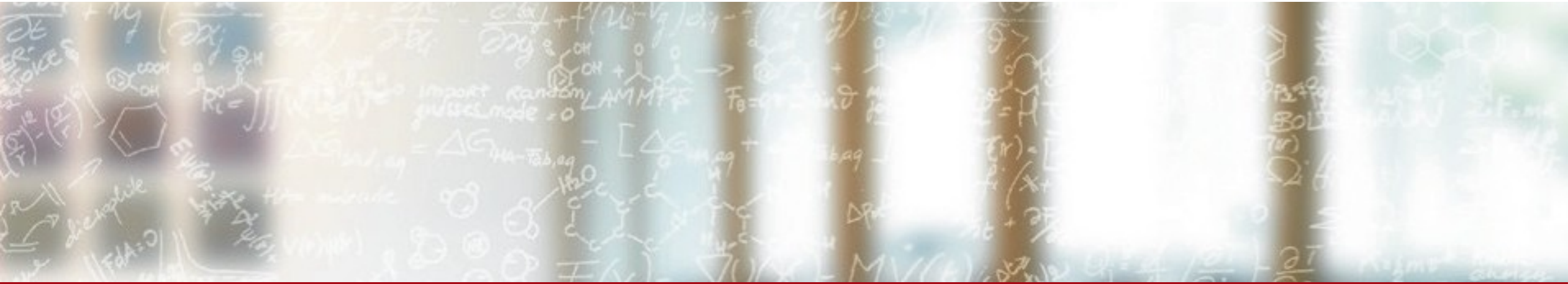




CSCS

Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETHzürich



The evolution of Slurm at CSCS: from monolithic service to multi-tenant vService

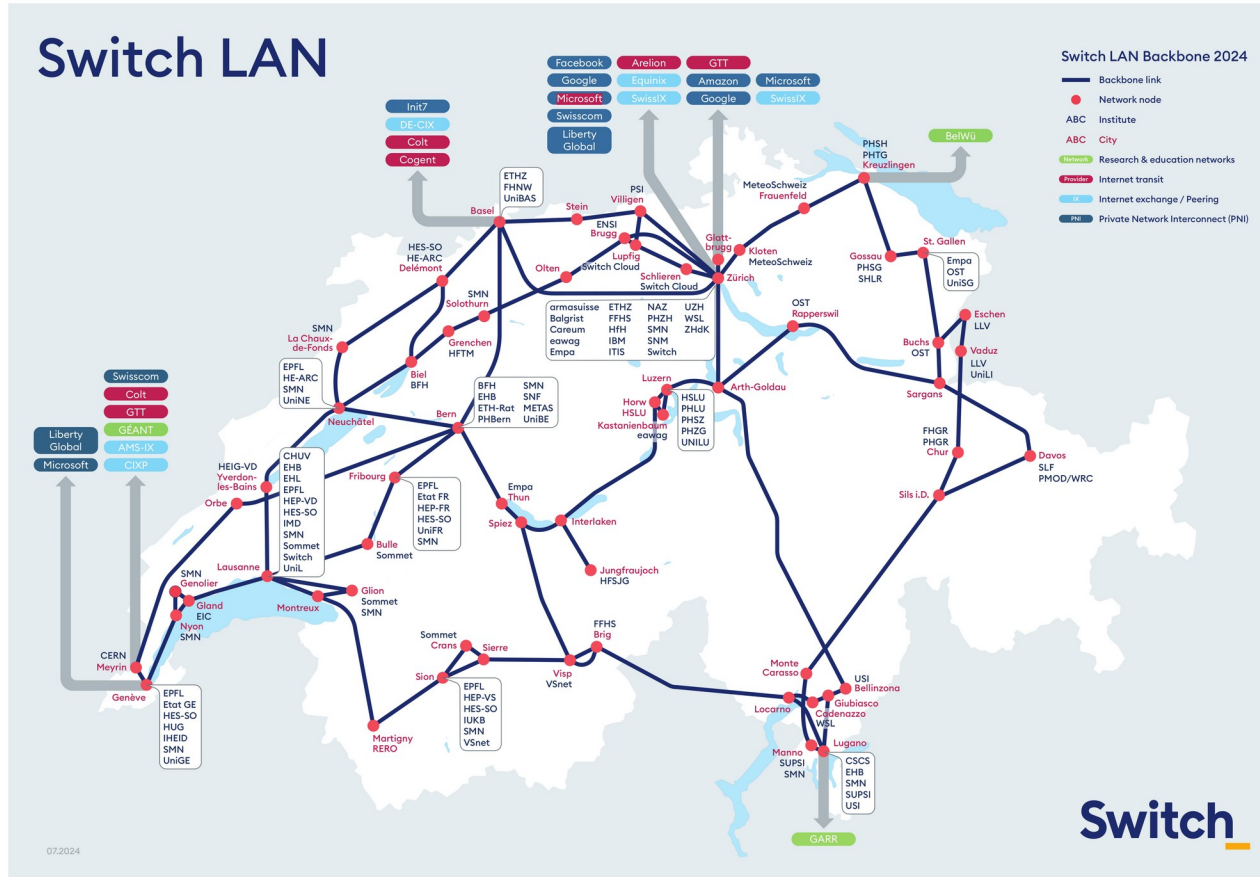
SLUG '24 Oslo

Gennaro Oliva, CSCS

2024-09-12

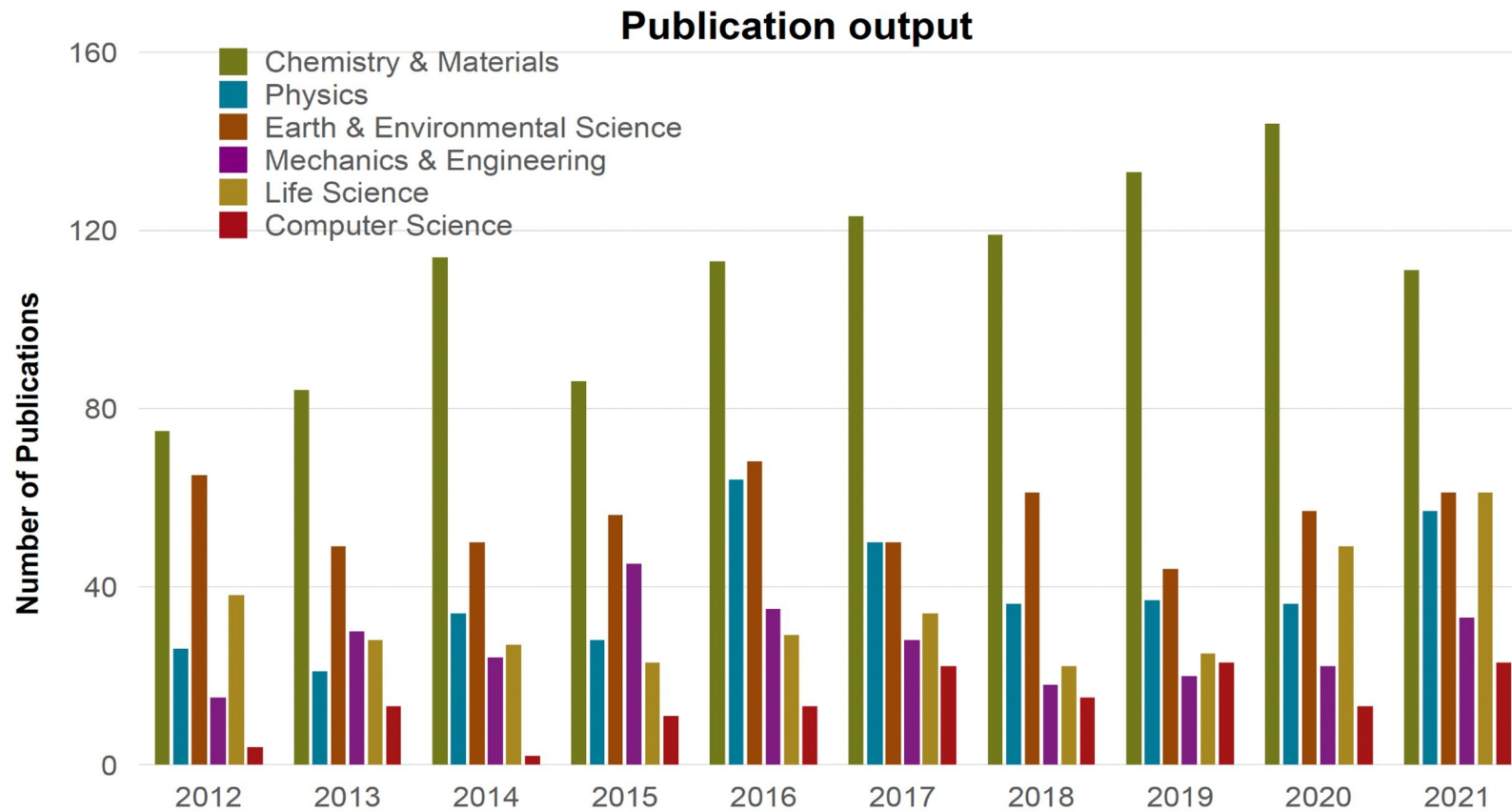
About CSCS

- The Swiss National Supercomputing Centre is the national high-performance computing centre of Switzerland
- The acronym CSCS comes from the Italian: Centro Svizzero di Calcolo Scientifico
- The Centre is located in Lugano, in the southern part of Switzerland, within the Italian-speaking region
- CSCS is well connected across Switzerland (and beyond)



CSCS Mission

- We develop and operate a high-performance computing and data research infrastructure that supports world-class science in Switzerland



CSCS Partnerships

- MeteoCH
- PSI
- CHIPP
- CTA / SKA
- HBP
- UZH
- EMPA
- Euler
- BBP
- ...

About ETH Zurich

- CSCS is a unit of ETH Zurich (Swiss Federal Institute of Technology Zurich) a leading international university renowned for its cutting-edge research, high-quality education, and innovation in science, technology, engineering, and mathematics



About Alps

- Alps is the flagship HPC system built on the HPE Cray EX architecture providing powerful resources for a range of scientific applications, from simulations to data analysis, while being energy-efficient and scalable to meet future needs



Alps locations

- Alps is a geographically distributed computing and storage infrastructure:
 - CSCS in Lugano
 - EPFL in Lausanne
 - Paul Scherrer Institute (PSI) in Villingen for data archiving
 - ECMWF in Bologna for access to meteorological data



Alps Specs

Model	HPE Cray EX
Interconnect	HPC Cray Slingshot-11 with 200 Gbps injection bandwidth per module / GPU
Scratch disk	100 + 10 PB on hard disk 5 + 1 PB on Solid State Disk (SSD)
Data archive and backup	2 x 130 PB tape libraries



CRAY®

Alps nodes overview

# of nodes	# of sockets per node	Total # of sockets	Processor(s)	Specifications
2,688	4	10,752	NVIDIA Grace-Hopper	72 ARM cores, 128 GB LPDDR 5X RAM, H100 GPU with 96 GB HBM3 memory
1,024	2	2,048	AMD EPYC 7742 CPU (Rome)	2x64 cores, 256/512 GB DDR RAM
144	1 CPU + 4 GPU	720	AMD EPYC host CPU + NVIDIA A100	64 cores, 128 GB DDR RAM) and 4 NVIDIA A100-96/80 (96/80 GB HBM2E
128	4	512	AMD Mi300A CPU+GPU	
24	1	24	AMD EPYC host CPU + AMD Mi250X GPU	64 cores, 128 GB RAM) and 4 AMD Mi250X GPU

Alps in the Top 500

List	Rank	System	Total Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
06/2024	6	HPE Cray EX254n, NVIDIA Grace 72C 3.1GHz, NVIDIA GH200 Superchip, Slingshot-11	1,305,600	270.00	353.75	5,194.00



Cray CSM

- Alps runs Cray System Management (CSM) software
- In CSM, cluster services runs as microservices on a dedicated kubernetes infrastructure:
 - Boot image creation
 - Boot services
 - Ansible configuration
 - Node health management
 - Monitoring
 - Node Power management
 - Hardware inventory management
 - Network Fabric management
 - Workload manager
 -
- These services generally see all the computing nodes to be part of a single cluster

Cray Slurm microservice

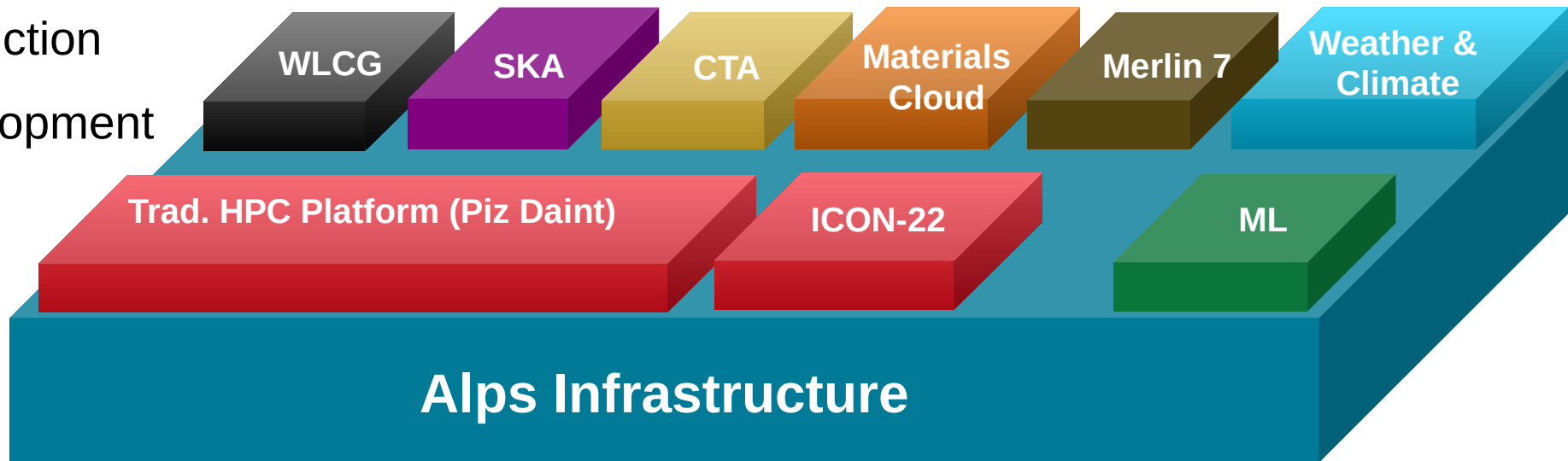
- Slurm has a dedicated kubernetes namespaces where Slurm control damon together with the database daemon and databases run as pods on the management plane

```
goliva@ncn-m001:~> kubectl -n rigi get pods
```

NAME	READY	STATUS	RESTARTS	AGE
rigi-pxc-operator-8578f8bcf6-7rpqp	1/1	Running	0	14d
slurmctld-5f5ddf6bd5-lp9zn	3/3	Running	0	14d
slurmctld-backup-679c6d5c9d-vcptn	3/3	Running	0	14d
slurmdb-haproxy-0	2/2	Running	0	14d
slurmdb-haproxy-1	2/2	Running	0	14d
slurmdb-haproxy-2	2/2	Running	0	14d
slurmdb-pxc-0	1/1	Running	0	11d
slurmdb-pxc-1	1/1	Running	0	11d
slurmdb-pxc-2	1/1	Running	0	11d
slurmdbd-7485b9f996-xcjc4	3/3	Running	0	14d
slurmdbd-backup-f44c75758-5gz7b	3/3	Running	0	14d
xb-cron-slurmdb-backup-202482421100-3r9er-td8l5	0/1	Completed	0	2d11h
xb-cron-slurmdb-backup-202482521100-3r9er-w7fmb	0/1	Completed	0	35h
xb-cron-slurmdb-backup-202482621100-3r9er-mjpkd	0/1	Completed	0	11h

Alps platforms

- CSCS currently hosts different research platform, sharing the same Hardware infrastructure
- Platform are subsets of Alps on top of infrastructure that enables the deployment of one or multiple vClusters
- There are multiple types of Platforms:
 - HPC
 - Grid
 - Production
 - Development
 - ...



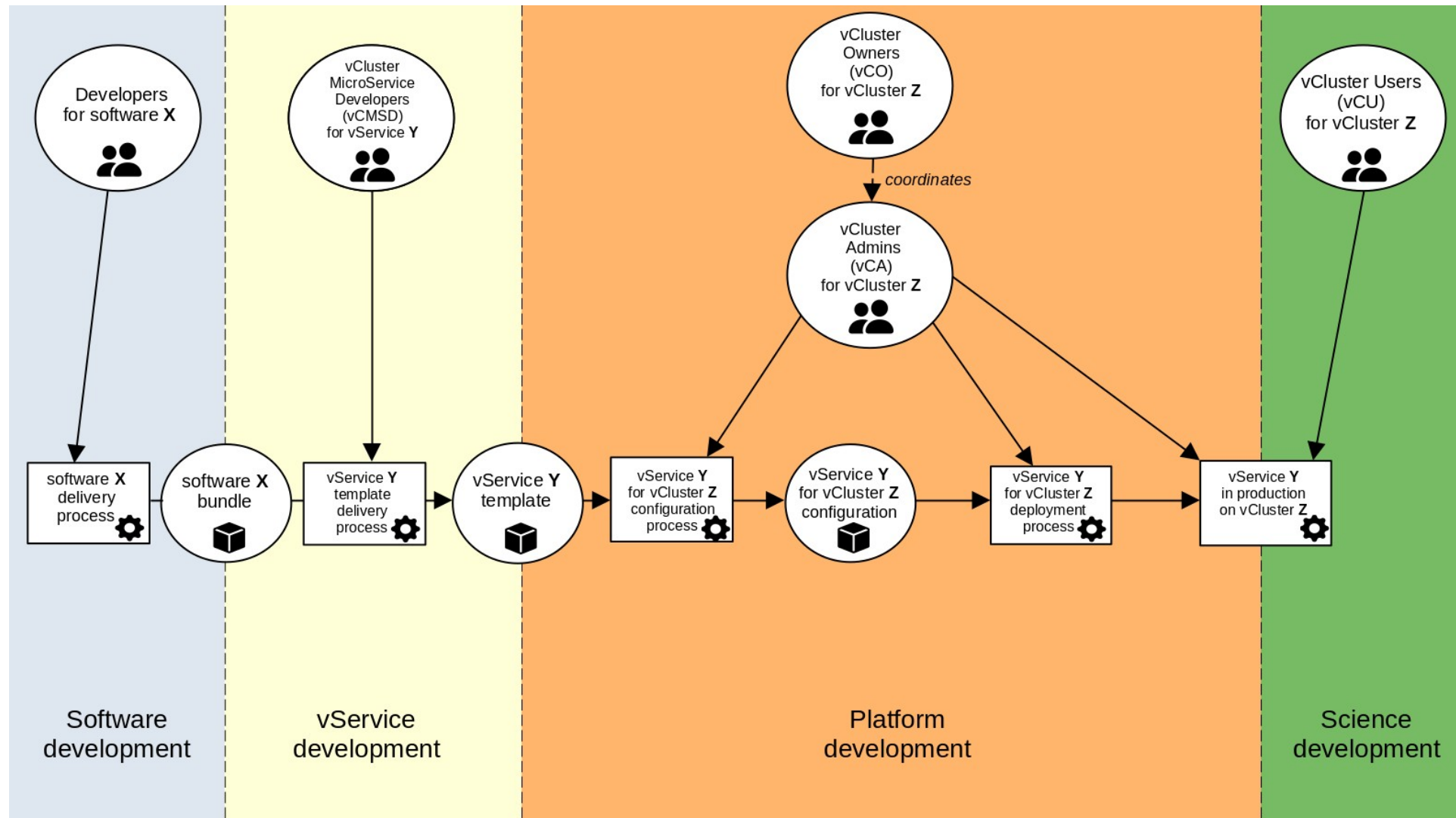
vCluster

- vClusters (versatile software-defined clusters) is based on Infrastructure-as-code (IaC) technology
- vClusters approach mixes HPC and cloud technologies resulting in a software-defined, multi-tenant cluster on a supercomputing ecosystem, that, together with software-defined storage, enable DevOps for complex, data-driven workflows
- vClusters software environment is composed of vServices

vService

- vService stands for "**versatile cluster microservice**" a small, independent, and loosely coupled service hosted on a vCluster
- Vservices follow DevOps practice of continuous integration and delivery for fast and safe deployment of features and updates
- Each vService comes with its own runtime software stack
- vServices features are:
 - Focused and small in scope designed to perform an specific task or closely related functionality
 - Independently deployable: changes to one vService should not propagate to others, reducing the integration effort
 - Loosely coupled: Interactions between vServices are clearly defined and limited

Vservice, vClusters and roles



Slurm vService

- Slurm vService implementation is built upon an automated, version-controlled deployment and management process, utilizing:
 - GitLab for CI/CD
 - Terraform for service provisioning
 - Kubernetes for orchestrating the Slurm controller and database daemons
 - Nomad for native slurmd agent deployment on the HPE Cray EX compute nodes

Slurm vService (Kubernetes)

- Kubernetes orchestrates containerized instances of Slurm daemons such as the Slurm controller (slurmctld), the Slurm database daemon (slurmdbd), and the Slurm rest daemon (slurm-restd), each running within dedicated pods in a Kubernetes tenant namespace
- This setup enhances management by:
 - Automating the deployment of Slurm components, ensuring high availability and fault tolerance
 - Utilizing Kubernetes secrets management to handle sensitive configurations securely, ensuring that components such as Munge (for authentication) are correctly configured across deployment

Slurm vService (Nomad)

- Nomad is employed to manage the deployment of native services on Compute Nodes
- Nomad orchestrates the deployment of the Slurm daemon (slurmd) on compute nodes, extensively using its capability to manage non-containerized tasks
- Key benefits include:
 - Nomad can manage native tasks using the “rawexec” task driver. This allows for direct interaction with compute node hardware, ensuring performance and no overheads
 - Nomad agents running on cluster’s compute nodes, allow a flexible mechanism to manage resources via software and fast reconfiguration

Slurm vService (Terraform)

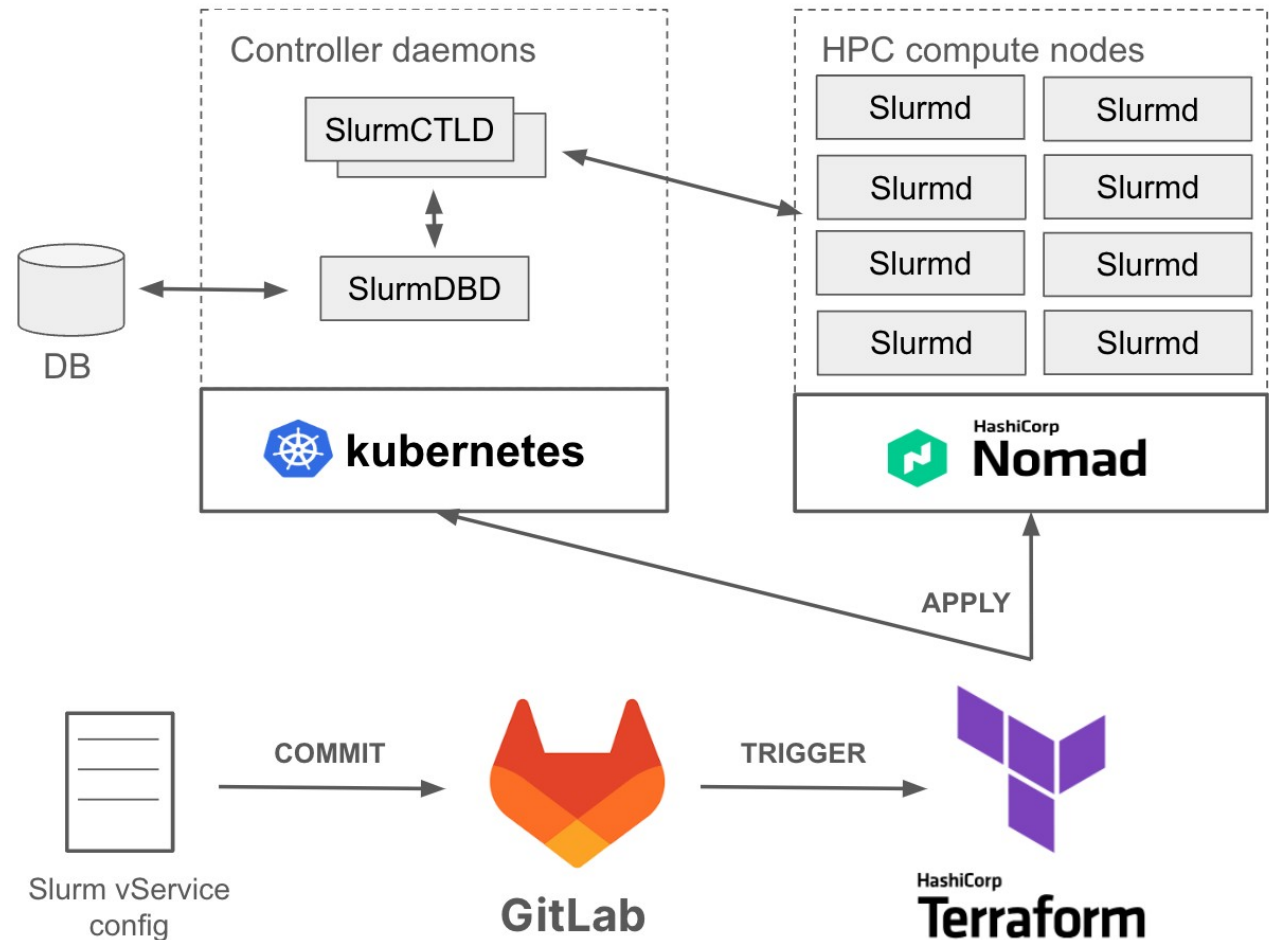
- Terraform plays a fundamental role in provisioning and managing the underlying infrastructure for both Kubernetes and Nomad environments
- By using IaC, Terraform ensures that all service components are deployed consistently and are reproducible
- This process is critical for:
 - Quick and consistent setup of the required services, reducing manual configuration errors and increasing deployment speed
 - Central management of both containerized and native service components, simplifying the complexity of operating diverse technological stacks while ensuring that the component deployment order is respected

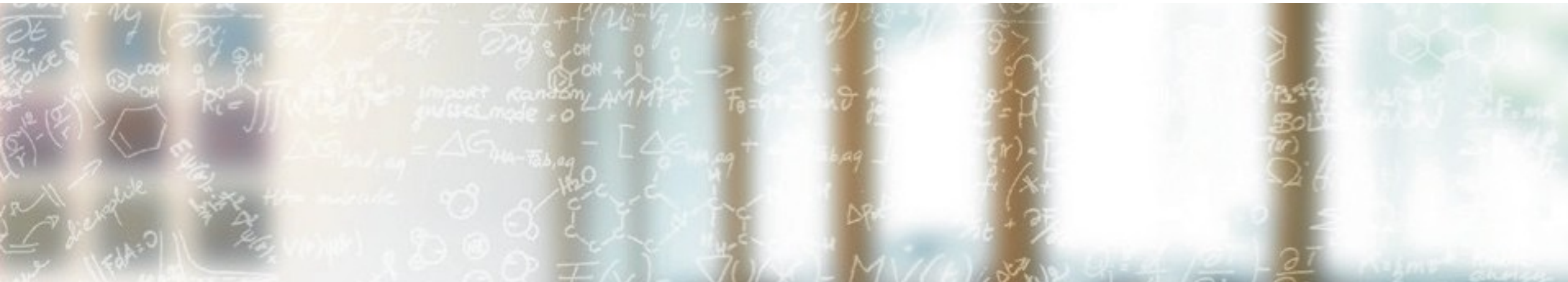
Slurm vService (Gitlab)

- GitLab integrates the entire deployment process through GitLab runner implementation of CI/CD pipelines, automating the deployment and version control of the Slurm implementation
- This ensures that all deployments are consistent, reproducible, and automated, streamlining the management of service changes:
 - Automated pipelines deploy updates and new configurations of services without downtime, improving the responsiveness to changes
 - Version control of configurations provides improved tracking and management for all service changes

Slurm vService workflow

- The vCluster admin commit changes to the GitLab repository that holds the Cluster configuration
- Changes trigger GitLab's CI/CD pipelines which interacts with Kubernetes and Nomad
- Kubernetes orchestrate the deployment of the containerized application
- Nomad schedules and manages native Slurm daemon tasks directly on the compute nodes





Thanks for your attention

SLUG '24 Oslo
Gennaro Oliva, CSCS
September, 2024