Session 1: Batch Systems, Resource Manager, and Scheduler

1. **Batch Systems:** Batch systems ek type ka computing system hai jisme multiple tasks ya jobs ko ek queue mein rakha jata hai aur phir sequentially process kiya jata hai. Matlab, users apni jobs ko submit karte hain, aur system automatic un jobs ko execute karta hai bina real-time interaction ke.

Example: Agar aapko kai files ko process karna hai, toh aap un sabhi tasks ko ek batch mein submit kar sakte hain aur system unhe sequentially process karega.

1. **Resource Manager:** Resource Manager ka main kaam hai system ke resources (jaise CPU, memory, storage) ko efficiently manage karna. Yeh monitor karta hai ki kis job ko kis resource ki zarurat hai aur job ko execute karne ke liye suitable resources allocate karta hai.

Example: Agar system mein 10 CPUs available hain, toh Resource Manager decide karega ki kaunsa task kis CPU pe run hoga, aur kis job ko priority milegi.

1. **Scheduler:** Scheduler ka kaam hai jobs ko execute karne ke liye schedule karna. Yeh decide karta hai ki kaunsa job kis waqt execute hoga, based on priority, available resources, aur job dependencies.

Types of Scheduler:

- **Long-term Scheduler:** Yeh decide karta hai ki kis job ko queue mein add kiya jayega.
- **Short-term Scheduler:** Yeh active jobs ko select karta hai jo abhi execute karni hain.
- **Medium-term Scheduler:** Yeh jobs ko swap in/out karta hai memory se disk tak, based on resources.

Session 2 & 3: Submitting and Managing Jobs, Writing the Batch Script

1. **Submitting and Managing Jobs:** Batch systems mein jobs ko submit aur manage karna kaafi important hota hai. Users apni jobs ko ek command ke through submit karte hain, aur system un jobs ko process karta hai.

Job Submission: Jobs ko submit karne ke liye aapko batch scheduler ke through commands deni padti hain, jaise qsub, sbatch, ya bsub (system ke upar depend karta hai).

Example Command (using qsub): "qsub my_batch_script.sh"

Managing Jobs: Jobs ko manage karte waqt aap check kar sakte hain ki job kis state mein hai (running, queued, completed). Aap commands jaise qstat, squeue, bjobs ka use kar sakte hain.

Example Command: "qstat" (Check job status) "squeue" (View running jobs in Slurm)

1. **Writing the Batch Script:** Batch script ek simple text file hoti hai, jisme instructions diye jate hain ki jobs kaise execute kiya jayenge. Yeh script system ko batata hai ki kis task ko execute karna hai, kis resources ki zarurat hai, aur kis environment mein run karna hai.

Basic Batch Script Structure:

- **Shebang Line:** Yeh line batati hai ki script kis interpreter ke through run hogi (usually bash).

#!/bin/bash

Job Details: Job ke liye resource requirements ko define karte hain, jaise memory, CPU cores, etc.

#SBATCH --job-name=my_job

#SBATCH --ntasks=1

#SBATCH --time=01:00:00

Job Commands: Yeh section actual commands ko define karta hai jo job ko run karte hain.

echo "Hello, World!"

Complete Batch Script Example

#!/bin/bash

#SBATCH --job-name=example_job

#SBATCH --output=result.txt

#SBATCH --ntasks=1

#SBATCH --time=00:30:00

Job commands

echo "Starting job"

sleep 10

echo "Job completed!"

Explanation:

- `#!/bin/bash:` Yeh shebang line hai jo system ko batata hai ki script bash interpreter ke through run hogi.
- `#SBATCH --job-name=example_job:` Yeh job ka name set karta hai.
- `#SBATCH --ntasks=1:` Yeh batata hai ki ek task (CPU) ki zarurat hai.
- `#SBATCH --time=00:30:00:` Yeh job ko 30 minutes tak run karne ka time limit set karta hai.
- `echo` aur `sleep` commands simple job commands hain jo script ko execute karte hain.

Job Monitoring and Managing Job States:

Job States:

- **Queued:** Job queue mein hai aur abhi execute nahi hui.
- **Running:** Job currently run ho rahi hai.
- **Completed:** Job execute ho gayi hai aur successfully finish ho gayi.
- **Failed:** Job execute nahi ho paayi, ya koi error aayi.

Commands to Manage Jobs:

- **qsub:** Job submit karne ke liye (PBS systems).
- **sbatch:** Job submit karne ke liye (Slurm systems).
- **bsub:** Job submit karne ke liye (LSF systems).
- **qstat, squeue, bjobs:** Job ka status check karne ke liye.

Conclusion: Batch Systems mein jobs ko efficiently manage karne ke liye Resource Manager aur Scheduler ka role hota hai. Ye ensure karte hain ki system resources efficiently use ho aur jobs ko proper sequence mein execute kiya jaye. Job Submission aur Management ko handle karne ke liye batch scripts likhna padta hai, jisme job ki requirements aur commands define karni padti hain. Aap batch script likhne ke baad usse submit kar sakte hain, aur system automatically jobs ko manage karega based on resource availability aur scheduling.

Session 4 & 5: Slurm Installation and Configuration, Submitting and Managing Jobs, Writing the Batch Script

1. **Slurm Installation and Configuration:**

- **Slurm** ek open-source job scheduler hai jo large-scale parallel systems mein use hota hai.

- Slurm ko install karne ke liye aapko pehle system requirements ko check karna padta hai (Linux distribution aur dependencies).
- Installation process ke liye aap Slurm ke official documentation ko follow kar sakte hain.
- Configuration ke liye `/etc/slurm/slurm.conf` file ko modify karna padta hai, jisme aapko compute nodes aur scheduler settings define karni hoti hain.

1. **Submitting and Managing Jobs:**

- Slurm jobs ko submit karne ke liye `sbatch` command ka use karte hain.
- Aap job ko submit karte waqt resources specify kar sakte hain, jaise CPU cores, memory, aur job ka execution time.

1. Example:

sbatch my_batch_script.sh

Writing the Batch Script:

- Batch script mein aapko resources aur job details specify karni hoti hain.
- Job ke liye resources jaise CPU, memory, aur time limit ko set karte hain.
- Script mein job ko execute karne ke liye commands specify ki jati hain.

Example:

#!/bin/bash

#SBATCH --job-name=my_job

#SBATCH --output=job_output.txt

#SBATCH --ntasks=1

#SBATCH --time=01:00:00

echo "Job Started"

sleep 10

echo "Job Completed"

Session 6 & 7: Managing Nodes, Setting Server Scheduling Policies

1. **Managing Nodes:**

- **Slurm** mein nodes ko manage karna zaruri hota hai. Node ek compute unit hai jahan jobs execute hoti hain.
- Nodes ko manage karne ke liye `scontrol` ya `sinfo` commands ka use kiya jata hai.
- Aap nodes ki status check kar sakte hain, jaise ki nodes available hain ya unavailable hain.

1. Example:

sinfo (Check node status)

scontrol show node=node_name (Detailed node information)

Setting Server Scheduling Policies:

- Server scheduling policies ko define karte waqt aap resource allocation, job priorities, aur fairness ko dhyan mein rakhte hain.
- **Slurm** mein scheduling policies ko `/etc/slurm/slurm.conf` file mein configure kiya jata hai.
- Aap jobs ko different policies ke basis par execute karne ke liye priority, fairshare, ya other parameters set kar sakte hain.

Example of setting scheduling policies in slurm.conf:

PriorityType=priority/multifactor

PriorityWeightAge=10000

PriorityWeightFairshare=10000

PriorityWeightJobSize=10000

Summary:

- **Slurm** ka use job scheduling aur resource management ke liye hota hai, jo large-scale computing environments mein helpful hai.
- Job submit karte waqt batch scripts likhna padta hai, jisme resources aur job execution commands specify karte hain.
- **Nodes management** aur **scheduling policies** set karna bhi zaruri hai taaki resources ka efficient use ho aur jobs effectively execute ho.

Session 8: Scheduler Algorithm

1. **Scheduler Algorithms:**

- Scheduler algorithms ka role hota hai ki wo decide kare ki kaunsa job, kaunse resource pe, aur kab execute hoga.

- Yeh algorithms resources ko efficiently allocate karte hain aur job execution ko optimize karte hain.

1. **Common Scheduler Algorithms:**

- **First Come First Serve (FCFS):** Isme jo job pehle submit hota hai, wo pehle execute hota hai. Simple hai par yeh efficient nahi hota in complex systems mein.
- **Shortest Job First (SJF):** Isme sabse chhoti job ko pehle execute kiya jata hai. Isse overall job completion time reduce hota hai.
- **Priority Scheduling:** Har job ko ek priority value assign ki jati hai, aur highest priority wali job ko pehle execute kiya jata hai.
- **Round Robin (RR):** Sab jobs ko equally time slice diya jata hai, aur fir jobs ko cyclic order mein execute kiya jata hai.
- **Multilevel Queue Scheduling:** Different job types ke liye alag queues hoti hain, aur har queue ki scheduling priority alag hoti hai.

1. **Slurm Scheduler Algorithm:**

- Slurm bhi different scheduling policies ko implement karta hai, jaise fairshare, priority-based, aur backfilling.
- **Fairshare** scheduling mein jobs ko resource allocation ka fairness ensure hota hai, taaki sabhi users ko resources mil sake.
- **Backfilling** mein short jobs ko long jobs ke wait time ke andar execute karna hota hai, taaki idle time ka waste na ho.

?Session 9 & 10: SLURM Accounting

1. **SLURM Accounting:**

- SLURM Accounting ka purpose hota hai system ke resource usage ka tracking aur monitoring. Yeh aapko job execution statistics aur resource consumption ka detailed analysis provide karta hai.
- Accounting ka use job performance ko analyze karne, billing systems ko support karne, aur resource allocation ka optimization karne ke liye hota hai.

1. **SLURM Accounting Features:**

- **Job Completion Records:** SLURM job completion ke baad job ke resources aur execution times ka record rakhta hai.
- **Resource Usage Metrics:** CPU time, memory usage, aur job run time jaise important metrics ko track karta hai.
- **Usage Reports:** SLURM accounting tools jaise `sacct` aur `sreport` aapko usage reports generate karne mein madad karte hain.

1. **SLURM Accounting Commands:**

- **sacct:** SLURM accounting ka primary command hai jo job ki status aur resource usage ko report karta hai.

1. Example:

sacct -j <job_id> # Job ke resource usage ko report karta hai

- **sreport:** Yeh SLURM accounting data ko summarize karta hai aur reports generate karta hai.

Example:

sreport cluster utilization # Cluster ke resource utilization ka report

1. **SLURM Accounting Database (slurmdbd):**

- SLURM Accounting ka data store karne ke liye ek separate database (slurmdbd) use hota hai.
- Is database mein job-related information store hoti hai, jo analysis aur billing purposes ke liye use hoti hai.
- `slurmdbd` ko configure karne ke liye SLURM configuration files ko modify karna padta hai.

1. **Usage:**

- SLURM Accounting ka use performance monitoring, billing, aur resource usage auditing ke liye hota hai.
- Iska integration accounting systems ke saath hota hai, jisse resource usage ka charge calculate kiya ja sakta hai.

Summary:

- **Scheduler Algorithms** ka main goal hota hai jobs ko resources efficiently allocate karna aur execution ko optimize karna.
- **SLURM Accounting** system resource usage ko track karta hai, jisme job ke performance aur resource consumption ki details hoti hain.
- SLURM ke accounting commands jaise `sacct` aur `sreport` ka use aapke jobs ke resource usage ko monitor aur analyze karne mein hota hai.

Assignments – Lab:

1. **Slurm Installation and Configuration:**

- Slurm (Simple Linux Utility for Resource Management) ko apne system pe install karna.
- Installation ke liye package manager (e.g., yum, apt, etc.) ka use karte hue Slurm ko install karein.
- Installation ke baad Slurm ko configure karne ke liye `/etc/slurm/slurm.conf` file ko modify karna padta hai, jisme aapko system resources, nodes, aur scheduling policies define karni hoti hain.

1. **Steps for Slurm Installation:**

- Use your package manager to install Slurm.

sudo apt-get install slurm-wlm # Ubuntu/Debian

sudo yum install slurm # CentOS/RHEL

Configuration file `slurm.conf` ko set karna hota hai jo Slurm ke working aur nodes ko define karta hai.

Slurm service ko start karne ke liye:

sudo systemctl start slurmctld

sudo systemctl enable slurmctld

Submitting Jobs:

- Slurm mein jobs ko submit karne ke liye `sbatch` command ka use hota hai. Batch script ko define karne ke baad, us script ko job scheduler ko submit karne ke liye `sbatch` command ka use karte hain.

Example of Job Submission:

- Job script (`my_job_script.sh`):

#!/bin/bash

#SBATCH --job-name=my_job

#SBATCH --ntasks=1

#SBATCH --time=01:00:00

echo "This is a test job."

Job ko submit karna:

sbatch my_job_script.sh

Managing Jobs:

- Job ko submit karne ke baad, aapko job ka status manage aur track karna padta hai. Slurm mein job ko manage karne ke liye kuch common commands use kiye jate hain.

Job status check karne ke commands:

- `squeue` - Ye command running jobs ko dikhata hai.

`sacct` - Completed jobs aur unki details ko check karne ke liye.

`scancel` - Job ko cancel karne ke liye.

Scheduling Policies Configuration:

- Slurm mein scheduling policies ko configure karne ke liye `slurm.conf` file mein changes karne padte hain.
- Aapko job priority, scheduling algorithm (jaise `fairshare`, `backfilling`), aur resource allocation rules define karne hote hain.

Example of configuring scheduling policies in `slurm.conf`:

- Set priority:

PriorityWeightJobSize=1000

PriorityWeightFairshare=100

Backfilling configuration (ensure that smaller jobs can run while larger jobs wait):

SchedulerType=sched/backfill

Summary:

- **Slurm Installation and Configuration:** Slurm ko install aur configure karte waqt aapko resources aur nodes define karni hoti hain.
- **Submitting Jobs:** Jobs ko `sbatch` command ke through submit kiya jata hai, jisme job script ko pass karte hain.

- **Managing Jobs:** Jobs ka status track karne ke liye commands jaise `squeue`, `sacct`, aur `scancel` ka use kiya jata hai.
- **Scheduling Policies Configuration:** Slurm mein scheduling policies ko modify karke job execution ko optimize karte hain, jaise priority aur backfilling.

Here's a structured document format based on the content you provided:

Batch Systems and Slurm Scheduler: Overview and Practical Guide

Session 1: Batch Systems, Resource Manager, and Scheduler

1.1 Batch Systems

- A computing system where tasks or jobs are queued and processed sequentially without real-time interaction.
- **Example**: Processing multiple files by submitting tasks in a batch.

1.2 Resource Manager

- Efficiently manages system resources (CPU, memory, storage).
- Allocates resources to tasks based on requirements.
- **Example**: Assigns tasks to available CPUs, balancing priorities.

1.3 Scheduler

- Decides when and how jobs are executed, ensuring efficient execution.

Types of Schedulers:

1. **Long-term Scheduler**: Adds jobs to the queue.
2. **Short-term Scheduler**: Selects active jobs for execution.
3. **Medium-term Scheduler**: Swaps jobs in/out of memory to optimize resources.

Session 2 & 3: Job Submission, Management, and Batch Script Writing

2.1 Submitting Jobs

- **Commands**:
- `qsub`: Submit job in PBS systems.
- `sbatch`: Submit job in Slurm systems.
- **Example**: `sbatch my_batch_script.sh`.

2.2 Managing Jobs

- Monitor job states: Queued, Running, Completed, Failed.
- **Commands**:
- `qstat`, `squeue`: View job status.
- `scancel`: Cancel jobs.

2.3 Writing Batch Scripts

Basic Structure:

```
#!/bin/bash
#SBATCH --job-name=my_job
#SBATCH --output=output.txt
#SBATCH --ntasks=1
#SBATCH --time=01:00:00

echo "Job Started"
sleep 10
echo "Job Completed"
```

Session 4 & 5: Slurm Installation, Configuration, and Job Submission

4.1 Slurm Installation

- Install via package manager:

```
sudo apt-get install slurm-wlm  # For Ubuntu/Debian
sudo yum install slurm          # For CentOS/RHEL
```

- Start the Slurm service:

```
sudo systemctl start slurmctld
sudo systemctl enable slurmctld
```

4.2 Slurm Configuration

- Modify `/etc/slurm/slurm.conf` to define compute nodes and scheduling policies.

Session 6 & 7: Managing Nodes and Scheduling Policies

6.1 Managing Nodes

- **Commands**:

- `sinfo`: Check node status.
- `scontrol show node=<node_name>`: Detailed node info.

6.2 Setting Scheduling Policies

- Configure in `/etc/slurm/slurm.conf`:

```
PriorityType=priority/multifactor
PriorityWeightAge=10000
PriorityWeightFairshare=10000
PriorityWeightJobSize=10000
```

Session 8: Scheduler Algorithms

8.1 Common Algorithms

1. **First Come First Serve (FCFS)**
2. **Shortest Job First (SJF)**
3. **Priority Scheduling**
4. **Round Robin (RR)**
5. **Multilevel Queue Scheduling**

8.2 Slurm Scheduling

- Implements **fairshare** and **backfilling** for optimized resource usage.

Session 9 & 10: SLURM Accounting

9.1 SLURM Accounting Features

- Tracks job resource usage (CPU, memory, etc.).
- Generates usage reports via commands like `sacct` and `sreport`.

9.2 Commands

- `sacct -j <job_id>`: View job resource usage.
- `sreport`: Generate summary reports.

9.3 SLURM Accounting Database

- Stores job execution details for analysis and billing.

Assignments: Practical Implementation

A. Slurm Installation and Configuration

1. Install Slurm using your package manager.

2. Modify `/etc/slurm/slurm.conf` to define nodes and resources.
3. Start Slurm service:

```
sudo systemctl start slurmctld
sudo systemctl enable slurmctld
```

B. Writing and Submitting Jobs

Job Script:

```
#!/bin/bash
#SBATCH --job-name=my_job
#SBATCH --ntasks=1
#SBATCH --time=01:00:00

echo "This is a test job."
```

Submit the job:

```
sbatch my_job_script.sh
```

C. Managing Jobs

- Monitor job status with `squeue` and `sacct`.
- Cancel jobs using `scancel`.

D. Scheduling Policies Configuration

- Configure priorities and backfilling in `slurm.conf`.

Summary

- **Batch Systems**: Automate task execution using Resource Manager and Scheduler.
- **Slurm**: A robust system for job scheduling, node management, and resource tracking.
- **Practical Focus**: Writing batch scripts, submitting jobs, and managing job states.

End of Document