Slurm Version 14.11

Jacob Jenson
jacob@schedmd.com

SchedMD LLC
http://www.schedmd.com

V14.11 - Highlights

- Core specialization
- Improved job array performance and scalability
- Support for heterogeneous generic resources
- CPU governor options
- Automatic job requeue policy based on exit value

V14.11 - Highlights

- Job "reboot" option for Linux clusters
- Database performance enhancements
- SelectTypeParameters option CR_PACK_NODES
- Support for non-consumable generic resources
- API usage statistics by user, type, count and time consumed

V14.11 – Core Specialization

- Support for reserving cores on a compute node for system services
  - Uses Linux cgroup
  - Isolate system overhead
- Specialized cores can be reserved on each node by default in slurm.conf
- Application can modify default specialized core count
  - --core-spec=#
  - Change from default requires whole node allocation

V14.11–Job Arrays

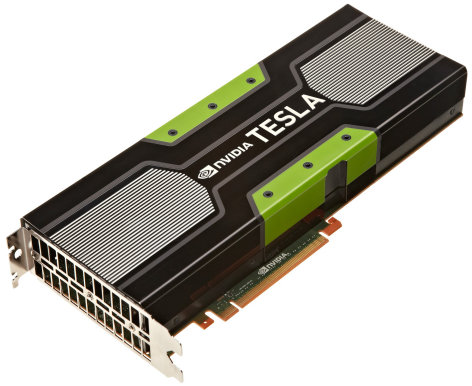
- New job array data structure
- Individual job records created as needed
  - Typically when a task is allocated resources rather than at submit time
- Many APIs modified to operate on job arrays instead of individual job records
- Removed 64,000 job array size limit
  - Practical limit 1,000,000 tasks

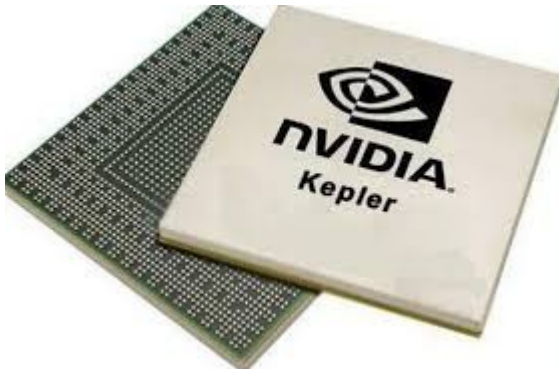
V14.11 – Job Array

	v14.03 60k tasks	v14.11 60k tasks	v14.11 1m tasks
Submit (sbatch)	2.6 sec	0.02 sec	0.02 sec
Status (squeue)	0.2 sec	0.02 sec	0.03 sec
Cancel (scancel)	0.2 sec	0.01 sec	0.01 sec

V14.11 – Heterogeneous Generic Resources

- Support different Generic Resource types
- Example
  - User specification of desired GPU types
    - --gres=gpu:kepler:1
    - --gres=gpu:kepler:1,gpu:tesla:1
    - --gres=gpu:2
      - Any GPU type is acceptable





V14.11 – Power Management

- Users can now set CPU governor or frequency
- Governor Options
  - OnDemand, Performance, PowerSave, Conservative and UserSpace
- Usage
  - --cpu-freq=OnDemand
  - --cpu-freq=high
- CPU governor and frequency are preserved with job preemption, including gang scheduling

V14.11 – API Statistics from sdiag

```
$ sdiag
….

Remote Procedure Call statistics by message type
REQUEST_JOB_INFO_SINGLE                 ( 2021) count:36    ave_time:228    total_time:8225
REQUEST_NODE_INFO                       ( 2007) count:36    ave_time:201    total_time:7246
REQUEST_BUILD_INFO                      ( 2001) count:24    ave_time:232    total_time:5570
REQUEST_PING                            ( 1008) count:24    ave_time:163    total_time:3912
REQUEST_COMPLETE_BATCH_SCRIPT           ( 5018) count:16    ave_time:439    total_time:7037
REQUEST_SUBMIT_BATCH_JOB                ( 4003) count:9     ave_time:432    total_time:3888
…..

Remote Procedure Call statistics by user
      jacob          (   1234)    count:190    ave_time:1838    total_time:349302
     joseph          (   1235)    count:26     ave_time:351     total_time:9147
….
```

The goal is to obtain data from slurmctld behavior helping to adjust and optimize configuration parameters or queues policies.

V14.11 – SelectTypeParameters

- CR_Pack_Nodes
  - Rather than evenly distributing a job's tasks across allocated nodes, pack tasks as tightly as possible on the nodes.

V14.11 – SelectTypeParameters

- CR_Pack_Nodes Example
  - Two node allocation; 8 cores/node; 10 tasks

Default Behavior

Tux001		Tux002	
Core 1	Task 0	Core 1	Task 5
Core 2	Task 1	Core 2	Task 6
Core 3	Task 2	Core 3	Task 7
Core 4	Task 3	Core 4	Task 8
Core 5	Task 4	Core 5	Task 9
Core 6		Core 6	
Core 7		Core 7	
Core 8		Core 8	

CR_PACK_NODES

Tux001		Tux002	
Core 1	Task 0	Core 1	Task 8
Core 2	Task 1	Core 2	Task 9
Core 3	Task 2	Core 3	
Core 4	Task 3	Core 4	
Core 5	Task 4	Core 5	
Core 6	Task 5	Core 6	
Core 7	Task 6	Core 7	
Core 8	Task 7	Core 8	

V14.11 – Reboot Option

- Job reboot option for Linux clusters
- Invokes the configured RebootProgram to reboot nodes allocated to a job before it begins execution
  - Clean environment

V14.11 – Database Speed

- Massive database performance enhancements
- Primarily benefit systems running many short lived jobs

	1001 node registrations	1001 job starts
14.03	7.05 sec	5.14 sec
14.11	3.20 sec	0.10 sec