LCI workshop
on high performance clustered computing

LCI Advanced Workshop 2025: **Advanced Slurm Reservations**

Josh Burks
HPC Systems Analyst
Arizona State University Research Computing
josh.burks@asu.edu

ASU Research Computing

Learning Goals

- Understand the purpose and mechanics of Slurm reservations
- Differentiate between reservation types (maintenance, magnetic, floating, flex)
- Create and manage time-limited private node access using reservations
- Learn to configure and troubleshoot advanced reservation scenarios
- Understand interactions between reservations, QOS, and job submission
- Apply best practices to maximize system utilization and user satisfaction

What is a Slurm Reservation

- A reservation guarantees resources for specific users or jobs during a time window
- Commonly used for:
  - Maintenance windows
  - High-priority workloads
  - System testing or training sessions
  - Private or shared cluster agreements
- Reservations can be exclusive or allow shared usage

Types of Slurm Reservations

- Maintenance Reservations: Block usage for planned downtime
- Magnetic Reservations: Pull eligible jobs into reservation
- Floating Reservations: Activate when conditions are met (e.g., drain complete)
- Flex Reservations: Auto-extend based on job usage
- Private Node Access: Temporarily assign nodes to a specific user/group
- Recurring Reservations: Repeat reservations on a schedule

Maintenance Reservations

- Use case: Blocking user access for hardware/software maintenance
- Typically exclusive (Flags=MAINT)
- Jobs are prevented from running on reserved nodes
- Existing jobs may need to be drained first
- Can be created with:
- scontrol create reservation starttime=... duration=... flags=maint nodes=...

Magnetic Flag Example

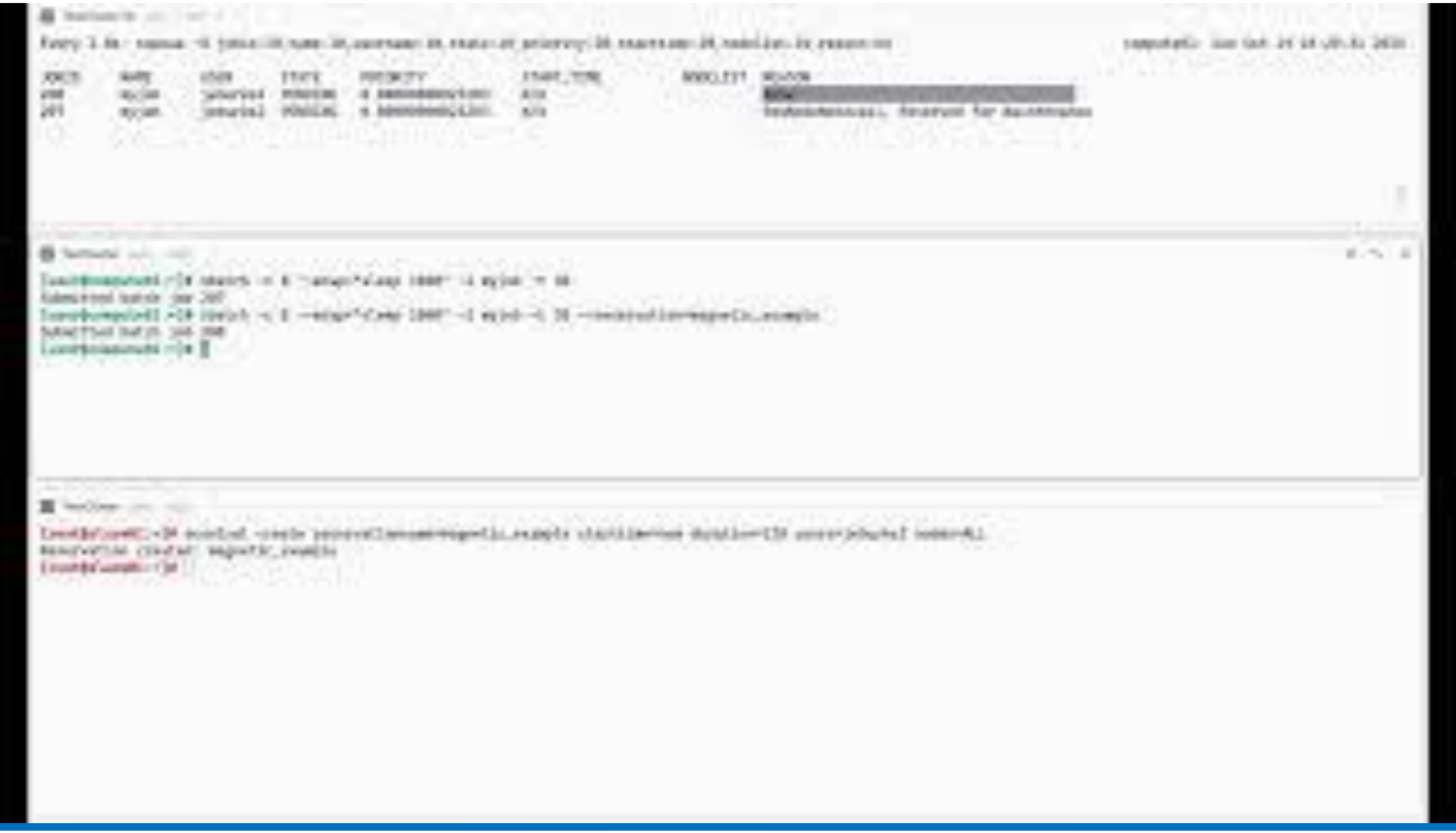
- The magnetic flag “attracts” jobs into the reservation that meet the criteria of the reservation
- ie, if user jeburks2 has a reserved node and submits a job without the –reservation SBATCH job, Slurm will attract the job into the reservation
- Mostly a ease of use for users
- Cannot "attract" heterogeneous jobs

Magnetic Flag Example

- Reserve node compute01 for user jeburks2, so any job that jeburks2 submits may run in the reservation on compute01

```
scontrol create reservationname=magnetic_example \
   starttime=now duration=120 users=jeburks2 \
   nodes=compute01 flags=magnetic
```

Magnetic Flag Example Video



FLEX Flag

- The FLEX flag makes it so jobs can start before the reservation begins and run after the reservation ends
- Jobs are not limited to only the nodes in the reservation
- Useful for guaranteeing a minimum set of resources or a time slot for a large job / group of jobs without limits
- Can provide a way to temporarily alleviate large job starvation by creating an anchor point for a large job

FLEX Flag Example

- Create a reserved resource on compute node 1 for 2 hours, but if idle resources permit is, user jeburks2 can run on more than 1 compute node

```
scontrol create reservationname=flex_example \
   starttime=now duration=120 users=jeburks2 \
   nodes=compute01 flags=flex
```

FLEX Flag Example Video

Recurring Reservations

LCI workshop
on high performance clustered computing

- Reservations can set to automatically repeat with special flags
- Each of these flags are mutually exclusive
- Useful for recurring training events

```
flags=DAILY

flags=HOURLY

flags=WEEKLY

flags=WEEKEND

flags=WEEKDAY
```

Floating Reservations

- A floating reservation in Slurm is one whose start time moves forward as real time passes.
- Example: starttime=now+60minutes flags=time_float
  - always stays 60 min ahead of current time.
- Purpose: prevent long jobs from starting on selected nodes while still allowing short ones.
- Commonly used to prep nodes for maintenance, upgrades, or vendor work without blocking short backfill jobs.

Floating Reservations Use Cases

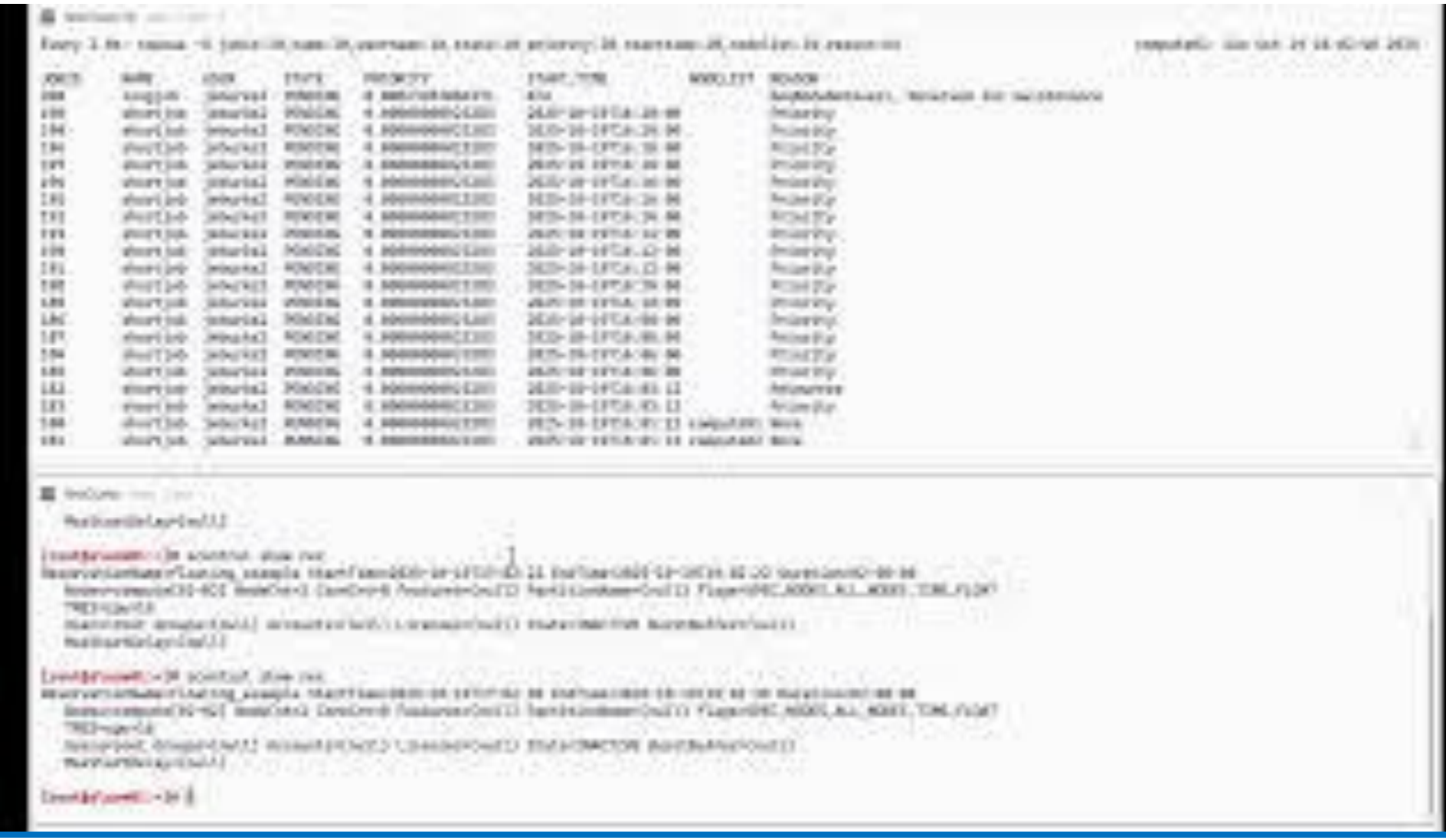
- Maintenance prep: stop 10-hour jobs from starting when hardware work may start “soon.”
- Rolling upgrades: mark nodes slated for kernel/firmware updates but don’t drain yet.
- Vendor or facility windows: floating reservations keep pushing until vendor arrival.
- Urgent debug/testing: reserve nodes for short-notice troubleshooting without wasting resources.
- Benefits:
  - Keeps scheduler efficient.
  - Avoids manual drain too early.
  - Ensures nodes are free when needed.

Floating Reservations Examples

- Stop jobs longer than 1 hour from starting on compute node 1 in preparation for a 2 hour downtime

```
scontrol create reservationname=floating_example \
   starttime=now+60minutes duration=120 \
   users=root nodes=ALL flags=time_float
```

Floating Reservations Example Video



Replace Flag

LCI workshop
on high performance clustered computing

- "Replace a node in the reservation when the node goes down"
- Use case: Give PI or group exclusive access to nodes for a period
- Use users= or accounts=
- Set NodeCnt to set count of nodes
- Combine with QOS to enforce job size, runtime, or priority policies
- Add recurring reservations for long-term agreements

```
scontrol create reservationname=pi_joe nodecnt=1
user=root starttime=now endtime=2999-12-31
partition=general flags=replace_down
```

Reserving Partial Nodes

- Useful for high-density systems (e.g., many-core nodes)
- Use CoreCnt= or NodeCnt= to specify reservation scope
- Allows better resource utilization without blocking full node

```
scontrol create reservationname=partial corecnt=1
user=root starttime=now endtime=now+60
partition=general
```

Interactions with QOS and Job Submission

Reservations override normal scheduling, but QOS still applies

Reservation does not increase job priority unless QOS allows

Users must still submit jobs with matching constraints:

--reservation=<name>

Correct partition, account, etc.

Q & A