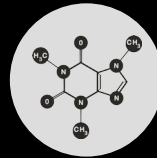


# Grupo de Ciencia Computacional HIMFG

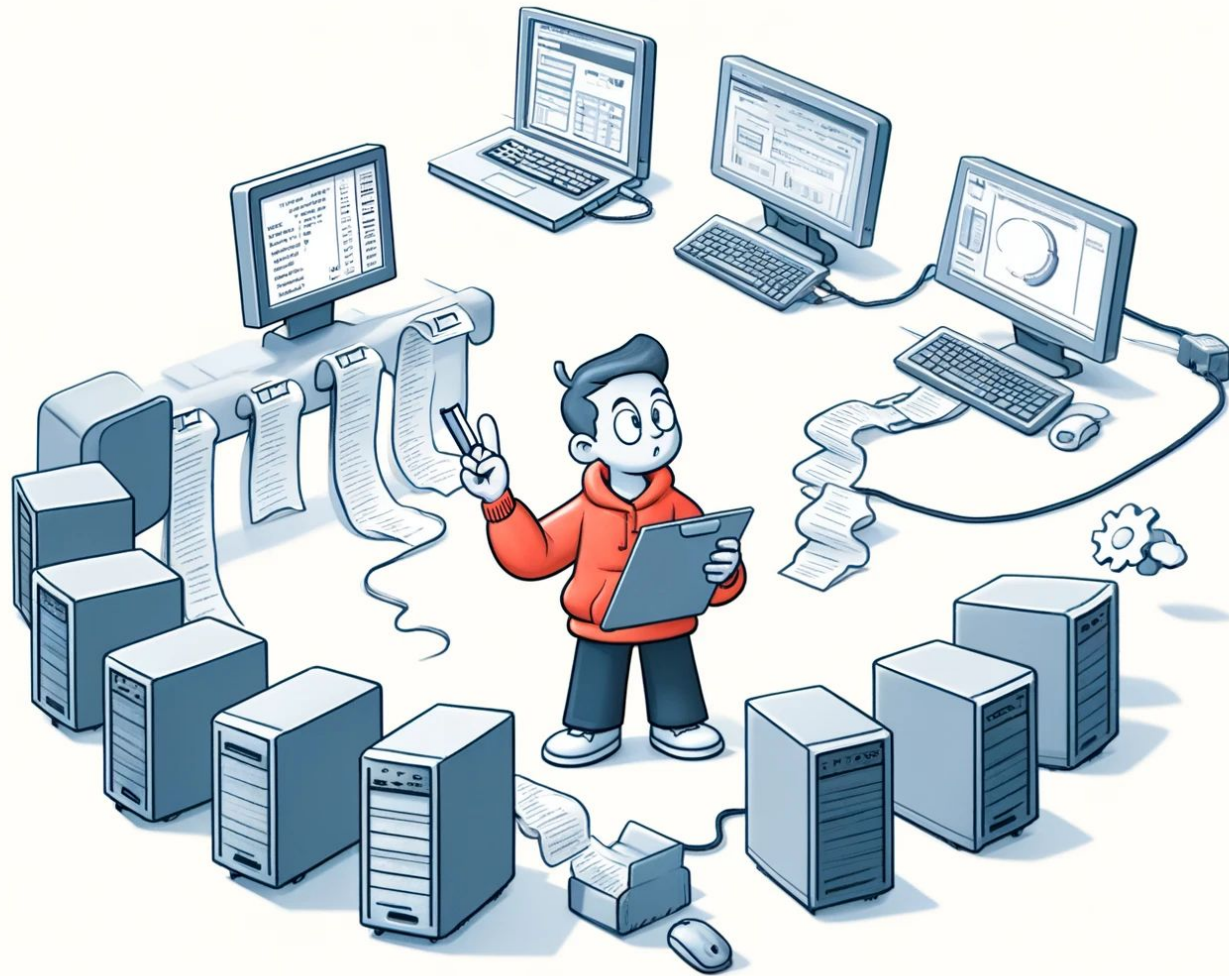


# Simple Linux Utility for Resources Management (Slurm)

¿Qué es un gestor de tareas y recursos?  
¿Cómo puede ayudarnos en el laboratorio?



**¿Qué es un gestor de  
tareas y recursos?**



**¿Para qué nos puede  
ayudar en el laboratorio?**

- Hacer un uso eficiente de los recursos.
- Ejecutar una lista de trabajos secuencialmente.
- Gestionar la ejecución de trabajos de distintos usuarios y grupos.

**¿Cómo se instala y  
administra?**







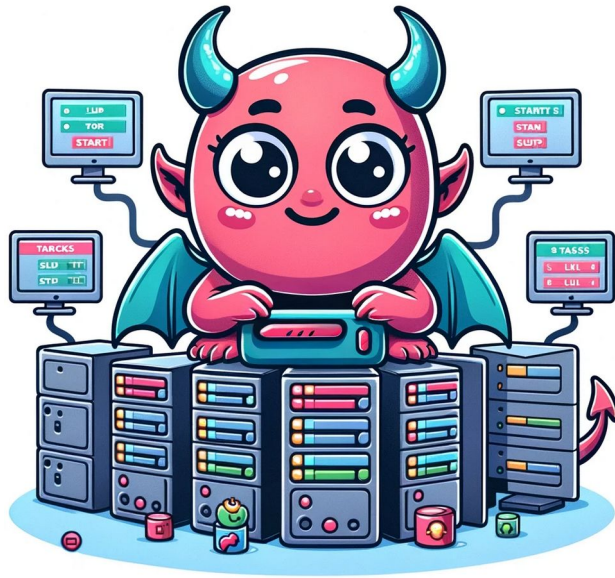
# Instalación para Ubuntu

```
sudo apt install slurmd slurmctld
```

```
diego@aleph: ~  
diego@aleph:~$ sudo apt install slurmd slurmctld -y  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  slurmctld slurmd  
0 upgraded, 2 newly installed, 0 to remove and 6 not upgraded.  
Need to get 737 kB of archives.  
After this operation, 2 013 kB of additional disk space will be used.  
Get:1 https://esm.ubuntu.com/apps/ubuntu jammy-apps-security/main amd64 slurmctld amd64 21.08.5-2ubu  
ntu1+esm1 [526 kB]  
Get:2 https://esm.ubuntu.com/apps/ubuntu jammy-apps-security/main amd64 slurmd amd64 21.08.5-2ubuntu  
1+esm1 [211 kB]  
Fetched 737 kB in 2s (333 kB/s)  
Selecting previously unselected package slurmctld.  
(Reading database ... 642138 files and directories currently installed.)  
Preparing to unpack .../slurmctld_21.08.5-2ubuntu1+esm1_amd64.deb ...  
Unpacking slurmctld (21.08.5-2ubuntu1+esm1) ...  
Selecting previously unselected package slurmd.  
Preparing to unpack .../slurmd_21.08.5-2ubuntu1+esm1_amd64.deb ...  
Unpacking slurmd (21.08.5-2ubuntu1+esm1) ...  
Setting up slurmctld (21.08.5-2ubuntu1+esm1) ...  
update-alternatives: using /usr/sbin/slurmctld-wlm to provide /usr/sbin/slurmctld (slurmctld) in aut  
o mode  
Setting up slurmd (21.08.5-2ubuntu1+esm1) ...  
update-alternatives: using /usr/sbin/slurmd-wlm to provide /usr/sbin/slurmd (slurmd) in auto mode  
Processing triggers for ureadahead (0.100.0-21) ...  
Processing triggers for man-db (2.10.2-1) ...  
diego@aleph:~$
```

**¿Pero qué hemos instalado?**

# ¿Pero qué hemos instalado?



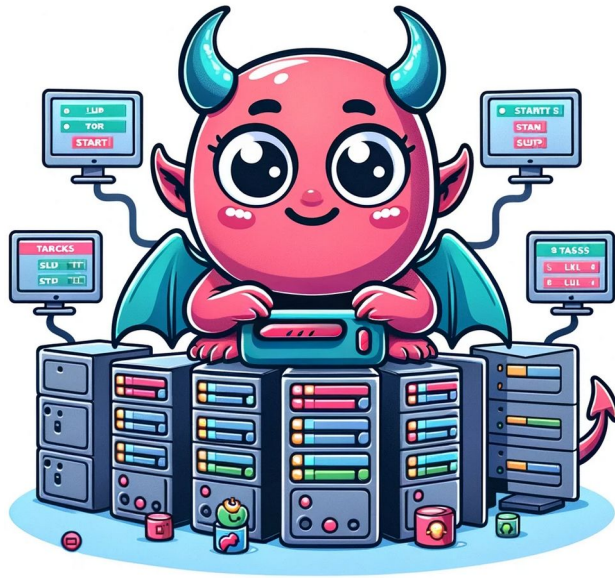
*Slurmctld*



*Slurmd*



# ¿Pero qué hemos instalado?



*Slurmctld*

- Controla y gestiona los nodos y sus recursos hardware
- Determina qué trabajos, cuándo y donde se ejecutan.
- Gestiona la cola de trabajos

# ¿Pero qué hemos instalado?

- Inicia, supervisa y finaliza los trabajos
- Le comunica a Slurmctld el estado de cada nodo
- Detecta y reporta fallos a Slurmctld en la ejecución de los trabajos



*Slurmd*

# Configuración de Slurm

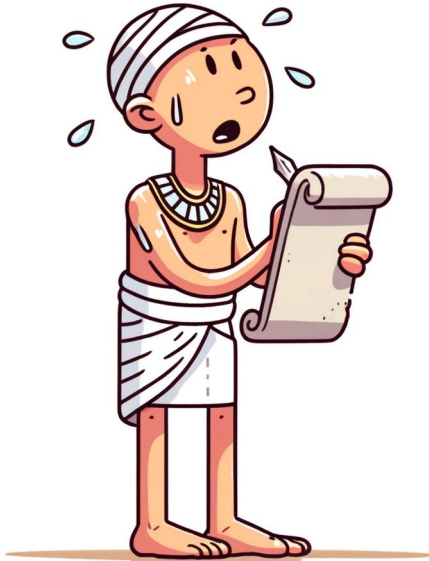
*(Una sólo máquina)*



*Fichero `/etc/slurm/slurm.conf`*

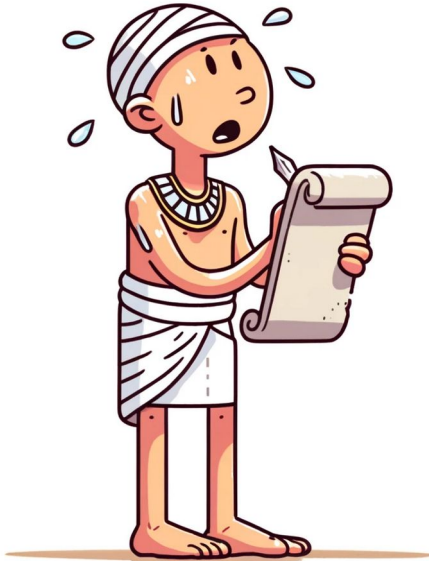


# /etc/slurm/slurm.conf



```
# slurm.conf file generated by configurator.html.  
# See the slurm.conf man page for more information.  
#  
ClusterName=Hobbes  
SlurmctldHost=localhost  
MpiDefault=none  
ProctrackType=proctrack/linuxproc  
ReturnToService=2  
SlurmctldPidFile=/var/run/slurmctld.pid  
SlurmctldPort=6817  
SlurmdPidFile=/var/run/slurmd.pid  
SlurmdPort=6818  
SlurmdSpoolDir=/var/lib/slurm/slurmd  
SlurmUser=slurm  
StateSaveLocation=/var/lib/slurm/slurmctld  
SwitchType=switch/none  
TaskPlugin=task/none
```

# /etc/slurm/slurm.conf



## # TIMERS

InactiveLimit=0

KillWait=30

MinJobAge=300

SlurmctldTimeout=120

SlurmdTimeout=300

Waittime=0

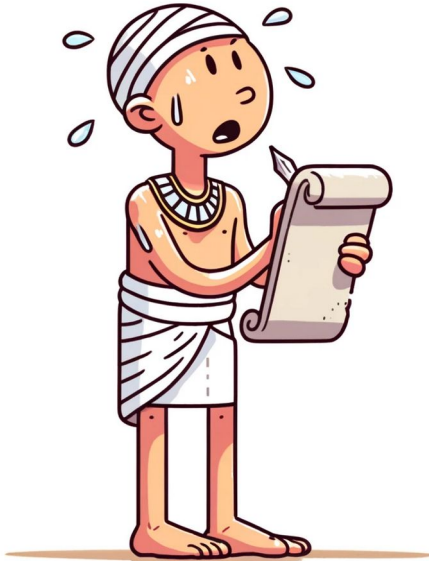
## # SCHEDULING

SchedulerType=sched/backfill

SelectType=select/cons\_tres

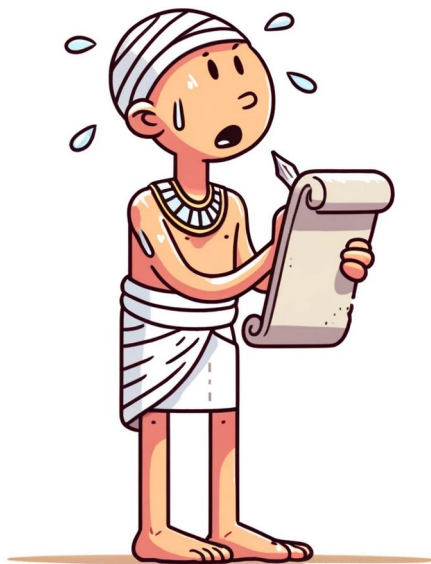
SelectTypeParameters=CR\_Core

# /etc/slurm/slurm.conf



```
# LOGGING AND ACCOUNTING
#AccountingStoragePort=
AccountingStorageType=accounting_storage/none
JobCompType=jobcomp/none
JobAcctGatherFrequency=30
JobAcctGatherType=jobacct_gather/none
SlurmctldDebug=info
SlurmctldLogFile=/var/log/slurm/slurmctld.log
SlurmdDebug=info
SlurmdLogFile=/var/log/slurm/slurmd.log
```

# /etc/slurm/slurm.conf



## # COMPUTE NODES

NodeName=localhost CPUs=12 RealMemory=30517  
State=UNKNOWN

## # PARTITIONS

PartitionName=tests Nodes=localhost Default=YES  
MaxTime=2:00:00 State=UP MaxCPUsPerNode=2  
PriorityJobFactor=10000 PriorityTier=65000

PartitionName=production Nodes=localhost  
Default=YES MaxTime=24:00:00 State=UP  
MaxCPUsPerNode=10 PriorityJobFactor=5000

PartitionName=priority Nodes=localhost Default=YES  
MaxTime=INFINITE State=UP  
PriorityJobFactor=65000 PriorityTier=65000  
AllowGroups=lab\_vips

# `/etc/slurm/slurm.conf`

## *(Un cluster de varias máquinas)*

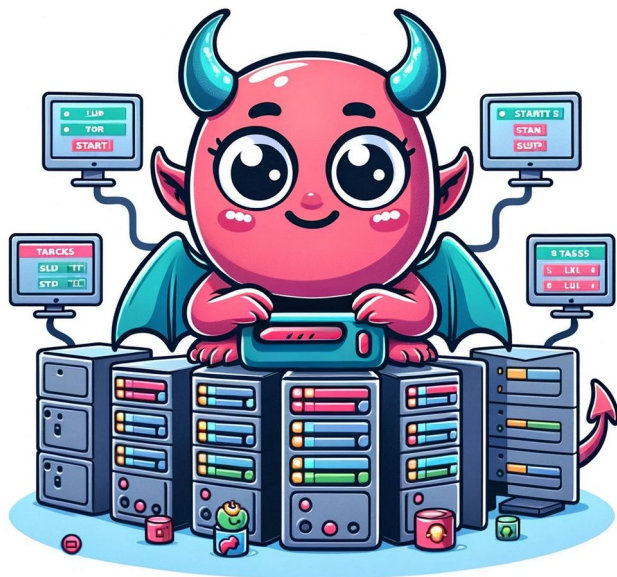
### # COMPUTE NODES

```
NodeName=ixtlilton NodeAddr=192.168.0.100 CPUs=20 RealMemory=62000 Sockets=2 CoresPerSocket=10  
ThreadsPerCore=1 Gres=gpu:GTX1080Ti:3 State=UNKNOWN  
NodeName=node01 NodeAddr=192.168.0.1 CPUs=20 RealMemory=62000 Sockets=2 CoresPerSocket=10 ThreadsPerCore=1  
Gres=gpu:RTX2080Ti:3 State=UNKNOWN  
NodeName=node02 NodeAddr=192.168.0.2 CPUs=20 RealMemory=62000 Sockets=2 CoresPerSocket=10 ThreadsPerCore=1  
Gres=gpu:RTX2080Ti:2,gpu:GTX1080Ti:1 State=UNKNOWN  
NodeName=node03 NodeAddr=192.168.0.3 CPUs=20 RealMemory=52000 Sockets=2 CoresPerSocket=10 ThreadsPerCore=1  
Gres=gpu:RTX2080Ti:2,gpu:GTX1080Ti:1 State=UNKNOWN  
NodeName=node04 NodeAddr=192.168.0.4 CPUs=20 RealMemory=62000 Sockets=2 CoresPerSocket=10 ThreadsPerCore=1  
Gres=gpu:GTX1080Ti:3 State=UNKNOWN
```

### # PARTITIONS

```
PartitionName=tests Nodes=ixtlilton Default=YES PriorityJobFactor=10000 MaxTime=3:00:00 State=UP OverSubscribe=NO  
Shared=Yes AllowGroups=uibcdf  
PartitionName=normal Nodes=node01,node02,node03,node04 Default=NO PriorityJobFactor=5000 MaxTime=24:00:00  
State=UP OverSubscribe=NO Shared=Yes AllowGroups=uibcdf  
PartitionName=prior Nodes=ixtlilton,node01,node02,node03,node04 Default=NO PriorityJobFactor=65000 PriorityTier=65000  
MaxTime=INFINITE State=UP OverSubscribe=NO Shared=Yes AllowGroups=uibcdf_priority
```

# Iniciamos los demonios...



```
sudo systemctl start slurmctld && sudo systemctl start slurmd
```

**iiiYa podemos usar Slurm!!!**

**Pero... ¿Cómo?**

# Inspeccionando las particiones y colas de jobs...



**sinfo:** Vista de la información de los nodos y particiones.  
**squeue:** Vista de los trabajos en cola.



# Un script de lanzamiento...

## **job.sh**

```
#!/bin/bash
#SBATCH --job-name=mi_trabajo      # Nombre del trabajo
#SBATCH --output=mi_trabajo_%j.log  # Nombre del archivo de salida (%j se reemplaza por el jobID)
#SBATCH --error=mi_trabajo_%j.err   # Nombre del archivo de errores (%j se reemplaza por el jobID)
#SBATCH --ntasks=1                  # Número de tareas (procesos)
#SBATCH --cpus-per-task=2           # Número de CPUs por tarea
#SBATCH --time=00:01:00             # Tiempo máximo de ejecución (en este caso, 1 minuto)
#SBATCH --partition=priority        # Partición (cola) a la que se enviará el trabajo

# Ejecutar el comando
python mi_script.py
```

# Enviando un trabajo a una cola...



```
sbatch job.sh
```

# ¿Te vas a atrever...?

Más documentación y foro técnico de soporte en:  
[github.com/Ciencia-Computacional-HIMFG/Slurm](https://github.com/Ciencia-Computacional-HIMFG/Slurm)