HPC – short introduction

Alexander Vapirev

ICTS, Leuven

14 November 2017

Overview

- What is the VSC?
- VSC infrastructure
- Software environment
- How to get started
- Training and events

Vlaams Supercomputer Centrum (VSC)

- VSC is a virtual organization, founded in 2007
  - Goals
    - Provide infrastructure for high performance computing
    - Provide support for high performance computing
  - Managed by the Research Foundation - Flanders (FWO) in partnership with the five Flemish university associations.
    - Universiteit Antwerpen
    - Vrije Universiteit Brussel
    - Universiteit Gent
    - KU Leuven
    - Universiteit Hasselt

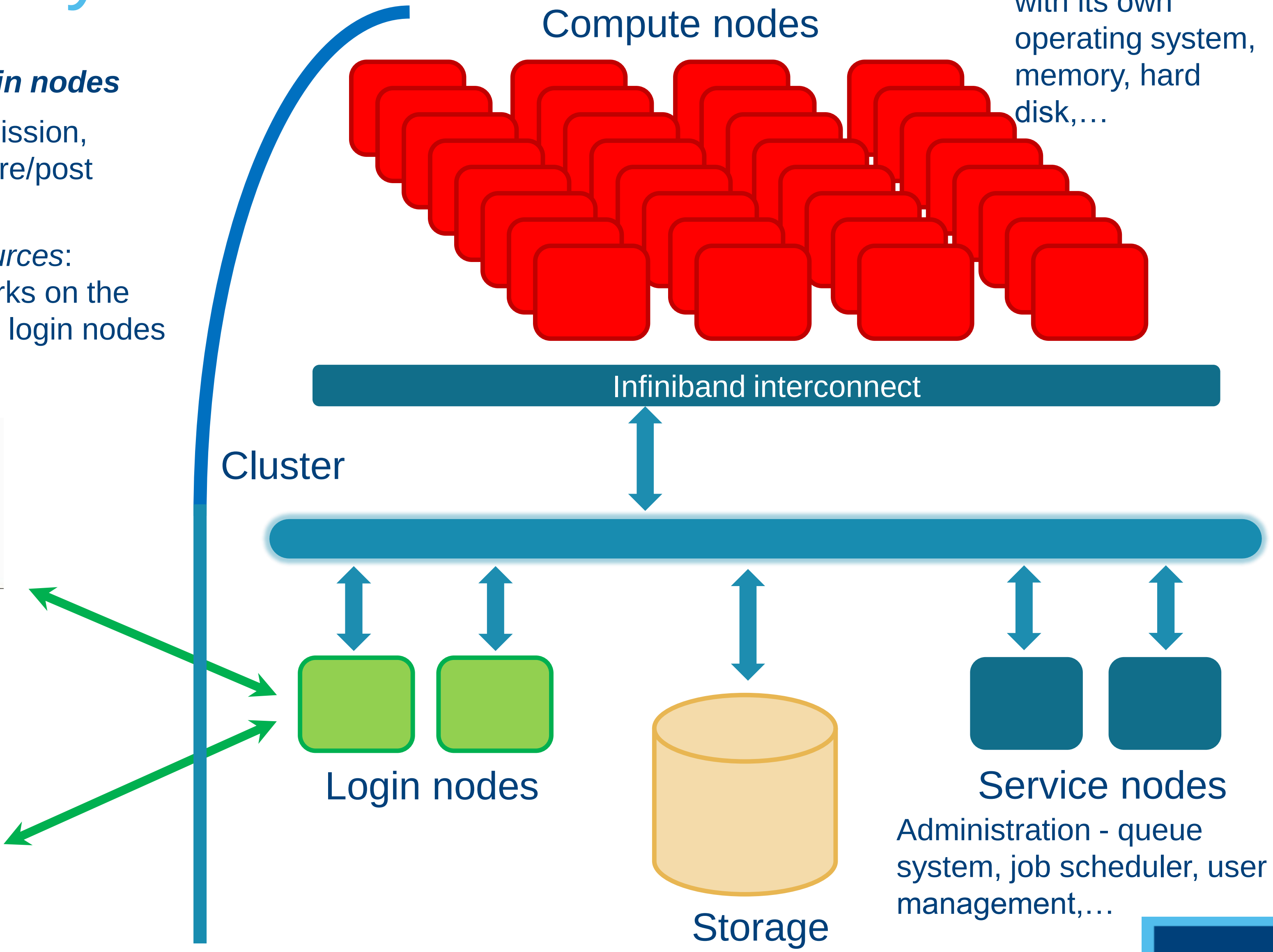
VSC services

- Helpdesk
- Consultancy
- Training and courses
- Information dissemination
- Networking opportunities
- User input & feedback

Bird's eye view on a cluster

Accessed via ***login nodes***

- For job submission, debugging, pre/post processing
- *Shared resources*: everyone works on the same (set of) login nodes


Researchers
Compute nodes
Many independent computers, each with its own operating system, memory, hard disk,…
Infiniband interconnect
Cluster
Login nodes
Storage
Service nodes
Administration - queue system, job scheduler, user management,…

What a cluster is ***not*** …

- An all-powerful and super smart computer that will ***automatically:***
  - Run your application much faster
  - Run your application in parallel

For that you will need to do some extra work

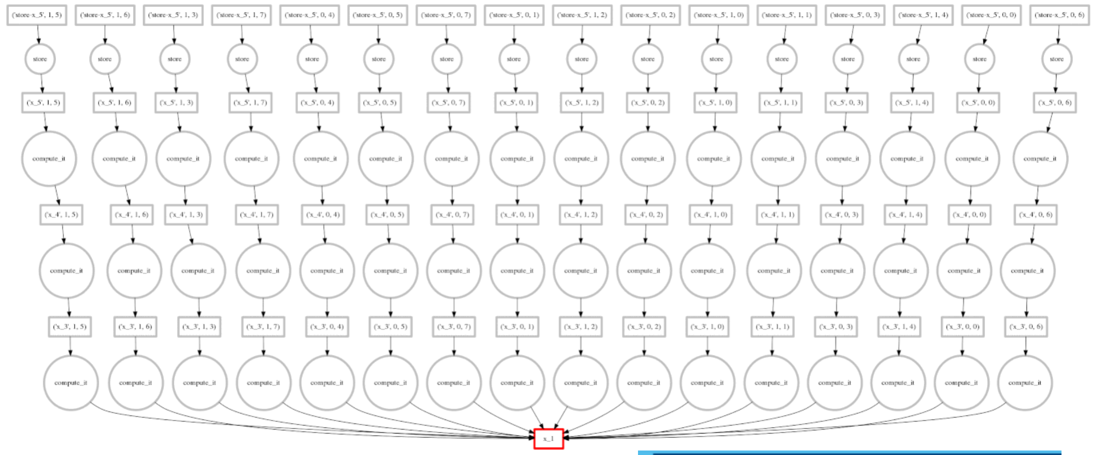
Parallel Computing

- **Serial:**
  - one program, on one core
- **’Embarrassingly parallel’ problems:**
  - lots of runs of one program, with different parameters
  - Hadoop, Spark
- **Problems that require ’real’ parallel algorithms**
  - OpenMP
  - MPI : Message Passing Interface

Parallel Computing

- **Serial:**
  - one program, on one core
- **’Embarrassingly parallel’ problems:**
  - lots of runs of one program, with different parameters
  - Hadoop, Spark
- **Problems that require ’real’ parallel algorithms**
  - OpenMP
  - MPI : Message Passing Interface

Python
R
Java

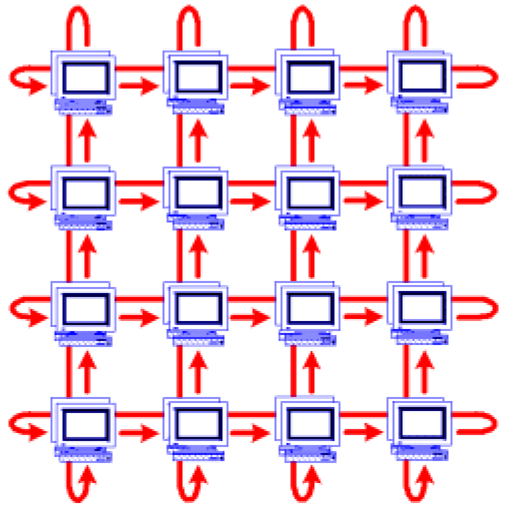


Credit: http://dask.pydata.org/en/latest/shared.html

Parallel Computing

- **Serial:**
  - one program, on one core
- **’Embarrassingly parallel’ problems:**
  - lots of runs of one program, with different parameters
  - Hadoop, Spark
- **Problems that require ’real’ parallel algorithms**
  - OpenMP
  - MPI : Message Passing Interface

C/C++
Fortran



Credit: https://decisionstats.com/2010/09/24/parallel-programming-using-r-in-windows/

Thinking and Cerebro

	ThinKing
Total nodes	144
Processor type	Ivy bridge (Xeon E5-2680v2)
Base Clock Frequency	2.8 GHz
Cores per node	20
Total cores	2,880
Memory per node (GB)	112x64/32x128
Memory per core (GB)	3.2/6.4
Peak performance (TF)	64.5
Network	Infiniband QDR 2:1
Cache (L1 KB/L2 KB/LLC MB)	10x(32i+32d)/10x256/25
Total LLC cache per core	2.5 MB

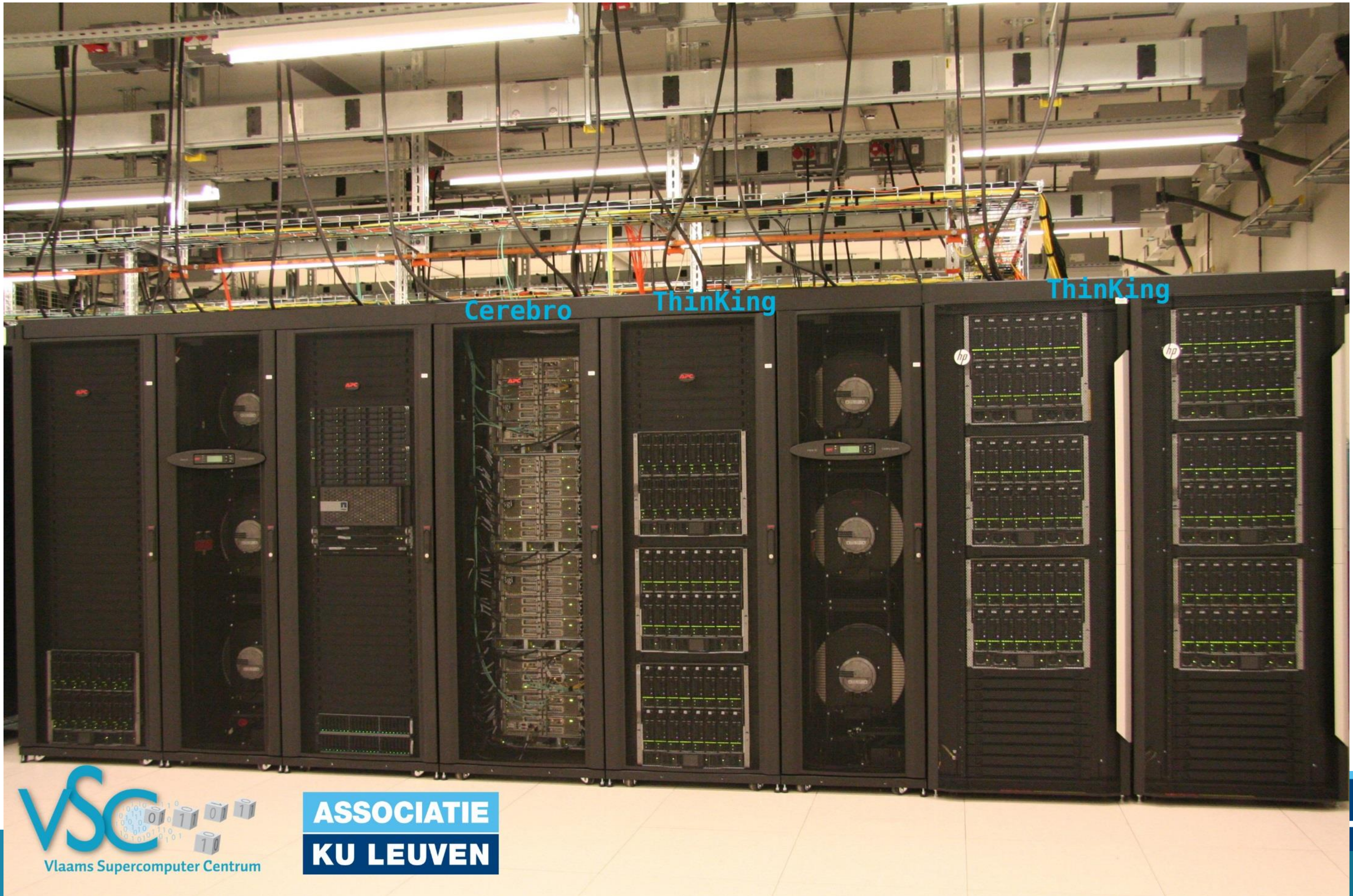
Compute-bound problems

	Cerebro
Total sockets	64
Processor type	Ivy bridge (Xeon E5-4650)
Base Clock Frequency	2.4 GHz
Cores per socket	10
Total cores	640
Memory per socket (GB)	16x128/48x256 = 14 TB
Memory per core (GB)	6.4/12.8
Peak performance (TF)	12.3
Network	NUMAlink62:1
Cache (L1 KB/L2 KB/LLC MB)	10x(32i+32d)/10x256/25
Total LLC cache per core	2.5 MB

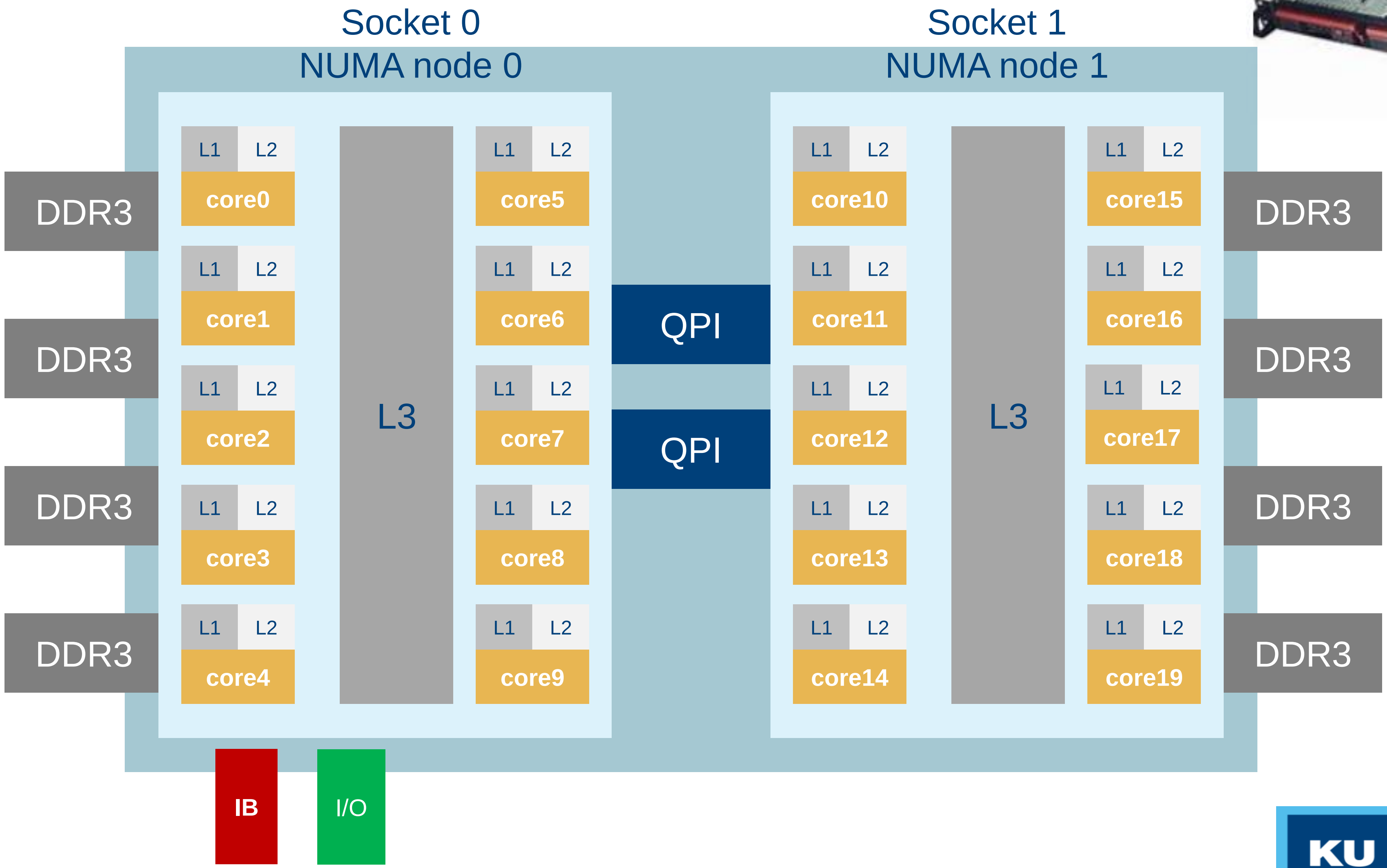
Memory-bound problems

KU LEUVEN

KU Leuven/UHasselt Tier 2



Compute node



Overview of the storage infrastructure

HPC cluster storage at KU Leuven consists of 3 different storage types, optimized for different usage, with different characteristics

- NAS storage, fully back-up with snapshots for `home` and `data`
- `Scratch` storage, fast parallel filesystem
- `Archive` storage, to store large amounts of data for long time (slow I/O)
- `Staging` similar to archive but faster I/O

Overview of the storage infrastructure

- **Home directory (3GB)**
  - Location available as `$VSC_HOME`
  - The data stored here should be relatively small (e.g., no files or directories larger than a gigabyte, although this is allowed), and not generating very intense I/O during jobs.
  - Typically all kinds of configuration files are stored here, e.g., ssh-keys, .bashrc, or Matlab, and Eclipse configuration, ...

Overview of the storage infrastructure

- **Data directory (75 GB)**
  - Location available as `$VSC_DATA`
  - A bigger 'workspace', for datasets, results, logfiles, ...
  - This filesystem can be used for higher I/O loads, but for I/O bound jobs, you might be better of using one of the 'scratch' filesystems.

Overview of the storage infrastructure

- **Scratch directories (starts from 100GB)**
  - Free temporary upgrade of quota
  - For temporary or transient data; there is typically no backup for these filesystems, and 'old' data is be removed automatically after 21 days (duriation of longest job allowed on the cluster)
  - Currently `$VSC_SCRATCH` (`$VSC_SCRATCH_SITE`) are defined, for space that is available per user per site

Tip:

- Do not use `/tmp` directory on compute node (very limited space ~10GB, once exceeded the system and your job will crash).
- Use `$VSC_SCRATCH_NODE` (`/local_scratch`) instead (~200GB)

Login nodes

- 8 login nodes - different purpose, different limits
  - 2 login nodes
    - login.hpc.kuleuven.be
    - login2.hpc.kuleuven.be

    Basic command line login
  - 2 nx nodes, access through server
    - nx1
    - nx2

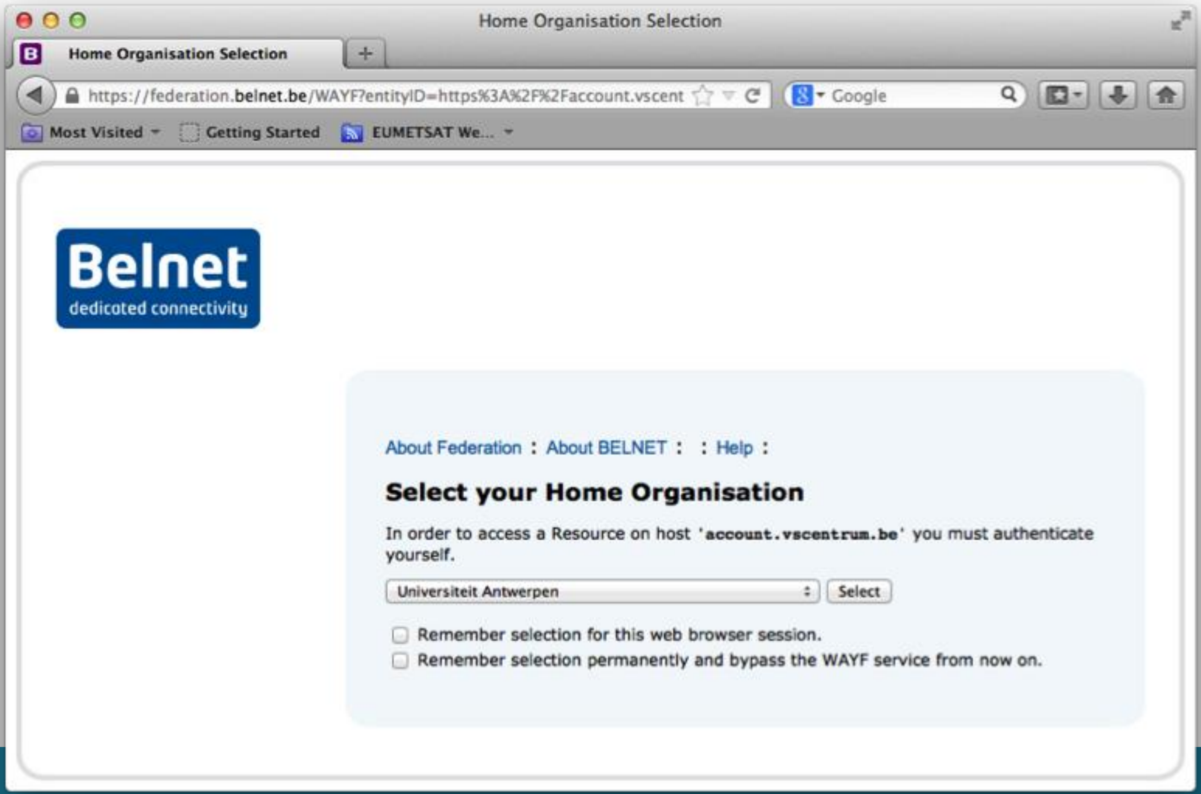
    GUI login
  - 4 interactive nodes, configurable for user groups
    - inode1
    - inode2
    - inode3
    - inode4

    Special requests (contact us)

KU LEUVEN

Account request

- https://account.vscentrum.be/
- KU Leuven: only from inside KU Leuven (PC-class, wifi, KU Leuven computer)
- U Hasselt: from inside the University or VPN


Home Organisation Selection
Home Organisation Selection
https://federation.belnet.be/WAYF?entityID=https%3A%2F%2Faccount.vscent
Google
Most Visited
Getting Started
EUMETSAT We...
Belnet
dedicated connectivity
About Federation : About BELNET : : Help :

Select your Home Organisation

In order to access a Resource on host 'account.vscentrum.be' you must authenticate yourself.

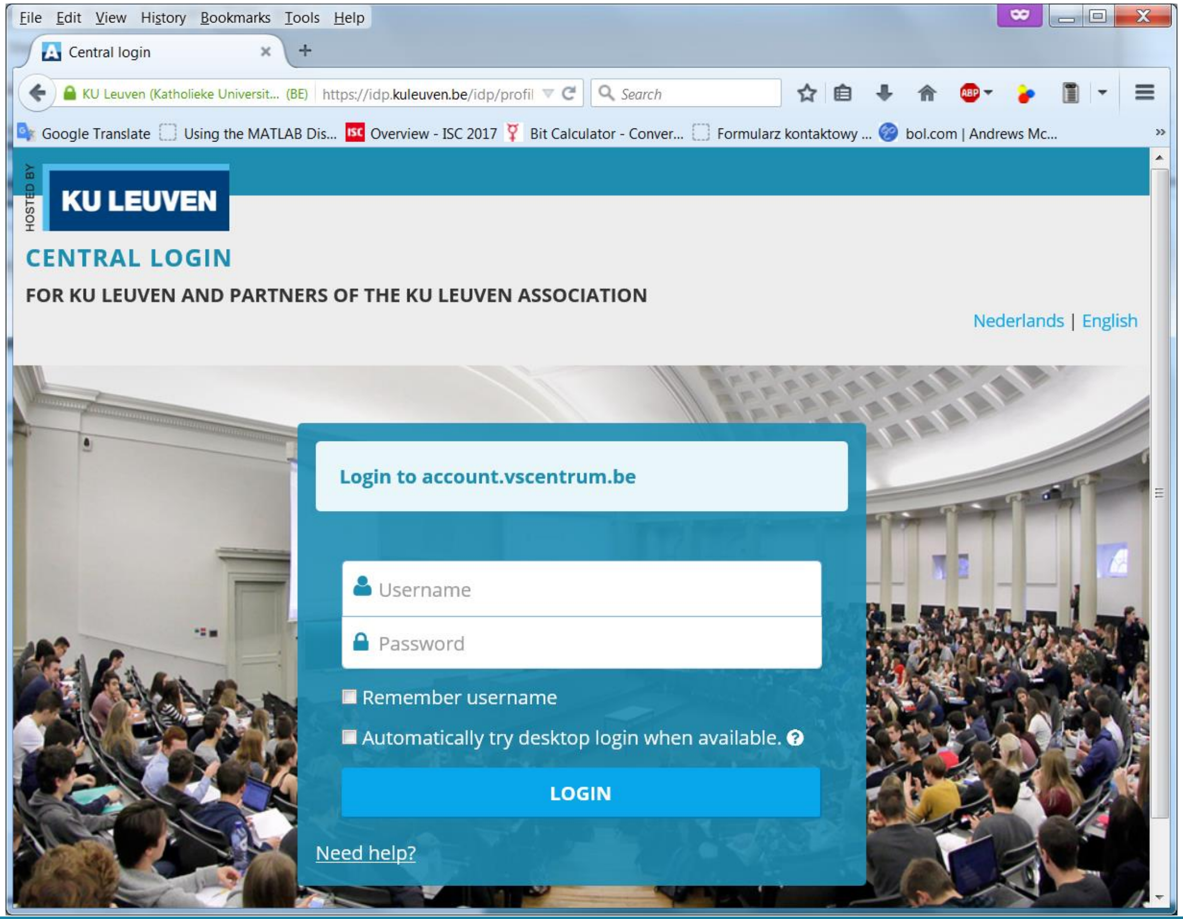
Universiteit Antwerpen [Select]

☐ Remember selection for this web browser session.
☐ Remember selection permanently and bypass the WAYF service from now on.

KU LEUVEN

Account request

- https://account.vscentrum.be/
- KU Leuven


File Edit View History Bookmarks Tools Help
Central login
KU Leuven (Katholieke Universit... (BE) https://idp.kuleuven.be/idp/profil
Search
Google Translate
Using the MATLAB Dis...
Overview - ISC 2017
Bit Calculator - Conver...
Formularz kontaktowy ...
bol.com | Andrews Mc...

HOSTED BY KU LEUVEN

CENTRAL LOGIN

FOR KU LEUVEN AND PARTNERS OF THE KU LEUVEN ASSOCIATION

Nederlands | English

Login to account.vscentrum.be

Username

Password

Remember username

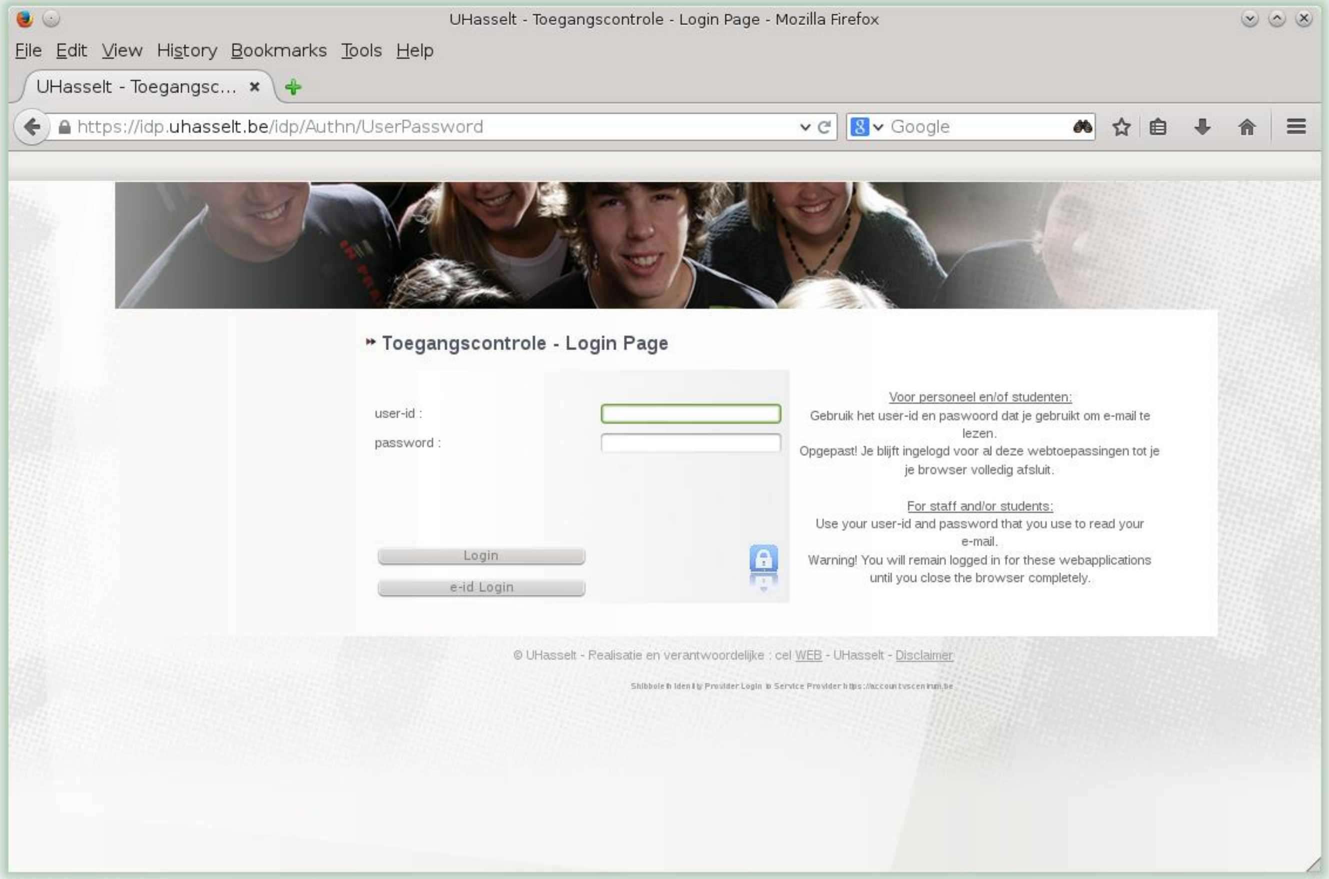
Automatically try desktop login when available.

LOGIN

Need help?

Account request

- https://account.vscentrum.be/
- U Hasselt


UHasselt - Toegangscontrole - Login Page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
UHasselt - Toegangsc...
https://idp.uhasselt.be/idp/Authn/UserPassword
Google

Toegangscontrole - Login Page

user-id :

password :

Voor personeel en/of studenten:
Gebruik het user-id en paswoord dat je gebruikt om e-mail te lezen.
Opgepast! Je blijft ingelogd voor al deze webtoepassingen tot je je browser volledig afsluit.

For staff and/or students:
Use your user-id and password that you use to read your e-mail.
Warning! You will remain logged in for these webapplications until you close the browser completely.

© UHasselt - Realisatie en verantwoordelijke : cel WEB - UHasselt - Disclaimer

Shibboleth Identity Provider Login to Service Provider https://account.vscentrum.be

Account request

- https://account.vscentrum.be/
- Authentication: Staff/student-id, e-mail
- In case of change – please inform us (access to the webpage may not be possible, SSH to the cluster not affected by that)

Connecting to the cluster

- Unlike the webpage – cluster is accessible from any place without VPN
- In case of change of your status (student-staff) please inform us – account may be set as inactive
- SSH
  - Secure Shell
  - SSH commonly uses port 22 to connect your computer to another computer on the Internet. It is most often used by network administrators as a remote login / remote control way to manage their business servers.

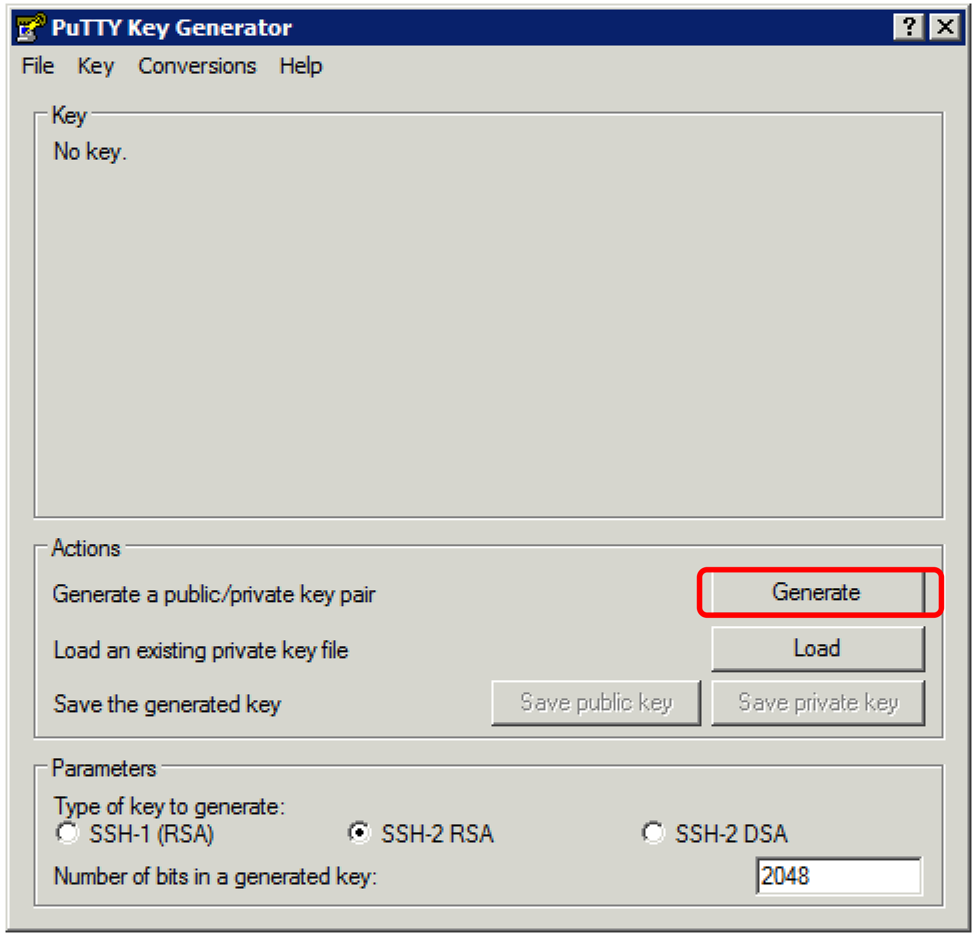
Generate the key

Linux/Mac OS users:

- Use “*ssh-keygen ” command to generate key pair*
- **Be sure to give your key a passphrase!**
- Default location: ~/.ssh/id_rsa (~/.ssh/id_rsa.pub)
- If other location: `ssh –i` *`localtion-of–the-file`* ……
- **keychain**: ssh agent to load the key with passphrase for current linux session

Generate the key

Windows users: use the PuTTYgen key generator.



No key.

Generate a public/private key pair

Load an existing private key file

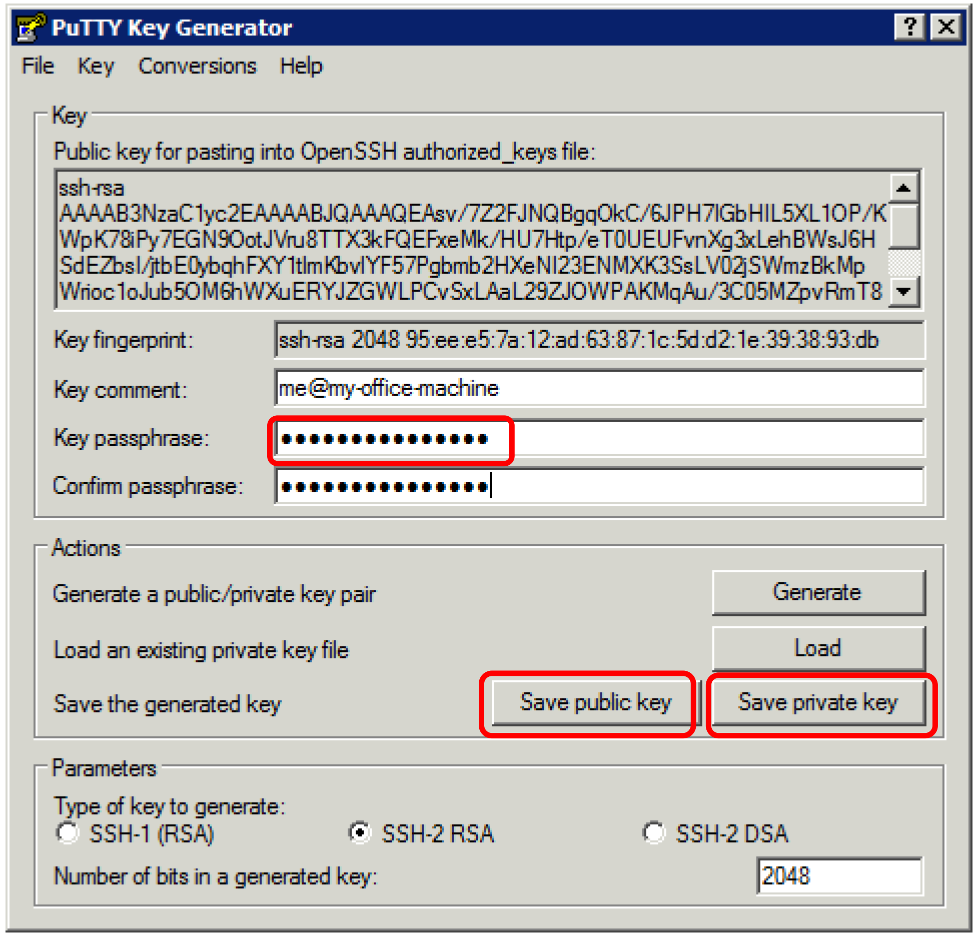
Save the generated key

Type of key to generate:
( ) SSH-1 (RSA) [✓ SSH-2 RSA] ( ) SSH-2 DSA

Number of bits in a generated key: 2048

Generate the key

Be sure to give your key a passphrase!



Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAsv/7Z2FJNQBgqOkC/6JPH7IGbHIL5XL1OP/K
WpK78iPy7EGN9OotJVru8TTX3kFQEFxeMk/HU7Htp/eT0UEUFvnXg3xLehBWsJ6H
SdEZbsI/jtbE0ybqhFXY1tlmKbvIYF57Pgbmb2HXeNI23ENMXK3SsLV02jSWmzBkMp
Wrioc1oJub5OM6hWXuERYJZGWLPCvSxLAaL29ZJOWPAKMqAu/3C05MZpvRmT8
```

Key fingerprint: ssh-rsa 2048 95:ee:e5:7a:12:ad:63:87:1c:5d:d2:1e:39:38:93:db

Key comment: me@my-office-machine

Key passphrase: ••••••••••••••••

Confirm passphrase: ••••••••••••••••

Actions

Generate a public/private key pair — Generate

Load an existing private key file — Load

Save the generated key — Save public key — Save private key

Parameters

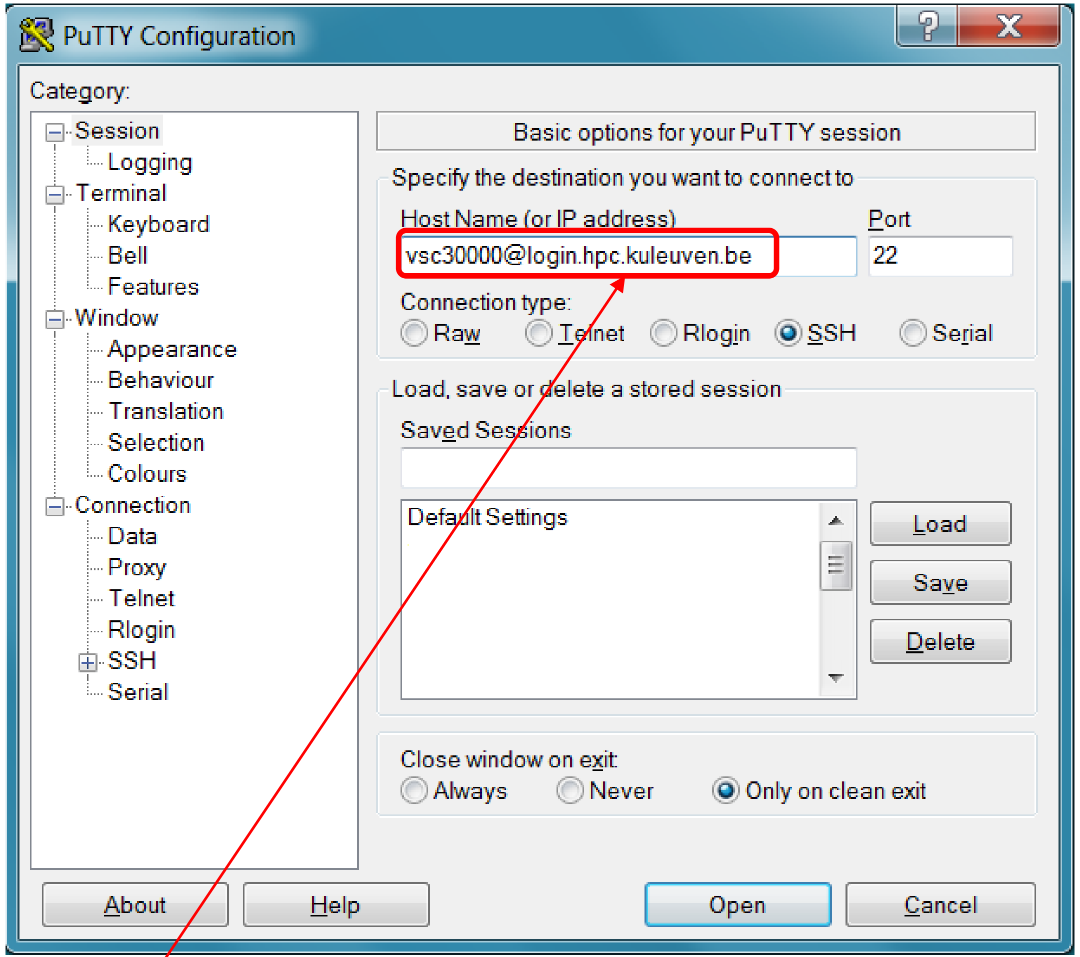
Type of key to generate:
SSH-1 (RSA) [✓ SSH-2 RSA] SSH-2 DSA

Number of bits in a generated key: 2048

Connecting to the cluster: text mode

Windows users:

- PuTTY is a simple-to-use and freely available GUI SSH client for Windows.
- Pageant can be used to manage active keys for PuTTY, WinSCP and FileZilla so that you don't need to enter the passphrase all the time.


PuTTY Configuration

Category:

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): vsc30000@login.hpc.kuleuven.be

Port: 22

Connection type: Raw Telnet Rlogin [✓ SSH] Serial

Load, save or delete a stored session

Saved Sessions

Default Settings

Close window on exit: Always Never [✓ Only on clean exit]

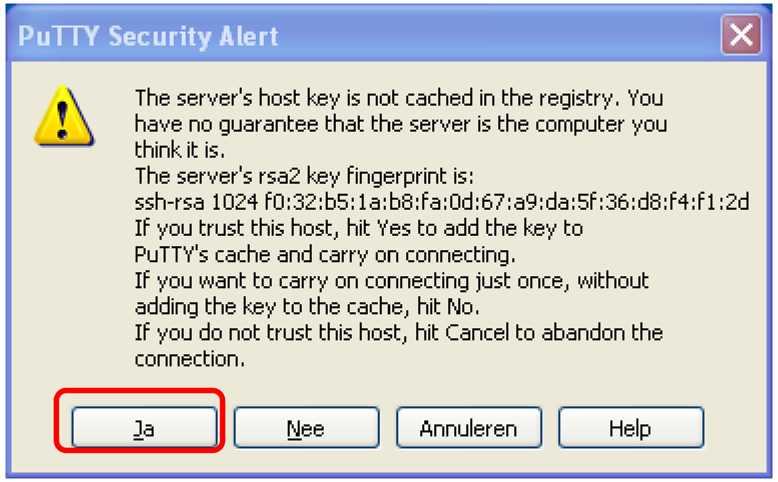
```
vsc30000@login.hpc.kuleuven.be
```

KU LEUVEN

Connecting to the cluster: text mode

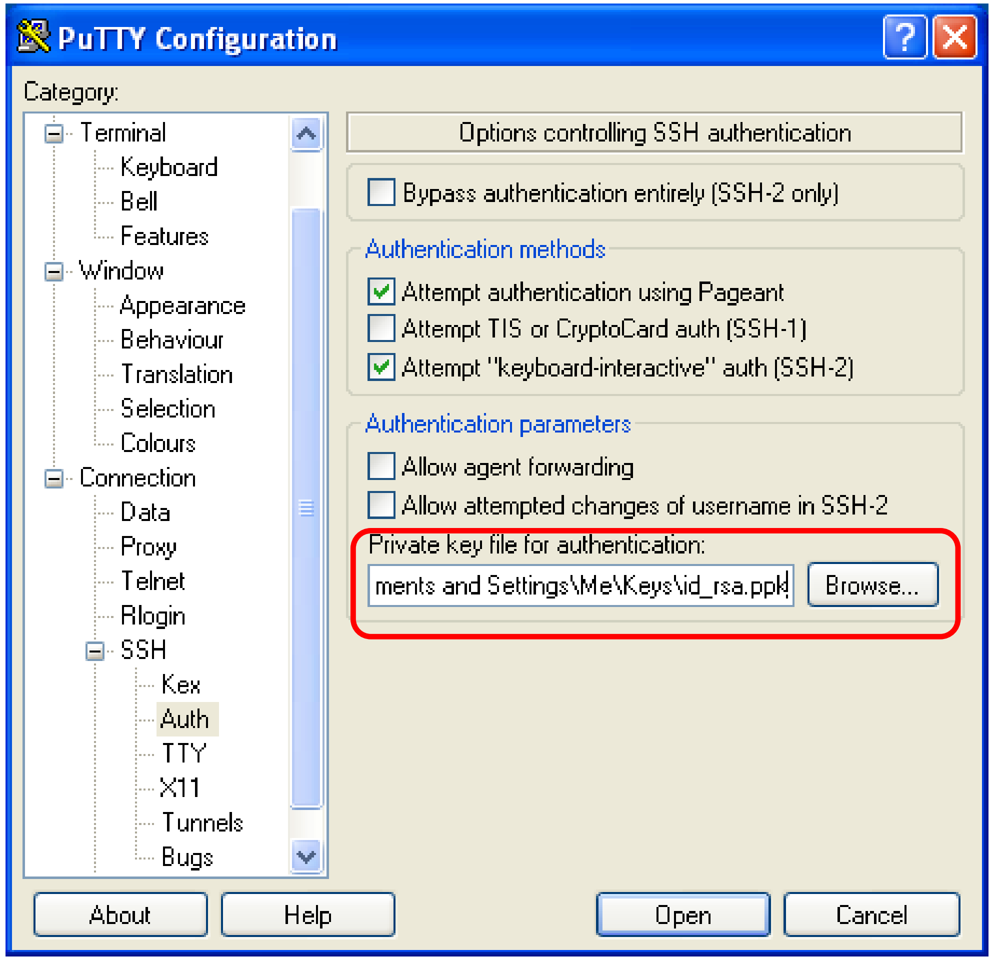
Windows users:

- PuTTY is a simple-to-use and freely available GUI SSH client for Windows.



PuTTY Security Alert

The server's host key is not cached in the registry. You have no guarantee that the server is the computer you think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 1024 f0:32:b5:1a:b8:fa:0d:67:a9:da:5f:36:d8:f4:f1:2d
If you trust this host, hit Yes to add the key to PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without adding the key to the cache, hit No.
If you do not trust this host, hit Cancel to abandon the connection.



PuTTY Configuration

Options controlling SSH authentication

- [ ] Bypass authentication entirely (SSH-2 only)

Authentication methods

- [✓] Attempt authentication using Pageant
- [ ] Attempt TIS or CryptoCard auth (SSH-1)
- [✓] Attempt "keyboard-interactive" auth (SSH-2)

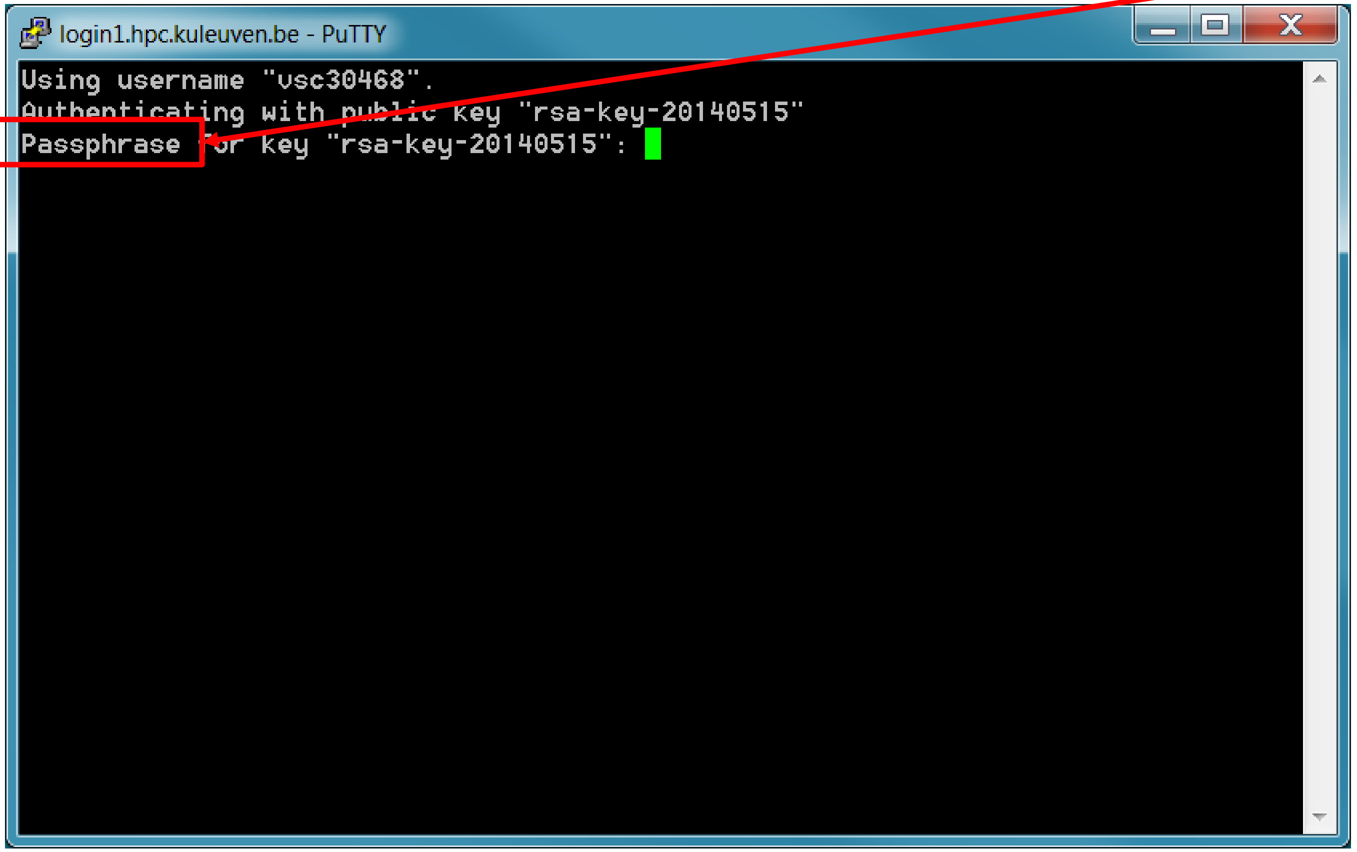
Authentication parameters

- [ ] Allow agent forwarding
- [ ] Allow attempted changes of username in SSH-2

Private key file for authentication:
ments and Settings\Me\Keys\id_rsa.ppk

Connecting to the cluster: text mode

Windows users:



login1.hpc.kuleuven.be - PuTTY

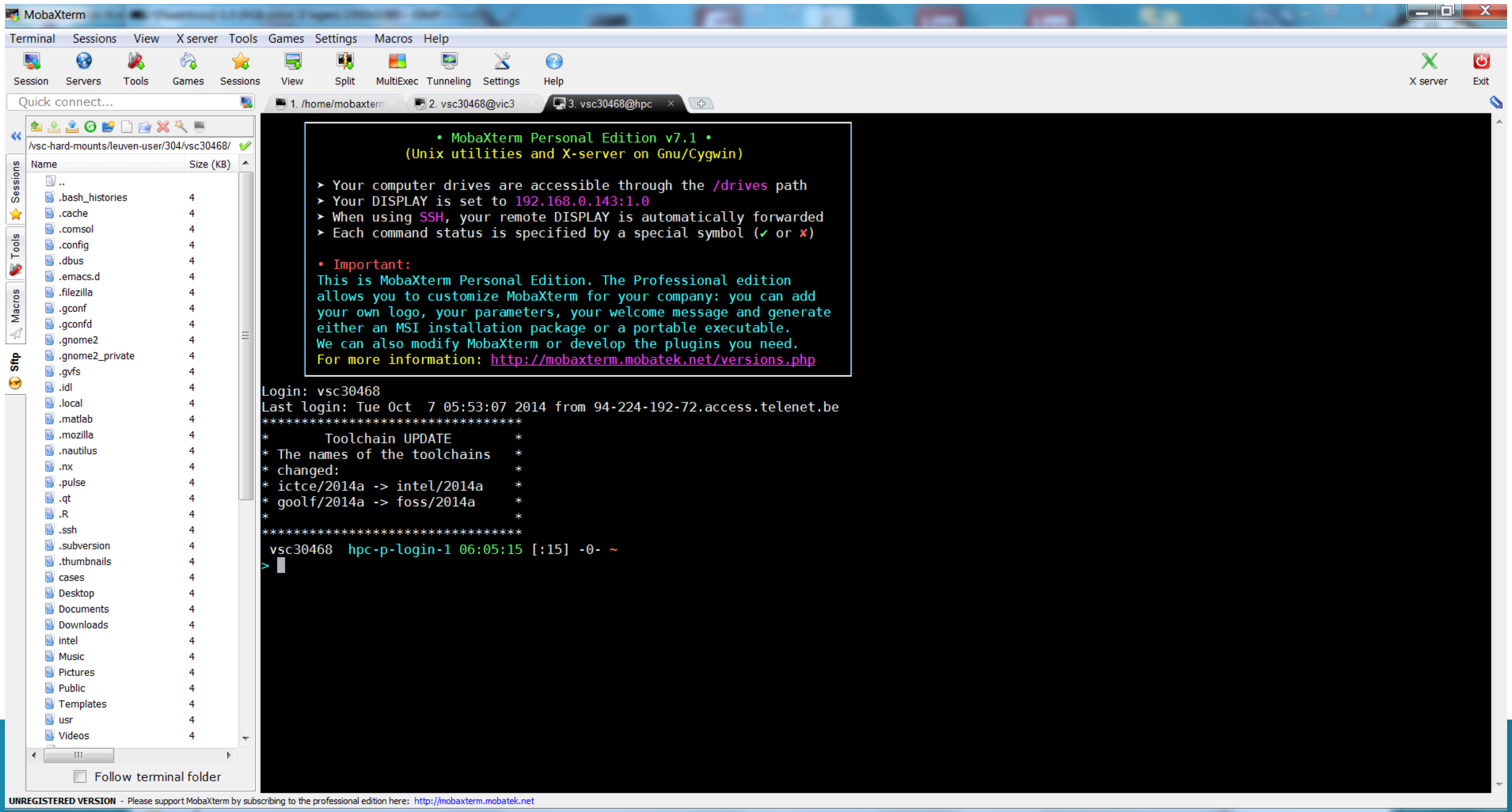
```
Using username "vsc30468".
Authenticating with public key "rsa-key-20140515"
Passphrase for key "rsa-key-20140515":
```

If asked for **password** no for passphrase – please stop connecting and contact suport, otherwise after a few attempts you will be blocked for 24hrs

Connecting to the cluster: All in one

Windows users:

MobaXterm - the complete toolbox for remote computing: an enhanced terminal with an X11 server, an SSH client and all the essential Unix commands to Windows desktop.


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Settings Help
X server Exit
Quick connect...
1. /home/mobaxterm
2. vsc30468@vic3
3. vsc30468@hpc
Sessions Tools Macros Sftp
Follow terminal folder

```
/vsc-hard-mounts/leuven-user/304/vsc30468/
Name                Size (KB)
..
.bash_histories     4
.cache              4
.comsol             4
.config             4
.dbus               4
.emacs.d            4
.filezilla          4
.gconf              4
.gconfd             4
.gnome2             4
.gnome2_private     4
.gvfs               4
.idl                4
.local              4
.matlab             4
.mozilla            4
.nautilus           4
.nx                 4
.pulse              4
.qt                 4
.R                  4
.ssh                4
.subversion         4
.thumbnails         4
cases               4
Desktop             4
Documents           4
Downloads           4
intel               4
Music               4
Pictures            4
Public              4
Templates           4
usr                 4
Videos              4
```

```
                • MobaXterm Personal Edition v7.1 •
            (Unix utilities and X-server on Gnu/Cygwin)

 ➤ Your computer drives are accessible through the /drives path
 ➤ Your DISPLAY is set to 192.168.0.143:1.0
 ➤ When using SSH, your remote DISPLAY is automatically forwarded
 ➤ Each command status is specified by a special symbol (✔ or ✘)

 • Important:
 This is MobaXterm Personal Edition. The Professional edition
 allows you to customize MobaXterm for your company: you can add
 your own logo, your parameters, your welcome message and generate
 either an MSI installation package or a portable executable.
 We can also modify MobaXterm or develop the plugins you need.
 For more information: http://mobaxterm.mobatek.net/versions.php

Login: vsc30468
Last login: Tue Oct  7 05:53:07 2014 from 94-224-192-72.access.telenet.be
*********************************
*       Toolchain UPDATE        *
* The names of the toolchains   *
* changed:                      *
* ictce/2014a -> intel/2014a    *
* goolf/2014a -> foss/2014a     *
*                               *
*********************************
 vsc30468  hpc-p-login-1 06:05:15 [:15] -0- ~
>
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: http://mobaxterm.mobatek.net

Job Scheduler: Moab

How jobs are scheduled:

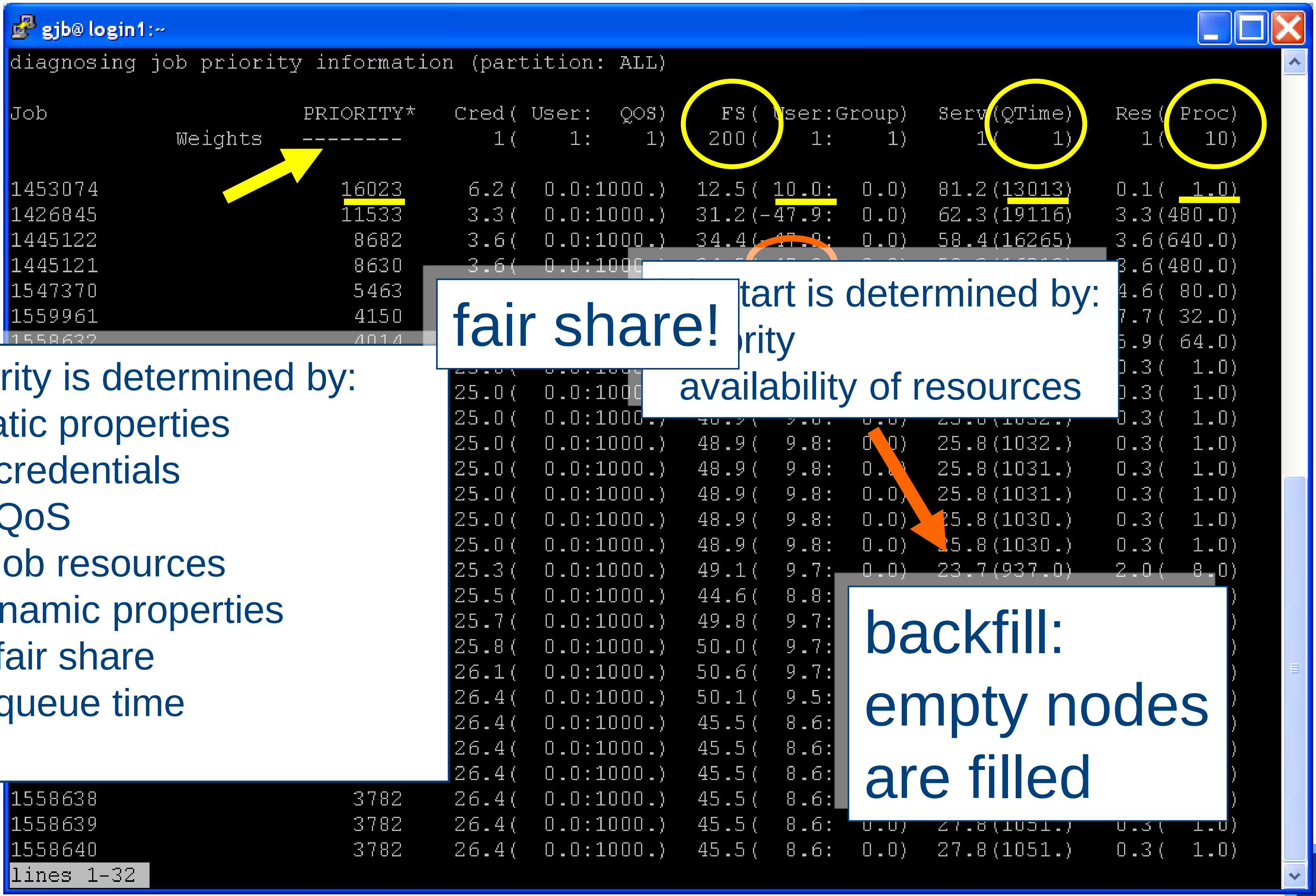
- The job scheduler decides when and where to run jobs using a priority queue
  - Queries resource manager for runnable jobs
  - Tries to optimize resource usage and
    - favors parallel jobs,
    - fair use of resources for all users (taking into account computation during past 7 days),
    - ensures liveliness, regardless of starting priority, job will eventually have highest priority
  - Orders resource manager to start (or stop …) jobs

Job Scheduler: Moab

How jobs are scheduled:

- the whole procedure is redone in every iteration (every minute or so)
- a job can lose its reservation, if a higher priority job gets submitted in the meantime
- a job can start earlier, if a job ends earlier than specified
- the “showstart” command gives a startime estimate obtained in the last iteration

Moab priorities



```
diagnosing job priority information (partition: ALL)

Job                  PRIORITY*   Cred( User:   QOS)   FS( User:Group)  Serv(QTime)   Res( Proc)
             Weights --------       1(    1:     1)  200(    1:     1)     1(     1)     1(   10)

1453074                  16023    6.2( 0.0:1000.)  12.5( 10.0:  0.0)  81.2(13013)   0.1(  1.0)
1426845                  11533    3.3( 0.0:1000.)  31.2(-47.9:  0.0)  62.3(19116)   3.3(480.0)
1445122                   8682    3.6( 0.0:1000.)  34.4(-47.9:  0.0)  58.4(16265)   3.6(640.0)
1445121                   8630    3.6( 0.0:100                                      3.6(480.0)
1547370                   5463                                                      4.6( 80.0)
1559961                   4150                                                      7.7( 32.0)
1558632                   4014                                                      5.9( 64.0)
                                                                                    0.3(  1.0)
                                 25.0( 0.0:1000                                     0.3(  1.0)
                                 25.0( 0.0:1000.)                                   0.3(  1.0)
                                 25.0( 0.0:1000.)  48.9(  9.8:  0.0)  25.8(1032.)   0.3(  1.0)
                                 25.0( 0.0:1000.)  48.9(  9.8:  0.0)  25.8(1031.)   0.3(  1.0)
                                 25.0( 0.0:1000.)  48.9(  9.8:  0.0)  25.8(1031.)   0.3(  1.0)
                                 25.0( 0.0:1000.)  48.9(  9.8:  0.0)  25.8(1030.)   0.3(  1.0)
                                 25.0( 0.0:1000.)  48.9(  9.8:  0.0)  25.8(1030.)   0.3(  1.0)
                                 25.3( 0.0:1000.)  49.1(  9.7:  0.0)  23.7(937.0)   2.0(  8.0)
                                 25.5( 0.0:1000.)  44.6(  8.8:
                                 25.7( 0.0:1000.)  49.8(  9.7:
                                 25.8( 0.0:1000.)  50.0(  9.7:
                                 26.1( 0.0:1000.)  50.6(  9.7:
                                 26.4( 0.0:1000.)  50.1(  9.5:
                                 26.4( 0.0:1000.)  45.5(  8.6:
                                 26.4( 0.0:1000.)  45.5(  8.6:
                                 26.4( 0.0:1000.)  45.5(  8.6:
1558638                   3782   26.4( 0.0:1000.)  45.5(  8.6:
1558639                   3782   26.4( 0.0:1000.)  45.5(  8.6:  0.0)  27.8(1051.)   0.3(  1.0)
1558640                   3782   26.4( 0.0:1000.)  45.5(  8.6:  0.0)  27.8(1051.)   0.3(  1.0)
lines 1-32
```

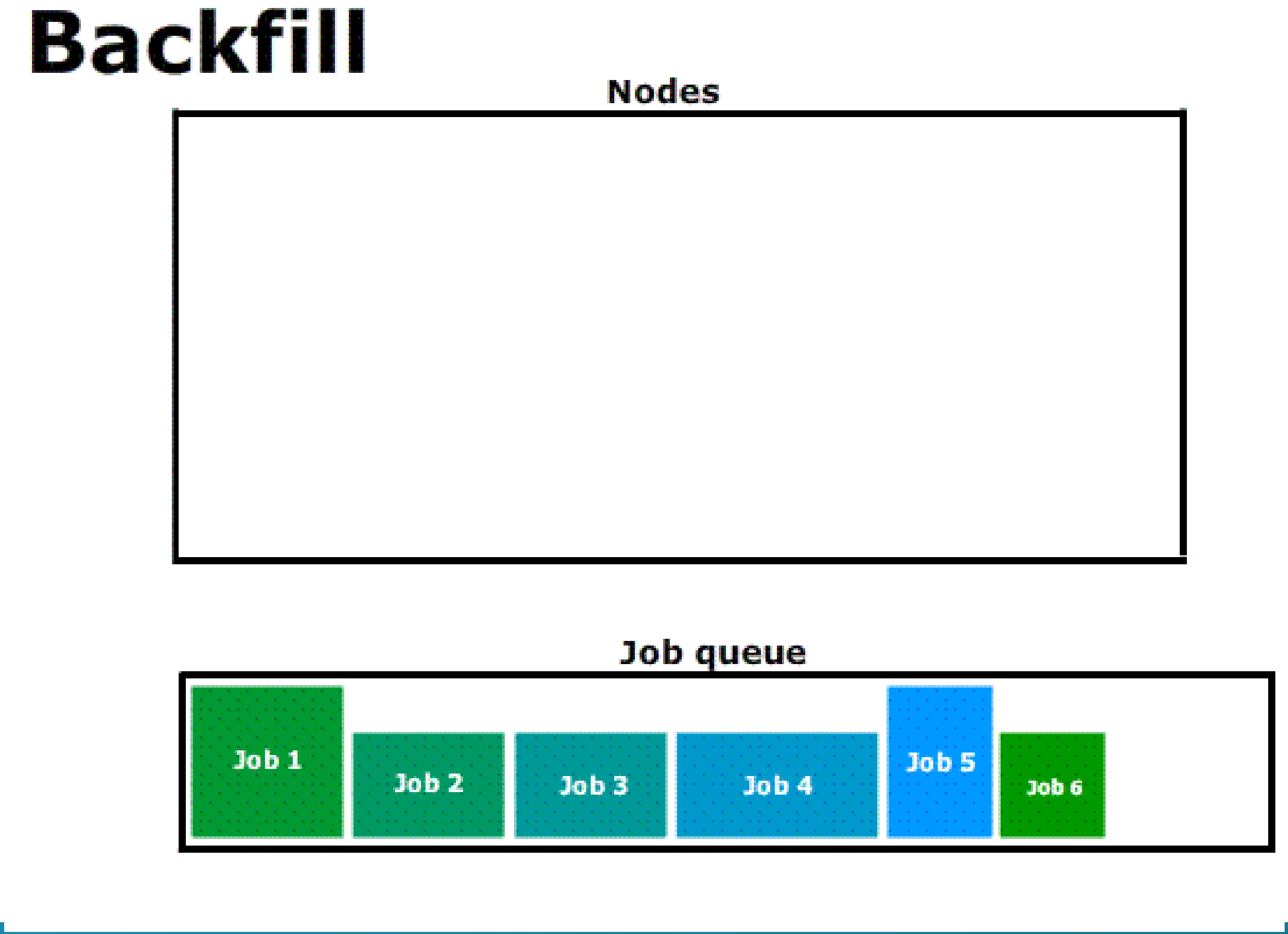
priority is determined by:
- static properties
  - credentials
  - QoS
  - job resources
- dynamic properties
  - fair share
  - queue time

fair share!

tart is determined by:
rity
availability of resources

backfill:
empty nodes are filled

Job Scheduler: Moab



Job Scheduler: Moab

- Some Moab commands:
  - `$ checkjob <job-id>` : shows job information
  - `$ showstart <job-id>` : shows the ***earliest*** time job *can, not will* start (at the time of the query)

Credits

How to check available credits?

```
$ module load accounting;
mam-balance
```

Credits card concept:

- Preauthorization: holding the balance as unavailable until the merchant clears the transaction
- Balance to be held as unavailable: based on requested resourced (walltime, nodes)
- Actual charge based on what was really used: used walltime (you pay only what you use, e.g. when job crashes)
- See output file:

```
Resource List:
neednodes=2:ppn=6,nodes=2:ppn=6,pmem=1gb,walltime=01:00:00
Resources Used:
cput=00:00:00,mem=0kb,vmem=0kb,walltime=00:00:02
```

Moab Allocation Manager

How to check the rates?

```
$ module load accounting;
gquote …
```

Example 1: `gquote -l walltime=01:00:00` 6.68

Example 2: `gquote -l nodes=1` 6.68

Example 3: `gquote -l nodes=1:ppn=1` 6.68

Example 4: `gquote -l nodes=1:ppn=20` 6.68

Example 5: `gquote -l walltime=01:00:00 –l nodes=1:haswell` 6.68

Example 6: `gquote -l walltime=01:00:00 –l nodes=1:ivybridge` 4.76

KU LEUVEN

Tool chains

- Default: 2014a
  - Switch: `$ source switch_to_2016a`
    `$ source switch_to_2015a`
    `$ source switch_to_2014a`
  - or `$ module unuse /apps/leuven/thinking/2014a/modules/all; module use /apps/leuven/thinking/2016a/modules/all`

Modules

Set the environment to use software package:

- `$ module available` or `module av R`
  - Lists all installed software packages
- `$ module av |& grep -i python`
  - To show only the modules that have the string 'python' in their name, regardless of the case
- `$ module load matlab/R2014a`
  - Adds the ‘matlab’ command in your PATH
- `$ module load GCC`
  - ‘Load’ the (default) GCC version
- `$ module list`
  - Lists all ‘loaded’ modules in current session
- `$ module unload R/3.1.2-foss-2014a-x11`
  - Removes all only the selected module, other loaded modules – dependencies are still loaded
- `$ module purge`
  - Removes all loaded modules from your environment

Software

- By default ThinKing software is listed (`$ module available`)
- Software of Cerebro and accelerators is different than of ThinKing.
- Some software for Haswell different than for Ivybridge (default)
- Software compiled on ThinKing might run on Cerebro/accelerators/… BUT…

  For MPI/GPU software better to recompile your programs on a specific platform (check available modules first)
- To check available modules:

```
$ module load cerebro/2014a
$ module load thinking2/2014a
$ module load K20Xm/2014a
$ module load K40c/2014a
$ module load phi/2014a
```

$\leftarrow$ Haswell nodes (`thinking2/2014a`)

$\leftarrow$ accelerator nodes (`K20Xm/2014a`, `K40c/2014a`, `phi/2014a`)

- To compile – ask for interactive job

Before you start

- Account https://account.vscentrum.be/req
  - Mailing list
  - Optionally: VSC Echo
- Do you have your key?

  https://www.vscentrum.be/client/windows/keys-putty

  https://www.vscentrum.be/client/linux/keys-openssh
- Credits (reasonable motivation)

  https://admin.kuleuven.be/icts/onderzoek/hpc/request-introduction-credits
- Stay up-to-date

  https://www.vscentrum.be/en/education-and-trainings

KU LEUVEN

Accounts

- Follow the procedure documented on: https://www.vscentrum.be/cluster-doc/account-request
- Apply for an account on
  - https://account.vscentrum.be
  - Create key pair and upload public key
- User IDs are of the form `vsc3xxxx`
- Login to
  - `login.hpc.kuleuven.be`
- Request introduction credits https://admin.kuleuven.be/icts/onderzoek/hpc/request-introduction-credits
- Request project credits https://admin.kuleuven.be/icts/onderzoek/hpc/request-project-credits

  https://icts.kuleuven.be/sc/forms/Aanvraagformulier_HPC_Credits
- Extra project credits (to add to existing project)

  https://admin.kuleuven.be/icts/onderzoek/hpc/extra-project-credits

Prices

Introduction credits	• Maximum 2000 • Valid up to 6 months	FREE
Project credits	• Commercial partner **0.12 €** • Internal funding, IWT, FWO, European funding **0.007 €**	
Storage	• /scratch, temporary quota upgrade FREE • /data 25 GB , 14.63 €/year • HPC archive 1 TB, 70 €/year • /staging 1 TB, 130 €/year	
Application support	• Software Installation • Debugging & profiling up to 5 days • Parallelization & optimization up to 5 days	FREE
Training	• Scheduled courses • Infosessions • Thematic workshops	FREE

Prices and information: https://icts.kuleuven.be/sc/english/HPC

KU LEUVEN

Questions

- Now
- Helpdesk:
  hpcinfo@icts.kuleuven.be or https://admin.kuleuven.be/icts/HPCinfo_form/HPC-info-formulier
- VSC web site:
  http://www.vscentrum.be/
  - VSC documentation
  - VSC agenda: training sessions, events
- Systems status page:
  http://status.kuleuven.be/hpc

  https://www.vscentrum.be/en/user-portal/system-status

KU LEUVEN

Backup slides

Job types

- **Batch jobs** are by far the most common, and allow for the most efficient use of the infrastructure. Essentially, a batch job is a bash shell script that is executed on a compute node, and that can spawn a parallel computation on many nodes. These jobs are placed in the queue, and the user can forget about it until it is finished.
- **Interactive jobs** are intended to work on one or more compute nodes interactively. This can be useful in the context of software development for debugging and profiling applications, or for interactive calculations or visualizations. Basically, one gets a shell on one of the compute nodes.

Interactive jobs

- `$ qsub -I`
  - opens shell on a compute node for 1h
- `$ qsub -I -X`
  - opens shell, with X-forwarding
- `$ qsub -I -l nodes=4:ppn=2,walltime=8:00:00`
  - open shell for 8h, with access to 4 nodes, 2 cores each
- `$ qsub -I -l pmem=6gb`
  - get a shell on a compute node with 6 GB of RAM available per core

Note: on 64GB node, only ± 60 GB available!

KU LEUVEN

Batch job with resource specifications

- From command line

```
$ qsub -l walltime=10:00:00 myjob.pbs
```

- In a PBS job script

```
#!/bin/bash -l
#PBS -l walltime=10:00:00
#PBS -l nodes=2:ppn=1
#PBS -l pmem=3gb
#PBS -N myjob2
#PBS -A default project
#PBS -m abe
#PBS -M my.name@icts.kuleuven.be
```

If not specified:
walltime=01:00:00

If not specified:
nodes=1:ppn=1

If not specified:
pmem=2400mb

If not specified: job will not start
- default_project = intro credits
- lp_my_project_= project credits

My first pbs script

```
#!/bin/bash -l
#PBS -l walltime=00:10:00
#PBS -l nodes=1:ppn=20:ivybridge
#PBS -l pmem=3gb
#PBS -N testjob
#PBS -A lp_hpcinfo_training
#PBS -m abe
#PBS -M my-name@kuleuven.be

module purge
source switch_to_2015a
module load matlab/R2016b

cd $PBS_O_WORKDIR
cd $VSC_SCRATCH

matlab -nojvm -nodisplay -r mat
~~exit 0~~
```

Clean the modules loaded by default in .bashrc
Load all the necessary modules (new shell is started for each job)

Go to the directory where your code and input files are located

Execute the code

Do not add unnecessary statements – in case of problem real exit code is overwritten, finding source of problem may be impossible

Extra requests

- **Submit to Cerebro:**
  - `module load cerebro`
  - `qsub…`
- **Submit to a GPU-node:**
  - `qsub -l partition=gpu <jobscript>`
  - `qsub -l partition=gpu,nodes=1:K20Xm <jobscript>`
- **Submit to a Phi node:**
  - `qsub -l partition=phi <jobscript>`
- **Submit to a debug node (**less than 30 min, max 2 nodes**):**
  - `qsub -lqos=debugging,walltime=15:00 <jobscript>`

PBS environment variables

When your PBS script runs, some environment variables are automatically available

- `PBS_O_WORKDIR=/vsc-hard-mounts/leuven-user/304/vsc30468` directory from where job was submitted
- `PBS_JOBID=20590968.moab.tier2.leuven.vsc` job ID of current job
- `PBS_JOBNAME=STDIN` user specified jobname
- `PBS_WALLTIME=3600` requested walltime
- `PBS_NUM_NODES=2` number of nodes allocated to the job
- `PBS_NP=40` number of execution cores for the job
- `PBS_NUM_PPN=20` number of procs per node allocated to the job
- `PBS_NODENUM=0` unique number 0 upwards for each different node
- `PBS_VNODENUM=0` the number of the physical core in your allocation. The numbering continues across the nodes in the allocation
- `PBS_NODEFILE=/var/spool/torque/aux//20590968.moab.tier2.leuven.vsc` file containing line delimited list on nodes allocated to the job
- `PBS_O_PATH` path variable used to locate executables within job script
- `PBS_TASKNUM=1` number of the task

Job’s life cycle

- Job submitted (qsub) – has job-ID

- Checking a job's status (showq, qstat, checkjob)
- Modifying job specifications (qalter)
- Deleting a job (qdel)

- Credits:
  - module load accounting -> mam-balance
  - using the project: qsub -A project_name ......
  - default_project=intro credits

Moab Allocation Manager

$$\#credits = \left(\boxed{0.000278 \cdot walltime} \cdot \#nodes\right) \cdot f_{type}$$

$\xrightarrow{}$ 1/3600

Project credits valid for all Tier-2 clusters

$$f_{type} = \begin{cases} 4.76 & \text{ThinKing ivybridge} \\ 2.86 & \text{GPU \& Xeon Phi} \\ 3.45 & \text{Cerebro} \\ 6.68 & \text{ThinKing haswell} \end{cases}$$

Example: `-l nodes=1:ppn=1:ivybridge,walltime=1:00:00`

$$\#credits = \left(0.000278 \cdot 3600 \cdot 1\right) \cdot 4.76 \approx 4.76$$

Example: `-l nodes=1:ppn=20:ivybridge,walltime=1:00:00`

$$\#credits = \left(0.000278 \cdot 3600 \cdot 1\right) \cdot 4.76 \approx 4.76$$

Glossary

- **RAM**: Random Access Memory used as working memory for the CPUs. On current hardware, the size of RAM is expressed in gigabytes (GB). The RAM is shared between the two CPUs on each of the sockets. Once the process that creates the data ends, the data in the RAM is no longer available. The complete RAM can be accessed by each core.
- **Walltime**: the actual time an application runs (as in clock on the wall), or is expected to run. When submitting a job, the walltime refers to the maximum amount of time the application can run. For accounting purposes, the walltime is the amount of time the application actually ran, typically less than the requested walltime.

KU LEUVEN

Glossary

- **Node-hour**: unit of work indicating that an application ran for a time t on n nodes, such that n*t = 1 hour. Using 1 node for 1 hour is 1 node-hour. This is irrespective of the number of cores on the node you actually use.
- **Core-hour**: unit of work indicating that an application ran for a time t on p cores, such that p*t = 1 hour. Using 20 cores, no matter on how many nodes, for 1 hour results in 20 core-hours.
- **Node-day**: unit of work indicating that an application ran for a time t on n nodes such that n*t = 24 hours. Using 3 nodes for 8 hours results in 1 node day.

Glossary

- **Memory requirement**: the amount of RAM needed to successfully run an application. It can be specified per process for a distributed application, expressed in GB.
- **Storage requirement**: the amount of disk space needed to store the input and output of an application, expressed in GB or TB.
- **Temporary storage requirement**: the amount of disk space needed to store temporary files during the run of an application, expressed in GB or TB.

Glossary

- **Single user per node policy**: indicates that when a process of user *A* runs on a compute node, no process of another user will run on that compute node concurrently, i.e., the compute node will be exclusive to user *A*. However, if one or more processes of user *A* are running on a compute node, and that node's capacity in terms of available cores and memory is not exceeded, processes part of another job submitted by user *A* may start on that compute node.

Glossary

- **Process**: an independent computation running on a computer. It may interact with other processes, and it may run multiple threads. A serial and shared memory application run as a single process, while a distributed application consists of multiple, coordinated processes.
- **Threads**: a process can perform multiple computations, i.e., program flows, concurrently. In scientific applications, threads typically process their own subset of data, or a subset of loop iterations.

Glossary

- **Serial application**: a program that runs a single process, with a single thread. All computations are done sequentially, i.e., one after the other, no explicit parallelism is used.
- **Multi-core application**: a multi-core application uses more than one core during its execution by running multiple threads, also called a shared memory application.
- **Distributed application**: an application that uses multiple compute nodes for its computations, concurrent computations are executed as processes. These processes communicate by exchanging messages, typically implemented by calls to an MPI library. Messages can be used to exchange data and coordinate the execution.

Archive Storage

- The archive tier is built with DDN WOS storage. It is intended to store data for longer term. The storage is optimized for capacity, not for speed. The storage by default is mirrored.
- No deletion rules are executed on this storage. The data will be kept until the user deletes it.
- **Use for**: Storing data that will not be used for a longer period and which should be kept. Compute nodes have no direct access to that storage area and therefore it should not be used for jobs I/O operations.
- **More info**: https://www.vscentrum.be/infrastructure/hardware/wos-storage.

Storage

- **Where to request**: https://admin.kuleuven.be/icts/onderzoek/hpc/hpc-storage.
- **More info**: https://icts.kuleuven.be/sc/english/research/HPC.

Generate the key

Linux/Mac OS users:

- Use “*ssh-keygen ” command to generate key pair*
- **Be sure to give your key a passphrase!**

```
user@desktop:~> ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
f6:61:a8:27:35:cf:4c:6d:13:22:70:cf:4c:c8:a0:23
```

KU LEUVEN