Introduction to Atos HPCF and ECS

Xavier Abellan

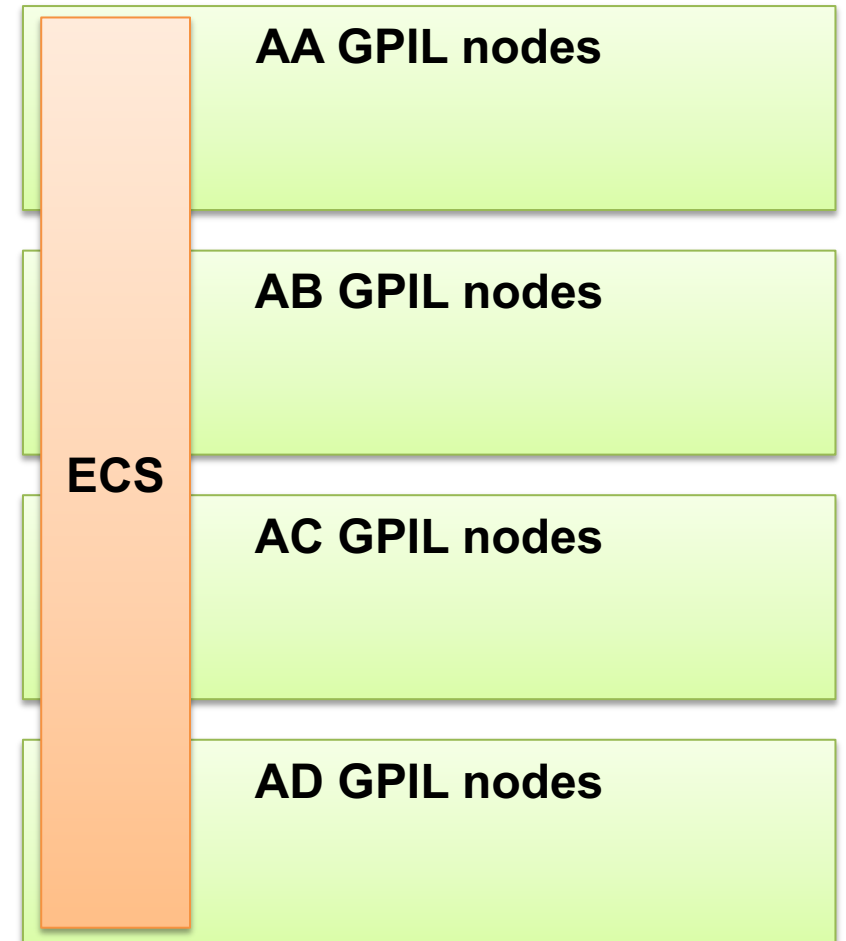xavier.abellan@ecmwf.int

ECMWF

ECMWF Atos HPCF

- 4 Complexes
  - Two in each Computer Hall
  - Each Complex consists of two partitions:
    - Parallel:
      - 1920 nodes for parallel compute
      - exclusive access (only one job in a node)
    - General Purpose (GPIL)
      - 112 nodes for general purpose use
        » More memory, local SSD
      - Shared access (many jobs in the same node)
- One Slurm scheduler in each complex
- Same Filesystems and applications everywhere

AA

AB

AC

AD

Parallel nodes

GPIL nodes

GPIL nodes

| *f queues | Login nodes | Interactive jobs |
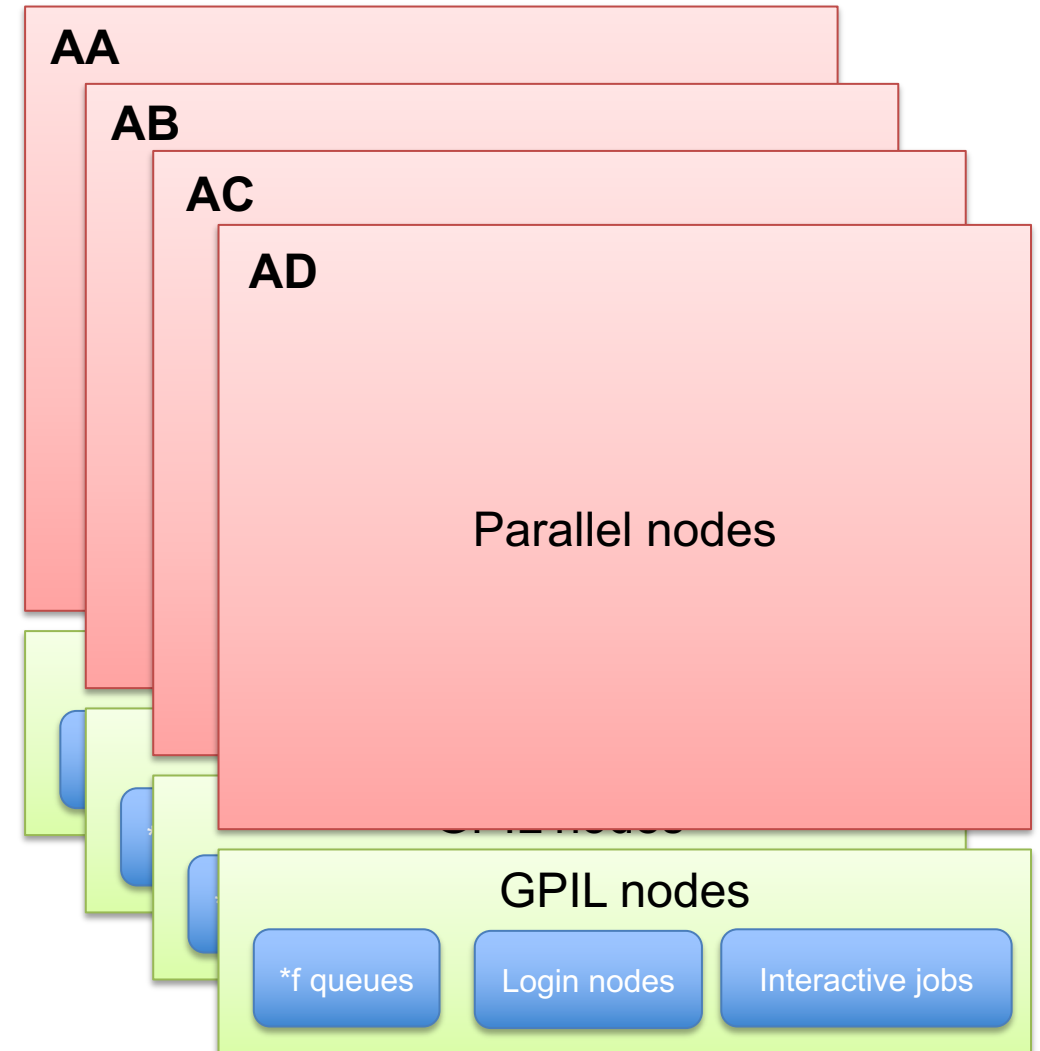
ECS – ECGATE Class Service

- For users with no formal HPCF access

  - Including those from Cooperating States

- "Virtual" complex with nodes from 4 complexes

- Same Apps and Filesystems as main complexes

- **Independent Slurm Batch system**

- **Serial or very small parallel workloads**

- **No SBU billing**

ECS	AA GPIL nodes
	AB GPIL nodes
	AC GPIL nodes
	AD GPIL nodes

Where to start

https://confluence.ecmwf.int/x/UxhbDg

Which door to use?

* GPIL: General Purpose Interactive Login node
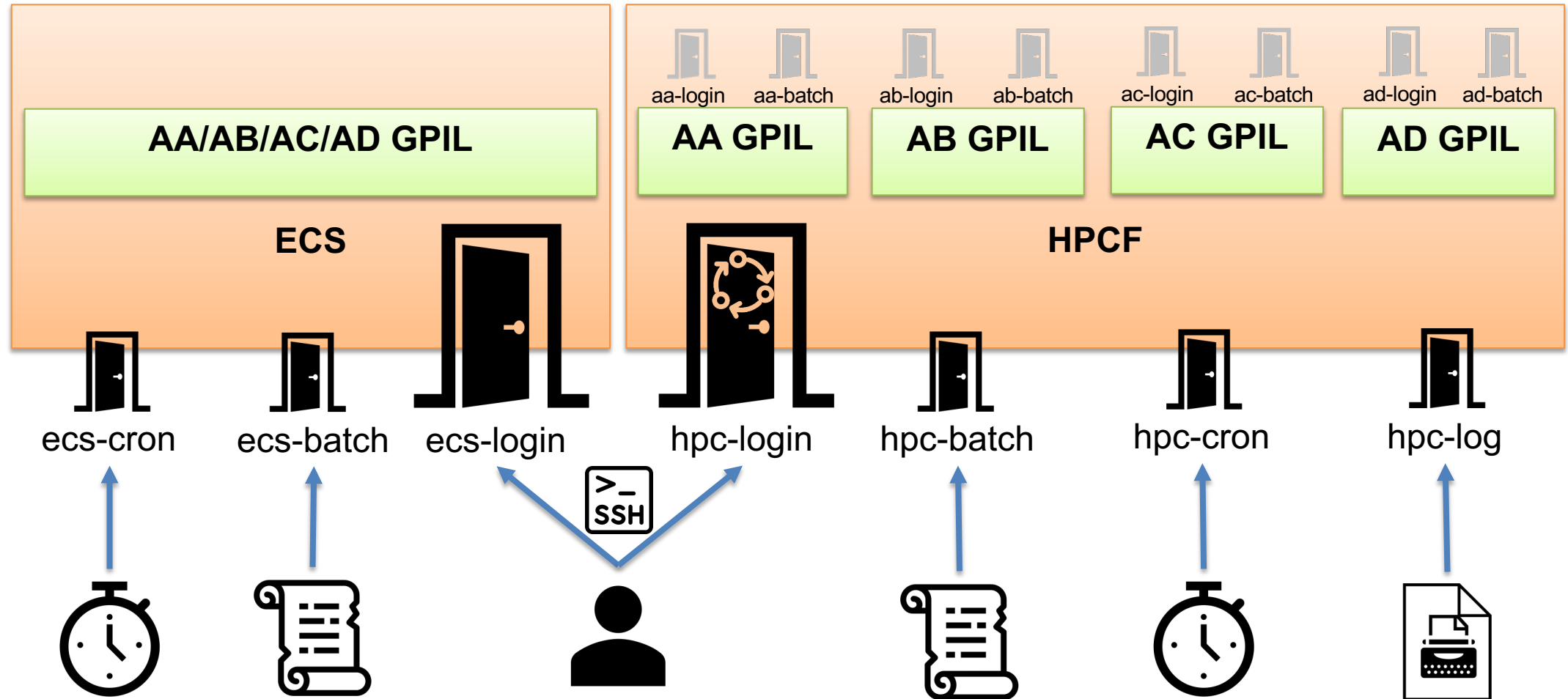
Shells

- Available login shells are **BASH** (default for new users) and **KSH**

  - No CSH support

- We can change your default shell: let us know via the [ECMWF Support Portal](#)

- You may customise your shell environment with the standard files:

  - ~/.profile and ~/.bash_profile

  - ~/.bashrc and ~/.kshrc

  - ~/.bash_logout

https://confluence.ecmwf.int/x/WxhbDg
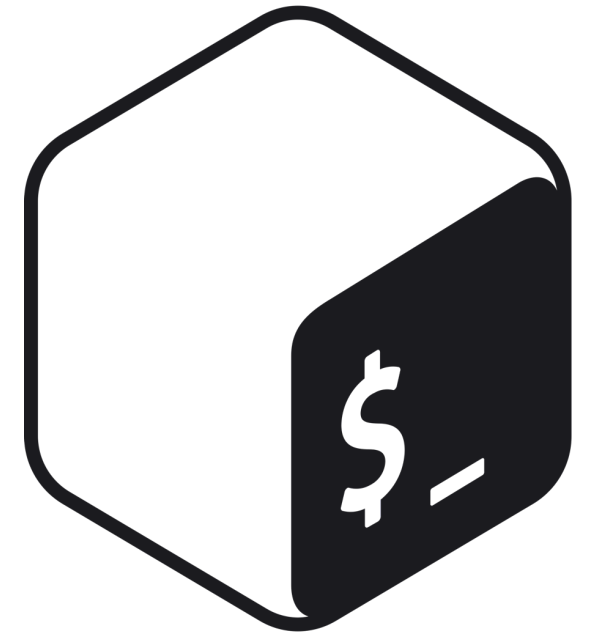
Storage

HOME — 10 GB

PERM — 500 GB

HPCPERM — 100 GB | 1TB

SCRATCH — 2 TB | 50 TB

ECFS

https://confluence.ecmwf.int/x/XBhbDg

Temporary Storage

TMPDIR
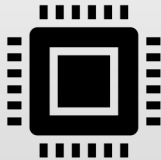
3 – 40 GB SSD
240 GB RAM

SCRATCHDIR

SCRATCH quota

**Memory
(/var)/tmp /dev/shm**

80% of node's memory

https://confluence.ecmwf.int/x/XBhbDg

File transfers

- Any tool using OpenSSH behind the scenes should work with teleport

    – Teleport should be configured appropriately if running from outside ECMWF
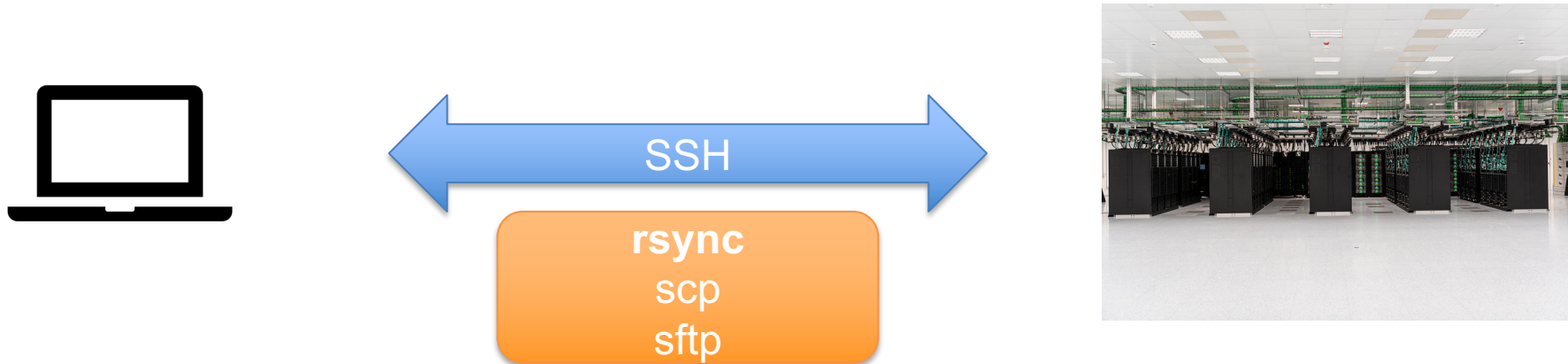

```
user@yourlaptop:~> rsync -avz mydataset hpc-login:/scratch/user/
user@yourlaptop:~> rsync -avz hpc-login:/scratch/user/mydataset .
```

- SFTP to boaccess.ecmwf.int

- ectrans may be used to push data from HPC to a remote site

- bbcp also possible for advanced use cases.

https://confluence.ecmwf.int/x/XRhbDg

Environment Modules

- Manage your software environment

- Over 400 modules available

- Basic commands:

  - module load / unload

  - module list

  - module avail / spider

  - module show

https://confluence.ecmwf.int/x/eA6UCg

```
[usxa@ac6-101 ~]$ module avail git

------------------------------------------ Global Aliases ------------------------------------------

------------------------------- /usr/local/apps/modulefiles/lmod/core -------------------------------
   git/2.26.0 (jun20:may20)    git/2.39.1 (D:may23:new)    gitflow/1.12.3    (may23:new)    ifs-git-tools/3.2.0 (D)
   git/2.30.1 (may21:old)      git/2.40.0                  ifs-git-tools/3.1.0

   Where:
   Aliases:  Aliases exist: foo/1.2.3 (1.2) means that "module load foo/1.2" will load foo/1.2.3
   D:         Default Module

If the avail list is too long consider trying:

"module --default avail" or "ml -d av" to just list the default modules.
"module overview" or "ml ov" to display the number of modules for each name.

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".

[usxa@ac6-101 ~]$ module load git
[usxa@ac6-101 ~]$ git --version
git version 2.39.1
[usxa@ac6-101 ~]$ ml

Currently Loaded Modules:
  1) gcc/8.5.0   2) prgenv/gnu   3) git/2.39.1


[usxa@ac6-101 ~]$ ml git/2.40.0

The following have been reloaded with a version change:
  1) git/2.39.1 => git/2.40.0

[usxa@ac6-101 ~]$ ml

Currently Loaded Modules:
  1) gcc/8.5.0   2) prgenv/gnu   3) git/2.40.0


[usxa@ac6-101 ~]$ git --version
git version 2.40.0
[usxa@ac6-101 ~]$ ml git/old

The following have been reloaded with a version change:
  1) git/2.40.0 => git/2.30.1

[usxa@ac6-101 ~]$ git --version
git version 2.30.1
[usxa@ac6-101 ~]$
```

Software Stack: ECMWF Software

- Some ECMWF software has moved to the **ecmwf-toolbox**

  - ecCodes

  - Magics

  - Metview

  - CodesUI

  - ODC

- A single `module load ecmwf-toolbox` to use them all!

- Run `module help ecmwf-toolbox` to get the details of the bundled packages and libraries

- Note that other packages, such as **ecFlow** still keep their standalone module:

https://confluence.ecmwf.int/x/YBhbDg

Python Support

- **Only Python 3 supported!**

- Traditional Python 3 available

  - ~ 400 extra Python modules

- Introducing **conda** for Python

  - Users can easily create their own environments to fully customise their Python experience

  - Internal conda channels available for ECMWF software

- Conda is implemented as an extra "prgenv":

  - If loaded, it **deactivates all other modules**.

  - Avoiding conflicts between conda packages and module packages

https://confluence.ecmwf.int/x/YRhbDg

Container support

- Docker is not supported

- You may use Apptainer if you wish to run containerised workloads.

  - Previously known as Singularity

  - Rootless containers

  - Supports docker and other OCI images

  - BYOE: Bring Your Own Environment. Develop in your laptop, run in our HPCF!

https://confluence.ecmwf.int/x/YhhbDg

```
$ module load apptainer
$ apptainer exec docker://ubuntu:latest cat /etc/os-release
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
Getting image source signatures
Copying blob 345e3491a907 done
Copying blob 57671312ef6f done
Copying blob 5e9250ddb7d0 done
Copying config 7c6bc52068 done
Writing manifest to image destination
Storing signatures
2021/06/07 17:51:35  info unpack layer:
sha256:345e3491a907bb7c6f1bdddcf4a94284b8b6ddd77eb7d93f09432b17b20f2bbe
2021/06/07 17:51:36  info unpack layer:
sha256:57671312ef6fdbecf340e5fed0fb0863350cd806c92b1fdd7978adbd02afc5c3
2021/06/07 17:51:36  info unpack layer:
sha256:5e9250ddb7d0fa6d13302c7c3e6a0aa40390e42424caed1e5289077ee4054709
INFO:    Creating SIF file...
NAME="Ubuntu"
VERSION="20.04.2 LTS (Focal Fossa)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 20.04.2 LTS"
VERSION_ID="20.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=focal
UBUNTU_CODENAME=focal
```

The new ecFlow service architecture

- 1 Ecflow server – 1 Virtual Machine

    1. User requests (once) access to the service.

    2. A VM is created and configured with:

        - Same HOME and PERM:

            - You may use HOME for your job standard output/error.

        - EcFlow running as a system service.

        - Troika (the ECMWF tool for submit / kill/ monitor jobs from ecFlow).

        - No other extra software present: avoid running local tasks.

    3. User gets hostname and it's all ready to go. Suites can be loaded/played straightaway.

        - Everyone uses the same default ecFlow port (3141).

        - No need for the user to start it the ecFlow server manually or use crontab.

        - No interference or competition with other users.

https://confluence.ecmwf.int/x/rSc0Dw

Interactive sessions

- Limited resources on standard SSH sessions on main login node

- ecinteractive: For more demanding interactive workload

```
$ ecinteractive -h
Usage :  /Users/usxa/bin/ecinteractive [options] [--] [CMD]


    -d|desktop      Submits a vnc job (default is interactive ssh job)
    -j|jupyter      Submits a jupyter job (default is interactive ssh job)
    -J|jupyters      Submits a jupyter job with HTTPS support (default is interactive ssh job)


    More Options:
    -h|help         Display this message
    -v|version      Display script version
    -p|platform     Platform (default hpc. Choices: hpc, ecs)
    -u|user         ECMWF User (default usxa)
    -A|account      Project account
    -c|cpus         Number of CPUs (default 2)
    -m|memory       Requested Memory (default 8 GB)
    -s|tmpdirsize   Requested TMPDIR size (default 3 GB)
    -g|gpu          Request a GPU (limited availability)
    -t|time         Wall clock limit (default 12:00:00)
    -f|forward      Ports to forward, comma separated (default NONE)
    -e|export       Environment variables to export, comma separated (default NONE)
    -k|kill         Cancel any running interactive job
    -q|query        Check running job
    -Q|quiet        Silent mode
    -o|output       Output file for the interactive job (default /dev/null)
    -x              set -x
```

https://confluence.ecmwf.int/x/dCc0Dw

Interactive sessions

Questions?

Hands on time!

https://confluence.ecmwf.int/x/-ihkFQ

- Logging in
- Storage spaces
- Software stack environment