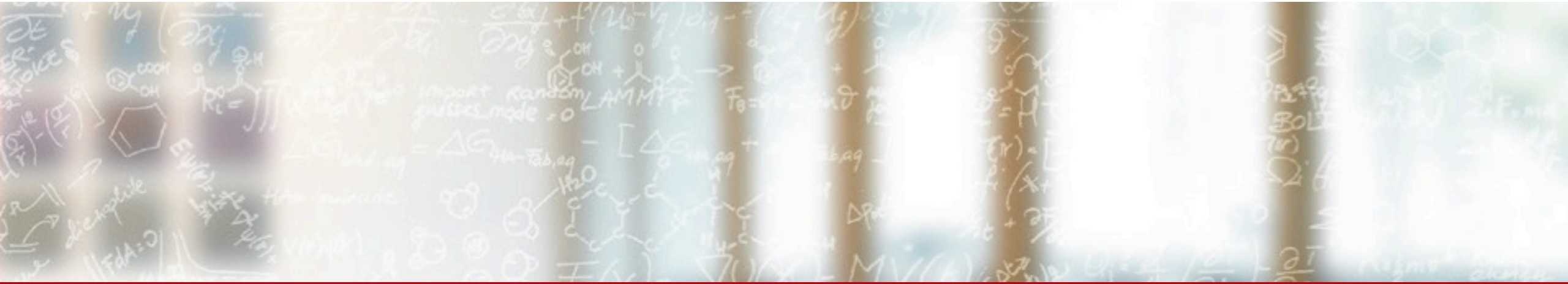




CSCS

Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



User Lab Policies & Service Catalog on Alps

User Lab Day 2023

Luca Marsella, CSCS

Outline of the Presentation

- Service Catalog
- User Lab Policies
- Troubleshooting
- Useful Links



Alps infrastructure in the CSCS machine room

Source: <https://www.cscs.ch/computers/alps>

Service Catalog

Computing Resources

GPU/hybrid nodes and multicore nodes (14:15 - 14:45 Alps Infrastructure)

Computing time on Cray systems is accounted in **node hours**

- Resources are assigned over **three-months windows**
 - **Quotas reset** on April 1st, July 1st, October 1st and January 1st
- Use thoroughly the quarterly compute budget within the time frame
- Unused resources in the allocation periods **cannot be recovered**

Check your budget in the **current allocation window**

- Overview of resources on <https://account.cscs.ch>

More information: <https://user.cscs.ch/access/accounting>

Storage Resources (14:45 - 15:15 Storage & Data Strategy)

/users and /project

- Access **r+w** from login nodes (UAN), storage for datasets, code and scripts
- Better performance with larger files (archive small files with **tar**)

Users are NOT supposed to run jobs here

- **Reliability over performance:** /project read-only on compute nodes (CN)
- Folders **backed up**: data retention until 3-months after the end of the project

Environment variables for quick access

- **\$HOME** pointing to personal folder (per cluster)
- **\$PROJECT** pointing to group folder

More information: https://user.cscs.ch/storage/file_systems

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 a dedicated Slurm queue ([Internal Transfer](#))

- The service submits jobs on the **datamover** cluster at no charge
- Direct access via ssh to the nodes of this cluster is not allowed

More information: <https://user.cscs.ch/storage/transfer>

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: <https://user.cscs.ch/storage/lts>

Multi-factor authentication

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 authenticators that follow the TOTP open standard
- Certified SSH keys are created with a **validity limited in time**

More information: <https://user.cscs.ch/access/auth/mfa>

Container runtime environment: Sarus

Sarus 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 information: <https://user.cscs.ch/tools/containers/sarus>

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

Users can make HTTP requests to perform the following 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: <https://user.cscs.ch/tools/firecrest>

Interactive computing with JupyterLab

JupyterLab is the web-based user interface for Project Jupyter

- Create and share documents with live code, equations, visualization, ...
- Same notebook document format as the classic Jupyter Notebook
- Ability to work with multiple documents using tabs or splitters side by side

CSCS JupyterLab powered by JupyterHub with a ready-made Python kernel

- You can add your own kernels based on your own virtual environments
- Multi-user hub spawning multiple instances of single-user Jupyter server
- Interactive execution of JupyterLab over single and multiple nodes

More information: <https://user.cscs.ch/tools/interactive/jupyterlab>

Regression testing with ReFrame

ReFrame is a framework for regression tests on HPC systems

- Abstract away the complexity of the interactions with the system
- Separate the logic of a regression test from the low-level details
- Users can write portable regression tests, focusing on functionality

CSCS provides ReFrame pre-configured for its systems:

- ReFrame can be accessed with **module load reframe-cscs-tests**
- The module provides the configuration as well as regression tests
- Users can run the tests already provided by CSCS staff

More information: <https://user.cscs.ch/tools/reframe>

User Environment (11:45 - 12:30 Alps User Environment)

Alps User Environment (uenv mount with compiler, MPI library,...)

- AI/ML: PyTorch, TensorFlow,...
- Materials Science: CP2K, QuantumESPRESSO, VASP,...
- MD Simulations: GROMACS, LAMMPS, NAMD,...
- Visualization: Paraview, Visit, VMD,...

Cray Programming Environment:

- programming environment on Cray systems with performance tools

More information:

<https://confluence.cscs.ch/display/KB/User+Environments+on+Alps>

User Lab Policies

General Policies

The code of conduct outlines proper practices

- **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 production with reference to the “*project ID ###*”

User Regulations define basic guidelines

- **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 Telematics Resources (“BOT”)**

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

Data Retention Policies

Data backup for **active projects**:

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

As soon as a the project **expires**:

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

No backup for data in **scratch**:

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

More information: <https://user.cscs.ch/storage/recovery>

Policies on the /scratch filesystem

Fast workspace for running jobs

- Designed for **performance** rather than reliability
- **Cleaning policy**: files **older than 30 days deleted** daily
- **No backup**: users should transfer data after job completion

Soft quotas to prevent performance degradation

- On **inodes** (files and folders) to avoid large numbers of small files
- On **disk occupancy** with a **grace time** to allow data transfer

More information: https://user.cscs.ch/storage/file_systems/scratch

Fair Usage Policies

Slurm workload manager

- The job scheduler is a shared resource among users submitting jobs
- **Do not submit large numbers of jobs** and commands at the same time
- We will be forced to kill jobs and limit new submissions

Login nodes:

- **Running applications on login nodes is not allowed**
- Submit your simulations with the Slurm scheduler on compute nodes
- **Heavy processes** running on login nodes will be **terminated**

More information: <https://user.cscs.ch/access/accounting/#policies>


Troubleshooting

What to do in case of trouble?


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

CSCS Service Desk

Log in



CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre



Welcome to the CSCS Service Desk! How can we help you ?

Useful links

Knowledge Base

User Portal

Tutorials

Login first!

You are currently not logged in. Please [login here](#) to submit a request and view your current and past requests.

Contact us

Only if you do not have an account or if your account is not accessible, you can [contact us here](#).

How to submit a support request

Useful links to the documentation available online

- Knowledge Base (KB) at <https://confluence.cscs.ch/display/KB>
- User Portal at <https://user.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

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

Please select a request type to open a case



Accounting

Administrative requests regarding your user or project accounts



Cloud

Questions regarding the openstack cluster Castor



Connection

Questions regarding SSH connections and keys using multi-factor authentication (MFA)



Network

Questions regarding SSL certificates, firewalls, servers



Scientific Applications

Questions regarding supported applications and libraries



Software Environment

Questions regarding compiling software, containers, user environment, interactive computing and visualization



Storage and Filesystems

Questions regarding backup, data transfer (globus, xfer), filesystem performance, object storage



System and job scheduling

Questions regarding system related issues with batch jobs and remote workflows (e.g. Unicore)



Other requests

Other requests for support

Example of a 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 access

Summary

Slurm job failed with error "..."

Description

Aa ▾ B I ... ≡ ▾ 🔗 <> + ▾

My username is <user name>, I submitted the job <job ID> on <system>. The job running <code name> exited with state **FAILED**. The job script (<script name>) and input files (<file list>) can be found in \$SCRATCH/failed_job, I have already given read access to with the command **chmod -R +r \$SCRATCH/failed_job**.
The instructions to reproduce the issue are ...

Please include the Slurm JobIDs of the batch jobs and briefly describe your workflow

System

Type to search ▾

Project

csstaff (reporter) x ▾

Attachment (optional)

📎 Drag and drop files, paste screenshots, or
browse

Monitoring your requests

Cases on 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

My cases

Open requests Created by anyone

Type Reference Created Summary

Issue details > [dom] Error using CDT 22.09 with PrgEnv-nvidia (SD-57760)

Comment on this request...

ACTIVITY

Luca Marsella (CSCS) 03/Mar/23 11:41 AM **LATEST**
Thanks [Vincenzo Annaloro \(CSCS\)](#)!

Vincenzo Annaloro (CSCS) 03/Mar/23 11:37 AM
Hi Luca,
I'm investigating the problem on DOM about the missing cdt/22.09 on ELOGIN and CNs.
Cheers,
Vincenzo

Your request status changed to: **In Progress** 03/Mar/23 9:59 AM

DETAILS

Description
Hi,
I have re-triggered the regression tests of the non default Cray PE 22.09 loading the module cdt/22.09 after the last intervention on Dom, as I have described in [VCMSA-88](#).

IN PROGRESS

Don't notify me
Share
Resolve
Cancel Request

SHARED WITH

Luca Marsella (CSCS)
Creator

Useful Links

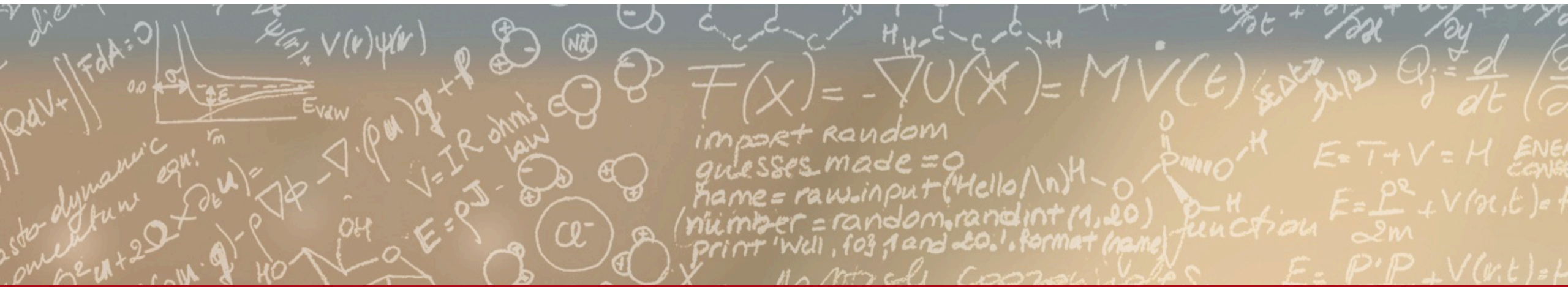
- User Portal
 - <http://user.cscs.ch>
- Knowledge Base
 - <https://confluence.cscs.ch/display/KB/CSCS+Knowledge+Base>
- CSCS Service Desk
 - <https://support.cscs.ch>



CSCS

Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



Thank you for your kind attention