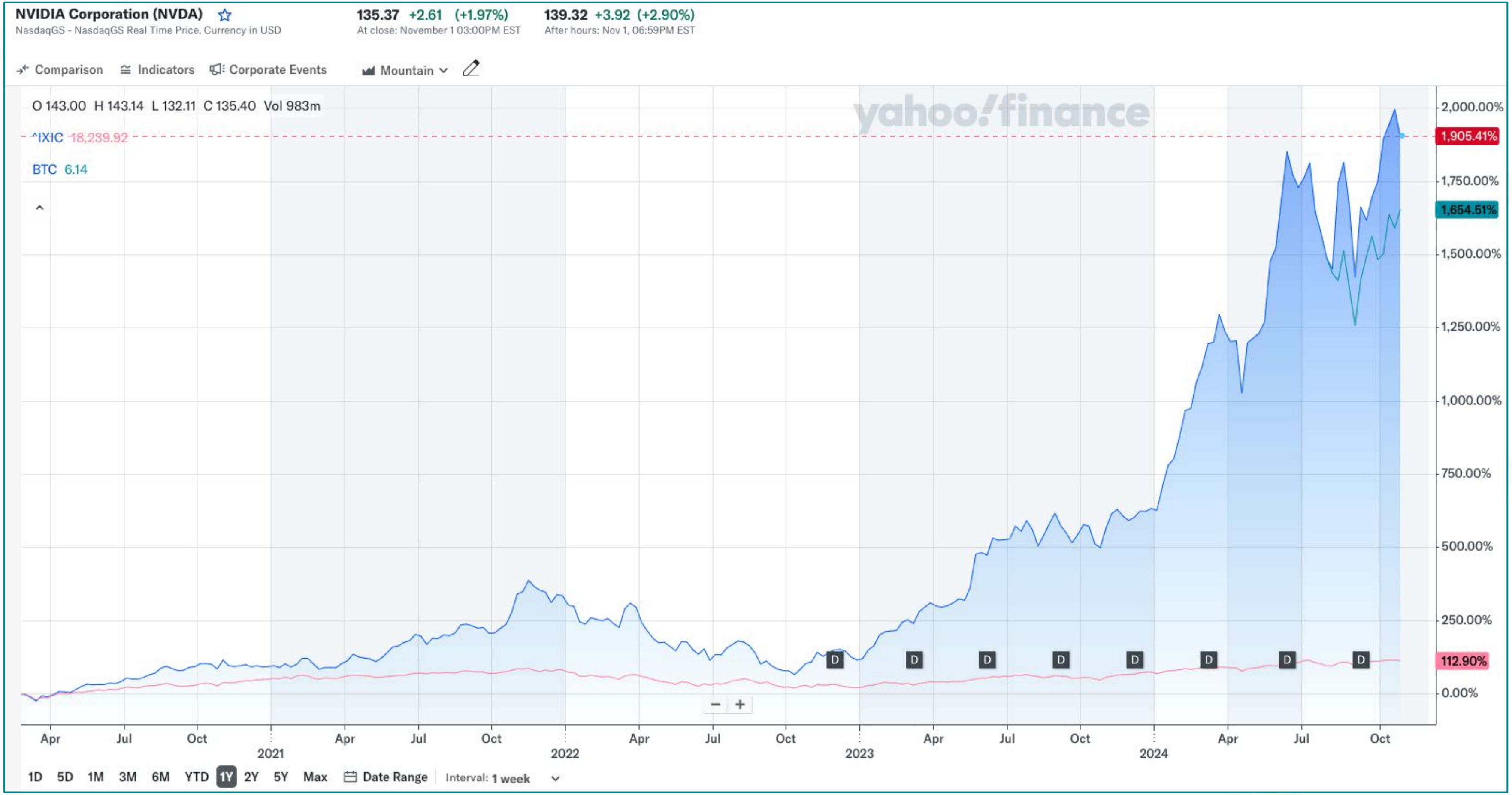
WORKSHOP GARR 2024

NET MAKERS

Cloud privato per l'AI

Un caso d’uso GARR

Alex.Barchiesi @ garr.it



NVIDIA Corporation (NVDA) ☆
NasdaqGS - NasdaqGS Real Time Price. Currency in USD

135.37 +2.61 (+1.97%)
At close: November 1 03:00PM EST

139.32 +3.92 (+2.90%)
After hours: Nov 1, 06:59PM EST

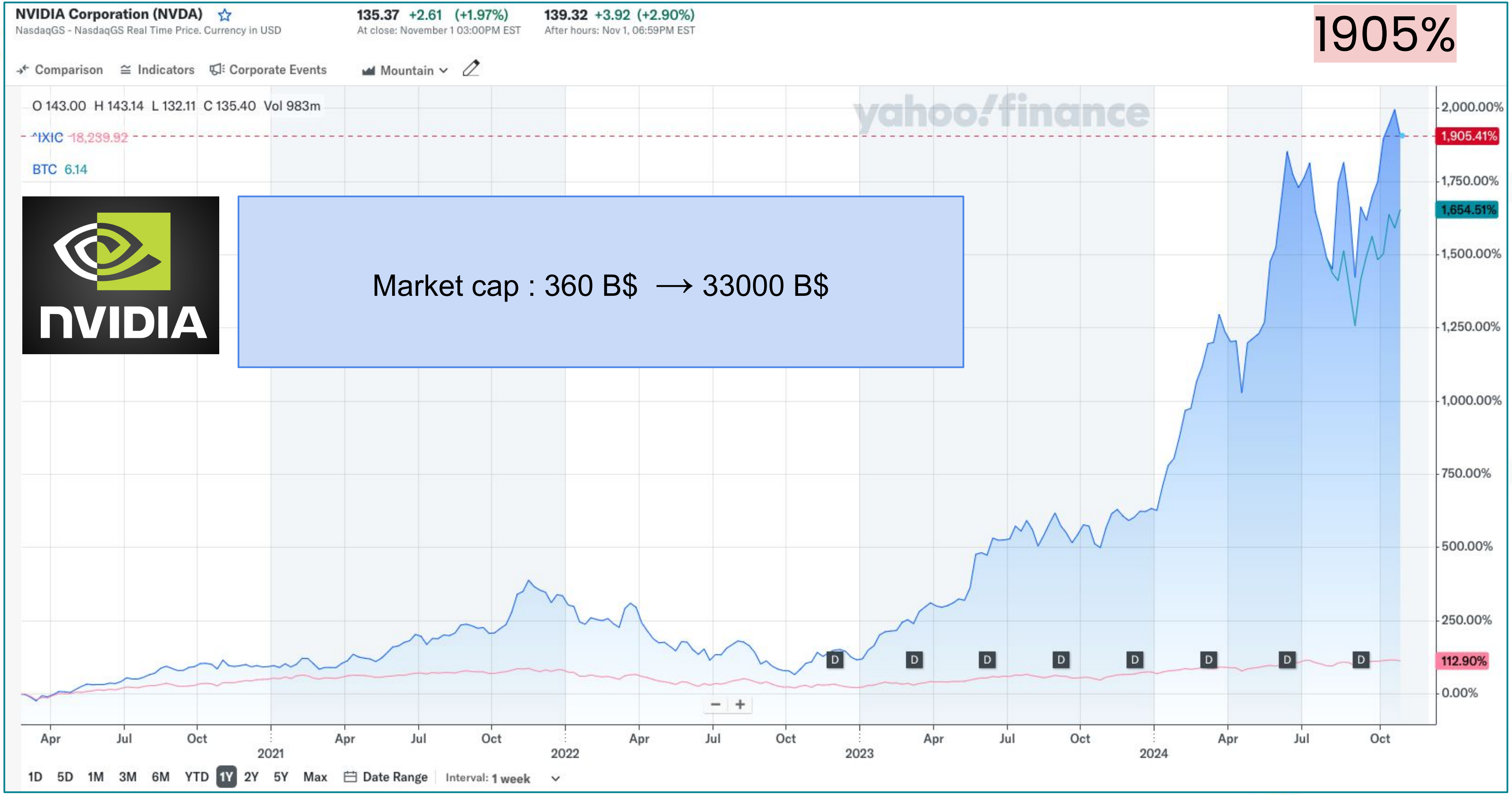
O 143.00 H 143.14 L 132.11 C 135.40 Vol 983m

^IXIC 18,239.92

BTC 6.14

Interval: 1 week

1905%



NVIDIA Corporation (NVDA) ☆
NasdaqGS - NasdaqGS Real Time Price. Currency in USD

135.37 +2.61 (+1.97%)
At close: November 1 03:00PM EST

139.32 +3.92 (+2.90%)
After hours: Nov 1, 06:59PM EST

O 143.00 H 143.14 L 132.11 C 135.40 Vol 983m

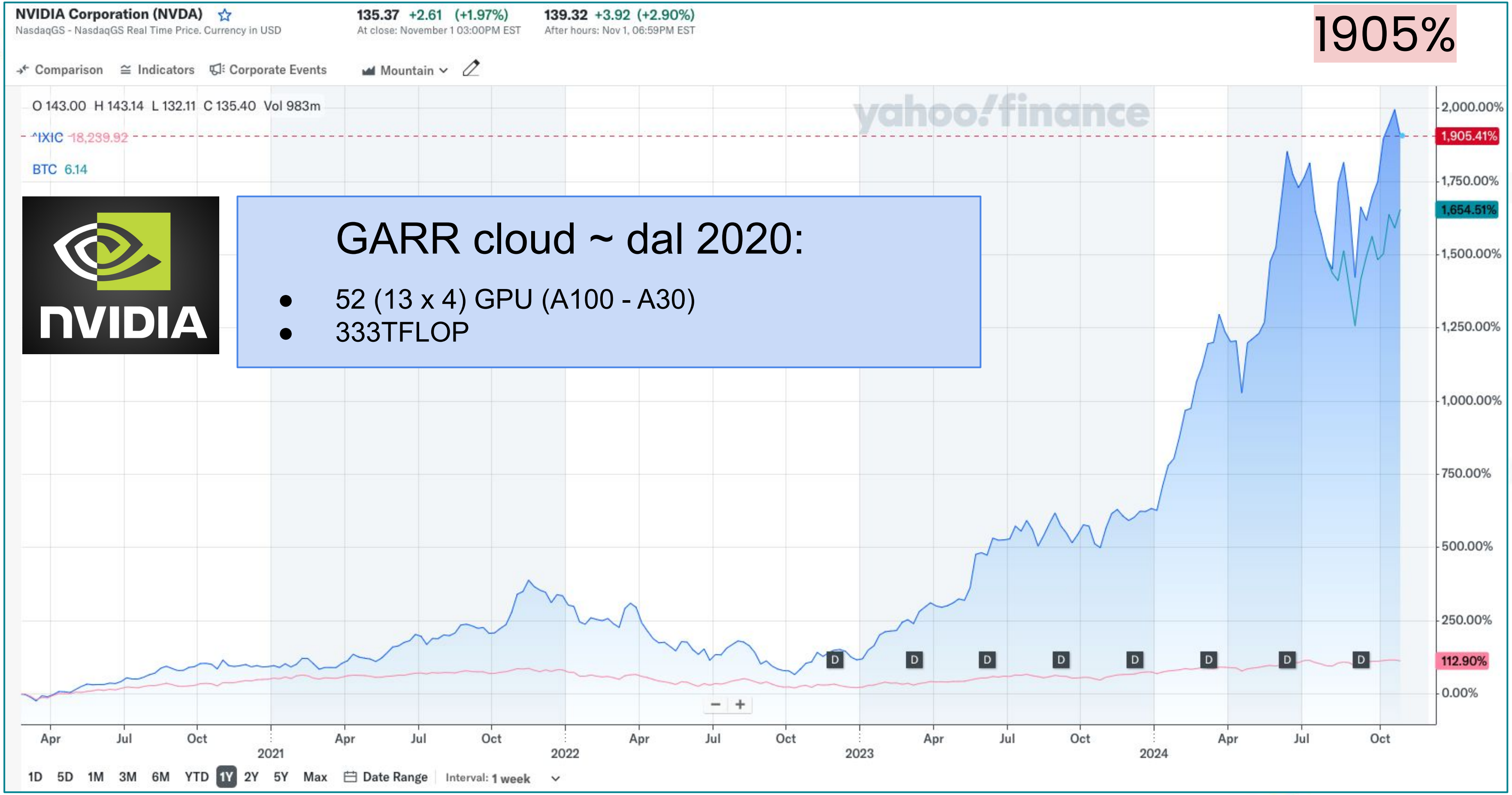
^IXIC ~~18,239.92~~

BTC 6.14

Interval: 1 week

Market cap : 360 B$ ⟶ 33000 B$

1905%



NVIDIA Corporation (NVDA)

NasdaqGS - NasdaqGS Real Time Price. Currency in USD

135.37 +2.61 (+1.97%)
At close: November 1 03:00PM EST

139.32 +3.92 (+2.90%)
After hours: Nov 1, 06:59PM EST

O 143.00 H 143.14 L 132.11 C 135.40 Vol 983m

^IXIC 18,239.92

BTC 6.14

GARR cloud ~ dal 2020:

- 52 (13 x 4) GPU (A100 - A30)
- 333TFLOP

Outline: cloud privato per l'AI

- Perchè - Why?
- Cosa - What?
- Come - How?
- Sviluppi successivi - Then?

Why: GPU - ML - not only a buzzword

- **Network optimization:** ML can **analyze** historical network traffic data to predict demand peaks and allow to adequately size network resources, avoiding congestion and ensuring efficient service. It can also **detect** anomalies in the network, such as cyberattacks or hardware malfunctions, enabling a rapid and effective response. Additionally, ML can be used to **find** the most efficient routes for network traffic, reducing latency and increasing network capacity.

- **Data management:** ML can be used to **classify and group** large amounts of data, facilitating search and analysis. It can also reduce data dimensionality, making it easier to visualize and analyze. ML can be used to identify **patterns** in data, which can be used to discover new knowledge and make informed decisions.

- **Research:** ML can be used to analyze large amounts of **scientific data**, discovering new knowledge and accelerating the research process. It can also be used to develop new mathematical models and algorithms, which can be used to solve complex problems in various fields.

- **Scientific collaboration:** ML can be used to develop tools that facilitate the sharing and analysis of scientific data among researchers. It can also be used to identify potential collaborators for research projects, based on their interests and expertise.

In summary, Machine Learning can offers enormous potential to improve the efficiency, reliability, and capacity of the network, as well as to develop new services and support scientific research.

Challenges and Considerations on private infrastructure

- High initial **investment** in hardware and infrastructure.
- Requires specialized **expertise** in Kubernetes, Kubeflow, and OpenStack.
- Maintenance and management of the **infrastructure**.
- *Potential* **complexity** in integrating different components.
- Careful planning and **resource allocation** are crucial for optimal performance.

What: Kubeflow the AI Orchestration Engine

- **simplifies** the deployment and management of machine learning (ML) workflows on Kubernetes.
- It provides a user-friendly **interface**
- provides a comprehensive **platform** for building, training, and deploying AI models.
- supports various ML **frameworks** like TensorFlow, PyTorch, and scikit-learn.
- offers **tools** for experiment tracking, model versioning, and pipeline management.
- allow researchers to focus on rapid experimentation with **shared** notebooks and data without worrying about the underlying infrastructure.

Kubeflow vs Jupyter hub

- Jupyter hub is a tool for **individual** data scientists to explore and experiment with data

- Kubeflow is a platform that enables **teams** to build, deploy, and manage large-scale ML pipelines.

How: Architecture Overview

Layered architecture ensures scalability, resilience, efficient resource utilization.

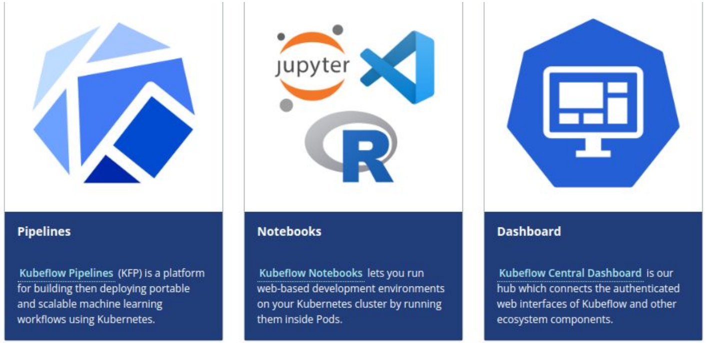
OpenStack provides the underlying IaaS.

Terraform manages OpenStack resources (VMs, networks, etc.).

RKE2 forms the **Kubernetes** cluster.

Ansible automates Kubeflow installation and configuration.

Kubeflow components are deployed on the RKE2 cluster.



Pipelines

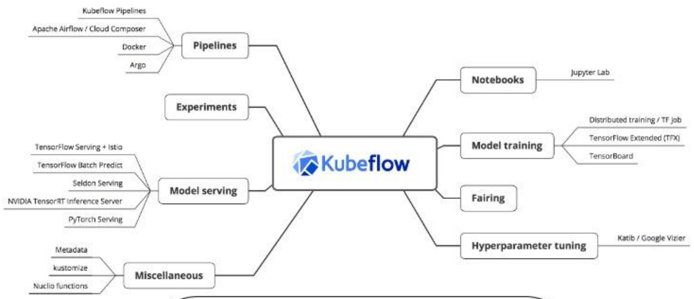
Kubeflow Pipelines (KFP) is a platform for building then deploying portable and scalable machine learning workflows using Kubernetes.

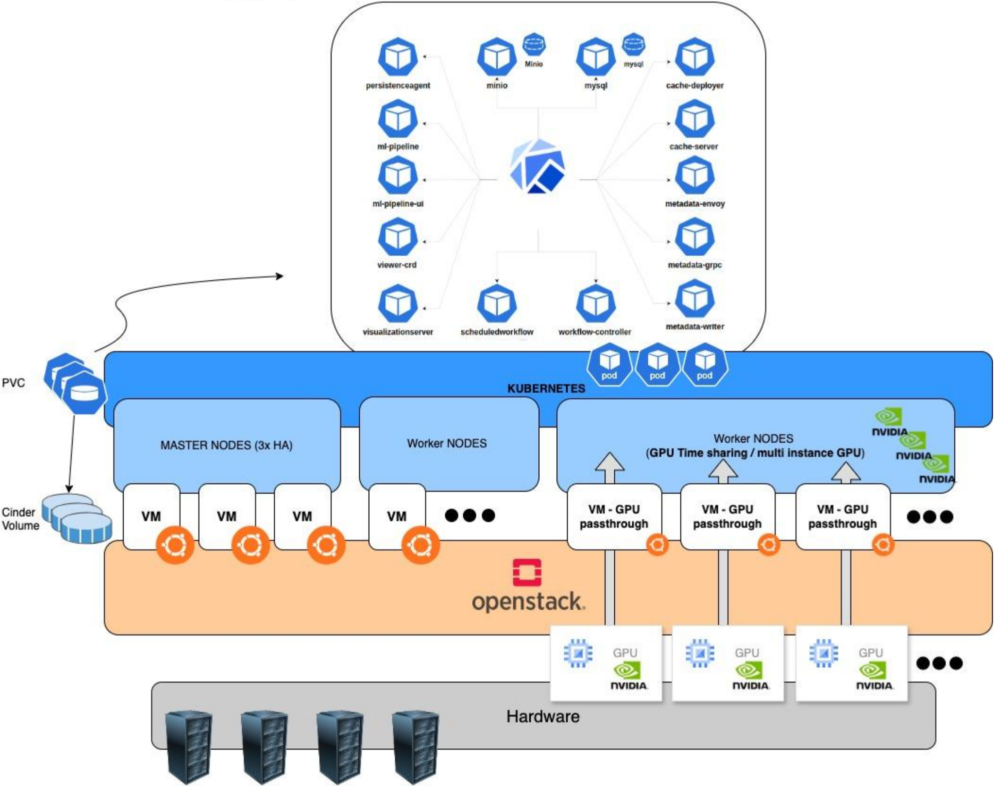
Notebooks

Kubeflow Notebooks lets you run web-based development environments on your Kubernetes cluster by running them inside Pods.

Dashboard

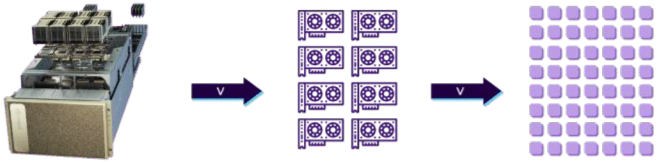
Kubeflow Central Dashboard is our hub which connects the authenticated web interfaces of Kubeflow and other ecosystem components.

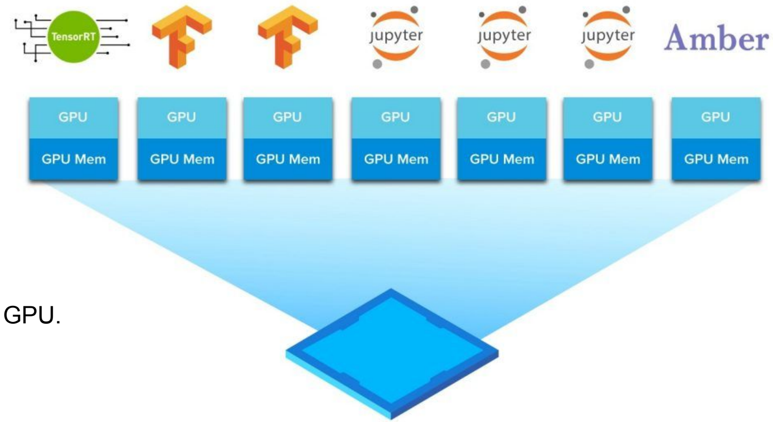




GPU timesharing and/or Multi Instance (?)

- **Multi-Instance GPU (MIG):** enables partitioning a single physical GPU into multiple isolated vGPU instances, providing granular control over GPU resources and allowing multiple users or applications to share a single GPU.





- **GPU Time-Sharing:** multiple applications or workloads can share a single physical/virtual GPU.

  How it Works:

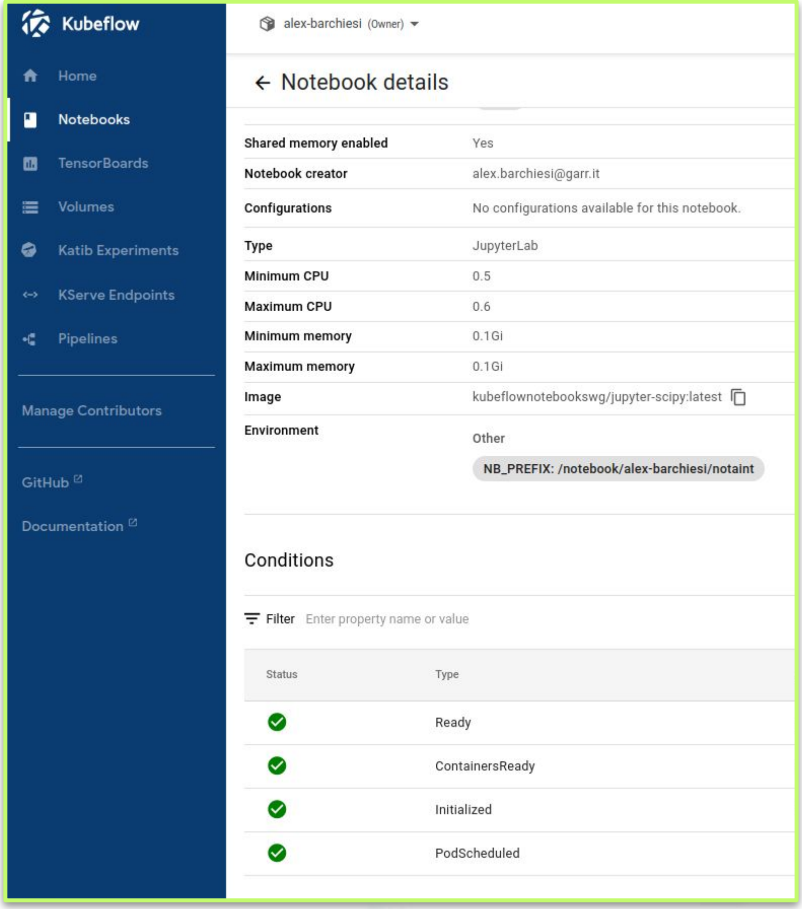
  1. **Time Slicing:** The GPU's processing time is divided into smaller time slices.
  2. **Task Switching:** The GPU switches between different tasks, allocating a portion of its resources to each task.
  3. **Resource Sharing:** Multiple workloads can share the GPU's memory and compute resources.

Kubeflow Notebooks

Interactive development environment for exploratory analysis, prototyping…

Some key features include:

- Native support for JupyterLab, RStudio, and Visual Studio Code (code-server).
- Users can create notebook **containers directly in the cluster**, rather than locally on their workstations.
- Admins can provide standard **notebook images** for their organization with required packages pre-installed.
- **Access control** is managed by Kubeflow’s RBAC, enabling easier notebook sharing across the organization.



← Notebook details

Shared memory enabled	Yes
Notebook creator	alex.barchiesi@garr.it
Configurations	No configurations available for this notebook.
Type	JupyterLab
Minimum CPU	0.5
Maximum CPU	0.6
Minimum memory	0.1Gi
Maximum memory	0.1Gi
Image	kubeflownotebookswg/jupyter-scipy:latest
Environment	Other NB_PREFIX: /notebook/alex-barchiesi/notaint

Conditions

Filter Enter property name or value

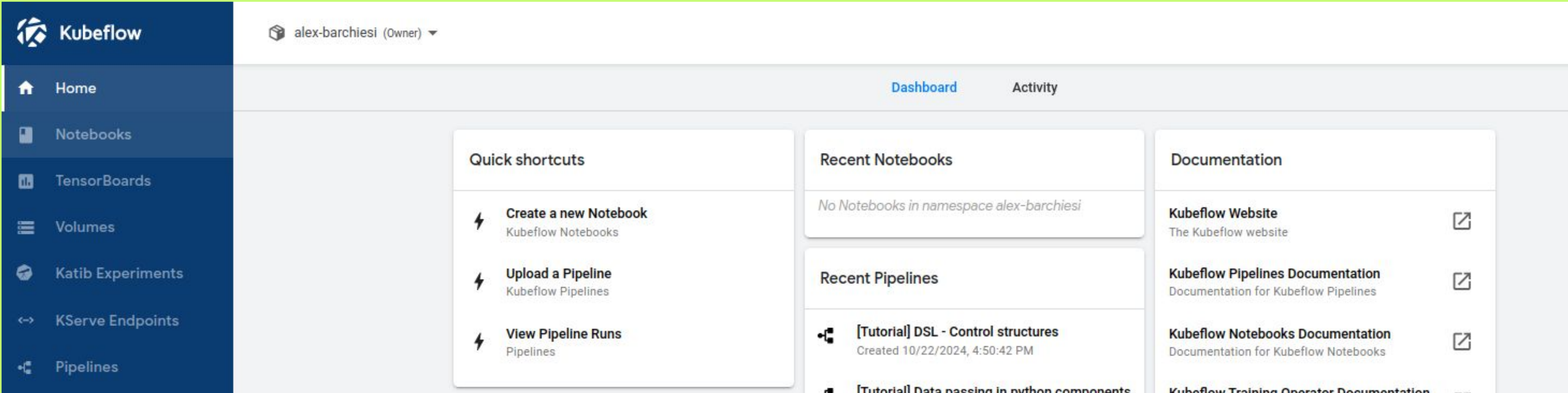
Status	Type
✓	Ready
✓	ContainersReady
✓	Initialized
✓	PodScheduled

Kubeflow Pipelines

Pipelines are for automating and managing entire ML workflows, from data ingestion to model deployment.

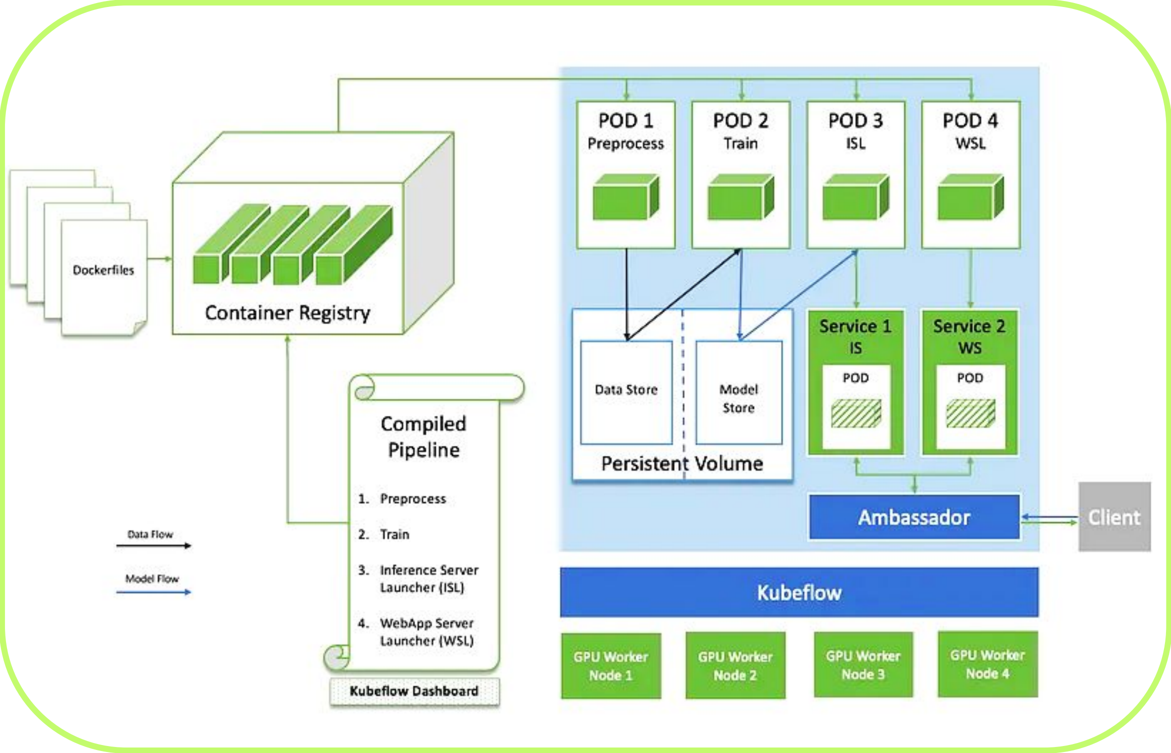
A pipeline is a definition of a more complex workflow that composes one or more components together. At runtime, each component execution corresponds to a single container execution, which may create ML artifacts. Pipelines may also feature control flow.

With KFP you can author components and pipelines, compile pipelines to an intermediate representation YAML, and submit the pipeline to run on a KFP-conformant backend.

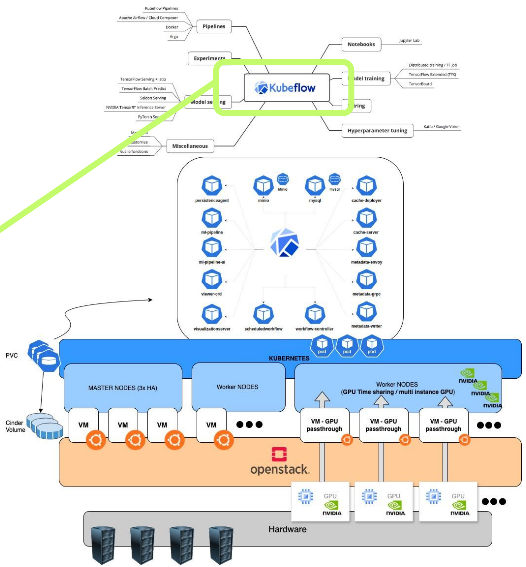


No Notebooks in namespace alex-barchiesi

Details



Dockerfiles
Container Registry
POD 1
Preprocess
POD 2
Train
POD 3
ISL
POD 4
WSL
Data Store
Model Store
Persistent Volume
Service 1
IS
POD
Service 2
WS
POD
Ambassador
Client
Compiled Pipeline
1. Preprocess
2. Train
3. Inference Server Launcher (ISL)
4. WebApp Server Launcher (WSL)
Kubeflow Dashboard
Data Flow
Model Flow
Kubeflow
GPU Worker Node 1
GPU Worker Node 2
GPU Worker Node 3
GPU Worker Node 4



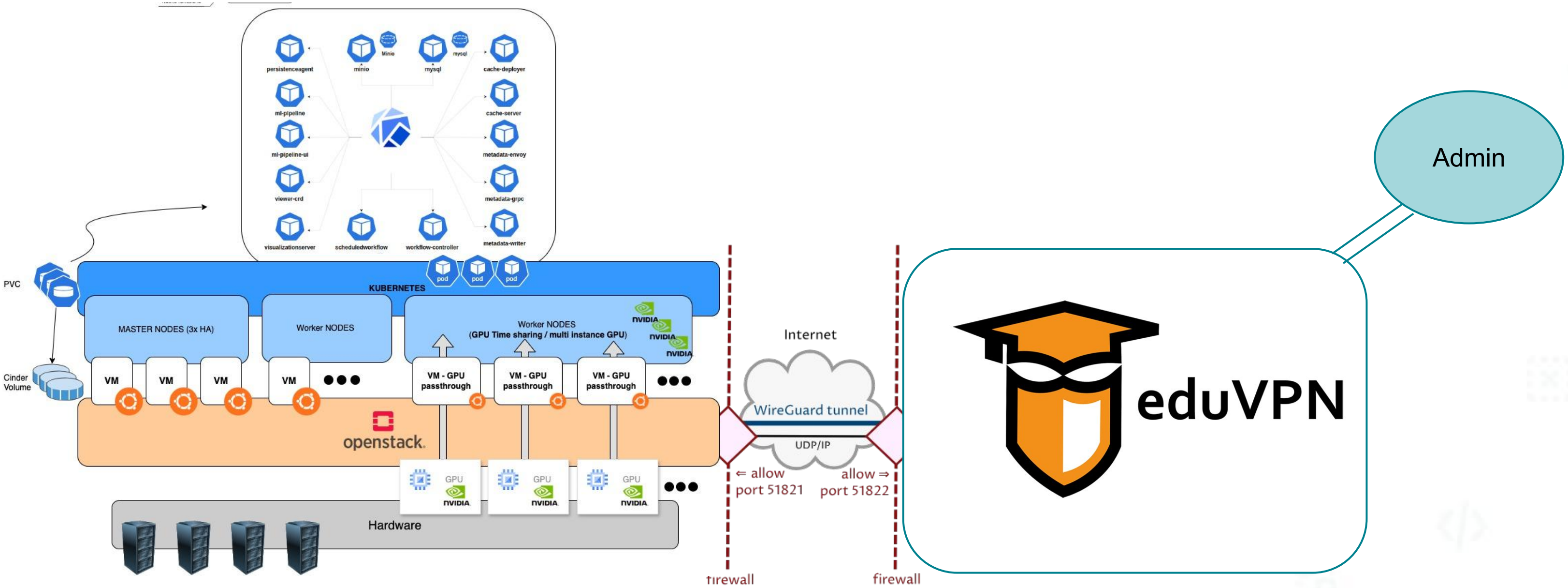
Kubeflow Pipelines
Apache Airflow / Cloud Composer
Docker
Argo
Pipelines
Experiments
TensorFlow Serving + Istio
TensorFlow Batch Predict
Seldon Serving
NVIDIA TensorRT Inference Server
PyTorch Ser
Model se ng
Me
ustomize
Nuclio functions
Miscellaneous
Kubeflow
Notebooks
Jupyter Lab
odel training
Distributed training / TF job
TensorFlow Extended (TFX)
TensorBoard
ring
Hyperparameter tuning
Katib / Google Vizier
persistenceagent
minio
Minio
mysql
mysql
cache-deployer
ml-pipeline
cache-server
ml-pipeline-ui
metadata-envoy
viewer-crd
metadata-grpc
visualizationserver
scheduledworkflow
workflow-controller
metadata-writer
pod
pod
pod
KUBERNETES
PVC
MASTER NODES (3x HA)
Worker NODES
Worker NODES
(GPU Time sharing / multi instance GPU)
NVIDIA
NVIDIA
NVIDIA
Cinder Volume
VM
VM
VM
VM
●●●
VM - GPU passthrough
VM - GPU passthrough
VM - GPU passthrough
●●●
openstack.
GPU
NVIDIA
GPU
NVIDIA
GPU
NVIDIA
●●●
Hardware

Security Considerations

By design: secure communication between components through TLS encryption.

Additional security: tunnel WG - eduVPN to access VM private network

TECH NOTE: OpenStack side you must take care of ALL the security and **disable** Neutron VM security management (port security)



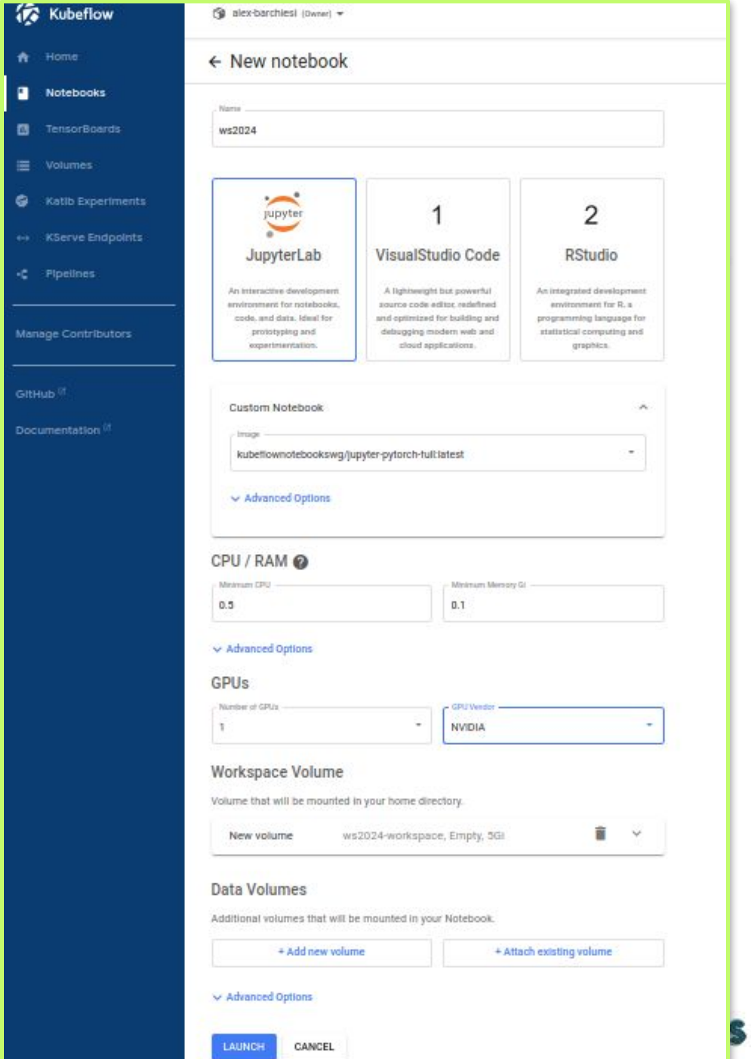
Then: Future development

- Scalability
- Administration
- Governance
    - Policy to maximize availability of resources (reduce inactivity)
    - Scheduling priority policy

Thanks

Special thanks and good luck to Alfredo.Funicello ~~@garr.it~~

Kubeflow Interfaces



← New notebook

Name: ws2024

JupyterLab
An interactive development environment for notebooks, code, and data. Ideal for prototyping and experimentation.

1
VisualStudio Code
A lightweight but powerful source code editor, redefined and optimized for building and debugging modern web and cloud applications.

2
RStudio
An integrated development environment for R, a programming language for statistical computing and graphics.

Custom Notebook

Image: kubeflownotebookswg/jupyter-pytorch-full:latest

Advanced Options

CPU / RAM

Minimum CPU: 0.5

Minimum Memory Gi: 0.1

Advanced Options

GPUs

Number of GPUs: 1

GPU Vendor: NVIDIA

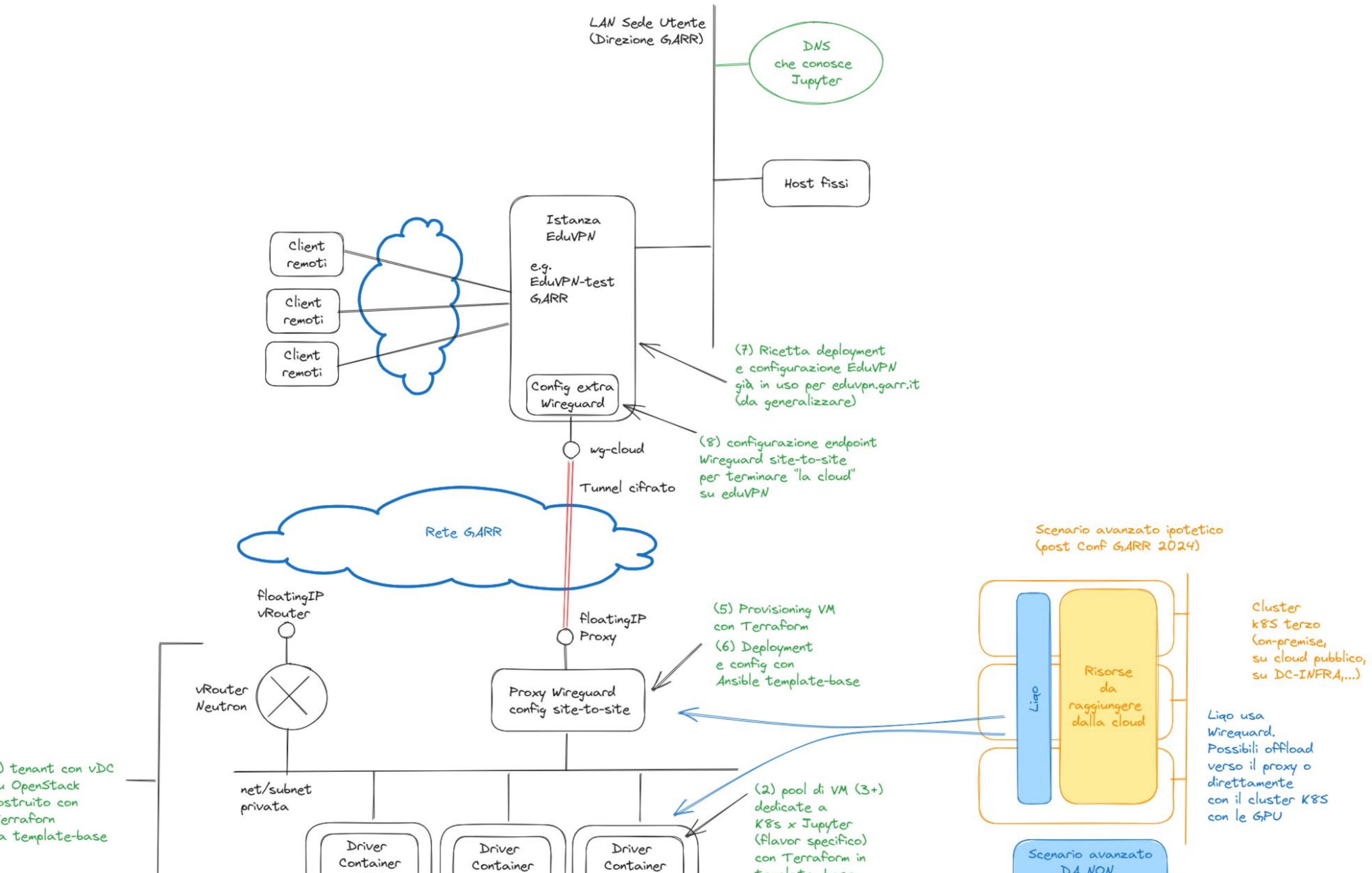
Workspace Volume

Volume that will be mounted in your home directory.

New volume ws2024-workspace, Empty, 5Gi

Data Volumes

Additional volumes that will be mounted in your Notebook.



(7) Ricetta deployment
e configurazione EduVPN
già in uso per eduvpn.garr.it
(da generalizzare)

(8) configurazione endpoint
Wireguard site-to-site
per terminare "la cloud"
su eduVPN

(5) Provisioning VM
con Terraform

(6) Deployment
e config con
Ansible template-base

) tenant con vDC
u OpenStack
ostruito con
erraforn
a template-base

(2) pool di VM (3+)
dedicate a
K8s x Jupyter
(flavor specifico)
con Terraform in

Scenario avanzato ipotetico
(post Conf GARR 2024)

Cluster
k8S terzo
(on-premise,
su cloud pubblico,
su DC-INFRA,...)

Liqo usa
Wirequard.
Possibili offload
verso il proxy o
direttamente
con il cluster K8S
con le GPU