SLURM Operation IBM BlueGene/Q

Danny Auble
da@schedmd.com

SchedMD LLC
Contributors and Collaborators

This development work was funded by Lawrence Livermore National Laboratory

With technical assistance from IBM
Outline

• BlueGene/Q hardware and software architecture
• SLURM architecture for BlueGene/Q
• SLURM configuration and use
• Differences from BlueGene/L and P systems
• Status
IBM BlueGene/Q Architecture

- Latest generation of IBM BlueGene series
- Nodes are diskless
- 5-dimension torus interconnect
- Full Linux on front-end nodes
- Lightweight Linux kernel on compute nodes
- Whole nodes must be allocated to jobs
BlueGene/Q Hardware

- BlueGene hardware building block is known as a mid-plane occupying half of a rack
  - On a BlueGene/Q mid-planes are scheduled in a 4-dimensional space
- Each mid-plane typically contains 512 compute nodes (c-nodes)
  - On a BlueGene/Q the c-nodes are arranged in a 4x4x4x4x2 5-dimensional torus
  - Each BlueGene/Q c-node has 16 usable cores
- Livermore's Sequoia machine will have
  - 192 mid-planes (3x4x4x4 torus)
  - 98,304 c-nodes
  - 1,572,864 cores
BlueGene/Q Software

- SLURM daemons do not execute directly on the c-nodes
- SLURM gets system state, allocates resources and performs other operations through use of IBM infrastructure
- This interface is entirely contained within a SLURM plugin (`src/plugins/select/bluegene`)
  - This plugin is used for all IBM BlueGene systems, but the logic in the plugin is different depending on the type of BlueGene
SLURM and BlueGene Functionality

• **SLURM**
  • Prioritizes queue(s) of work and enforces limits
  • Decides when and where to start jobs
  • Terminates job when appropriate
  • Accounts for jobs

• **IBM BlueGene Software**
  • Allocates and releases resources for jobs based off SLURM input
  • Launches tasks
  • Monitors node health
\textit{srun} Command

- \textit{srun} creates a job step (as on other SLURM systems), but rather than launching the user application directly, launches a single instance of \textit{runjob} on one of the BlueGene/Q front-end nodes
  - Options are translated to the extent possible
  - SLURM job step is created for record keeping purposes
SLURM Architecture for BlueGene/Q (Detailed)

- **Slurmctld**
  - (SLURM controller daemon)
  - (primary or backup)
  - Coordinates all activities

- **Slurmd**
  - (SLURM job daemons)
  - (Active on one or more service nodes)
  - Runs batch script

- IBM BlueGene Software

- Compute Nodes
1. User submits script

Slurmctld
(SLURM controller daemon)
(primary or backup)
Coordinates all activities

Slurmd
(SLURM job daemons)
Runs batch script

IBM BlueGene Software

Compute Nodes
1. User submits script
2. Slurmctld changes network switches, boots c-nodes and allocates resources to some user
1. User submits script

2. Slurmctld changes network switches, boots c-nodes and allocates resources to some user

3. Slurmctld sends script to slurmd
1. User submits script
   Slurmctld (SLURM controller daemon) (primary or backup) Coordinates all activities
   Slurmd (SLURM job daemons) Runs batch script

2. Slurmctld changes network switches, boots c-nodes and allocates resources to some user
   IBM BlueGene Software

3. Slurmctld sends script to slurmd
   Compute Nodes

4. Slurmd runs script
   #!/bin/bash srun a.out
1. User submits script
2. Slurmctld changes network switches, boots c-nodes and allocates resources to some user
3. Slurmctld sends script to slurmd
4. Slurmd runs script
5. Srun creates a job step that executes runjob
1. User submits script

2. Slurmctld changes network switches, boots c-nodes and allocates resources to some user

3. Slurmctld sends script to slurmd

4. Slurmd runs script

5. Srun creates a job step that executes runjob

6. runjob launches user tasks
# Sample slurm.conf file for BlueGene system
# Selected portions
#
SelectType=select/bluegene
#
FrontEndName=front[00-03]      # Where slurmd daemons run
NodeName=bgq[0000x2333]       
PartitionName=batch Nodes=bgq[0000x2333] MaxTime=24:00:00
sview of Emulated System
Differences from BlueGene/P

- More dimensions (in place of split cables)
  - Easier to pack jobs, especially in dynamic mode
- Multiple Users can be allowed to run various allocation sizes in a single block
  - More efficient use of smaller machines
  - Can be operated more like a traditional Linux cluster
- An allocation can run multiple job steps per allocation
- Accounting information is available for job steps
  - Native srun command is wrapper for runjob command
Status

- Partial implementation in SLURM version 2.3
- Full implementation in SLURM version 2.4
  - Multiple job allocations within a single block
  - More error handling
  - Better system monitoring
  - Advanced reservations can specify sizes of individual blocks