Institute for the Wireless Internet of Things

at Northeastern University

GPUs Overview

Ali Saeizadeh

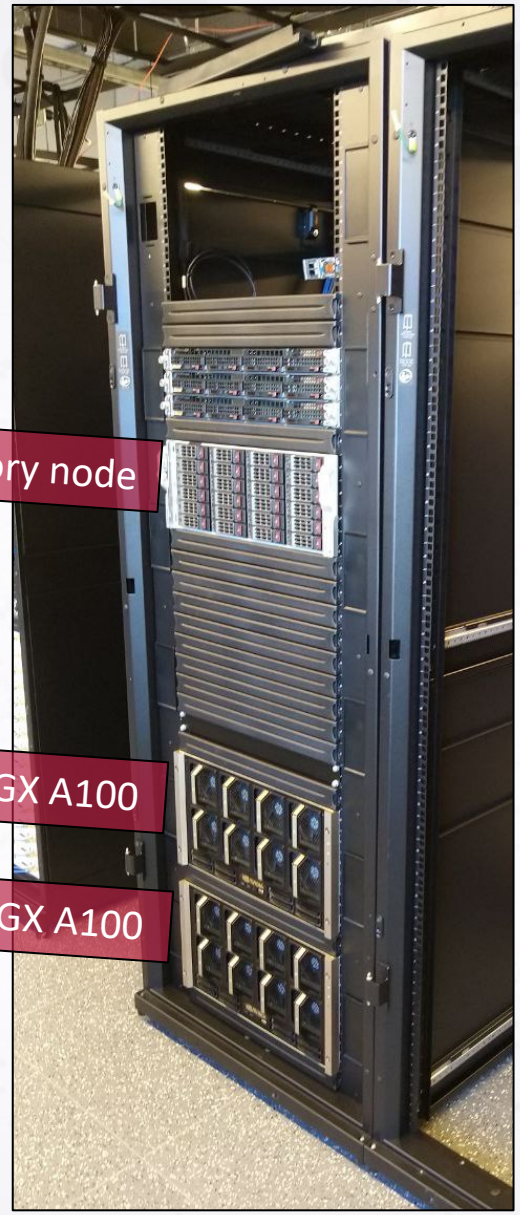
AI Jumpstart Integration

MASSACHUSETTS TECHNOLOGY COLLABORATIVE

- Build GPU-accelerated software-defined, cloud-native applications for the 5G vRAN
  - 2x NVIDIA DGX A100 (8 GPUs each, 40GB RAM)
  - 1x Large memory node (6 GPUs, 3TB of RAM)
  - 1x MLX InfiniBand switch

8 NVIDIA A100 GPU 40 GB

Mellanox InfiniBand switch



Large memory node

NVIDIA DGX A100

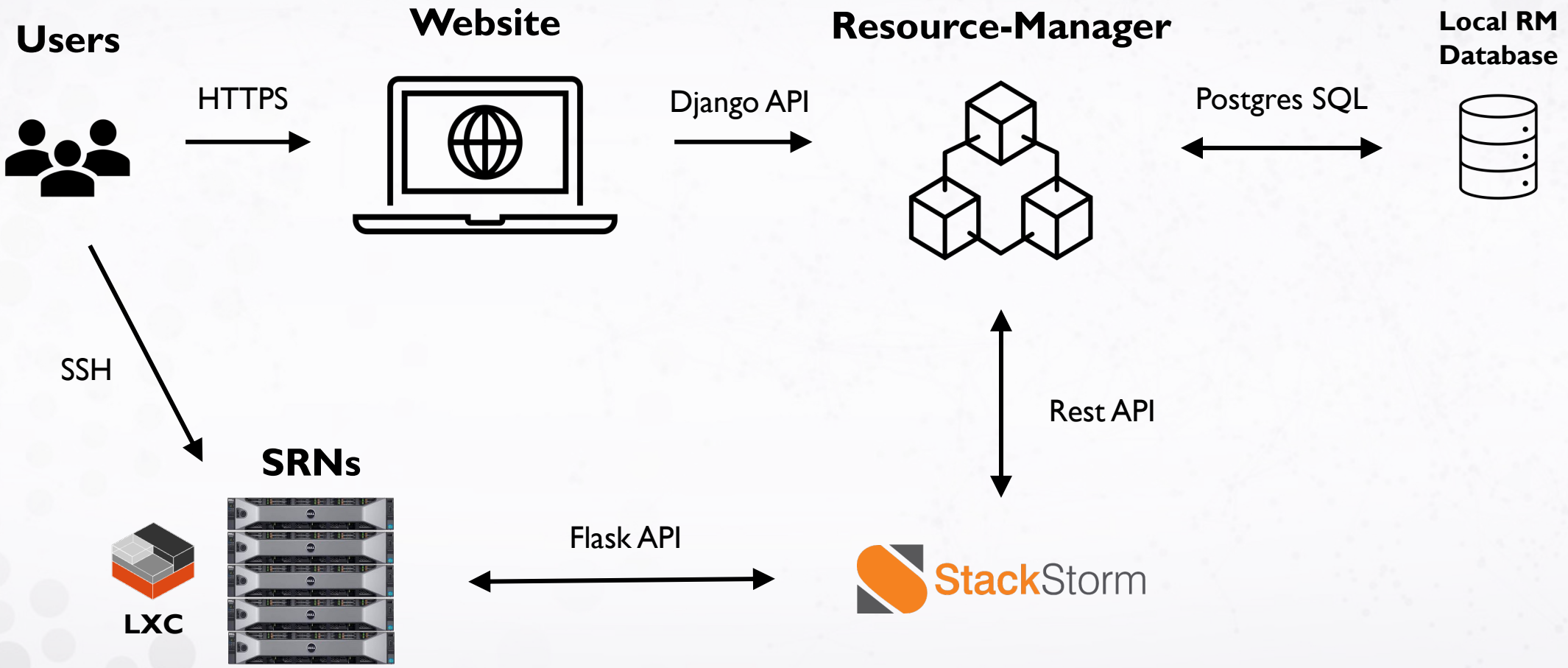
NVIDIA DGX A100

How do we **integrate** the new rack into Colosseum?

HashiCorp Nomad

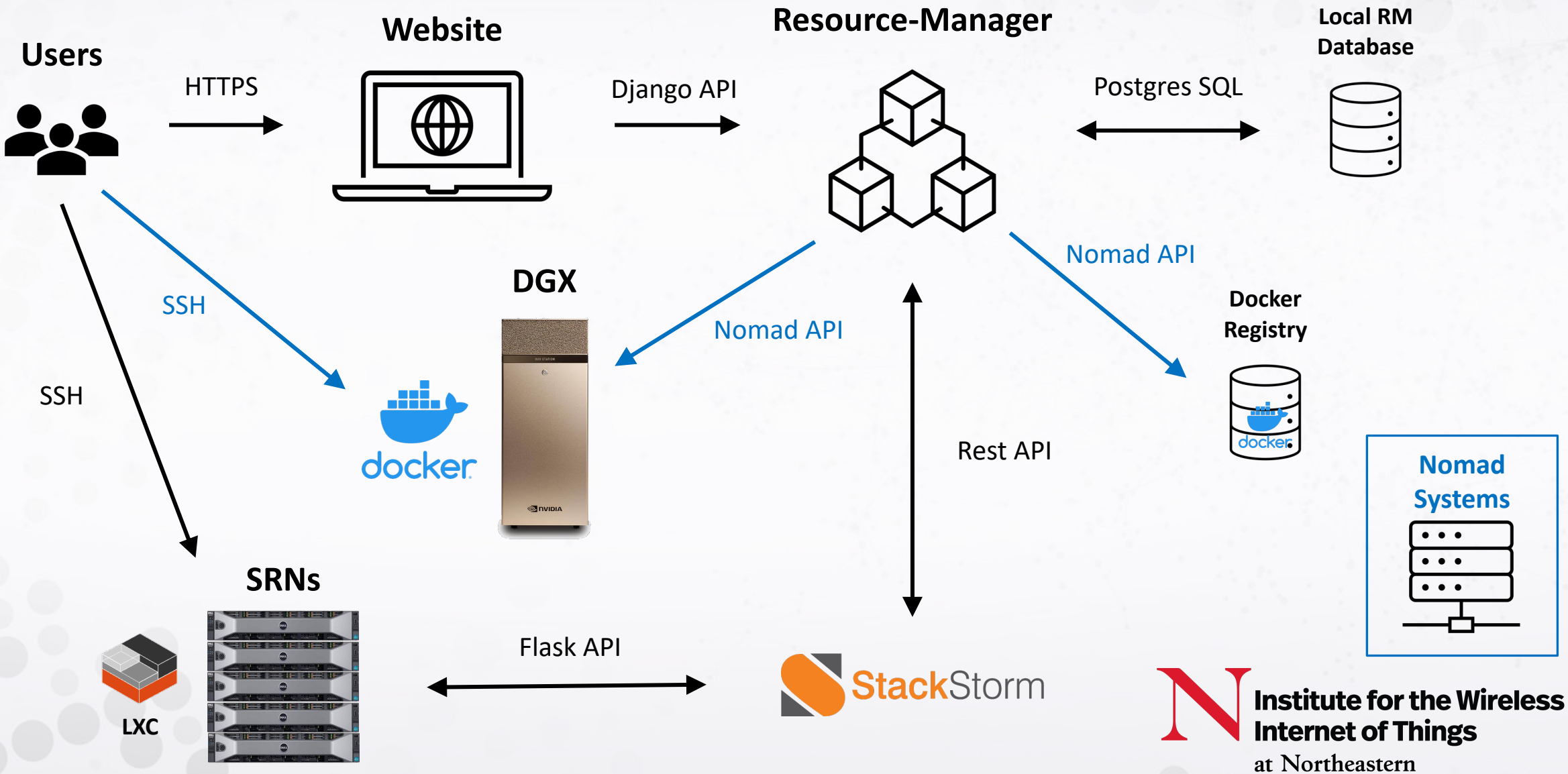
Institute for the Wireless Internet of Things at Northeastern

Current SRN reservation flow diagram



Institute for the Wireless Internet of Things at Northeastern

New GPU Reservation Flow Diagram



Nomad: Overview

HashiCorp Nomad

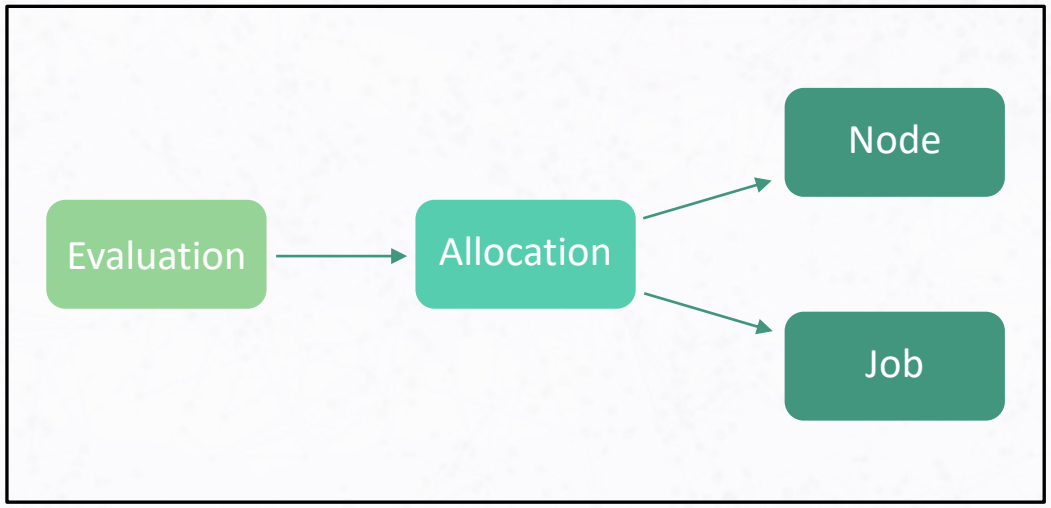
- Workload **orchestrator** to deploy and manage containers.
- Built-in support for **GPU** workloads and device plugins.
- **HashiCorp** ecosystem (Vault, Consul…). Vault Consul
- Compare to the more popular kubernetes:
  - Smaller scope
  - More simple and lightweight
  - More scalable (5k vs 10k nodes)
  - Flexible workload support
  - Consistent deployment

Institute for the Wireless Internet of Things at Northeastern

Nomad: Working principles

HashiCorp Nomad

- Interface with Nomad via **RESTful** HTTP API.
- 5 main Nomad concepts:
  - **Job**: declarative description of tasks.
  - **Node**: servers to schedule tasks.
  - **Allocation**: tasks and nodes mapping.
  - **Deployment**: allocations trackers.
  - **Evaluation**: scheduling and status.



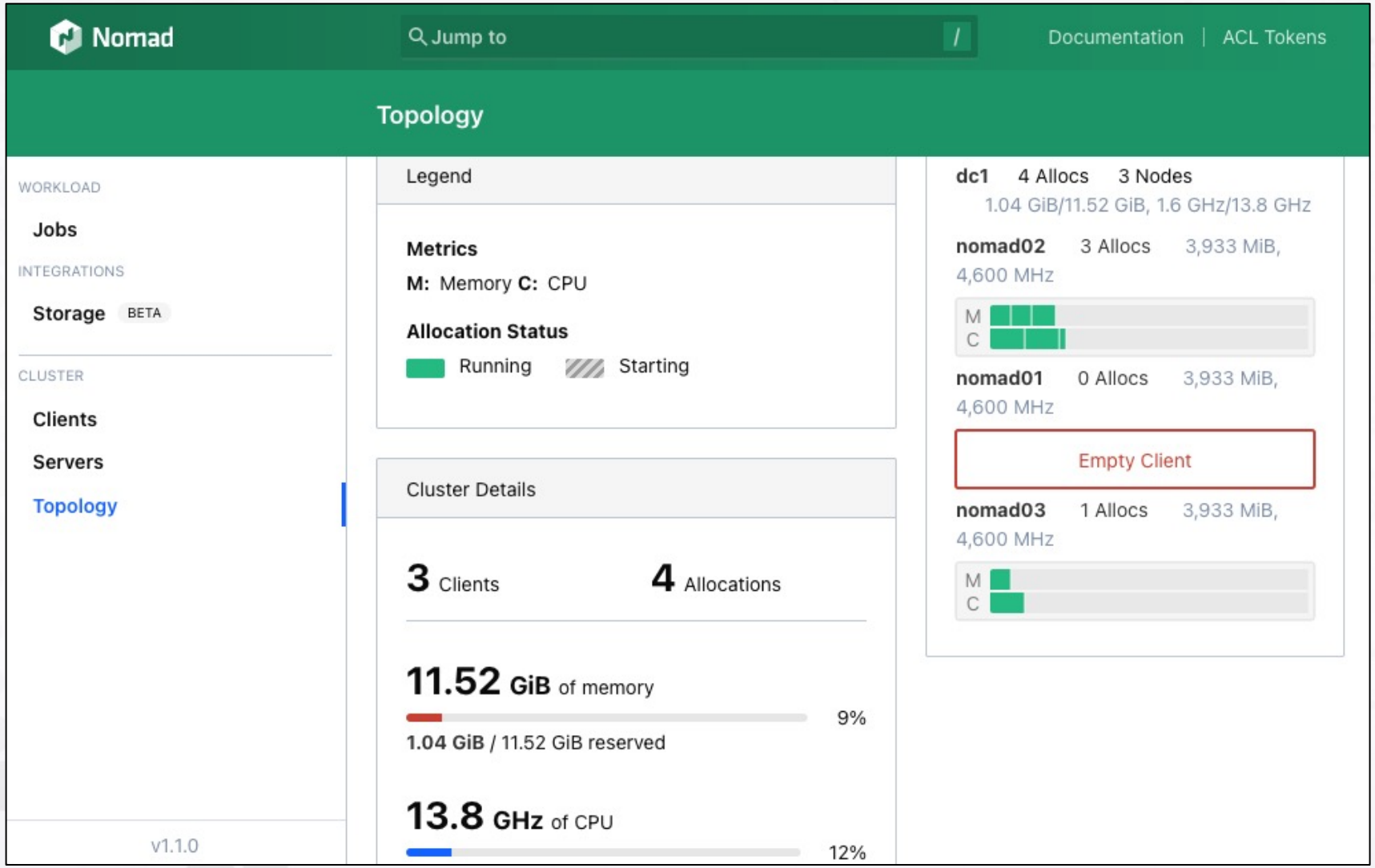
- Support of tokens and ACLs for **authentications**.

More info: *https://www.nomadproject.io/api-docs*

Institute for the Wireless Internet of Things at Northeastern

Nomad: GUI and Job example

HashiCorp Nomad



Example_job.nomad

```
job "gpu-test2" {
  datacenters = ["colosseum"]
  type = "service"
  constraint {
    attribute = "${attr.unique.hostname}"
    regexp    = "dgx1"
  }
  group "smi" {
    task "smi" {
      driver = "docker"
      config {
        image = "nvidia/cuda:11.0-base"
        command = "nvidia-smi"
      }
      resources {
        device "nvidia/gpu" {
          count = 1
        }
      }
      network {
        port "web" {
          static = 81
          to = 5001
}}}}}
```

Nomad is also being considered to run and manage all **Colosseum** containers.

Institute for the Wireless Internet of Things at Northeastern

Nomad Ecosystem Implementation

Main Entities

HashiCorp Nomad

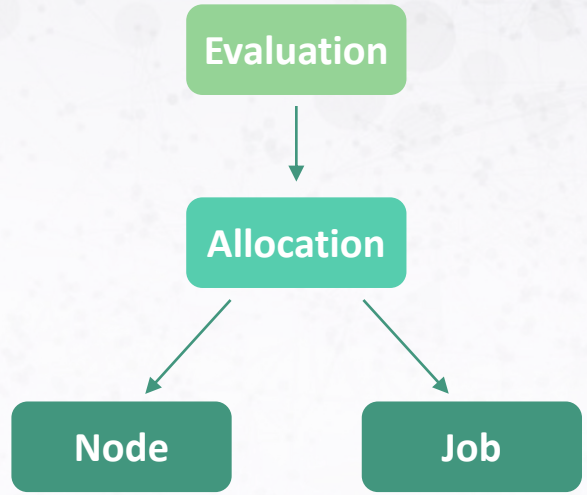
- Workload orchestrator
- Container manager
- Simple, light, scalable

HashiCorp Consul

- Service mesh tool
- Service discovery/check
- Secure connectivity

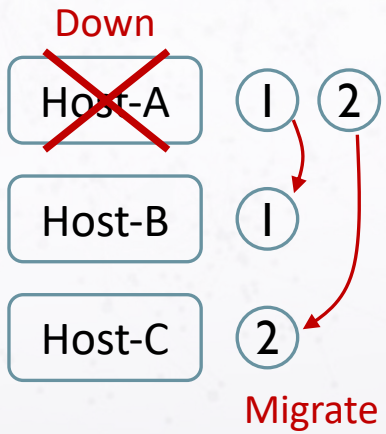
traefik

- Reverse proxy
- Load balancer
- Completely integrated



Environment	Dev	Stage	Prod
Consul Servers	3	5	7
Nomad Servers	3	5	7
Nomad Workers (CPU)	4	4	4
Nomad Workers (GPU)	1	2	2

Servers	3	5	7
Quorum	2	3	4
Failure Tolerance	1	2	3



Institute for the Wireless Internet of Things at Northeastern

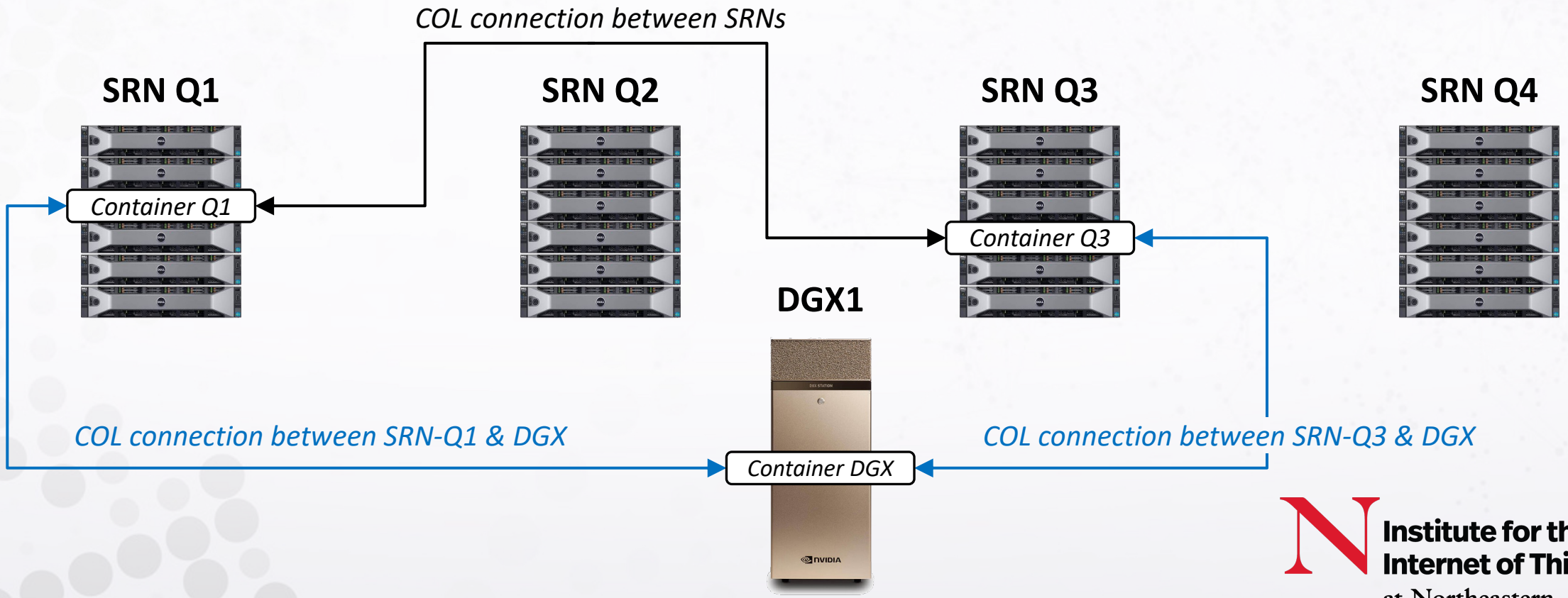
Possible use case examples

1. **Training** of large-scale datasets collected on Colosseum.
   - **Current** data flow: Generate ➔ Export ➔ Train ➔ Import ➔ Use
   - **New** data flow: Generate ➔ Train ➔ Use
   - Much **faster** and more **efficient**.

2. **Real-time** AI-driven signal processing for the full 5G protocol stack.
   - **Online training** for the O-RAN xApps.
   - Efficient **real-time tuning** of several users' parameters.
   - Highly **scalable**.

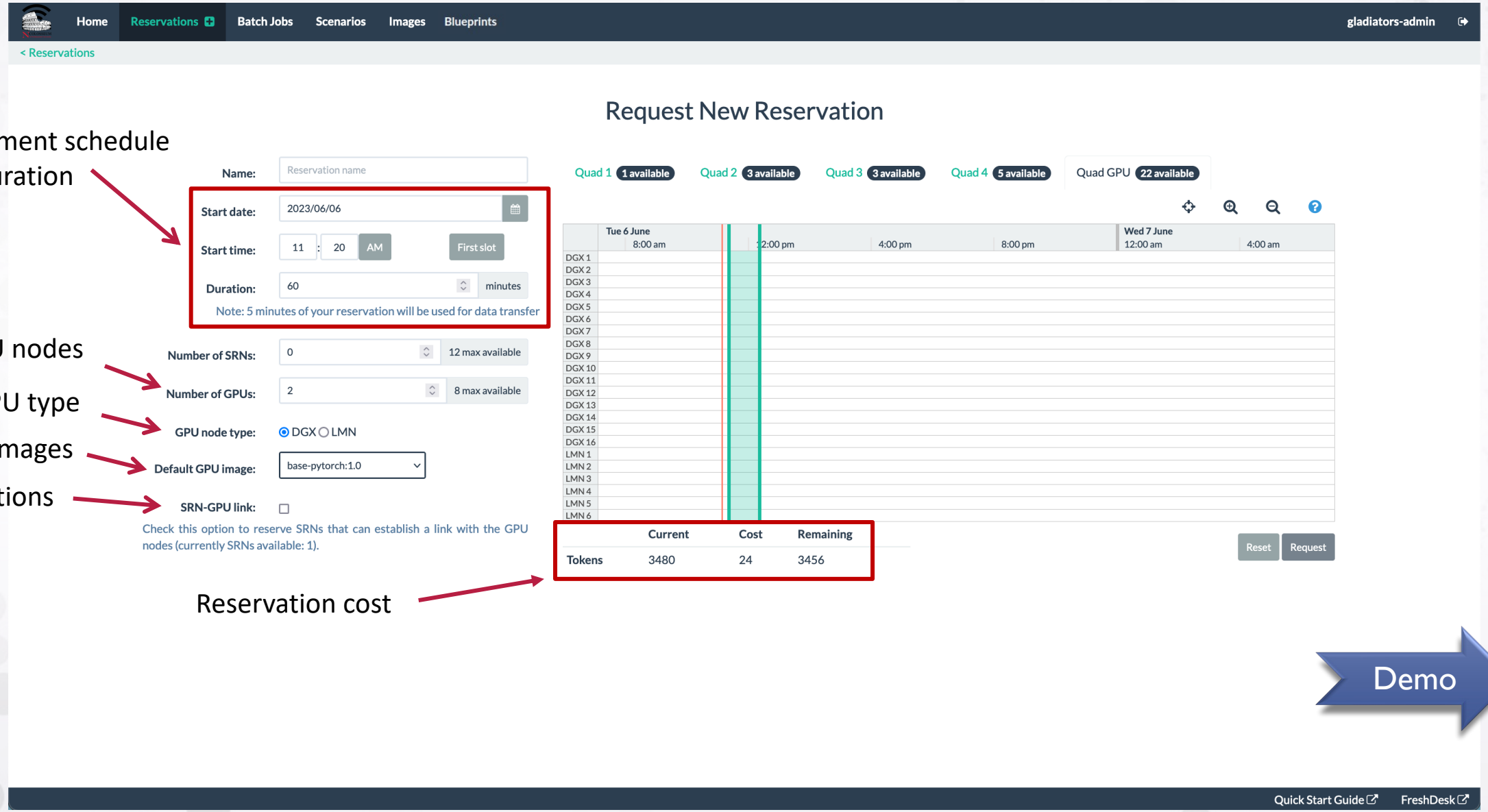
Possible use case examples

3. **Direct link** between USRP and GPU to immediately process IQ samples.
   - Need consideration on constrain **limits** (latency, number of users…).
   - Architectural **challenges** to be solved.



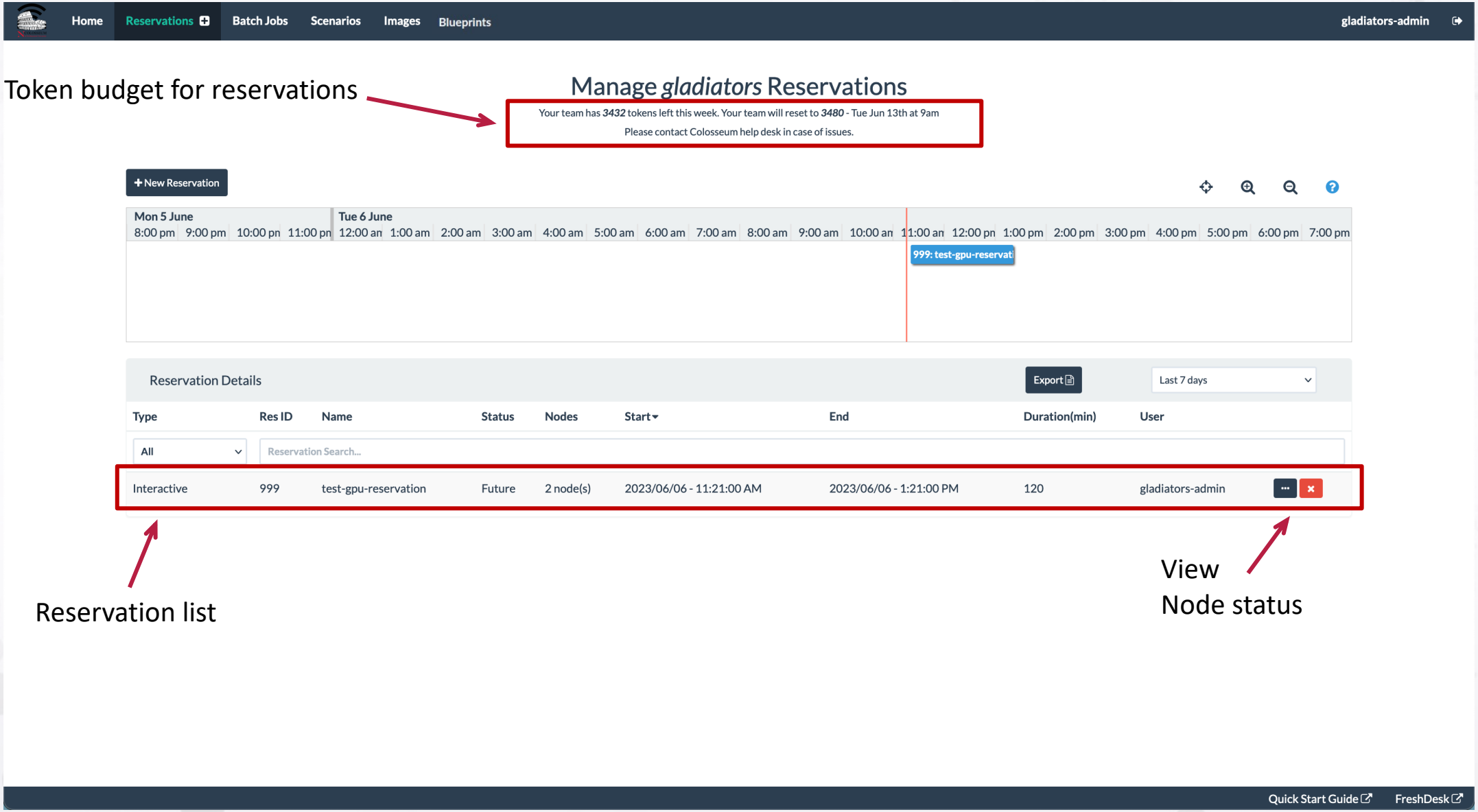
Institute for the Wireless Internet of Things at Northeastern

Start a GPU Reservation

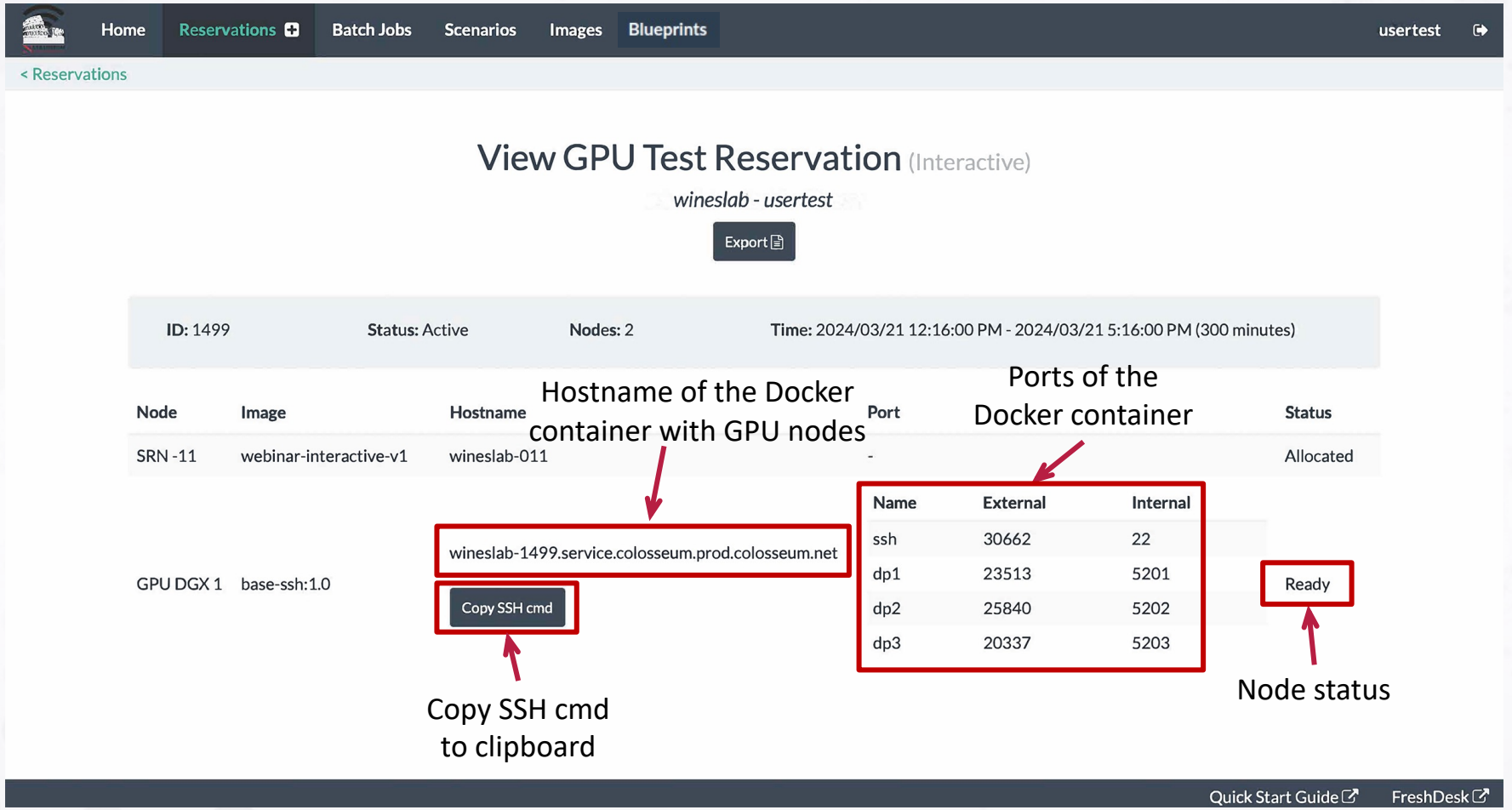


Demo

View Existing Reservations



View Reservation Details

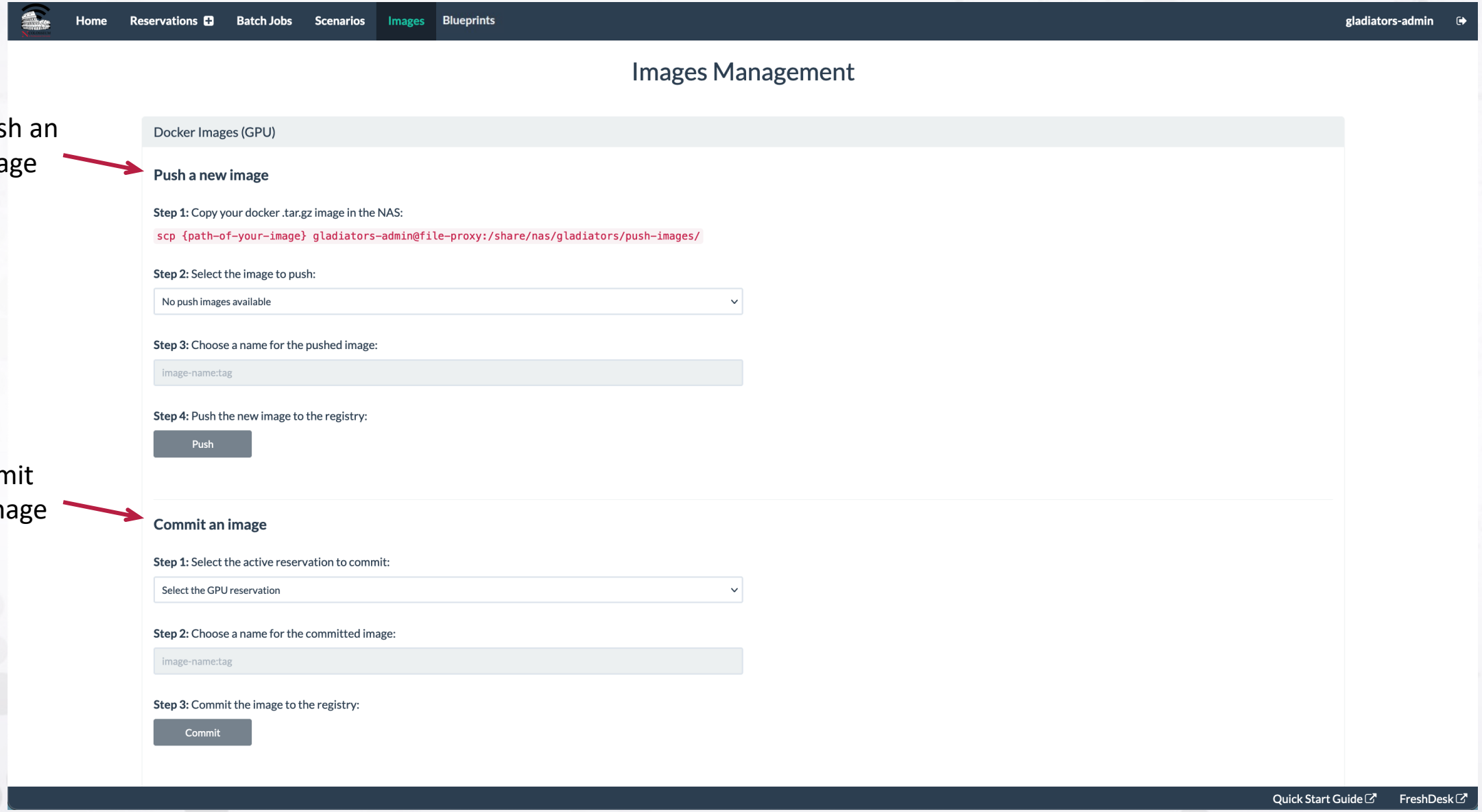


2 types of ports:

- **ssh**: to ssh into the container.
- **dp** (dynamic port): to interact with external apps from SRNs (e.g., iPerf server).

Login example (from your pc): *ssh wineslab-1499.service.colosseum.prod.colosseum.net –p 30662*

Images Management



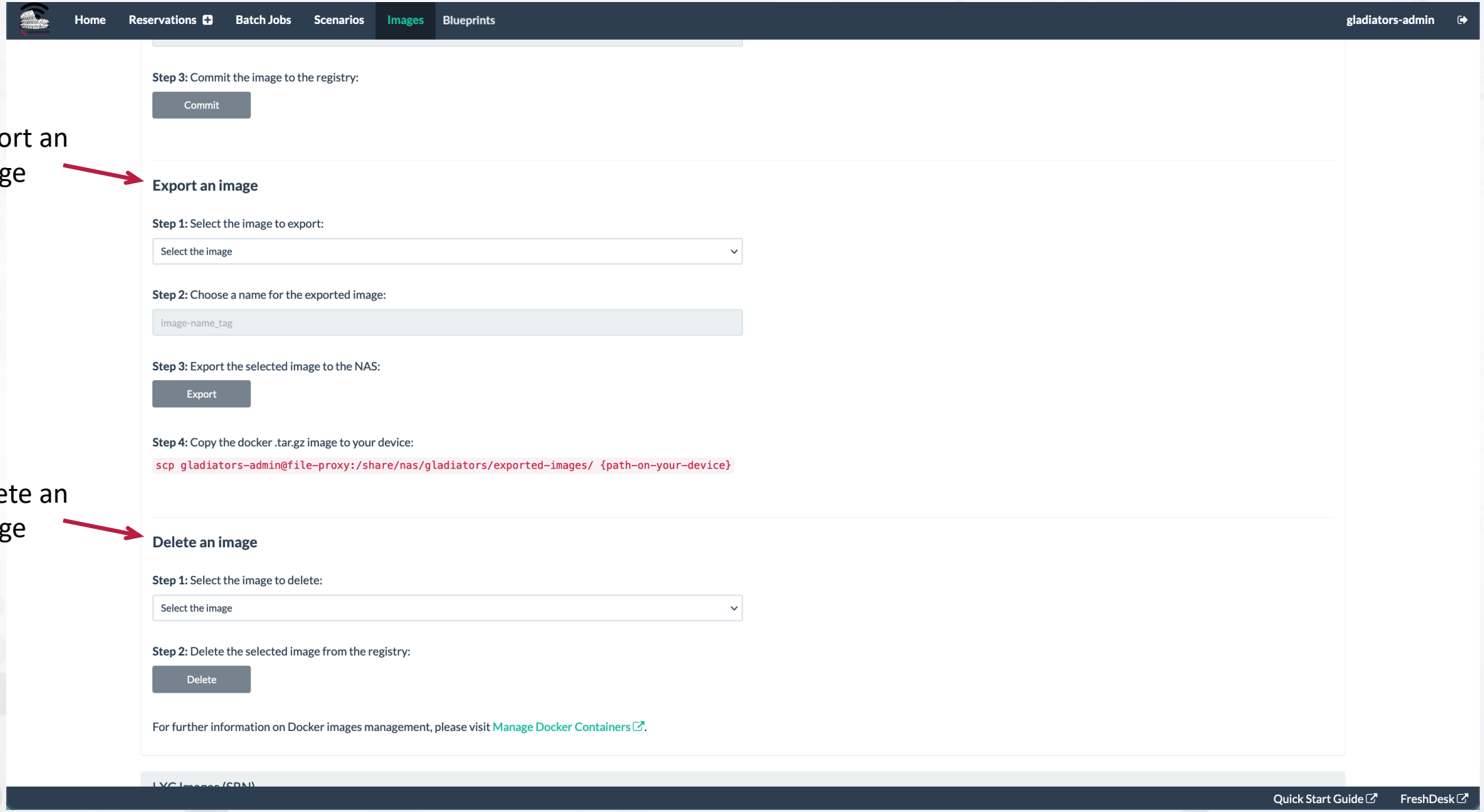
Push an image

Commit an image

Images Management

Export an image

Delete an image



GPU Links to Colosseum Resources and Documentations

- Colosseum Home Page
  - http://colosseum.net
  - https://northeastern.edu/colosseum
- Colosseum Experiment Portal
  - https://experiments.colosseum.net
  - https://experiments.colosseum.net/images
- GitHub Colosseum Sample Dockerfiles
  - https://github.com/colosseum-wiot/colosseum-dockerfiles/
- Freshdesk Guides
  - Logging into a GPU: Link
  - Manage Docker containers: Link
  - Connecting GPU and SRN Reservations: Link

GPU Assignment

GPU Assignment

- We are going to follow an example from Pytorch

Steps to Follow

1. Reserve a time slot for 1 DGX GPU
2. Pick the Pytorch image from common images
3. Connect to the Container using ssh

Steps to Follow

- `wget` the the code and the dataset on your container
- check the `nvidia-smi` for GPUs availability
- Run the code and have fun using `python`
- You can check the results by
  - VS Code / PyCharm
  - scp / rsync

LVX VERITAS VIRTVS

Institute for the Wireless Internet of Things

at Northeastern University

Thank You! (Questions?)

NSF

Platforms for Advanced Wireless Research

DARPA

MITRE

MASSACHUSETTS TECHNOLOGY COLLABORATIVE

COLOSSEUM at Northeastern University