

## slurm.conf

1. **DefaultPartition / PartitionDefault:** Yeh default partition ka naam hota hai jo cluster mein job scheduling ke liye set hota hai. Agar aapne partition specify nahi kiya, toh yeh partition use hota hai.
2. **DefaultQueue:** Yeh queue ka naam hai jo default ke roop mein set hota hai, jisme jobs by default submit ki jaati hain.
3. **PartitionType:** Yeh partition ka type define karta hai, jaise ki compute, interactive, etc.
4. **MaxTime / PartitionTimeLimit:** Yeh limit specify karta hai ki ek job kitni der chal sakti hai, maximum time limit ke roop mein.
5. **TimeLimit / MaxWallTime:** Yeh job ki total execution time ki limit hoti hai, jo system ne set ki hoti hai.
6. **MaxJobsPerUser:** Yeh maximum number of jobs define karta hai jo ek user ek time pe submit kar sakta hai.
7. **MaxConcurrentJobs:** Yeh limit set karta hai ki ek user kitni jobs simultaneously run kar sakta hai.
8. **JobLimit / UserMaxJobs:** Yeh limit define karta hai ki ek user kitni total jobs submit kar sakta hai, chahe wo running ho ya pending.
9. **PriorityType / SchedulingPolicy:** Yeh job scheduling ki policy ko define karta hai, jisme priorities ka use hota hai.
10. **JobPriority / PriorityWeight:** Yeh job ki priority ko define karta hai. Jitni zyada priority weight, utni jaldi job schedule hogi.

## SLURM Commands:

1. **scontrol update node:** Node ki properties ko update karta hai, jaise resources ya status.
2. **scontrol set state:** Node ya job ki state change karta hai, jaise idle, allocated, etc.
3. **scontrol modify node:** Node ke settings ya attributes ko modify karta hai.
4. **scontrol reconfigure:** SLURM configuration ko re-load karta hai without restarting SLURM daemons.
5. **sinfo:** Cluster ke available resources ko dekhne ka command.
6. **scontrol show node :** Specific node ka status aur information dikhata hai.
7. **squeue -N :** Specific node pe jobs ka status show karta hai.
8. **slurmd status :** Node pe SLURM daemon ka status check karta hai.

## SLURM Job Directives:

1. **#SBATCH --time / #SBATCH --duration / #SBATCH --walltime / #SBATCH --max-time:** Yeh job submission mein time duration specify karte hain. Isse aap job ki maximum running time set kar sakte hain.

Yeh sab terms aur commands SLURM job scheduler ko manage karte hain aur cluster resources ko efficiently handle karne mein madad karte hain.

#### 1. **#SBATCH --cpus:**

- Isse aap job ke liye total number of CPUs specify karte hain. Agar aapne `--cpus` set kiya hai, toh yeh batata hai ki job ko kitne CPUs chahiye.
- Example: Agar aapko ek job ko 8 CPUs chahiye, toh aap `#SBATCH --cpus=8` likhenge.

#### 1. **#SBATCH --cpus-per-task:**

- Yeh specify karta hai ki har task ko kitne CPUs chahiye. Agar aapka job multiple tasks ko run karega, toh yeh har task ke liye CPU count define karta hai.
- Example: Agar aapke job mein har task ko 4 CPUs chahiye, toh `#SBATCH --cpus-per-task=4` use karenge.

#### 1. **#SBATCH --tasks-per-node:**

- Yeh directive define karta hai ki har node pe kitne tasks run karne chahiye. Agar aap multiple nodes use kar rahe hain, toh har node pe kitne tasks run karne hain, yeh specify hota hai.
- Example: Agar aap chahte hain ki har node pe 2 tasks run ho, toh `#SBATCH --tasks-per-node=2` likhenge.

#### 1. **#SBATCH --cpu-limit:**

- Yeh limit set karta hai ki ek job ko maximum kitne CPUs allocate kiye ja sakte hain. Yeh limit aapke job ke total CPU usage ko restrict karta hai.
- Example: Agar aap chahte hain ki job ko 16 CPUs se zyada allocate na ho, toh `#SBATCH --cpu-limit=16` likhenge.

In short, yeh directives aapko apne job ke liye CPUs aur tasks manage karne mein madad karte hain, taaki aapke jobs efficiently run ho sakein without resource wastage.

#### 1. **#SBATCH --mem-per-node:**

- Isse aap specify karte hain ki har node ko kitni memory chahiye. Agar aap multiple nodes use kar rahe hain, toh har node ke liye kitni memory allocate karni hai, yeh define hota hai.
- Example: Agar har node ko 16 GB memory chahiye, toh `#SBATCH --mem-per-node=16G` likhenge.

#### 1. **#SBATCH --memory:**

- Yeh aapke job ke liye total memory allocate karta hai. Yeh memory requirement job ke liye ek overall limit set karta hai.
- Example: Agar aap apne job ko 32 GB memory allocate karna chahte hain, toh `#SBATCH --memory=32G` likhenge.

### 1. **#SBATCH --mem:**

- Yeh bhi memory allocation ke liye use hota hai aur `--memory` ke equivalent hai. Iska purpose same hai, jo total memory requirement ko specify karta hai.
- Example: `#SBATCH --mem=8G` likhna matlab job ke liye 8 GB memory allocate karna.

### 1. **#SBATCH --max-memory:**

- Yeh directive job ki maximum memory limit specify karta hai. Agar job memory limit exceed karta hai, toh job ko terminate kar diya jaata hai.
- Example: Agar aap chahte hain ki job ko zyada se zyada 64 GB memory mile, toh `#SBATCH --max-memory=64G` likhenge.

### **Summary:**

- **--mem-per-node:** Har node ke liye memory allocation.
- **--memory / --mem:** Total memory allocation job ke liye.
- **--max-memory:** Maximum memory limit set karta hai jo job use kar sakta hai.

Yeh directives aapke job ko memory efficiently allocate karne mein madad karte hain, taaki aapka job crash na ho aur resources ka optimal use ho.

Here's a brief explanation of each SLURM priority-related directive in Hinglish:

### 1. **#SBATCH --priority:**

- Yeh directive job ki priority ko set karta hai. Agar aap ek job ko higher priority dena chahte hain, toh isse use karte hain. Higher priority jobs ko pehle schedule kiya jaata hai.
- Example: `#SBATCH --priority=1000` likhna matlab aapke job ki priority ko 1000 set kiya gaya hai.

### 1. **#SBATCH --priority-weight:**

- Yeh weight assign karta hai job ki priority ko, jise SLURM scheduling decision mein use karta hai. Zyada weight ka matlab zyada priority. Yeh weight normally scheduling policies ke hisaab se kaam karta hai.
- Example: `#SBATCH --priority-weight=10` ka matlab job ki priority ko 10 weight assign kiya gaya hai.

### 1. **#SBATCH --job-priority:**

- Yeh specific job ke liye priority level ko define karta hai. Yeh job ko kisi particular priority queue mein daal ke uske execution ko affect karta hai.
- Example: `#SBATCH --job-priority=2000` likhna matlab job ko high priority dikhana.

### 1. **#SBATCH --job-size:**

- Yeh directive job ke size ko define karta hai, jaise ki job ki resource requirements (CPUs, memory, etc.). Yeh job ki scheduling ko influence karta hai ki kitni resources ki zarurat hai job ko properly run karne ke liye.
- Example: `#SBATCH --job-size=large` likhna matlab job ko high resource demand wala job mark karna.

## Summary:

- **--priority**: Directly job ki priority set karta hai.
- **--priority-weight**: Job ki priority ke liye weight assign karta hai, jisse scheduling pe impact padta hai.
- **--job-priority**: Job ke liye specific priority level set karta hai.
- **--job-size**: Job ke resource requirements ko define karta hai.

Yeh directives aapke jobs ki scheduling aur execution ko influence karte hain, taaki aapke job ko zaroori resources mil sakein aur scheduling efficient ho.

Here's a brief explanation of each SLURM command in Hinglish:

### 1. **scontrol**:

- Yeh command SLURM ke internal control ko manage karne ke liye use hota hai. Isse aap jobs, nodes, partitions, aur SLURM configurations ko modify kar sakte hain.
- Example: `scontrol show job <job-id>` - Isse aap specific job ka detailed status dekh sakte hain.

### 1. **sstat**:

- Yeh command running jobs ke status ko dekhne ke liye use hota hai. Isse aap ek job ke performance metrics jaise ki CPU usage, memory usage, etc., ko check kar sakte hain.
- Example: `sstat -j <job-id> jobinfo` - Yeh command specific job ki detailed performance information dikhata hai.

### 1. **squeue**:

- Yeh command SLURM ke queue ko check karne ke liye use hota hai. Isse aap job status, job IDs, job owners, aur pending jobs ki details dekh sakte hain.
- Example: `squeue` - Yeh sabhi jobs ka status show karega, including running, pending, aur completed jobs.
- Example: `squeue -u <user-name>` - Specific user ke jobs ka status show karta hai.

### 1. **sinfo**:

- Yeh command cluster ki resources aur nodes ki availability ko dekhne ke liye use hota hai. Isse aap dekh sakte hain ki kaunse nodes available hain, unka state kya hai, aur kis partition mein resources available hain.
- Example: `sinfo` - Yeh cluster ke nodes aur partitions ka general status dikhata hai.
- Example: `sinfo -N` - Yeh specific nodes ke status dikhata hai.

## Summary:

- **scontrol**: SLURM ke internal resources aur jobs ko manage aur control karta hai.
- **sstat**: Running jobs ki performance aur resource usage ko show karta hai.
- **squeue**: Jobs ke queue aur unke statuses ko display karta hai.
- **sinfo**: Cluster resources, nodes, aur partitions ki availability ko check karta hai.

Yeh commands aapko SLURM cluster ko monitor aur manage karne mein madad karte hain.

In Slurm, a workload manager for high-performance computing (HPC) clusters, several configuration files are used to define and control the behavior of Slurm daemons and the cluster itself. The primary configuration files are:

## 1. **slurm.conf**

This is the main configuration file for Slurm, and it is crucial for the operation of the system. It is typically located in `/etc/slurm/` or `/etc/` directory.

It contains parameters defining the cluster's resources (e.g., nodes, partitions), scheduling policies, job priorities, and various Slurm daemon settings.

### **Key entries include:**

- `ClusterName`
- `ControlMachine`
- `SlurmdPort`
- `SchedulerType`
- `NodeName`
- `PartitionName`

## 2. **slurmdbd.conf**

This file is used for Slurm's database daemon (`slurmdbd`), which stores job accounting data. It is typically located in `/etc/slurm/`.

The `slurmdbd.conf` file configures how Slurm interacts with a database (often MySQL or MariaDB) for accounting purposes.

### **Key entries include:**

- `DbdHost`
- `AuthType`
- `StoragePass`
- `StorageUser`

## 3. **slurmd.conf**

This file is used for configuring the Slurm daemon that runs on compute nodes (slurmd). It is generally located in `/etc/slurm/` or `/etc/` on each node.

This configuration defines the behavior of the compute node's Slurm daemon, such as the communication settings between slurmd and slurmctld.

**Key entries include:**

- `NodeName`
- `SlurmdPort`
- `SlurmctldHost`

## **4. cgroup.conf**

This file is used for controlling cgroup settings in Slurm. It defines parameters related to the Linux cgroup resource management, which helps manage CPU, memory, and other resources on compute nodes.

It's typically found in `/etc/slurm/` if cgroup integration is enabled.

**Key entries include:**

- `CgroupAutomount`
- `ConstrainCores`
- `ConstrainMemory`