THE CYPRUS INSTITUTE
RESEARCH•TECHNOLOGY•INNOVATION

THE CYPRUS ANTI-CANCER SOCIETY

EURO²

Introduction to Supercomputing

How to Use an HPC Machine

Chris Stylianou

Research Engineer

CaSToRC

eurocc.cyi.ac.cy

Co-funded by the European Union

Republic of Cyprus

RESEARCH & INNOVATION FOUNDATION

EuroHPC Joint Undertaking

A Bit About Myself

- Research Engineer at CaSToRC, PhD
- Collaborations Task Leader for EuroCC2
- Area of expertise in High Performance Computing (HPC)
- Contact & Info:
  - Email: c.stylianou@cyi.ac.cy
  - Website: cstyl.github.io

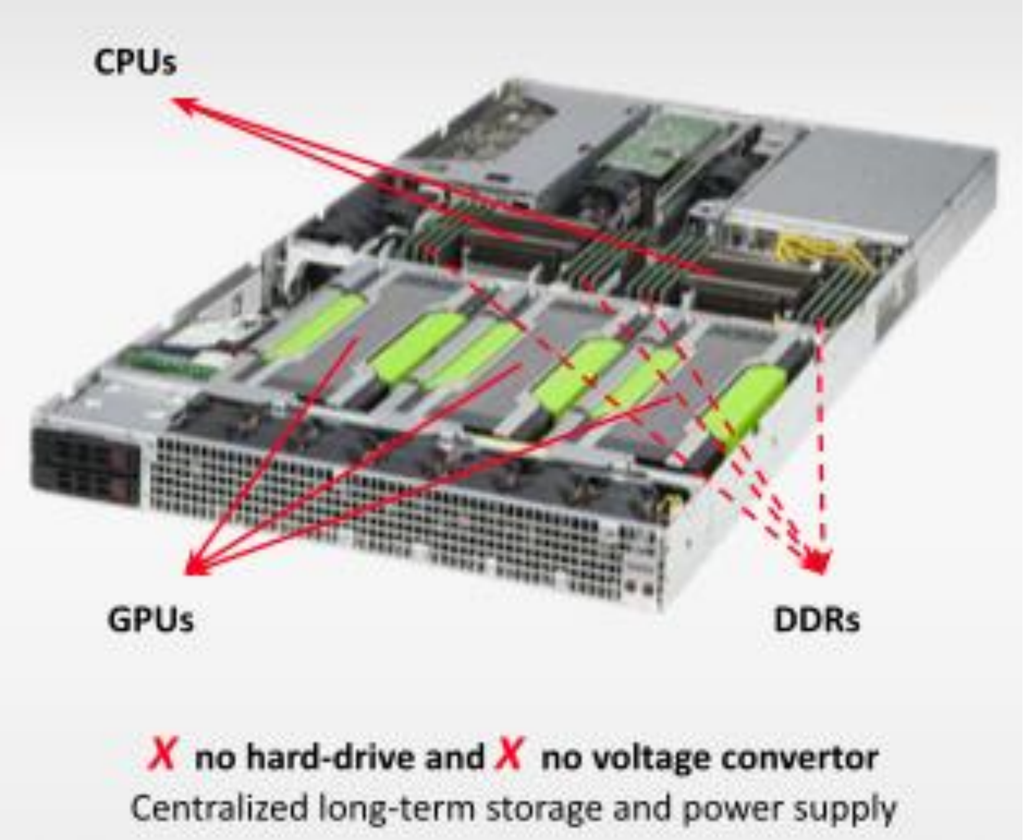
What is a Supercomputer?

Inside a Supercomputer

Compute Node

vs

Desktop



X no hard-drive and X no voltage convertor

Centralized long-term storage and power supply

A node is ~5x faster than a desktop

CPU

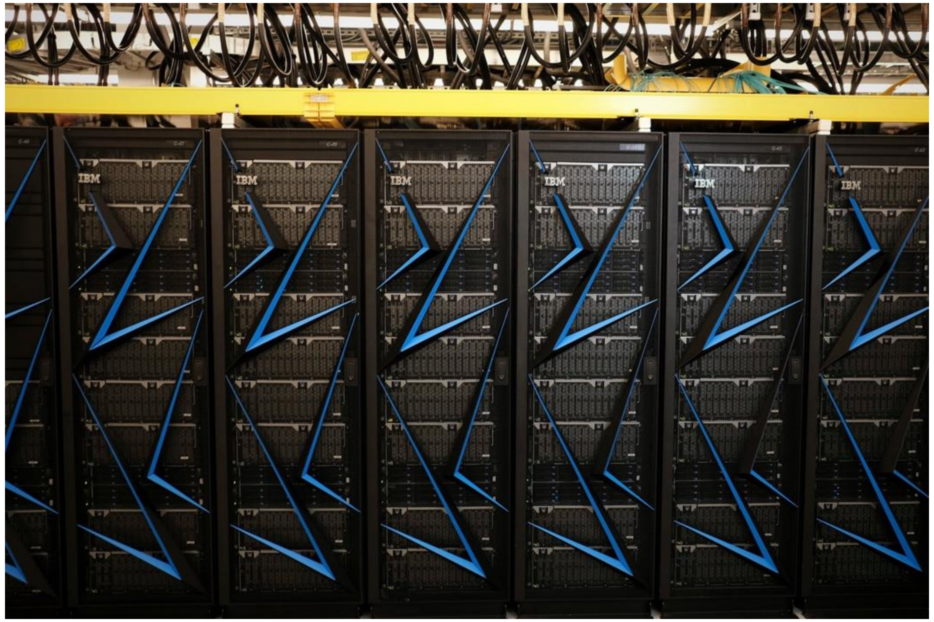
GPU

DDRs

Inside a Supercomputer

Compute Node

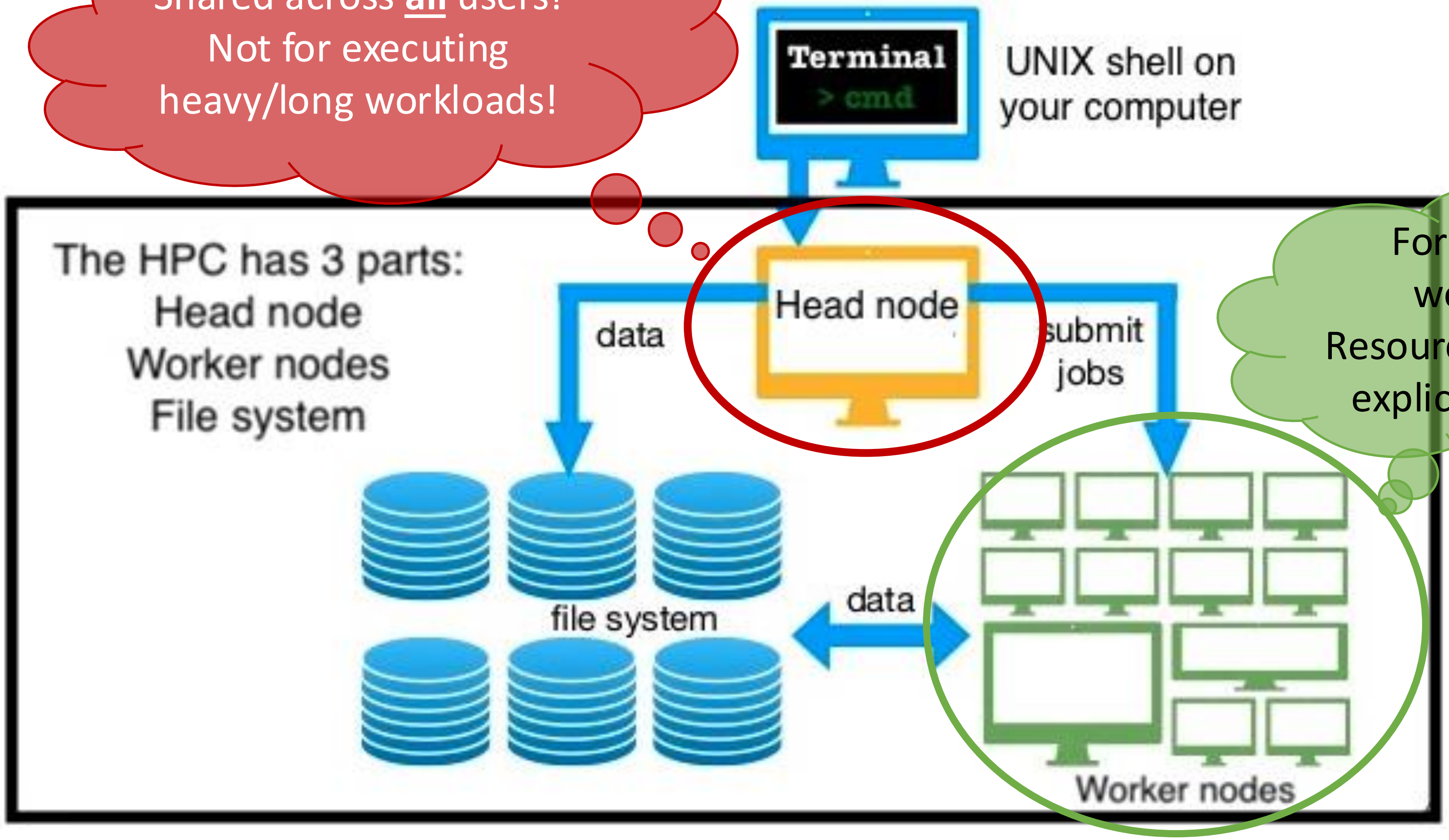
Rack



Supercomputer

The Cluster Architecture

Shared across **all** users!
Not for executing heavy/long workloads!



For executing workloads.
Resources **scheduled** explicitly per user.

Setting Up Config File

Linux/Mac:

Open a new Terminal Session.

In `~/.ssh` directory create a `config` file:

```
$ cd ~/.ssh
$ vim config
```

Configure file as shown in image. **Note** you will need to change the `User` field to your own and `IdentityFile` to the path of your ssh key generated for the hackathon.

Windows:

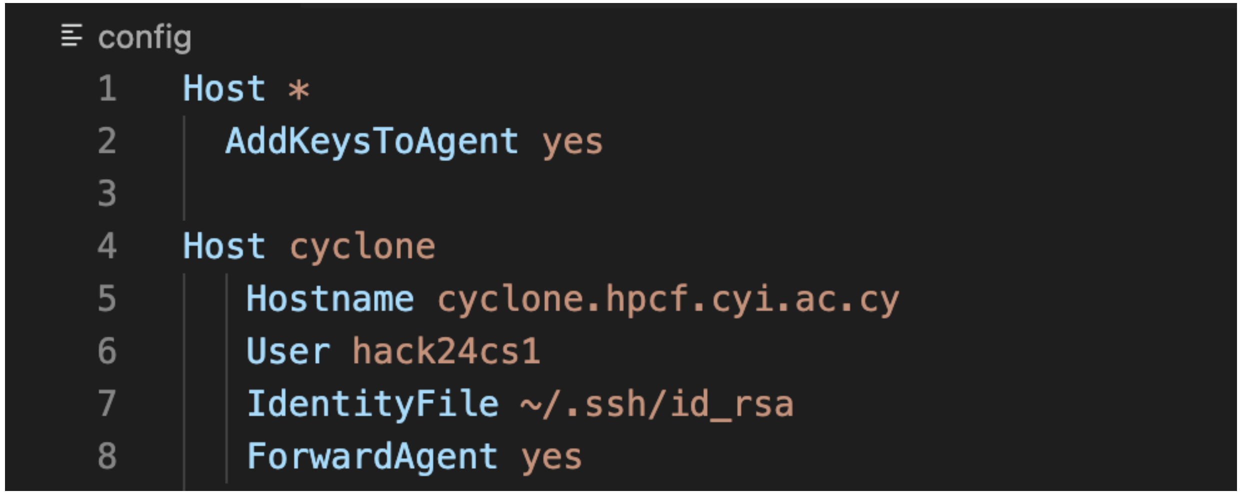
Open a new PowerShell.

In `<path\to\home>\.ssh` directory create a `config` file.

Remove any extention (e.g., .txt) from the `config` file!

Open file and configure file as shown in image.

Note you will need to change the `User` field to your own and `IdentityFile` to the path of your ssh key generated for the hackathon.



```
Host *
  AddKeysToAgent yes

Host cyclone
   Hostname cyclone.hpcf.cyi.ac.cy
   User hack24cs1
   IdentityFile ~/.ssh/id_rsa
   ForwardAgent yes
```

Accessing Cyclone Via SSH

Linux/Mac/Windows:

Open a Terminal Session.

```
$ ssh cyclone
```

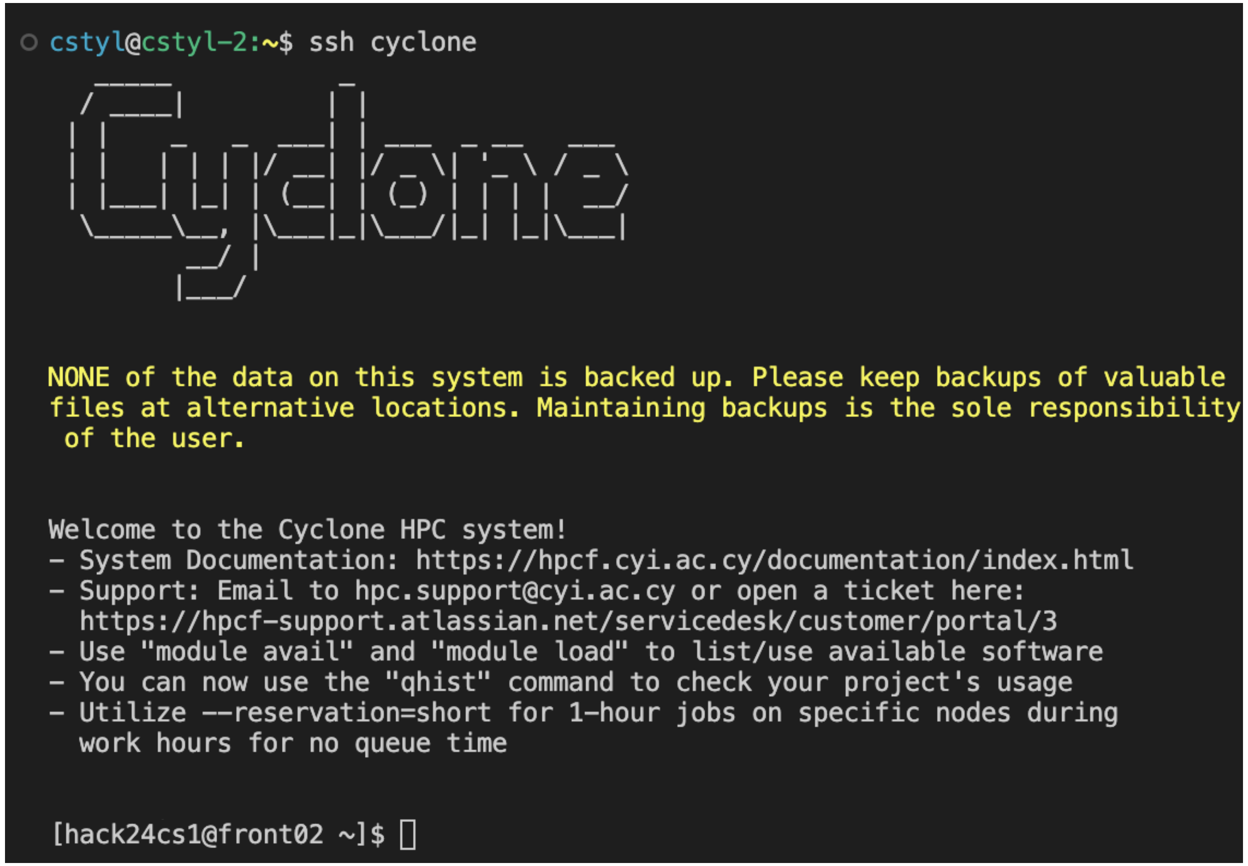
If that didn't work, possibly `config` wasn't picked up. Try:

```
$ ssh-add <path/to/ssh/key>
$ ssh <Username>@cyclone.hpcf.cyi.ac.cy
```

Windows:

If you have created your key using puTTY you will need to save it using the **OpenSSH format**.

More on https://hpcf.cyi.ac.cy/documentation/login.html



```
○ cstyl@cstyl-2:~$ ssh cyclone
   _____           _
  / ____|         | |
 | |    _   _  ___| | ___  _ __   ___
 | |   | | | |/ __| |/ _ \| '_ \ / _ \
 | |___| |_| | (__| | (_) | | | |  __/
  \_____\__, |\___|_|\___/|_| |_|\___|
         __/ |
        |___/


NONE of the data on this system is backed up. Please keep backups of valuable
files at alternative locations. Maintaining backups is the sole responsibility
 of the user.


Welcome to the Cyclone HPC system!
- System Documentation: https://hpcf.cyi.ac.cy/documentation/index.html
- Support: Email to hpc.support@cyi.ac.cy or open a ticket here:
  https://hpcf-support.atlassian.net/servicedesk/customer/portal/3
- Use "module avail" and "module load" to list/use available software
- You can now use the "qhist" command to check your project's usage
- Utilize --reservation=short for 1-hour jobs on specific nodes during
  work hours for no queue time


[hack24cs1@front02 ~]$ ▯
```

Scheduling System - SLURM

Definition:

- Simple Linux Utility for Resource Management (SLURM) is an open-source **job scheduler** used in Linux clusters **to manage and allocate resources**.

Key Features:

- Job scheduling and management.
- Resource allocation.
- Monitoring and accounting.

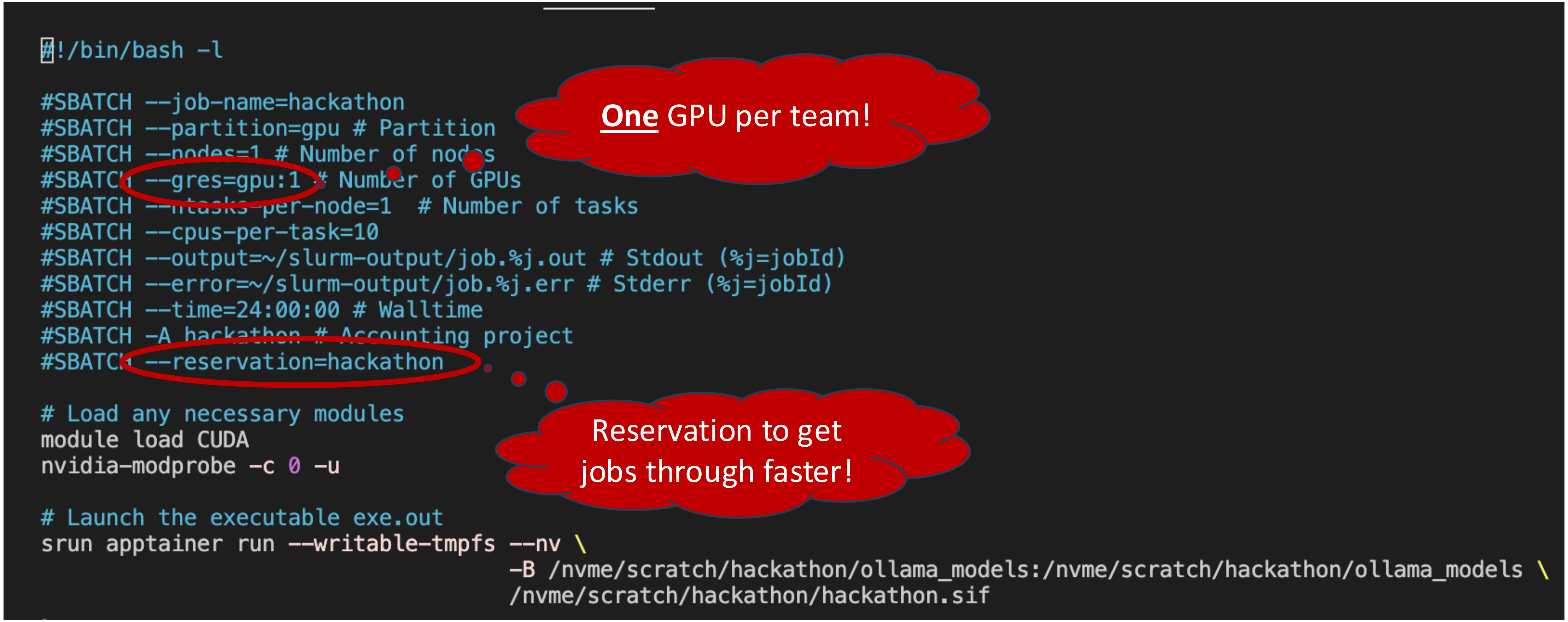
Basic Workflow:

1. **Submit a Job:** Use `sbatch` to submit a job script.
2. **Monitor a Job:** Use `squeue` to view job status.
3. **Job Completion:** Job finishes and results are available.

Essential SLURM Commands

Category	Command	Action
Job Submission	sbatch <job_script>	Submit a batch job
	srun <command>	Run a command in parallel
Job Monitoring	squeue	View queued jobs
	squeue --user=$USER	View MY queued jobs
	scontrol show job <job_id>	Detailed job info
Job Management	scancel <job_id>	Cancel a job
	scontrol hold <job_id>	Hold a job
	scontrol release <job_id>	Release a held job
Resource Allocation	sinfo	Information about nodes and partitions
	scontrol show node <node_id>	Show Node details

An Example Job Script – launch_hackathon.sub



```
#!/bin/bash -l

#SBATCH --job-name=hackathon
#SBATCH --partition=gpu # Partition
#SBATCH --nodes=1 # Number of nodes
#SBATCH --gres=gpu:1 # Number of GPUs
#SBATCH --ntasks-per-node=1  # Number of tasks
#SBATCH --cpus-per-task=10
#SBATCH --output=~/slurm-output/job.%j.out # Stdout (%j=jobId)
#SBATCH --error=~/slurm-output/job.%j.err # Stderr (%j=jobId)
#SBATCH --time=24:00:00 # Walltime
#SBATCH -A hackathon # Accounting project
#SBATCH --reservation=hackathon

# Load any necessary modules
module load CUDA
nvidia-modprobe -c 0 -u

# Launch the executable exe.out
srun apptainer run --writable-tmpfs --nv \
                                    -B /nvme/scratch/hackathon/ollama_models:/nvme/scratch/hackathon/ollama_models \
                                    /nvme/scratch/hackathon/hackathon.sif
```

One GPU per team!

Reservation to get jobs through faster!

Launching a Jupyter Notebook on Cyclone – Job Allocation

Only one user: Launch the job from Cyclone:

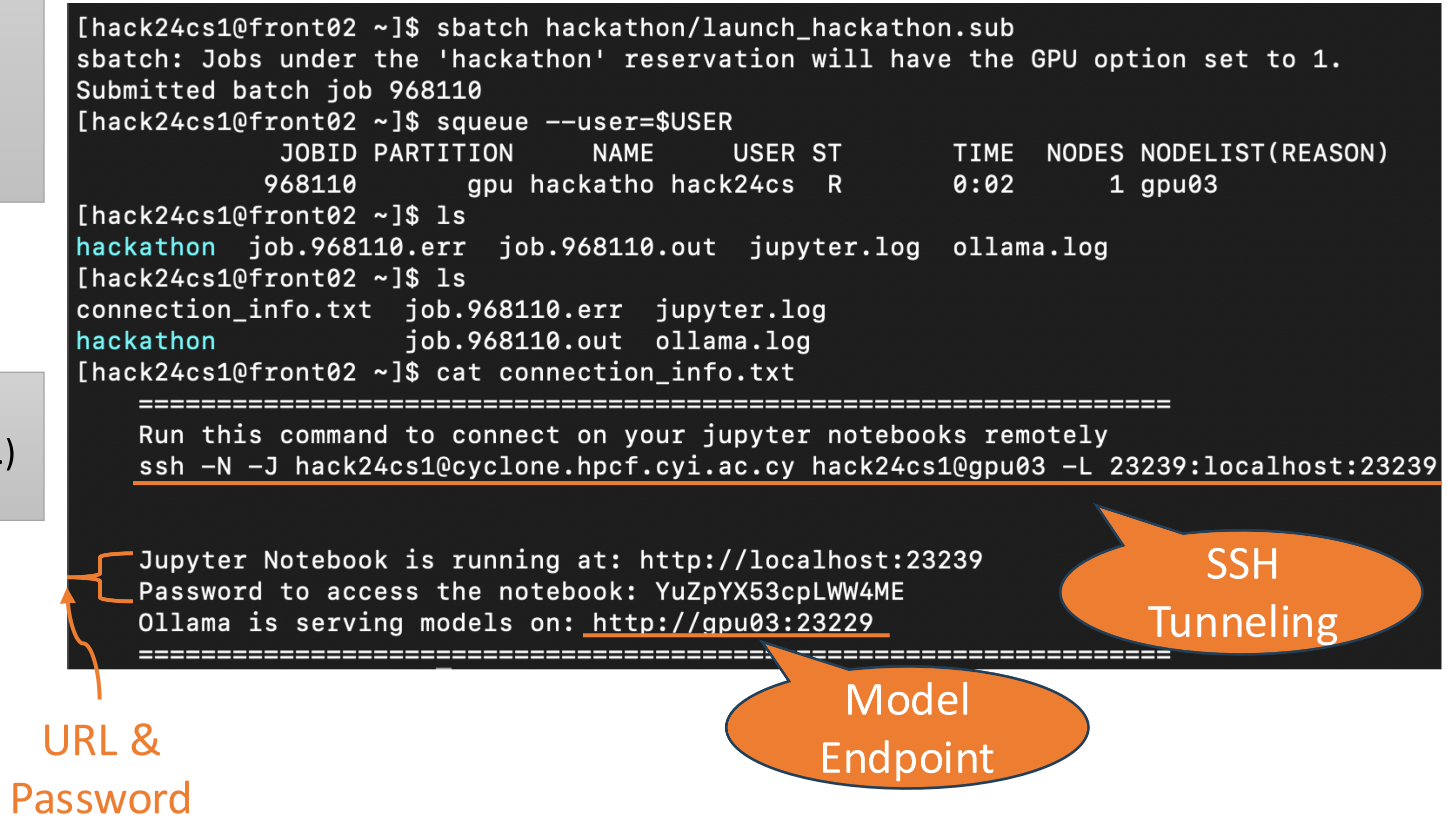
```
$ cd ~/
$ sbatch hackathon/launch_hackathon.sub
sbatch: Jobs under the 'hackathon' reservation will have the GPU
option set to 1.
Submitted batch job 968110
```

Check job status. After a while the status (ST) should be set to running (R):

```
$ squeue --user=$USER
JOBID    PARTITION    NAME      USER      ST   TIME   NODES NODELIST(...)
968110       gpu     hackatho hack24cs R   0:00     1       gpu03
```

Once the job starts running, a `connection_info.txt` should appear in your `$HOME` directory containing info to:

- Setup SSH tunneling
- URL for accesing notebook on your browser
  - Password
- Model endpoint.



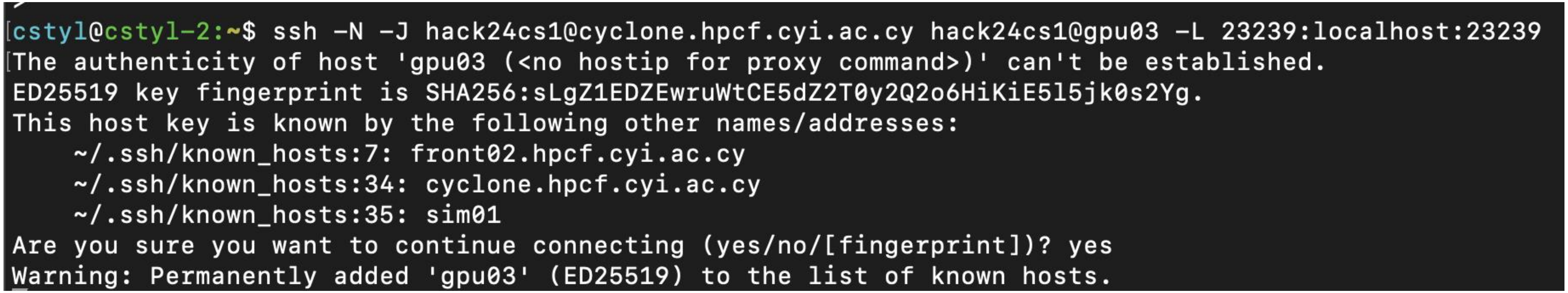
```
[hack24cs1@front02 ~]$ sbatch hackathon/launch_hackathon.sub
sbatch: Jobs under the 'hackathon' reservation will have the GPU option set to 1.
Submitted batch job 968110
[hack24cs1@front02 ~]$ squeue --user=$USER
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
            968110       gpu hackatho hack24cs  R       0:02      1 gpu03
[hack24cs1@front02 ~]$ ls
hackathon  job.968110.err  job.968110.out  jupyter.log  ollama.log
[hack24cs1@front02 ~]$ ls
connection_info.txt  job.968110.err  jupyter.log
hackathon            job.968110.out  ollama.log
[hack24cs1@front02 ~]$ cat connection_info.txt
    =================================================================
    Run this command to connect on your jupyter notebooks remotely
    ssh -N -J hack24cs1@cyclone.hpcf.cyi.ac.cy hack24cs1@gpu03 -L 23239:localhost:23239


    Jupyter Notebook is running at: http://localhost:23239
    Password to access the notebook: YuZpYX53cpLWW4ME
    Ollama is serving models on: http://gpu03:23229
    =================================================================
```

Launching a Jupyter Notebook on Cyclone – SSH Tunneling

- Launch a new terminal session

```
$ ssh -N -J <Username>@cyclone.hpcf.cyi.ac.cy <username>@gpu03 -L 23239:localhost:23239
```

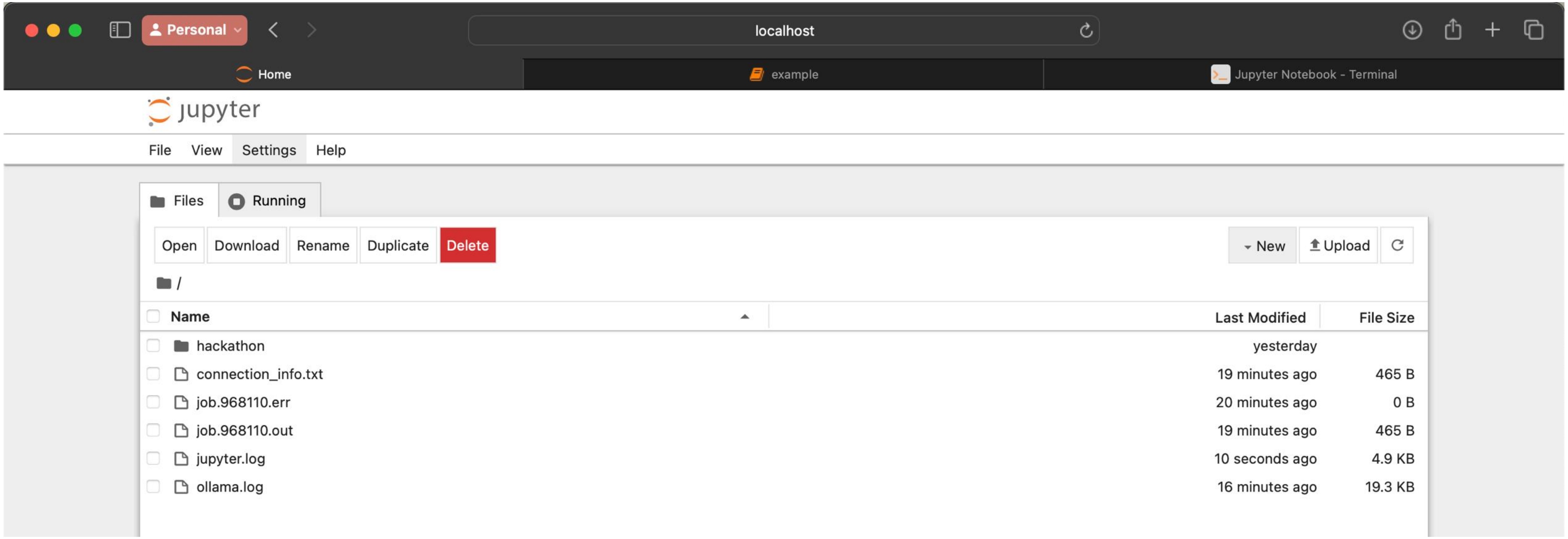
- If prompted about establishing authenticity of the host, press ”Y” and enter.



```
cstyl@cstyl-2:~$ ssh -N -J hack24cs1@cyclone.hpcf.cyi.ac.cy hack24cs1@gpu03 -L 23239:localhost:23239
The authenticity of host 'gpu03 (<no hostip for proxy command>)' can't be established.
ED25519 key fingerprint is SHA256:sLgZ1EDZEwruWtCE5dZ2T0y2Q2o6HiKiE5l5jk0s2Yg.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:7: front02.hpcf.cyi.ac.cy
    ~/.ssh/known_hosts:34: cyclone.hpcf.cyi.ac.cy
    ~/.ssh/known_hosts:35: sim01
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'gpu03' (ED25519) to the list of known hosts.
```

Launching a Jupyter Notebook on Cyclone – Open On Browser

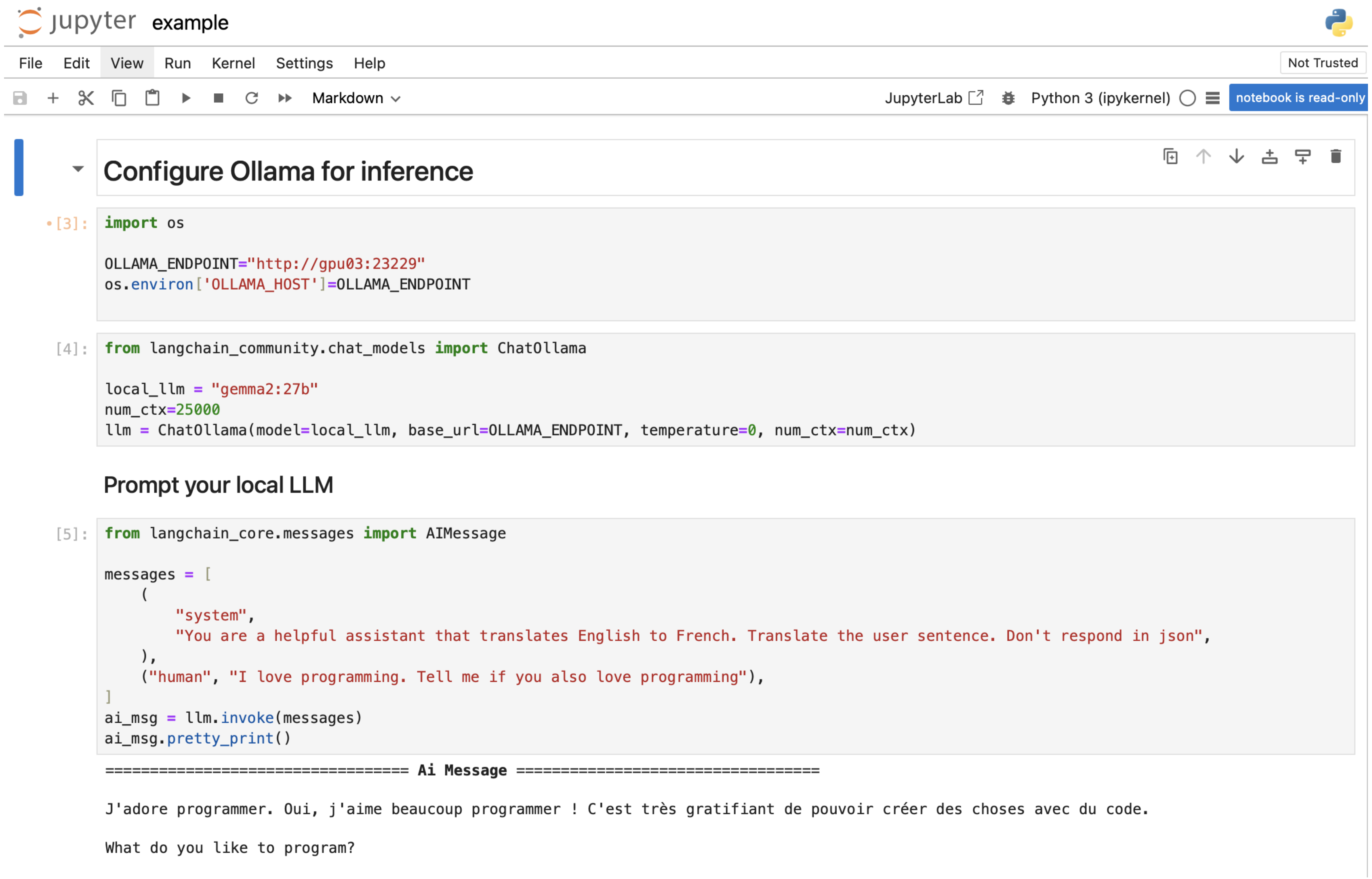
- After the SSH Tunnel has been established, open your browser and enter the link and password to Jupyter Notebook. E.g.,:
  - `http://localhost:23239`
  - `YuZpYX53cpLWW4ME`



Name	Last Modified	File Size
hackathon	yesterday	
connection_info.txt	19 minutes ago	465 B
job.968110.err	20 minutes ago	0 B
job.968110.out	19 minutes ago	465 B
jupyter.log	10 seconds ago	4.9 KB
ollama.log	16 minutes ago	19.3 KB

Launching a Jupyter Notebook on Cyclone – Test

- Open `hackathon > example.ipynb`
- Run first three cells to prompt your local LLM!
  - First time it runs might take a while until it loads the model.
- Share with each team member the URL and Password to open the model on their own browser!



Configure Ollama for inference

•[3]:

```python
import os

OLLAMA_ENDPOINT="http://gpu03:23229"
os.environ['OLLAMA_HOST']=OLLAMA_ENDPOINT
```

[4]:

```python
from langchain_community.chat_models import ChatOllama

local_llm = "gemma2:27b"
num_ctx=25000
llm = ChatOllama(model=local_llm, base_url=OLLAMA_ENDPOINT, temperature=0, num_ctx=num_ctx)
```

Prompt your local LLM

[5]:

```python
from langchain_core.messages import AIMessage

messages = [
    (
        "system",
        "You are a helpful assistant that translates English to French. Translate the user sentence. Don't respond in json",
    ),
    ("human", "I love programming. Tell me if you also love programming"),
]
ai_msg = llm.invoke(messages)
ai_msg.pretty_print()
```

```
================================== Ai Message ==================================

J'adore programmer. Oui, j'aime beaucoup programmer ! C'est très gratifiant de pouvoir créer des choses avec du code.

What do you like to program?
```

Happy Hacking!!

More information:

https://castorc.cyi.ac.cy/
https://eurocc.cyi.ac.cy/

Contact us at:
eurocc-contact@cyi.ac.cy

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) and Germany, Bulgaria, Austria, Croatia, Cyprus (co-funded by the EU within the framework of the Cohesion Policy Programme “THALIA 2021-2027”), Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia under grant agreement No 101101903.