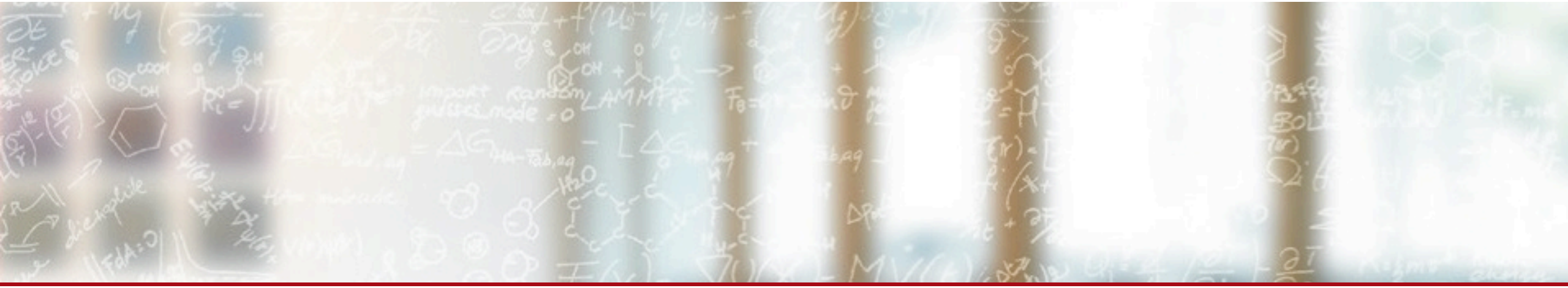
CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



Getting Started on Alps Platforms

Webinar for the CSCS User Community, Feb 13 2025

Luca Marsella, Fawzi Roberto Mohamed and Matthias Kraushaar, Service Managers at CSCS

Outline of the Webinar

- Introduction of Platforms
- Policies and Service Catalogue
- Documentation and Troubleshooting
- Community Slack Space and Status Page

Alps infrastructure in the CSCS machine room
Source: https://www.cscs.ch/computers/alps

Introduction of Platforms

Glossary

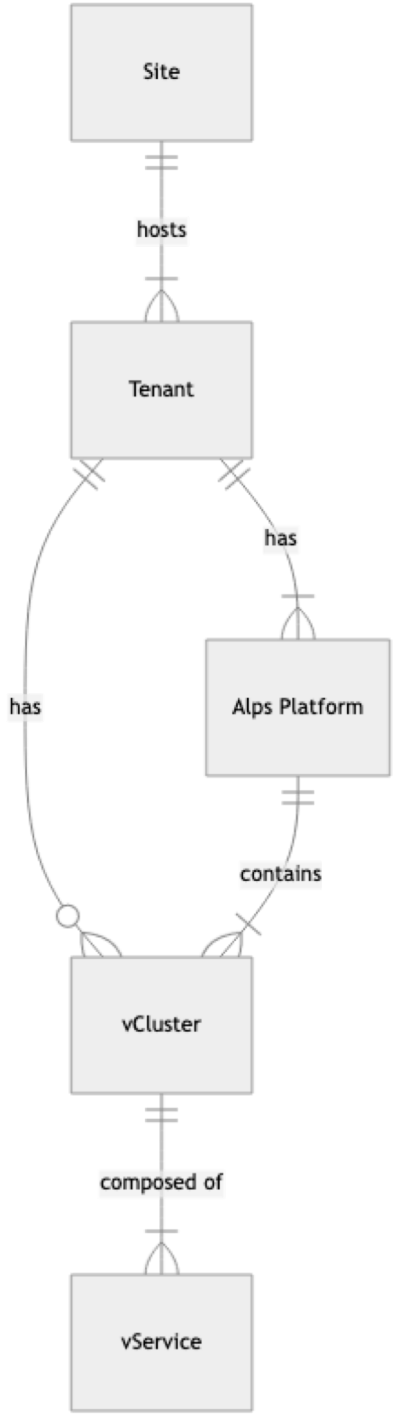
Platform A subset of Alps on top of the infrastructure that enables the deployment of one or multiple vClusters

Site A physical place with the facilities to operate the infrastructure (CSCS in Lugano, EPFL in Lausanne, PSI in Villingen, ECMWF in Bologna)

Tenant Organization holding the underlying platform (CSCS, MCH, PSI,…)

vCluster (**v**ersatile **cluster**) Fusion of HPC and cloud technologies providing software-defined clusters on a supercomputing ecosystem

vService (**v**ersatile cluster micro**service**)
Small, independent, and loosely-coupled service hosted on a vCluster



HPC Platform

The HPC Platform targets general purpose HPC computing on Alps

Resources

- **GPU** nodes on **Daint.Alps** (quad-socket Grace-Hopper superchip nodes)
- **Multicore** nodes on **Eiger** (dual-socket AMD CPU-only nodes)

Audience includes

- **User Lab** (Peer reviewed allocation programs)
- **Partners** (EMPA, ETH Zurich, Marvel, USI, UZH …)
- cscs2go (targeting primarily Swiss researchers from Academia and SMEs)

User and Allocation Management

Account and Resources Management Tool at https://account.cscs.ch

Users

- Access your profile, list your projects (active and closed ones)
- Monitor your project's compute budget (quarterly allocation and usage)
- Check the usage of storage resources on home and scratch

Principal Investigators (PIs)

- Invite users, manage project membership and access control
- Allow or deny users access to systems and project's data
- Define deputies and delegate project's administration

More information on the tool at https://confluence.cscs.ch/x/VQBYLw

Submission and Monitoring of Batch Jobs

CSCS systems use the Slurm workload manager for user jobs management

Most frequently used Slurm commands

- **sbatch script.sh** – submit a Slurm batch script
- **squeue -u $USER** – check the status of your jobs
- **scancel <jobID>** – remove your <jobID> from the queue

The command **sinfo** provides information on nodes and partitions

- **sinfo -sp normal** partition summary with **TIMELIMIT** and **NODES(A/I/O/T)**

  normal* up 1-00:00:00 751/13/60/824
- Please note that nodes might be reserved: **scontrol show reservation**

More information on **Daint** and **Eiger** in the CSCS Knowledge Base

Machine Learning Platform (MLP)

The Machine Learning Platform targets AI/ML computing on Alps

Resources

- **GPU** nodes on **Clariden** (quad-socket Grace-Hopper superchip nodes)
- Additional vClusters with NVIDIA A100 GPUs

Audience is primarily the **Swiss AI** Initiative

- Build capacities for advanced, large-scale AI systems for the benefit of society
- Over 70 professors from Swiss universities and research centres
- Co-led by ETH Zurich and EPFL

MLP

- Docs:
  - https://docs.cscs.ch/
- Clariden
  - Main cluster
  - ~1300 GH nodes
- Bristen
  - ~30 AMD+A100
  - Dev and inference
  - Public IPs

Knowledge Base

- Access and Accounting
- User Guides
  - Alps (Clariden) User Guide
    - **Getting started on Clariden**
    - Storage in Clariden
    - Debug in your containers with ...
    - How To Use Container images...
  - Alps (Eiger) User Guide
  - Daint
  - Running jobs
- Programming Environment
- Storage
- Tools
- Tutorials
- Frequently Asked Questions (FAQ)
- How-to articles
- Troubleshooting articles
- Scientific Applications
- Hidden content (CSCS Staff only)

Getting started on Clariden

Clariden is a vcluster (versatile software-defined cluster) that's part of the Alps system.

A short summary of the hardware available in the nodes:

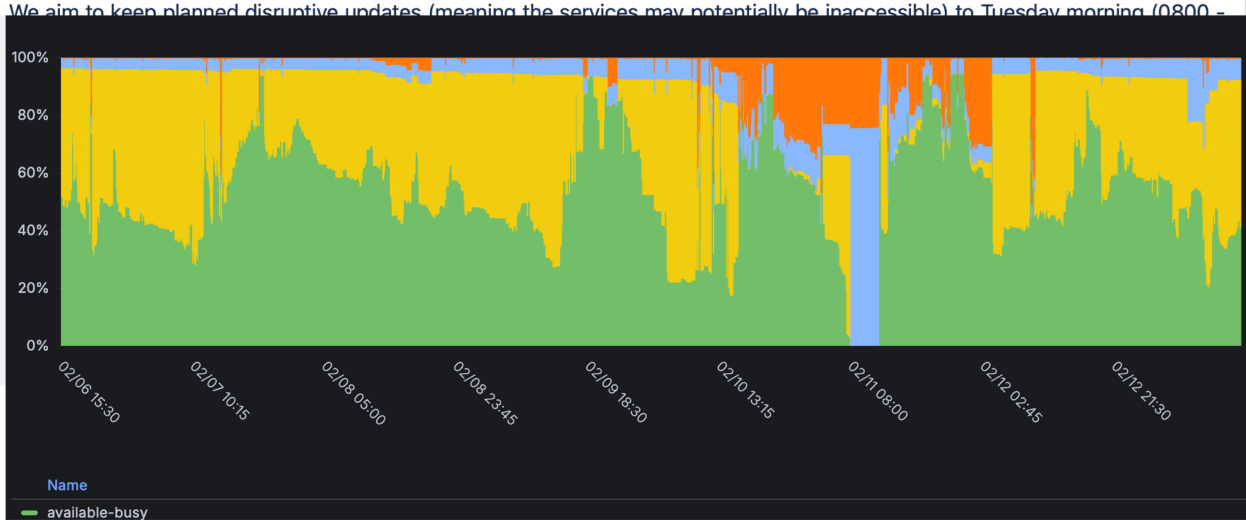
Partition	NNodes	GPUs per node	GPU	GPU memory	Max time
normal	1298	4	GH200	96GB	24 hours
debug	32	4	GH200	96GB	30 minutes

Each node consists of 4xGH200 superchips. Each superchip is a unified memory system consisting of a Grace CPU and a Hopper GPU with a 900GB/s NVLINKC2C connect. The Grace CPUs share 512GB of LPDDR5X memory. Each individual Hopper GPU has 96GB of HBM3 memory with 3000GB/s read/write, totaling 896GB of unified memory available within each node.

More information on the available partitions can be found with `scontrol show partitions.`

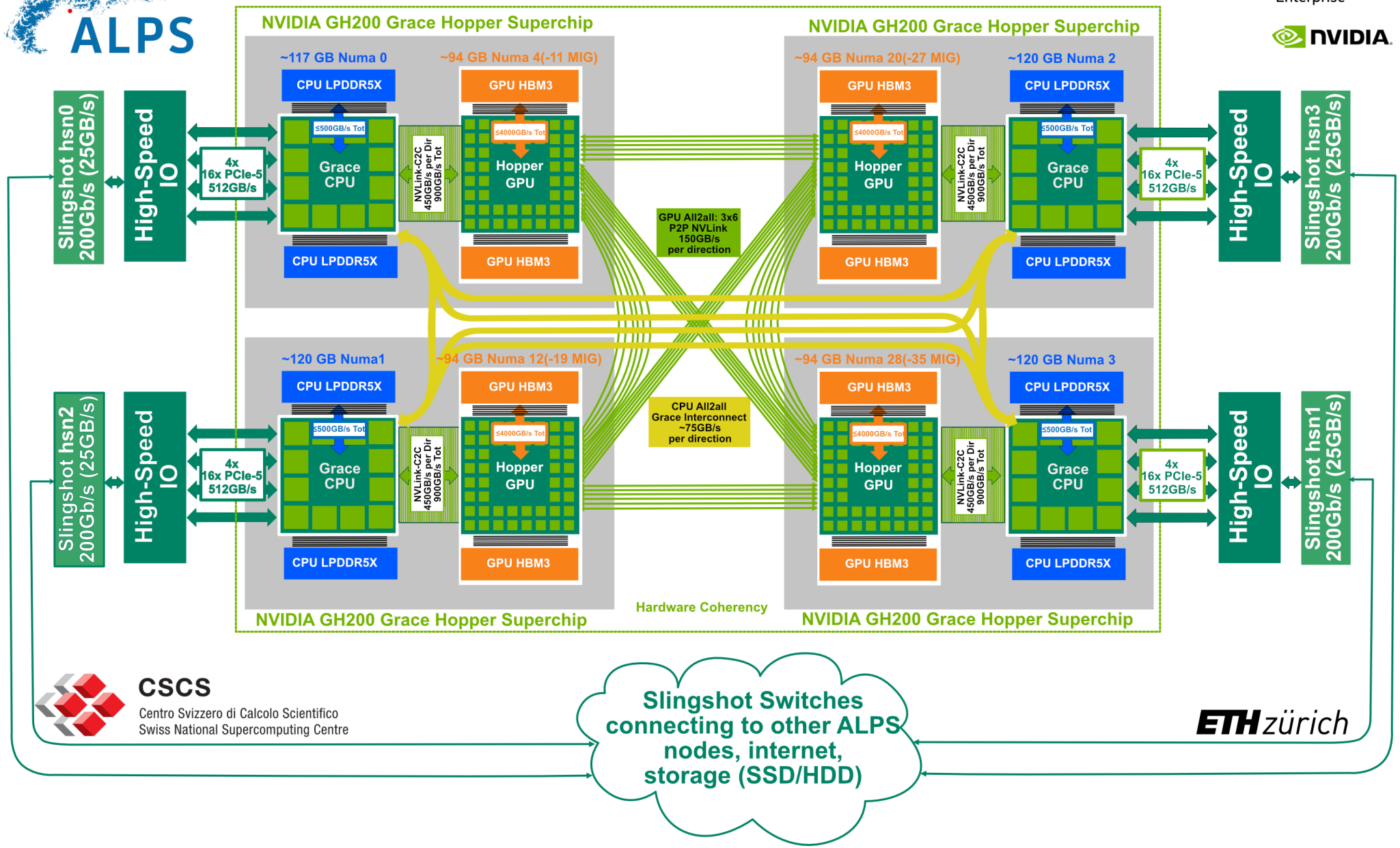
Maintenance

- We aim to keep planned disruptive updates (meaning the services may potentially be inaccessible) to Tuesday morning (0800 -



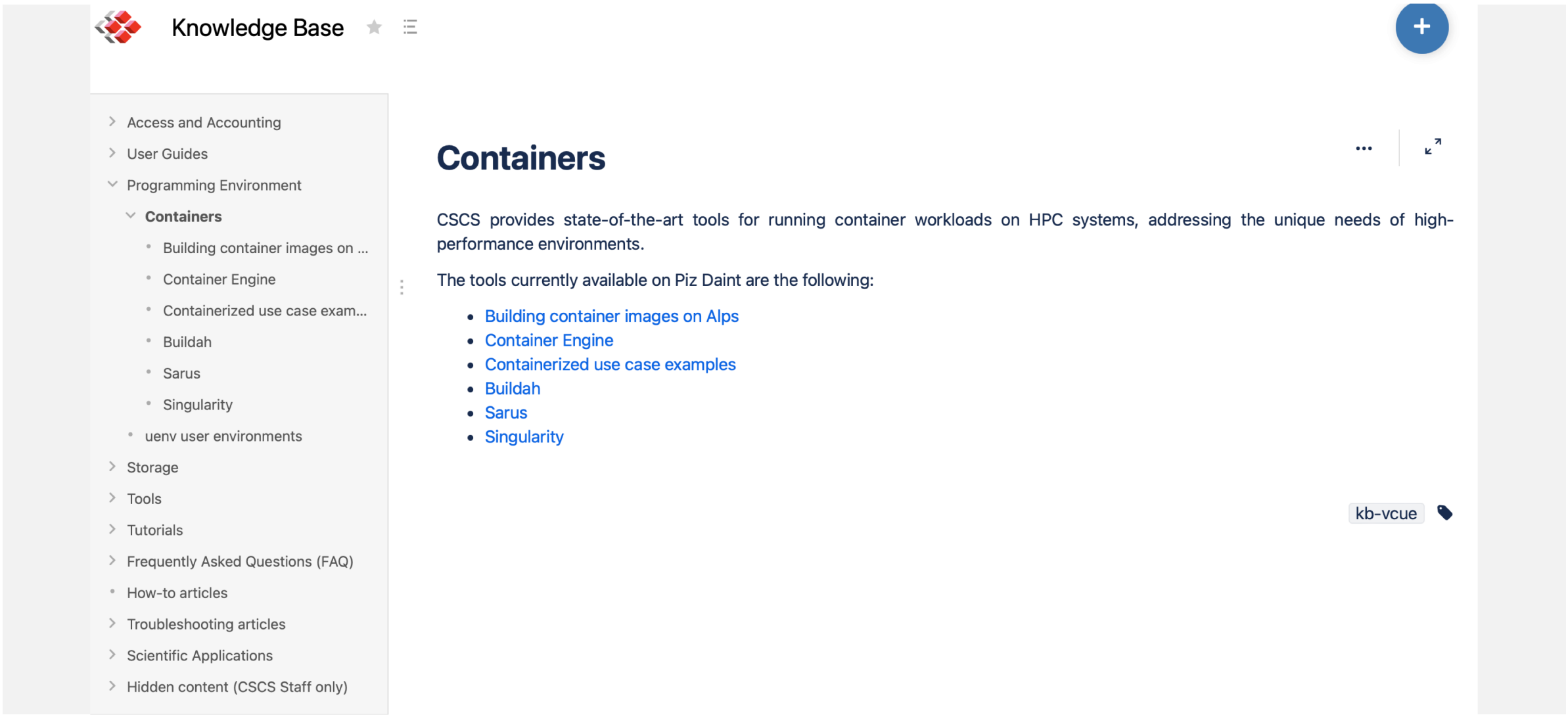
CSCS ALPS HPE Cray EX GH200 Node

4xGH200 NVIDIA Grace Hopper Superchips ~477 GB (CPU) + ~378 GB (GPU) = ~855 GB Unified memory

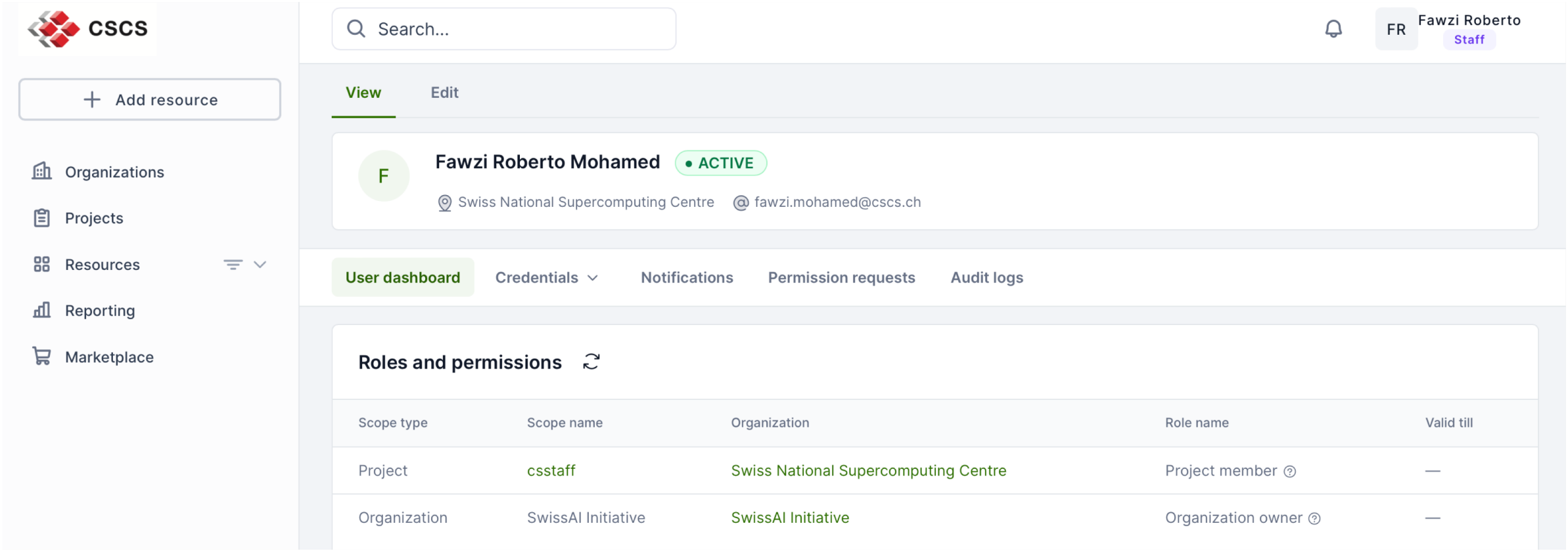


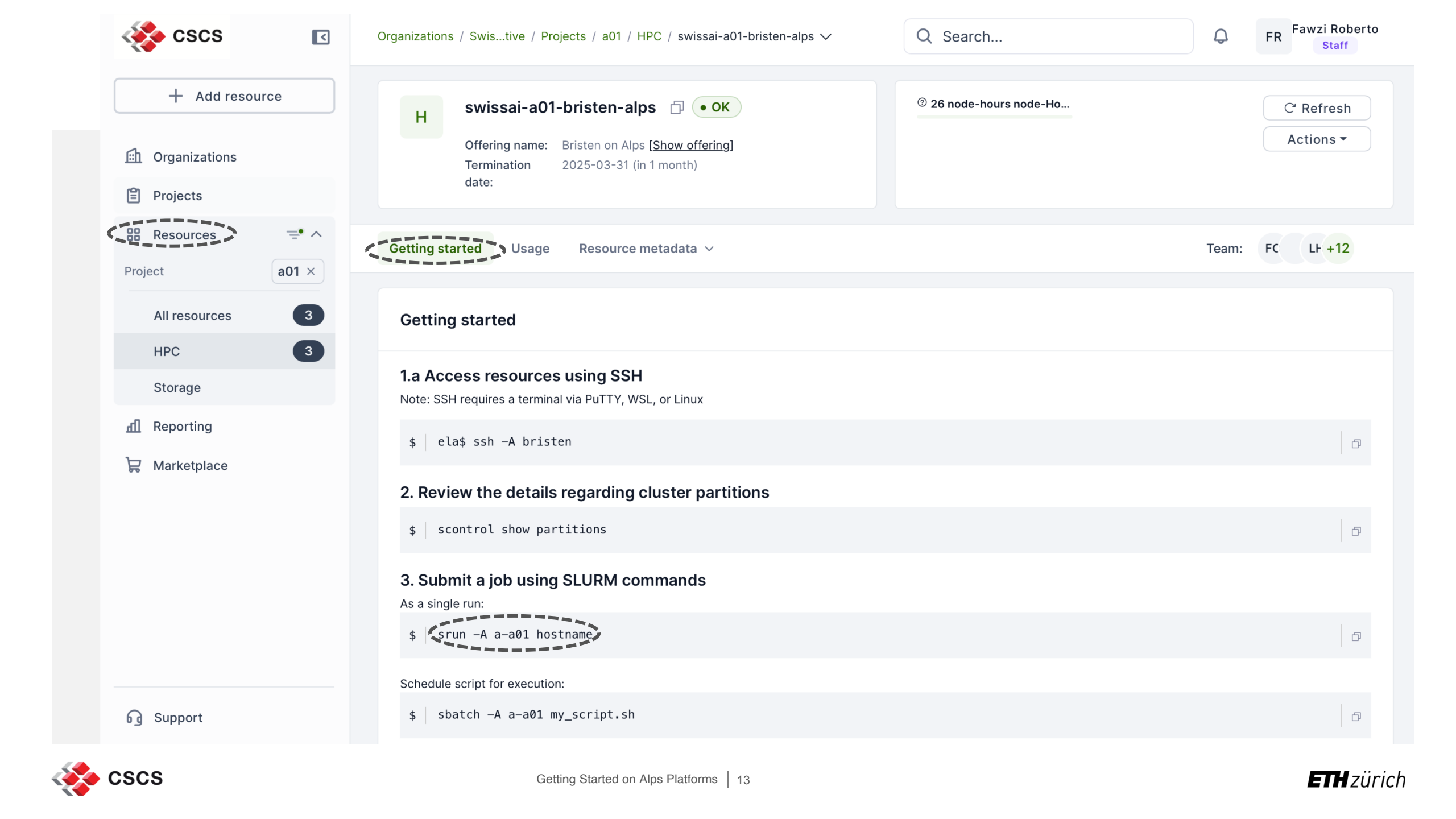
CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

Containers

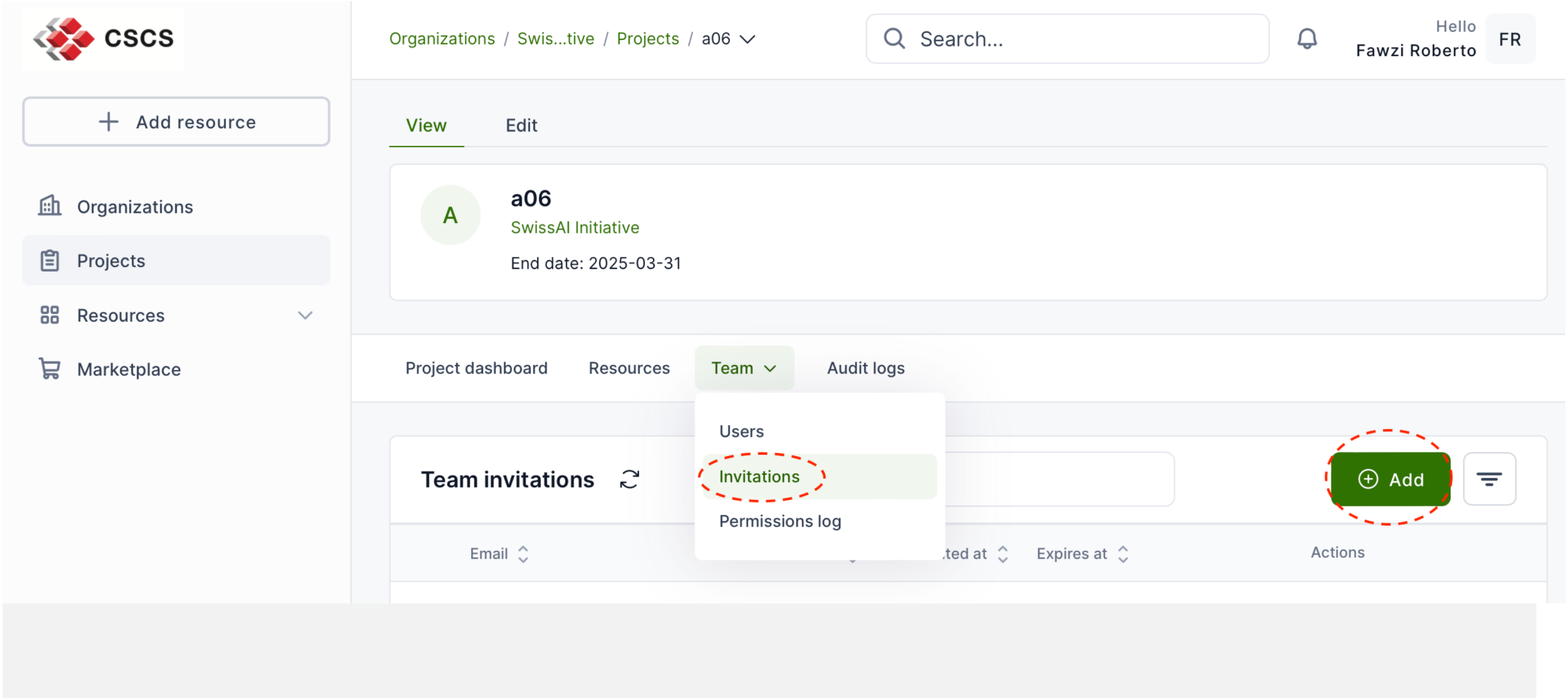

Knowledge Base
Access and Accounting
User Guides
Programming Environment
Containers
Building container images on ...
Container Engine
Containerized use case exam...
Buildah
Sarus
Singularity
uenv user environments
Storage
Tools
Tutorials
Frequently Asked Questions (FAQ)
How-to articles
Troubleshooting articles
Scientific Applications
Hidden content (CSCS Staff only)
Containers
CSCS provides state-of-the-art tools for running container workloads on HPC systems, addressing the unique needs of high-performance environments.
The tools currently available on Piz Daint are the following:
Building container images on Alps
Container Engine
Containerized use case examples
Buildah
Sarus
Singularity
kb-vcue

User and budget management: https://portal.cscs.ch/



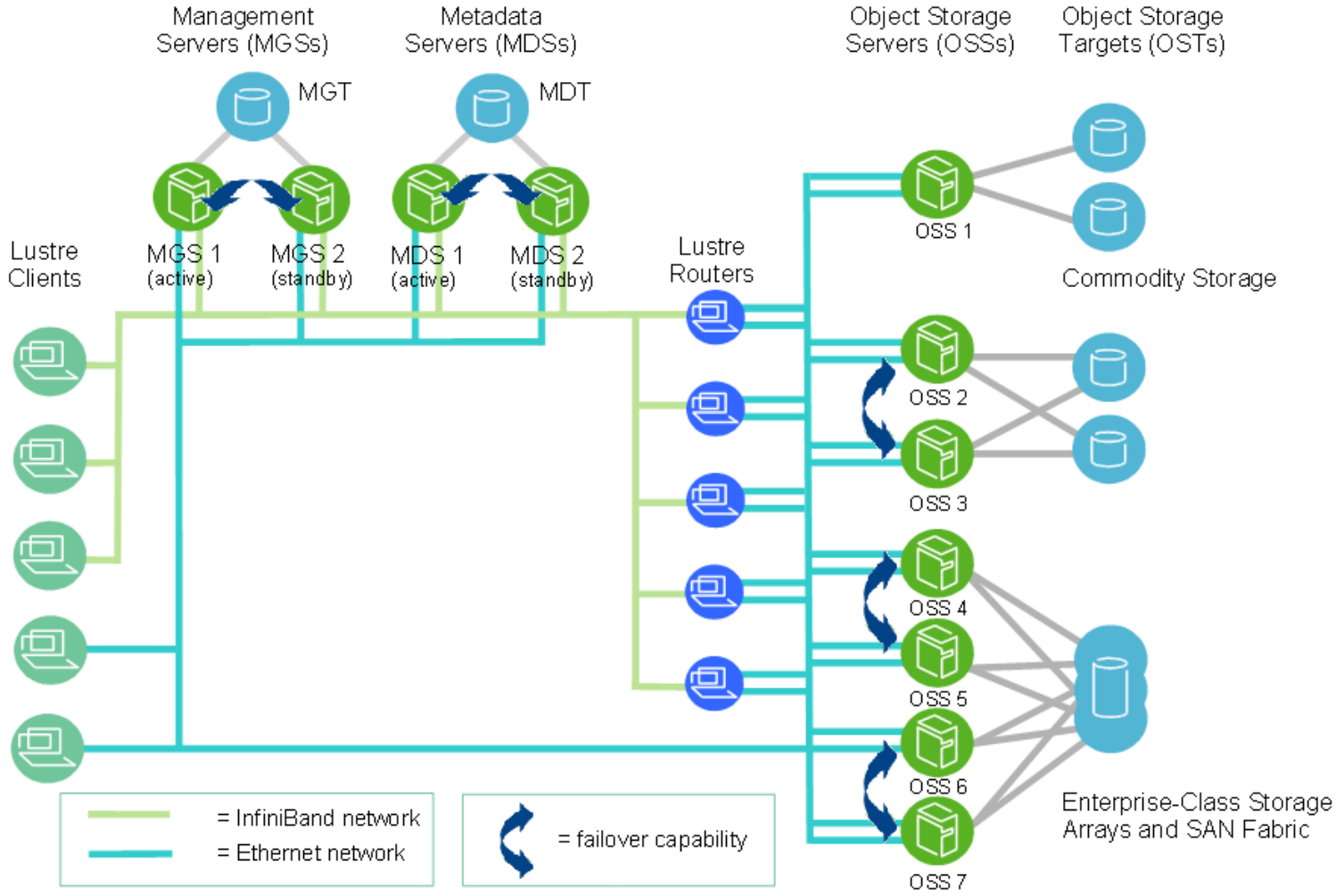

CSCS
Add resource
Organizations
Projects
Resources
Project
a01
All resources 3
HPC 3
Storage
Reporting
Marketplace
Support
Organizations / Swis...tive / Projects / a01 / HPC / swissai-a01-bristen-alps
Search...
FR
Fawzi Roberto
Staff
H
swissai-a01-bristen-alps
OK
Offering name: Bristen on Alps [Show offering]
Termination date: 2025-03-31 (in 1 month)
26 node-hours node-Ho...
Refresh
Actions
Getting started
Usage
Resource metadata
Team: FC LH +12
Getting started
1.a Access resources using SSH
Note: SSH requires a terminal via PuTTY, WSL, or Linux
$ ela$ ssh -A bristen
2. Review the details regarding cluster partitions
$ scontrol show partitions
3. Submit a job using SLURM commands
As a single run:
$ srun -A a-a01 hostname
Schedule script for execution:
$ sbatch -A a-a01 my_script.sh

Invite new members



Lustre

- lfs osts
  - capstor: 160
  - iopsstor: 20
- large file:
  - lfs getstripe -d dir/path
- capstor
  - lfs setstripe --stripe-cou
- iopsstor
  - lfs setstripe --stripe-count 10 --stripe-size 4194304 directory/file



Climate and Weather Platform (CWP)

The Climate and Weather Platform targets climate science computing on Alps

Resources

- **GPU** nodes on **Säntis** (quad-socket Grace-Hopper superchip nodes)

Audience includes

- **EXCLAIM**
  Project for ICON-based km-scale climate simulations (ETH Zurich)
- **C2SM** Center for Climate Systems Modelling
  Eawag, EMPA, ETH Zurich, MeteoSwiss, WSL
- User Lab projects in climate and weather domains

Säntis vCluster of the CWP

DNS: `santis.alps.cscs.ch` (currently also `santis.cscs.ch` – will be deactivated by end of 25Q1)

Slurm cluster providing 3 partitions

SLURM partition	Default wall time	Max. wall time	limitations
normal	01:00:00	24:00:00	-
debug	00:30:00	00:30:00	Single node
xfer	06:00:00	24:00:00	Single node

Major services currently available

- **User Environments (UENVs)**
- **Container Engine**
- **CI/CD via Gitlab runners**
- **FirecREST (RESTful API to access the system programmatically)**

Services currently in development

- **JupyterHub**

Policies

General Policies

The code of conduct outlines some responsibilities of the user community

- **Access to Source Codes:** you agree to make codes available for support
- **Scientific Advisory Committee:** committee members must not be contacted
- **Acknowledgements:** you must acknowledge the use of CSCS resources in all publications related to your allocation with reference to the "*project ID ###*"

User Regulations define guidelines for the usage of CSCS resources

- **Accounts are personal** and sharing them is forbidden
- **Data ownership**: access to and use of data of other accounts without prior consent from the principal investigator is strictly prohibited
- **ETH Zurich Acceptable Use Policy for Information and Communications Technology**

Access to CSCS resources **may be revoked to users violating these policies**

Data Retention Policies

Data backup for **active projects**:

- Data in **/users** and **/capstor/store** folders is backed up (past 90 days)
- Data in **/capstor/store** folders removed **3 months** after the **end of the project**

As soon as a project **expires**:

- Data backup is **disabled** immediately
- **No data recovery** after the final data removal

No backup for data on **scratch file system(s)**:

- No recovery in case of **accidental data loss**
- No recovery of data deleted due to the cleaning policy

More information on **Data Recovery** at https://confluence.cscs.ch/x/kYBOLw

Policies of the scratch file system

The capacity storage (capstor) features 100 PB in a Lustre file system

Cleaning policy: files **older than 30 days deleted** daily

No backup: you should transfer data after batch job completion

Soft quotas to prevent performance degradation

- Quota on **total number of files and folders** (inodes**)**
- Quota on **amount of data** with a **grace time** to allow data transfer

More information on **scratch** at https://confluence.cscs.ch/display/KB/scratch

Fair Usage Policies

Slurm workload manager

- The batch scheduler is a shared resource
- **Do not continuously poll the scheduler** to determine job states
- **Do not submit hundreds of jobs** in a short space of time
- We may be forced to kill jobs and limit new submissions

Login (User Access) nodes

- **Compute- or memory-intensive processes are not permitted**
- Heavy processes will be terminated **without warning**
- **Submit Slurm batch jobs** to run application codes on compute nodes

Support Policies

We offer timely response with **direct assistance** or **escalation processes**

- Support is **limited to the scope** of your project proposal
- We cannot guarantee the resolution of all issues

Supported community codes and user applications

- We are committed to help you run up-to-date supported community codes
  CSCS staff cannot fix application-specific issues
- Best-effort support deploying and optimizing user applications

Prioritization case impact, complexity, knowledge transfer, time to solution

Collaborative support consult online docs and provide detailed information

Resource Allocation Policies

Computing time on vClusters is allocated and accounted in **node hours**

- Resources are assigned in **three-month windows**
  - **Quotas are reset** on April 1st, July 1st, October 1st and January 1st
- Try to use your compute budget evenly and consistently across the windows
- Unused resources in a given window **cannot be transferred to the next**

More information on **Policies** at https://confluence.cscs.ch/x/UQBYLw

Service Catalogue

Multi-factor authentication and SSH connections

Users are required to authenticate using multi-factor authentication (MFA)

- MFA implemented at CSCS as **two-factor authentication**
- Users receive an email with the details about the procedure
  - One factor is the **user login and password pair** ("thing you know")
  - The other factor is the **one-time password** (OTP, "thing you have")

MFA applies to web-based services and SSH connections

- CSCS supports MFA authenticators that follow the TOTP open standard
- SSH is only possible with keys generated by the CSCS SSHService
- **Keys are valid for 24 hours**, after which a new key must be generated

More on **MFA** at https://confluence.cscs.ch/x/VwBYLw

Programming Environment

We provide several ways to build software for running on vClusters

- **uenv** developed and fully supported at CSCS
- Cray Programming Environment (CPE) supported via HPE/Cray
- Containers (often a good choice for machine learning workloads)

No modules are loaded by default at login!

- You need to load the module **cray** first with the following command
  module load cray
- Then the modules available in the default CPE can be listed with
  module avail

More on **CPE** at https://confluence.cscs.ch/x/_gD0E

uenv: User Environments

uenv provide programming environments and application software

Typically application-specific, domain-specific or tool-specific

- **uenv** contains only what is required for the application or tool to run
- CSCS provides some ready-made and supported user environments
- Advanced users can create their own **uenv** with the tools provided

Packaged in a single **squashfs** file

- Can be shared in an artifactory or stored on a file system
- Independent of each other and of CPE
- Built on top of the base image, not the CPE

uenv: Tools

Command Line Interface (CLI)

- **uenv** command line tool for environments
- **uenv image find** search for uenv image that can be pulled

Slurm plugin at https://github.com/eth-cscs/slurm-uenv-mount

- Manages the loading of images on compute nodes

Stackinator at https://eth-cscs.github.io/stackinator

- Tool for generating images from a declarative recipe

Github with CSCS images at https://github.com/eth-cscs/alps-spack-stacks

- CI/CD pipeline to build, test and deploy images

uenv: Mount Points

/user-environment

- **Programming environments**
  Compilers, libraries, profilers (e.g.: HDF5, FFTW, MPI, OpenBLAS…)
  Can be application specific (e.g.: supporting ICON builds)
- **Application environments**
  CP2K, GROMACS, LAMMPS, NAMD, QuantumEspresso, VASP…

/user-tools

- **Supporting tools**
  - Debuggers (DDT)
  - Text editors
  - Visualization (ParaView, Visit, …)

uenv: Benefits for You

Single image in a SquashFS file
- Managed in a registry/artifactory
- Performance decoupled from file system

Defined by a **simple declarative recipe**
- The same environment can be easily rebuilt
- Useful after system upgrades

Small set of system dependencies
- Only need to rebuild if **libfabric** or **Slurm** are changed

More on **uenv** at https://confluence.cscs.ch/x/bYDTKQ

Container Engine

Container Engine **runs Linux containers** on HPC environments

- Developed following the specific requirements of HPC systems
- Spawns isolated containers built by users for a specific application
- Extensible runtime by means of OCI hooks for custom hardware

Compatible with the Open Container Initiative (OCI) standards

- Pulls from registries with OCI Distribution Specification or Docker
- Imports and convert images with the OCI Image Format (e.g. Docker)
- Uses an OCI-compliant runtime to spawn the container process

More on **Container Engine** at https://confluence.cscs.ch/x/B4BFLg

Continuous Integration / Continuous Deployment (CI/CD)

CI/CD to **build containers** and run them at CSCS

- Dockerfile with build instructions
- Name for the container in the registry
- The build will be taken care by the CI implementation

Pipelines: git providers with webhooks (GitHub, GitLab, Bitbucket)

- A typical pipeline consists of at least one build job and one test job
- Build job: a new container with your most recent code changes is built
- Test job: the new container is run within an MPI job

More on **CI/CD** at https://confluence.cscs.ch/x/UAXJMw

FirecREST

FirecREST is a RESTful API for managing HPC resources at CSCS

- Integrate FirecREST into **web-enabled portals and applications**
- Securely **access CSCS services** (job submissions, data transfer,…)
- Hands-on introduction with PyFirecREST
  - Python wrappers for the FirecREST API to showcase its functionality

HTTP requests to perform various operations

- Basic **system utilities** like ls, mkdir, mv, chmod, chown,…
- Actions with the **Slurm** workload manager (submit, query, cancel jobs)
- **Data transfer**: internal (between CSCS systems) and external

More information on **FirecREST** at https://confluence.cscs.ch/x/ewBWM

Interactive Computing with JupyterLab

JupyterLab is the web-based user interface for Project Jupyter

- Create and share **documents with live code**, equations, visualization, …
- Ability to work with multiple documents using tabs or splitters side by side

We have refactored the JupyterHub implementation for higher performance

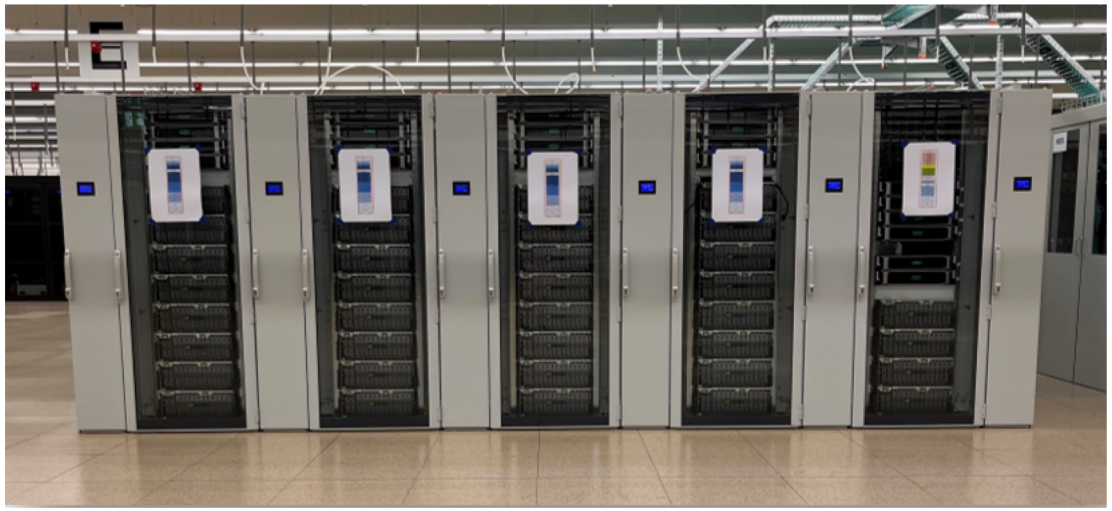
- It will be available soon on our three large vClusters
- Notebook servers are launched on compute nodes
- Much improved performance as Python components are on SquashFS
- Integration with **uenv**: add your kernels based on virtual environments

More on **JupyterLab** at https://confluence.cscs.ch/x/fQBWM

Storage Resources: Capacity Storage

Most vClusters on Alps mount the capacity storage (**capstor**) as scratch file system

- 100 PB Lustre built on spinning disks
- **Best I/O performance** on **large files**
- Raw performance
  1 TB/s I/O bandwidth
  300K write IOPS and 1.5M read IOPS



Storage Resources: Flash File System

The ML Platform also mounts the flash file system iopstor

- 3 PB Lustre filesystem built on Flash memory
- High **Input/Output operations per second (IOPS)**
- Raw performance
  240 GB/s write and 600 GB/s read bandwidth
  13 M write IOPS and 18.4 M read IOPS

Storage Resources: /users and /capstor/store

Storage for datasets, code and scripts

- Access **r+w** from the User Access Nodes (UAN)
- Better performance with larger files (archive small files with **tar)**

Permissions and data retention

- Folders **backed up**: data retention until 3-months after projects end

Environment variables for quick access

- **$HOME** pointing to personal folder **/users/$USER**
- **$PROJECT and $STORE** pointing to User Lab folder under **/capstor/store**

More on **File Systems** at https://confluence.cscs.ch/x/w4DOLg

Data Transfer

Data transfer service to get files from/to CSCS file systems (External Transfer)

- Service implemented using the Globus Online Endpoint
- The CSCS endpoint requires authentication with CSCS credentials

Data mover service with the dedicated Slurm **xfer** queue (Internal Transfer)

- The service submits jobs on the **datamover** cluster at no charge

More information on **Data Transfer** at https://confluence.cscs.ch/x/IIBOLw

Long Term Storage

Storage repository with **long term retention** capabilities

- Provide **persistent identifiers (PID)** and set public access if needed
- Data easily **accessible from a web browser** (HTTPS protocol)
- RESTful API to **integrate with third party** applications
- Scalable service that can cope with **large volumes** of data
- **Resiliency** due to data protection measures against failures

User Lab: up to 2 TB of LTS storage quota (for 10 years) free of charge per project

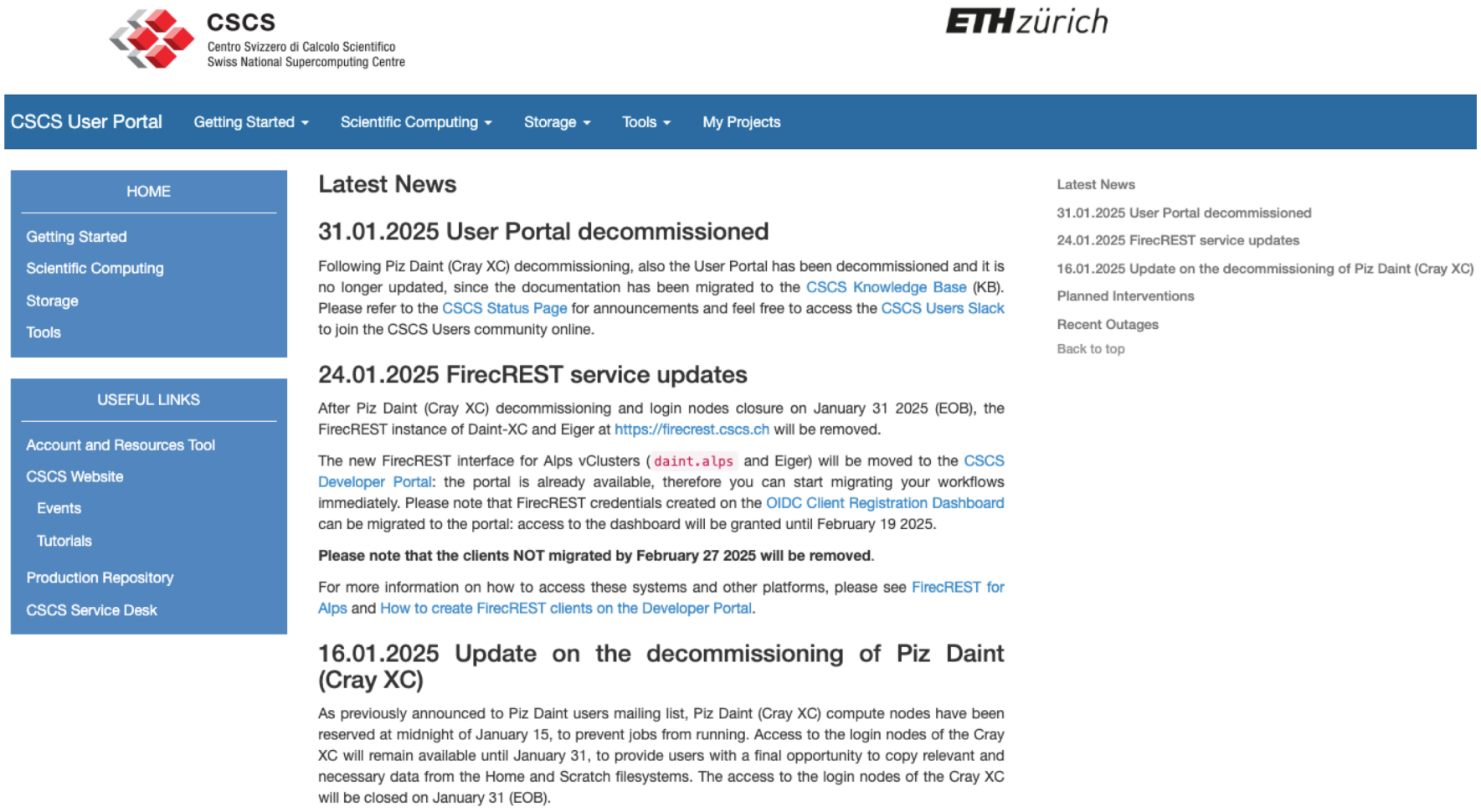
More information on **Long Term Storage** at https://confluence.cscs.ch/x/p4BOLw

Documentation

CSCS User Portal decommissioned

Documentation migrated CSCS Knowledge Base
http://docs.cscs.ch

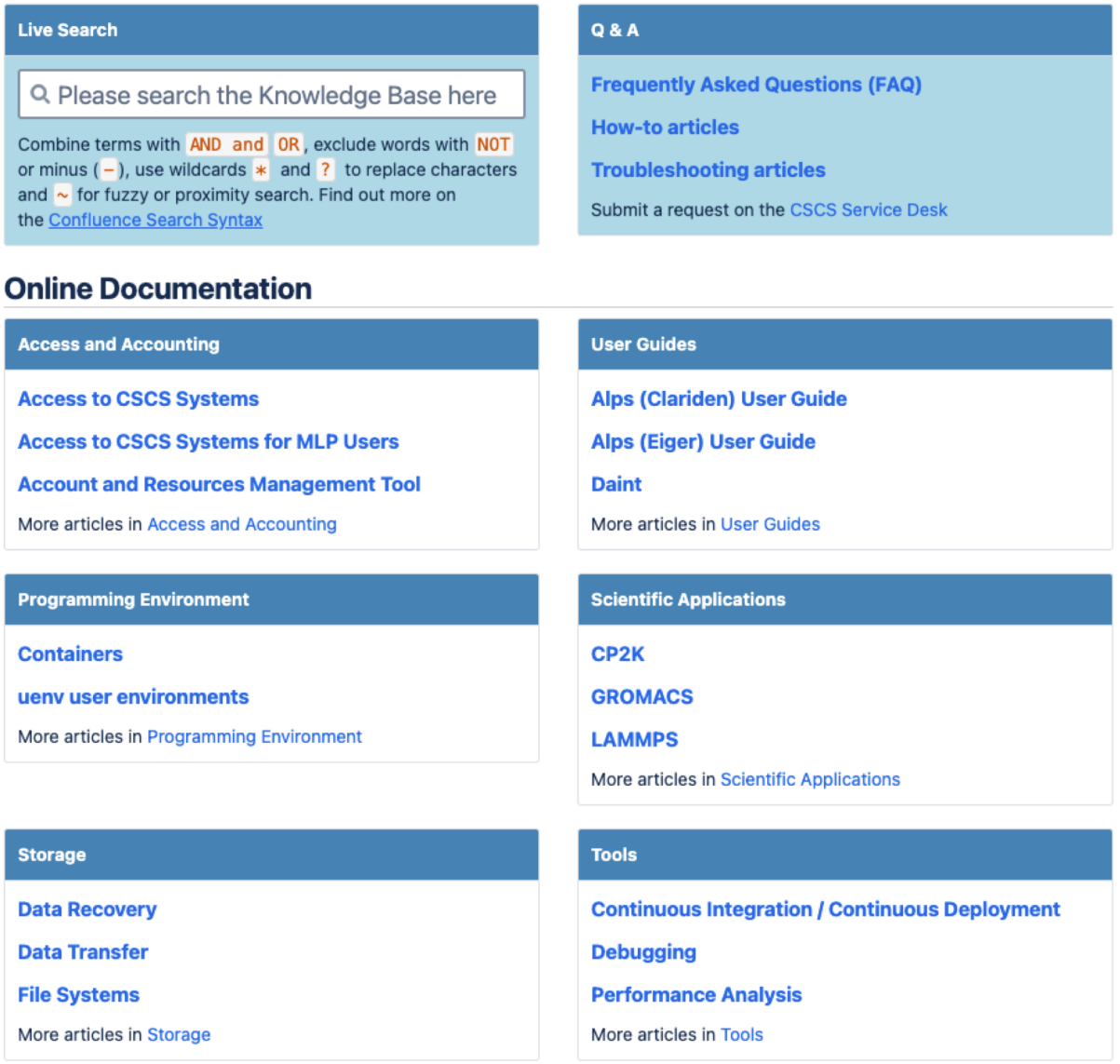
Announcements on CSCS Status Page
https://status.cscs.ch



CSCS Knowledge Base (KB) at https://docs.cscs.ch

The KB replaces the User Portal

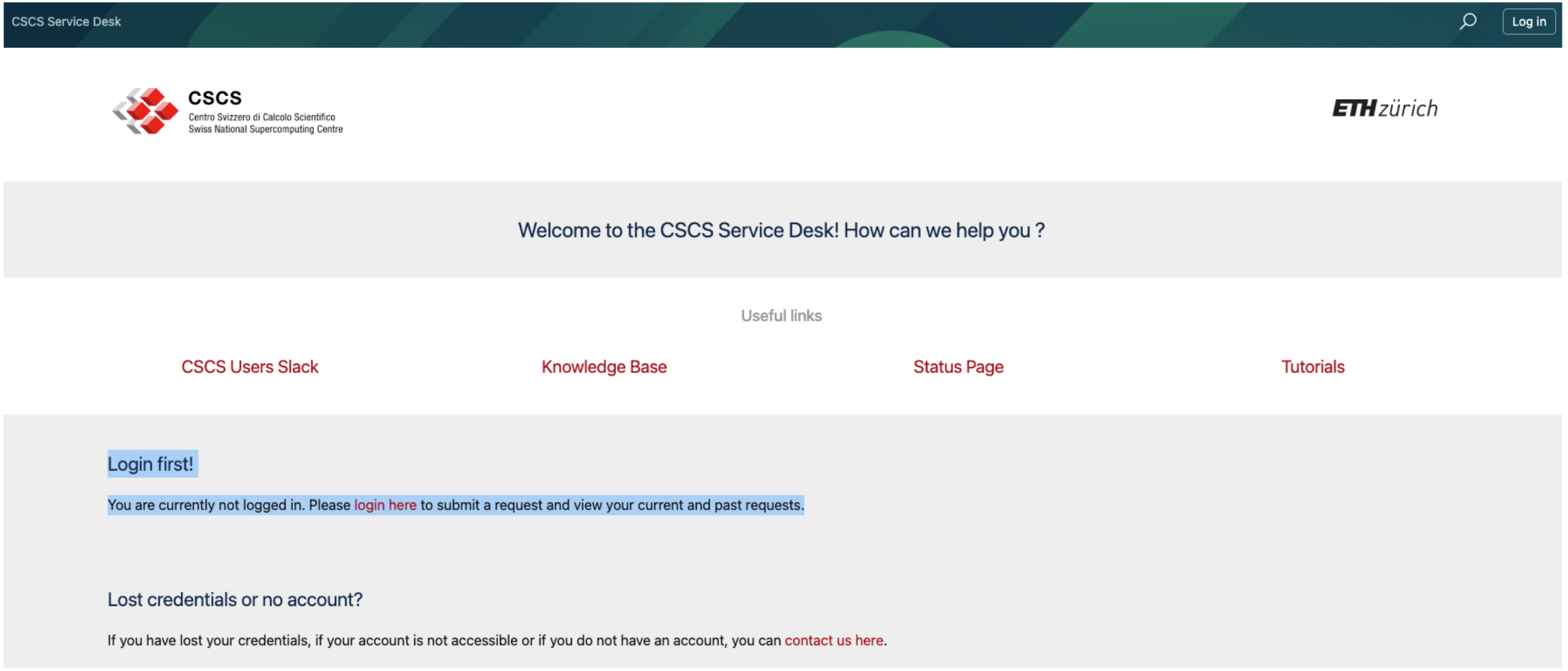
- Online Documentation can be **searched** easily with Confluence
- **Articles** displayed in panels following User Portal names
- New sections **How-to-articles** and **Troubleshooting articles**



Troubleshooting

What to do in case of trouble?

Access the CSCS Service Desk at https://support.cscs.ch



How to submit a support request

Useful links to documentation and status available online

- Knowledge Base (KB) at https://docs.cscs.ch
- Status Page at https://status.cscs.ch
- Tutorials at https://www.cscs.ch/publications/tutorials

Submit a support request **if you can't find a solution** in the documentation

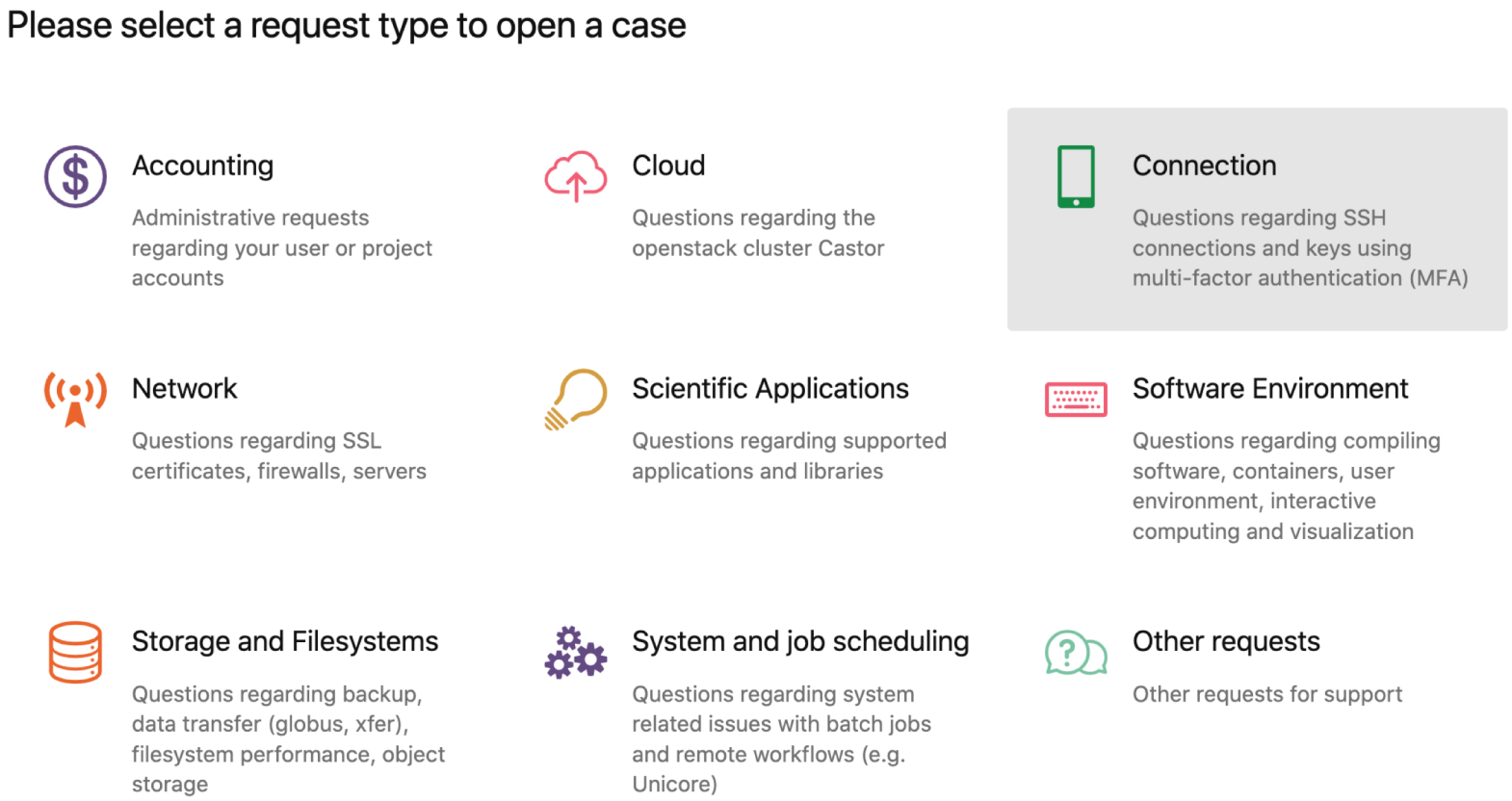
- **Login first** using your CSCS credentials
- The contact form should only be used if you cannot login

More on **How to submit a support request** at https://confluence.cscs.ch/x/LYH8LQ

Request types in the CSCS Service Desk

Requests types are only available after logging in

- Choose the best matching **request type**
- Use the type **Other requests** only if you can't find a match
- A better match will help us react faster



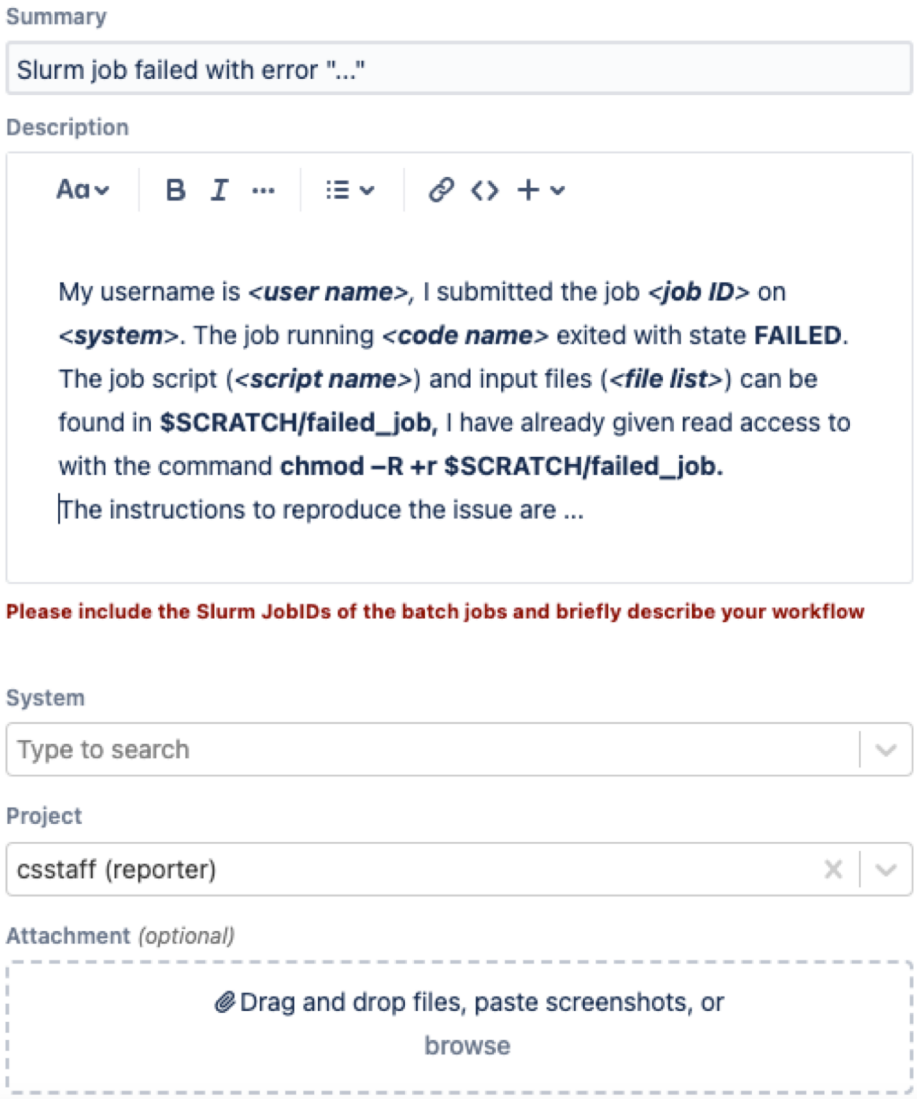
Example of a good support request

Summary

- Matching articles that might help will pop-up from the KB as you type

Description

- Please provide the information needed to **reproduce the issue**
- Include the **Slurm jobid** and the path to the **Slurm job script**
- Copy scripts and source files to **$SCRATCH** and give us the necessary permissions to access



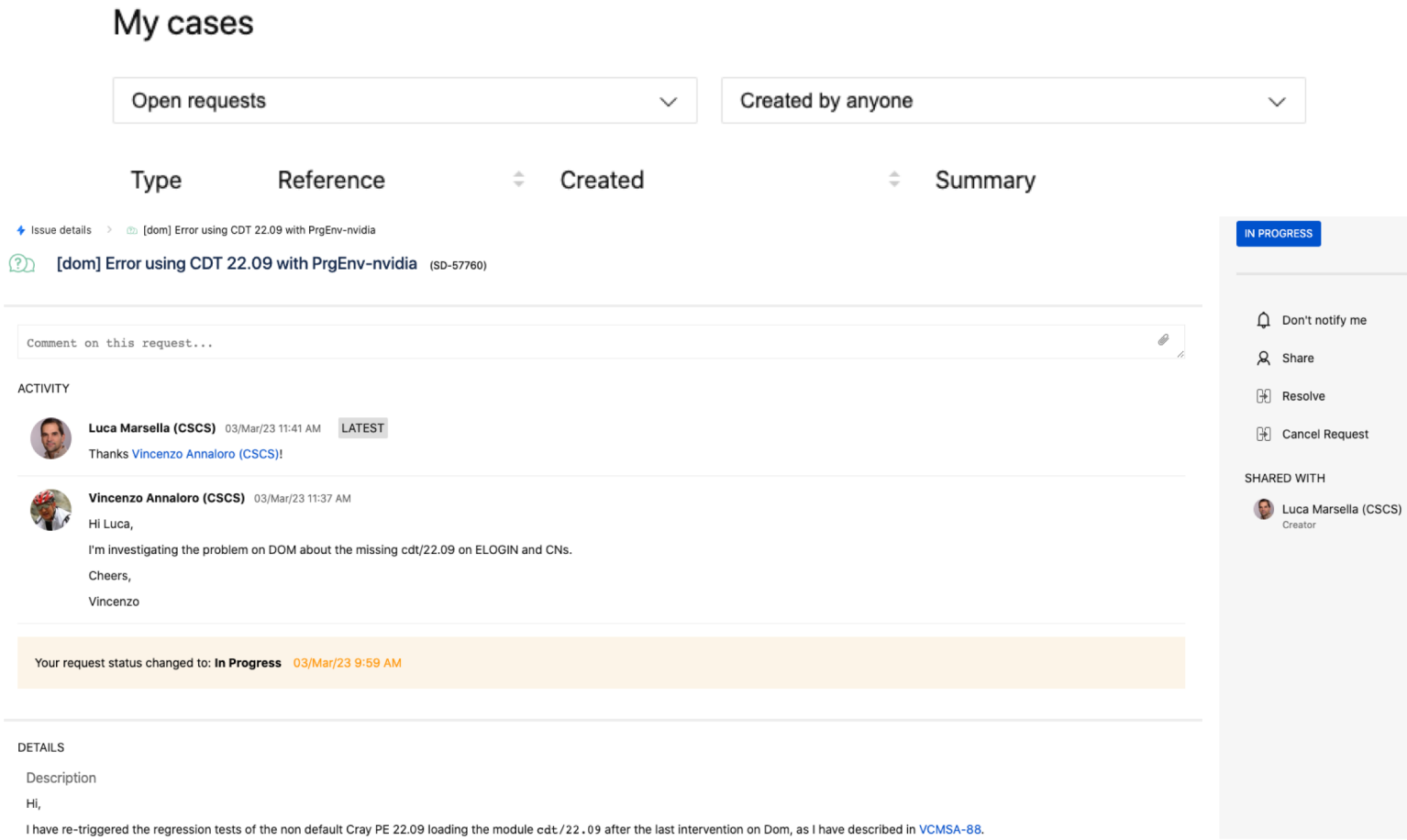
Monitoring your requests

Cases in the Service Desk:

- **Filter** the list of cases
- Check case **status**
- **Review** the messages

Select a specific case:

- **Share** with other users
- **Resolve** the case if solved
- **Cancel** if sent by mistake



Follow the links in the notifications to add comments online to your case
Replies by email won't be forwarded to your open support requests

Community Slack Space

Community Slack Space

CSCS Users Slack https://cscs-users.slack.com

A new space for you **to interact with fellow CSCS users**

- **Community** driven approach
- Get **rapid responses** to quick questions
- **Share experiences** and best practices

Recommendations

- Encourage your team members to join!
- Please always **be kind and considerate** to others
- Misbehaviour could result in a ban from the Slack space

Channels

The space includes several channels

- **#daint**
  Channel dedicated to the vCluster **daint.alps**
- **#general**
  Default Slack channel for general purpose posts
- **#introductions**
  Learn more about your fellow users and their science

Discover **more channels** on the Slack space

The Slack space **is not** meant for submitting support requests

- Use the CSCS Service Desk at https://support.cscs.ch

Status Page

Status Page at https://status.cscs.ch

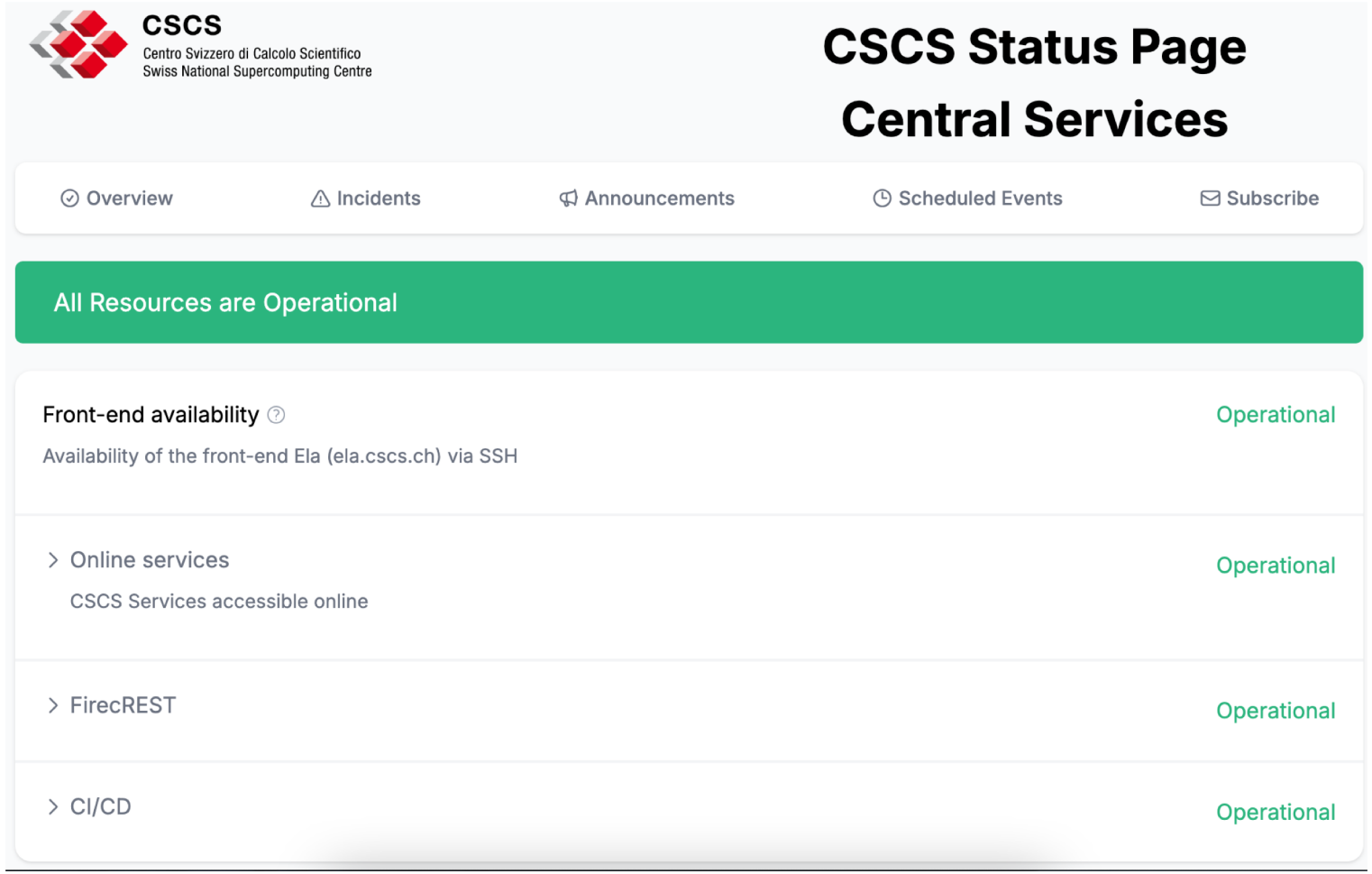
The Status Page reports

- **Incidents**
- **Announcements**
- **Scheduled Events**

Status of resources

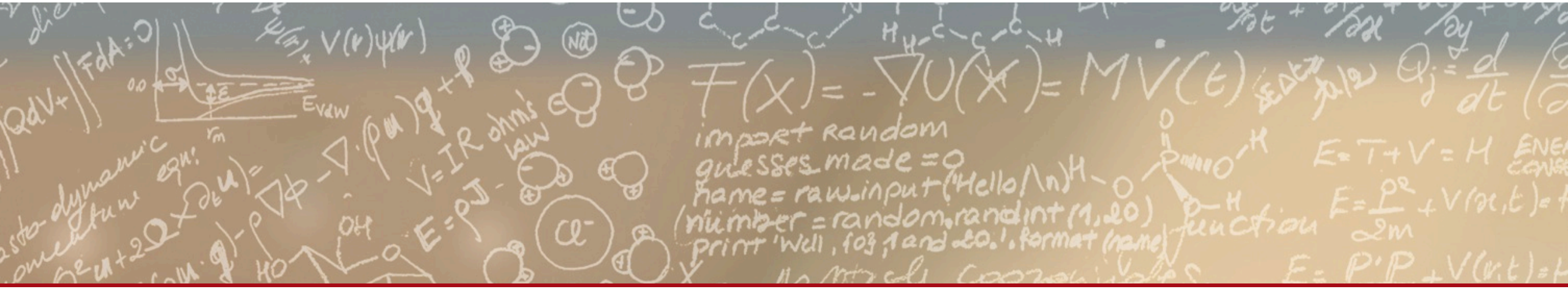
- **Online services**
- Front end system

Subscribe to get notifications



Useful Links

- CSCS Service Desk
  - https://support.cscs.ch
- CSCS User Slack
  - https://cscs-users.slack.com
- Knowledge Base
  - https://docs.cscs.ch
- Status Page
  - https://status.cscs.ch



Thank you for your kind attention