Slurm Roadmap

Morris Jette, Danny Auble (SchedMD)
Yiannis Georgiou (Bull)

Exascale Focus

- Heterogeneous Environment
- Scalability
- Reliability
- Energy Efficiency
- New models (Cloud/Virtualization/Hadoop)

Following Releases

- Agile development with two major releases per year plus periodic bug fixes
- Switching to Ubuntu version number scheme
  - <year>.<month>
  - 13.12 == December 2013
  - 14.06 == June 2014
  - 14.12 == December 2014

Upcoming Release 13.12

Optimize MPI Network Use

- New switch/generic plugin
- Captures network interface configuration on each node (name and IP address)
- Provides job with network interface details about allocated nodes using PMI2
- Step/Job Epilog Hierarchical Communications

Scheduling

- New partition configuration parameters: AllowAccounts, AllowQOS, DenyAccounts, DenyQOS
- Added load-based scheduling
  - Allocate resources on least loaded node first
  - Can improve serial job performance, but maximizes fragmentation

Failure Management

- Configurable hot-spare nodes
- Running jobs can replace or remove failed or failing nodes
- Jobs can extend time limit based upon failures
- Jobs can drain nodes they perceive to be failing
- Configurable access control lists and limits

Energy Use Optimizations

- Integration of Layout framework, which will be the basis for optimizing resource management, job placement and scheduling based on resources characteristics
- Power cap enforcement and considerations in job placement
- Reservations of Power cap
- Job energy consumption as a new factor in fair-share scheduling

Hadoop Integration

- Work being performed by Intel
- Eliminates need for dedicated Hadoop cluster
- Better scalability
  - Launch: Hadoop/YARN (~N), Slurm (~log N)
  - Wireup: Hadoop/YARN (~N^2), Slurm (~log N)
- No modifications to Hadoop
  - Completely transparent to existing applications

Slurm-Hadoop Architecture

- Java Daemon
  - Intercepts calls from client and translates for Slurm
- Slurmctld plugin
  - Accepts and processes RM requests
  - Web interface for job/system status
- Slurmd plugin
  - Spawn local processes, support shuffle and other ops

Slurm-Hadoop Status

- Under development
- Tentative schedule
  - Demo at SC13
  - Release 4Q 2013 or 1Q 2014
  - Slurm plugins in version 13.12
- Translator daemon will be available from Intel

Other Features

- Licenses Management in Accounting and support of FlexLM (Flexnet Publisher)
- Stable version of Jobacct_gather/cgroup plugin
- Support of PAM with cgroups
- Improved sview scalability
- Added job_container plugin infrastructure
- Improved integration with Cray systems

Future Release 14.06

Heterogeneous resources

- Distributed architecture to support the management of resources with MIC
  - Lightweight slurmd upon MIC
  - Router capabilities on slurmd based on static and dynamic trees
- Support of I/O as new resource along with data locality
- Extension on the Job requirements description to support heterogeneous resources

Improved GPU and MIC Support

- Support for heterogeneous GPUs
  - salloc --gres=gpu:1 …
  - salloc --gres=gpu:tesla:1,gpu ...
- Use MIC to offload work from CPUs or as independent compute node

Scalability and scheduling

- Scalability optimizations for MPI initialization
- Improved scheduling support for job dependencies (e.g. pre-processing, post-processing, co-processing on I/O nodes, etc.) to optimize overall system utilization

Failure Management

- Distribute hot spare resources through system
- Optimize job’s replacement resources with respect to network latency

Energy Use Optimizations

- Work likely to continue for several years
- Finer grain Power Management
- Optimize system throughput with respect to varying power caps
  - Consider DVFS or other models in conjunction with power caps
- Limit rate of change in system power consumption

Release 14.12 and beyond

Directions

- Multi-parameter scheduling based on the layout framework
- Fault-tolerance and jobs dynamic adaptation through communication protocol between Slurm , MPI libraries and the application
- Network Communication Scalability optimizations