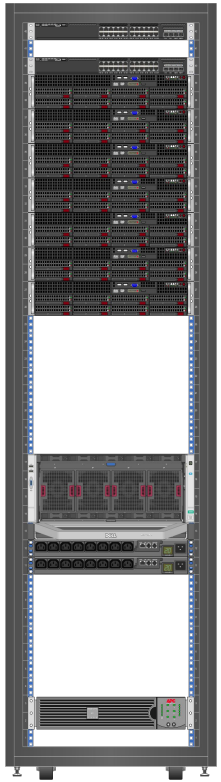
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA

UTILIZANDO SLURM EN EL CLUSTER DEL DIINF

CRISTÓBAL ACOSTA
COORDINADOR TI - DIINF - USACH

HARDWARE

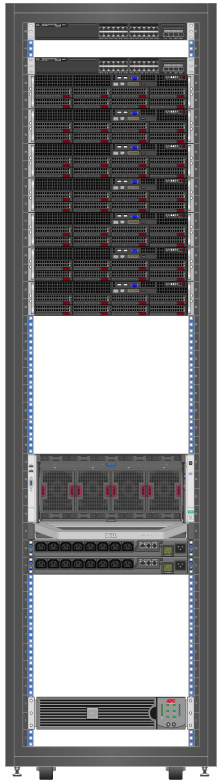


- EL CLUSTER DEL DIINF CUENTA CON 1 LOGIN NODE Y 4 COMPUTE NODES CONECTADOS EN RED USACH (ACCESO CABLEADO)

```
$ ssh username@xi.diinf.usach.cl
```

- XI NETRAIDER 64 LT SERVERS
  - BATCH PARTITION (2 NODES)
    - 2x 32-Core/64-Thread 3rd Gen AMD EPYC 7513
    - 256GB (16x16G) DDR4-3200 ECC Registered SDRAM
  - GPU PARTITION (2 NODES)
    - 1x 24-Core/48-Thread 3rd Gen AMD EPYC 7443P
    - 128GB (8x16GB) DDR4-3200 ECC Registered SDRAM
    - 2x NVIDIA A30 for PCIe-24GB HBM2-3584 CUDA, 336 Tensor Cores-PCIe 4.0 x16 /1x NVLink for A30

USER ACCCESS



- SO UBUNTU 22.04 LTS
- `/home/xi` COMPARTIDO POR NFS
  - LA CARPETA DEL USUARIO ES VISIBLE EN TODOS LOS NODOS
- DEPENDIENDO DEL REQUERIMIENTO (BATCH, GPU) UN USUARIO PUEDE ACCEDER A GPU O NO

```
$ ssh username@xi.diinf.usach.cl
$ pwd
/home/XI/username
$ sinfo --format="%.10P %.10D %.15N"
 PARTITION      NODES         NODELIST
    batch*          2    xicpu[02-03]
       GPU          2    xigpu[01-02]
```

SOFTWARE

- SLURM 22.05.2
- OPENMPI 4.0.4
- C/C++ 11.3.0
- PYTHON 3.10.6
- ~~MATLAB R2021A~~
- ~~COMSOL MULTIPHYSICS - MODULE AC/DC~~
- NVIDIA DRIVER 525-125.06 - CUDA 12.0
- NVIDIA HPC-SDK 22.1

CLUSTER MANAGEMENT AND JOB SCHEDULING SYSTEM FOR [...] LINUX CLUSTERS*

- NOS PERMITE ENCOLAR N TRABAJOS EN EL CLUSTER
- SLURM SE PREOCUPA DE PLANIFICARLOS (CUÁNDO Y DÓNDE)

(*) Quick Start User Guide: https://slurm.schedmd.com/quickstart.html

SLURM: DEFINICIONES

- **NODE**
  - RECURSO COMPUTACIONAL (AKA SERVIDOR)
- **PARTITION**
  - CONJUNTO LÓGICO DE NODOS (UN NODO PUEDE PERTENECER A VARIAS PARTICIONES)
  - PUEDE CONSIDERARSE COMO UNA COLA (QUEUE) CON RESTRICCIONES PARA LOS JOBS
- **JOB**
  - CANTIDAD DE RECURSOS ASIGNADOS A UN USUARIO POR UN DETERMINADO TIEMPO
- **JOB STEP**
  - CONJUNTO DE TAREAS (POSIBLEMENTE PARALELAS) EN UN JOB

(*) Quick Start User Guide: https://slurm.schedmd.com/quickstart.html

SLURM: COMANDOS

- **SINFO**
  - INFORMACIÓN SOBRE NODOS Y PARTICIONES
- **SRUN**
  - PLANIFICAR UN JOB O UN JOB STEP
- **SQUEUE**
  - CONSULTAR POR JOBS PLANIFICADOS EN LAS PARTICIONES DE SLURM
- **SCANCEL**
  - PARA SEÑALIZAR UN JOB O UN JOB STEP (CANCELARLA)
- **SBATCH**
  - CORRER UN SCRIPT

(*) Quick Start User Guide: https://slurm.schedmd.com/quickstart.html

SINFO

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
batch*       up   infinite      2   idle xicpu[02-03]
GPU          up   infinite      2   down xigpu[01-02]

$ sinfo -R # --list-reasons
REASON               USER      TIMESTAMP           NODELIST
En mantencion        root      2023-07-04T11:38:47 xigpu[01-02]

$ sinfo --format="%.10P %.10a %.10D %.10T %.15N"
 PARTITION      AVAIL      NODES      STATE        NODELIST
    batch*         up          2       idle    xicpu[02-03]
       GPU         up          2       down    xigpu[01-02]
```

https://slurm.schedmd.com/sinfo.html

SRUN

```
$ srun /bin/hostname
xicpu02

$ srun -N 2 /bin/cat /etc/hostname # --nodes=<minnodes-[maxnodes]>
xicpu02
xicpu03

$ srun -N 2 /bin/false
srun: error: xicpu02: task 0: Exited with exit code 1
srun: error: xicpu03: task 1: Exited with exit code 1

$ srun -N 2 -n 3 /bin/hostname # --nodes=<minnodes-[maxnodes]> --ntasks=<number>
xicpu03
xicpu02
xicpu02
```

https://slurm.schedmd.com/srun.html

SQUEUE

```
$ srun -N 2 -n 5 -J hang01 hang & # int main(){while(1);} # --job-name=<name>
$ srun -N 2 -n 5 -J hang02 hang &
```

```
$ squeue --format="%.6i%.13P%.13j%.10u%.10T%.10M%.15l%.10D%.20R"

JOBID PARTITION NAME    USER       STATE   TIME  TIME_LIMIT  NODES  NODELIST(REASON)
504   batch     hang01  slurmtest  RUNNING 3:56  UNLIMITED   2      xicpu[02-03]
505   batch     hang02  slurmtest  RUNNING 3:53  UNLIMITED   2      xicpu[02-03]

$ squeue
JOBID PARTITION     NAME     USER      ST     TIME   NODES NODELIST(REASON)
504   batch         hang01   slurmtes  R      24:54  2     xicpu[02-03]
505   batch         hang02   slurmtes  R      24:51  2     xicpu[02-03]
```

https://slurm.schedmd.com/squeue.html

SCANCEL

```
$ srun -N 2 -n 5 -J hang01 hang & # int main(){while(1);} # --job-name=<name>
$ srun -N 2 -n 5 -J hang02 hang &
```

```
$ scancel 504
srun: Job step aborted: Waiting up to 32 seconds for job step to finish.
slurmstepd: error: *** STEP 504.0 ON xicpu02 CANCELLED AT 2023-07-06T11:24:37 ***
srun: error: xicpu03: task 4: Terminated
srun: error: xicpu02: tasks 0-3: Terminated

[1]-  Exit 143                srun -N 2 -n 5 -J hang01 hang
```

https://slurm.schedmd.com/scancel.html

SALLOC

```
# -p, --partition=<partition_names>
# -N, --nodes=<minnodes>[-maxnodes]|<size_string>
# -t, --time=<time>
# --mem-per-cpu=<size>[units]
$ salloc -p GPU -N 1 -t 01:00:00 --mem-per-cpu=1G
salloc: Granted job allocation 488
salloc: Waiting for resource configuration
salloc: Nodes xigpu01 are ready for job

$ nvidia-smi -L # login node
NVIDIA-SMI has failed …

$ srun nvidia-smi -L # compute node con GPU
GPU 0: NVIDIA A30 (UUID: GPU-3404b858-1ee2-2e76-98be-a539d0b89ebd)
```

https://slurm.schedmd.com/salloc.html

SACCT

- UNA VEZ FINALIZADA LA TAREA, ES IMPORTANTE DEVOLVER LOS RECURSOS
  - VENTAJA (PARA EL USUARIO): LAS TAREAS CORREN INMEDIATAMENTE EN LOS RECURSOS ASIGNADOS
  - DESVENTAJA (PARA NOSOTROS): LOS RECURSOS DEBEN LIBERARSE (EXIT, SCANCEL)

```
$ sacct # displays accounting data for all jobs and job steps...
JobID           JobName  Partition    Account  AllocCPUS      State ExitCode
------------ ---------- ---------- ---------- ---------- ---------- --------
488          interacti+        GPU    default          2    RUNNING      0:0
488.extern       extern               default          2    RUNNING      0:0
488.0        nvidia-smi               default          2  COMPLETED      0:0
488.1          hostname               default          2  COMPLETED      0:0
488.2             false               default          2     FAILED      1:0
```

https://slurm.schedmd.com/sacct.html

SCONTROL

```
# node, partition, job, reservation
$ scontrol show node xigpu02 # <ENTITY>[=<ID>] or <ENTITY> [<ID>]
NodeName=xigpu02 Arch=x86_64 CoresPerSocket=24
   CPUAlloc=0 CPUEfctv=48 CPUTot=48 CPULoad=0.16
   AvailableFeatures=(null)
   ActiveFeatures=(null)
   Gres=gpu:A30:2(S:0)
   NodeAddr=xigpu02 NodeHostName=xigpu02 Version=22.05.2
   OS=Linux 5.15.0-76-generic #83-Ubuntu SMP Thu Jun 15 19:16:32 UTC 2023
   RealMemory=122240 AllocMem=0 FreeMem=126558 Sockets=1 Boards=1
   State=IDLE ThreadsPerCore=2 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=GPU
   BootTime=2023-07-11T09:11:35 SlurmdStartTime=2023-07-11T09:11:54
   LastBusyTime=2023-07-11T09:11:54
   CfgTRES=cpu=48,mem=122240M,billing=48,gres/gpu=2 ...
```

https://slurm.schedmd.com/scontrol.html

SBATCH

```
$ sbatch job.slurm
Submitted batch job 509
```

```
#!/bin/bash
#SBATCH --job-name=test
#SBATCH --partition=batch
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --output=job-%u-%x-%A.out # job-slurmtest-test-509.out
#SBATCH --error=job-%A.err

/bin/hostname
```

https://slurm.schedmd.com/sbatch.html

JOB ARRAY

- UN JOB ARRAY PERMITE SISTEMATIZAR LA CREACIÓN DE JOBS
  - OPCIÓN --ARRAY=1-30, --ARRAY=1,3,5,7, -ARRAY=1-7:2 (1,3,5,7)
  - EXISTE LA VARIABLE DE AMBIENTE SLURM_ARRAY_TASK_ID

```
#!/bin/bash
#SBATCH --job-name=job_array
#SBATCH --partition=GPU
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --array=1-7:2 # Se crean 4 jobs
#SBATCH --output=job-%A_%a.out # job-523_1.out, job-523_3.out, job-523_5.out...
#SBATCH --error=job-%A_%a.err # job-523_1.err, # job-523_3.err, # job-523_5.err...

/bin/hostname
```

https://slurm.schedmd.com/job_array.html

CONTENEDORES

- ES POSIBLE SOMETER JOBS EN CONTENEDORES
  - NVIDIA/ENROOT
    - *A SIMPLE, YET POWERFUL TOOL TO TURN TRADITIONAL CONTAINER/OS IMAGES INTO UNPRIVILEGED SANDBOXES*
    - EN GENERAL, LOS CONTENEDORES SE UTILIZAN PARA ENCAPSULAR, ESTO NO FUNCIONA BIEN EN HPC
    - ENROOT MODIFICA IMÁGENES PARA CREAR CONTENEDORES "NO TAN AISLADOS"
  - NVIDIA/PYXIS
    - *PYXIS IS A SPANK PLUGIN FOR THE SLURM WORKLOAD MANAGER. IT ALLOWS UNPRIVILEGED CLUSTER USERS TO RUN CONTAINERIZED TASKS THROUGH THE SRUN COMMAND*
    - SE AGREGAN NUEVAS OPCIONES PARA UTILIZAR CON SRUN

```
$ srun --container-image=python:3.11.4 -p batch python3 --version
pyxis: importing docker image ...
Python 3.11.4
```

https://github.com/NVIDIA/enroot
https://github.com/NVIDIA/pyxis

CONTENEDORES

```
$ srun --container-image=alpine:3.18.2 -p batch grep PRETTY /etc/os-release
pyxis: importing docker image ...
PRETTY_NAME="Alpine Linux v3.18"

$ srun --container-image=alpine:3.18.2 -p batch \
  --container-mounts=/etc/os-release:/host/os-release \
  grep PRETTY /host/os-release
pyxis: importing docker image ...
PRETTY_NAME="Ubuntu 22.04.2 LTS"

$ srun --container-image=alpine:3.18.2 -p batch \
  --container-save="${HOME}/enroot_images/alpine:3.18.2.sqsh" \
  grep PRETTY /etc/os-release
pyxis: importing docker image ...
PRETTY_NAME="Alpine Linux v3.18"
```

https://github.com/NVIDIA/pyxis

CONTENEDORES

```
# Acá utilizamos una imagen almacenada en ${HOME}/path/to/the/image
$ srun --container-image="${HOME}/enroot_images/alpine:3.18.2.sqsh" -p batch \
  grep PRETTY /etc/os-release
PRETTY_NAME="Alpine Linux v3.18"
```

https://github.com/NVIDIA/pyxis

SBATCH Y CONTENEDORES

```
$ sbatch container-job.slurm
Submitted batch job 584
```

```
#!/bin/bash
#SBATCH --job-name=container-job
#SBATCH --partition=GPU
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --gres=gpu:A30:1 # -G 1
#SBATCH --output=job-%A.out
#SBATCH --error=job-%A.err

srun \
--container-image="${HOME}/enroot_images/nvidia+cuda+12.2.0-base-ubuntu22.04.sqsh" \
nvidia-smi
```

DEMO

ENTRENANDO UN CLASIFICADOR UTILIZANDO GPU Y CONTENEDORES CON SLURM

ENTRENANDO UN CLASIFICADOR

- **DATASET: MINIST**
  - A LARGE DATABASE OF **HANDWRITTEN DIGITS** THAT IS COMMONLY USED FOR TRAINING VARIOUS IMAGE PROCESSING SYSTEMS
- **MODELO: TF.KERAS.MODELS.SEQUENTIAL**
  - SE OBTIENE LA DATA
  - SE DIVIDE EN DATA DE ENTRENAMIENTO Y DATA PARA TESTEAR
  - SE CREA EL MODELO Y SE ENTRENA
  - SE TESTEA EL RESULTADO

https://en.wikipedia.org/wiki/MNIST_database
https://www.tensorflow.org/tutorials/quickstart/beginner

USACH

JULIO - 2023

Comisión Nacional de Acreditación CNA-Chile

7 universidad acreditada años

ÁREA DE GESTIÓN INSTITUCIONAL - ÁREA DE DOCENCIA DE PREGRADO - ÁREA DE INVESTIGACIÓN
ÁREA DE DOCENCIA DE POSTGRADO - ÁREA DE VINCULACIÓN CON EL MEDIO - HASTA FEBRERO 2028