NVIDIA Requirements for AI Clouds

Technical Reference Manual

TRM-12815-001 | March 2026

Document History

TRM-12815-001

Version	Date	Authors	Description of Change
2.1	Feb 26, 2026		Initial version

Table of Contents

Purpose and Intent

These are the standards and expectations for NVIDIA Cloud Partners (NCPs) operating NVIDIA GPU-accelerated AI cloud infrastructure. They cover the full operational stack, from compute and Kubernetes to storage, networking, security, telemetry, and fleet management, and expand on the NVIDIA hardware reference design and NCP Software Reference Guide.

NVIDIA is not seeking bespoke implementations. All capabilities described here must be Generally Available (GA) to all customers, or on a clear path to GA.

NVIDIA will also consider additional capabilities that an NCP offers or may offer in the future.

Service Delivery SLAs

NCPs should be able to demonstrate ability to meet below SLA by category and operational requirements to be considered for offtake.

Services Delivery Timelines

The NCP must demonstrate API readiness, transport establishment at least 12 weeks ahead of GPU delivery, and the ability to provide Dev capacity with the API integrated 6 weeks prior to GPU and cluster delivery.

Access to Data Mover cluster prior to GPU capacity availability to move data ~2 weeks ahead of GPU cluster delivery.

SLA and SLO

Managed K8s

- Control Plane SLA target: Financially-backed 99.95%+ uptime for production.

Storage

- Performance (QoS): Must provision needed throughput requested for minimum bandwidth and IOPS.
- Home Directory Storage:
    - Availability: Over 99% availability for unplanned incidents. Exclusive of scheduled maintenance.
    - Durability: Over 99.99% for any FS less than 1 PB
- High Speed Storage Service Requirements:
    - Availability (SLO): Must meet 99.99% availability in a 30-day rolling SLO exclusive of maintenance
- High-Speed Storage Filesystem Requirements
    - End to End Availability: Over 99.5% uptime per PB
    - Durability: Over 99.999% durability per PB

Operational Requirements

- Technical specialist/engineer available as needed.
- Slack channel monitored by technical specialist / engineer.
- 24x7 support available per partner standard incident severity procedures.
- Service impacting incidents, planned, and unplanned maintenance events are communicated to NVIDIA.
- For planned maintenance, NVIDIA can schedule maintenance windows via APIs / console tools - avoiding unexpected outages + the ability for NVIDIA to provide feedback.

Exemplar Cloud Workload Performance

NVIDIA Exemplar Cloud seeks to improve performance per TCO with hardware and software recipes, references, tools, and capabilities. Run the latest publicly available release from https://github.com/NVIDIA/dgxc-benchmarking **(Always pick the latest release version from the GH repo)** to be successfully completed on 1 uniform HW cluster type. Please run all the workloads for a given release and share the results in the template below.

Feature	Min	Target Measurement
Benchmarking for exemplar cloud	512 GPU cluster	Performance within 5% of NVIDIA Reference

Compute and Network Provisioning

This section outlines the requirements for provisioning compute and network. Compute instances can be provided as either Bare Metal instances (via BMaaS) or Virtual Machines (via VMaaS) to support the NVIDIA DGX Cloud engagement. All operations must be controlled via a fully documented and secure API, gRPC or REST preferred. All systems are expected to scale and perform at scale.

General, Compute and Lifecycle Management

Requirement Area	Description
API/CLI Access	DGXC must have API or CLI access to the NCP provisioning system for: (1) Node lifecycle management (create, update, delete, list or manage power states (reboot, on/off power cycle); (2) Network configuration; (3) Inventory and topology discovery (storage discussed in storage section)
Declarative Resource Interfaces	For resources requiring multiple steps and a workflow, please provide the appropriate mechanism. A terraform provider is preferred. E.g. automating filesystem provisioning.
NVLink-Aware Allocation	For NVL72 the API must support NVLink domain-aware allocation.
Instance States	Must support clear instance states (where applicable). For example, provisioning, running, degraded, maintenance required, stopping, stopped, terminating, terminated.
Tagging	Support for user-defined tags/labels and cloud-init metadata on instances.
Console Access	Serial console access is required (read-only sufficient, interactive preferred).
If VMaaS present: # VMs/Node	GPU Nodes: no more than one VM per Node. General Purpose CPU Nodes: More than one per node preferred, with ability to select via memory/core count shape.
Stable Identifiers	All resources (e.g. nodes, switches) must have a stable and persistent ID that does not change during the lifespan, even when it goes offline for a service event. VMs must also have a stable identifier.
Firmware	Between tenants, all firmware must be brought to a known good state.

Boot Process and Disks

Requirement Area	Description
Image Deployment and Updates	API-driven workflow allowing DGXC to integrate and deploy vendor-provided or custom disk images via bare-metal or VM provisioning.
Access to Instance Metadata from Guest OS	Support for cloud-init and instance metadata discovery via link-local addresses or virtual devices.
Custom Disk Images	Support for custom OS images (either of: raw, qcow2, etc). API calls: get, list, create, delete.
Node Local Storage	GPU and CPU nodes support access to node local storage (NVMe / SSD) for use as scratch storage or for caching services.

SDN and Virtual Networking

This section covers the virtual networking requirements. Physical transport and network are discussed later in the document.

Requirement Area	Description
Virtual Networking	Full API/CLI lifecycle management (Create, Read, Update, Delete, List) for software-defined private networks. Must support non-conflicting **BYOIP** (including **7.0.0.0/8**) and stable private IP allocations.
Security Groups	Support for VPC-style security groups (or equivalent). Must define scope/application at workload, node, and subnet/tenant levels.
Security Operations	Full API/CLI capabilities to Create, Read, Update, and Delete security groups, including defined audit processes and policy propagation timing.
Tenant Isolation	Hard logical or physical network segmentation for management (BMC), user traffic, and storage-specific operations.
Floating/Movable IP	Ability to atomically switch a floating private IP between nodes via API within <10 seconds without requiring an instance reboot.
Localized DNS	Support for custom, localized DNS settings to enable internal domain resolution to private endpoints (e.g. storage endpoints).
VPC Peering	Support for cross-virtual-network connectivity with full bandwidth and no "hairpin" routing.
All-to-All Communication	(Storage) hosts must be able to route to each other with all-to-all communication capability. L3 routing is acceptable; systems are not required to be on the same L2 subnet.

Kubernetes As a Service (KaaS) Requirements

NVIDIA wants a managed Kubernetes service from the NCP.

Kubernetes Conformance

- Certified Upstream Versions: Official CNCF-certified versions only; no proprietary forks.
- Timely Updates: support the three most recent minor releases (in the maintenance window); new minor versions must be available within 4-6 weeks of the upstream release; automated control plane security patching.
- Standard APIs: Unrestricted access to core APIs (Deployments, Secrets, etc.).
- No (Mandatory) Proprietary Extensions: Core functionality must remain portable to other standard environments.
- Kubernetes Proxy: Strict adherence to upstream proxy requirements.
- Dynamic Resource Allocation (DRA): Enabled regardless of upstream feature status (Beta/GA).

Kubernetes Operational Excellence

- Automated Lifecycle Management: API/CLI/UI provisioning; <10 min control plane bring-up.
- Versioning: Provider-managed control plane upgrade processes.
- Zero-Downtime Upgrades: Minor version control plane updates without app downtime or maintenance windows.
- Node Upgrades: Automated, user-initiated rolling updates respecting disruption budgets.
- HA Control Plane: Redundant architecture with etcd separation.
- Backup and Disaster Recovery: Supported recovery within defined RPO/RTO; needs to be auditable and testable .

Robust K8s Security

- Control Plane Isolation: per tenant k8s control plane outside of the tenant cluster/VPC.
- Access Controls: Cluster endpoint must provide network access controls.
- IAM Integration: Native provider IAM or OIDC/SAML for RBAC; support for Workload/Node identities.
- Service Accounts: Standard SA support (k8s API server) with accessible OIDC endpoints.
- Encryption: at-rest encryption for etcd and secrets.
- Cert Management: 60-day rotation; support for both provider and customer-managed keys.
- Observability: Mandatory traffic flow logs (L3) including pod-to-pod traffic.

Kubernetes Versioning and Upstream Compliance

- Alignment: Support N-2 minor releases; desired release within 6 weeks of upstream.
- Patching: Automated control plane patches with minimal/no downtime.
- Upgrade Paths: Documented, automated in-place upgrades for nodes and control planes.
- EOL Policy: Defined notification periods for version deprecation.

Kubernetes Component and Extension Requirements

- API Extensions: Mandatory support for CRDs and Validating/Mutating Admission Controllers.
- CNI: Standard compliance; supports Network Policies; IPv4/IPv6 dual-stack desired.
- CSI: Support for dynamic provisioning, snapshots, and resizing. Drivers manageable via Helm/Kustomize.
- DRA: The service should support the Kubernetes Dynamic Resource Allocation API for specialized hardware resources (e.g. GPUs)
- Autoscaling: Upstream Cluster Autoscaler integration required; Karpenter preferred.
- GPU Operator: Full compatibility with the mainline NVIDIA GPU Operator (automated drivers/plugin management).

Security and Identity Management

Identity and Access

- **Authentication:**
  - **Users:** Mandatory integration with OIDC
  - **In-Cluster:** IAM role assignment to nodes; must support short-lived access tokens.
  - **Out-of-Cluster:** Credential-based service accounts (non-human identity) with long-term credentials
  - Service (TBD)
- **Authorization (RBAC):** Mandatory least-privilege RBAC for all API actions (Compute, Storage, Network).
- **Kubernetes Identity:** Support for standard Service Accounts (SA), OIDC endpoints, and Workload/Node IAM identities.
- **Storage Identity:** LDAP integration (RFC2307bis) for POSIX-based access control.
- **Audit**: Audit logs must be retained and be accessible by NVIDIA at least 30 days

Network Isolation and Encryption

- **Tenancy Model:** Hard physical or logical isolation for network, data, and compute. Separation of control planes and tenants.
- **BMC Security:** Out-of-band management (BMC) must be on a dedicated, restricted network (physically separate or VLAN/VRF-isolated).
- **Network Traffic Encryption:** Encryption and mutual authentication (mTLS or equivalent) for all east-west and north-south traffic.

Edge Network Security

- **Private Access:** No public internet access by default; all API endpoints (e.g. K8s API Server) must be restricted via firewall/private link.
- **Edge Network Security Policy:** All traffic must be filtered via Security Groups.
- **Enforcement**: NCP must specify the enforcement technology (e.g., Hardware firewalls, SDN, DPUs/SmartNICs) and its specific placement in the packet path.
- **Threat Intelligence and Scale**: Ability to subscribe to GeoIP threat and Embargo feeds and import them into security groups. NCP should share the max supported records/rules.
- **MACSec protection links**: Protect links between NCP Data Center and NVIDIA POP and Object store.

Hardware Security and Compliance

- **SOC 2:** SOC2 type 1 or better is required covering Security, Availability, and Confidentiality.
- **At-Rest Data Protection:** Mandatory encryption of all data at rest (e.g. local NVMe/SSD, network-attached storage) via Self-Encrypted Drives (SED).
- **Data Sanitization:** Cryptographic erase of all data drives between tenants and during hardware replacement (CPU/GPU/Storage). Memory sanitization between tenants.
- **Hardware Root of Trust:** Mandatory support across all platforms for TPM/vTPM.

Breakfix Requirements

The NCP must provide a specific "Breakfix API" to support fleet reliability. Any node-level remediation must not impact other parts of the tenancy; specifically, NVLink must be re-configured properly to take a node out of the tenancy.

Breakfix API Capabilities

The API must enable the lifecycle actions:

- **Compute:** Power-cycle individual nodes or reset a VM instance.
- **GPU:** Reset GPUs on an individual node.
- **Maintenance:** Return an individual node and a rack to the Provider for maintenance
- **Cordon:** Mark a node as unschedulable for new workloads (but finish existing)
- **Replace**: Request a host-replacement when health thresholds are breached
- **Events:**
    - Query for any upcoming/current maintenance events for a node or rack
    - Query for any retirement notices for a node/rack.
    - Query for historical / status information for equipment repair.
    - Event information should include:
        - **Ticket open date**
        - **Ticket update date**
        - **Ticket close date**
        - **Hardware Stable Identifier** (e.g., node ID)
        - **Hardware category/type impacted** (e.g., GPU, fan, interconnect)
        - **Maintenance/Error/fault description** (some short description of the issue)
        - **Action**: Categorization of action (e.g. repairs done on faulty GPUs to resolve the fault)
        - **Provider Account ID**
        - **Ticket ID**
        - **Node Handover Date** (Date when the node was deployed in Production)
- **Diagnostics:**
    - Identify serial numbers of installed hardware (chassis, baseboard, network adapters, CPU, GPU, etc). Obfuscated but stable identifiers are also OK.
    - Inspect firmware versions of compute nodes and NV switch trays.
    - Obtain nodes' kernel log messages available via BMC to aid in diagnostic of faulty nodes.

Telemetry Requirements

The telemetry requirements are comprised of two core components that require alignment between DGX Cloud and the NCP:

1. **Delivery Method:** *How* telemetry will be delivered by NCP to DGX Cloud for ingestion
2. **Telemetry Scope:** *What* telemetry the NCP will deliver to DGX Cloud

Delivery Method

NCP shall deliver all required telemetry, including metrics and logs, in a manner that allows for ingestion into DGX Cloud systems. The preferred methodology is natively via the OpenTelemetry Protocol with a latency of no longer than 120 seconds.

Telemetry Scope

DGX Cloud will provide the NCP with a detailed specification document with the required metrics and logs. Upon receipt, the NCP shall be required to provide a formal written response detailing the following:

- Confirmation of its ability to deliver the specified metrics and logs.
- Projected timelines for delivery.
- Specific technical details, including metric names, label names, and label values.

Network Telemetry

The NCP shall provide network telemetry across the following domains:

- North-South (Front-End) Network (client-facing and external interconnects)
- East-West (Back-end) Network (GPU/GPU interconnects)
- Management Network (control plane and orchestration traffic)
- NVSwitch Fabric (intra-node GPU switching, applicable for only GB200 and beyond clusters)
- Host Network (NIC-level and server connectivity)

Logs

DGX Cloud will require the NCP to provide logs from various network technologies, including but not limited to:

1. Fabric Manager logs for the NVLink domain *(where applicable)*
2. Subnet Manager logs for the NVLink domain *(where applicable)*
3. UFM Event logs
4. General Switch Logs
5. Switch syslogs
6. Switch kernel logs
7. BMC SEL logs
8. syslogs

Storage Requirements

NCP must provide shared storage solutions (where applicable) that are manageable via standard APIs and UI, including auditing rights for NVIDIA access.

Home Directory Storage

- **Quota Feature:** Configurable filesystem-wide limit, default user/gid quota settings, and per uid/gid overrides.
- **Accounting:** Usage accounting for uid/gids must be available when the feature is enabled.

Requirement Area	Description
File Service uid/gid Quota feature	Configurable filesystem-wide limit, default user/gid quota settings, and per uid/gid overrides available. Usage accounting for uid/gids when the feature is enabled.
Must be NFS storage	• BCM requires NFS protocol shared storage to work • Access control based on DLs requires POSIX

High-Speed Storage Service Requirements

Requirement Area	Description
Provisioning APIs	Storage provisioning may be via vendor portal/API or NCP portal/API.
Performance (QoS)	Must provision needed throughput requested for minimum bandwidth and IOPS.
Integration	K8s: CSI support Breakfix API required to report storage issues
Quota Support	Ability for quota limits to be set on specific user workloads / volumes
Upgrade, maintenance	Provider / NCP initiates desired maintenance. NVIDIA can schedule actual maintenance and can defer maintenance up to 2 weeks. Upgrades should be non-disruptive.

High-Speed Storage Filesystem Requirements

Requirement Area	Description

Parallel high speed filesystem	Parallel or multi-path high-speed filesystem that supports scaling to thousands of simultaneous clients while sustaining requested performance.
Single file system size	It must be possible to allocate a file system of at least 1 PiB even if the initial request is less. Growing to > 10PiB as cluster size increases. This hard requirement may be higher for a specific site and if so will be communicated via the ancillary services document.
Multiple Filesystems (fungible total capacity)	Can have >1 filesystem within our total capacity. Minimal file system size <= 50 TiB.
Filesystem expansion	Live file system expansion is supported, in terms of capacity, inodes, IO performance, and metadata operations performance. Performance should scale linearly with capacity.
Client	Ability to describe your client: In-Kernel, userspace, or bare-metal client installation requirements. Support integration with client kernels / OS used by NVIDIA, as needed. DKMS-enabled packages available for Ubuntu 20.04, 22.04, and 24.04-based operating systems. ARM64 versions compatible with GB200-ready kernels are mandatory, e.g. Linux 6.8.x. Managed Storage Service Provider will provide client configuration best practices and configuration guidelines for filesystem options and kernel module configuration to reliably achieve optimal performance on ARM and x86_64-based clients.
Quota (User, project and group)	Must support soft and hard quotas - uid / gid / project(directory)-id quotas with enforcement.
Root-squash	Nvidia needs to be able to enable or disable and manage root-squash at any time.
flock	It must be possible to mount the file system with flock.
Ability to Audit Changes	Enable Nvidia to have access to changelog data for filesystem auditing and detailed user operations tracking. Tracking by uid/gid, create files, create dirs, rename files, rename dirs, delete files, delete dirs.
HA	All services are required to tolerate any critical component failure in the backend and provide continued client access to all storage services in such cases.

Multi-Node Coherency	One second or less for client attribute and dentry cache updates/invalidates.
Client Multipathing	Clients must have multipathing to all storage servers.

Data Movement Systems Requirements

The Data Movement system is used to copy data from an external data source (NVIDIA,other Cloud,etc) to the NCP data center.

Requirement	Description
Dedicated K8s Cluster for Data Movement stack	Provider managed k8s cluster (or ability to stand up our own).
Data Mover Nodes (CPU)	Dedicated CPU nodes for running data mover - needs high performance networking (exact quantity will be communicated via ancillary services doc).
Access to same GPU storage	Same filesystem as mounted on GPU nodes mounted on the Data Mover nodes (or ability to mount the same filesystem via CSI) .
Access to nvidia corp net	Dedicate link (as described in network transport) to NVIDIA corp net, preferably with vpn, but otherwise with stable IP for allowlisting.
Stable egress IP	Stable IP to IP allowlist access to Nvidia services. (e.g. similar to NAT Gateway).

DGXC-Managed Storage System Deployment

For scenarios where the storage system software will be deployed and managed by DGXC rather than the NCP, the following requirements apply. These requirements enable DGXC to operate storage systems (such as high-speed parallel filesystems, capacity object storage, or block storage) using NCP-provided infrastructure while maintaining operational control.

Host Provisioning and Lifecycle

Requirement Area	Description
Operating System Support	NCP must support a workflow that allows DGXC storage operators to integrate vendor-provided or storage-specific operating system images via bare-metal or VM provisioning for storage servers. The workflow must: (a) Allow DGXC to deploy custom OS images (e.g., vendor-enhanced kernels for Lustre, Rocky Linux, Ubuntu 20.04/22.04/24.04).

Requirement Area	Description
Drive Sanitization Policy	Cryptographically erase data drive contents between storage system tenants with full attestation of host firmware. Must support an optional flag to skip drive sanitization during break/fix flows (e.g., power supply replacement) where tenancy does not change. Critical hardware component replacements may require sanitization without override, this is inclusive of GPU / CPU node local storage.
Stable IP Assignment	Storage nodes must support static IP addressing that remains stable during host lifecycle operations and does not reset between maintenance events.
Out-of-Band Failure Detection	NCP must provide the ability to detect system failures out-of-band, including device, network, memory, and drive failures, enabling DGXC to proactively respond to hardware issues.
Topology Observability	NCP must provide visibility into failure domains to enable DGXC to provision storage nodes with physical diversity. Storage systems must be able to provision nodes that purposefully span failure domains for resilience.
BlueField/DPU Support	For storage systems utilizing BlueField-based architectures, the host provisioning system must support lifecycle management and specific configuration requirements for BlueField "JBOF" systems that export NVMe-oF to hosts.

Network Transport and Fabric Visibility

Backend Switch Fabric API

The purpose of this API is to expose sufficient information about the cluster's network topology to enable efficient scheduling, placement, and optimization of multi-node GPU workloads. Understanding the network hierarchy between compute instances and switches, as well as intra-node NVLink domains, is essential for minimizing communication latency and maximizing throughput. Thus, this applies to North-South, East-West, and NVLink networks (not MGMT). See the appendix for a DGXC recommended reference implementation.

For each compute node, the API must provide visibility into the backend network switches connecting the node to the core.

- **Identification:** Each switch must be identified by a unique, stable identifier. A "switch" may represent a physical switch or a logical connectivity domain.
- **Structure:** API may be gRPC or REST. Response structure may include multiple nodes (pagination expected).
- **Topology:** Switch info can be returned as an ordered array of IDs (e.g., leaf, spine, core) or separate fields for each tier.

NVLink Domain API

- **Requirement:** For compute nodes supporting NVLink (e.g., GB200, GB300, Vera Rubin), the API shall return the unique identifier of the NVLink domain associated with each node.
- **Implementation:** Can be a separate API method or part of the Backend Switch Fabric API.

Transport and Networking requirements

Non-Conflicting IP space allocation for the DGXC cluster

Purpose:
Ensure DGXC GPU clusters deployed in NCP can access the NVIDIA DGXC/CorpIT network directly via routing exchange. DGXC Cluster IP address must be non-conflicting with existing NVIDIA private IP space.

Requirements:

- Bring Your Own IP (BYOIP): NCP shall support the ability for NVIDIA to bring and allocate its own IP private address space for DGXC GPU clusters.
- NCP shall provide a possibility to create static IP allocations that persist across instance restarts and re-creations. That includes floating IP allocations.
- DoD space: NCP shall support allocation and use of the 7.0.0.0/8 IPv4 address space for DGXC GPU cluster deployments. This IP space shall be considered equivalent to RFC1918 addresses
- Routing Support: NCP must support advertising and routing of BYOIP prefixes within the NCP environment and across interconnects (Private Cloud Interconnect, IPSec, etc.)

Connection to NVIDIA CorpIT Network

Purpose:
Provide connection from DGXC GPU clusters within NCP to NVIDIA CorpIT for internal Command and Control and admin access.

Requirements:

- **Bandwidth:** Low bandwidth (Up to 10Gbps).
- **Transport:** Private Cloud interconnect + VIF + BGP (preferred for better performance/security). DGXC will establish connectivity to NCP through a mutually agreed Point of Presence (POP) using Private Cloud Interconnect, functionally equivalent to AWS Direct Connect, GCP Dedicated Interconnect, Azure ExpressRoute, and OCI FastConnect. Connectivity will be provisioned with a Virtual Interface (VIF) and routing established via BGP. The interconnect will be used to exchange private IP space (RFC1918, as well as 7.0.0.0/8) between DGXC and NCP.

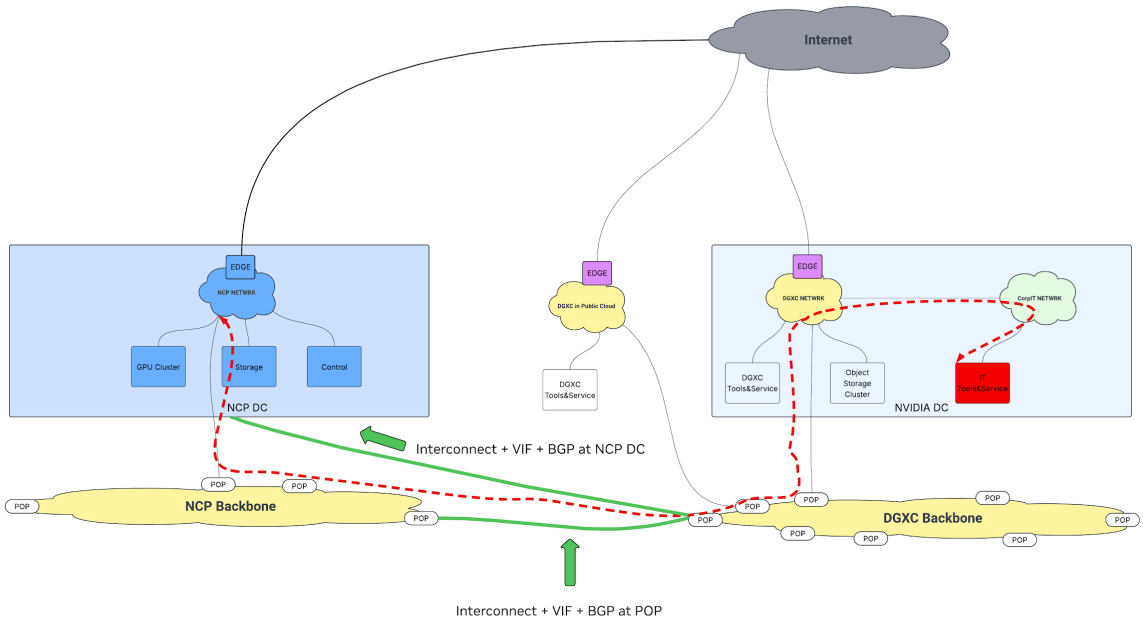


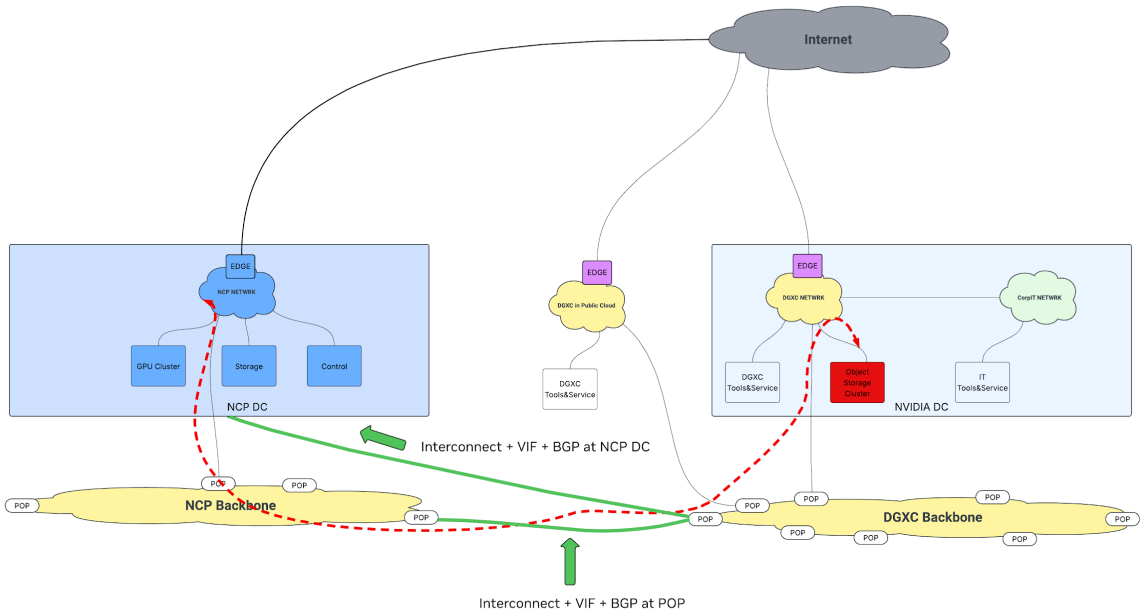
Figure: Private Cloud interconnect + VIF + BGP for CorpIT access

Connection to DGXC Storage

Purpose:
Enable high-bandwidth, end to end MACsec-encrypted (fail-closed) access between the DGXC GPU clusters within NCP and NVIDIA DGXC on-premises object storage for large-scale data movement.

Requirements:

- **Transport:** Private Cloud interconnect + VIF + BGP (preferred for better performance/security). DGXC will establish connectivity to NCP through a mutually agreed Point of Presence (POP) using Private Cloud Interconnect, functionally equivalent to AWS Direct Connect, GCP Dedicated Interconnect, Azure ExpressRoute, and OCI FastConnect. Connectivity will be provisioned with a Virtual Interface (VIF) and routing established via BGP. The interconnect will be used to exchange private IP space (RFC1918, as well as 7.0.0.0/8) between DGXC and NCP.



Cluster Local Internet Access

Purpose:
Provide general Internet access from DGXC GPU clusters within NCP to Internet, including NVIDIA DGXC hosted services on third-party public cloud services.

Requirements:

- **Cluster Internet access:** Egress NAT IPs should be a static pool dedicated to only Nvidia Cluster/Tenancy/VPC. These persistent IP addresses must be used exclusively for DGXC traffic and shall not be shared with or carry traffic from other NCP tenants.

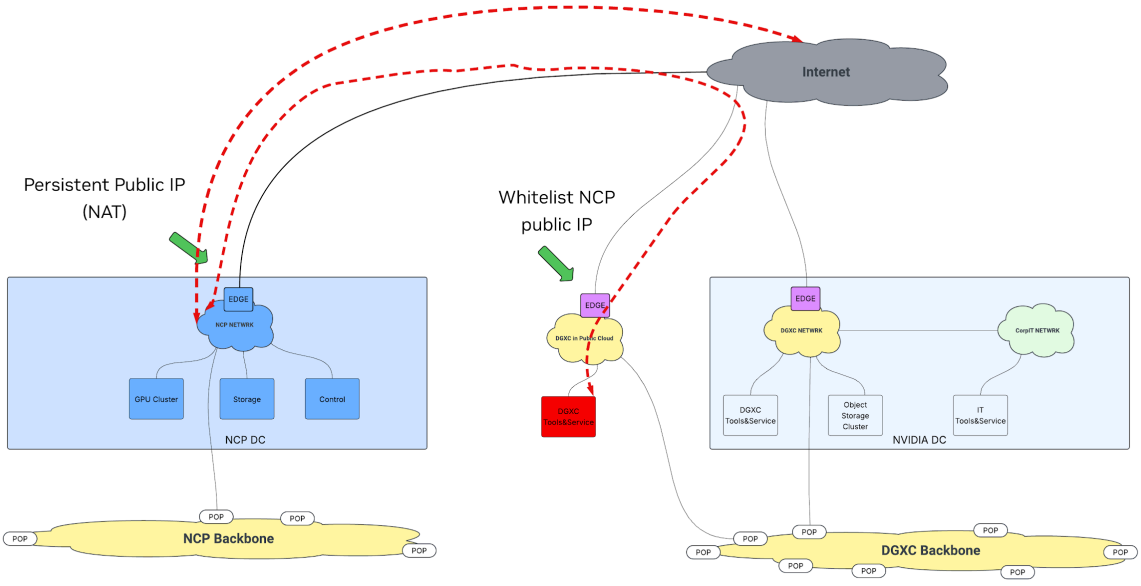


Figure: Public internet for DGXC hosted Services access

- **Availability:** Must support redundant upstream paths to ensure connectivity under failure.

Capacity and Fleet Management

This section defines the essential metrics required for standardized monitoring and reporting of fleet health in partner engagements to support operations and contractual SLAs.

Required Governance Metrics
The core metrics needed to track fleet health are:

- **Delivered:** Nodes/GPUs provisioned and available to NVIDIA, allocated to a specific account/project/tenant.
- **Healthy:** Nodes/GPUs functioning and meeting SLA requirements, allocated to a specific account/project/tenant.
- **Reserved:** Resources allocated to a specific account/project/tenant.
- **Total Active/In-Use:** Nodes/GPUs currently in use within a specific account/project/tenant.

Resource Governance API Metrics
The Resource Governance API must return the following information for each node:

- **Node ID** (Unique identifier for a GPU node)
- **Health State** (Healthy/Unhealthy classification)
- **Instance ID** (Identifier for virtual workload)
- **Creation Timestamp** (Time workload/node was created)
- **Hardware Type** (Descriptor for the hardware model)
- **GPU Count** (Number of GPUs per node)
- **Top-levelAccount/ID** (Identifier for the top-level organization/account)
- **Sub-LevelProject/ID** (Identifier for the nested project/sub-account)
- **In Use** (True/False status indicating if the GPU Node is turned on and in use)
- **Region** (Region of the data center where nodes are deployed)

Resource Discovery APIs
It is not acceptable to have capacity be “handed” to DGXC through a phone, slack or email message. For example, when cluster first comes online, nodes/racks are likely being handed off weekly (or more frequently). Instead, please provide the following mechanism (and we can poll):

- **Programmatic Capacity Discovery:** All newly delivered capacity must be discoverable via a centralized API. This "Resource Index" must provide a resource stable identifer and some information on why it’s being provided (e.g. capacity fulfillment on gb300 project, break-fix / RMA return to cluster, etc).

Logical Compartmentalization and Resource Isolation

To ensure performance consistency and security, the NCP must support strict logical and physical isolation of NVIDIA’s reserved capacity.

- ***Capacity Reservations:*** *A mechanism to logically group and "pin" a set of resources (compute, network, storage) to accounts (or equivalent constructs) in an NVIDIA tenancy*
- ***Atomic Allocation:*** *Support for reserving a "topology block" as a single unit, ensuring all resources in that block share identical performance characteristics and security boundaries.*

Unified Health and Lifecycle APIs

NVIDIA requires a "single source of truth" for the health of both physical hosts and logical compute primitives.

- **Per-Host Health:** Real-time API access to the health bits of physical hardware (GPU state, thermal status, memory health).
- **Primitive-Level Status:** Health aggregation at the cluster, nodegroup, or reservation level to identify broad infrastructure failures (e.g., a spine switch failure affecting a whole block).

Appendix

This section contains links to reference documents and implementation guides to provide additional details if NCPs need them.

Implementation Guidance

Reference documents provide additional information on implementing some of the above requirements.

1. Network Topology Discovery: https://github.com/NVIDIA/topograph
2. Exemplar Cloud Website: https://www.nvidia.com/en-us/data-center/ai-cloud-performance/

Other Feature Considerations (Not Required)

1. Disk Cloning: Disk cloning capability (for network-attached block devices). It should be possible to clone a disk even on a running instance.