Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
RESTART

FIGO

Federated Infrastructure for GPU Orchestration

INW 2025 - 21st Italian Networking Workshop

Moena (TN), Jan. 20-22

Stefano Salsano[1,2], A. Mayer[1,2], P. Lungaroni[1,2], L. Bracciale[1,2], P. Loreti[1,2], A. Detti[1,2], M. Orazi[1]

(1) University of Rome Tor Vergata, (2) CNIT

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
RESTART

Unleashing GPU Power: Addressing Researchers' Demand

- **Exploding Demand for GPUs**

- **Challenges Faced by Researchers**
  - **High Costs**: GPUs are expensive, limiting access for smaller research groups.
  - **Limited Availability**: Scarcity of cutting-edge GPU models, creating bottlenecks in research workflows.
  - **Underutilization**: GPUs often sit idle due to exclusive (static) allocations.

Solution : dynamic sharing and a federated infrastructure

1) Allocation of GPUs to VMs and Container needs to be **dynamic**

2) Research groups that owns servers with GPUs can make them available over a **federated** infrastructure.

A booking web GUI is used to reserve the resources.

The owners of the GPUs have priority (and even pre-emption) on their resources… when the GPUs are not used, they will be shared.

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
RESTART

FIGO - Federated Infrastructure for GPU Orchestration

- Enable dynamic sharing of GPUs across federated testbeds.
- Maximize resource utilization while reducing idle times.
- Make high-performance GPUs more accessible to a broader research community.

Technical issues with current GPU allocation mechanisms

Virtual Infrastructure Managers like OpenStack **statically** assign GPUs to VM.

Once the GPU is allocated to the VM, it cannot be given to other VMs even if the VM is not running.

NOT EFFICIENT!

FIGO - Technical details

The **incus** orchestrator (lxd) is used to manage containers and VMs that use the GPUs

FIGO is basically a wrapper over incus, with commands to:

- handle instances (VMs and containers)
- manage GPU profiles
- administer users
- manage remote servers offering CPU and GPU resources

FIGO - Command Line Interface (CLI) Commands

Command	Description
`figo instance`	Manage instances (VMs and containers), including creation, listing, starting, stopping, IP configuration, key management, and direct bash execution.
`figo gpu`	Handle GPU management, including monitoring status, adding/removing GPU profiles, listing profiles, and retrieving PCI addresses.
`figo profile`	Manage profiles by showing, dumping, listing, copying, deleting, and initializing them.
`figo user`	Manage users, including adding, editing, listing, granting access, and deleting users with detailed control over project association and VPN setup.
`figo remote`	Manage remote servers, including listing, enrolling, and deleting remotes for federated infrastructure.
`figo project`	Manage projects by creating, listing, and deleting isolated environments for users or research groups.
`figo operation`	Monitor and manage operations across the federated testbed, including status and progress tracking.
`figo vpn`	Configure VPN routes for secure communication, supporting various router types and dynamic configuration options.

FIGO - Example commands

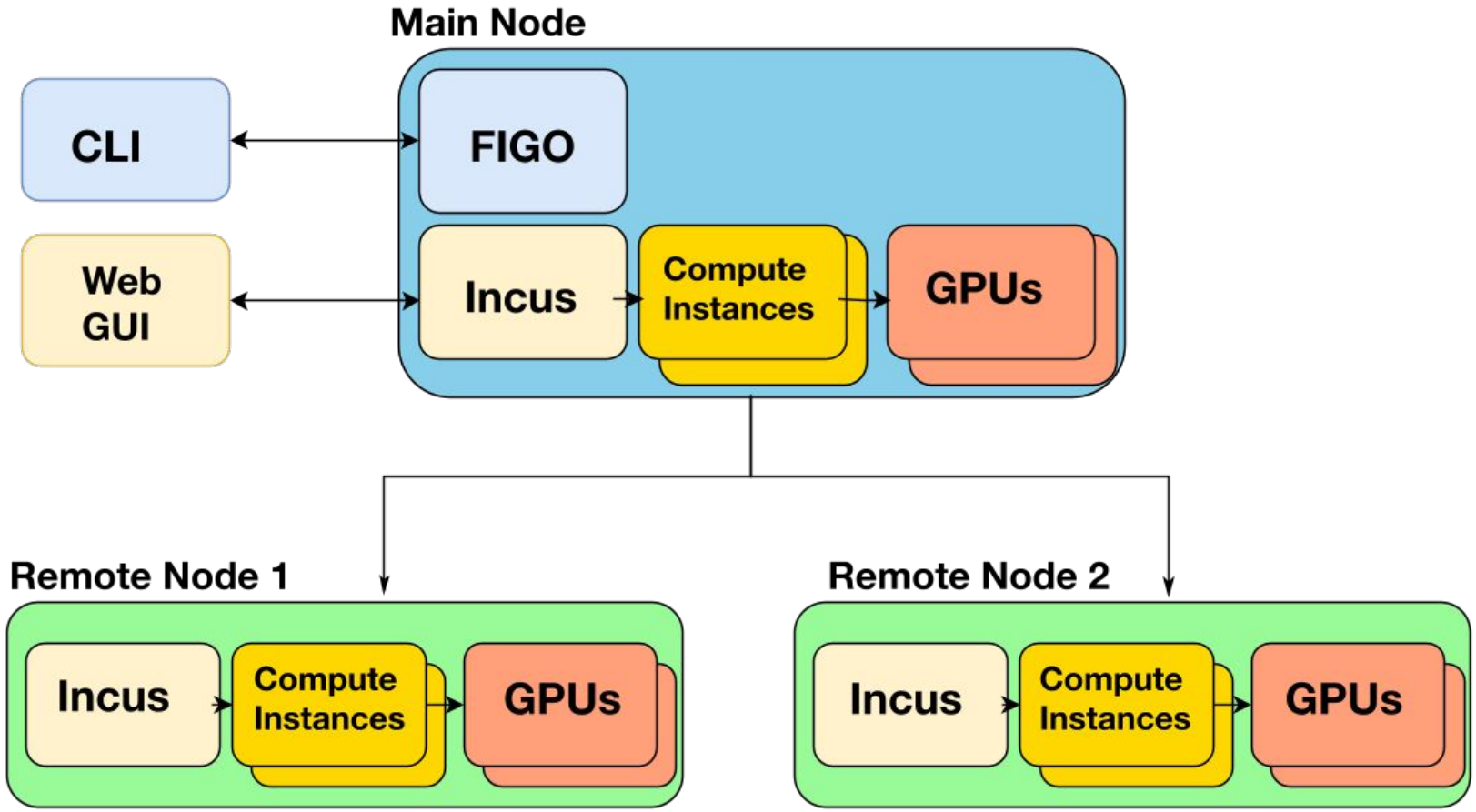
```
# Create a new user
figo user add john_doe -e john@example.com -n "John Doe" -o
"Example Corp" -w -k -s

# Enroll a remote node
figo remote enroll my_remote 192.168.1.100 8443 ubuntu
~/.config/incus/client.crt /var/lib/incus/server.crt
--loc_name main
```

FIGO - Example commands

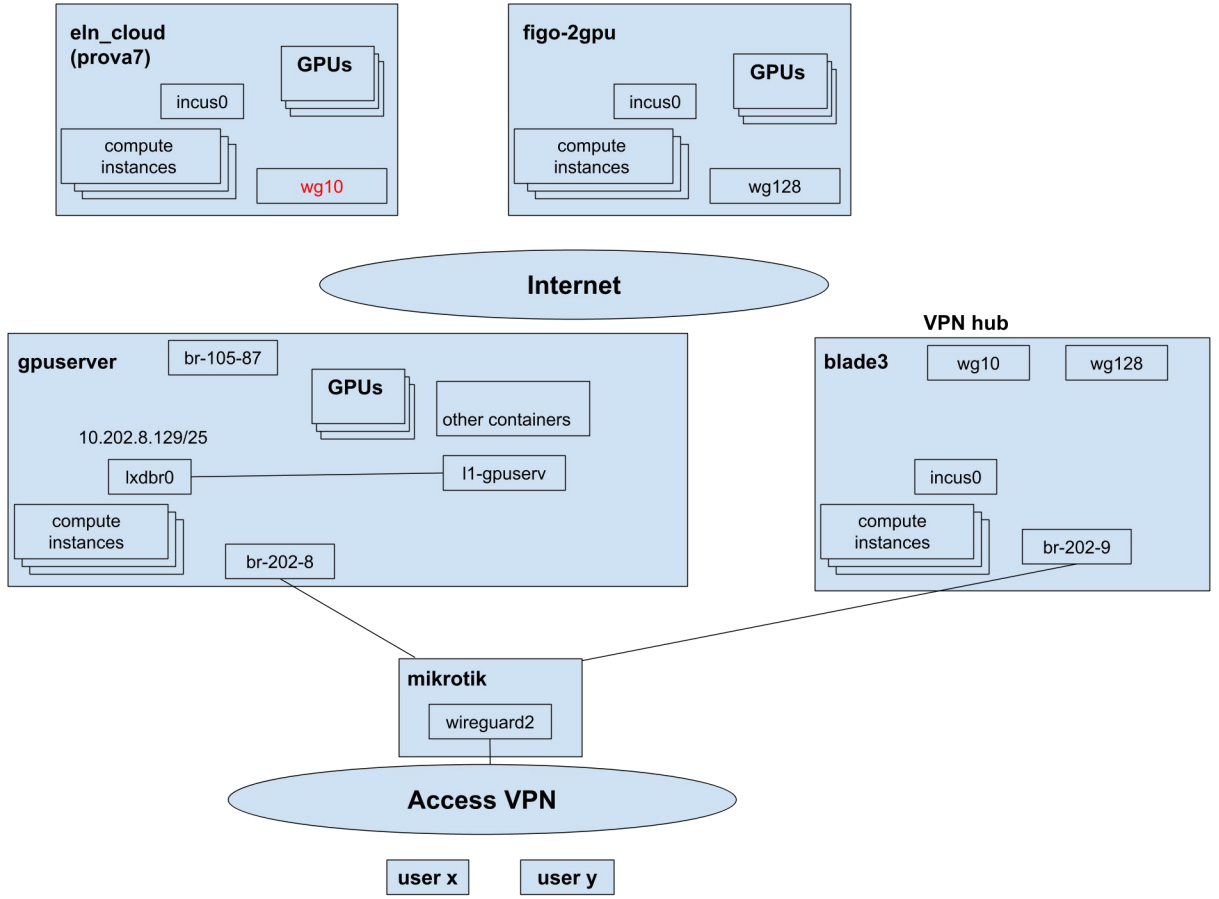
```
# Attach a GPU to an instance on a remote in a project
figo gpu add my_remote:my_project.instance_name
```

```
# Add a route using a host address with expl. user and port
figo vpn add route 10.10.128.0/24 via 10.10.10.2 type
mikrotik host 160.80.10.2 --user myuser --port 22
```

FIGO - Architecture

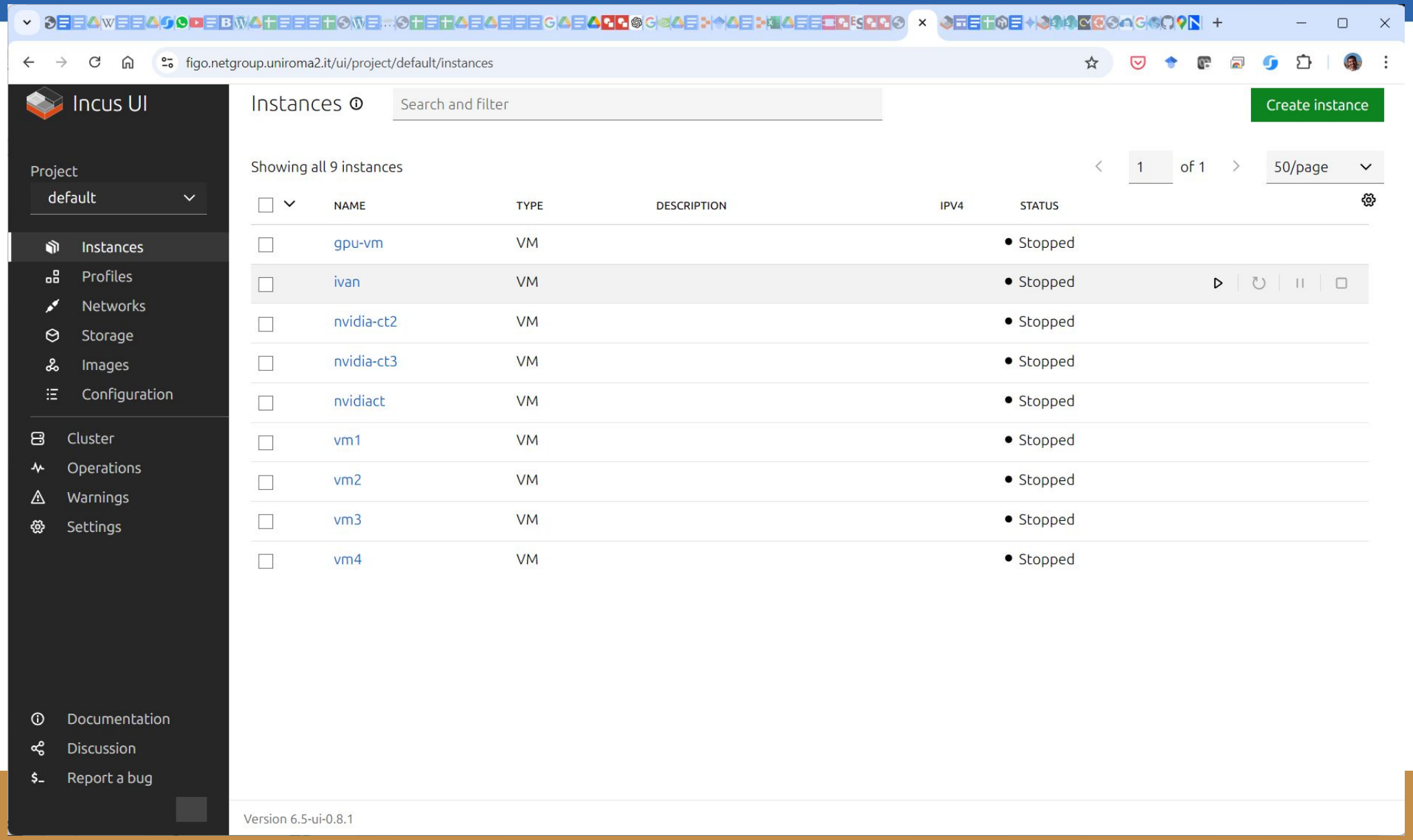


FIGO - Physical infrastructure



Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
RESTART

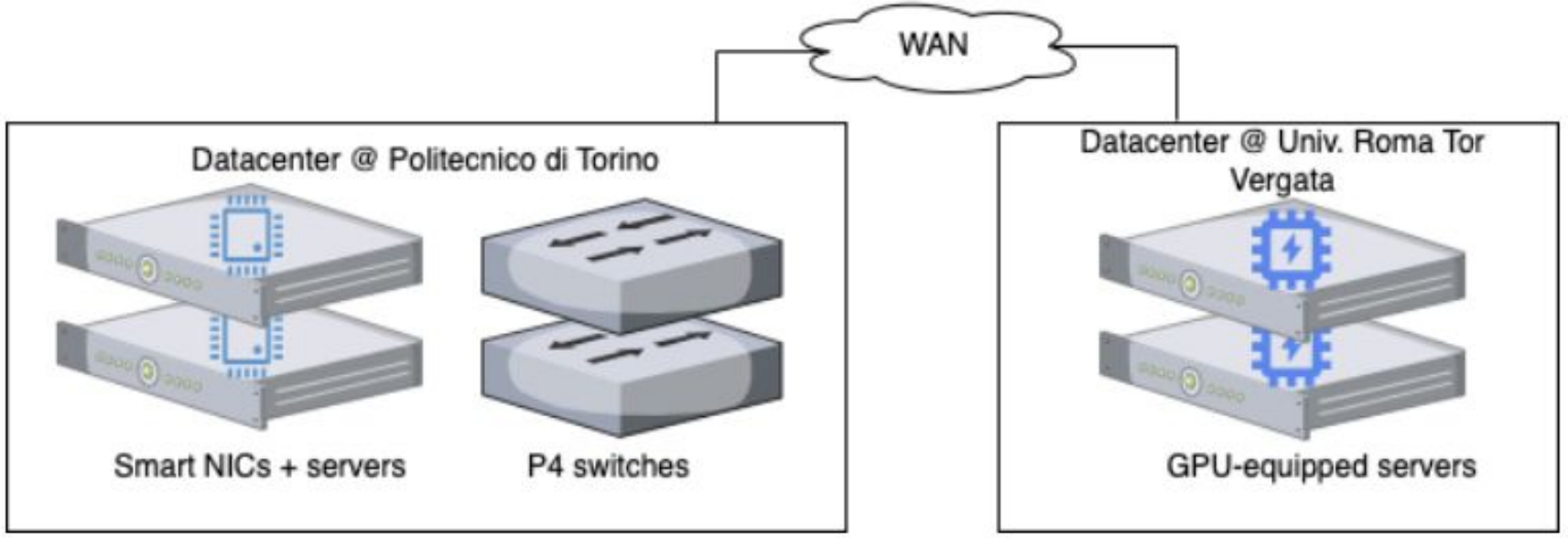
Incus UI



FIGO - Open source repository

https://github.com/StefanoSalsano/figo

Beyond FIGO - Sharing Resources in the RESTART Distributed Testbed



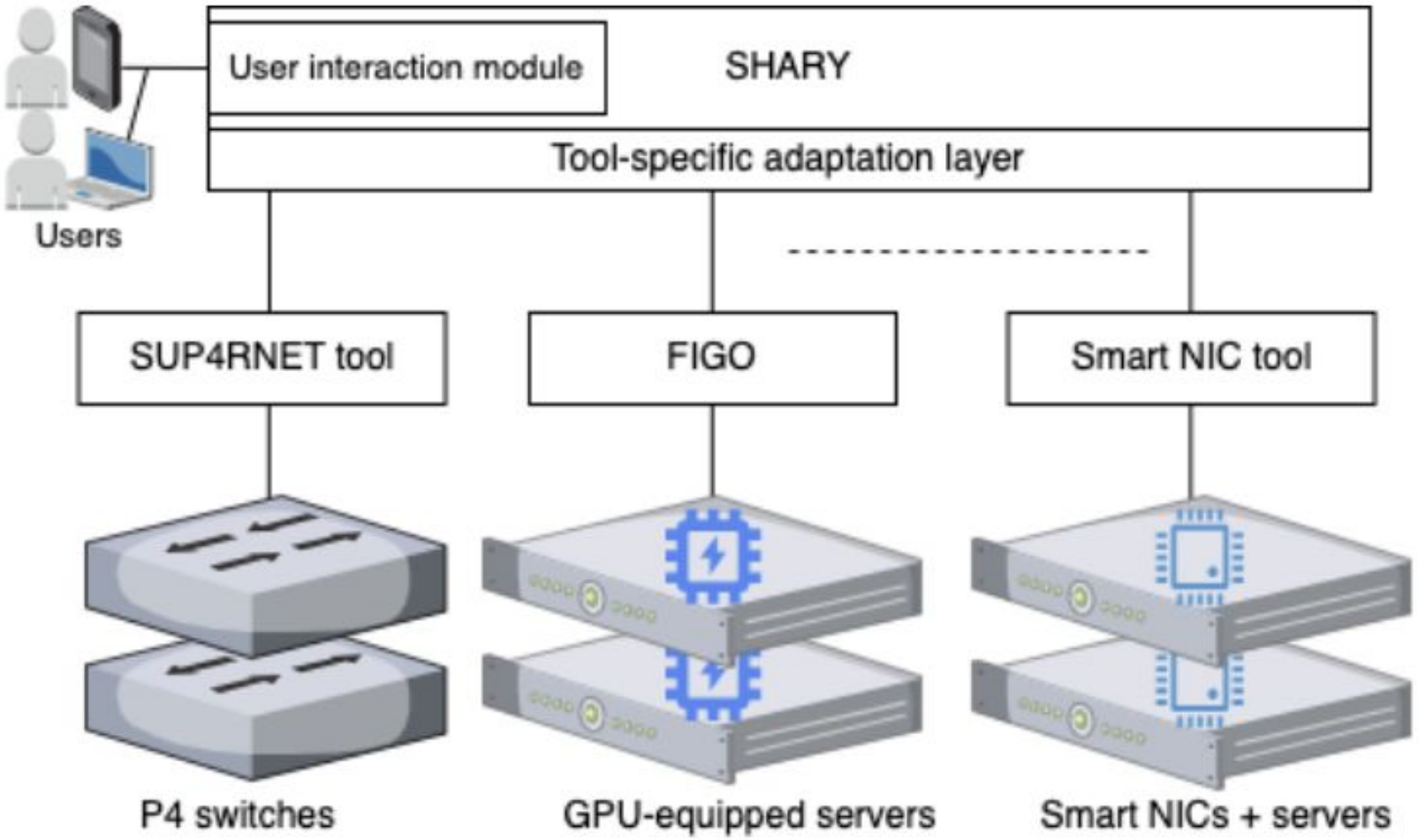
The RESTART distributed testbed offers more than GPUs:

- Smart NICs - Programmable (P4) Switches

SHARY - SHaring Any Resource made easY

A common web front end to manage reservations

Specific modules to adapt to the resources



Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
RESTART

Sharing GPUs and Programmable Switches in a Federated Testbed with SHARY

Accepted to NOMS 2025

S. Salsano[1,2], A. Mayer[1,2], P. Lungaroni[1,2], P. Loreti[1,2], L. Bracciale[1,2], A. Detti[1,2], M. Orazi[1]

P. Giaccone[3], F. Risso[3], A. Cornacchia[4], C. F. Chiasserini[3]

(1) University of Rome Tor Vergata, (2) CNIT,
(3) Politecnico di Torino, (4) KAUST

FIGO : main benefits

Centralized management and configuration of hw and sw

Pre-configured containers and VM images to simplify the research

Increase the utilization of GPU resources (sharing within a research group and between research groups), get access to more resources

Provide GPU access to researchers who do not have GPUs

Questions for you

Do you already have GPUs in your lab?

Are you waiting for GPUs?

Interested in joining the initiative?

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
RESTART

FIGO

Federated Infrastructure for GPU Orchestration

Contact: **stefano.salsano@uniroma2.it**


RESTART