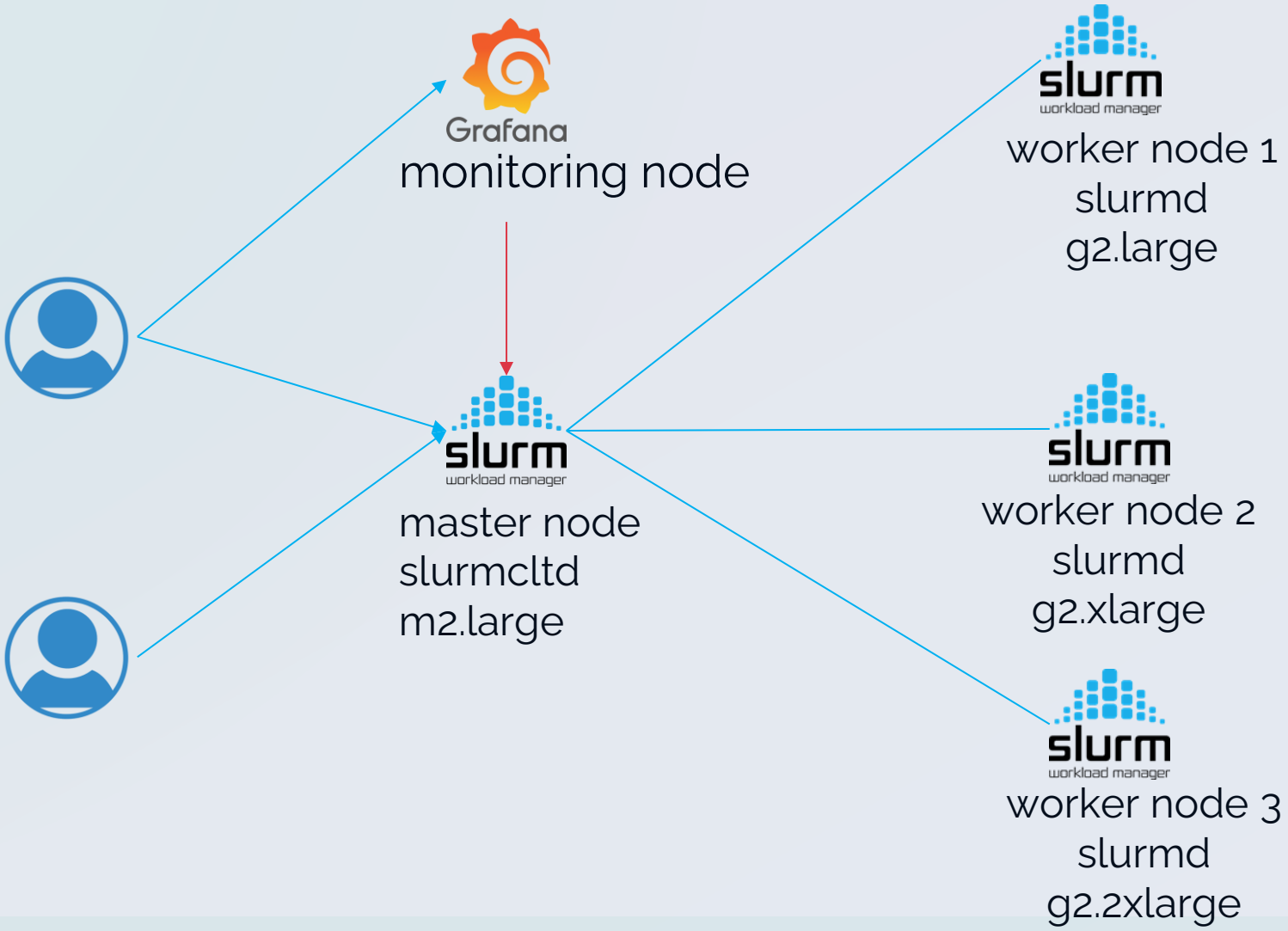
Slurm szolgáltatás elérése és használata a HUN-REN Cloudon

Tartalom

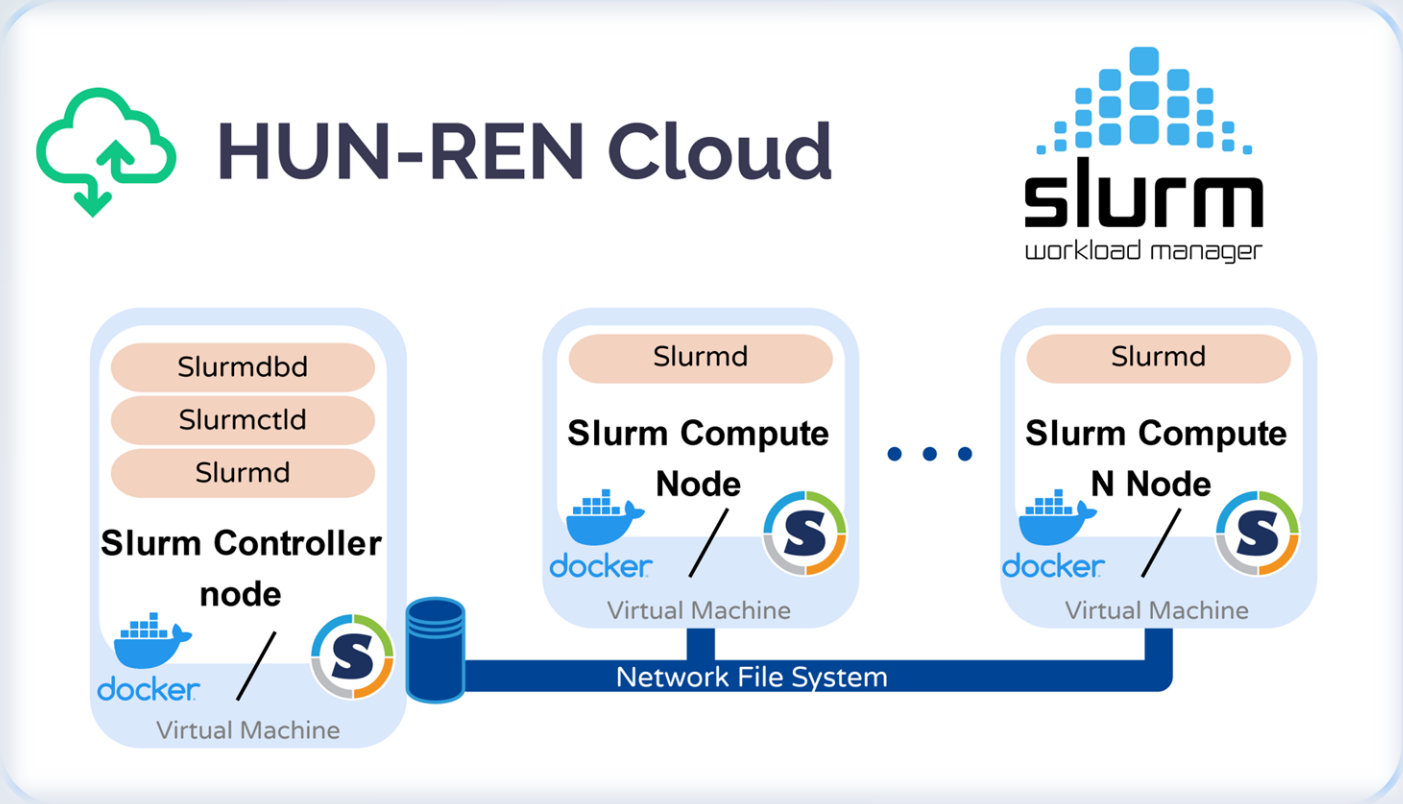
- Slurm szolgáltatás elérése
- Slurm szolgáltatás használata
- Integrált szolgáltatások
  - Slurm GPU partíciók
  - Singularity
  - MP & MPI könyvtárak

Slurm szolgáltatás



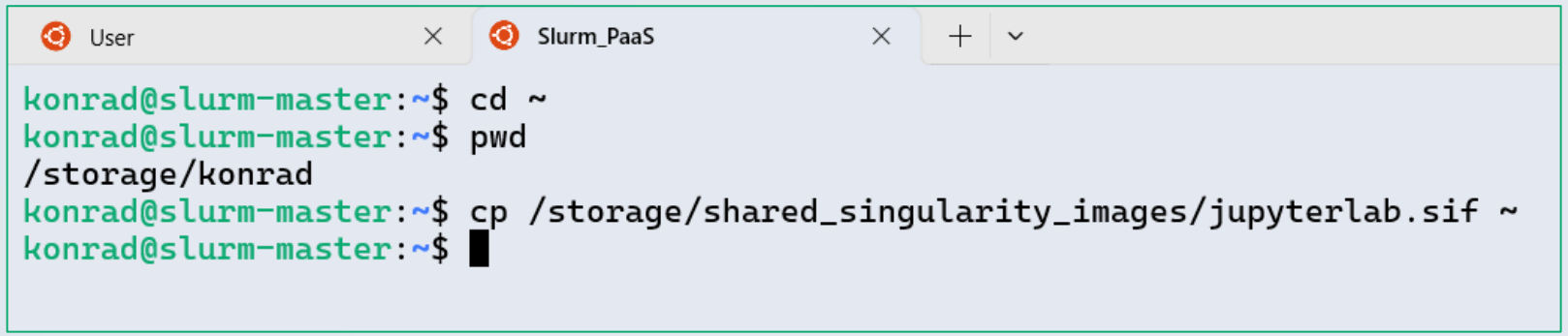
Slurm szolgáltatás

- Számítási erőforrások hatékonyabb, dinamikusabb elosztása egy közös számítási környezetben
- Magas rendelkezésre állás



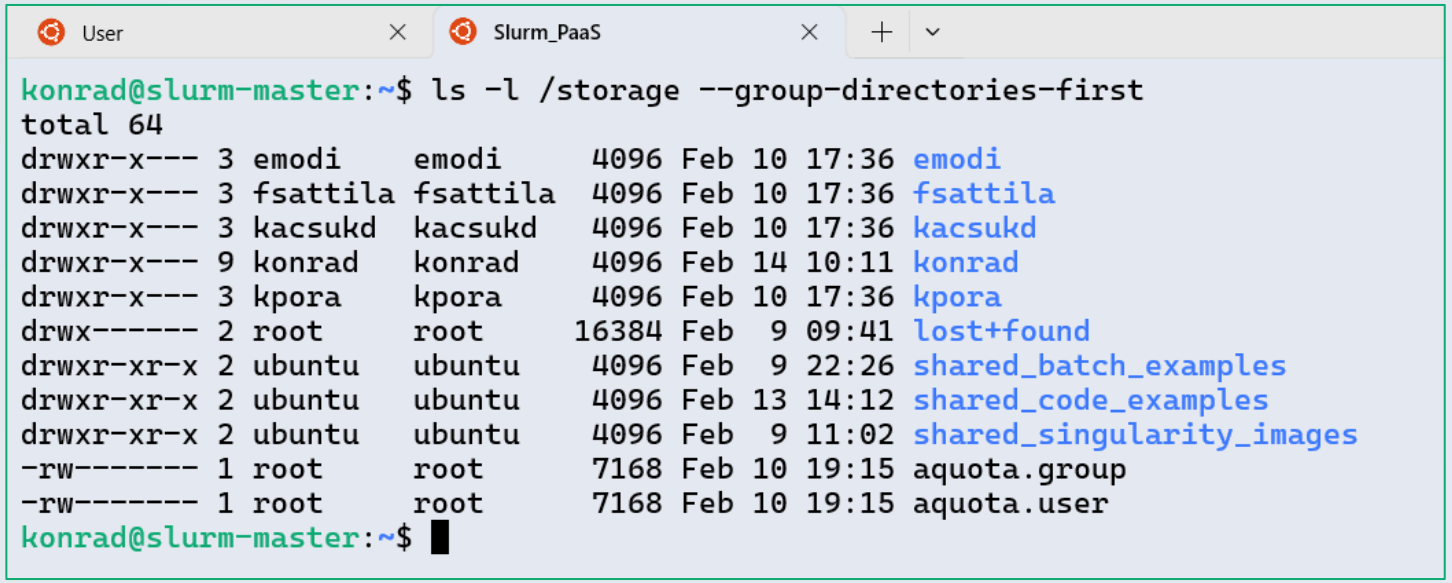
Slurm szolgáltatás

- */storage* mappa → közös hálózati meghajtó
- A felhasználók */home* mappái itt találhatóak
  - Célszerű minden esetben innen dolgozni
- Példa kódok és image-ek a */storage/shared_* könyvtárak alatt találhatóak



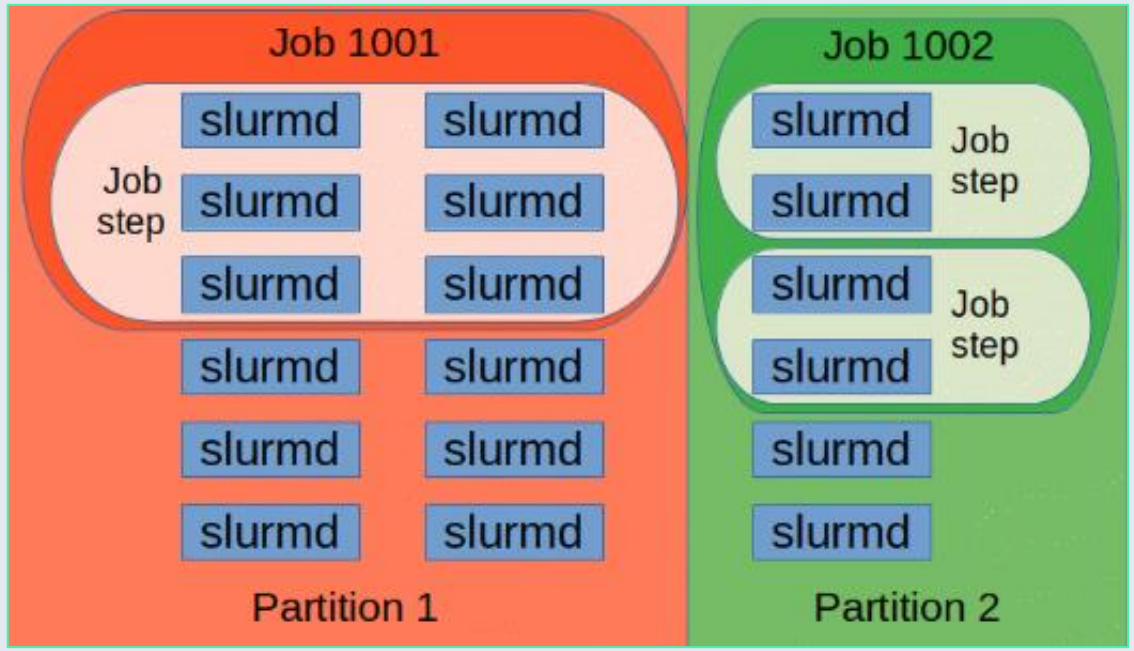
Slurm szolgáltatás

- ~13 TB közös tárhely
  - 100 GB / user
  - *shared_* könyvtárak



Slurm szolgáltatás

- Slurm klaszter
  - Partíciók
    - Slurm node-ok (virtuális gépek)
      - Feladat (Batch / Job)
        - Feladat lépések (Job steps)



- Adott virtuális gép tagja lehet több partíciónak is
- Jelen felosztás → GPU erőforrások szerint

GPU partíciók

- Jelenlegi rendelkezésre álló erőforrások
  - m2.large → 4
  - g2.large → 11
  - g2.xlarge → 2
  - g2.2xlarge → 1

Név	VCPU	RAM	GPU RAM
g2.large	4	16GB	8GB
g2.xlarge	8	32GB	16GB
g2.2xlarge	16	64GB	32GB

GPU partíciók

- Tervezett rendelkezésre álló erőforrások
  - m2.large → 4
  - g2.large → 10
  - g2.xlarge → 4
  - g2.2xlarge → 2

Név	VCPU	RAM	GPU RAM
g2.large	4	16GB	8GB
g2.xlarge	8	32GB	16GB
g2.2xlarge	16	64GB	32GB

GPU partíciók

- Skálázható klaszter
  - A GPU erőforrások dinamikusan hozzárendelhetőek a számítási környezethez
  - Felhasználói igényekhez igazított erőforrás menedzsment
  - 1 Slurm Node : 1GPU

User | Slurm_PaaS

```
konrad@slurm-master:~$ sinfo
PARTITION                 AVAIL  TIMELIMIT  NODES  STATE NODELIST
batch_cpu_m2.large           up 1-00:00:00      4   idle slurm-master,slurm-worker-cpu-m2-large-[1-3]
batch_gpu_g2.large_8*        up 1-00:00:00      7   idle slurm-worker-gpu-g2-large-[1-7]
batch_gpu_g2.xlarge_16       up 1-00:00:00      2   idle slurm-worker-gpu-g2-xlarge-[1-2]
batch_gpu_g2.2xlarge_32      up 1-00:00:00      1   idle slurm-worker-gpu-g2-2xlarge-1
interactive_cpu_m2.large     up 7-00:00:00      4   idle slurm-master,slurm-worker-cpu-m2-large-[1-3]
interactive_gpu_g2.large_8   up 7-00:00:00      7   idle slurm-worker-gpu-g2-large-[1-7]
konrad@slurm-master:~$
```

GPU partíciók

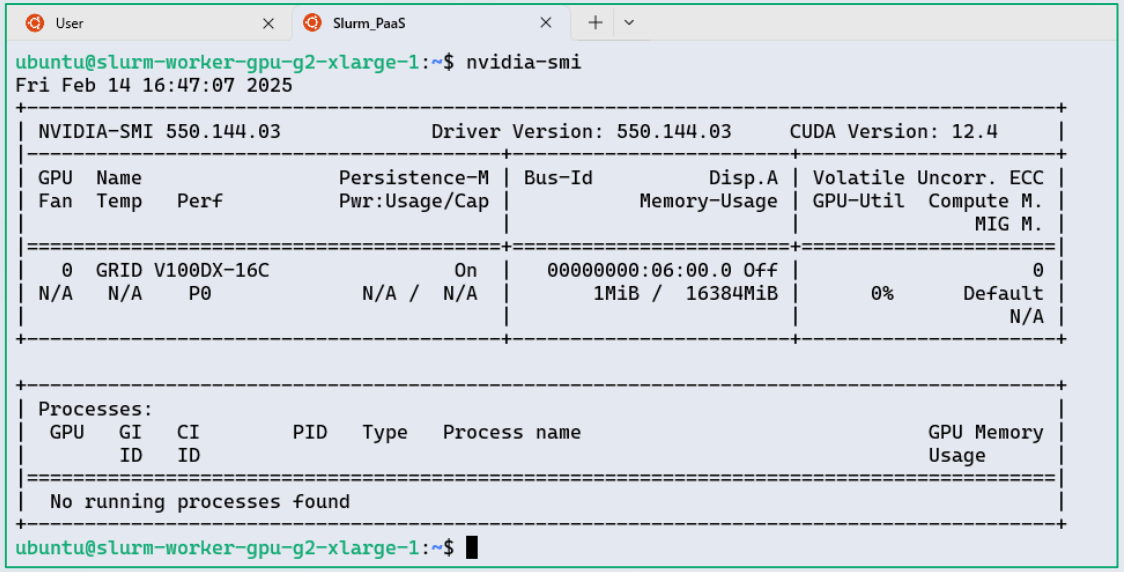
- 4 **Batch** partíció
  - csillag szimbólum → alapértelmezett partíció
  - **sbatch –*p* <partíció> <feladat.batch>** → partíció kiválasztása

User | Slurm_PaaS

```
konrad@slurm-master:~$ sinfo
PARTITION                 AVAIL  TIMELIMIT  NODES  STATE NODELIST
batch_cpu_m2.large           up 1-00:00:00      4   idle slurm-master,slurm-worker-cpu-m2-large-[1-3]
batch_gpu_g2.large_8*        up 1-00:00:00      7   idle slurm-worker-gpu-g2-large-[1-7]
batch_gpu_g2.xlarge_16       up 1-00:00:00      2   idle slurm-worker-gpu-g2-xlarge-[1-2]
batch_gpu_g2.2xlarge_32      up 1-00:00:00      1   idle slurm-worker-gpu-g2-2xlarge-1
interactive_cpu_m2.large     up 7-00:00:00      4   idle slurm-master,slurm-worker-cpu-m2-large-[1-3]
interactive_gpu_g2.large_8   up 7-00:00:00      7   idle slurm-worker-gpu-g2-large-[1-7]
konrad@slurm-master:~$
```

Integrált szolgáltatások – GPU partíciók

- NVIDIA GRID 550.144.03



Programok	Támogatott verziók
Cuda	11.0 - 12.3
Pytorch	1.12.0 - 2.2.x
Jupyterlab	3.0 – 4.x
Tensorflow	2.10.0 - 2.15.x

Job létrehozása– példa

- Lépések
  - .batch fájl (job) létrehozása
  - **#SBATCH** paraméterek definiálása
  - Futtatni kívánt állomány definiálása
    - Shell parancs
    - Shell szkript
    - Kód (Python, C++ etc)
  - Job beküldése
  - Eredmények lekérdezése

User | Slurm_Paas

```
 GNU nano 7.2
#!/bin/bash
#SBATCH --job-name=helloworld_job
#SBATCH --output=helloworld_job.out
#SBATCH --error=helloworld_jobb.err

echo "Hello  World"
srun hostname
pwd
```

Slurm_PaaS

```
konrad@slurm-master:~/demo_batch$ sbatch helloworld.batch
Submitted batch job 300
konrad@slurm-master:~/demo_batch$ cat helloworld_job.out
Hello  World
slurm-worker-gpu-g2-large-1
/storage/konrad/demo_batch
konrad@slurm-master:~/demo_batch$
```

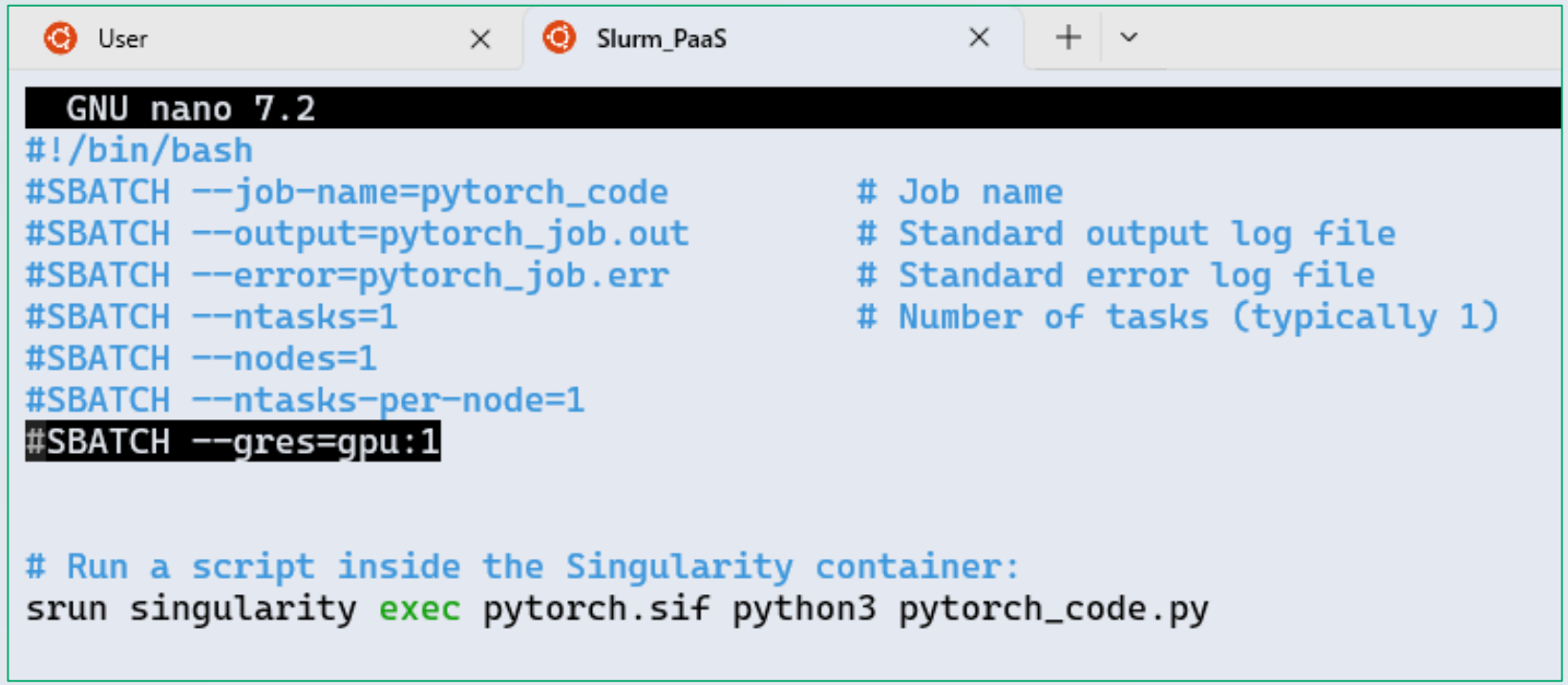
Slurm használata - #BATCH job paraméterek

- **#BATCH**
  - Adott feladatra vonatkozó beállítások
  - Kimeneti fájlok
  - Erőforrás allokálás feladat szinten
  - Nem számít a sorrend

- #SBATCH --job-name=example_job
- #SBATCH --output=example_job.out
- #SBATCH --error=example_job.err
- #SBATCH --time=00:01:00
- #SBATCH --partition
- #SBATCH --gres=gpu:nvidia:1

BATCH paraméter GPU használathoz

- **#BATCH –-gres=gpu:1** → batch paraméter GPU használathoz
  - a Slurm feladatütemező számára szükséges



Slurm szolgáltatás elérése

- SSH kapcsolat
  - domain → slurm.science-cloud.hu
  - publikus IP cím → 193.225.251.214
- Proxy kapcsolat (optional)
  - böngésző proxy beállítások
- SCP vagy rsync fájlok és mappák fel-és letöltéséhez

Slurm használata - parancsok

- **sinfo** → Megjeleníti a klaszter node-ok állapotát
- **sacctmgr show user** → Kilistázza a felhasználókat és jogosultságaikat.
- **sbatch example_job.batch** → Beküld egy batch feladatot az ütemezési sorba.
- **scancel <job_ id>** → Megszakít egy futó vagy várakozó feladatot az id alapján.
- **sstat <job_ id>** → Valós idejű statisztikákat mutat egy futó feladatról.
- **squeue** → Kilistázza az aktuális felhasználó várakozó és futó feladatait.
- **sacct** → Kilistázza minden korábban és jelenleg futó feladat státuszát.

Slurm használata – feladat állapotok

- **Running (R)** → A feladat épp fut.
- **Completed (CD)** → A feladat sikeresen lefutott.
- **Pending (PD)** → A feladat erőforrásokra várakozik.
- **Suspended (S)** → A feladat felfüggesztése került (rendszer által).
- **Canceled (CA)** → A feladat visszahívásra került (felhasználó által).
- **Failed (F)** → A feladat hibára futott.
- **Timeout (TO)** → A feladat elérte az időkorlátot még a sikeres futás előtt.

Slurm használata – squeue

User | Slurm_PaaS

```
konrad@slurm-master:~$ sbatch example2.batch
Submitted batch job 141
konrad@slurm-master:~$ squeue -u konrad
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
               141 batch_gpu example2   konrad  R       0:07      1 slurm-worker-gpu-g2-large-1
konrad@slurm-master:~$
```

User | Slurm_PaaS

```
konrad@slurm-master:~$ cat example2.out
Job started at: Fri Feb 14 16:05:04 UTC 2025
Running on node: slurm-worker-gpu-g2-large-1
Job output - iteration 1
Job output - iteration 2
Job output - iteration 3
Job output - iteration 4
Job output - iteration 5
```

- Futó job állapotának lekérdezése
  - **squeue** parancs

Slurm használata – hibakeresés

- Hibaüzenet a feladat futtatását követően
  - Klaszter vagy futtató környezet szintű hibák

User | Slurm_PaaS

```
konrad@slurm-master:~$ nano mpi4py.batch
konrad@slurm-master:~$ sbatch mpi4py.batch
Submitted batch job 148
konrad@slurm-master:~$ watch squeue -u konrad
konrad@slurm-master:~$ cat mpi4py_test_148.out
konrad@slurm-master:~$ cat mpi4py_test_148.err
--------------------------------------------------------------------------
WARNING: An invalid value was given for btl_tcp_if_include.  This
value will be ignored.

  Local host: slurm-worker-gpu-g2-large-1
  Value:      eth0
  Message:    Unknown interface name
--------------------------------------------------------------------------
```

Slurm használata – Parameter sweep

- Azonos művelet futtatása különböző bemeneti adathalmazokon

User | Slurm_Paas

```
konrad@slurm-master:~/demo_paramsweep/n2$ ls -lh
total 40K
-rw-rw-r-- 1 konrad konrad  107 Feb 18 12:57 data_1.txt
-rw-rw-r-- 1 konrad konrad  107 Feb 18 12:57 data_2.txt
-rw-rw-r-- 1 konrad konrad  107 Feb 18 12:57 data_3.txt
-rw-rw-r-- 1 konrad konrad  107 Feb 18 12:57 data_4.txt
-rw-rw-r-- 1 konrad konrad  104 Feb 18 12:57 data_5.txt
-rw-rw-r-- 1 konrad konrad  107 Feb 18 12:57 data_6.txt
-rw-rw-r-- 1 konrad konrad  105 Feb 18 12:57 data_7.txt
-rw-rw-r-- 1 konrad konrad  107 Feb 18 12:57 data_8.txt
-rwxrwxr-x 1 konrad konrad  476 Feb 18 12:57 gen_data.sh
-rw-rw-r-- 1 konrad konrad 2.6K Feb 18 14:19 paramsweep_n2
konrad@slurm-master:~/demo_paramsweep/n2$
```

```
echo "Starting distributed job steps processing..."

# Get the node list
NODELIST=($(scontrol show hostname $SLURM_JOB_NODELIST))
NUM_NODES=${#NODELIST[@]}

# Process each file as a separate job step
for i in {1..8}; do
    # Distribute tasks across available nodes in a round-robin fashion
    NODE_INDEX=$(($(($i-1)) % $NUM_NODES))
    NODE=${NODELIST[$NODE_INDEX]}

    echo "Starting step for file $i on node $NODE"
    # Run as a job step with srun, specifically on the selected node
    srun --nodes=1 \
         --nodelist=$NODE \
         --output=step_${i}_%j.out \
         --error=step_${i}_%j.err \
         --gres=gpu:nvidia:1 \
         /bin/bash -c "
        echo 'Processing data_${i}.txt on '\`hostname\`;
        INPUT_FILE=data_${i}.txt;
        OUTPUT_FILE=results_${i}.txt;
        # Check if input file exists
        if [ ! -f \$INPUT_FILE ]; then
            echo \"Error: Input file \$INPUT_FILE not found!\";
            exit 1;
```

Slurm használata – Parameter sweep

User | Slurm_Paas

```
 GNU nano 7.2                                                                 param
#!/bin/bash
#SBATCH --job-name=steps_job              # Job name
#SBATCH --output=steps_%j.out            # Output file name
#SBATCH --error=steps_%j.err             # Error file name
#SBATCH --time=01:00:00                  # Time limit: 1 hour
#SBATCH --mem=4G                         # Memory per node
#SBATCH --cpus-per-task=1                # CPU cores per task
#SBATCH --nodes=2                        # Use 2 available GPU nodes
#SBATCH --ntasks-per-node=4              # Allow 4 tasks per node (we have 8 tasks total)
#SBATCH --partition=batch_gpu_g2.large_8 # Specify the GPU partition (optional)
#SBATCH --gres=gpu:nvidia:1              # GPU requirement per task
```

- Művelet(ek) futtatása 8 bemeneti adathalmazzal 2 node-on
  - nntasks-per-node=4

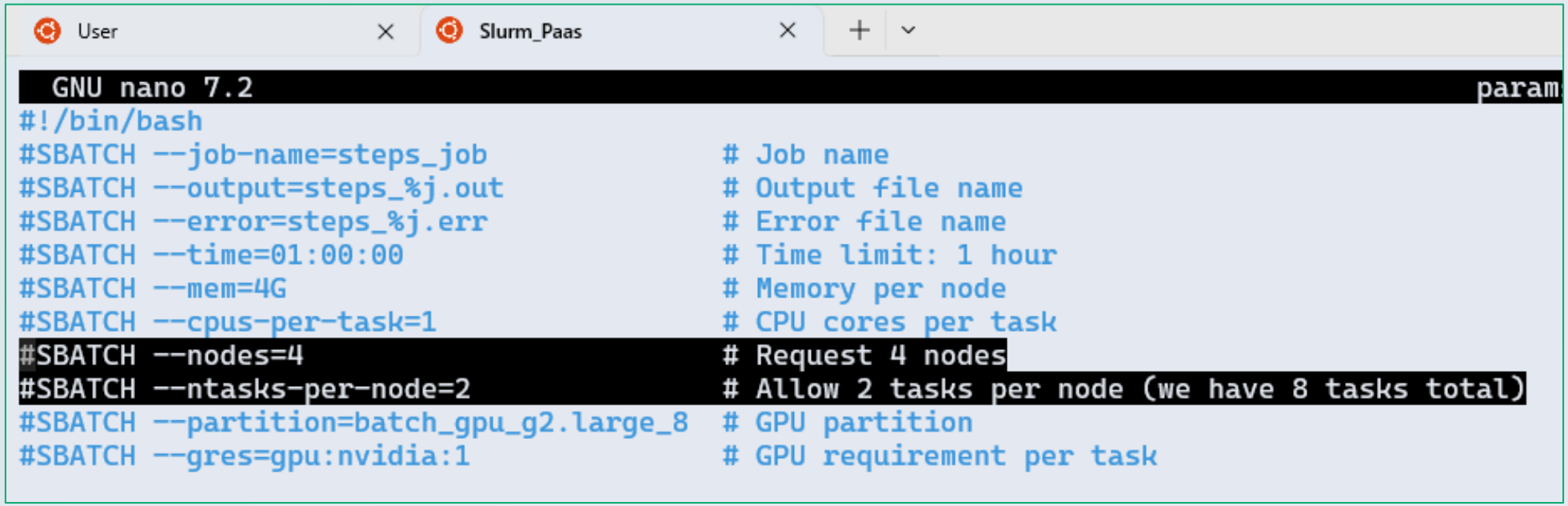
Slurm használata – Parameter sweep

Slurm_PaaS

```
konrad@slurm-master:~/demo_paramsweep/n2$ sbatch paramsweep_n2.batch
Submitted batch job 278
konrad@slurm-master:~/demo_paramsweep/n2$ watch squeue
konrad@slurm-master:~/demo_paramsweep/n2$ cat steps_278.out
Starting distributed job steps processing...
Starting step for file 1 on node slurm-worker-gpu-g2-large-1
Step 1 submitted to node slurm-worker-gpu-g2-large-1
Starting step for file 2 on node slurm-worker-gpu-g2-large-2
Step 2 submitted to node slurm-worker-gpu-g2-large-2
Starting step for file 3 on node
Step 3 submitted to node
Starting step for file 4 on node
Step 4 submitted to node
Starting step for file 5 on node
Step 5 submitted to node
Starting step for file 6 on node
Step 6 submitted to node
Starting step for file 7 on node
Step 7 submitted to node
Starting step for file 8 on node
Step 8 submitted to node
All steps submitted
konrad@slurm-master:~/demo_paramsweep/n2$
```

- Művelet(ek) futtatása 8 bemeneti adathalmazzal 2 node-on
  - nntasks-per-node=4

Slurm használata – Parameter sweep



- Művelet(ek) futtatása 8 bemeneti adathalmazzal 4 node-on
  - nntasks-per-node=2

Slurm használata – Parameter sweep

Slurm_PaaS

```
konrad@slurm-master:~/demo_paramsweep/n4$ sbatch paramsweep_n4.batch
Submitted batch job 279
konrad@slurm-master:~/demo_paramsweep/n4$ watch squeue
konrad@slurm-master:~/demo_paramsweep/n4$ cat steps_279.out
Starting distributed job steps processing...
Starting step for file 1 on node slurm-worker-gpu-g2-large-1
Step 1 submitted to node slurm-worker-gpu-g2-large-1
Starting step for file 2 on node slurm-worker-gpu-g2-large-2
Step 2 submitted to node slurm-worker-gpu-g2-large-2
Starting step for file 3 on node slurm-worker-gpu-g2-large-3
Step 3 submitted to node slurm-worker-gpu-g2-large-3
Starting step for file 4 on node slurm-worker-gpu-g2-large-4
Step 4 submitted to node slurm-worker-gpu-g2-large-4
Starting step for file 5 on node
Step 5 submitted to node
Starting step for file 6 on node
Step 6 submitted to node
Starting step for file 7 on node
Step 7 submitted to node
Starting step for file 8 on node
Step 8 submitted to node
All steps submitted
konrad@slurm-master:~/demo_paramsweep/n4$
```

- Művelet(ek) futtatása 8 bemeneti adathalmazzal 4 node-on
  - nntasks-per-node=2

Több node - #BATCH job paraméterek

- Memória
  - #SBATCH –-mem
  - #SBATCH –mem-per-task
  - #SBATCH –-mem-per-cpu
- CPU
  - #SBATCH --nnodes
  - #SBATCH –-ntask
  - #SBATCH --ntasks-per-node
  - #SBATCH --cpus-per-task

Több node - #BATCH job paraméterek

- 2 node
- 4 feladat (job step) node-onként
- 2 CPU és 4GB memória feladatonként
  - #SBATCH --nodes=2
  - #SBATCH --ntasks-per-node=4
  - #SBATCH --cpus-per-task=2
  - #SBATCH --mem-per-task=4G

- 8 feladat
- CPU feladatonként
  - #SBATCH --ntasks=8
  - #SBATCH --cpus-per-task=1

Több node - #BATCH job paraméterek

User | Slurm_Paas

```
 GNU nano 7.2                                                                 param
#!/bin/bash
#SBATCH --job-name=steps_job              # Job name
#SBATCH --output=steps_%j.out            # Output file name
#SBATCH --error=steps_%j.err             # Error file name
#SBATCH --time=01:00:00                  # Time limit: 1 hour
#SBATCH --mem=32G                        # Memory per node
#SBATCH --cpus-per-task=1                # CPU cores per task
#SBATCH --nodes=4                        # Request 4 nodes
#SBATCH --ntasks-per-node=2              # Allow 2 tasks per node (we have 8 tasks total)
#SBATCH --partition=batch_gpu_g2.large_8 # GPU partition
#SBATCH --gres=gpu:nvidia:1              # GPU requirement per task
```

User | Slurm_Paas

```
konrad@slurm-master:~/demo_paramsweep/n4$ sbatch paramsweep_n4.batch
sbatch: error: Batch job submission failed: Memory required by task is not available
konrad@slurm-master:~/demo_paramsweep/n4$
```

- Nem megfelelő memória paraméterezés

Több node - #BATCH job paraméterek

User | Slurm_Paas

```
 GNU nano 7.2
#!/bin/bash
#SBATCH --job-name=steps_job              # Job name
#SBATCH --output=steps_%j.out             # Output file name
#SBATCH --error=steps_%j.err              # Error file name
#SBATCH --time=01:00:00                   # Time limit: 1 hour
#SBATCH --mem=4G                          # Memory per node
#SBATCH --cpus-per-task=1                 # CPU cores per task
#SBATCH --nodes=8                         # Request 4 nodes
#SBATCH --ntasks-per-node=2               # Allow 2 tasks per node (we have 8
                                          ition
                                          irement per task
```

User | Slurm_Paas

```
konrad@slurm-master:~/demo_paramsweep/n4$ sbatch paramsweep_n4.batch
sbatch: error: Batch job submission failed: Node count specification invalid
konrad@slurm-master:~/demo_paramsweep/n4$ sinfo
PARTITION                AVAIL  TIMELIMIT  NODES  STATE NODELIST
batch_cpu_m2.large          up 1-00:00:00      4   idle slurm-master,slurm-worker-cpu-m2-large-[1-3]
batch_gpu_g2.large_8*       up 1-00:00:00      7   idle slurm-worker-gpu-g2-large-[1-7]
batch_gpu_g2.xlarge_16      up 1-00:00:00      2   idle slurm-worker-gpu-g2-xlarge-[1-2]
batch_gpu_g2.2xlarge_32     up 1-00:00:00      1   idle slurm-worker-gpu-g2-2xlarge-1
interactive_cpu_m2.large    up 7-00:00:00      4   idle slurm-master,slurm-worker-cpu-m2-large-[1-3]
interactive_gpu_g2.large_8  up 7-00:00:00      7   idle slurm-worker-gpu-g2-large-[1-7]
konrad@slurm-master:~/demo_paramsweep/n4$
```

- Nem megfelelő node szám paraméterezés

Slurm használata – job steps

- Több **srun** parancs (job steps) egyetlen *.batch* feladatban
  - *for* ciklusba ágyazott **srun** parancsok
  - szekvencálisan futó srun parancsok
  - futtatott **srun** parancsok száma → nincs limit
  - potenciális erőforrás limitáció
  - amennyiben az **srun** parancsok futási ideje átlépi az időkeretet, a feladat *CANCELED* státuszba kerül

Köszönöm a figyelmet!