DCSR Clusters and storage

The DCSR clusters and storage facilities in more detail

Wally

Wally is a general purpose cluster for both high throughput and parallel jobs.

It is composed of 96 nodes each with 16 cores (2 x 8 core Intel SandyBridge) and 64 GB of memory giving a ratio of 4GB per core.

The nodes are connected via QDR Infiniband in non blocking groups of 32 and also have a gigabit ethernet connection for administration and interactive use.

Wally has 384 TB of high performance scratch space available at /scratch/wally - please remember that this space is not backed up and old files are purged on a regular basis.

Rack F14

Rack F13

Axiom (obsolete)

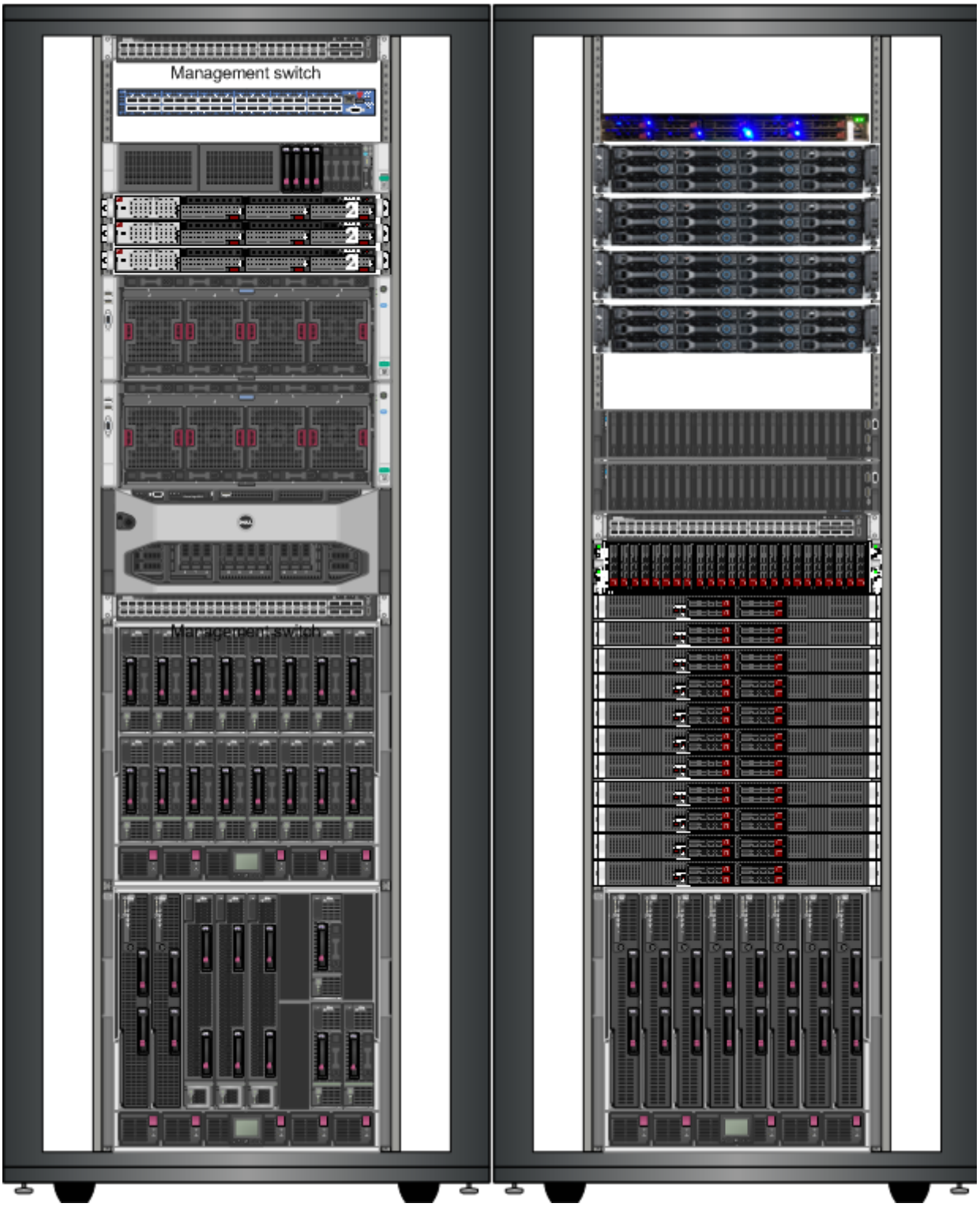
Axiom is a special purpose cluster composed of a number of different types of machine with some notable machine types given in the following table.

In particular there are a number of nodes with large amounts of memory.

Cores	Memory	Processor type	Number of machines
16	32 GB	Intel SandyBridge	10
24	64 GB	Intel Haswell	4
24	64 GB	Intel Xeon Gold	5
40	512 GB	Intel Xeon Gold	2
32	512 GB	AMD Opteron	10
64	756 GB	Intel Haswell	1
60	1 TB	Intel IvyBridge	1
80	1 TB	Intel Broadwell	2
80	1.5 TB	Intel Xeon Gold	1

Rack F17

Rack F16



Jura

Jura is a cluster for the analysis of sensitive data and is primarily used by the CHUV.

Computing ressources

- 10 compute nodes
  - cpt01: CPUs=40 Boards=1 SocketsPerBoard=4 CoresPerSocket=10 ThreadsPerCore=1 RealMemory=515712
  - cpt02: CPUs=32 Boards=1 SocketsPerBoard=4 CoresPerSocket=8 ThreadsPerCore=1 RealMemory=257754
  - cpt[03-04]: CPUs=48 Boards=1 SocketsPerBoard=2 CoresPerSocket=12 ThreadsPerCore=2 RealMemory=257680
  - cpt[05-06]: CPUs=48 Boards=1 SocketsPerBoard=2 CoresPerSocket=12 ThreadsPerCore=2 RealMemory=64156
  - cpt[07-08]: CPUs=160 Boards=1 SocketsPerBoard=4 CoresPerSocket=20 ThreadsPerCore=2 RealMemory=1031536
  - cpt09: NodeName=cpt09 CPUs=160 Boards=1 SocketsPerBoard=4 CoresPerSocket=20 ThreadsPerCore=2 RealMemory=3095999
  - cpt10: NodeName=cpt10 CPUs=160 Boards=1 SocketsPerBoard=4 CoresPerSocket=20 ThreadsPerCore=2 RealMemory=999282

- 4 nodes with Xeon PHI accelerators
  - cpt[03-04]: 82:00.0 Co-processor: Intel Corporation Xeon Phi coprocessor 31S1 (rev 11)
  - cpt[05-06]: 82:00.0 Co-processor: Intel Corporation Xeon Phi coprocessor 5100 series (rev 11)

- Login node
  - frt: CPUs=48 Boards=1 SocketsPerBoard=2 CoresPerSocket=12 ThreadsPerCore=2 RealMemory=65697804
  - 15 TB local disk space

Storage ressources

- Fast scratch based on SSD
  - /scratch/beegfs 112 TB
  - Not purged

- Data directory
  - /data 160 TB
  - For static datasets (including reference ones (TCGA, ADNI et al)

- Not purged

ATTENTION /data directory is NOT BACKED UP

- Archive with encrypted tapes
  - /archive
  - 600 TB available
  - Data are copied transparently on two tape libraries located in two different datacenters for disaster recovery

Getting ressources on Jura

- For sensitive data only
- Organized by PI
- Use DCRS request form and specify Sensitive or Personal data
- https://conference.unil.ch/research-resource-requests/

Accessing the infrastructure from UNIL

- Any user is expected to take a short training to get familiar with the environment, the do's and dont's
- Once the demand is approved, you will receive a mail with a QR-Code like

- You need an app like Google Authenticator or FreeOTP on your smartphone to scan it
- Google Authenticator:

  https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2&hl=en

  https://apps.apple.com/us/app/google-authenticator/id388497605
  FreeOTP:

https://play.google.com/store/apps/details?id=org.fedorahosted.freeotp&hl=en

https://apps.apple.com/us/app/freeotp-authenticator/id872559395

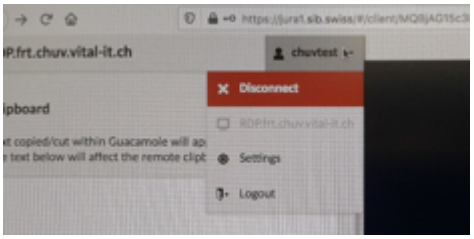
- Go to https://jura.dcsr.unil.ch web site and log in with your **UNIL credentials**
- 

- Enter the code displayed by the application
- 

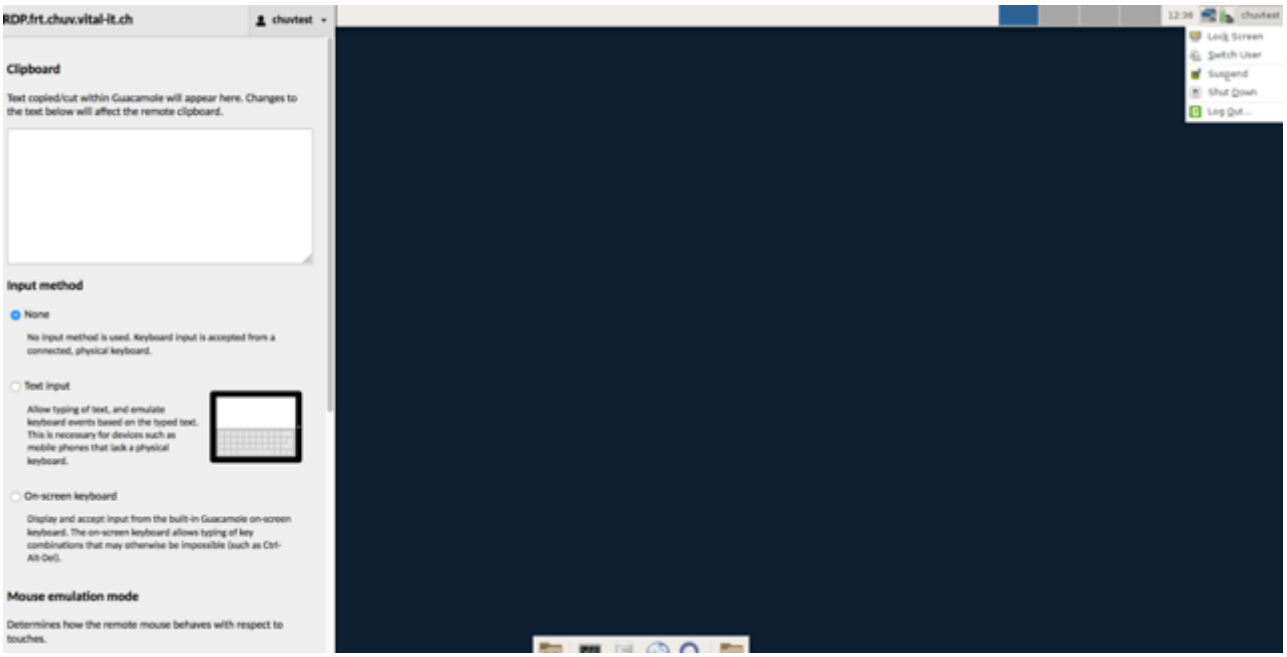
- Congratulations! you are now logged in

ATTENTION PROPER LOG OUT

- CTRL+ALT+SHIFT to display guacamole menu



- Or session logout



Transferring data in

- Transfer your data to the Jump Host

```
sib-1-24:~ someuser$ sftp someuser@jura.dcsr.unil.ch
Password:
Verification code:
Connected to someuser@jura.dcsr.unil.ch.
sftp> dir
data
sftp> cd data
sftp> dir
sftp> put AVeryImportantFile.tgz
Uploading AVeryImportantFile.tgz to /data/AVeryImportantFile.tgz
AVeryImportantFile.tgz
```

- The verification code of the Google Authenticator or FreeOTP is required
- Transfer your data from the Jump Host

  ```
  [someuser@frt ~]$ sftp jura.dcsr.unil.ch
  Password:
  Verification code:
  Connected to jura.dcsr.unil.ch.
  sftp> cd data
  sftp> dir
  AVeryImportantFile.tgz
  sftp> get AVeryImportantFile.tgz
  Fetching /data/AVeryImportantFile.tgz to AVeryImportantFile.tgz
  /data/AVeryImportantFile.tgz
  ```
- To repeatedly transfer large files from reputable external sources a direct access can be granted.
- The verification code of the Google Authenticator or FreeOTP is required but if you have many files to transfer we can set up an automated system

Transferring code in/out

There is a DCSR managed Git service accessible from Jura. More information can be found at

https://wiki.unil.ch/ci/books/service-de-calcul-haute-performance-%28hpc%29/page/why-is-there-a-dcsr-gitlab-service-and-what-is-it

Accessing the infrastructure from CHUV

`ssh<unil-username>@stockage-horus.chuv.ch`

Curnagl

Kesako?

Curnagl (Romanche), or Chocard à bec jaune in French, is a sociable bird known for its acrobatic exploits and is found throughout the alpine region. More information is available at

https://www.vogelwarte.ch/fr/oiseaux/les-oiseaux-de-suisse/chocard-a-bec-jaune

It's also the name of our new compute cluster which will replace the Wally and Axiom clusters. See the end of the page for an overview of the migration process.

Further documentation will be provided in the coming weeks.

As the configuration is being actively worked on some details may change without warning.

For testing and interventions the cluster may be unavailable for certain periods without warning - we expect to pass into "production" running during May at which point the service can be considered stable. In particular a number of network changes are foreseen in order to connect the cluster to the datacenter firewall infrastructure.

If you experience unexpected behaviour or need assistance please contact us via helpdesk@unil.ch starting the mail subject with DCSR Curnagl

How to connect

The login node is **curnagl.dcsr.unil.ch**

Before connecting please add the host's key to your list of known hosts:

echo "curnagl.dcsr.unil.ch ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBCunvgFAN/X/8b1FEIxy8p3u9jgfF0NgCl7CX4ZmqIhaYis2p7AQ34foIXemaw2wT+Pq1V9dCUh18mWXnDsjGrg=" >> ~/.ssh/known_hosts

Please be aware that you must be connected to the VPN if you are not on the campus network.

Then simply `ssh username@curnagl.dcsr.unil.ch` where username is your UNIL account (same as for wally/axiom)

The login node must not be used for any form of compute or memory intensive task apart from software compilation and data transfer. Any such tasks will be killed without warning.

Hardware

Compute

The cluster is composed of 72 compute nodes of which eight have GPUs. All have the same 24 core processor.

Number of nodes	Memory	CPU	GPU
52	512 GB	2 x AMD Epyc2 7402	-
12	1024 GB	2 x AMD Epyc2 7402	-
8	512 GB	2 x AMD Epyc2 7402	2 x NVIDIA A100

Network

The nodes are connected with both HDR Infiniband and 100 Gb Ethernet. The Infiniband is the primary interconnect for storage and inter-node communication.

Storage

The storage is provided by a Lenovo DSS system and the Spectrum Scale (GPFS) parallel filesystem.

/users

Your home space is at /users/username and there is a per user quota of 50 GB and 100,000 files.

Please note that it is up to the user to copy over data from Axiom/Wally to the new system as explained in migration section.

We would like to remind you that all scripts and code should be stored in a Git repository.

Initially there is no backup of this space - this will be put in place over the coming weeks.

/scratch

The scratch filesystem is the primary working space for running calculations.

The scratch space runs on SSD storage and ***has an automatic cleaning policy*** so in case of a shortage of free space files older than 2 weeks (starting with the oldest first) will be deleted.

Initially this cleanup will be triggered if the space is more than 90% used and this limit will be reviewed as we gain experience with the usage patterns.

The space is per user and there are no quotas (*). Your scratch space can be found at /scratch/username

e.g. `/scratch/ulambda`

Use of this space is not charged for as it is now classed as temporary storage.

* *There is a quota of 50% of the total space per user to prevent runaway jobs wreaking havoc*

/work

The work space is for storing data that is being actively worked on as part of a research project. Projects have quotas assigned and while we will not delete data in this space there is no backup so all critical data must also be kept on the DCSR NAS.

The structure is the same as on the Axiom and Wally scratch spaces:

`/ work / FAC / FACULTY / INSTITUTE / PI / PROJECT`

This space can, and should, be used for the installation of any research group specific software tools including python virtual environments.

Partitions

There are 3 main partitions on the cluster:

interactive

The interactive partition allows rapid access to resources but comes with a number of restrictions, the main ones being:

- Only one job per user at a time
- Maximum run time of 8 hours but this decreases if you ask for lots of resources.

For example:

CPU cores requested	Memory requested	GPUs requested	Run Time Allowed

4	32	-	8 hours
8	64	-	4 hours
16	128	1	2 hours
32	256	2	1 hour

We recommend that users access this using the `Sinteractive` command. This partition should also be used for compiling codes.

This partition can also be accessed using the following sbatch directive:

```
#SBATCH -p interactive
```

Note on GPUs in the interactive partition

There is one node with GPUs in the interactive partition and in order to allow multiple users to work at the same time these A100 cards have been partitioned into 2 instances each with 20GB of memory for a total of 4 GPUs.

The maximum time limit for requesting a single GPU is 2 hours.

For longer jobs and to have whole A100 GPUs please submit batch jobs to the gpu partition.

Please do not block resources if you are not using them as this prevents other people from working.

If you request too many resources then you will see the following error:

```
salloc: error: QOSMaxCpuMinutesPerJobLimit
salloc: error: Job submit/allocate failed: Job violates accounting/QOS policy (job submit limit, user's size and/or time limits)
```

Please reduce either the time or the cpu / memory / gpu requested.

cpu

This is the main partition and includes the majority of the compute nodes. Interactive jobs are not permitted. The partition is configured to prevent long running jobs from using all available resources and to allow multi-node jobs to start within a reasonable delay.

The limits are:

Normal jobs - 3 days

Short jobs - 12 hours

Normal jobs are restricted to ~2/3 of the resources which prevents the cluster being blocked by long running jobs.

In exceptional cases wall time extensions may be granted but for this you need to contact us with a justification before submitting your jobs!

The cpu partition is the default partition so there is no need to specify it but if you wish to do so then use the following sbatch directive

```
#SBATCH -p cpu
```

gpu

This contains the GPU equipped nodes.

To request resources in the gpu partition please use the following sbatch directive:

```
#SBATCH -p gpu
```

The limits are:

Normal jobs - 1 day

Short jobs - 6 hours

Normal jobs are restricted to ~2/3 of the resources which prevents the cluster being blocked by long running jobs.

These limits will be reviewed during the initial testing period. To request the number of GPUs per node please use:

```
--gres=gpu:N
```

where N is 1 or 2.

Software

The DCSR software stack is now loaded by default so when you connect you will see the following:

```
$ module avail

 /dcsrsoft/spack/hetre/v1.1/spack/share/spack/lmod/Zen2-IB-test/linux-rhel8-x86_64/Core
   cmake/3.20.0    gcc/9.3.0     mpfr/3.1.6
   cuda/11.2.2     git/2.31.0    xz/5.2.5
```

```
Use "module spider" to find all possible modules and extensions.
Use "module keyword key1 key2 ..." to search for all possible modules matching
any of the "keys".
```

To see more packages load a compiler (gcc) - note that the Intel compilers will be available in the near future

```
$ module load gcc

$ module avail

 /dcsrsoft/spack/hetre/v1.1/spack/share/spack/lmod/Zen2-IB-test/linux-rhel8-x86_64/gcc/9.3.0
   admixtools/7.0.1         gsl/2.6                      python/2.7.18
   bamaddrg/0.1             htslib/1.10.2                python/3.8.8       (D)
   bamtools/2.5.1           intel-tbb/2020.3             qtltools/1.3.1
   bcftools/1.10.2          julia/1.6.0                  r/4.0.4
   bedtools2/2.29.2         maven/3.6.3                  rsem/1.3.1
   blast-plus/2.11.0        miniconda3/4.9.2             star/2.7.6a
   bowtie2/2.4.2            mvapich2/2.3.5               stream/5.10-openmp
   cmake/3.20.0      (D)    nlopt/2.6.1                  stream/5.10        (D)
   eigen/3.3.9              octave/6.2.0                 tskit/0.3.1
   fftw/3.3.9               openblas/0.3.14-openmp       xz/5.2.5           (D)
   gdb/10.1                 openblas/0.3.14        (D)   zlib/1.2.11
   gmsh/4.7.1-openmp        openjdk/11.0.8_10
   gnuplot/5.2.8            perl/5.32.1

- /dcsrsoft/spack/hetre/v1.1/spack/share/spack/lmod/Zen2-IB-test/linux-rhel8-x86_64/Core --
   cmake/3.20.0    cuda/11.2.2    gcc/9.3.0 (L)    git/2.31.0    mpfr/3.1.6    xz/5.2.5

  Where:
   D:  Default Module
   L:  Module is loaded

Use "module spider" to find all possible modules and extensions.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the
"keys".
```

The provided versions for key tools are:

- GCC - 9.3.0

- Python - 3.8.8
- R - 4.0.4
- MPI - mvapich2 2.3.5

Not all tools currently installed on the Wally/Axiom stack are currently available as we are proceeding with updates to new versions but more will be available in the coming weeks.

The old (Vital-IT) /software stack will be made available but is unsupported and there is no guarantee that it will work correctly.

Further information

Migration from Axiom and Wally

The /work space will be accessible from the Axiom and Wally login nodes which can be used to transfer important data from the /scratch/wally and /scratch/axiom spaces to /work

Note: initially this space will only be visible from wally-front1

On Curnagl your home directories on Axiom and Wally will be available (read only) under /oldusers/username - e.g. `/oldusers/ulambda`

It is your responsibility to copy data and please take the opportunity to clean and organise things.

Axiom and Wally decommissioning

At the very latest the systems and storage will remain available until the end of 2021.

Axiom

Job submission to Axiom will be stopped at the end of April and the nodes powered off.

The Axiom scratch space will remain visible on the login nodes until later in the year.

Wally

Job submission to Wally will continue until later in the year but no hardware problems will be resolved.

The new software stack will be made available but without MPI codes as the interconnect does not work correctly with recent MPI implementations.

Storage on the Clusters

There are three main types of storage available and each is intended for a particular type of use. A summary is given in the following table:

	Mount point or access method	**Size and limits**	**Lifetime and Backup**	**Cost**
Home	/users	2.3 TB but 10 Gb 20'000 files quota	Permanent Once a day	-
Scratch	/scratch/axiom /scratch/wally	233 TB 346 TB	Will be purged **NO BACKUP**	95 CHF/TB
Work	/work	1000 TB	3 days snapshots **NO BACKUP**	77.08 CHF/TB
UNIL NAS	/nas from curnagl.dcsr.unil.ch \\nasdcsr.unil.ch from desktop clients	3.4 PB	2 snapshots per day Replication on second site	120.78 CHF/TB for 2 copies

Quarterly maintenance

Maintenance schedule 2020

In order to provide a stable service, regular maintenance periods are required to allow intrusive work on the clusters to be carried out.

There are 4 one day (exceptionally two days) downtimes per year for minor interventions and updates.

When	Notes
Q1 - March 30/31	2 days due to recabling
Q2 - June 29	
Q3 - Oct 26	1 day because of Electrical recabling
Q4 - TBC	

In addition there is an annual downtime week in January to allow for major work and software upgrades.

The next planned maintenance week is in January 2021.

The following sections give an overview of the changes carried out and details of any user visible effects that users of the DCSR clusters should be aware of.

March 2020

Jobs submitted before the maintenance period

Please note that the scheduler will not start jobs that are expected to finish after 7am on the 30th of March. This means that any long jobs submitted in the run up will be held in the queue even if there are free nodes so please take care to specify the shortest wall time possible.

For example, on Tuesday the 24th of March, if there are free nodes, a 5 day job will run but a 7 day job will remain waiting in the queue (state PD).

Whilst we will make every effort to maintain the state of the queue we cannot guarantee that your pending jobs will still be present after the maintenance.

Please be aware that after the maintenance not all nodes will be immediately available. The remainder will be brought online in the days following the downtime.

User Visible Changes

New Partition Structure

In order to simplify the management of the clusters the partition structure will be changed. The new partitions are:

- debug - 4 nodes in Wally to allow for quick tests with one job per user at any time
- wally - all nodes in the Wally sub-cluster
- axiom - all nodes in Axiom sub-cluster

This means that there are no longer partitions by wall time and all limits are imposed automatically by a job submit plugin and appropriate Quality of Service (QoS) policies.

The maximum run time remains 10 days. In order to request an allocation on Axiom that lasts for one week the required directives are:

```
#SBATCH --time 7-0
#SBATCH --partition axiom
```

HyperThreading turned off

HyperThreading is a CPU feature that allows two threads to share one execution core and can improve throughput in a number of typical enterprise computing scenarios. For HPC codes it generally degrades performance and makes it difficult to correctly and safely share nodes as well as to run multi-node MPI tasks. For this reason it will be disabled on all Axiom nodes and is already turned off for Wally.

The core count on Axiom will be reduced by 50% after this change so nodes that previously reported 64 cores will now report 32 and so on. Job scripts may need to be updated to reflect this change.

Default wall time of 15 minutes

The default run time for all jobs will be set to 15 minutes - this means that if you have not requested longer via an SBATCH directive then your job will be terminated after 15 minutes.

Topology aware scheduling

On Wally it will be possible to specify that you want all allocated nodes to be on the same Infiniband leaf switch. Whilst this may improve performance for communication intensive tasks, it can also lead to a significant increase in queuing time. It is also possible to specify how long you are prepared to wait for.

For example: requesting 8 nodes on the same switch and being prepared to wait 12 hours for this.

```
#SBATCH --nodes 8
#SBATCH --switches 1@12:00:00
```

Other Changes

SLURM 19.05.x and configuration changes

Update to SLURM 19.05.x and diverse changes to the SLURM configuration to improve performance and usability.

Infiniband network rebalancing

In order to increase robustness an extra IB switch will be added to both fabrics (Wally and Axiom) and certain nodes moved to the new leaf switch.

OS update to RedHat 7.7

General security and functionality updates. No user visible changes are expected.

Storage updates

Updates and maintenance on the BeeGFS /scratch file systems including new version and rebalancing.

June 2020

Jobs submitted before the maintenance period

Please note that the scheduler will not start jobs that are expected to finish after 8am on the 29th of June. This means that any long jobs submitted in the run up will be held in the queue even if there are free nodes so please take care to specify the shortest wall time possible.

For example, on Tuesday the 23rd of June, if there are free nodes, a 5 day job will run but a 7 day job will remain waiting in the queue (state PD).

Whilst we will make every effort to maintain the state of the queue we cannot guarantee that your pending jobs will still be present after the maintenance.

Please be aware that after the maintenance not all nodes will be immediately available. The remainder will be brought online in the days following the downtime.

User Visible Changes

R with multithreaded BLAS

R will be able to take advantage of multiple CPU cores by using multi-threaded linear algebra libraries (OpenBLAS or MKL).

In order to set the level of parallelism you can use the OMP_NUM_THREADS environment variable

Updated software stack

The new software environment will receive a minor update - the same applications will be available but sometimes with minor version changes. For example the version of Python is now 3.7.7 and R has moved to 3.6.3

In the case of problem please let us know! The previous stack is still available via the following command:

```
source /dcsrsoft/spack/bin/setup_old_dcsrsoft
```

Other Changes

Infiniband recabling

The recabling of the Infiniband networks will be completed

Ethernet recabling and reconfiguration

Improvements to the 10 Gb/s network

BeeGFS updates

Update to 7.1.5 and hopefully fewer bugs!

Oct 2020

Electrical recabling

The recabling of the Axiom and Wally racks has to be performed because of electrical security regulations

Q4 2020