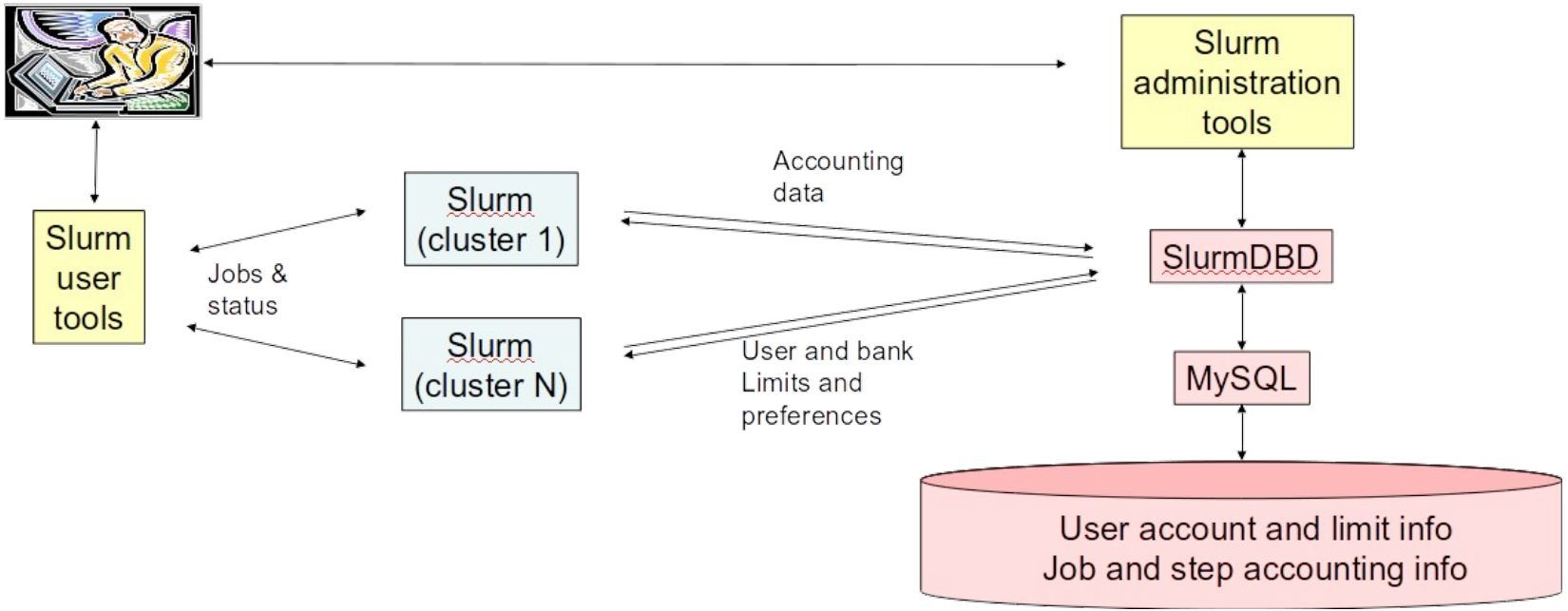
Slurm Workload Manager Overview
SC15

Danny Auble
da@schedmd.com

Slurm Workload Manager Overview

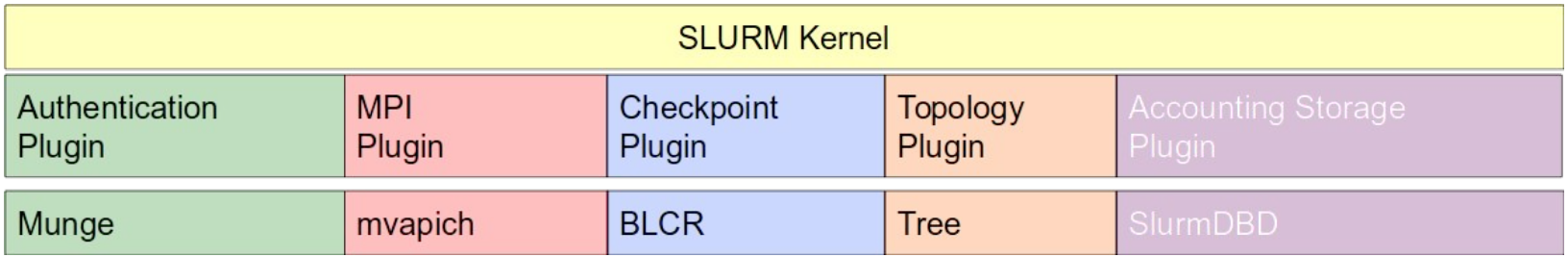
- Originally intended as simple resource manager, but has evolved into sophisticated batch scheduler
- Able to satisfy scheduling requirements for major computer centers with use of optional plugins
- No single point of failure, backup daemons, fault-tolerant job options
- Highly scalable (3.1M core Tianhe-2 at NUDT)
- Highly portable (autoconf, extensive plugins for various environments)
- Open source (GPL v2)
- Operating on many of the world's largest computers
- About 500,000 lines of code today (plus test suite and documentation)

Enterprise Architecture



Architecture

- Kernel with core functions plus about 100 plugins to support various architectures and features
- Easily configured using building-block approach
- Easy to enhance for new architectures or features, typically just by adding new plugins



Scheduling Capabilities

- Fair-share scheduling with hierarchical bank accounts
- Preemptive and gang scheduling (time-slicing parallel jobs)
- Integrated with database for accounting and configuration
- Resource allocations optimized for topology
- Advanced resource reservations
- Manages resources across an enterprise

Multifactor Prioritization Plugin

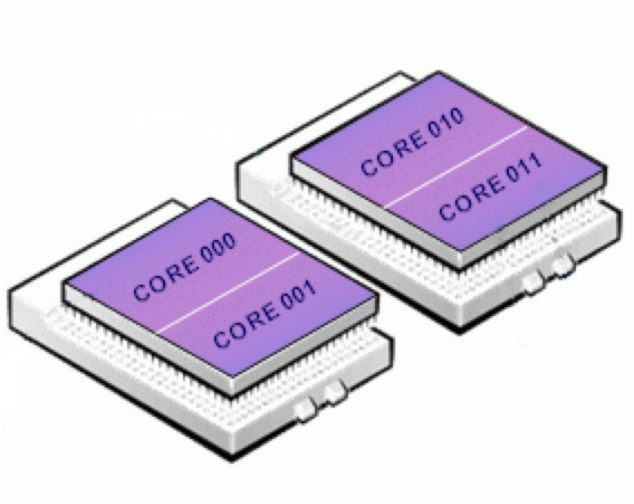
- Jobs can be prioritized using highly configurable parameters
  - Job Age
  - Job Partition
  - Job size
  - Job Quality Of Service (QOS)
  - User and account's fair-share allocation

Scalability

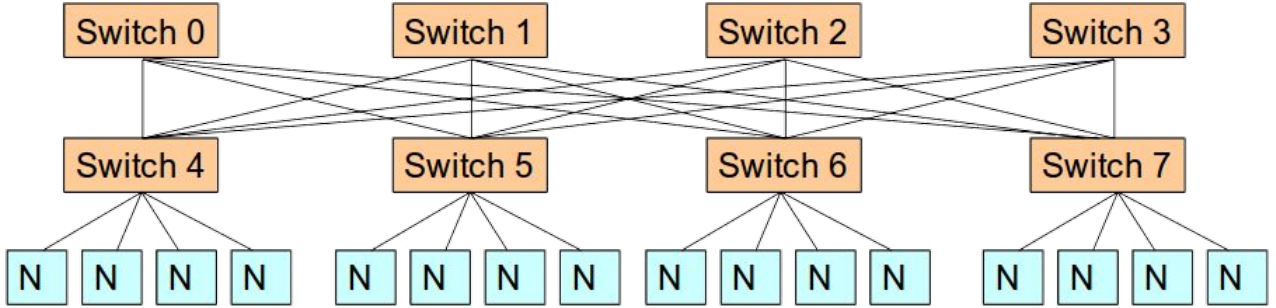
- Everything is multi-threaded
- Separate read and write locks on the various data structures in the daemons
- No single point of failure
- RPCs designed to minimize bottlenecks from control daemon as much as possible

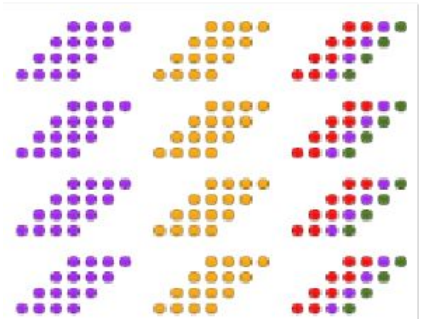
On-node Topology Optimization

- Users have complete control over task layout across the nodes, sockets, cores and threads to optimize application performance



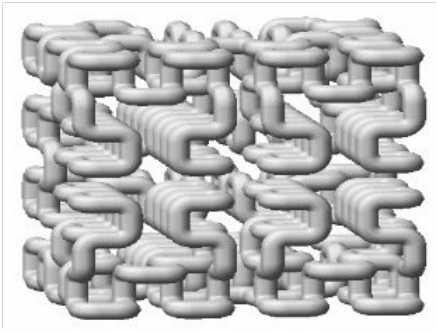
Topology Plugin Optimization





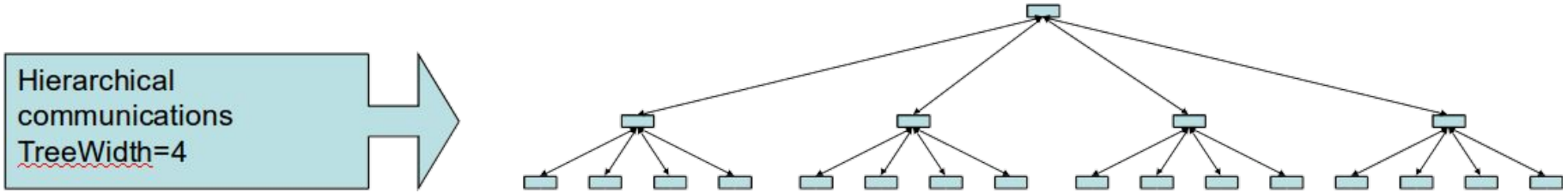
Job layout on BlueGene (sview output)

3-D Hilbert curve for Sun Constellation



Communications

- Hierarchical communications with configurable fanout and fault-tolerance



Communications

- All commands and configuration files are designed to compress host names using a prefix and numeric suffix
- Easy to configure large systems

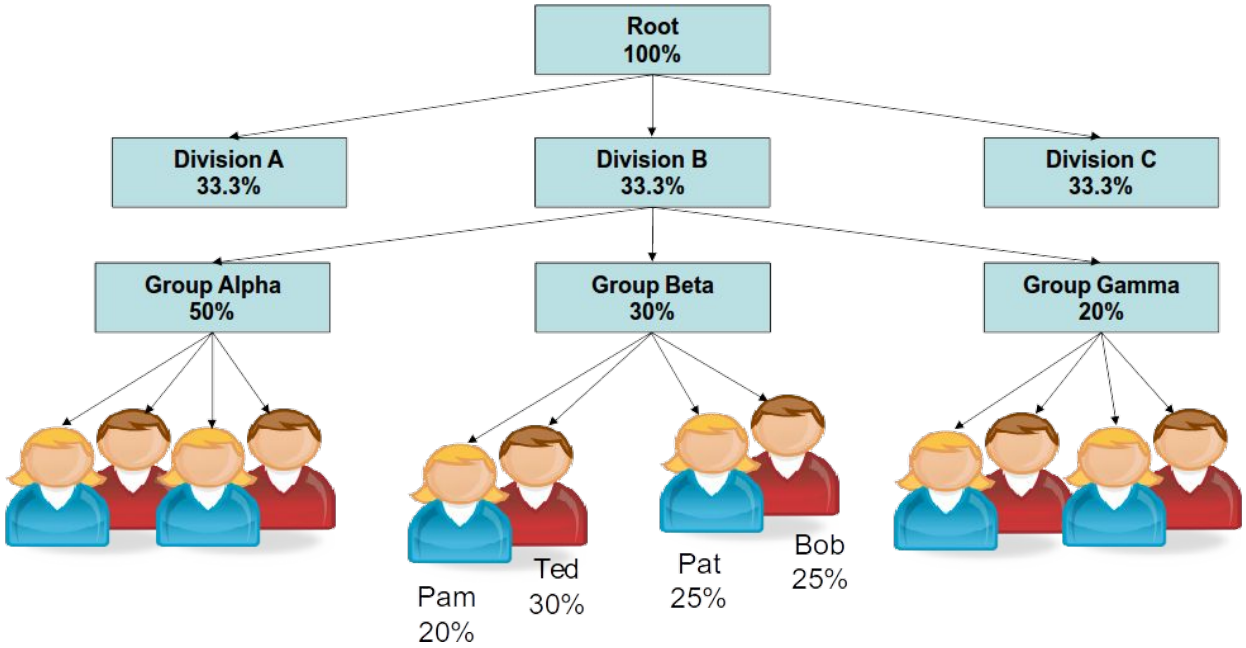
```
# Sample Slurm configuration file (excerpt)
#
NodeName=tux[0-1023] Sockets=4 CoresPerSocket=6
#
PartitionName=debug Nodes=tux[2-17] Default=yes
Maxtime=30
PartitionName=batch Nodes=tux[18-1023] MaxTime=24:00:00
```

Database Use

- Job accounting information written to a database plus
  - Information pushed out to scheduler daemons
  - Fair-share resource allocations
  - Many limits (max job count, max job size, etc)
  - Based upon hierarchical accounts
    - Limits by user AND by accounts

> *"All I can say is wow – this is the most flexible, useful scheduling tool I've ever run across."*
> Adam Todorski, Rensselaer Polytechnic Institute

Hierarchical Account Example



Advanced Features

- Scheduling for generic resources (e.g. GPUs, MICs)
- User control over CPU frequency (performance and energy use)
- Real-time accounting down to the task level
  - Identify specific tasks with high CPU or memory usage
  - Record energy consumption by job
- Job profiling
  - Periodically capture each task's memory, CPU, power, network and I/O

15.08 Features

- Version 15.08.0 released on August 31
  - Massive changes from version 14.11
  - Diff file >250,000 lines
- Trackable Resources (TRES): Tracks utilization of memory, GRES, burst buffer, license, and any other configurable resources in the accounting database
- Per-Partition QOS
- Burst Buffers: a cluster-wide high-performance file system
- Network Topologies Optimizations, New parameters and environment variables...