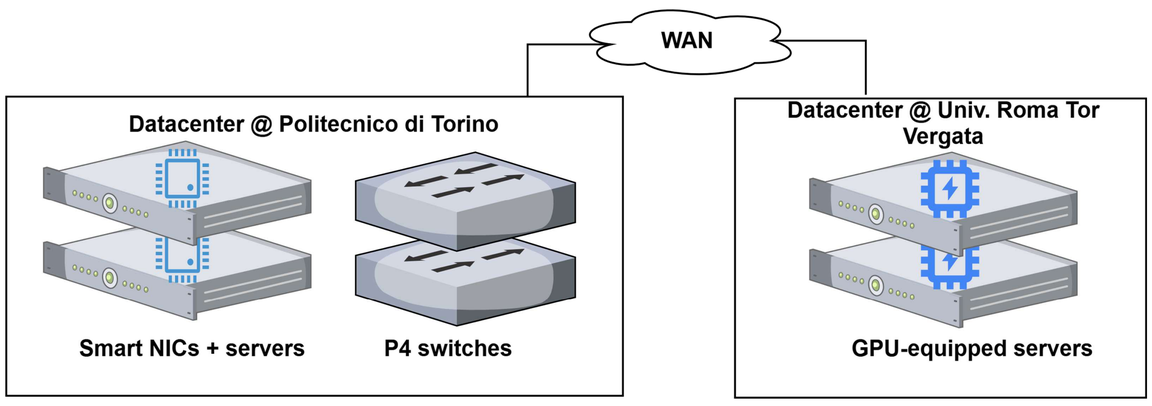
Sharing GPUs and Programmable Switches in a Federated Testbed with SHARY

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

RESTART Distributed Testbed



- **Servers with GPUs**
- **Smart NICs**
- **Programmable (P4) Switches**

Sharing resources in Federated Testbed

Underutilization of Specialized Hardware:

- GPUs and other high-performance resources are often idle despite high demand.
- Fixed allocations lead to inefficiencies in federated environments.

Static Resource Allocation Limits Flexibility:

- Traditional reservation models rely on **static partitions** or **fixed time slots**.
- These approaches fail to adapt to **variable workload demands**.

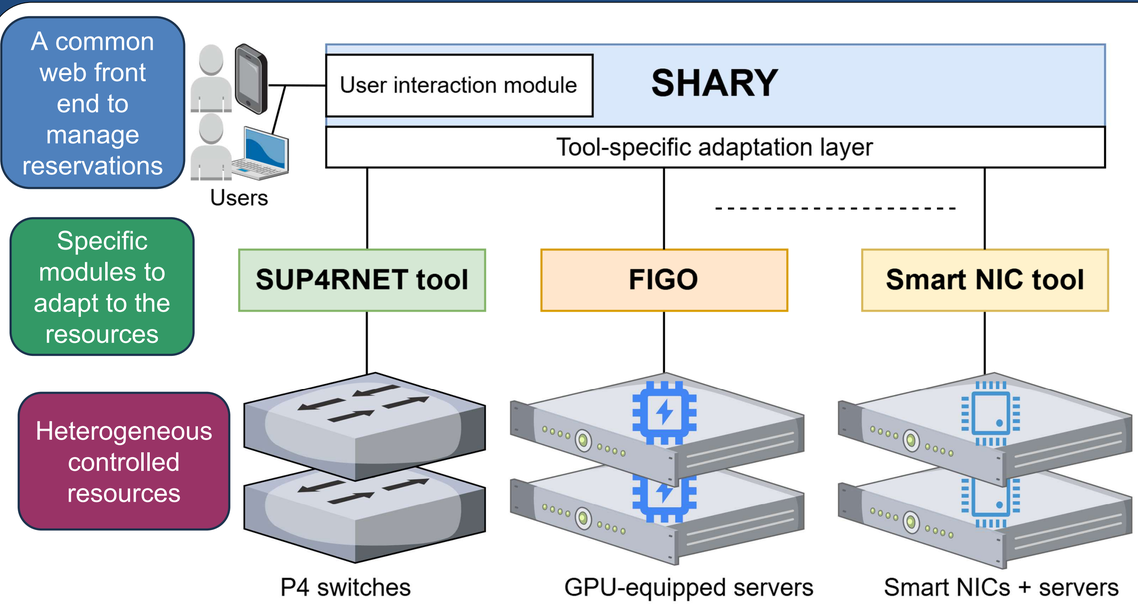
Heterogeneity of Resources and Interfaces:

- Federated testbeds integrate diverse hardware (GPUs, CPUs, SmartNICs, FPGAs…).
- Different managem. protocols and software environments complicate resource sharing.

Lack of Dynamic Coordination Mechanisms:

- Scheduling across **multiple sites** with different infrastructures is challenging.
- Researchers need **on-demand access** rather than rigid booking models.

SHARY - SHaring Any Resource made easY



Addressed Gaps

1 Rigid Reservation Systems Lead to Wasted Resources

- Traditional calendar-based systems rely on fixed time slots, leading to underutilization.
- Researchers book longer-than-needed slots to avoid interruptions.
- Need for adaptable and dynamic reservation mechanisms.

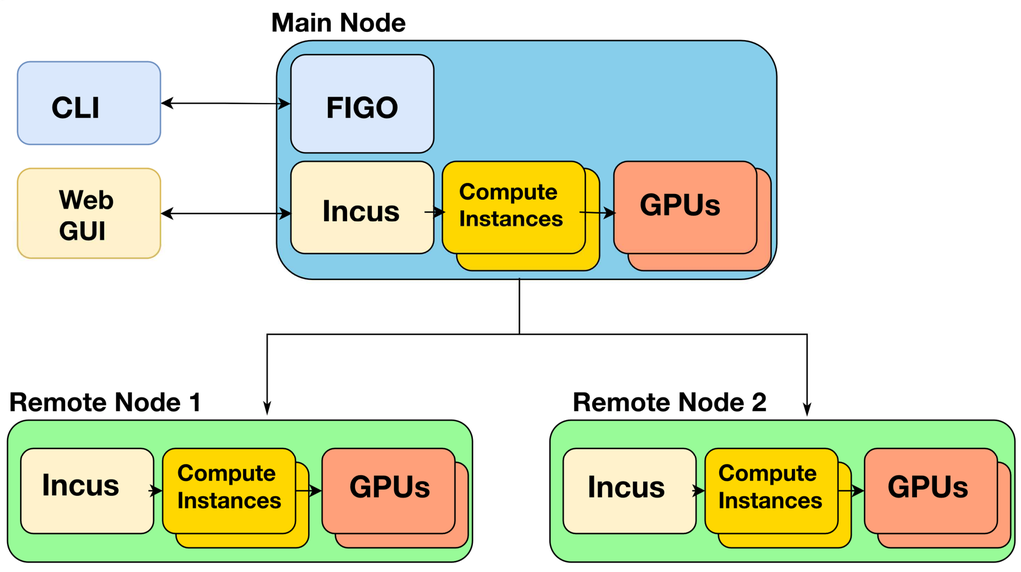
2 Inefficient GPU Sharing Mechanisms

- Static partitioning methods waste capacity when full GPU power is not required.
- Real-time reallocation is needed to maximize utilization.
- FIGO introduces on-the-fly GPU allocation based on user demand.

3 Lack of Multi-Tenant Support in Networking Hardware

- Programmable networking devices lack native multi-tenancy.
- Currently, switching between experiments requires reconfiguration → disruptive.
- Need for parallel access and isolation mechanisms to allow multiple users.

FIGO - Federated Infrastructure for GPU Orchestration

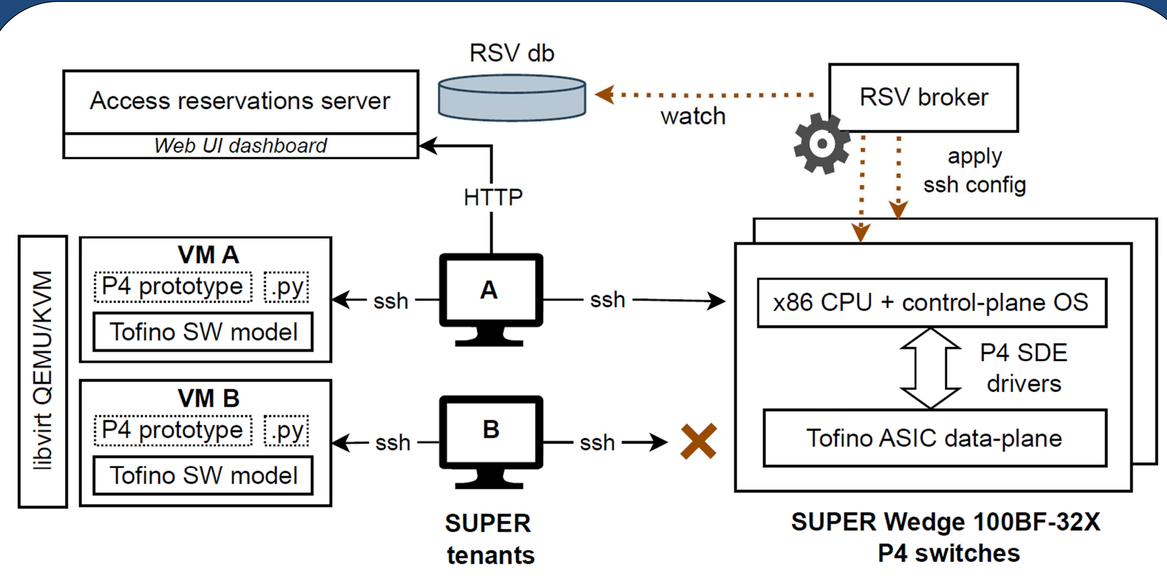


FIGO architecture: each Remote Node runs an instance of incus, coordinated by main node

Flexible GPU Orchestration Framework

- Dynamically allocates GPUs based on real-time demand, reducing idle time.
- Integrates with SHARY for flexible, automated resource booking.
- Optimizes GPU usage across federated testbeds, lowering costs and accelerating AI research.

Federating P4 Switches with SUP4RNET



Architecture of the SUP4RNET P4 cluster and enabled P4 development workflow

Multi-Tenant Support for Programmable Switches

- Enables secure, isolated sharing of P4-based switches among multiple users.
- Reduces downtime by managing dynamic access without full reconfigurations.
- Supports both parallel and sequential access, enhancing network experimentation.

This work was partly supported by the RESTART project ("RESearch and innovation on future Telecommunications systems and networks, to make Italy more smART"), funded by the European Union – NextGenerationEU under the National Recovery and Resilience Plan (NRRP) – Mission 4, Component 2, Investment 1.3 (PE00000001 - programma “RESTART”).