User Guide

Jakub Kropáček

September 2021

VSB TECHNICAL UNIVERSITY OF OSTRAVA | IT4INNOVATIONS NATIONAL SUPERCOMPUTING CENTER

Documentation

docs.it4i.cz

IT4Innovations Documentation | Search | sccs/docs.it4i.cz

General | Storage | Clusters | Software

General

- Home
- Obtaining Login Credentials
- Accessing the Clusters
- Applying for Resources
- Certificates FAQ
- Resource Allocation and Job Execution
- Connect to the Clusters
- Satisfaction and Feedback
- PRACE User Support
- API Documentation
- Withdrawal from service
- PROJECT Storage Availability

Documentation

Welcome to the IT4Innovations documentation. The IT4Innovations National Supercomputing Center operates the Barbora and Salomon supercomputers. The supercomputers are available to the academic community within the Czech Republic and Europe, and the industrial community worldwide. The purpose of these pages is to provide comprehensive documentation of the hardware, software, and usage of the computers.

How to Read the Documentation

1. Select the subject of interest from the left column or use the Search tool in the upper right corner.
2. Scan for all the notes and reminders on the page.
3. If more information is needed, read the details and **look for examples** illustrating the concepts.

Getting Help and Support

Note

Contact support[at]it4i.cz for help and support regarding the cluster technology at IT4Innovations. For communication, use the **Czech**, **Slovak**, or **English** language. Follow the status of your request to IT4Innovations here. The IT4Innovations support team will use best efforts to resolve requests within thirty days.

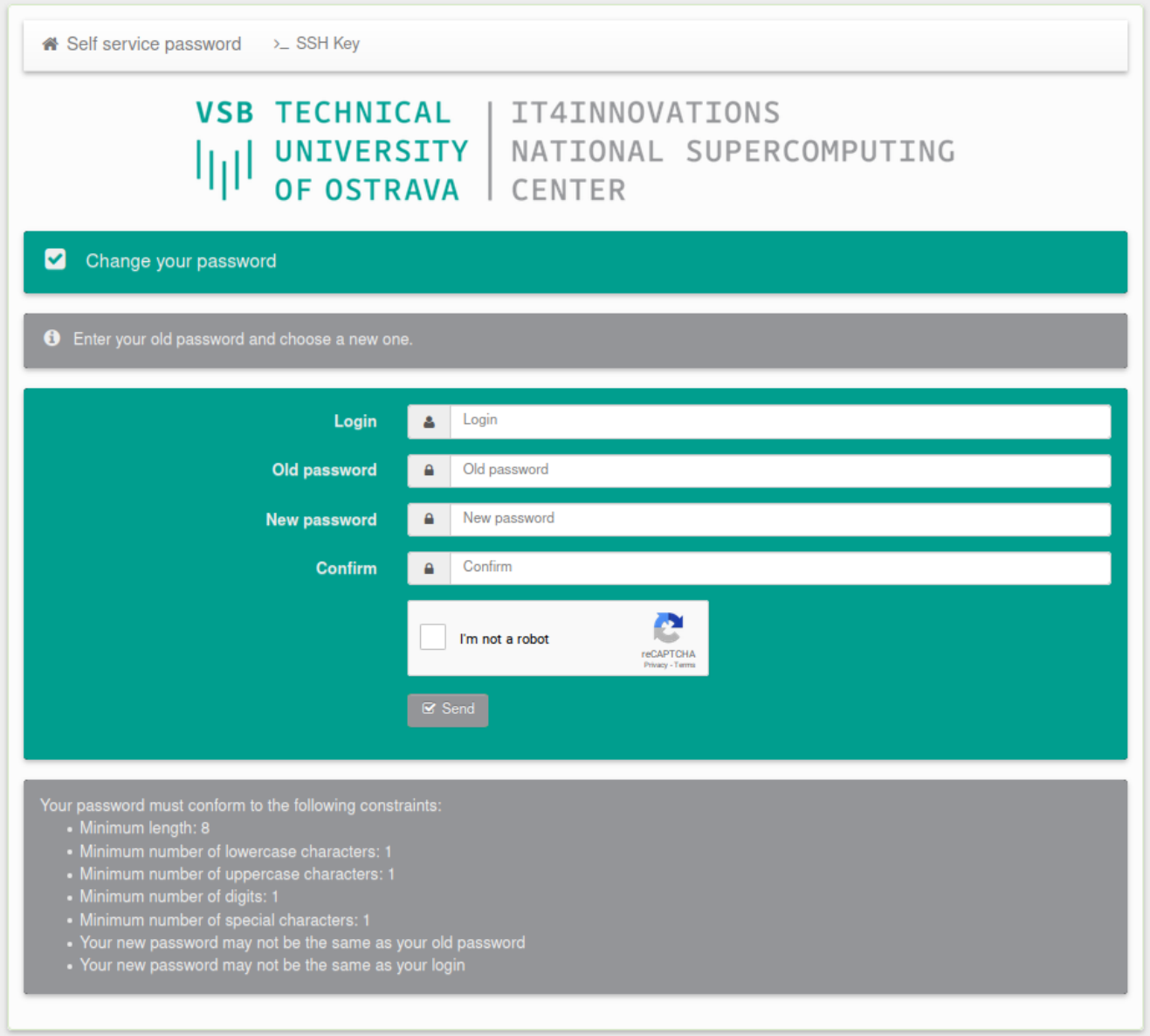
Table of contents

Actions

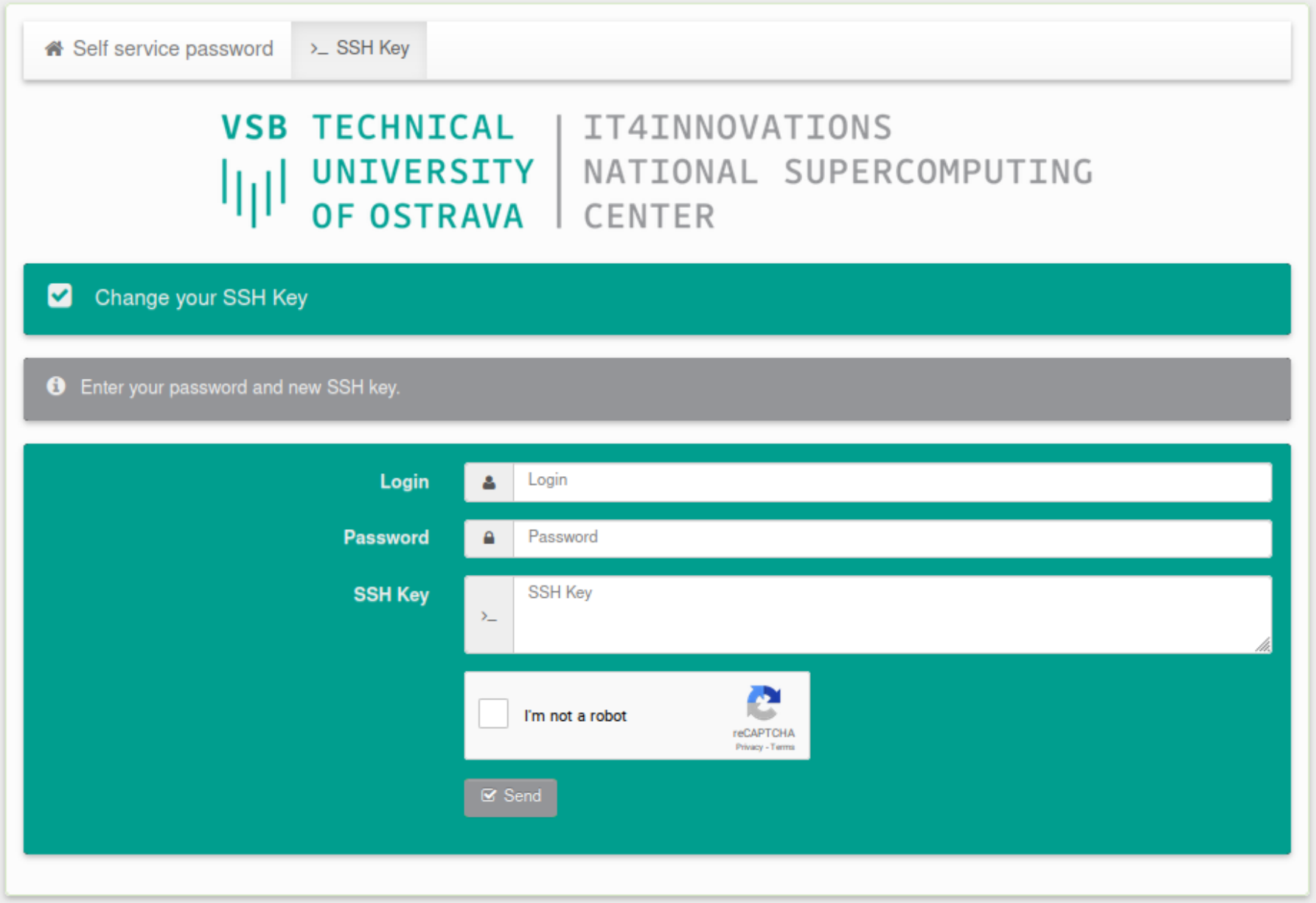
- Edit This Page
- Request Change
- Get Support

Changing Login Credentials

extranet.it4i.cz/ssp



Changing Login Credentials



scs.it4i.cz

IT4Innovations Information System

Self-service portal to manage your HPC resources

IT4Innovations offers HPC resources which are provided on project basis. You can apply for the resources and manage them here.

By signing up for and by signing in to this service you accept our:

- Acceptable Use Policy
- User´s duties.

Single Sign-On | External users

Sign-in with an IT4Innovations account, a Federation account or a public service account.

Sign in with **It4innovations** account

Sign in with **eduID.cz**

Sign in with **eduGAIN**

Sign-in with a social net if attached to your account.

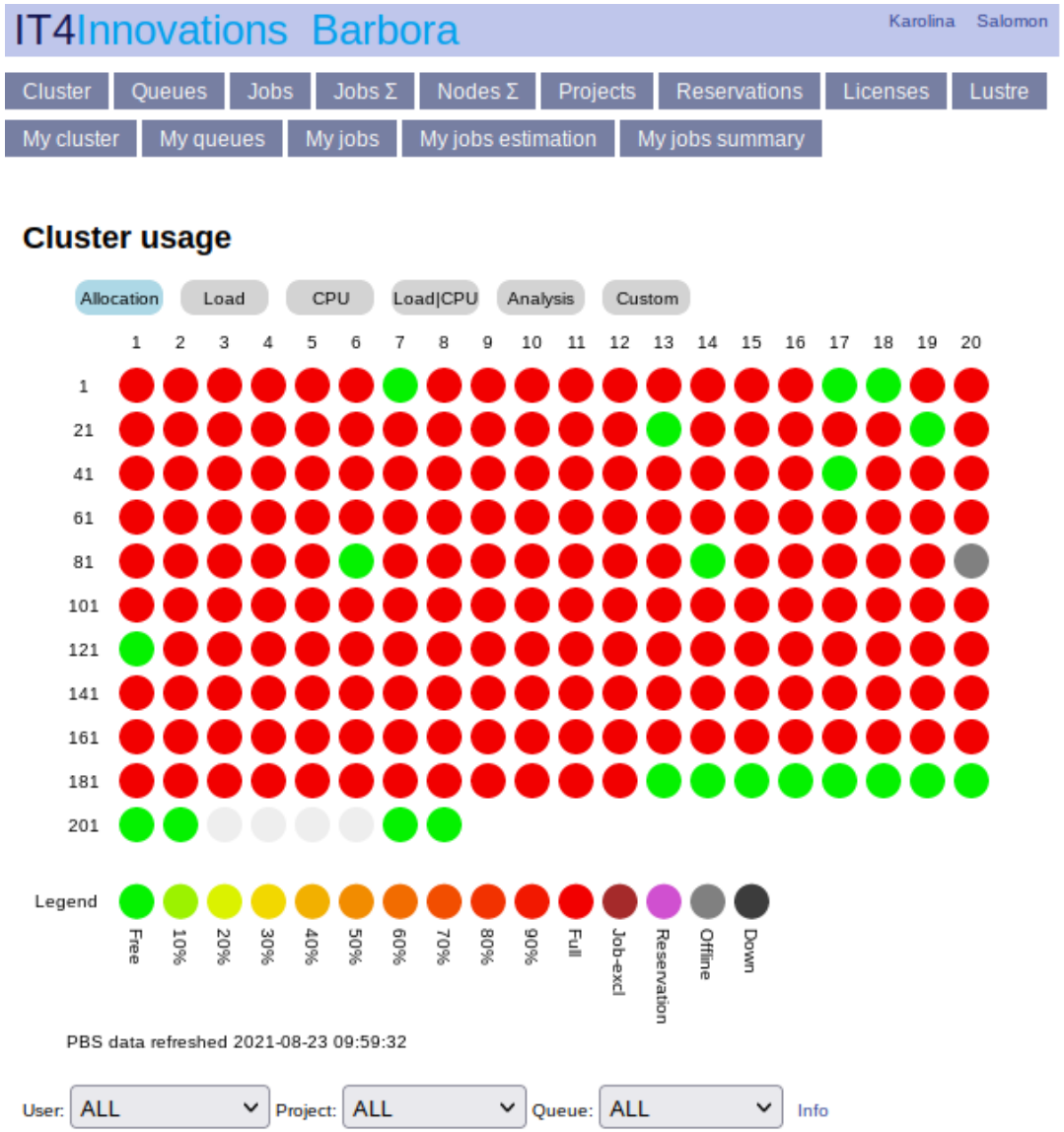
Sign in with **GitHub**

Sign in with **LinkedIn**

Sign in with **Twitter**

Dashboards

it4i.cz/en/for-users/cluster-utilization



First Login

- 2 encrypted emails - login credentials and ssh key
- login credentials used for accessing information systems
- ssh key to access clusters

Linux

- see docs.it4i.cz/general/shell-and-data-access/
- Windows users - install Windows Linux Subsystem https://docs.microsoft.com/en-us/windows/wsl/install
- to connect to a cluster, use
  - `ssh -i path/to/identity_file username@clustername.it4i.cz`
  - `ssh -i path/to/identity_file username@loginX.clustername.it4i.cz`
- local configuration folder `/home/$USER/.ssh/`
- `/home/$USER/.ssh/config` - create your ssh aliases
- `man 5 ssh_config`

First Login

```
$ cat ~/.ssh/config
# contents of $HOME/.ssh/config

Match User username
  IdentityFile ~/.ssh/id_rsa

Host bar1 bar2
  User username

Host bar1
  HostName login1.barbora.it4i.cz
Host bar2
  HostName login2.barbora.it4i.cz
...

$ ssh bar1
                  ____              _
                 |  _ \            | |
                 | |_) | __ _ _ __| |__   ___  _ __ __ _
                 |  _ < / _‘ | ’__| ’_ \ / _ \| ’__/ _‘ |
                 | |_) | (_| | |  | |_) | (_) | | | (_| |
                 |____/ \__,_|_|  |_.__/ \___/|_|  \__,_|


                  ...running on Red Hat Enterprise Linux 7.x

[username@login1.barbora ~]$
```

Computational Environment

- supported shell environments - bash, tcsh, zsh, ksh, dash
  - DGX-2 - bash only
- environment configuration - .bashrc

```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi

# users compilation path
export MODULEPATH=${MODULEPATH}:/home/$USER/.local/easybuild/modules/all

# User specific aliases and functions
alias q="qstat -u $USER"
```

- grep, gawk, sed, and many more programs available
- do not load modules in your profile

Computational Environment

- module system which allows dynamically changing users' environment
- management of different software versions
- hierarchical naming system
  `software_name/software_version-toolchain-toolchain_version-suffix`

Basic Commands

- `ml`, `module list` - lists all loaded modules
- `ml av`, `module avail` - lists available modules
- `ml spider`, `module spider` - lists module information
- `ml del`, `module unload` - unloads selected module
- `ml purge` - unloads all but sticky modules (`--force` flag required)

Available Software

- modules bundled into communities
- `ml software_name/software_version-toolchain-toolchain_version-suffix`
- software dependencies are loaded automatically
- i.e. VASP/6.2.1-intel-2020b-mkl=sequential also loads OpenMPI, MKL, and other

Software Communities

- `bio` - Bioinformatics, biology and biomedical (BLAST, FastQC, GROMACS, ...)
- `chem` - Chemistry, computational chemistry and quantum chemistry (ABINIT, ORCA, TURBOMOLE, QuantumESPRESSO, ...)
- `compiler` - Compilers (GCC, GCCcore, NVHPC, intel, ...)
- `data` - Data management and processing tools (FLAC, HDF5, netCDF, h5py, ...)
- `lang` - Languages and programming aids (Java, Perl, Python, ...)
- `phys` - Physics and physical systems simulations (COMSOL, Gaussian, VASP, ...)
- `tools` - General purpose tools (binutils, cURL, EasyBuild, gzip, ...)
- ...

Computational Environment

PBS

- Portable Batch System = job scheduler
  - documentation at https://www.altair.com/pbs-works-documentation/
- check PBS version: `qmgr --version`
- CPU-time accounting at the end of the job

Job Submission

- at most 100 jobs in the queue, all of them can be job arrays
- at most $1,500$ jobs in a job array
- at most $4,000$ jobs and subjobs in the queue in total
- resource reservations - accounted even when not utilized!

Computational Environment

Basic Commands

- `qsub` - submit a job
- `qstat` - batch client to request status of one or more jobs
- `qdel` - delete job from the queue
  - `qdel -W force` - if everything fails
- every command has it's own `man` page

```
$ qsub -A OPEN-00-00 -N job_submission_example -q qexp -l select=2 -l walltime=00:01:00
      job_script.sh
123456.isrv1
$ qstat -u $USER

isrv1:
                                                                  Req'd  Req'd   Elap
Job ID                  Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
123456.isrv1            username qexp     job_submis    --    2  72    --  00:01 Q   --

$ qdel 123456
$
```

Job Scripts

- submit job parameters inside the job script using the #PBS comment

```
$ cat jobscript.sh
#!/bin/bash
#PBS -A OPEN-00-00
#PBS -N jobname
#PBS -q qexp
#PBS -l select=2:ncpus=36:mpiprocs=36
#PBS -l walltime=01:00:00
#PBS -S /bin/bash

<your job goes here>
$ qsub jobscript.sh
123456.isrv1
$ qstat -u $USER

isrv1:
                                                                Req'd  Req'd   Elap
Job ID                  Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
-------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
474395.isrv1            username qexp     jobname       --    2  72    --  01:00 Q   --
$
```

Computational Environment

Advanced Options

- `qhold` - changes job's status to hold (`H`)
- `qrls` - releases the job back into the queue (status `Q`)
- `qalter` - alter some of the job's settings

```
$ qsub job_script.sh
123456.isrv1
$ qhold 123456
$ qstat -u $USER

isrv1:
                                                                  Req'd  Req'd   Elap
Job ID                  Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
-------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
123456.isrv1            username qexp     jobname       --    2  72    --  00:45 H   --
$ qalter  -l walltime=00:30:00 123456
$ qrls 123456
$ qstat -u $USER

isrv1:
                                                                  Req'd  Req'd   Elap
Job ID                  Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
-------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
123456.isrv1            username qexp     jobname       --    2  72    --  00:30 Q   --
```

Dependency Job Submission

- a way to chain jobs that depend on each other using qsub parameters
  - `after, afterok, afternotok, afterany`
    - current job can be run after selected one(s)
  - `before, beforeok, beforenotok, beforeany`
    - selected job(s) can be run before current one

```
$ cat chain.sh
#!/bin/bash
first=$(qsub job1.sh) && echo $first
second=$(qsub -W depend=afterok:$first job2.sh) && echo $second
$ ./chain.sh
123456.isrv1
123457.isrv1
$ qstat -u $USER

isrv1:
                                                            Req'd  Req'd   Elap
Job ID                  Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
123456.isrv1            username qexp     job1          --    1  36    --  01:00 Q   --
123457.isrv1            username qexp     job2          --    1  36    --  01:00 H   --
```

Computational Environment

'Looped' jobs

How to run jobs that are too long for the selected queue? There is only one requirement: software must allow saving of progress and restarting of the job.

Example

Your molecular dynamics simulation produces **4 nanoseconds** of trajectory **in 24 hours**, and you would like to calculate **12 nanoseconds in total**, which means you need 72 hours. A little bit more, in fact, lest your job gets killed for running up to the wall time. If you were to use `qlong`, you would probably have to wait for a long time before the PBS frees enough nodes for you to utilize. You can, however, use **several** `qprod` **jobs with reduced wall time** in order to utilize **backfilling**. Let's say your wall time will be 24 hours, you would therefore need to run 4 jobs in a sequence.

Example taken from https://www.nics.tennessee.edu/computing-resources/running-jobs/job-chaining

Computational Environment

```
$ cat loop.sh
#!/bin/bash
one=$(qsub job_script.sh) && echo $one
for id in $(seq 2 4); do
        two=$(qsub -W depend=afterok:$one restart.sh) && echo $two
        one=$two
done
$ ./loop.sh
123456.isrv1
123457.isrv1
123458.isrv1
123459.isrv1
$ qstat -u $USER
Job ID                  Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
123456.isrv1            username qprod    job_script    --    1  36    --  24:00 Q   --
123457.isrv1            username qprod    restart       --    1  36    --  24:00 H   --
123458.isrv1            username qprod    restart       --    1  36    --  24:00 H   --
123459.isrv1            username qprod    restart       --    1  36    --  24:00 H   --
```

Example taken from https://www.nics.tennessee.edu/computing-resources/running-jobs/job-chaining

Job Queue Policy

- resources allocated in a fair-share fashion based on available project resources
- each user can consume roughly the same amount of resources per week
- accessing via the job queues - job scheduler

Queues

- qexp, the Express queue: running and testing of small jobs, max walltime of 1h
  - maximum allocation of 4 nodes, 1 running job and 5 queued at most
  - does not consume resources, project specification not required
- qprod, the Production queue: normal production runs, max walltime of 48h
- qlong, the Long queue: long production runs, max walltime of $3 \cdot \text{qprod} = 144\text{h}$
- qfree, the Free resource queue: utilization of free resources, max walltime of 12h
- qnvidia, qfat, the Dedicated queues: access to Nvidia accelerated and Fat nodes

Job Execution Priority

- scheduler decides the job priority based on (in order of importance):
  1. queue priority
  2. fair-share priority
  3. eligible time

Queue Priority

- priority of the queue in which the job is waiting prior to execution
- most influential part of the execution priority: jobs with higher queue priority always have higher execution priority than jobs in lower priority queue

queue	priority
qexp	150
qprod	0
qlong	0
qnvidia	200
qfat	200
qfree	-1024

Job Execution Priority

Fair-Share Priority

- ranking of jobs with equal queue priority based on recent resource usage (decay half-time of 168 hours)
- calculated per project, fair-share priority shared by all members of the project
- `Resource_List.fairshare` attribute of a job

$$\texttt{fairshare\_priority} = \texttt{MAX\_FAIRSHARE} \cdot \left(1 - \frac{\texttt{usage}_{\texttt{project}}}{\texttt{usage}_{\texttt{total}}}\right)$$

$$\texttt{MAX\_FAIRSHARE} = 1,000,000$$

Eligible Time

- time accrued while the job is waiting for resources
- sorting of jobs with equal queue and fair-share priority, lowest priority of allocated
- `eligible_time` job attribute

Job Execution Priority

Job Sort Formula

$$\texttt{job\_execution\_priority} = 1,000 \cdot \texttt{queue\_priority} + \frac{\texttt{fairshare\_priority}}{1,000} + \\ + \frac{\texttt{eligible\_time}}{864,000}$$

- Job Backfilling
  - PBS optimizes the cluster usage to minimize idle time
  - smaller jobs with lower execution priority may overtake larger jobs with higher priority
  - specify the `walltime` attribute to take advantage of this

Example

High priority job with a `walltime = 20:00:00` requires 14 nodes to run. 10 nodes are already free to be utilized, however, the remaining 4 will be available in 6 hours. This means that the other 10 nodes may be utilized by lower priority jobs, granted their `walltime` does not exceed 6 hours.

Resource Accounting Policy

- CPU time defined in terms of wall-clock core-hours (WCH) and normalized core-hours (NCH)
- 1 WCH = 1 core allocated for 1 hour of wall-clock time
- normalized core-hours relate different systems to Salomon

$$\text{NCH} = F \cdot \text{WCH}$$

System	F
Salomon	1.00
Barbora	1.40
Barbora GPU	4.50
DGX-2	5.50
Karolina	1.00
Karolina GPU	1.00

Example

Allocating three non-accelerated nodes on Barbora (36 cores) for 2 hours results in $1.4 \cdot 3 \cdot 36 \cdot 2 = 302.4$ NCH to be charged.

Storage

- 3 main shared file systems:
  - HOME - NFS filesystem
  - SCRATCH - NFS filesystem
  - PROJECT - parallel Lustre filesystem
- accesible via the Infiniband network

HOME filesystem

- meant for preparation, evaluation, processing, and storage of data generated by active projects
- mounted at `/home`, user directories reside at `/home/login`
- default quota of 25 GB, entries limit of $500,000$
- system is backed up (not meant for versioning!), files are kept until the end of the user's lifecycle (user is not associated with any active project for a period of one year)

Storage

SCRATCH filesystem

- intended for temporary scratch data generated during the calculation, and high-performance I/O
- mounted at `/scratch` with default quota of 10 TB, 10,000,000 entries per user
- files that are not accessed for more than 90 days are automatically deleted!

PROJECT filesystem

- central storage for projects' / users' data
- mounted at `/mnt`, project folders available at `/mnt/proj[1-3]`
  - `$ it4i-get-project-dir OPEN-00-00`
    `/mnt/proj3/open-00-00`
- data shared between all clusters, space quota of 20 TB, inodes limit of 5,000,000

IT4I Portal Clients

- accessing SCS information directly from cluster
- use `-h, --help` to list possible options

Available Commands

- `it4icheckaccess` - shows whether you have acccess to a specific queue
- `it4ifree` - shows basic PBS accounting information
- `it4ifsusage` - shows file system usage
  - `it4iuserfsusage`
  - `it4iprojectfsusage`
- `it4imotd` - shows the Message of the Day

IT4I Portal Clients

```
[opr0019@login1.barbora ~]$ it4ifree

Projects I am participating in
==============================
PID          Type    Days left       Total    Used NCHs    My NCHs     Free
----------   ------  -----------  --------  -----------  ---------  -------
OPEN-15-43   S       inactive      1443000      1548330      25623        0
OPEN-15-57   S       inactive     10106000     11024219        373        0
OPEN-18-33   M-1     expired       4297000      4908330    2372709        0
OPEN-18-33   M-2     31            3705000      3793202    2372709        0
OPEN-18-33   M-3     upcoming      2681000            0    2372709  2681000
OPEN-18-33   M-4     upcoming            1            0    2372709        1
OPEN-20-38   S       5              135000       136427     115565        0

Projects I am Primarily Investigating
=====================================
PID          Type    Login        Used NCHs
----------   ------  ---------    -----------
OPEN-20-38   S       opr0019         115565
                     retracted        20862
```