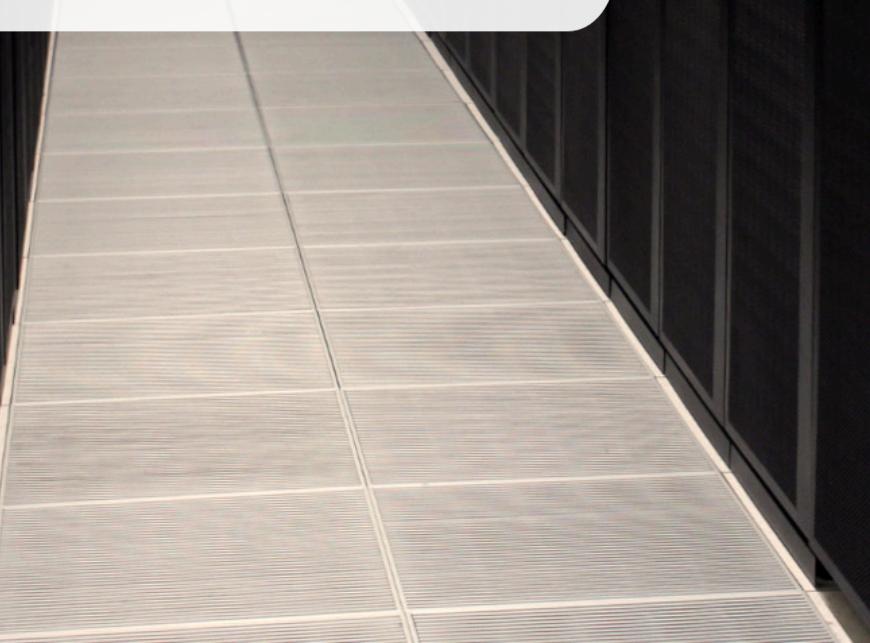


SURF

# Using a supercomputer

## Hands-on introduction

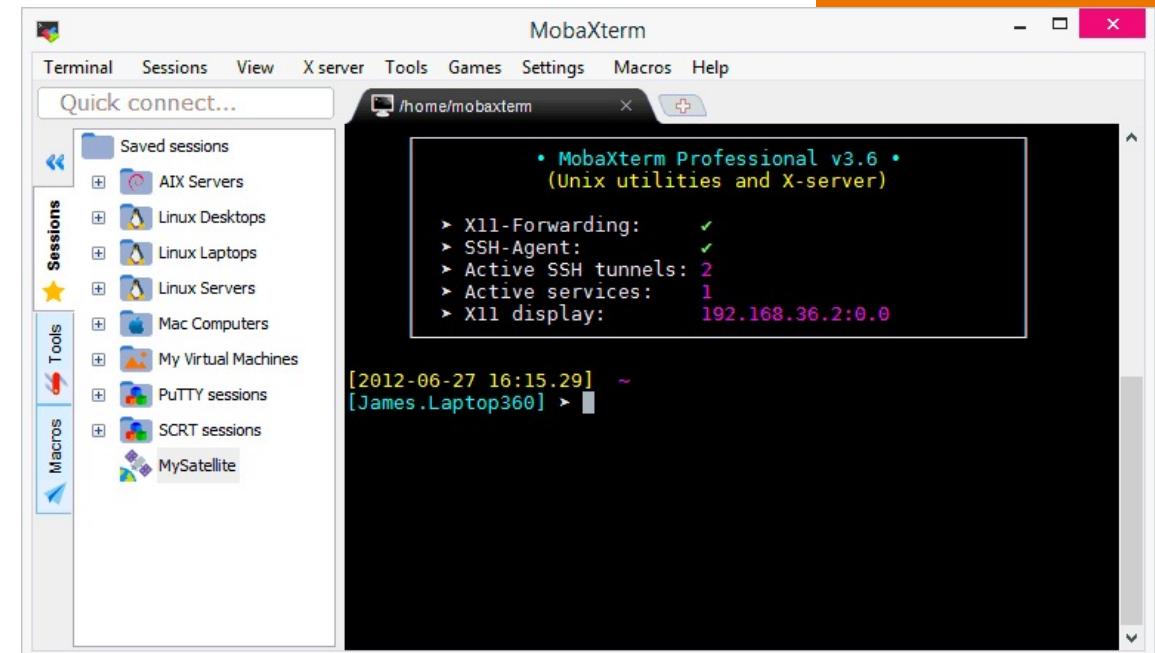
Benjamin Czaja, Marco Verdicchio  
HPC Advisors, SURF



# Working with a Supercomputer

## Install UNIX tools on your local machine

- Windows
  - Putty (<https://www.putty.org/>)
  - MobaXterm (<http://mobaxterm.mobatek.net>)
- Mac OSX
  - Terminal (pre-installed)
  - iTerm2 (<https://iterm2.com>)
- Linux
  - You are already well equipped!



MobaXterm interface

# Linux and Cluster Computing

## 1) Introduction

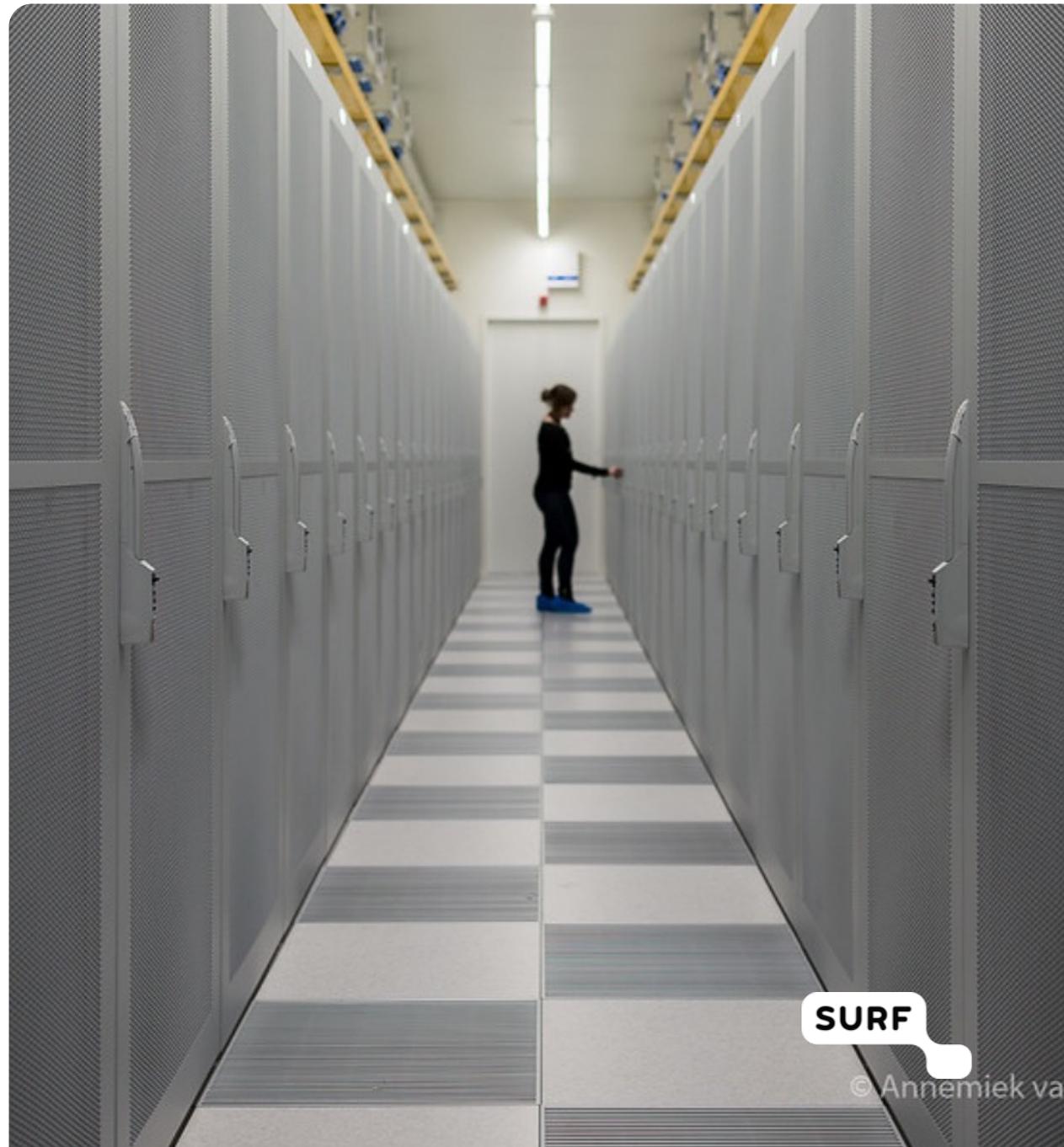
- a. What is a computer?
- b. What is a supercomputer?
- c. How is a supercomputer different from the cloud
- d. Access and usage

## 2) Introduction to Linux

- a. Getting started
- b. Usage and basic commands
- c. Shell script programming

## 3) Running jobs on the HPC system

- a) Interact with the batch scheduler
- b) Run a “real” scientific workflow



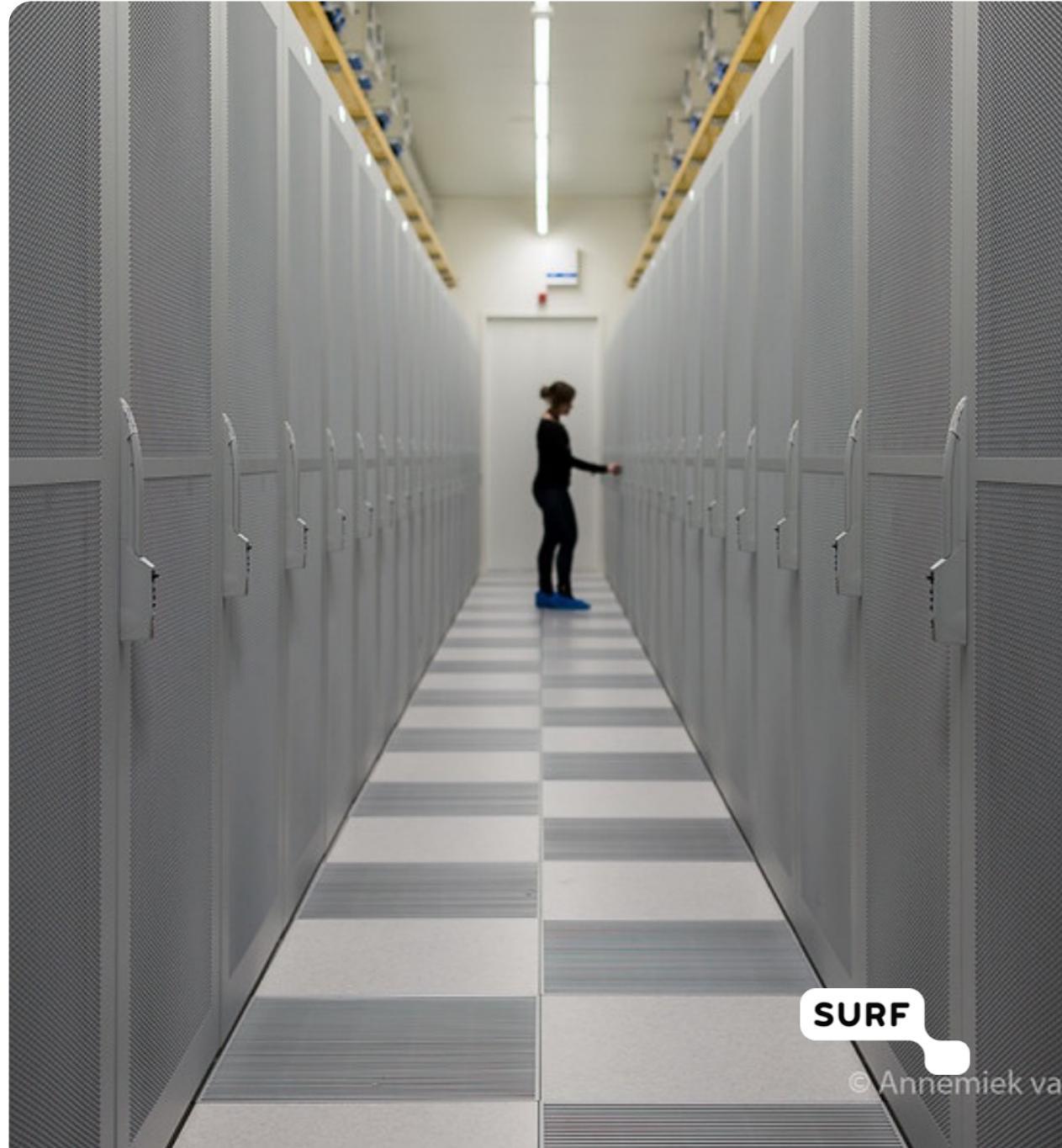
# SURF Bootcamp

Hands-on introduction using a supercomputer

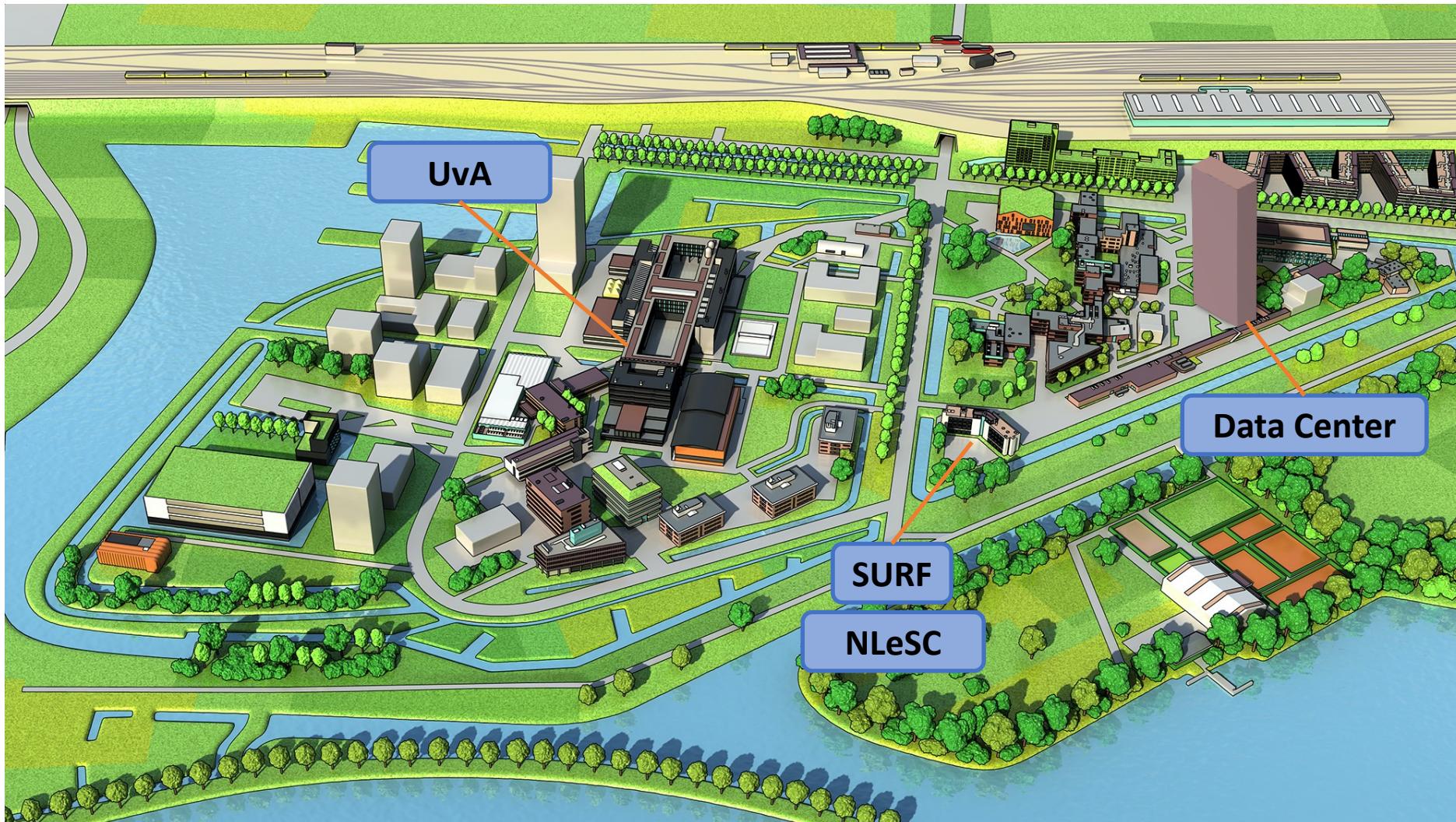
- Slides and exercises



<https://github.com/sara-nl/Intro-to-HPC>



# Location of SURF Amsterdam



**SURF**

# High-performance computing (HPC) is ...

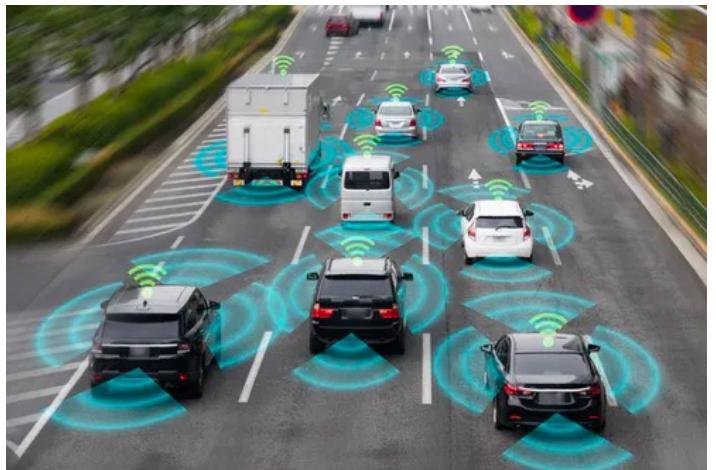
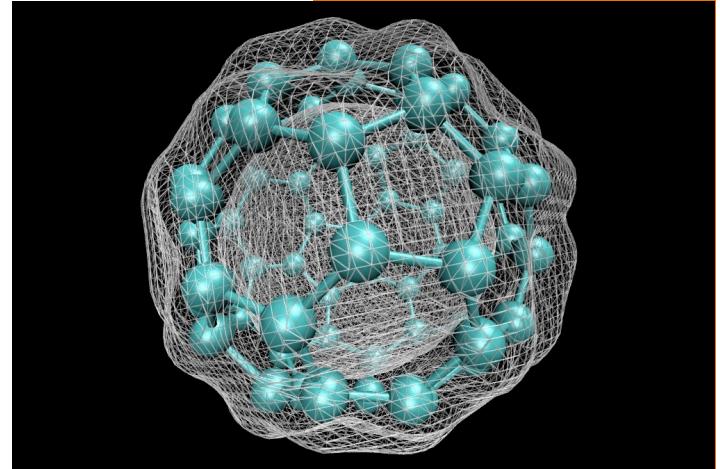


Image source: <https://www.trustedreviews.com/news/gran-turismo-ps5-will-complete-form-series-3895762>

Image source: <http://www.militarysystems-tech.com/articles/l-3-link-achieves-world-s-first-boeing-787-8-full-flight-simulator-level-d-certifications>

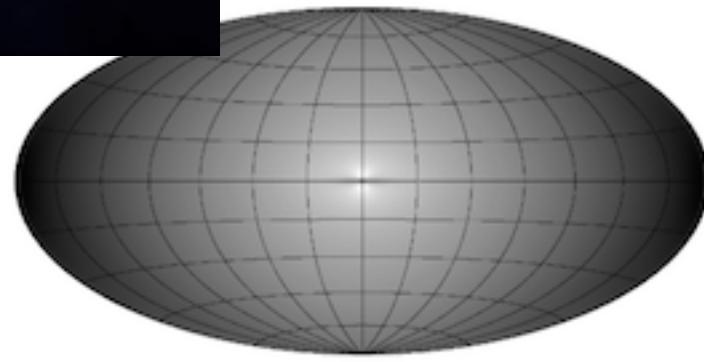
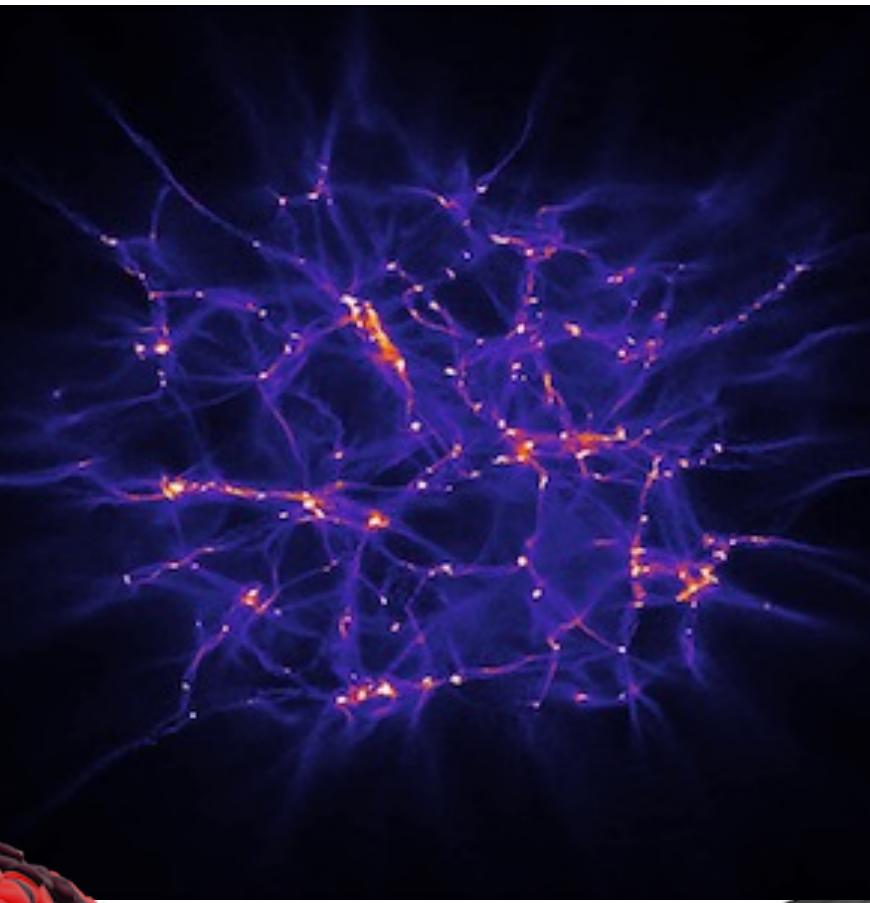
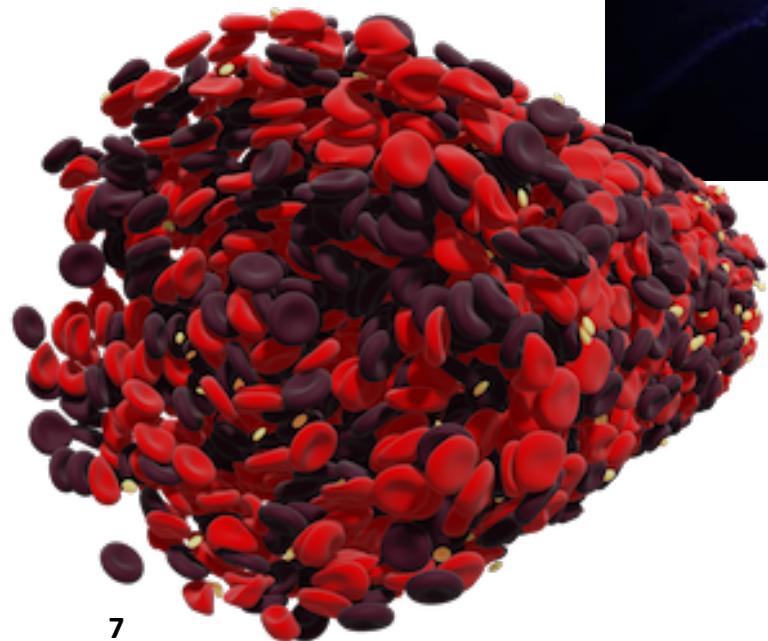
Image source: <https://www.zdnet.com/article/canberra-stands-up-au9-7m-transport-office-to-prep-for-autonomous-vehicles>

Image source: <https://spacenews.com/astranis-lands-anchor-customer-for-its-first-small-geo-satellite>

6 Image source: [https://medium.com/@info\\_89535/applications-of-fullerene-in-medicine-20942944e41d](https://medium.com/@info_89535/applications-of-fullerene-in-medicine-20942944e41d)

Image source: <https://thestrategybridge.org/the-bridge/2016/8/16/a-new-plan-using-complexity-in-the-modern-world>

# Who am I?



SURF

# High-performance computing (HPC) ...

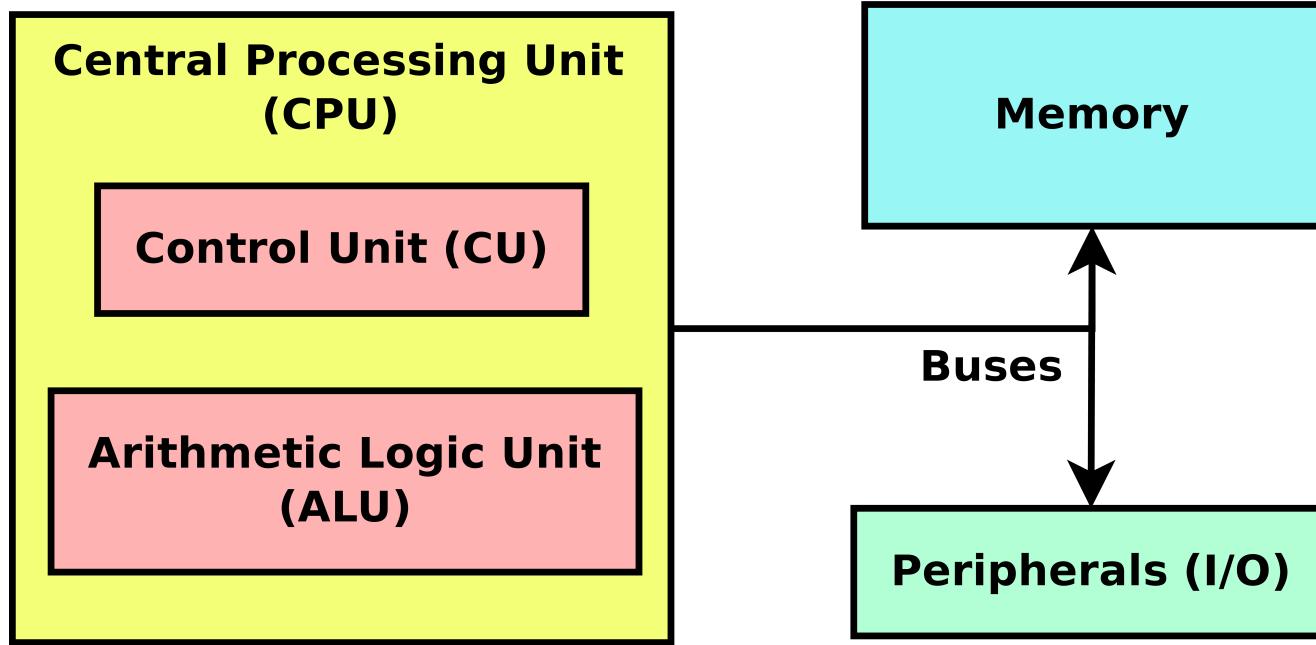
- ... *is an area of computer-based computation. It includes all computing work that requires a high computing capacity or storage capacity.*
- ... *is the use of parallel processing for running advanced application programs efficiently, reliably and fast.*
- ... *refers to the practice of aggregating computing power in a way that delivers much higher performance than one could get out of a typical desktop computer or workstation in order to solve large problems in science, engineering, or business.*
- ... *is the use of super computers and parallel processing techniques for solving complex computational problems.*



# **WORKING WITH A SUPERCOMPUTER**

- a. What is a computer?
- b. What is a supercomputer?
- c. Access and usage

# A computer is ...



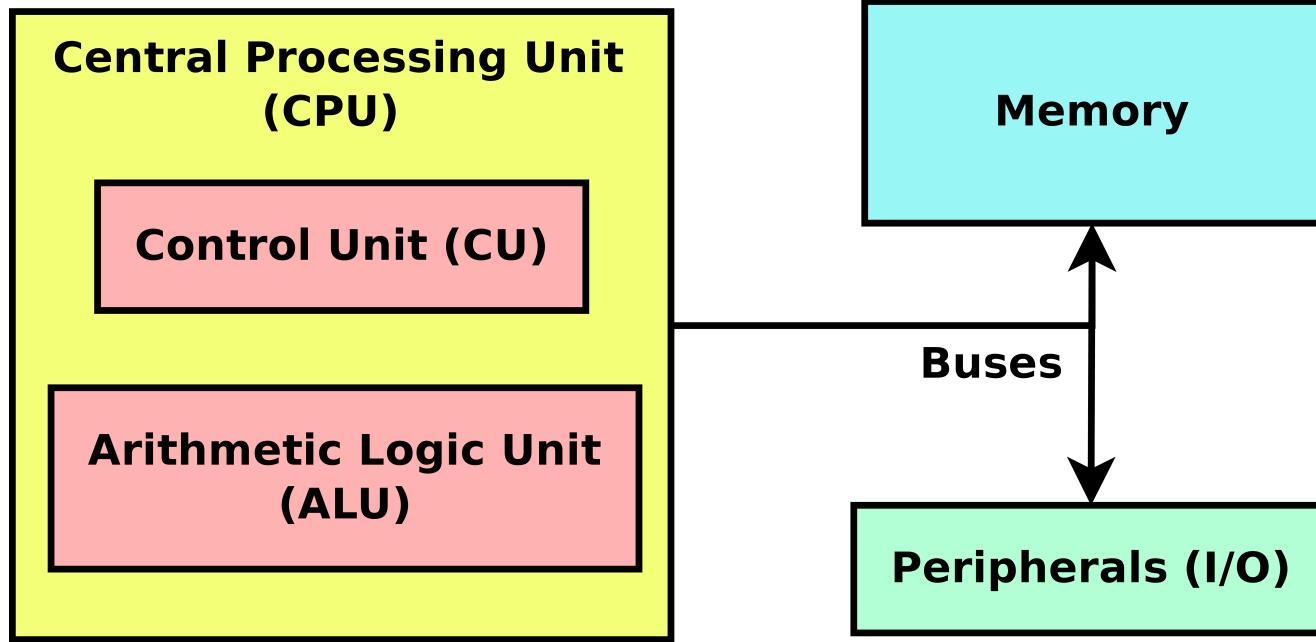
# Peripherals (I/O) are ...



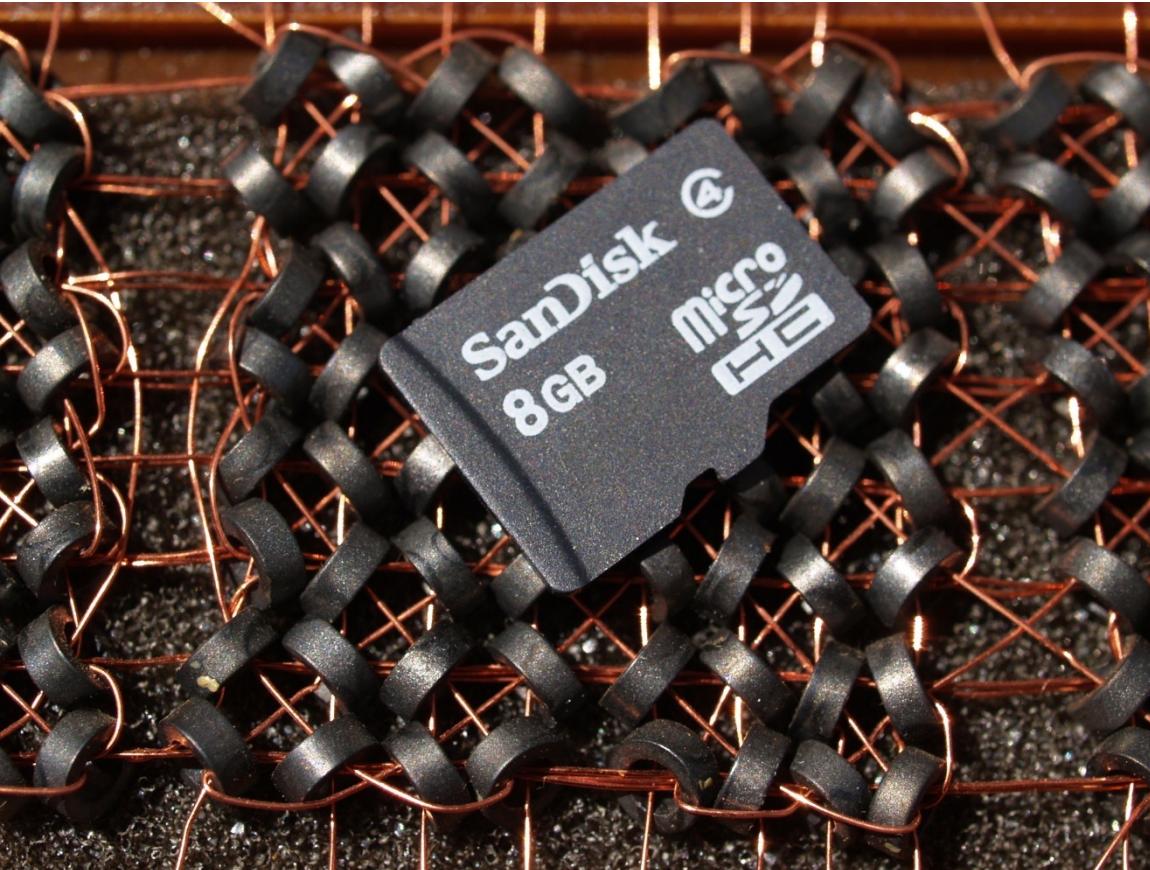
*Image source: [https://media.sciencephoto.com/image/t4150120/800wm/T4150120-Piles\\_of\\_discarded,\\_redundant\\_computer\\_keyboards.jpg](https://media.sciencephoto.com/image/t4150120/800wm/T4150120-Piles_of_discarded,_redundant_computer_keyboards.jpg)*

*Image source: <https://static.guim.co.uk/sys-images/Guardian/Pix/pictures/2014/2/10/1392028631237/Pile-of-computer-monitors-008.jpg>*

# A computer is ...

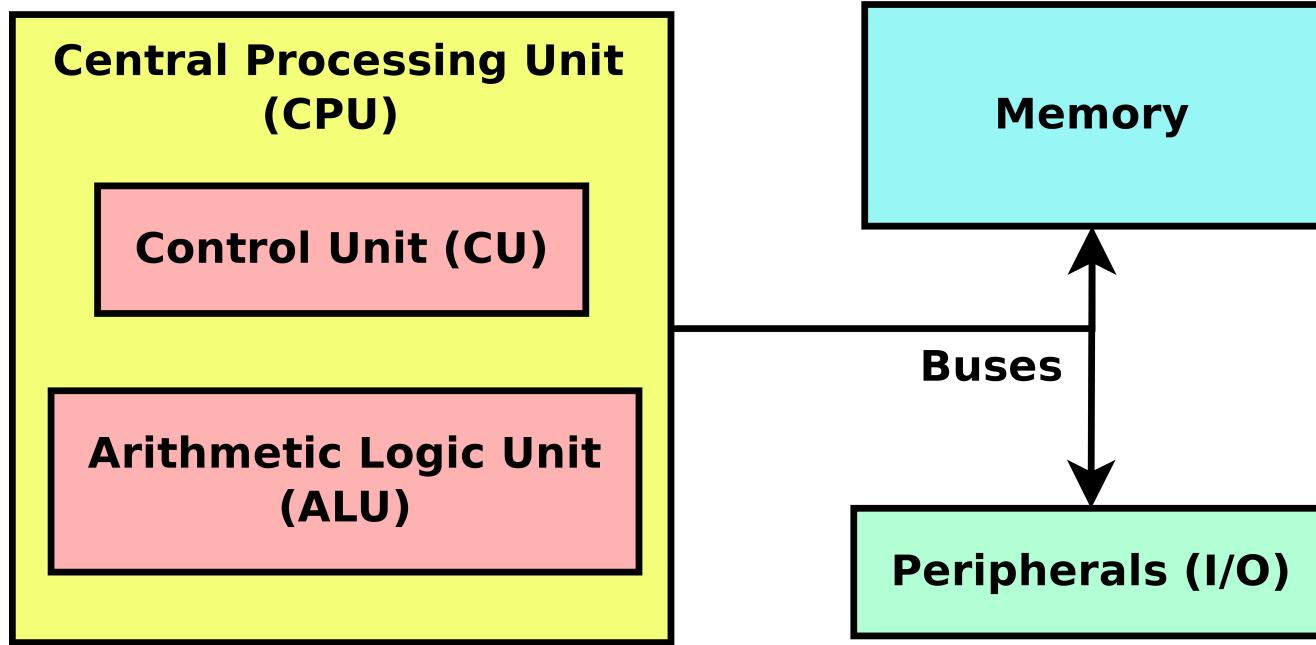


# A memory is ...

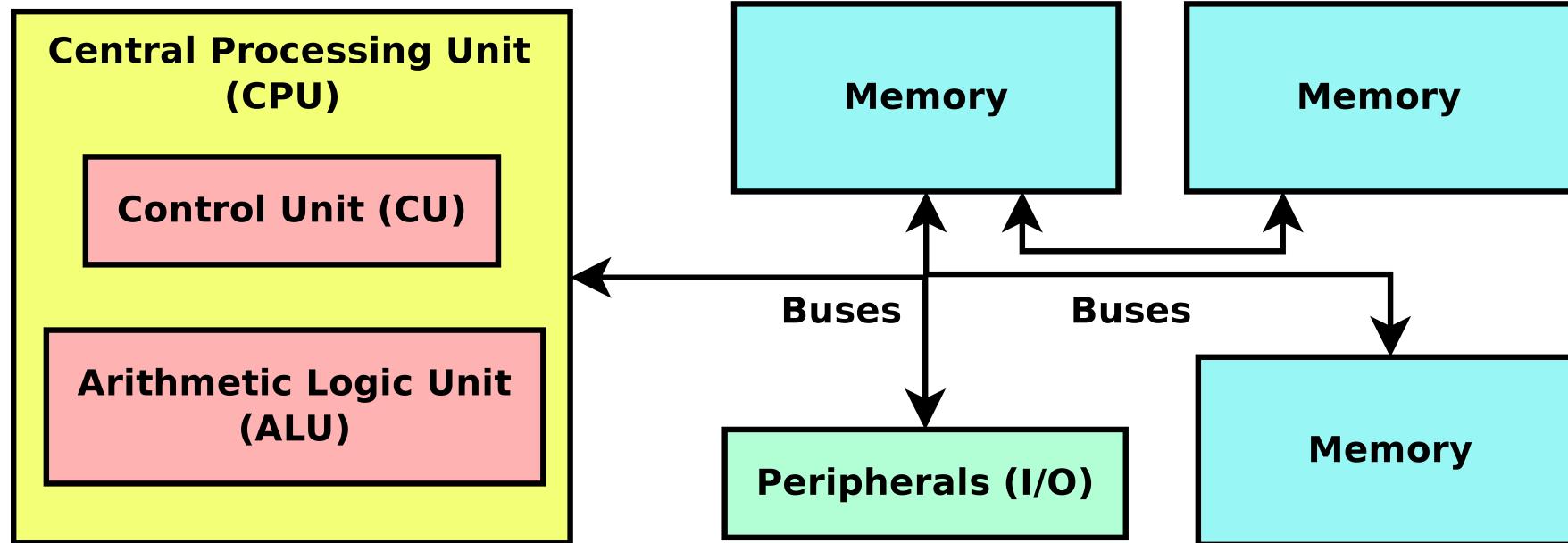


*Image source: [https://upload.wikimedia.org/wikipedia/commons/c/c0/8 bytes vs. 8Gbytes.jpg](https://upload.wikimedia.org/wikipedia/commons/c/c0/8%20bytes%20vs.%208Gbytes.jpg)*

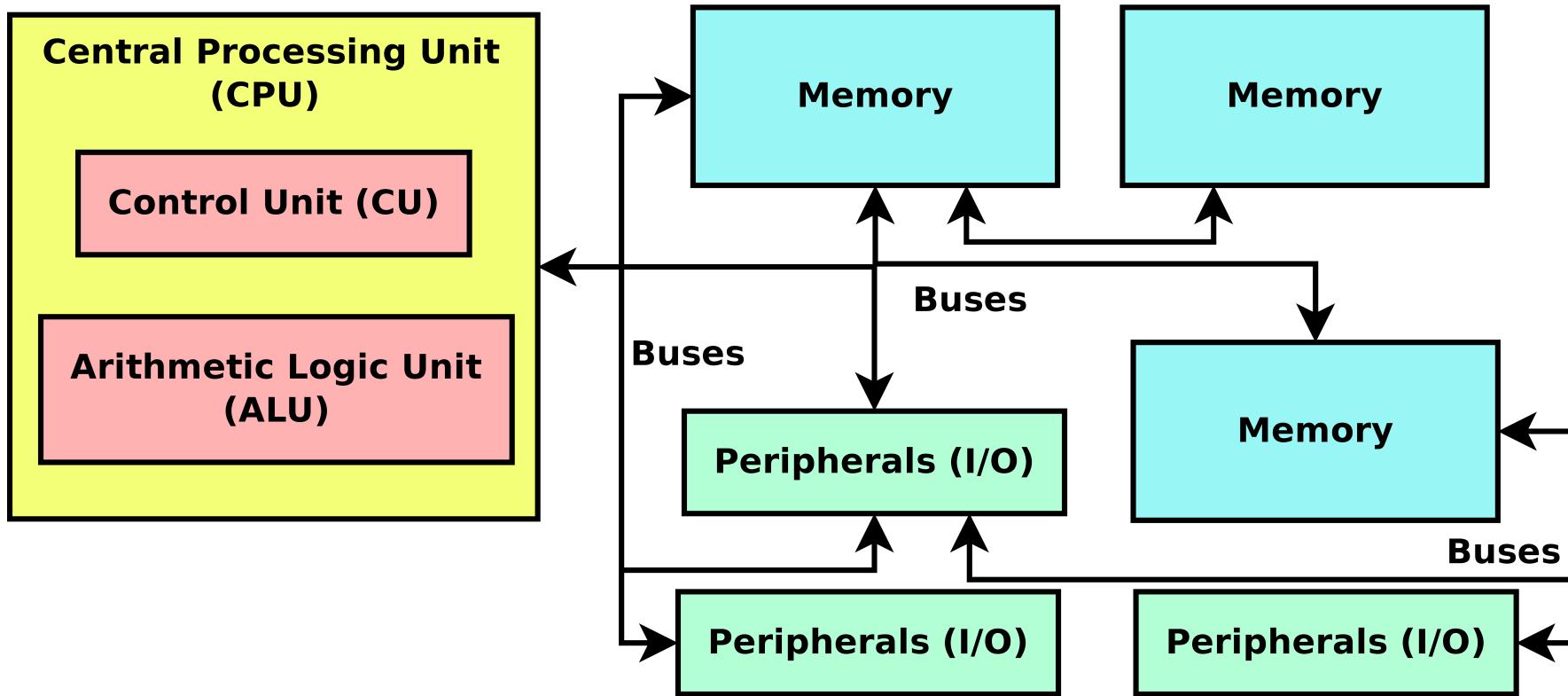
# A computer is ...



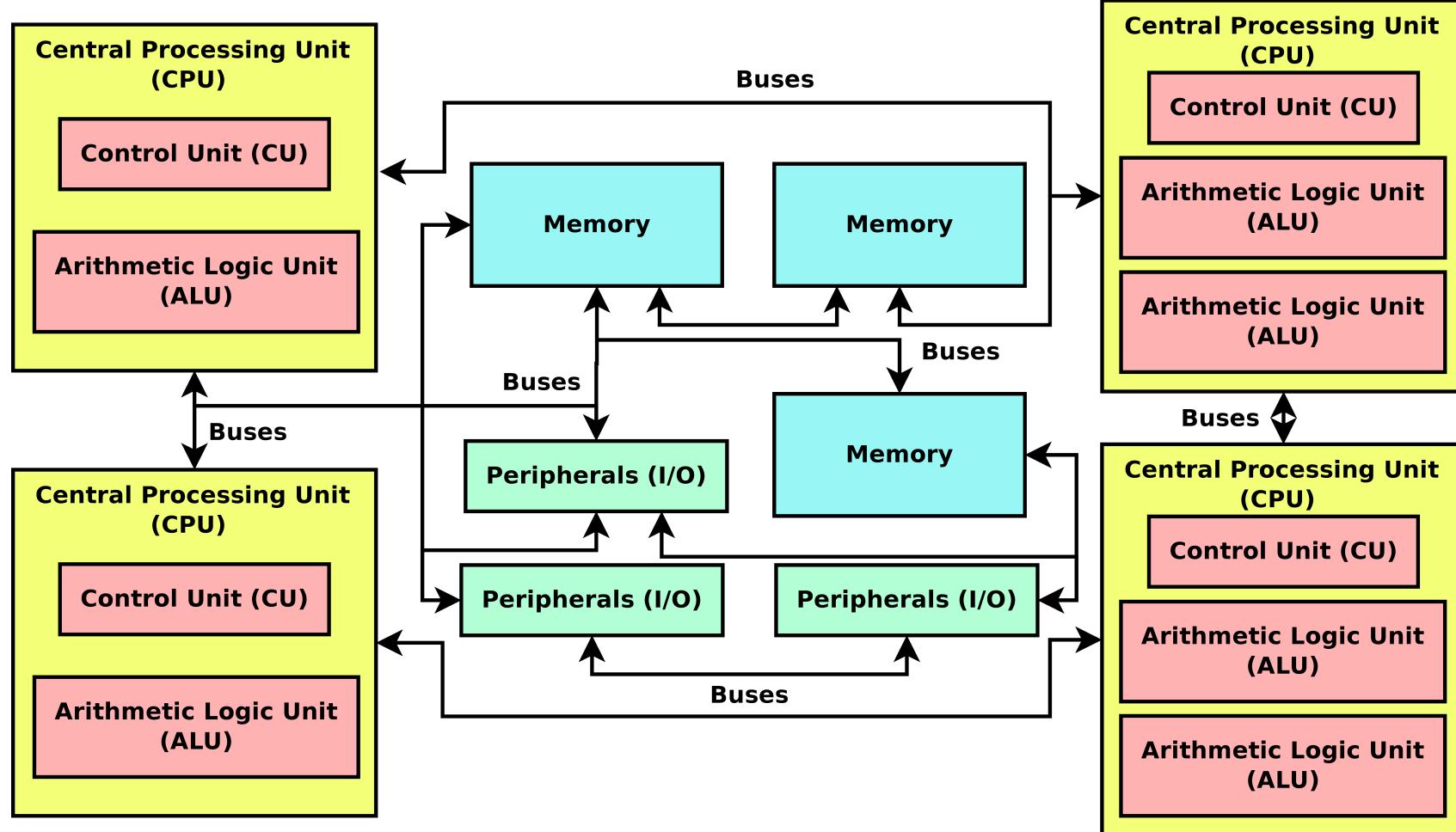
# A larger computer could be ...



# A larger computer could be ...

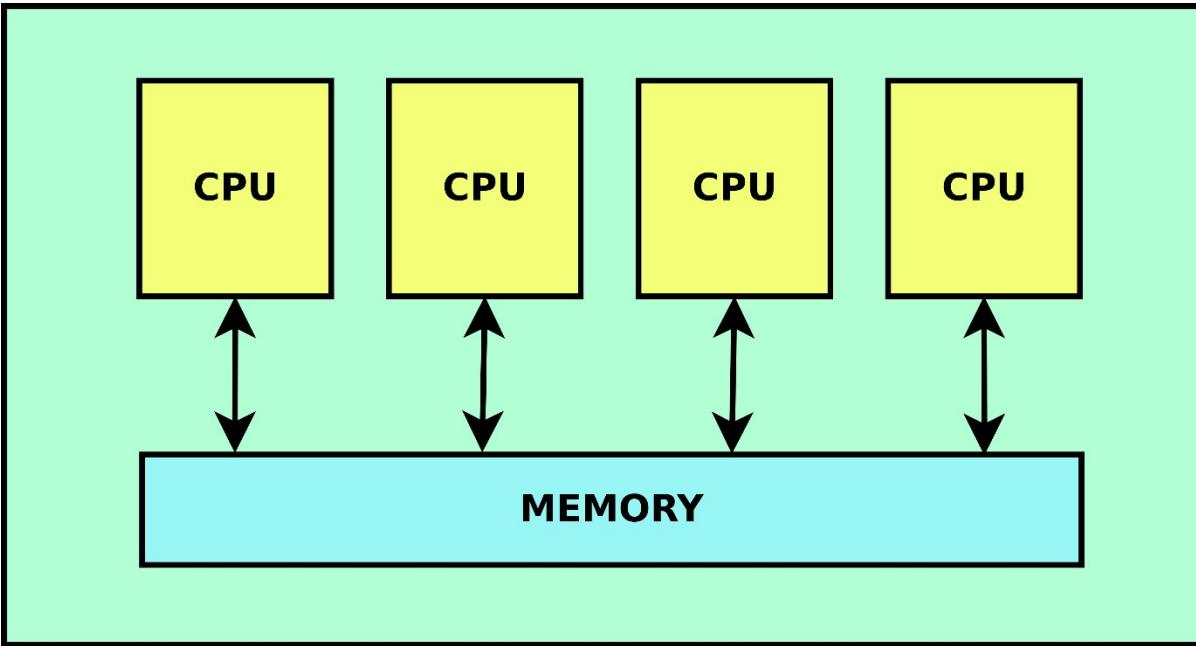


# A larger computer could be ...



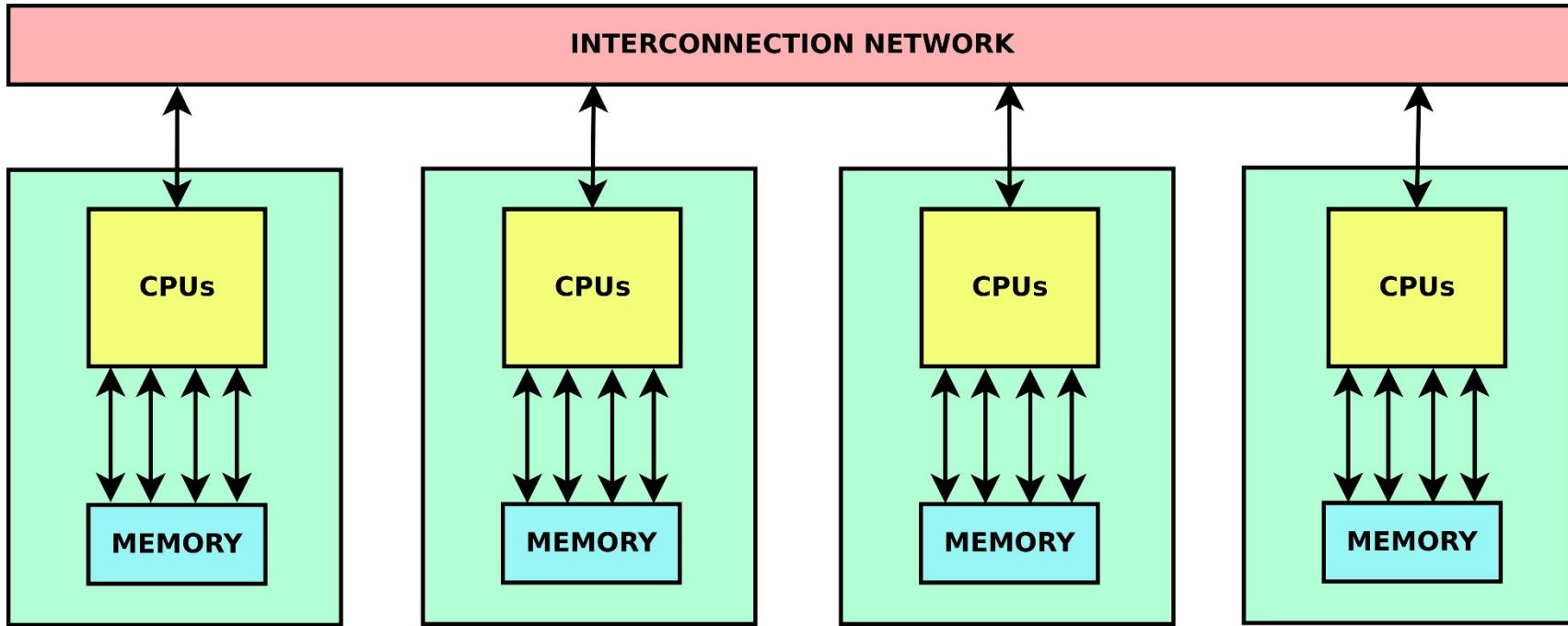
SURF

# A larger computer actually is ...

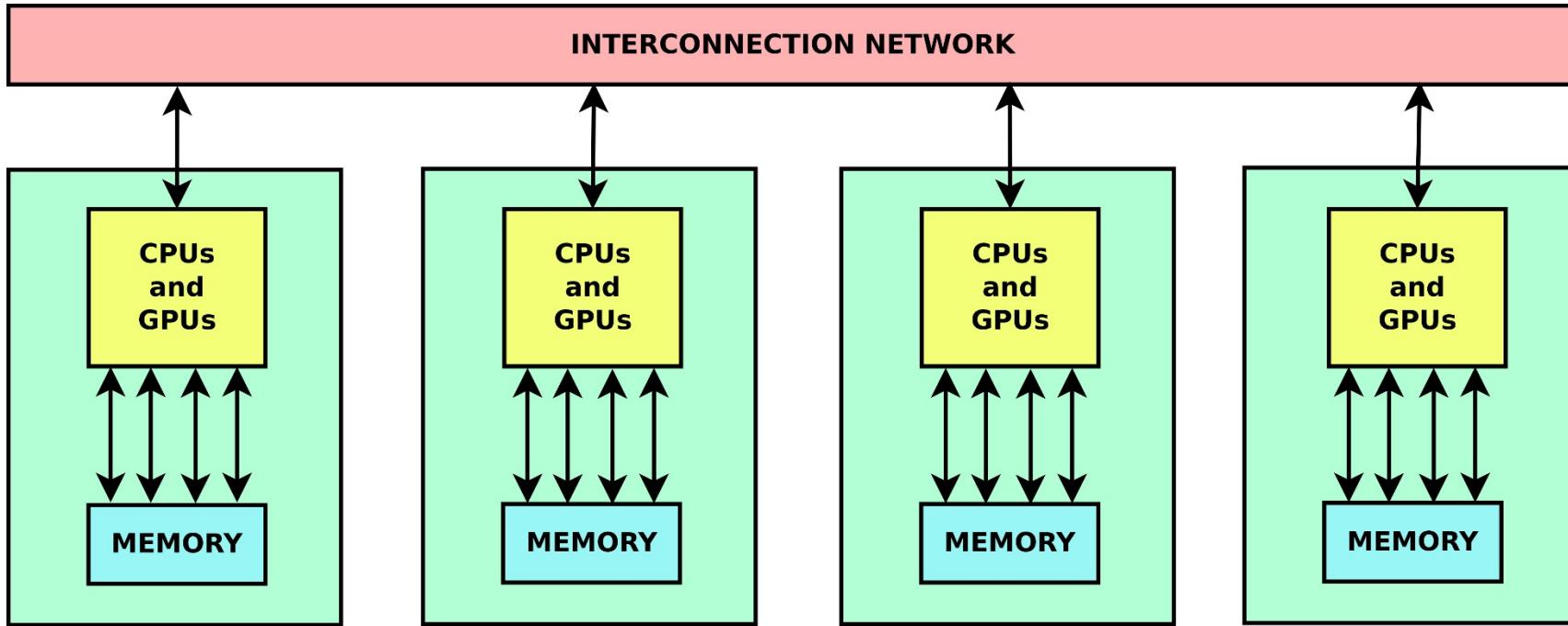


SURF

# A larger computer actually is ...



# A larger computer actually is ...



# Schematic overview of a supercomputer

A compute cluster is a group of tightly couple operating system instances that work together so closely that it can be seen as a single computer

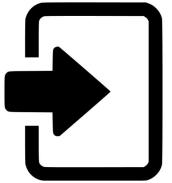


# Dutch national supercomputers: performance increase

| Year | Machine                            | R <sub>peak</sub> (GFlop/s) | kW    | GFlop/s/ kW |
|------|------------------------------------|-----------------------------|-------|-------------|
| 1984 | CDC Cyber 205 1-pipe               | 0.1                         | 250   | 0.0004      |
| 1988 | CDC Cyber 205 2-pipe               | 0.2                         | 250   | 0.0008      |
| 1991 | Cray Y-MP/4128                     | 1.33                        | 200   | 0.0067      |
| 1994 | Cray C98/4256                      | 4                           | 300   | 0.0133      |
| 1997 | Cray C916/121024                   | 12                          | 500   | 0.024       |
| 2000 | SGI Origin 3800                    | 1,024                       | 300   | 3.4         |
| 2004 | SGI Origin 3800 +SGI Altix 3700    | 3,200                       | 500   | 6.4         |
| 2007 | IBM p575 Power5+                   | 14,592                      | 375   | 40          |
| 2008 | IBM p575 Power6                    | 62,566                      | 540   | 116         |
| 2009 | IBM p575 Power6                    | 64,973                      | 560   | 116         |
| 2013 | Bull bullx DLC                     | 250,000                     | 260   | 962         |
| 2014 | Bull bullx DLC                     | ~ 1,000,000                 | 520   | 1923        |
| 2017 | Bull bullx DLC + KNL               | ~ 1,840,000                 | 850   | 2168        |
| 2021 | Lenovo AMD (1 <sup>st</sup> phase) | ~ 6,100,000                 | 610   | 10000       |
| 2016 | Raspberry PI 3 (35 euro)           | 0.44                        | 0.004 | 110         |

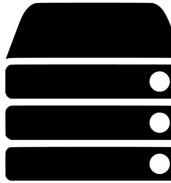


# Working with a Supercomputer



## Login node(s)

- Editing and transferring files
- Compile programs
- Prepare simulations



## Compute nodes

- Multicore nodes
- Large memories
- High-speed interconnections



## Batch scheduler

- Resource allocation
- Job queueing
- Accounting and

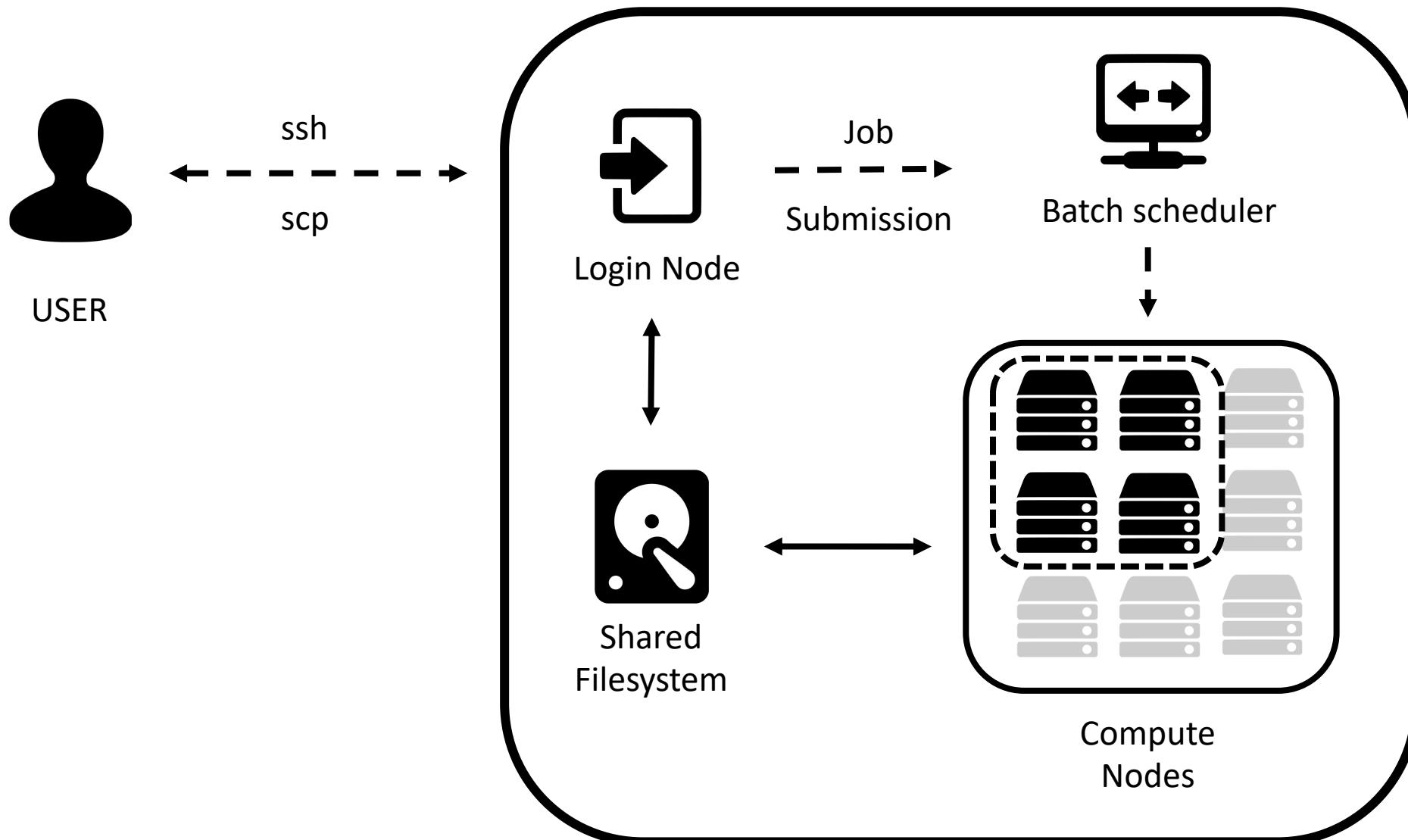


## File system

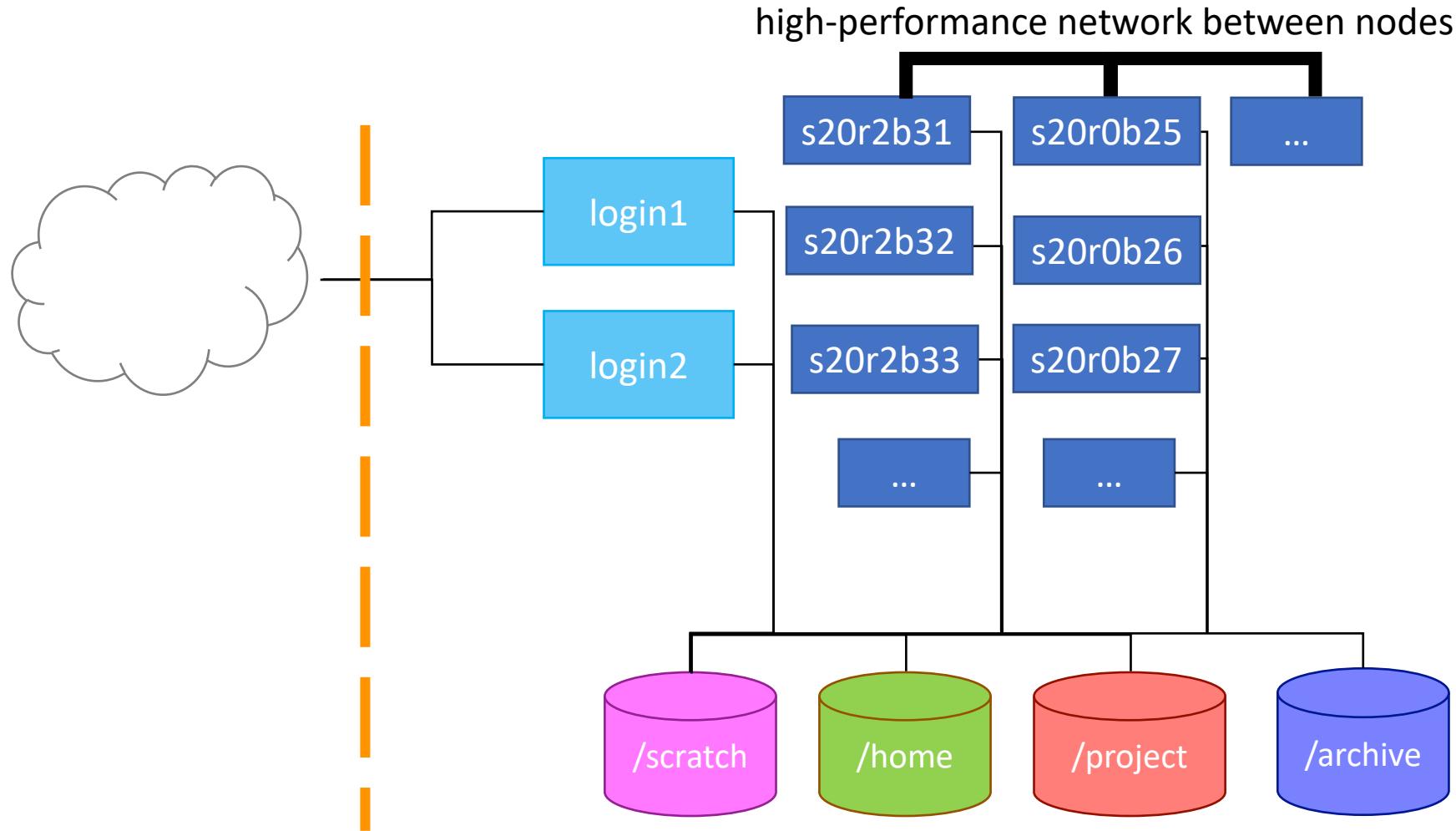
- Parallel FS
- Efficient I/O
- Node local disks



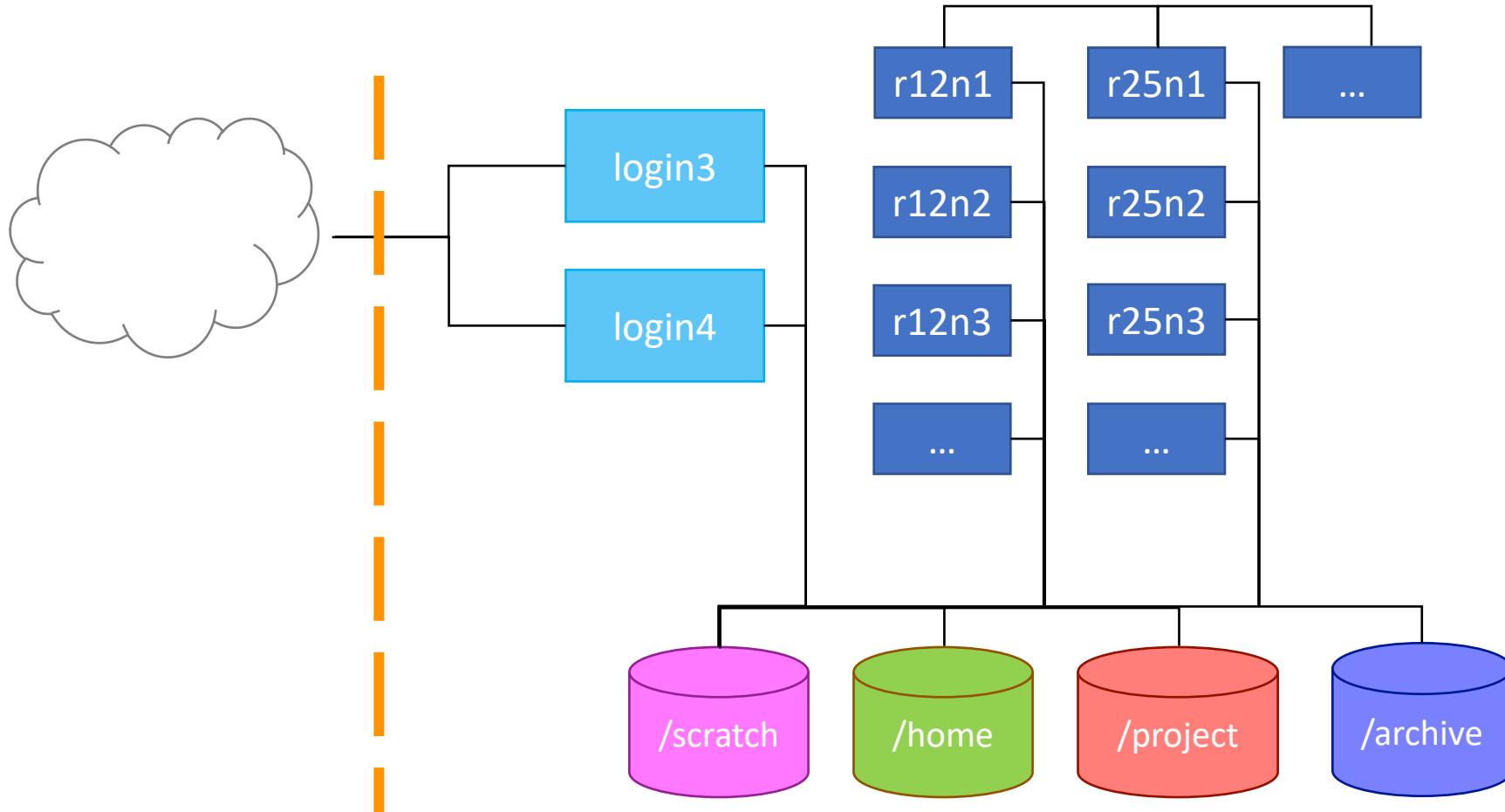
# Working with a Supercomputer



# Specific example: MareNostrum architecture



# Specific example: Lisa architecture



# Working with a Supercomputer

Is NOT like this....



SURF

# Working with a Supercomputer

SURF

# **INTRODUCTION TO LINUX**

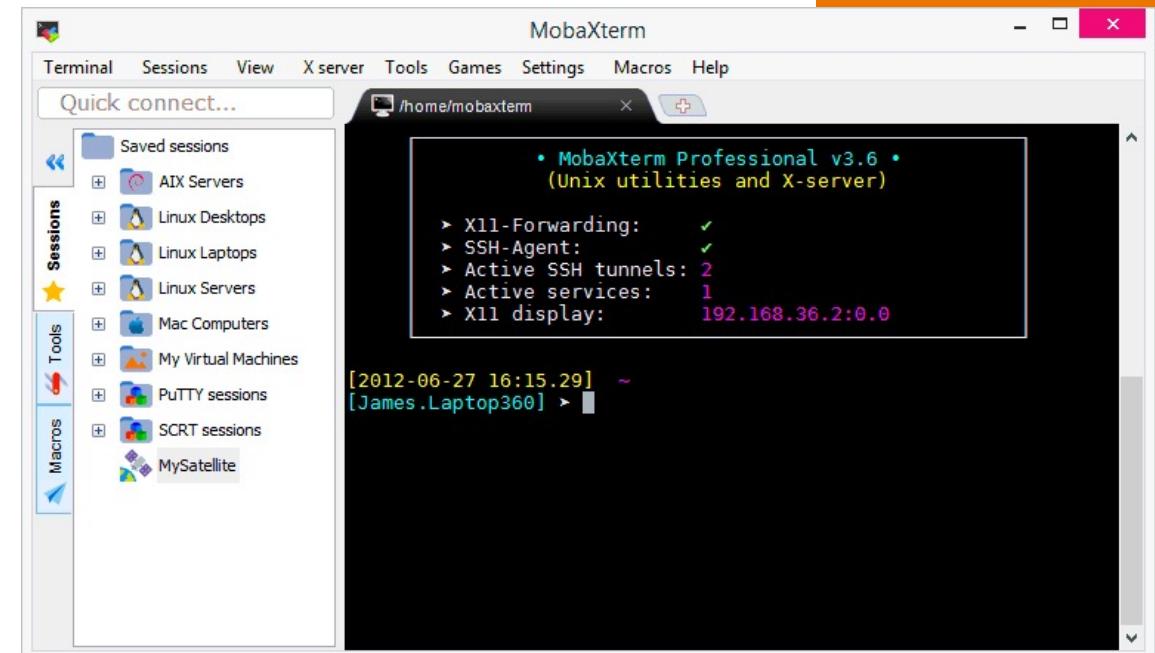
- a. ACCESSING THE SYSTEM (SSH)**
  
- b. BASIC COMMANDS AND USAGE**
  
- c. SHELL SCRIPTING**



# Working with a Supercomputer

## Install UNIX tools on your local machine

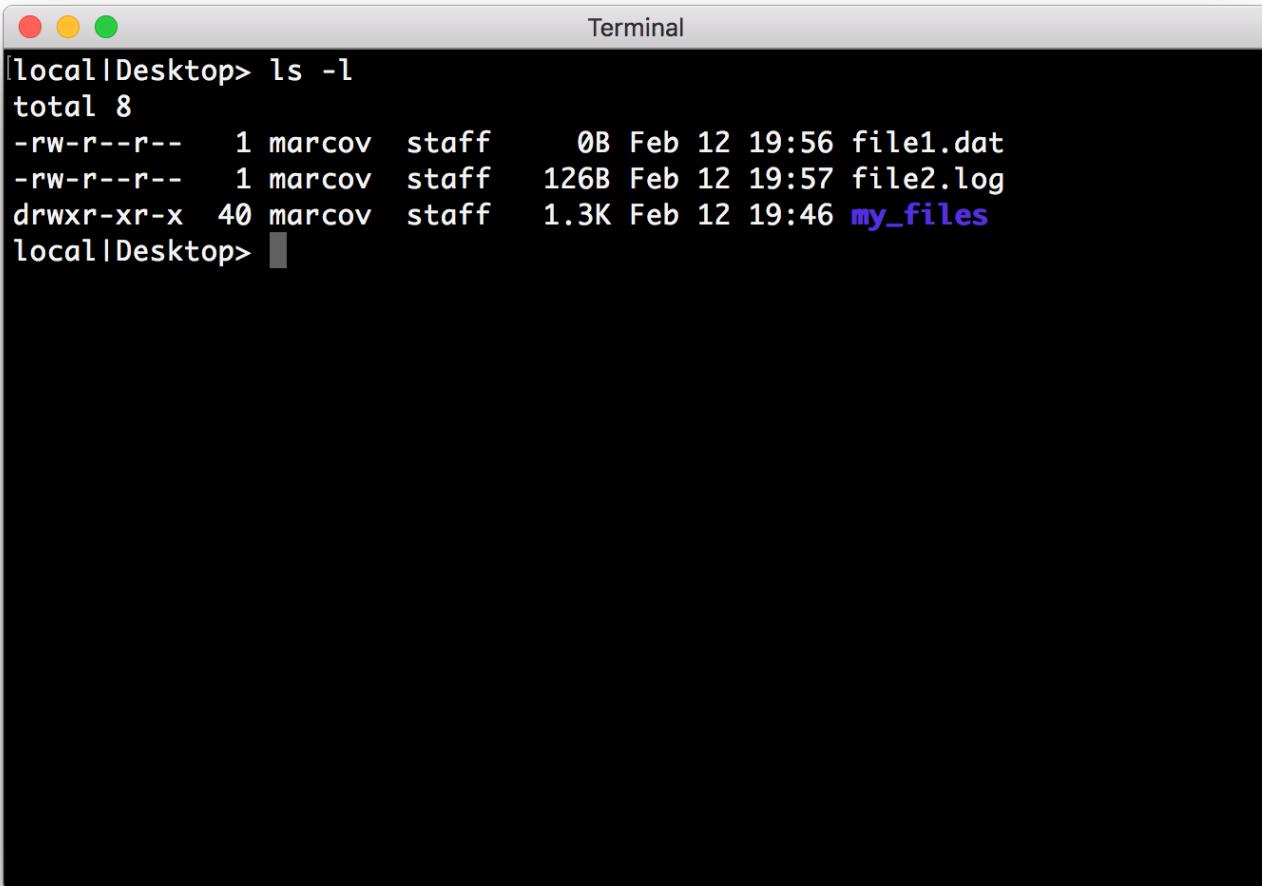
- Windows
  - Putty (<https://www.putty.org/>)
  - MobaXterm (<http://mobaxterm.mobatek.net>)
- Mac OSX
  - Terminal (pre-installed)
  - iTerm2 (<https://iterm2.com>)
- Linux
  - You are already well equipped!



*MobaXterm interface*

# Working with a Supercomputer

## The Terminal and the command line



```
[local|Desktop> ls -l
total 8
-rw-r--r--  1 marcov  staff      0B Feb 12 19:56 file1.dat
-rw-r--r--  1 marcov  staff   126B Feb 12 19:57 file2.log
drwxr-xr-x  40 marcov  staff  1.3K Feb 12 19:46 my_files
local|Desktop> ]
```

# Working with a Supercomputer

## SSH, or Secure SHell

- establishing a cryptographically secured connection
- authenticating each side to the other
- passing commands and output back and forth

```
$ ssh nct00021@mn1.bsc.es  
Password:
```

The SURF logo consists of the word "SURF" in a bold, black, sans-serif font, positioned within a white speech bubble shape with a black outline.

SURF

# Working with a Supercomputer

## SSH, or Secure SHell

- establishing a cryptographically secured connection
- authenticating each side to the other
- passing commands and output back and forth

```
$ ssh nct00021@mn1.bsc.es  
Password:
```

username

MareNostrum login node

# Working with a Supercomputer

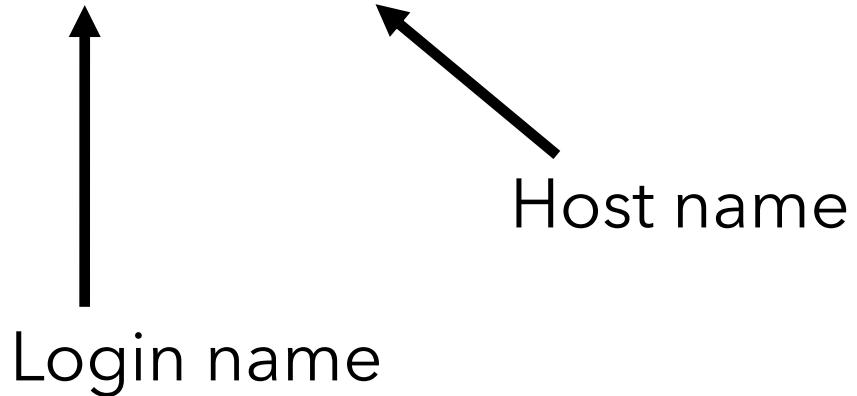
**You are logged in!!!**

# Introduction to Linux



- After successful login

```
nct00021@login1:~>
```



- Now the system is ready to accept commands



# Introduction to Linux - Exercise

- Explore the file system
  - ls, cd, pwd
- Create and edit files and directories
  - mkdir, touch, cp, mv, nano, less, vi
- Use variable (env. variables)
  - \$USER, \$HOME
- System tools
  - history, find, zip, tar

```
nct00021@login1:~> cd VU_HPC-master
```

```
nct00021@login1:~> ls  
bin file1.txt file2.log
```

```
nct00021@login1:~> nano README.md
```

```
nct00021@login1:~> echo $USER
```

```
nct00021@login1:~> history
```

# Introduction to Linux

- Structure of a Linux command

```
nct00021@login1:~>mkdir -p dir1/subdir
```

## Command

The UNIX shell (bash) tries to find a program called 'mkdir' and takes care that the system executes it.

## Options

Passed to the command as a parameter(s) to change its default behavior.

## Arguments

Taken as input by the program.

- Case-sensitive (everything!)
- Spaces used to separate command, options and arguments

# Introduction to Linux



- First command

```
nct00021@login1:~> date  
Thu Nov 14 11:00:00 CET 2019  
nct00021@login1:~>
```

- A little bit more

```
nct00021@login1:~> uname -a  
Linux login1 4.4.120-92.70-default #1 SMP Wed Mar 14 15:59:43 UTC  
2018 (52a83de) x86_64 x86_64 x86_64 GNU/Linux  
nct00021@login1:~>
```

# Introduction to Linux



- Where to find help?

“*--help*” flag

```
nct00021@login1:~> uname --help
```

“*man*” built-in command

```
nct00021@login1:~> man uname
```

“*info*” built-in command

```
nct00021@login1:~> info uname
```

# Introduction to Linux



- Where I am?

```
nct00021@login1:~> pwd  
/home/nct00/nct00021
```

- What files are there?

```
nct00021@login1:~> ls  
bin  file1.txt  file2.txt
```

```
nct00021@login1:~> ls -l  
total 1  
drwxr-xr-x 2 nct00021 nct00 4096 May  5  2010 bin  
-rw-r--r-- 1 nct00021 nct00      0 Feb 11 15:24 file1.txt  
-rw-r--r-- 1 nct00021 nct00      0 Feb 11 15:24 file2.txt
```

# Introduction to Linux



- Create directory

```
nct00021@login1:~> mkdir dir  
nct00021@login1:~> mkdir -p dir2/subdir
```

- Remove directory

```
nct00021@login1:~> rmdir dir
```

- Remove directory and its content (be careful!)

```
nct00021@login1:~> rm -r dir2
```

# Introduction to Linux



- Change directory

```
nct00021@login1:~> cd dir/subdir  
nct00021@login1:~/dir/subdir>
```

- Going "one level" up

```
nct00021@login1:~/dir/subdir> cd ..  
nct00021@login1 :~/dir/>
```

- Going back to home folder (~)

```
nct00021@login1:~/dir/> cd  
nct00021@login1:~>
```

# Introduction to Linux



- Relative path

```
nct00021@login1:~> ls dir/mydata.out  
dir/mydata.out
```

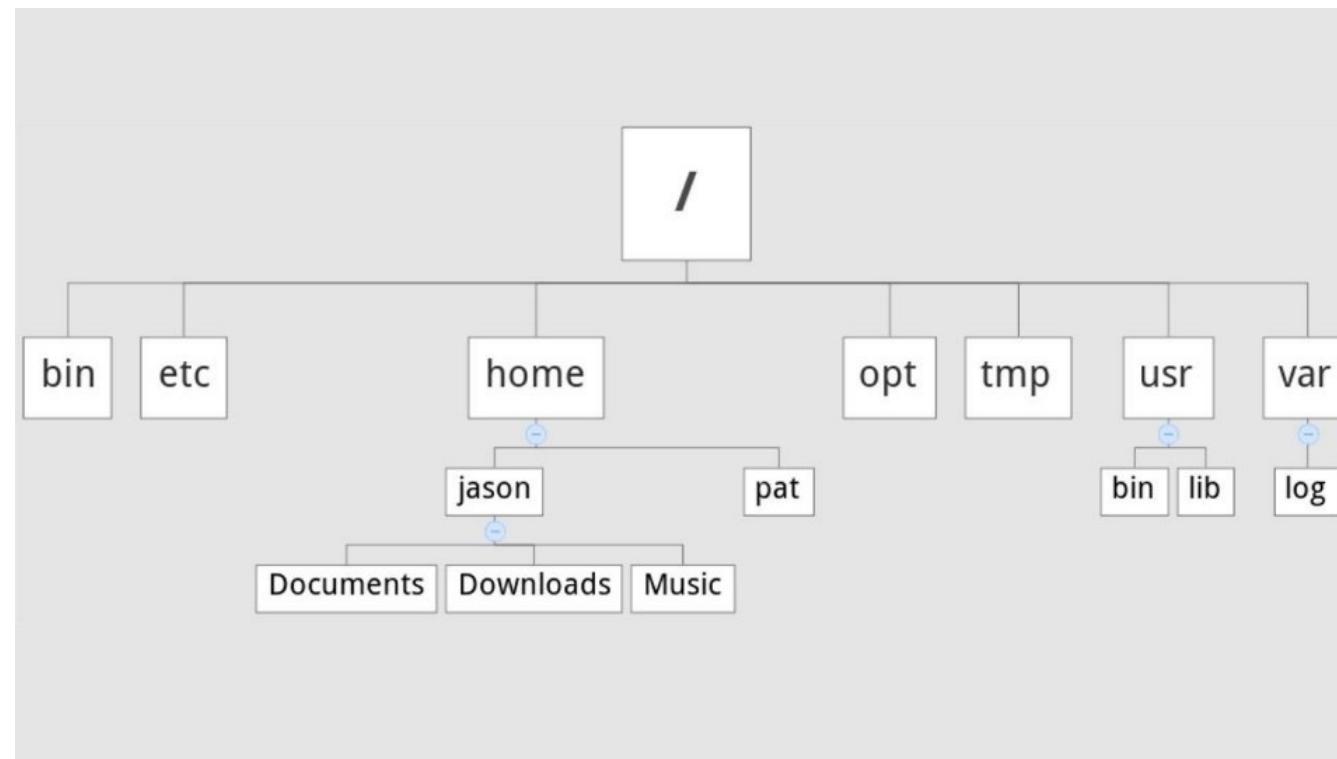
- Absolute path

```
nct00021@login1:~> ls ~/dir/mydata.out  
/home/nct00021/dir/mydata.out
```

# Introduction to Linux



- Directory Structure and Hierarchy



# Introduction to Linux



- Copy files

```
nct00021@login1:~> cp file1.txt file1.copy
```

- Copy files into directories

```
nct00021@login1:~> cp file1.* dir/  
nct00021@login1:~> cp file1.txt dir/file1.copy2
```

- Move/Rename files

```
nct00021@login1:~> mv file1.txt dir/file.txt.new  
nct00021@login1:~> mv file1.txt file.txt.old
```

# Introduction to Linux



- Copy files to remote host

```
benjamic@local:~> scp file1.txt nct00021@mn1.bsc.es:
```

- Copy files from remote host to local machine

```
benjamic@local:~> scp nct00021@mn1.bsc.es:file1.txt .
```

- Copy folder from your local machine to a remote host

```
benjamic@local:~> scp -r Intro-to-HPC nct00021@mn1.bsc.es:
```

# Introduction to Linux - Exercise

- Download the material from the website:

<https://github.com/sara-nl/Intro-to-HPC>

- Transfer the zip file to your account on MareNostrum:

```
~> scp main.zip nct00021@mn1.bsc.es:
```

- #### Extract zip on the host:

~> ssh nct00021@mn1.bsc.es

~> Password

```
~> unzip main.zip
```

- ## Clone the git repo

```
~>git clone https://github.com/sara-nl/Intro-to-HPC.git
```

# Introduction to Linux



- Extract zipped files

```
nct00021@mn1.bsc.es:~> unzip main.zip  
nct00021@mn1.bsc.es:~> tar -zxvf main.tar.gz
```

- Open formatted files (e.g.: text files)

```
nct00021@mn1.bsc.es:~> nano Intro-to-HPC/README.md  
nct00021@mn1.bsc.es:~> emacs Intro-to-HPC/README.md  
nct00021@mn1.bsc.es:~> vim Intro-to-HPC/README.md
```

# Introduction to Linux

## Shell script programming

What if I want to run many bash commands?

...maybe in a workflow?

### Bash scripts

- The “Shell” is the program which read commands and run other programs. Bash is a type of “Shell”.
- A *bash script* is a plain text file which contains a series of commands
- Any command you can run on the command line can be put into a script (v.v.)
- It will be executed like a normal program: `./script.sh`

# Introduction to Linux

## Shell script programming

- Loops for, while, until
- Conditional statements: If, else, then
- Functions
- Variables
- and much more

A screenshot of a Google search results page for the query "bash programming". The search bar at the top contains the text "bash programming". Below the search bar, there are navigation links for "All", "Videos", "Images", "News", "Shopping", "More", "Settings", and "Tools". A message indicates "About 38,500,000 results (0.66 seconds)". The first result is a link to "BASH Programming - Introduction HOW-TO" from tldp.org, dated Jul 27, 2017. The snippet describes it as an introduction to basic-intermediate shell scripts. Below the result, a "People also search for" box lists related terms like "bash programming pdf", "simple bash script example", and "print in bash script". To the right, a "People also ask" box has three collapsed questions: "What is bash programming?", "What does \$() mean in bash?", and "What is bash useful?".

[Bash Scripting: Everything you need to know about Bash-shell ...](#)  
https://itnext.io › bash-scripting-everything-you-need-to-know-about-bash... ▾  
Sep 10, 2019 - In this article, we are going to cover almost every single topic there is in Bash programming. This articles mainly focus on programming spec ...

[Shell programming with bash: by example, by counter-example](#)  
matt.might.net › articles › bash-by-example ▾  
As an interactive shell, bash is a terse language for initiating and directing computations. As a scripting language, bash is a domain-specific language for ...

[Bash scripting cheatsheet - Devhints](#)  
https://devhints.io › bash ▾  
Variables · Functions · Interpolation · Brace expansions · Loops · Conditional execution · Command substitution · One-page guide to Bash scripting.

[Understanding Bash: Elements of Programming | Linux Journal](#)  
https://www.linuxjournal.com › content › understanding-bash-elements-pr... ▾  
Sep 28, 2018 - Ever wondered why programming in Bash is so difficult? Bash employs the same constructs as traditional programming languages; however, ...

# **RUNNING JOBS ON THE BATCH SYSTEM**

- a. INTERACT WITH THE BATCH SYSTEM**
  
- b. RUN A REAL SCIENTIFIC WORK FLOW**

# Module management: useful commands

- module avail - available modules in the system
- module load <mod> - load <mod> in the shell environment
- module list - show a list of all loaded modules
- module unload <mod> - remove <mod> from the environment
- module purge - unload all modules
- module whatis <mod> - show information about <mod>



# Running jobs on the HPC system

- Let's try it the module command
- Get which version of Python is currently available

```
nct00021@mn1.bsc.es:~> gcc --version  
nct00021@mn1.bsc.es:~> which gcc
```

- Try to load a different version of Python

```
nct00021@mn1.bsc.es:~> module load intel/2017.4  
nct00021@mn1.bsc.es:~> module load gcc/11.2.0
```

- Check again.

# Running jobs on the HPC system

## The batch jobs scheduler

- Supercomputers use job schedulers to distribute computational tasks over the available nodes.
- Instead of executing commands interactively, you prepare a job script
  - Script containing the commands to execute
  - Resource characteristics (specific)
- The batch system is responsible for allocating cores, processors or nodes to a job.



# Running jobs on the HPC system

## The batch jobs scheduler

- It allows to run MANY jobs at the same time
  - The system takes care that they are run efficiently on the available resources.
- Multiusers, queue system
  - A batch system allows users to always submit jobs, even if a lot of people are using the system at the same time. In addition take care of budgeting and fair resource usage.
- System load balance
  - The system takes care of balancing the load across nodes and during time. In a batch system, most jobs may be submitted during office hours, but the scheduler will continue to start jobs at night as nodes become available.



# Running jobs: useful commands of the SLURM scheduler

- `sbatch <jobscript>`
  - submit a job to the scheduler
- `squeue -j <job_id>`
  - inspect the status of job `<job_id>`
- `squeue -u <user_id>`
  - inspect all jobs of user `<user_id>`
- `scancel <job_id>`
  - cancel job `<job_id>` before it runs
- `scontrol show job <job_id>`
  - show estimated job start



# Anatomy of a job script

- Job scripts consist of:

- the “shebang” line: #!/bin/bash
- scheduler directives
- command(s) that load software modules and set the environment
- command(s) to prepare the input
- command(s) that run your main task(s)
- command(s) to save your output

```
#!/bin/bash
```

```
#SBATCH --job-name="firsttest"  
#SBATCH --nodes=1  
#SBATCH --ntasks=10  
#SBATCH --time=00:01:00  
#SBATCH --partition=normal
```

```
module load 2021  
module load foss/2021a
```

```
cp -r <my_folder> $TMPDIR  
cd $TMPDIR
```

```
srun a.out
```

```
cp -r $TMPDIR/* ~/results
```

# Introduction to Linux



- Submit job to the queue:

```
nct00021@login1:~> sbatch <job script>
```

- Show running and queued jobs:

```
nct00021@login1:~> squeue -u $USER
```

- Remove a job from the queue or kill it if running:

```
nct00021@login1:~> scancel -u $USER
```



# Introduction to batch - Exercise

- Move into the exercise folder:  
batch/example1

- Inspect the file “slurm.sub”

- Run the script via bash

```
nct00021@login1:~> bash slurm.sub
```

- submit the script to the batch scheduler

```
nct00021@login1:~> sbatch slurm.sub
```

# Running jobs on the HPC system - Ex

- Move into the exercise folder:  
batch/example2
- Inspect the file “pythonopenmp.sub”

- Submit the job to the queue

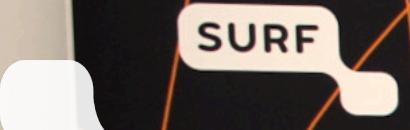
```
nct00021@login1:~> sbatch pythonopenmp.sub
```

- Analyse the results.

# Running jobs: best practices

- Give the scheduler a realistic *walltime* estimate
- Your home directory is slow. Use `$TMPDIR`.
- Load software modules as part of your job script – this improves reproducibility
- Run parallel versions of your programs (and use “`srun`” to ask SLURM to run multi-process applications)

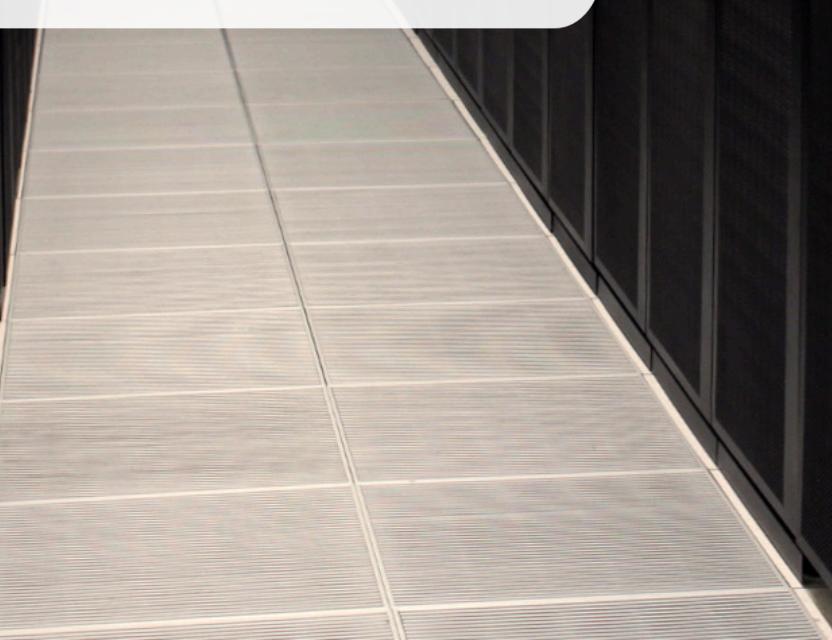




SURF

# Thank you!

Benjamin Czaja, Marco Verdicchio  
HPC Advisors, SURF



# Download the portable edition of MobaXterm

The screenshot shows a web browser window with the URL [mobaxterm.mobatek.net/download-home-edition.html](http://mobaxterm.mobatek.net/download-home-edition.html) highlighted with a red box. The page content is as follows:

MobaXterm Home Edition

Download MobaXterm Home Edition (current version):

[MobaXterm Home Edition v20.1 \(Portable edition\)](#) (highlighted with a red box)

[MobaXterm Home Edition v20.1 \(Installer edition\)](#)

Download previous stable version: [MobaXterm Portable v12.4](#) [MobaXterm Installer v12.4](#)

You can also get early access to the latest features and improvements by downloading MobaXterm Preview version:

[MobaXterm Preview Version](#)

By downloading MobaXterm software, you accept [MobaXterm terms and conditions](#)

You can download MobaXterm and plugins sources [here](#)

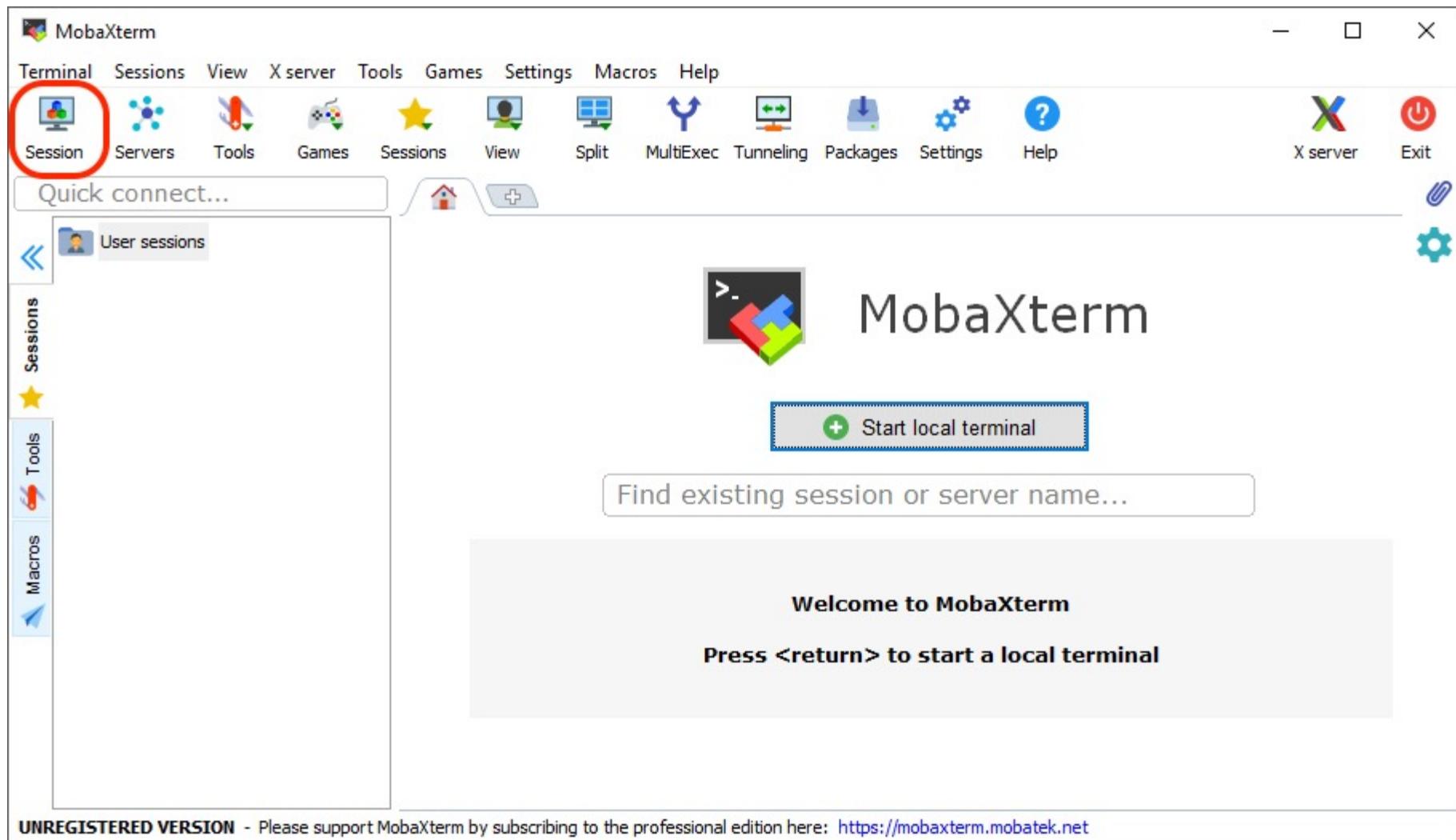
**i** If you use MobaXterm inside your company, you should consider subscribing to [MobaXterm Professional Edition](#): your subscription will give you access to professional support and to the "Customizer" software. This customizer will allow you to generate personalized versions of MobaXterm including your own logo, your default settings and your welcome message.  
Please [contact us](#) for more information.

Changelog

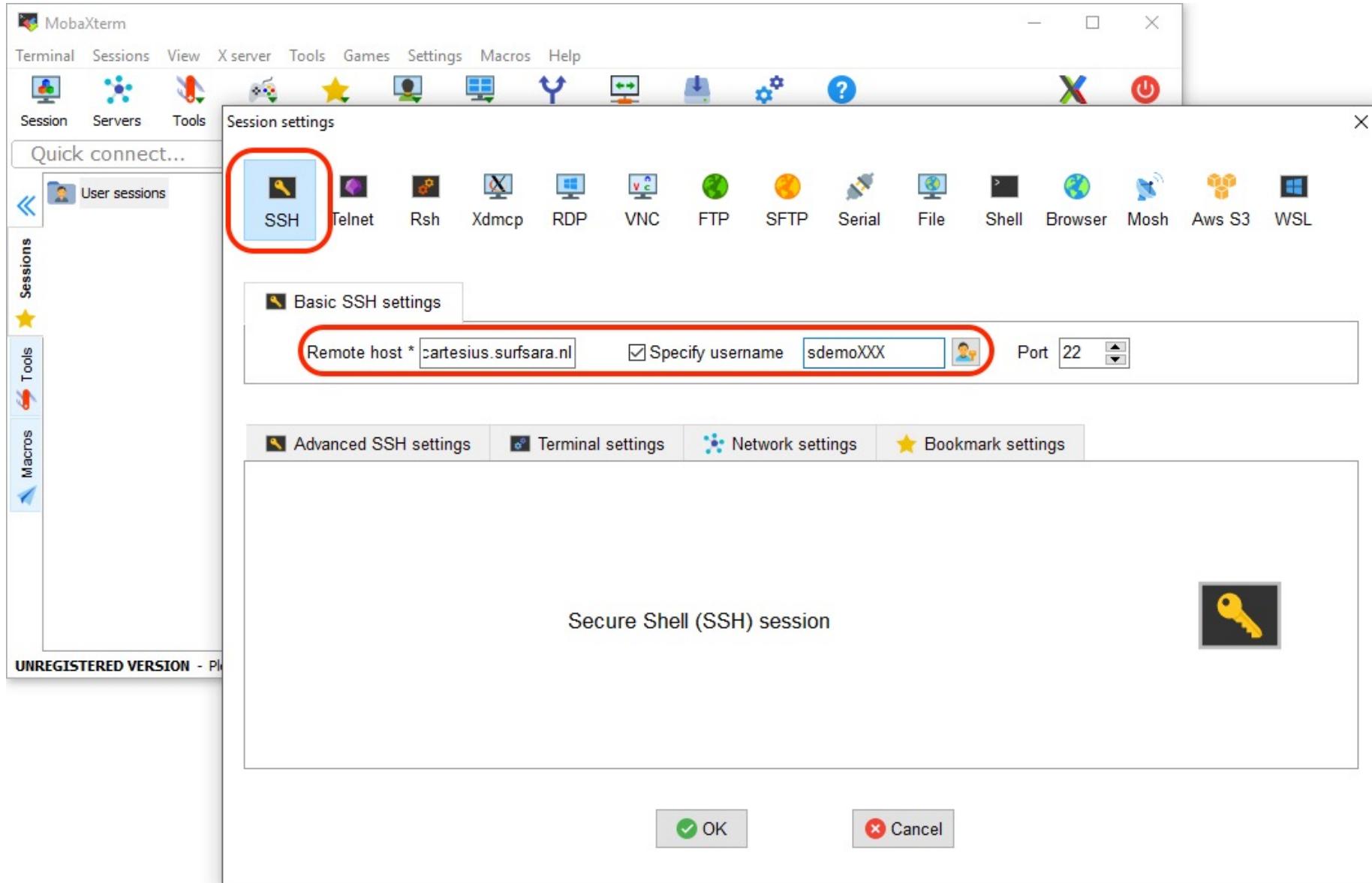
Version 20.1 (2020-02-28)

- **Improvement:** fixed some slowness issues when a stored SSH key is located on a network share that no more exists
- **Improvement:** SSH connections to inconsistent servers that send improper SSH signature are tried in SSHv2 by default

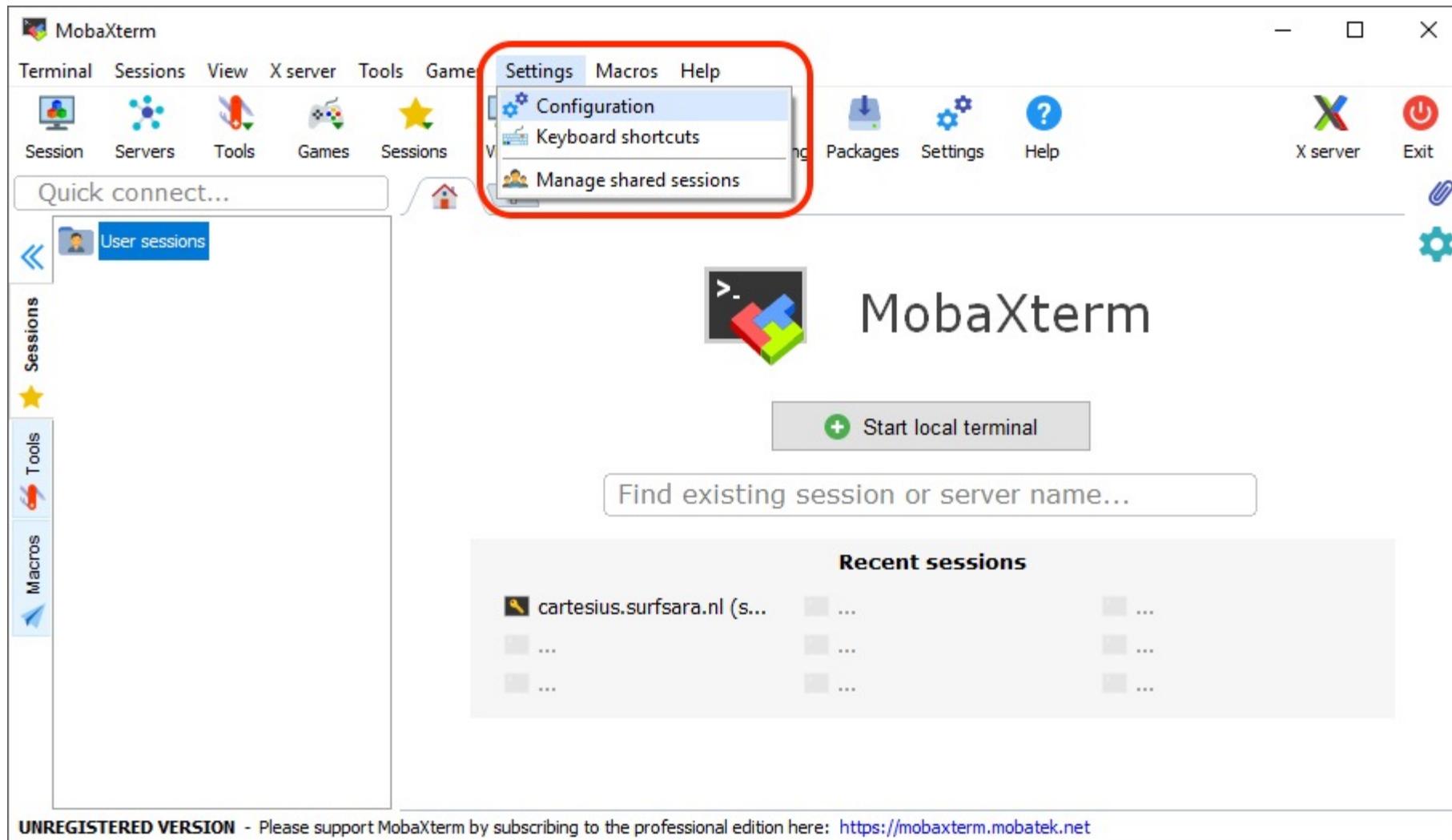
# Open a new remote session



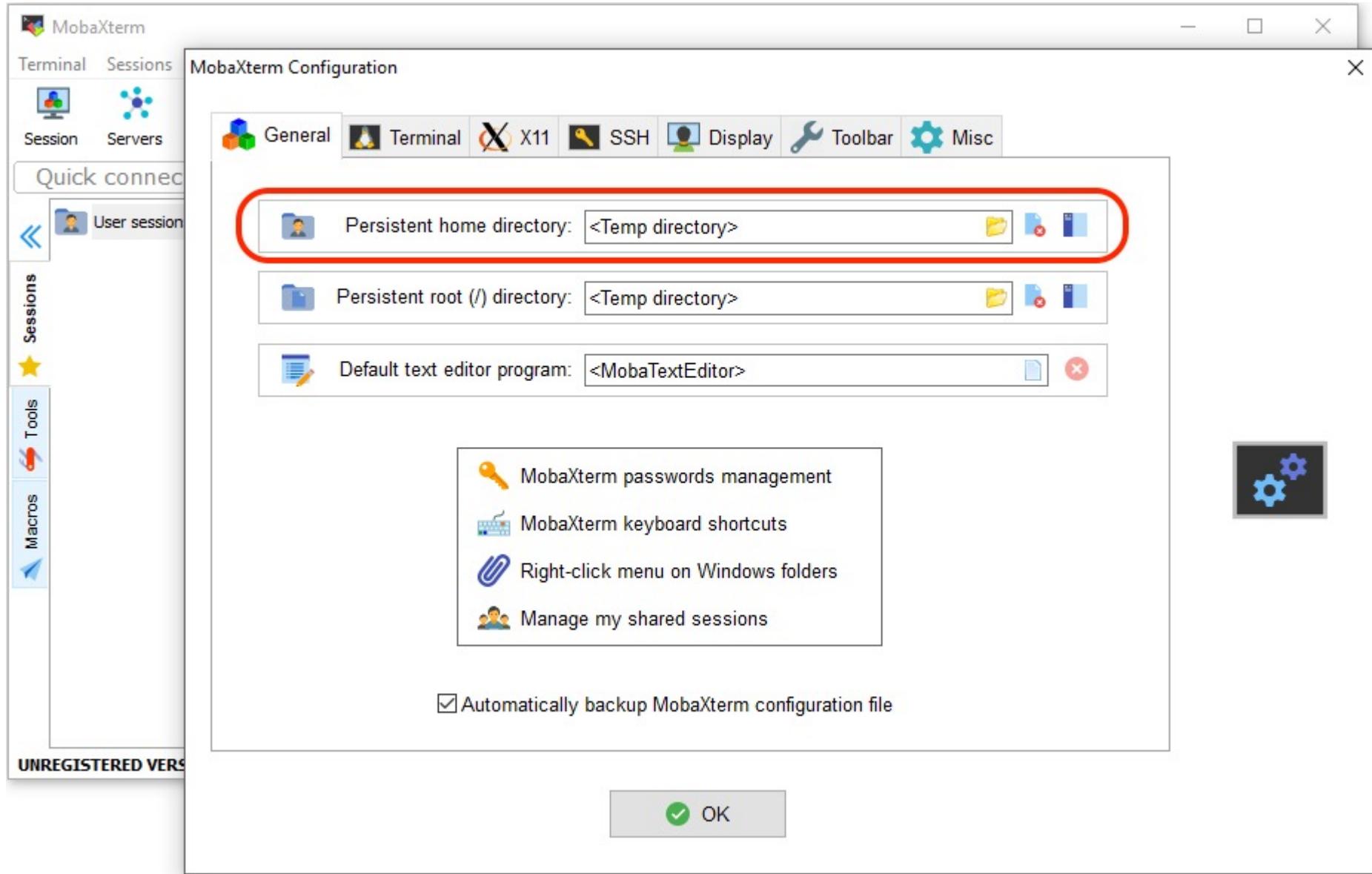
# Open a new remote session



# Change the root/home folders (via Settings->Configuration)



# Change the root/home folders (via Settings->Configuration)



# Introduction to Linux

## Shell script programming

```
lcur0001@login4:~> ls -l
total 0
drwxr-xr-x 2 lcur0001 lcur0001 4096 May  5  2010 bin
-rw-r--r-- 1 lcur0001 lcur0001   10 Feb 13 01:17 source.c
-rwxr-xr-x 1 lcur0001 lcur0001 1528 Feb 13 01:17 main.x
```

## File permissions

Every file/directory has 9 permission bits associated (+ initial to diff):

- 3 user based permission groups: owner(2-4), group(5-7), all users(8-10)
- x3 permission types: read(r), write(w), execute(x)

Permission can be changed with the command “chmod”

# Introduction to Linux - Exercise

- Move into the exercise folder:  
linux-cluster-computing/introlinux
- Inspect the “simple.sh” file
- Change permission of the script to allow execution  

```
lcuur0001@login4:~> chmod +x simple.sh
```
- Run the script

```
lcuur0001@login4:~> ./simple.sh
```

# Introduction to Linux - Exercise

- Move into the exercise folder:  
linux/example2
- Inspect the file “compute\_factorial.sh”  
and the folder “src”
- Change permission of the script to allow execution

```
lcur0001@login4:~> chmod +x compute_factorial.sh
```

- Run the script

```
lcur0001@login4:~> ./compute_factorial.sh
```