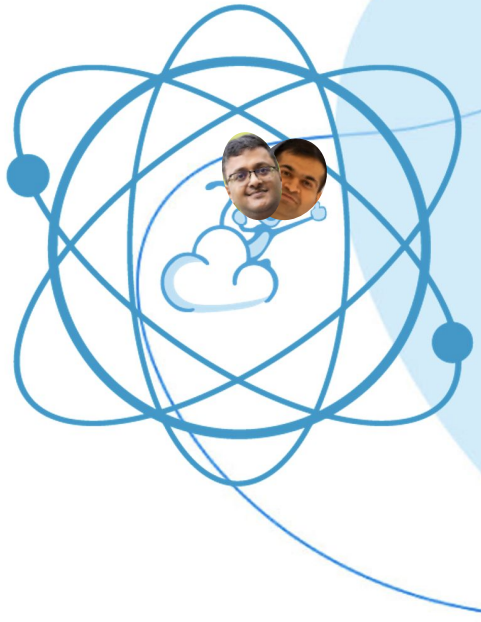
GPU Integration Proposal

Vishesh Jindal @ CCC24 Madrid

Staff Software Engineer, ShapeBlue

https://www.youtube.com/watch?v=UyJTcy69ncg

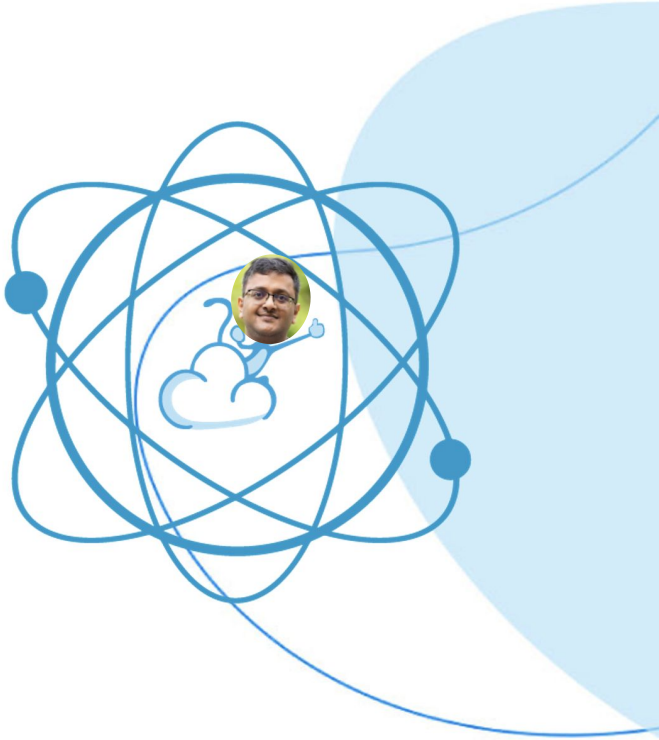
CloudStack GPU Integration
Collaborative Effort



CloudStack GPU Integration - Technical Preview

- Pull Request submitted by Vishesh: https://github.com/apache/cloudstack/pull/11143
- Feature to ship as Technical Preview in ACS 4.21**

*** depends on community acceptance*



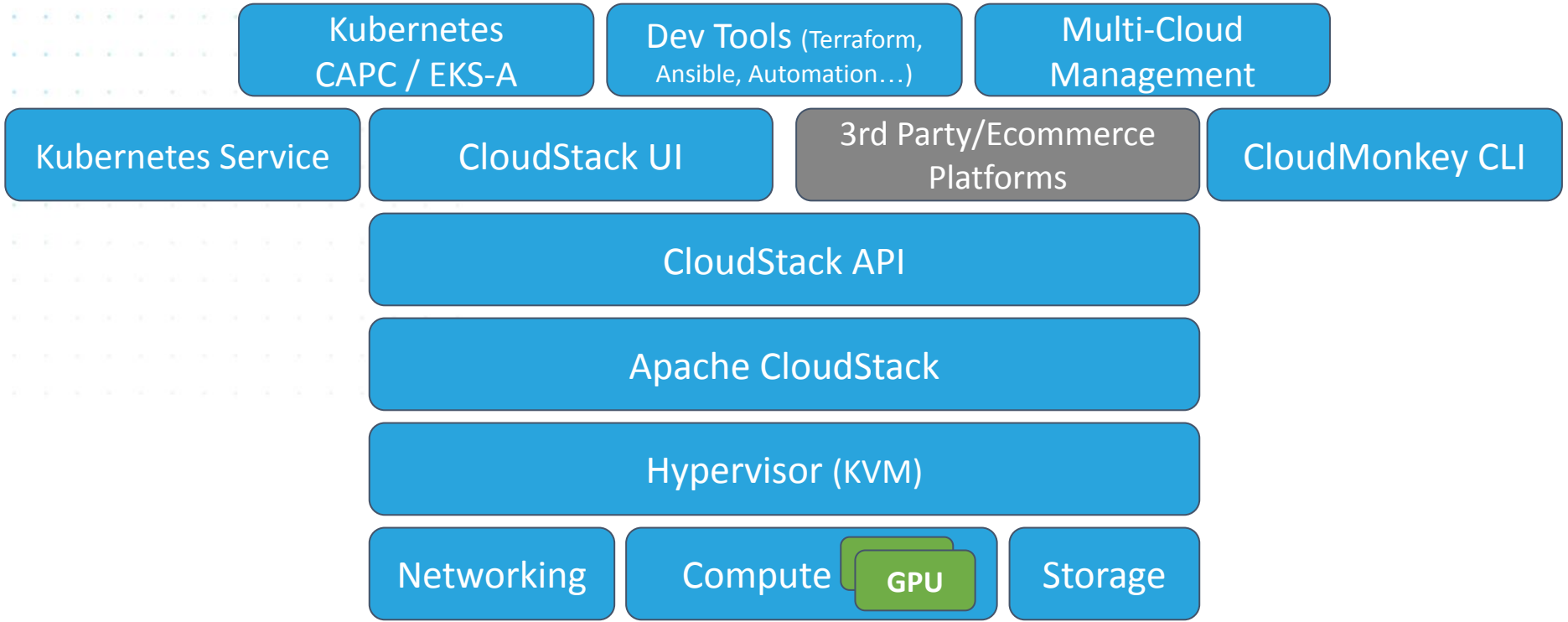
Why this feature?

Why Have this Feature?

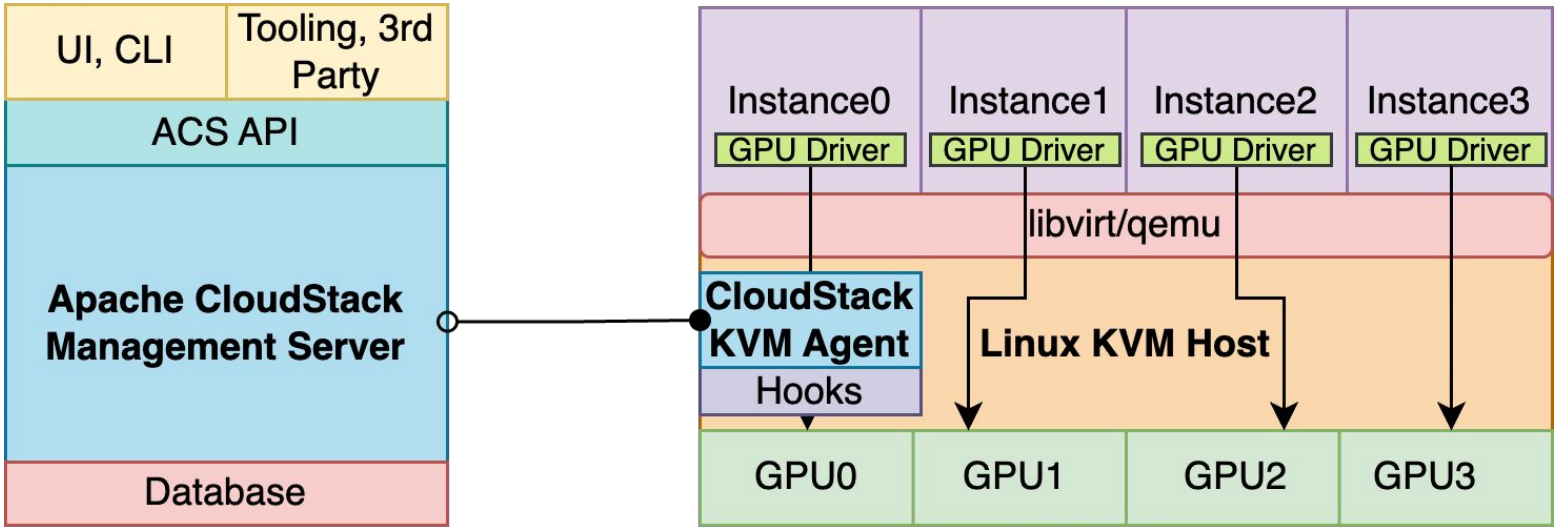
- Satisfy variety of GPU use-cases
- Enable CSPs to offer GPU-backed compute offerings
- Data Locality & Sovereignty
- Self-service access to GPUs
- Benefit end-users: AI/ML engineers, VDI users, R&D Teams

GPU-Integration Design & Architecture

Apache CloudStack Architecture

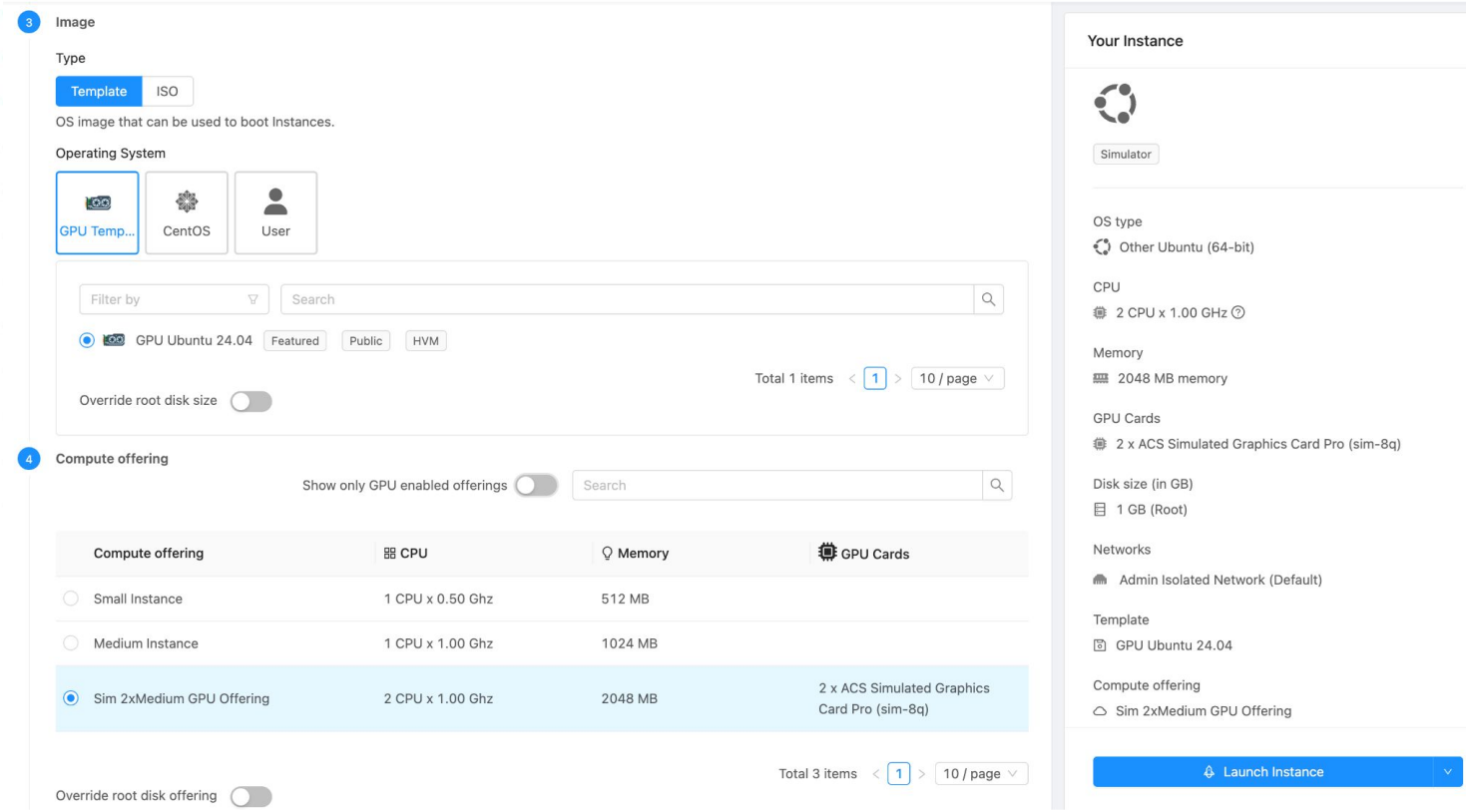


Feature Design & Architecture

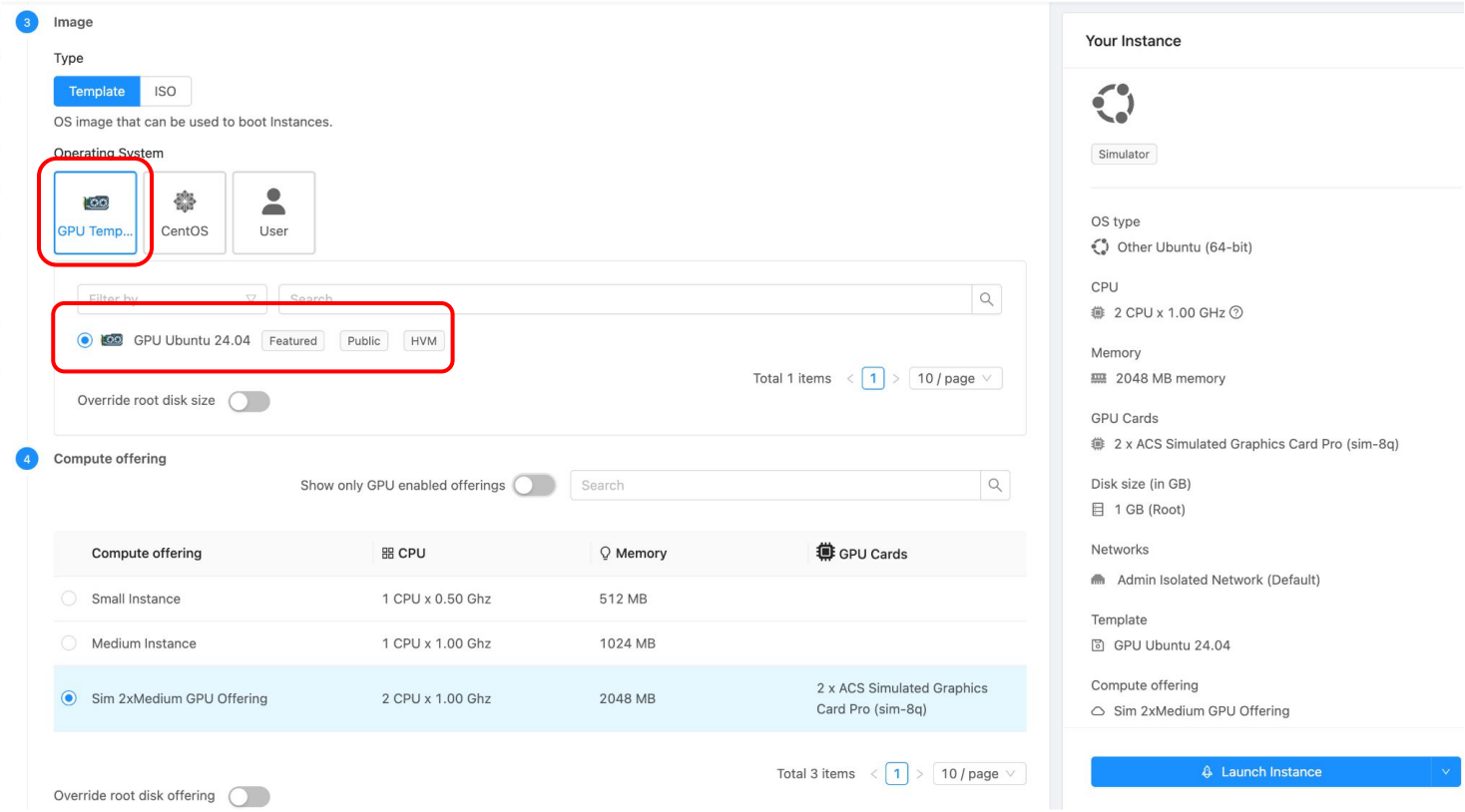


GPU-Integration Demo

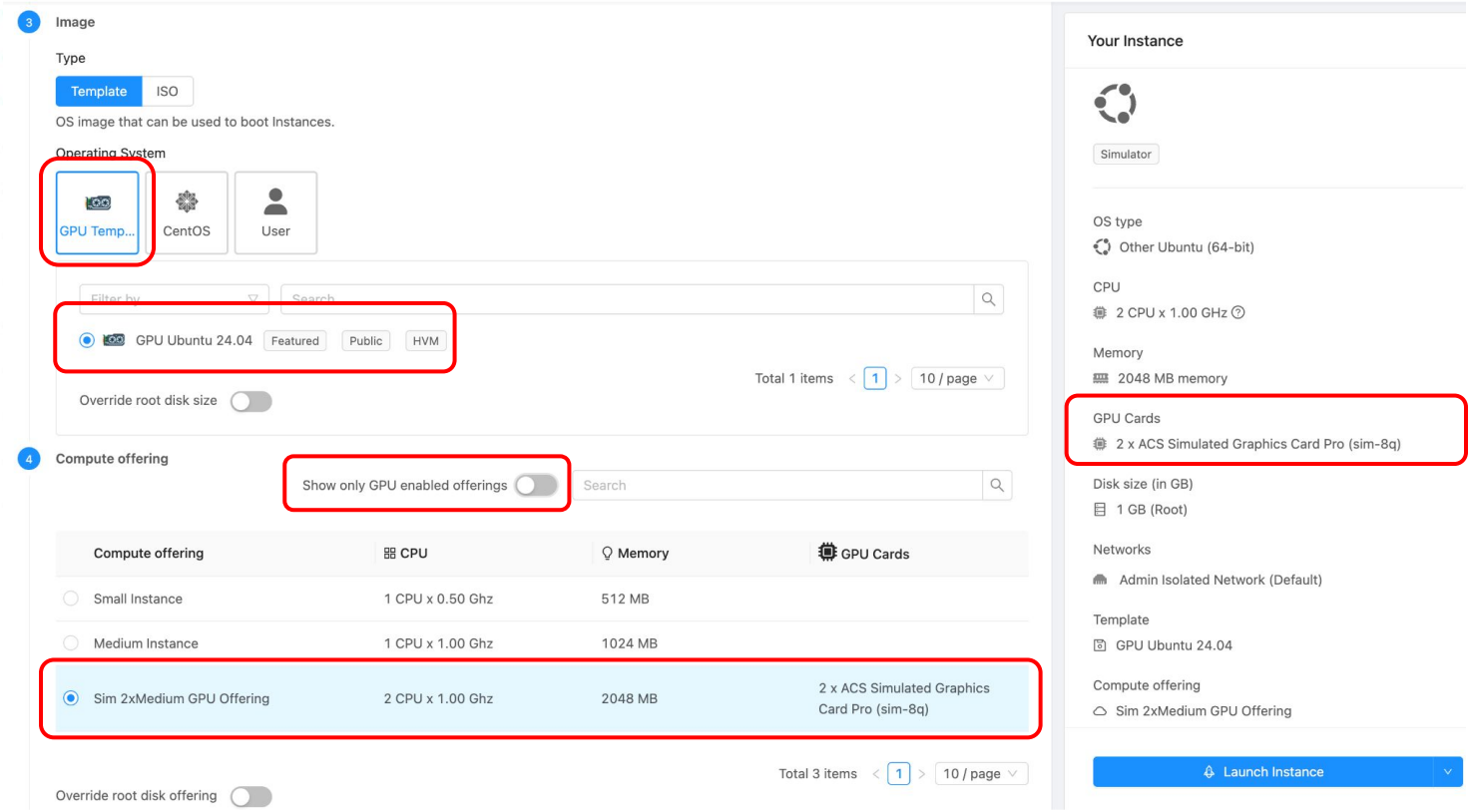
GPU-Enabled Instance Deployment



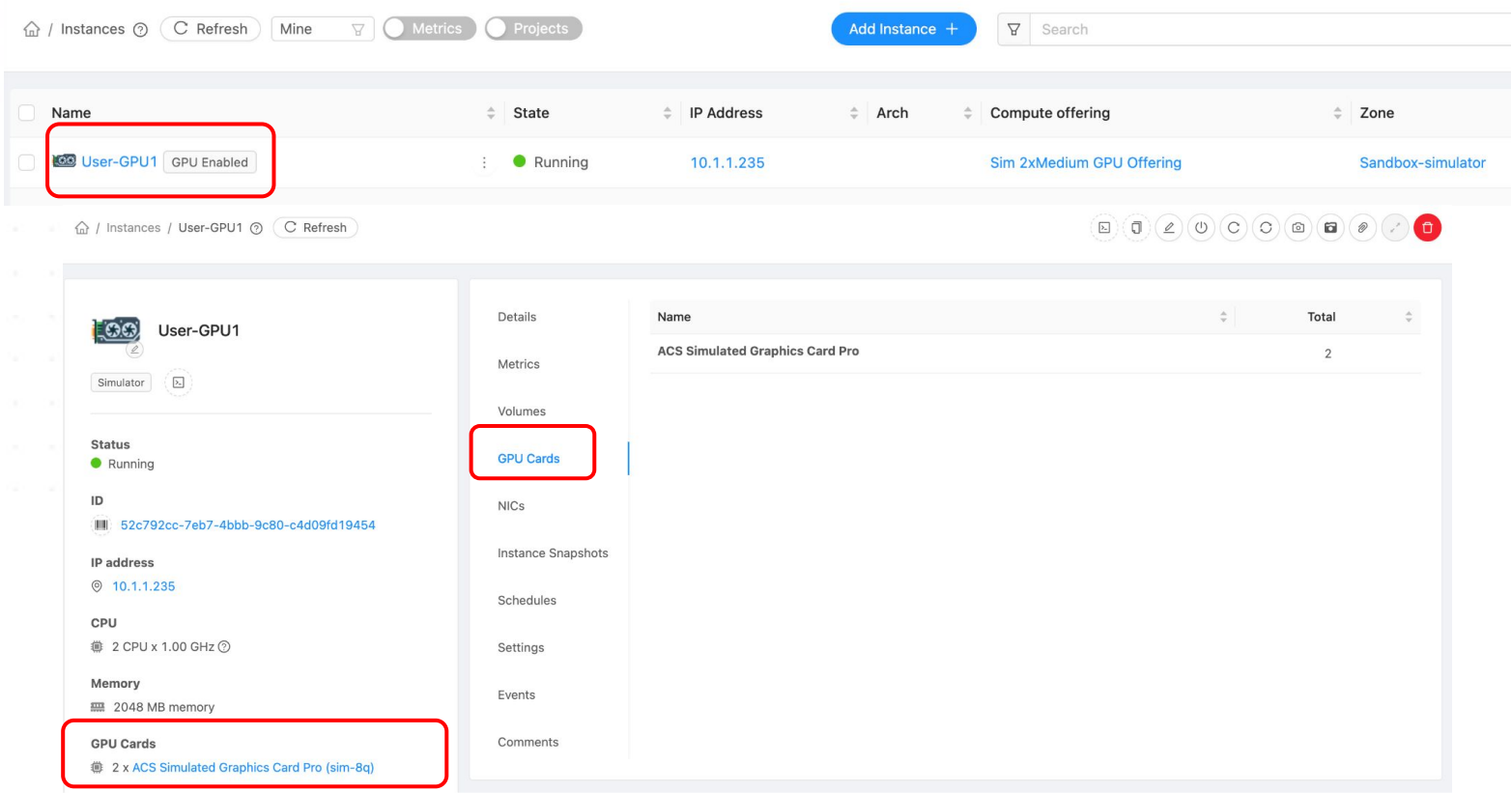
GPU-Enabled Instance Deployment



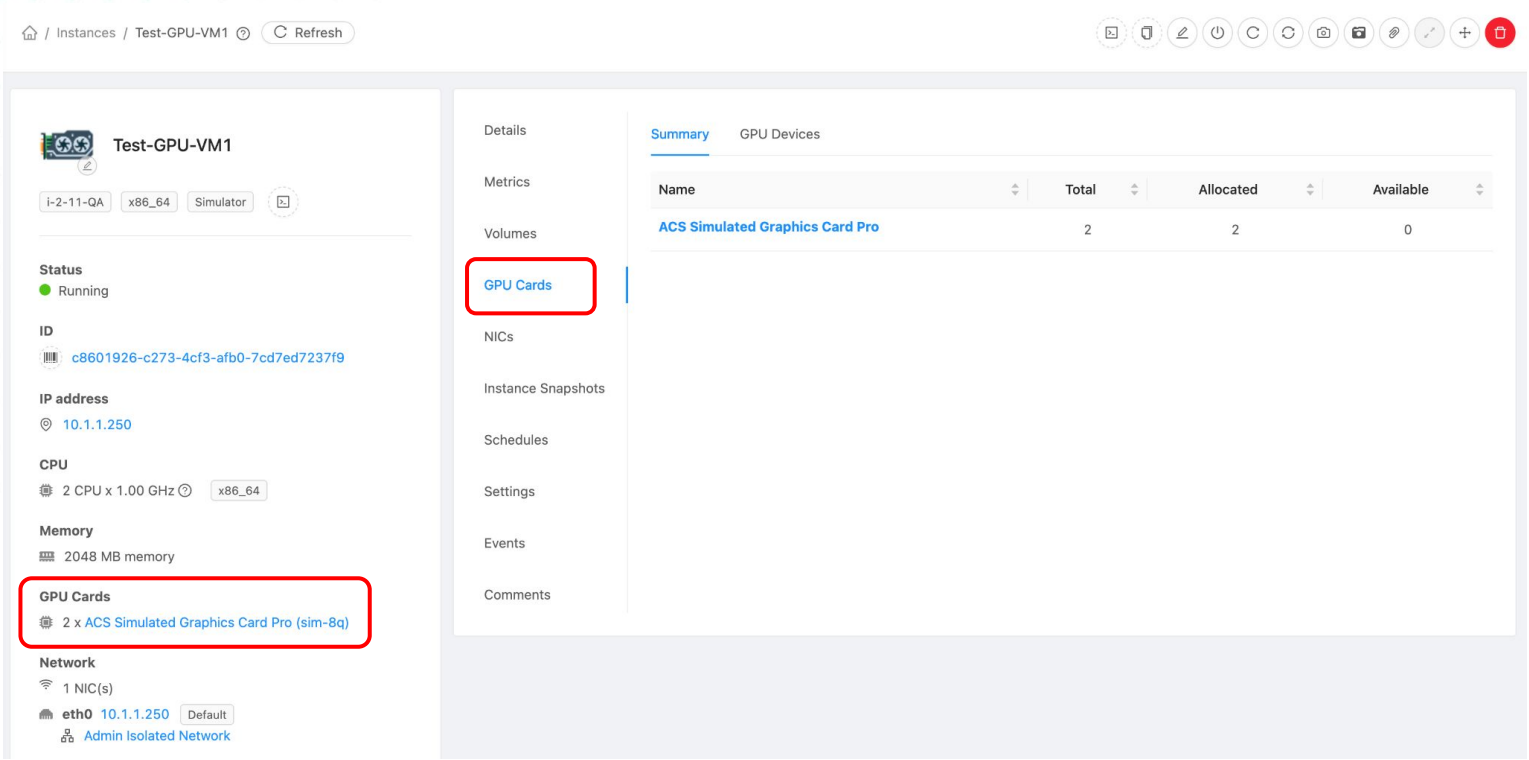
GPU-Enabled Instance Deployment



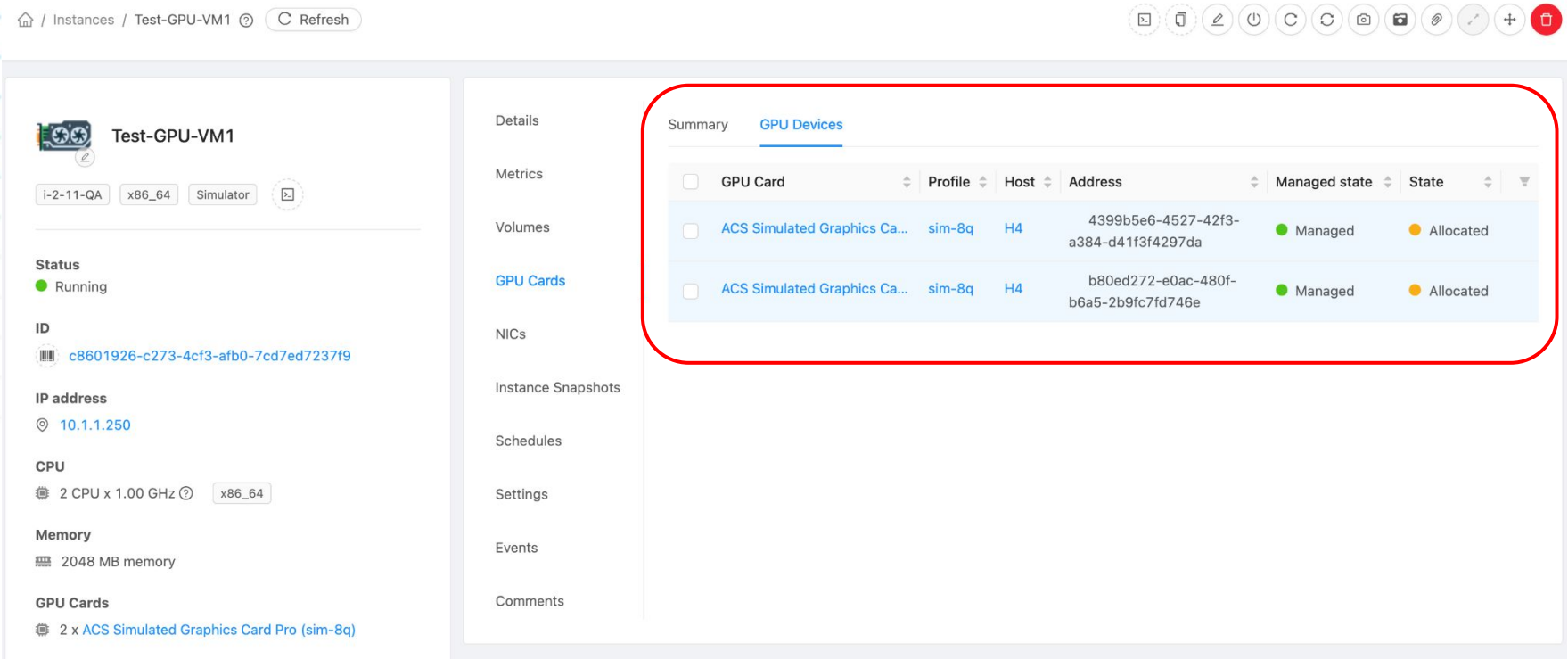
GPU-Enabled Instance (User View)



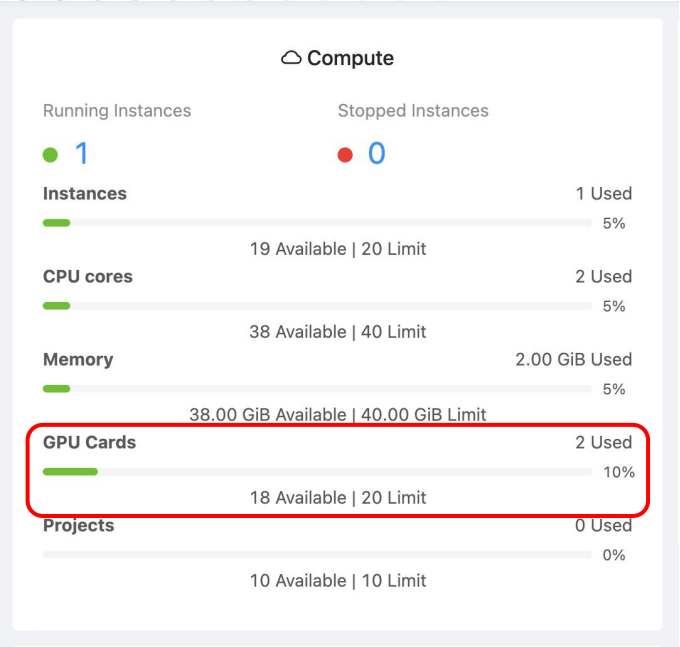
GPU-Enabled Instance (Admin View)



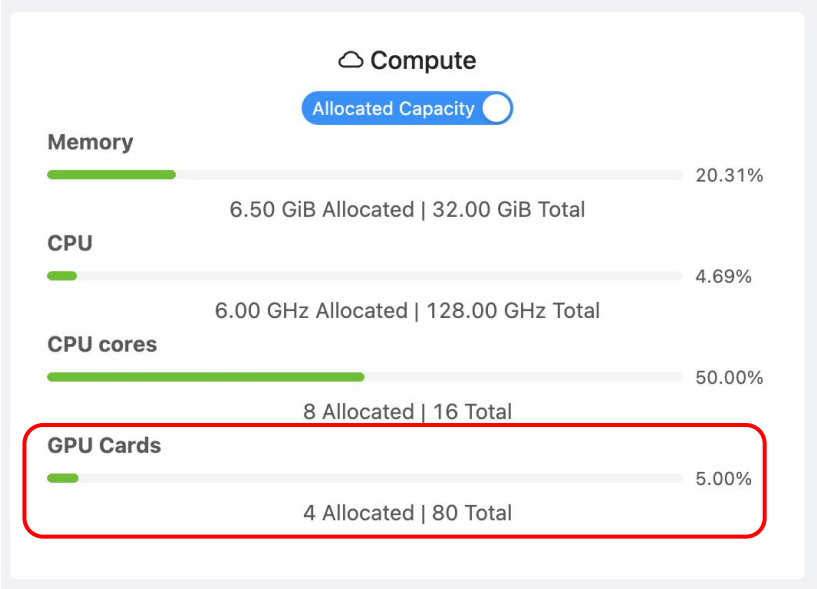
GPU-Enabled Instance (Admin View)



Dashboard GPU Allocation / Limit

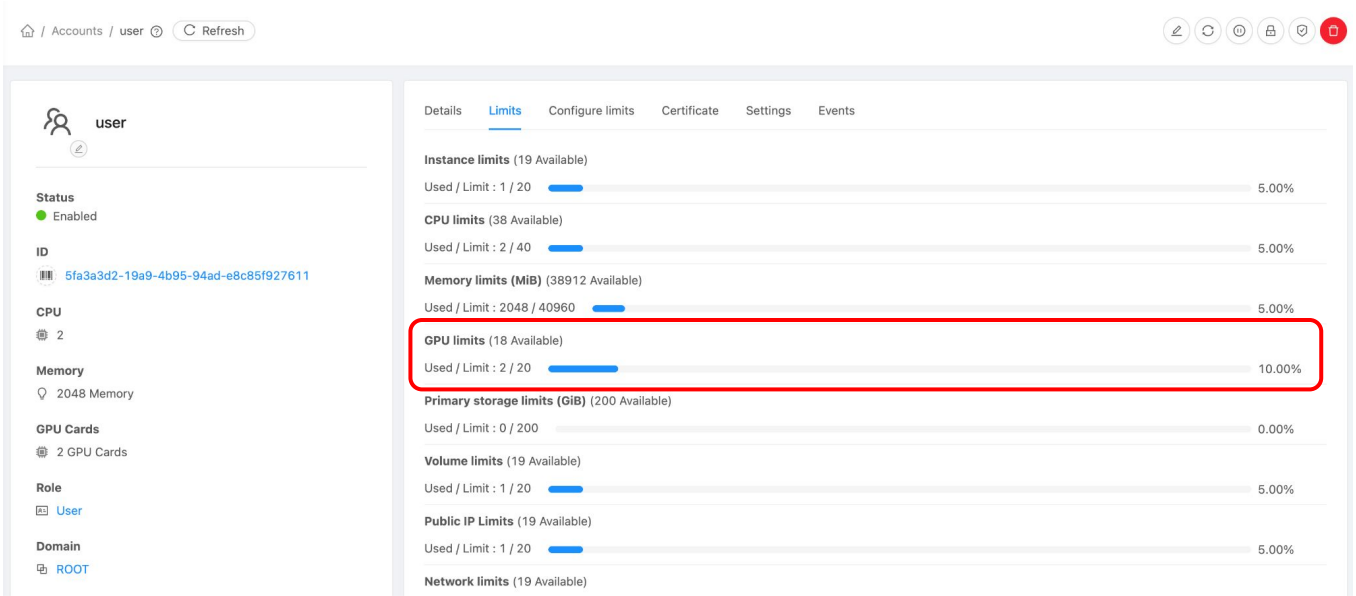


User Dashboard



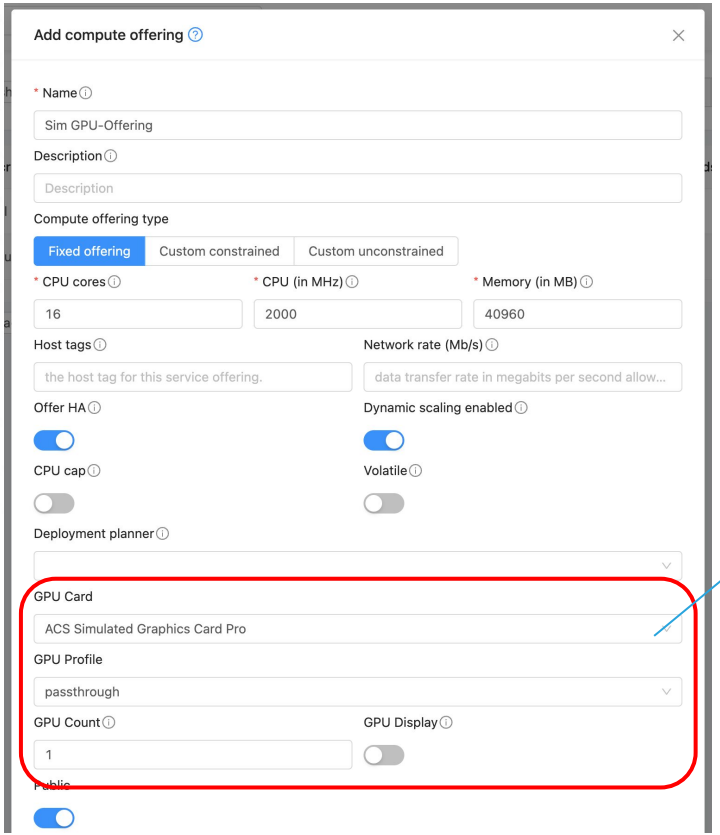
Admin Dashboard

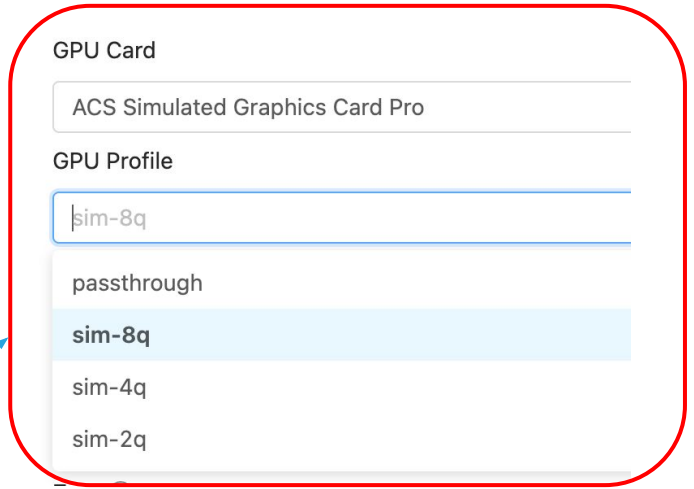
Tenant GPU Limits



- `max.account.gpus` - The default maximum number of GPU devices that can be used for an account.
- `max.domain.gpus` - The default maximum number of GPU devices that can be used for a domain.
- `max.project.gpus` - The default maximum number of GPU devices that can be used for a project.
- *`gpu.detach.on.stop`* (default: false, scope: domain): When true the GPU devices are detached from stopped instances, available for allocation to other instances.

GPU-Enabled Service Offerings





GPU-Enabled Service Offerings

/ Compute offerings | Refresh | Active | Add compute offering + | Search

Name	Description	State	CPU cores	CPU (in MHz)	Memory	GPU Cards
Small Instance	Small Instance	Active	1	500	512	
Medium Instance	Medium Instance	Active	1	1000	1024	
Sim 2xGPU Offering GPU Enabled	Sim 2xGPU Offering	Active	16	2000	40960	2x ACS Simulated Graphics Card Pro(sim-8q)

Showing 1-3 of 3 items | 1 | 20 / page

Usage Tracking: GPU-Instances

GPU-enabled instance usage for tenants is tracked (for billing and other purposes) via GPU-enabled service offerings for allocated and running VMs:

Usage Type 1 - RUNNING_VM

Usage Type 2 - ALLOCATED_VM

GPU Inventory Management

/ Hosts | Refresh | All | Metrics | Add host + | Search

Name	State	Resource state	IP Address	Arch	Hypervisor	System VMs	Version	Cluster name	Zone	Management Server
H1 GPU Enabled	Up	Enabled	172.16.15.2	x86_64	Simulator		4.21.0.0-SNAPSHOT	C0	Sandbox-simulator	ref-trl-8904-k-mol8-vishesh-jindal-mgmt1.sofia.shapeblue.com
H2 GPU Enabled	Up	Enabled	172.16.15.10	x86_64	Simulator		4.21.0.0-SNAPSHOT	C0	Sandbox-simulator	ref-trl-8904-k-mol8-vishesh-jindal-mgmt1.sofia.shapeblue.com
H3 GPU Enabled	Up	Enabled	172.16.15.23	x86_64	Simulator		4.21.0.0-SNAPSHOT	C1	Sandbox-simulator	ref-trl-8904-k-mol8-vishesh-jindal-mgmt1.sofia.shapeblue.com
H4 GPU Enabled	Up	Enabled	172.16.15.14	x86_64	Simulator		4.21.0.0-SNAPSHOT	C1	Sandbox-simulator	ref-trl-8904-k-mol8-vishesh-jindal-mgmt1.sofia.shapeblue.com

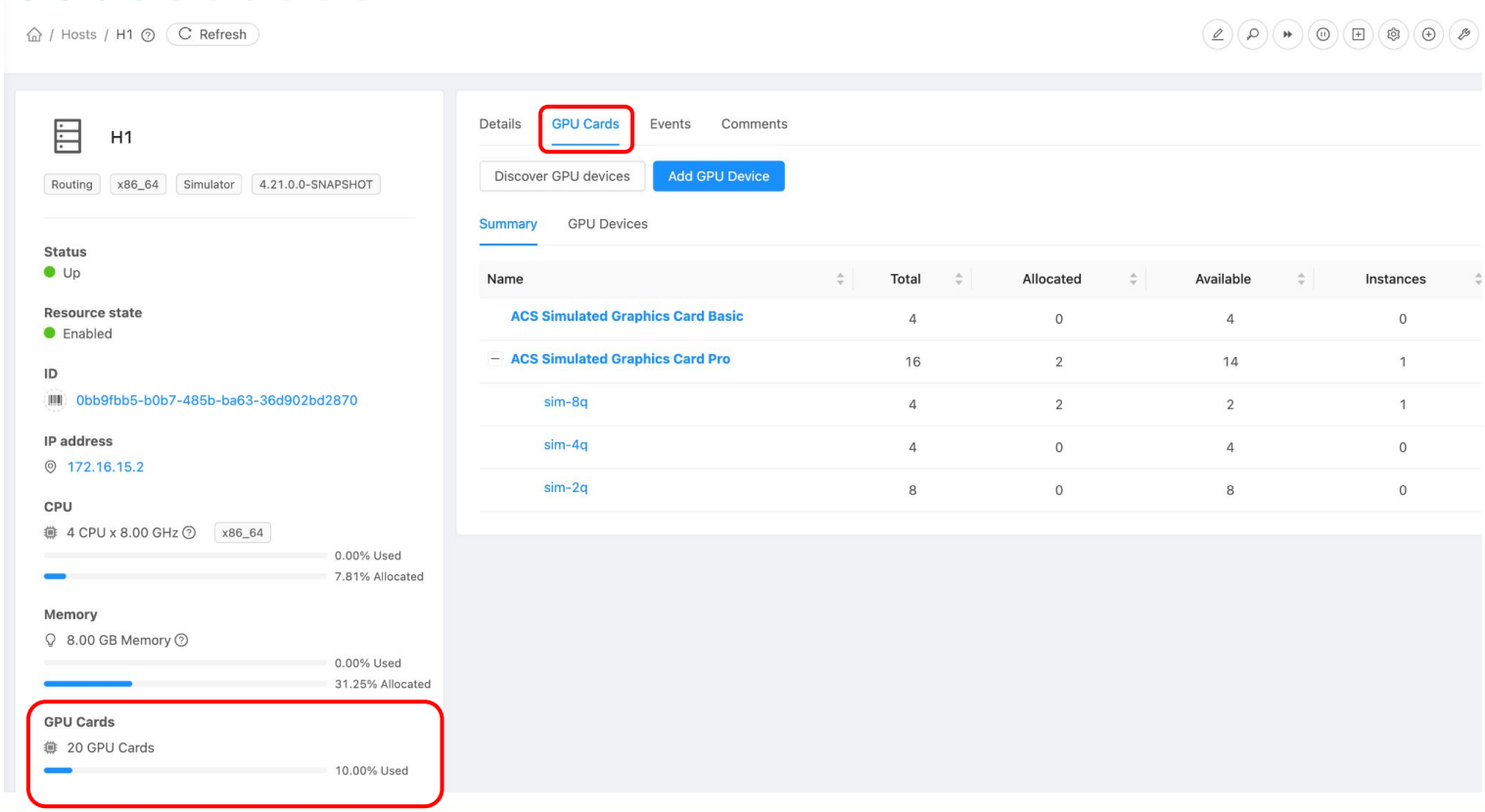
Showing 1-4 of 4 items | 1 | 20 / page

Host BIOS Setup

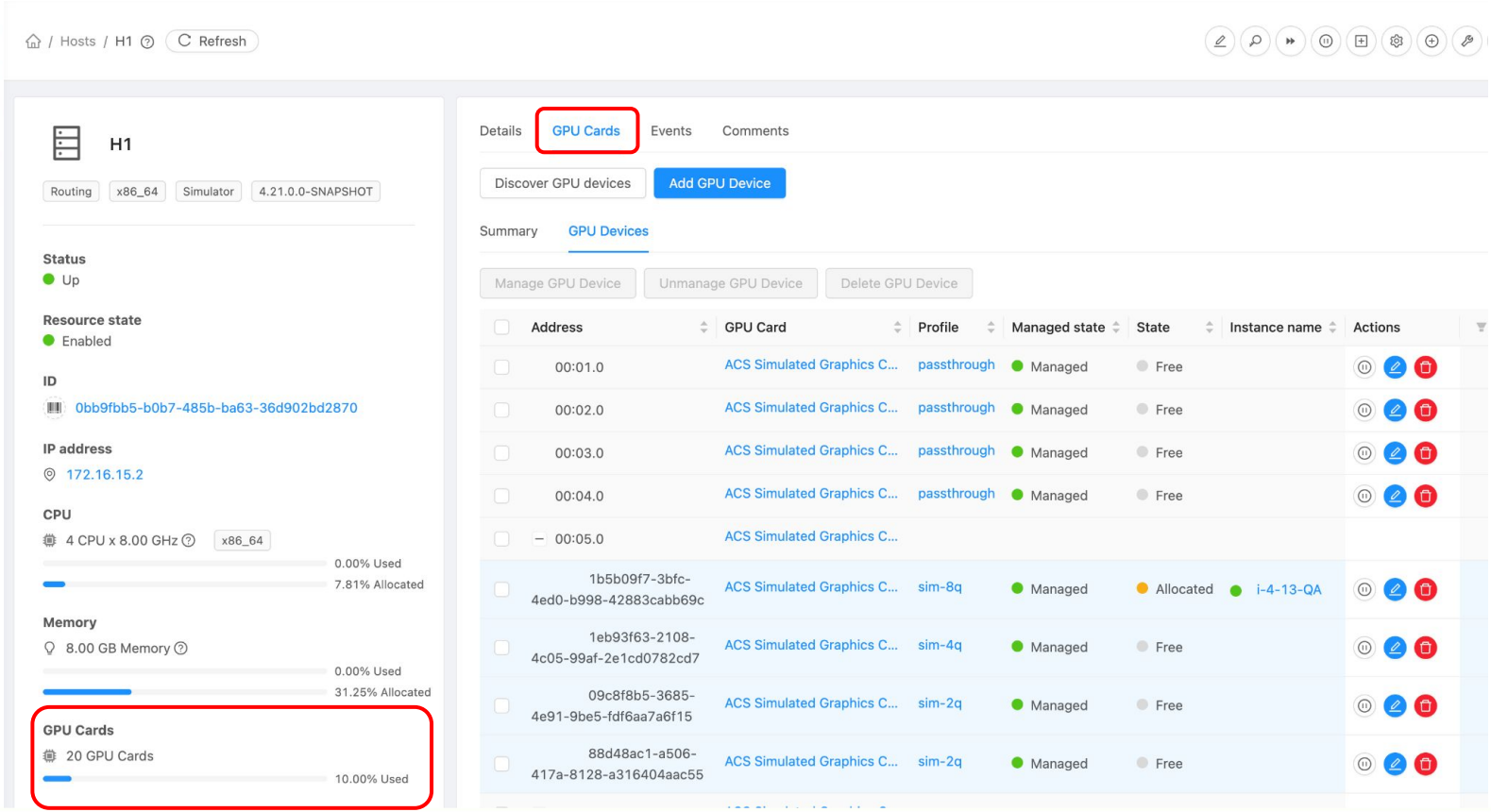
Preparing GPU-based Hosts

- Enable SR-IOV: Required for creating virtual function (VF) for sharing PCI GPU
- Enable VT-d (Intel) or IOMMU (AMD): Required for VFIO-PCI and shared GPU, for secure DMA and device isolation
- Enable Virtualisation: Intel VMX or AMD SVM
- PCI settings: GPU card(s) specific tunings
  - Above 4G Decoding
  - PCIe ARI Support
  - …

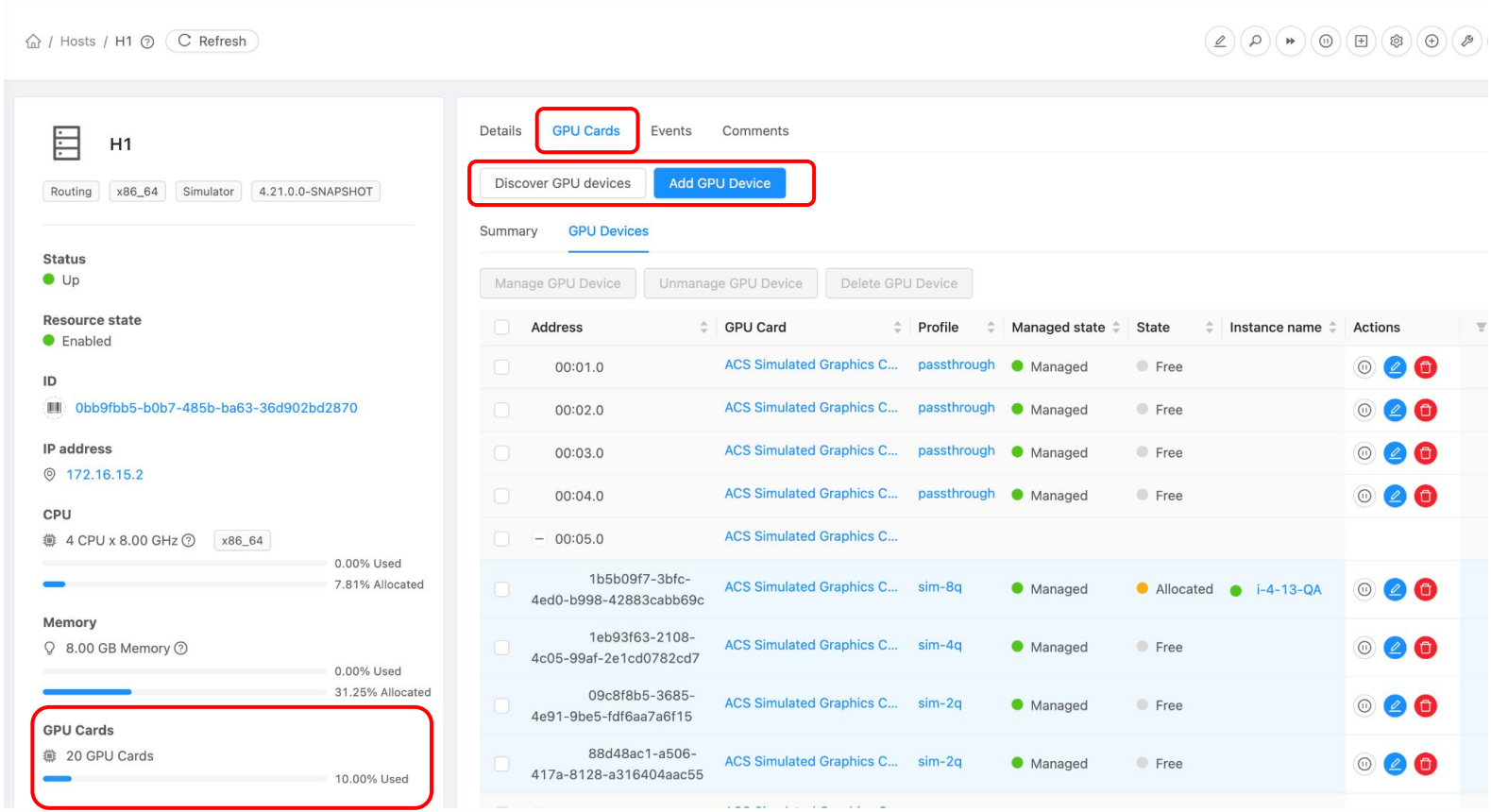
GPU Inventory Management



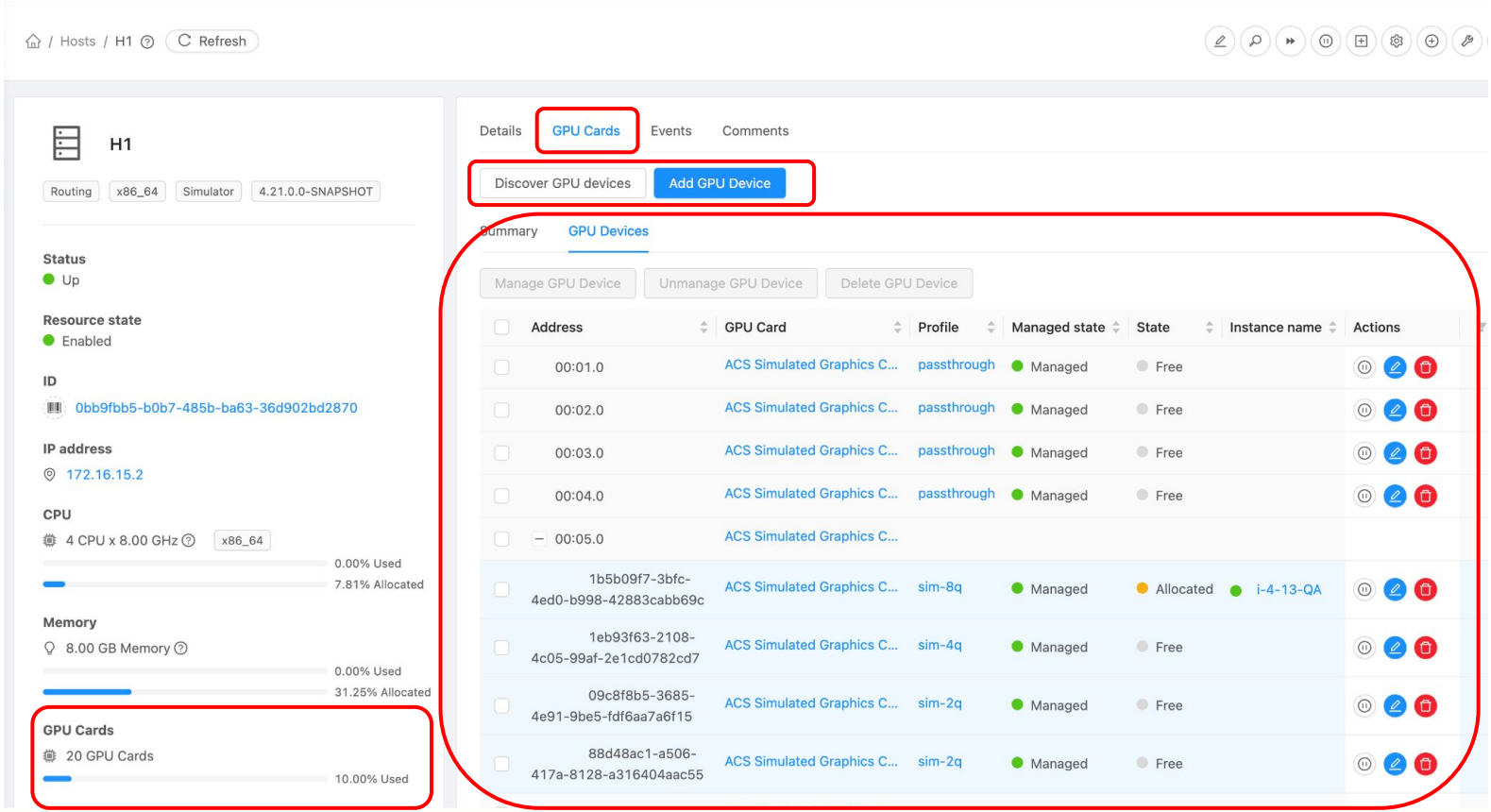
GPU Inventory Management



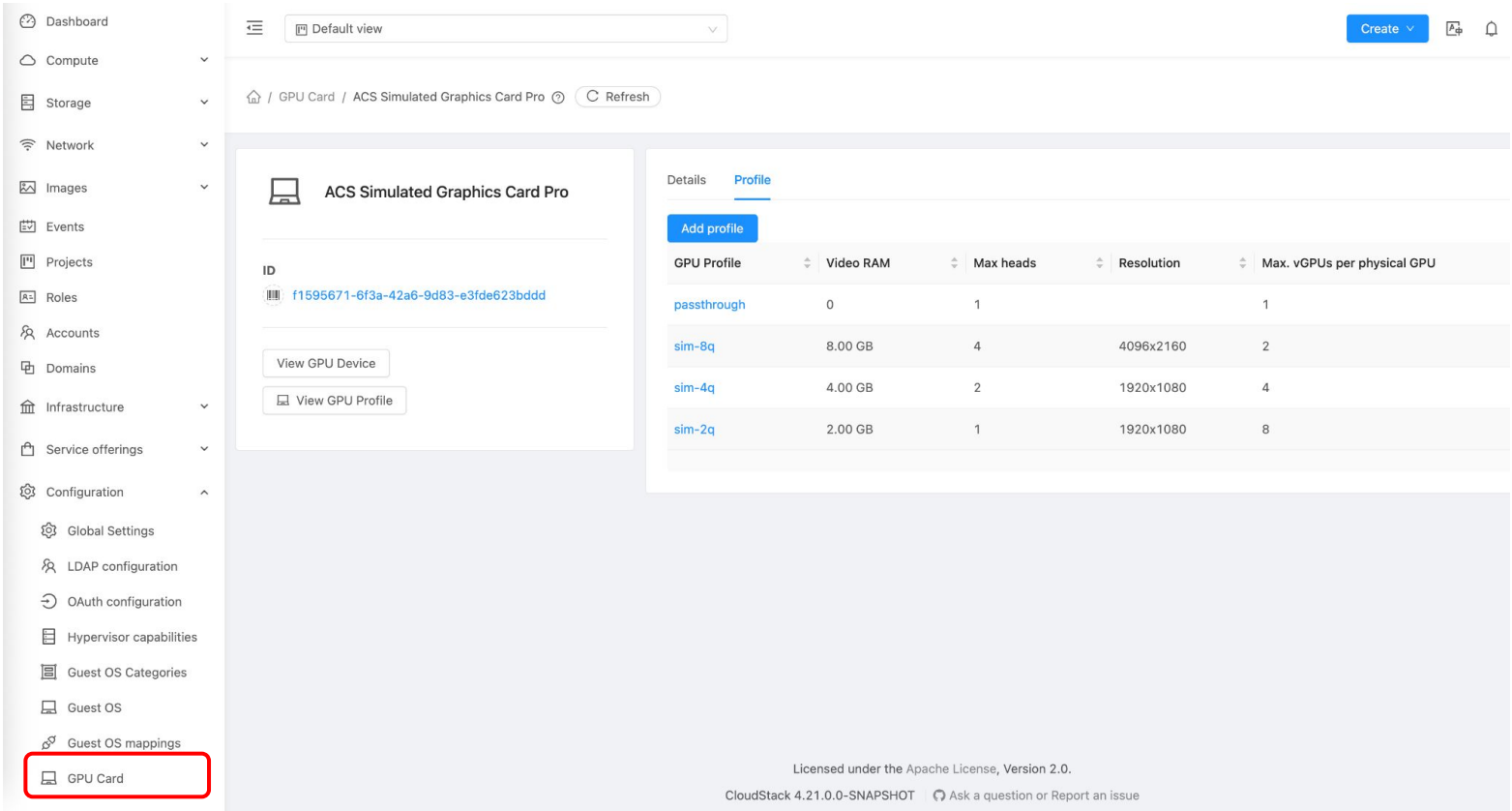
GPU Inventory Management



GPU Inventory Management



GPU Classification: GPU Card Types



Host Allocation & Orchestration

- Finds a set of host(s) based on host-GPU inventory, service offering, GPU limits and other input parameters

- Allocates GPU(s) on host to the instance based on service offering, GPU(s) availability and other parameters (such as NUMA)

Customisable Agent Hooks
GPU Auto-Discovery Script

Operator customisable GPU auto-discovery script

/etc/cloudstack/agent/hooks/gpudiscovery.sh


```
{
  "gpus": [
  {
    "pci_address": "01:00.0",
    "vendor_id": "10de",
    "device_id": "25a0",
    "vendor": "NVIDIA Corporation",
    "device": "GA107M [GeForce RTX 3050 Ti Mobile]",
    "driver": "nvidia",
    "pci_class": "3D controller [0302]",
    "iommu_group": "16",
    "pci_root": "0000:01:00.0",
    "numa_node": -1,
    "sriov_totalvfs": 0,
    "sriov_numvfs": 0,
    "full_passthrough": {
      "enabled": 1,
      "libvirt_address": {
        "domain": "0x0000",
        "bus": "0x01",
        "slot": "0x00",
        "function": "0x0"
      },
      "used_by_vm": null
    },
    "vgpu_instances": [],
    "vf_instances": []
  }
 ]
}
```

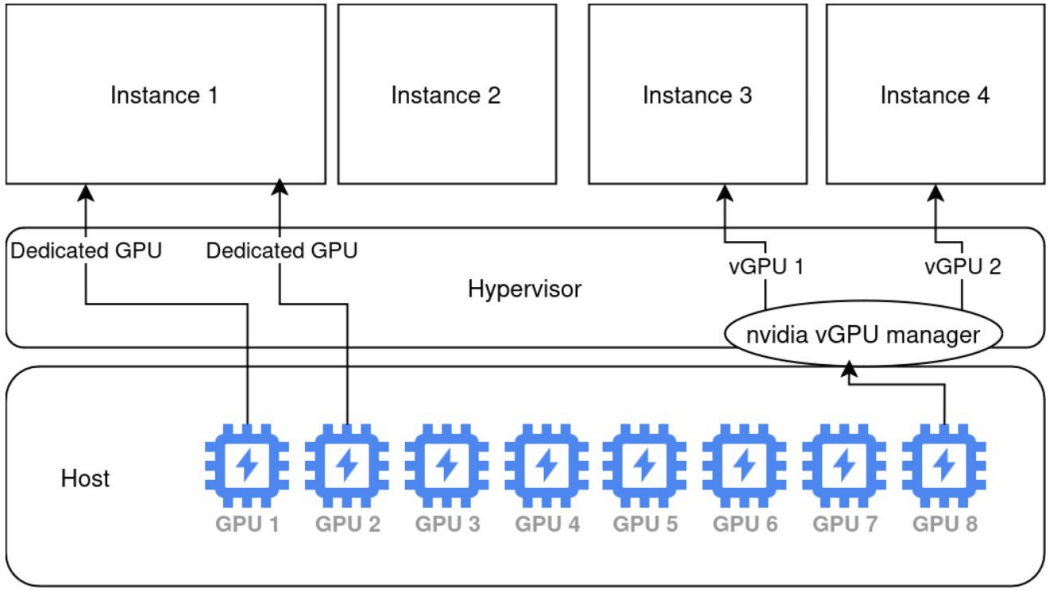
Customisable Agent Hooks
Domain XML Transformer & States

In addition to the Groovy-based agent hooks, following are supported:

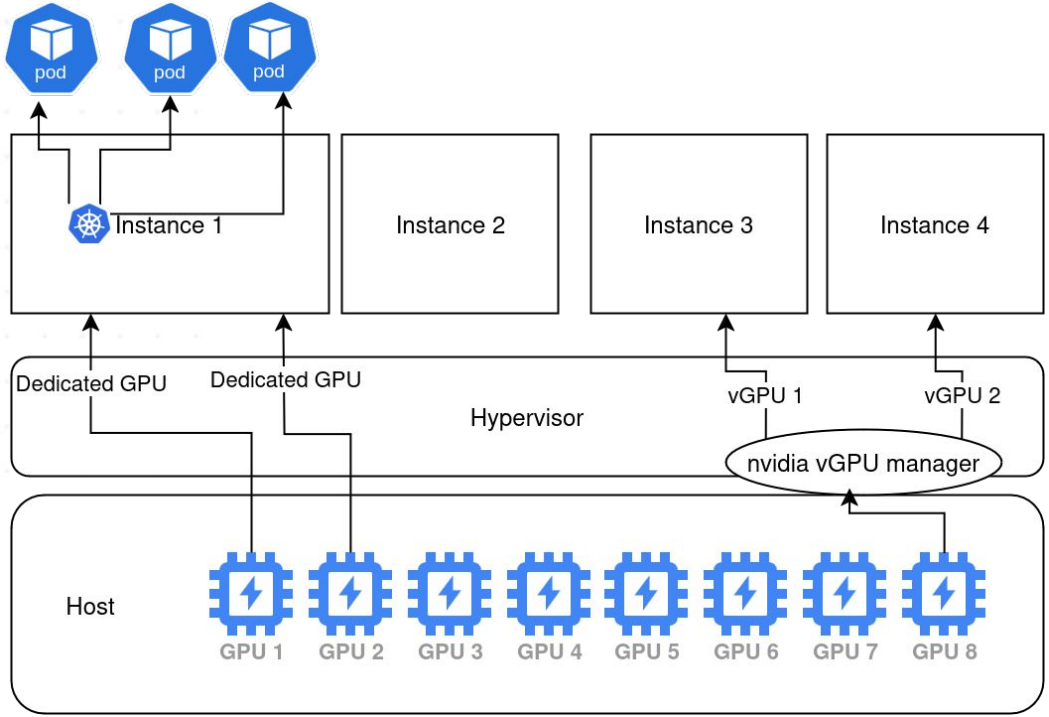
- Libvirt Domain XML Transformer **libvirt-vm-xml-transformer.sh**
- VM Start hook **libvirt-vm-state-change.sh**
- VM Stop hook **libvirt-vm-state-change.sh**

Location: **/etc/cloudstack/agent/hooks**

GPUs with Instances



GPUs with K8S (CKS/CAPC/EKS-A)



GPUs with Baremetal?

Yes - Possible!

Checkout the new
CloudStack Extensions Framework

Assumption & Limitations

- The role of CloudStack is limited to discover the devices on the host and assign/unassign them to instances. Admin/operator must configure the GPU devices properly on the KVM hosts.
- Admins/users must build custom templates with GPU tools and drivers specific to their environments and GPUs.

- Unsupported actions (as of now):
  - VM snapshot with memory for GPU enabled VM is not supported.
  - Dynamic scaling of VMs with GPU attached.
  - Live Migration of GPU enabled VM is not supported.
    - Cold migration is supported.
    - Migration with vGPU may work, but not verified/tested.

Understanding Current GPU Technologies

GPU Vendors (Currently)

	NVIDIA	AMD	Intel
Platform	CUDA	ROCm	oneAPI
Virtualisation	vGPU (GRID)	MxGPU (SR-IOV)	Basic/Flex
Current Standing	Market Leader	Growing Contender	Catching up
Ecosystem	PyTorch, TensorFlow...	Improving support in PyTorch, etc.	Laggard

Other contenders: Apple (Metal), Qualcomm (Adreno)…

GPU Types

- **Passthrough GPU**: full access of GPU card

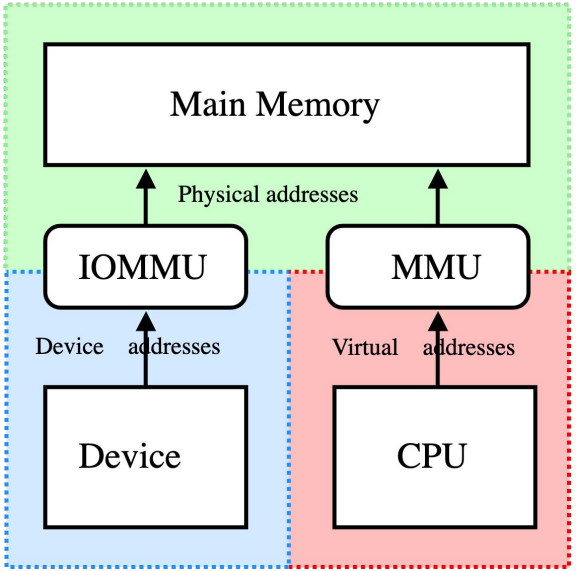
- **Shared GPU**: sliced / time-shared / partitioned GPU access
  - SR-IOV based (AMD MxGPU, Intel Flex)
  - vGPU (Nvidia)
  - MIG (Nvidia)
  - …

IOMMU: What is that?

The I/O Memory Management Unit (IOMMU) provides memory remapping services for I/O devices.

It adds support for address translation and system memory access protection on direct memory access (DMA) transfers from peripheral devices.

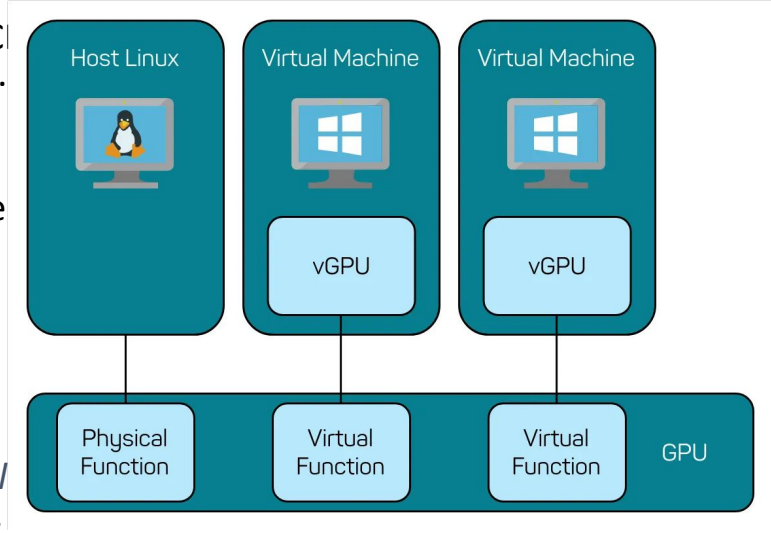
IOMMU maps device-visible virtual addresses (also called device addresses or memory mapped I/O addresses in this context) to physical addresses.



SR-IOV: What is that?

Single Root IO Virtualization (SR-IOV) allows the isolation of PCI Express resources for manageability and performance reasons. A single physical PCI Express bus can be shared in a virtual environment using the SR-IOV specification.

- Physical Functions: ability to move data in and out of the device.
- Virtual Functions (VF): lightweight PCIe functions that support data flowing but also have a restricted set of configuration resources.



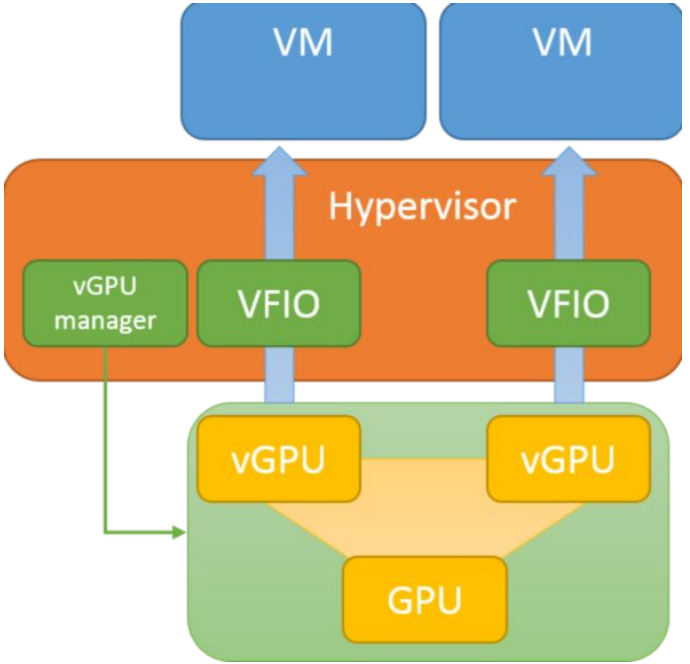
The SR-IOV allows different virtual machines (VMs) in a virtual environment to share a single PCI Express hardware interface.

TL;DR: SR-IOV enables VMs to access shared GPU(s) via Virtual Function

VFIO: What is that?

Virtual Function I/O (VFIO) *driver* is an IOMMU/device agnostic (Linux) framework for exposing direct device access to userspace, in a secure, IOMMU protected environment.

TL;DR: VFIO enables direct access of GPU Device to VM

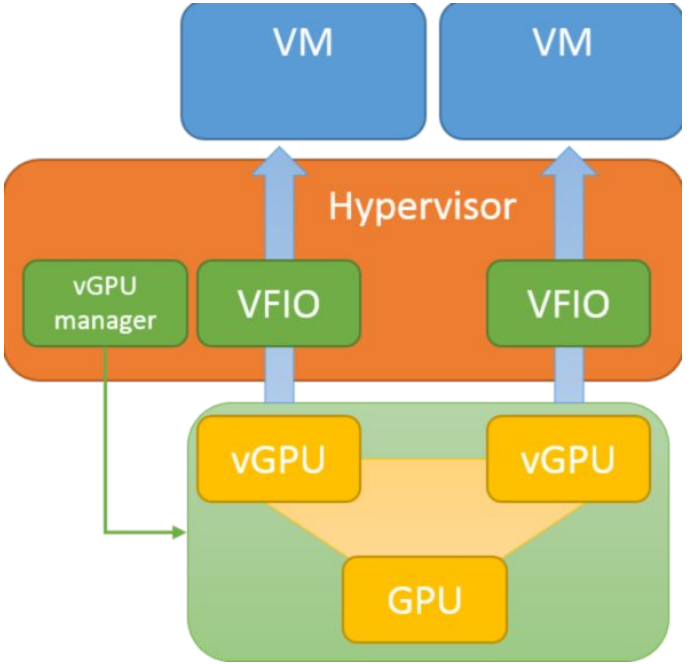


MDEV: What is that?

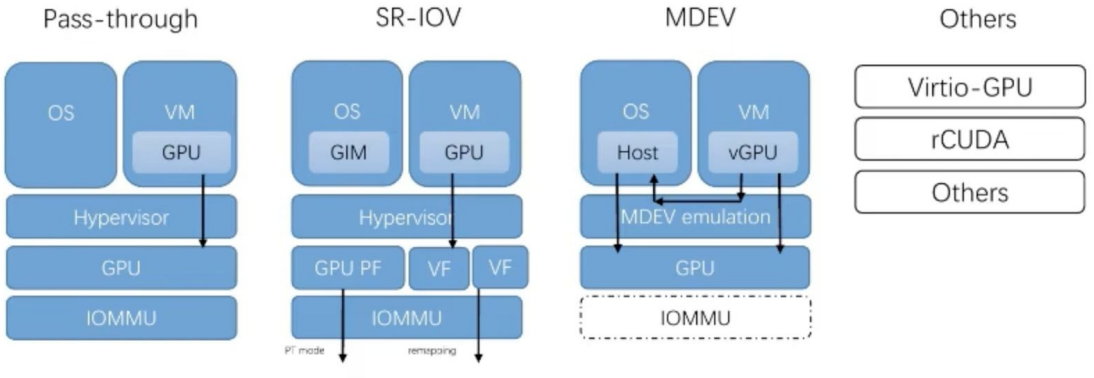
Mediated device (mdev) is a virtual device created by the host kernel from a physical PCI device, like a GPU, and exposed to VMs as a virtual PCI device.

The mdev framework allows the host device driver (e.g., NVIDIA, Intel, AMD) to partition and manage hardware resources, while still allowing each guest to use its own slice with near-native performance.

TL;DR: MDEV creates GPU partitions that can be used with VM using VFIO-mdev



GPU Virtualisation



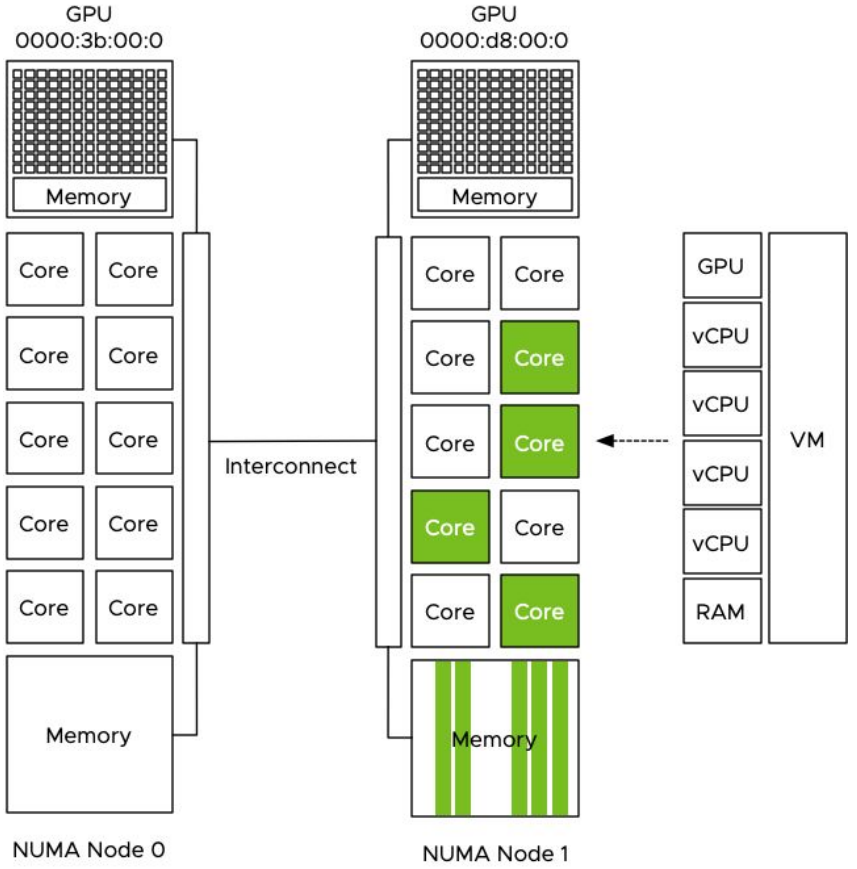
GPU Virtualisation

Feature	MDEV	SR-IOV	VFIO Passthrough
Sharing	Yes	Yes	No
Uses VFIO	Yes (`vfio_mdev`)	Yes (`vfio-pci`)	Yes (`vfio-pci`)
Granularity	Fine-grained virtual functions	Hardware-based VFs	Entire physical device
Device support	Software-defined (via driver)	Hardware-defined	Full passthrough
Needs IOMMU	Yes	Yes	Yes
Examples	NVIDIA vGPU	AMD MxGPU, Intel Flex	Full GPU passthrough
Guest Driver	Vendor vGPU driver	Vendor driver	Vendor driver

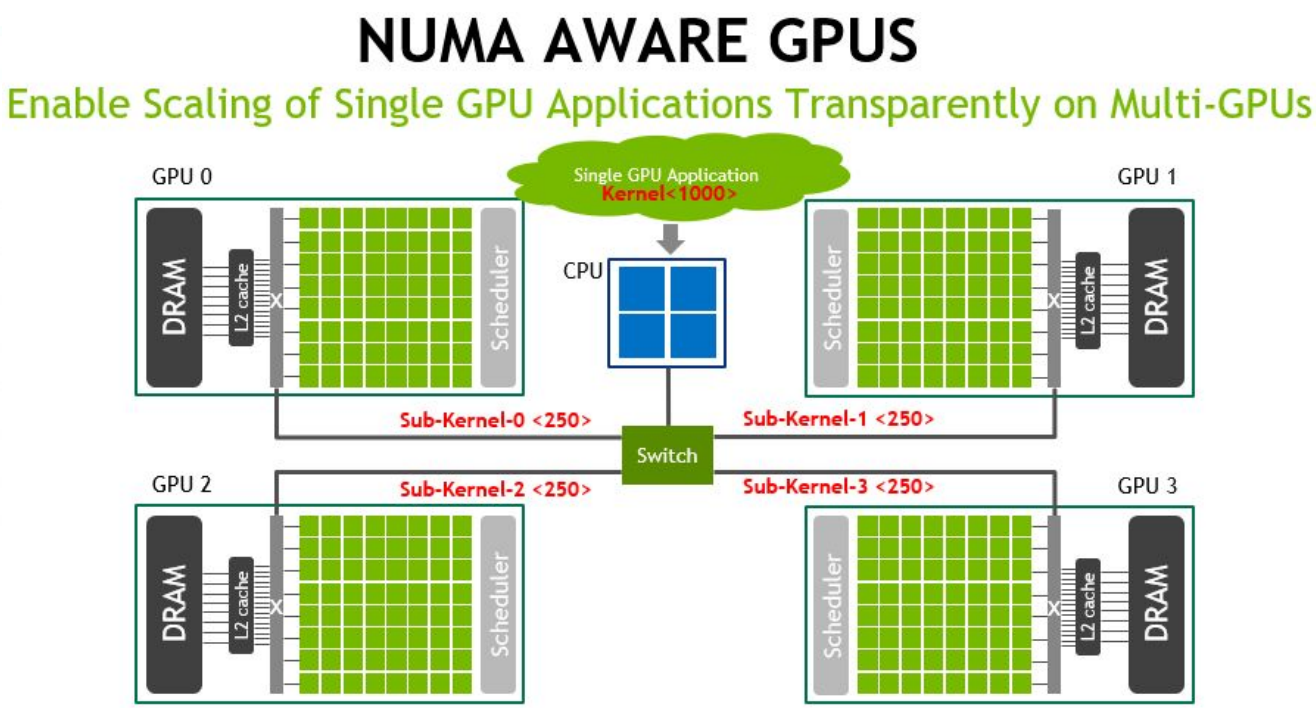
NUMA (Non-uniform memory access)

- NUMA systems are server platforms with more than one system bus.
- These platforms can utilize multiple processors on a single motherboard, and all processors can access all the memory on the board.

TL;DR: GPU placements on different NUMA nodes can have severe performance impact.

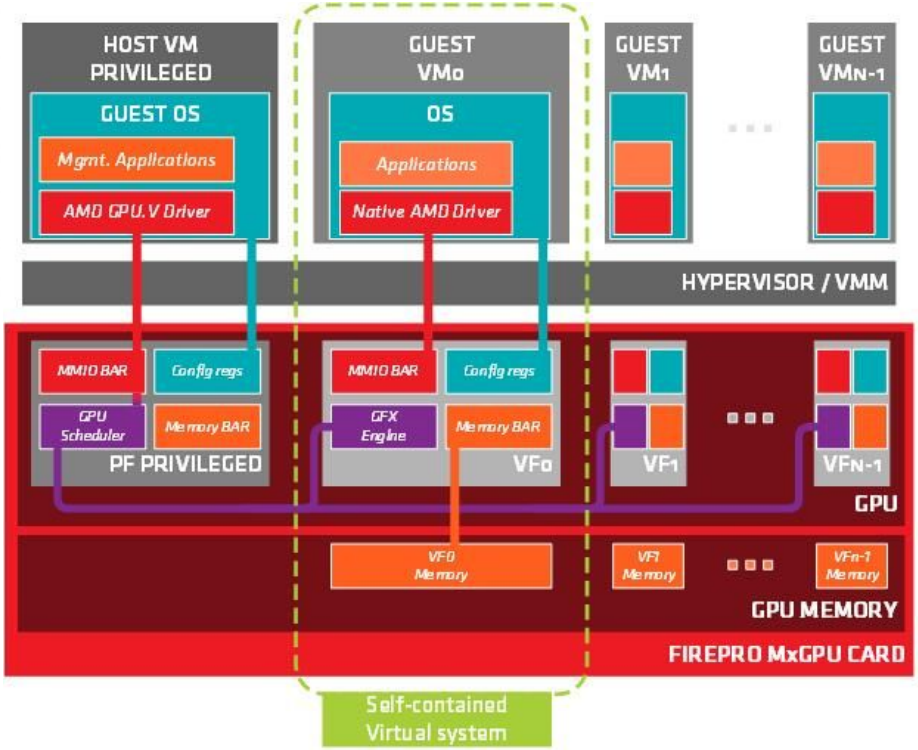


NUMA (Non-uniform memory access)



- Dynamic and asymmetric NVLINK bandwidth reconfiguration
- Dynamic NUMA-aware caching strategies

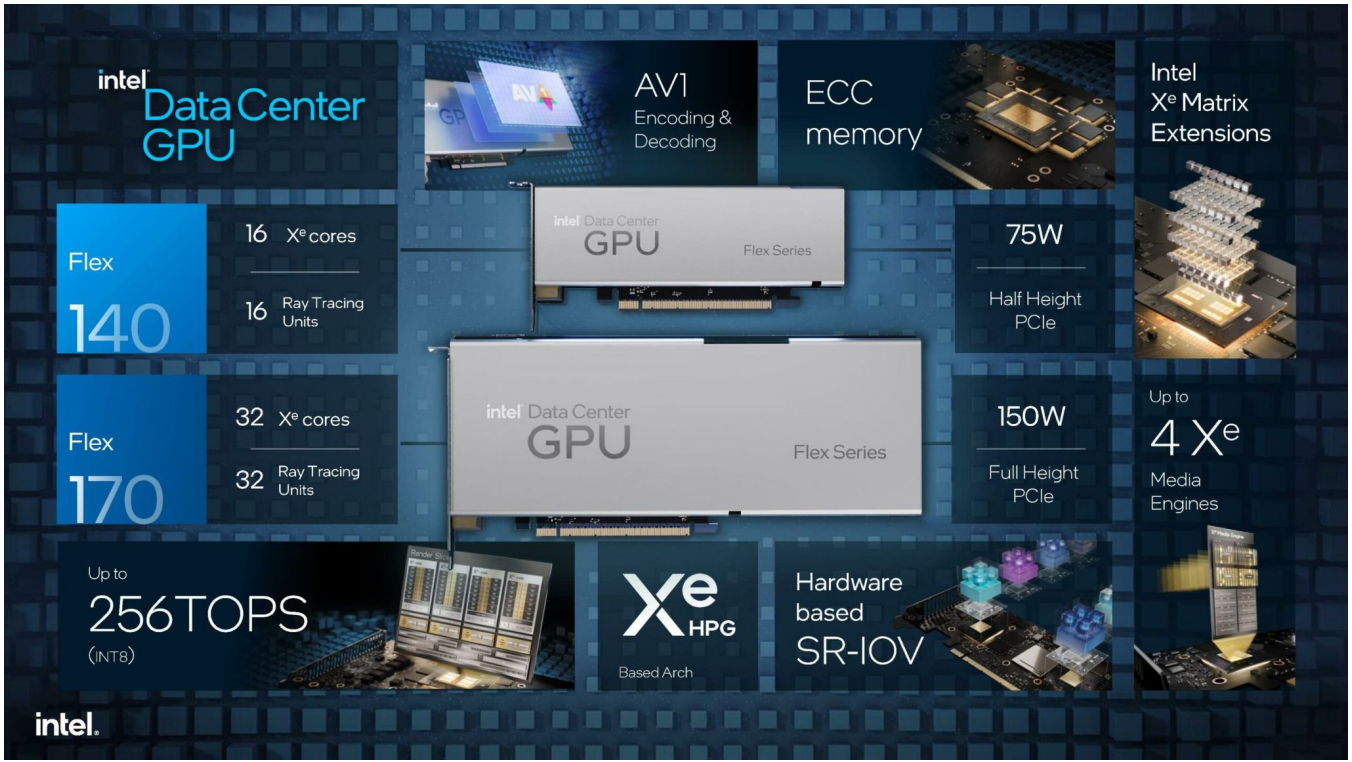
Example: AMD MxGPU (SR-IOV)



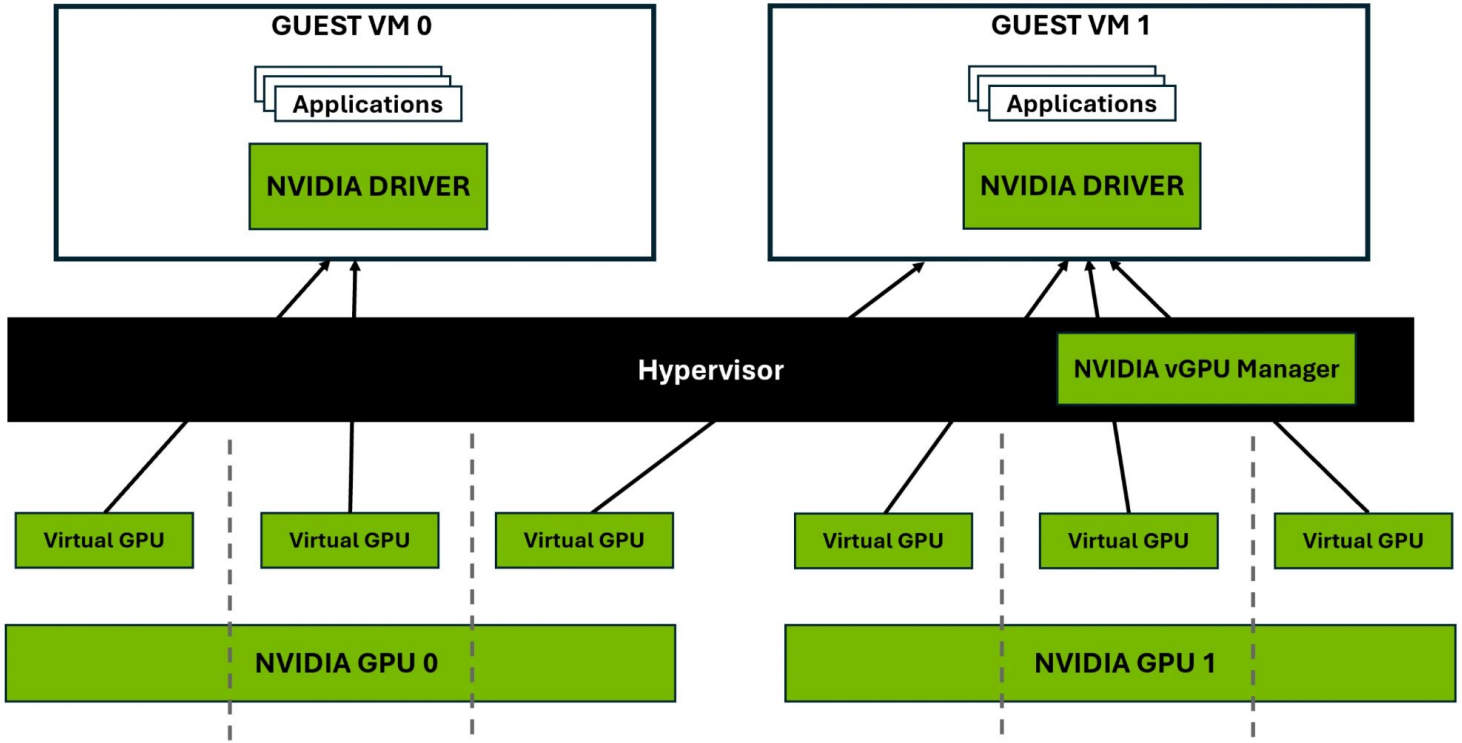
AMD GPU Cards

- MI210X
- MI300X
- MI350X

Example: Intel DC GPU Flex (SR-IOV)



Example: NVIDIA vGPU (MDEV)



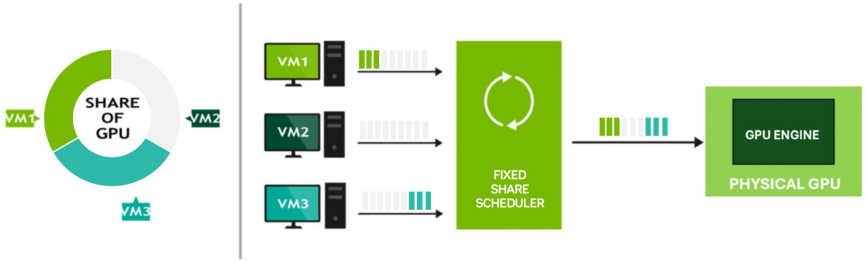
https://docs.nvidia.com/vgpu/knowledge-base/latest/vgpu-features.html

Example: NVIDIA vGPU Schedulers

Best Effort Round-Robin Scheduler

- VRAM Partitioned
- Time-sliced Sharing of Entire GPU

Equal-Share Scheduler

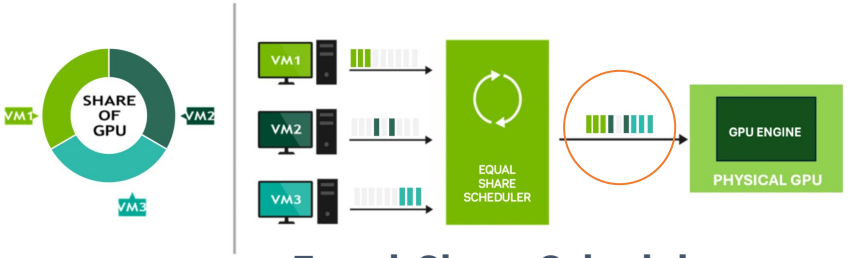


Fixed-Share Scheduler

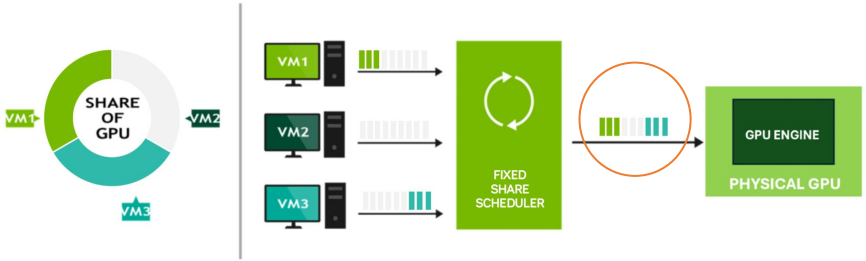
Example: NVIDIA vGPU Schedulers

Best Effort Round-Robin Scheduler

- VRAM Partitioned
- Time-sliced Sharing of Entire GPU

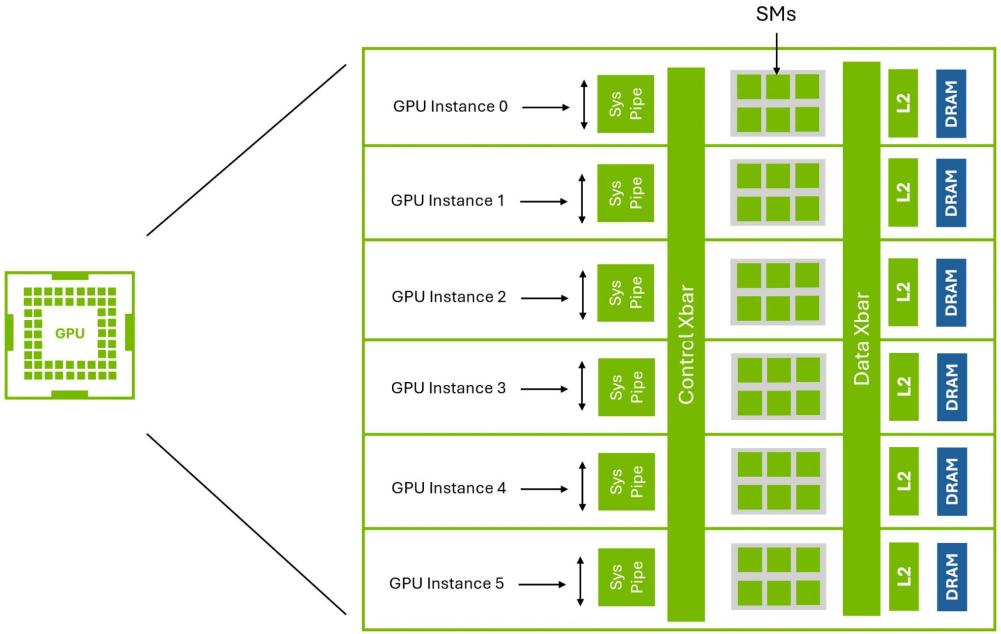


Equal-Share Scheduler

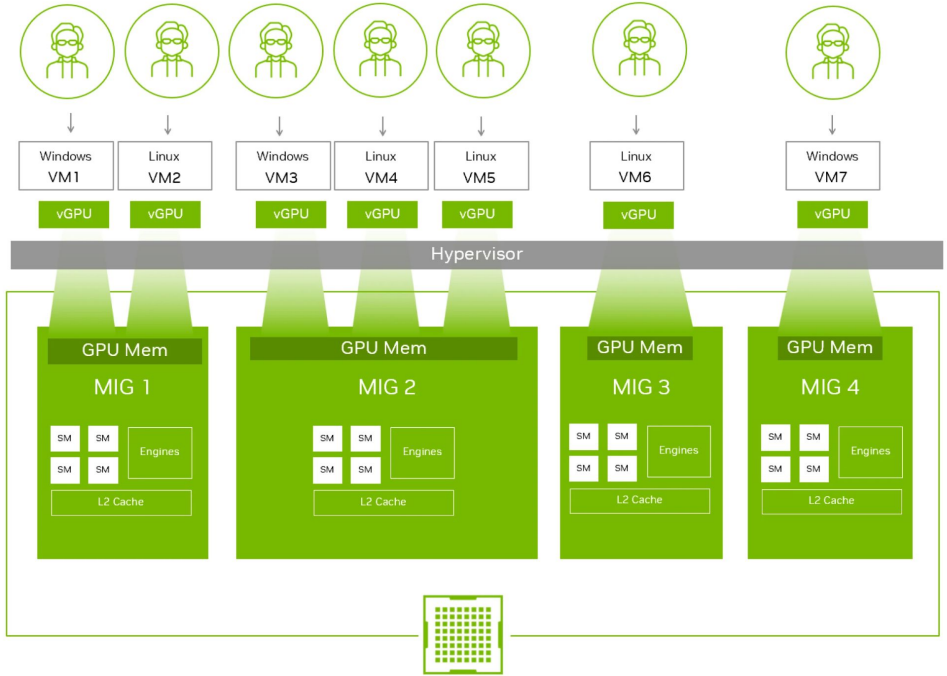


Fixed-Share Scheduler

Case Study: NVIDIA Multi-Instance GPU (MIG) (VFIO-PCI)



Case Study: MIG based Multi-Tenancy



RTX PRO 6000 Blackwell Server Edition

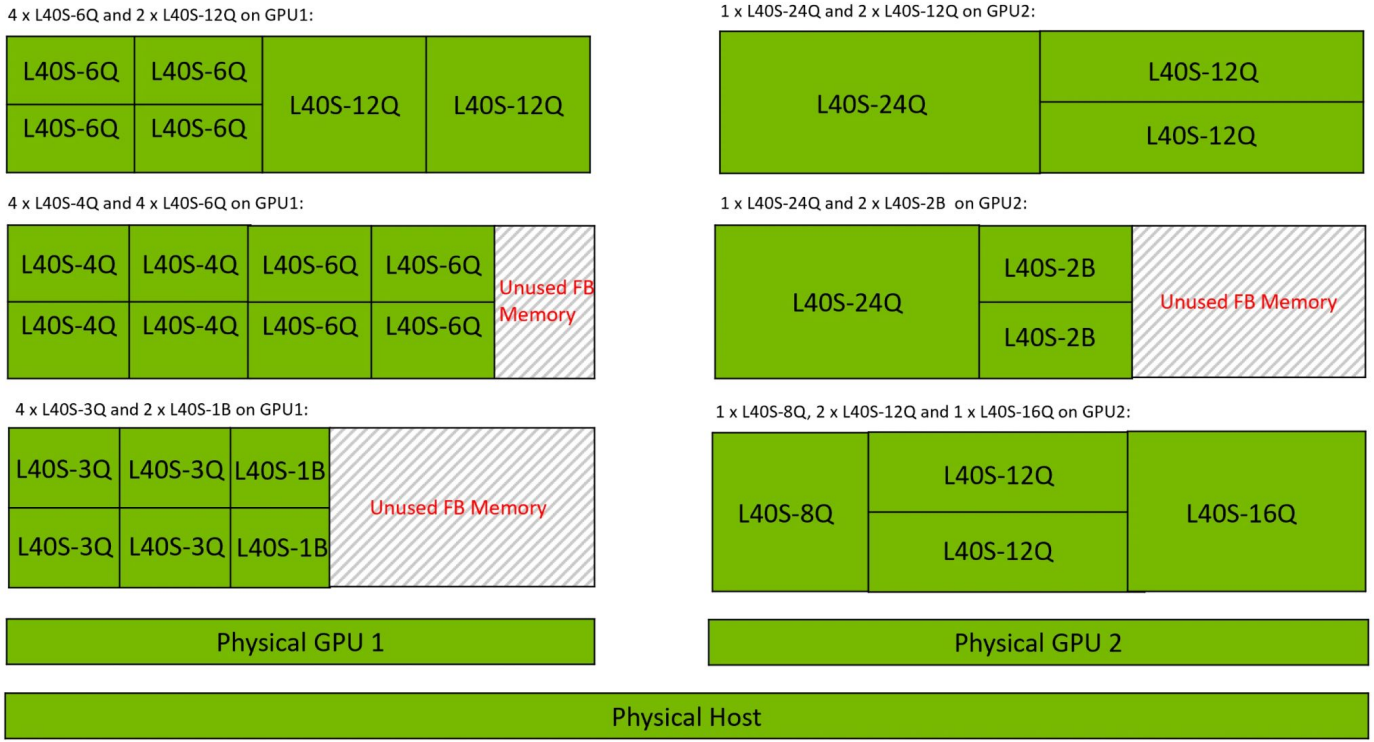
Shared GPU Type	Vendor(s)	Concurrent?	Isolation Level	Notes
vGPU (NVIDIA GRID)	NVIDIA	Yes	Medium	Licensed, great for VDI/AI-Infer.
SR-IOV / MxGPU	AMD, Intel	Yes	High	Hardware-dependent
Time-Sliced Sharing	NVIDIA, others	No	High	Simple, but not parallel
MIG (NVIDIA)	NVIDIA A100, etc.	Yes	Very High	Each instance fully isolated
API/Driver Level Sharing	All (via OS)	Yes	Low	Easiest for containers

Understanding NVIDIA vGPUs

Series	Optimal Workload
Q-series	Virtual workstations for creative and technical professionals who require the performance and features of Quadro technology
B-series	Virtual desktops for business professionals and knowledge workers
A-series	App streaming or session-based solutions for virtual applications users

https://docs.nvidia.com/vgpu/latest/grid-vgpu-user-guide/index.html

NVIDIA Heterogeneous vGPU Profiles



Example: NVIDIA L40S GPU Card

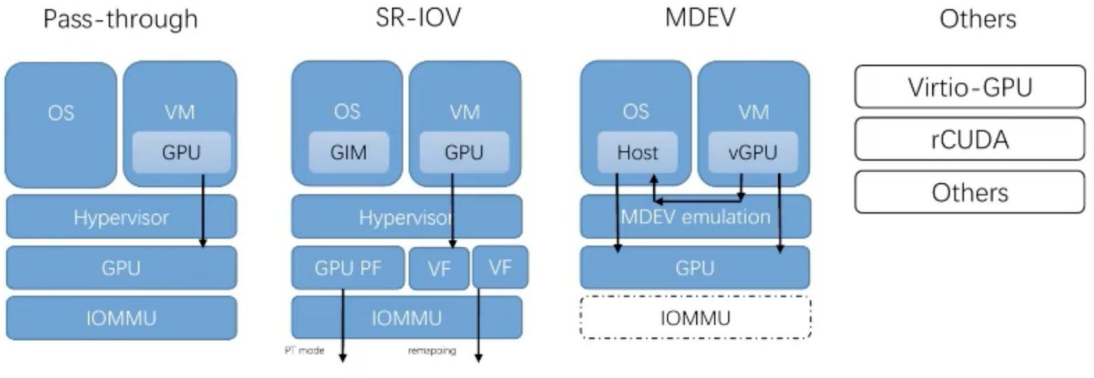
Designing & Deploying GPU-Enabled IaaS Cloud

Popular GPU Vendors (Currently)

	NVIDIA	AMD	Intel
Platform	CUDA	ROCm	oneAPI
Virtualisation	vGPU (GRID)	MxGPU (SR-IOV)	Basic/Flex
Current Standing	Market Leader	Growing Contender	Catching up
Ecosystem	PyTorch, TensorFlow...	Improving support in PyTorch, etc.	Laggard

Other contenders: Apple (Metal), Qualcomm (Adreno)…

GPU Usage & Virtualisation



GPU Virtualisation

Feature	MDEV	SR-IOV	VFIO Passthrough
Sharing	Yes	Yes	No
Uses VFIO	Yes (`vfio_mdev`)	Yes (`vfio-pci`)	Yes (`vfio-pci`)
Granularity	Fine-grained virtual functions	Hardware-based VFs	Entire physical device
Device support	Software-defined (via driver)	Hardware-defined	Full passthrough
Needs IOMMU	Yes	Yes	Yes
Examples	NVIDIA vGPU	AMD MxGPU, Intel Flex	Full GPU passthrough, NVIDIA MIG
Guest Driver	Vendor vGPU driver	Vendor driver	Vendor driver

Supported KVM Hypervisor: NVIDIA GPUs

- Red Hat Enterprise Linux 9 & 8 with KVM hypervisor
- Ubuntu 24.04 LTS and 22.04 LTS with KVM hypervisor

https://docs.nvidia.com/vgpu/latest/product-support-matrix/index.html

GPU IaaS Clouds Use-Cases

AI / Machine Learning / Deep Learning	Model training, inferencing, Computer Vision, NLP…
Virtual Desktops (VDI w/GPU Acceleration)	Access GPU-intensive apps remotely
Cloud Gaming & Streaming	Game and video streaming services
Graphics & Media	Video and 3D Rendering, Production
High Performance Computing (HPC)	Scientific simulation, fluid dynamics, molecular modeling…
World Simulation	Industrial simulations, urban planning, metaverse apps
Data Analytics & Big Data	GPU accelerated analytics
Cryptocurrency / Blockchain	GPU-based mining, blockchain compute

Picking the appropriate GPUs

Example

Shared GPU Workloads: (vGPU/MxGPU/VF SR-IOV)

- NVIDIA L2
- NVIDIA L4
- NVIDIA L20
- NVIDIA L40
- NVIDIA L40S
- NVIDIA A2
- NVIDIA A10
- NVIDIA A16
- NVIDIA A40
- Tesla V100S, Tesla V100
- AMD MI210X, MI300X, MI350X
- Intel DC Flex Series…

More at:
https://www.nvidia.com/en-us/data-center/graphics-cards-for-virtualization/
https://instinct.docs.amd.com/projects/virt-drv/en/latest/userguides/Getting_started_with_MxGPU.html
https://www.intel.com/content/www/us/en/products/details/discrete-gpus/data-center-gpu/flex-series.html

AI Workloads:

- NVIDIA H100
- NVIDIA H800
- NVIDIA A800
- NVIDIA A40
- NVIDIA A30
- NVIDIA A10
- NVIDIA A16
- NVIDIA A2
- NVIDIA L4
- NVIDIA L40
- NVIDIA L40S
- NVIDIA V100, …

More at:
https://docs.nvidia.com/ai-enterprise/4.1/pdf/nvidia-ai-enterprise-release-notes.pdf

Considering Passthrough GPUs

- Any workload requiring large VRAM
  - High-end AI/ML training
  - 3D work or game development
- Latency-Sensitive or Real-Time Workloads
  - Video Transcoding
  - Robotics Control and Industrial Automation
- Security and Isolation needs
- GPU Virtualisation not available

Considering Shared GPUs

- VDI or Desktop as a Service (Graphics Acceleration)
- Moderate-workloads:
  - VRAM intensive AI/ML Inferencing
  - Cloud gaming or media streaming
  - SaaS/Remote apps requiring Short-Medium GPU Bursts
  - Dev/Testing or CI/CD for GPU based Apps
  - Education & Learning Apps, such as Jupyter notebooks
- Media Transcoding and Encoding at scale
- Workloads requiring high-availability and uptime

Challenges & Future Work

Challenges & Future Work

- Accessibility of GPU-based hosts for feature development and QA
  - Test vGPU, MIG and passthrough: Nvidia, AMD, Intel…
  - Test Live VM Migration for vGPU case

- Validate and improve built-in GPU auto-discovery and domain transformer scripts against different GPU cards/models

- Improve feature implementation and aim for production readiness with ACS 4.22.

- GPU Metrics & Expand support for other hypervisors.

Thank You

#CSIndiaUG2025