

# Federated Cluster Support



Brian Christiansen and Morris Jette  
SchedMD LLC

Slurm User Group Meeting 2015

# Background



- Slurm has long had limited support for federated clusters
  - Most commands support a “--cluster (-M)” option to route requests to different clusters
- Submitted jobs are routed to one cluster
- Each cluster operates independently
  - Job IDs on each cluster are independent (two jobs can have same ID)
  - No cross-cluster job dependencies
  - No job migration between clusters
  - No unified view of system state, each cluster largely independent

# New Capabilities



- **Job Migration**

- Pending jobs automatically migrated to less busy clusters

- **Fault Tolerance**

- Participating clusters will take over work of a failed cluster

- **Cross-cluster Job Dependencies**

- **Unified Views**

- **Easy Administration**

- Add/remove clusters to/from the federation with simple configuration change, no extra information required in database

# Related Work



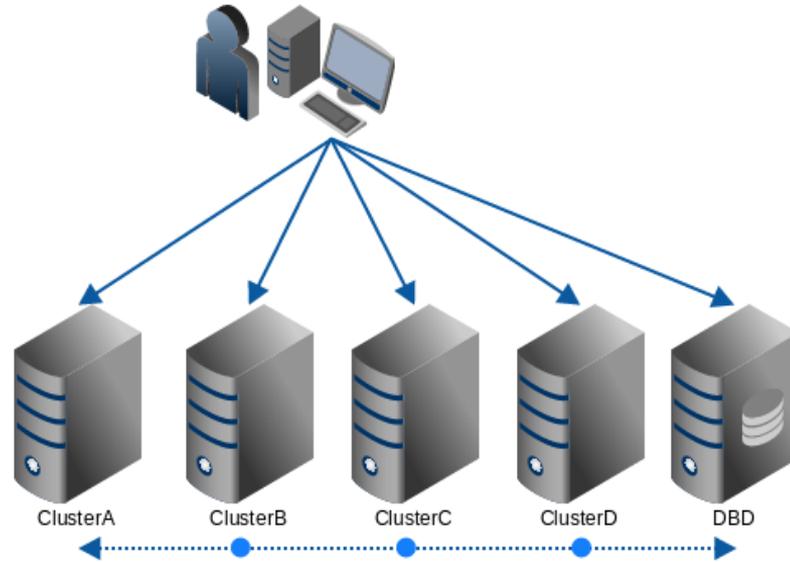
- Some work has been done on addressing these shortcomings in functionality, but was lacking in scalability and was never integrated into the official release
- The major problem was the use of a single daemon to manage the mapping of job ID to cluster
- The slurmdbd maintained a table identifying which job IDs were on each clusters
- Slurmdbd used for job dependency testing
- Placed very heavy load on slurmdbd to locate jobs

# Design Goals



- **Performance:** Little to no reduction in throughput of each cluster, performance scales with cluster count
- **Scalability:** No reduction in scalability of individual clusters, able to support many federated clusters
- **Fault tolerant:** No single point of failure
- **Ease of use:** Unified enterprise-wide view, minimize change in user interface
- **Stability:** No change in behavior for clusters not explicitly placed into a federation

# Eliminating the Bottleneck



# Eliminating the Bottleneck

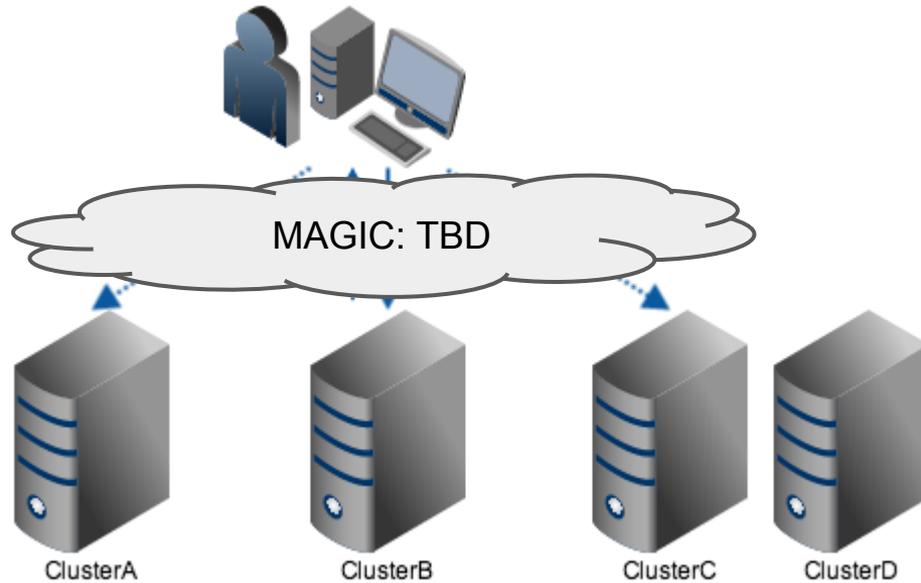


- Need mechanism to identify the cluster associated with a job ID without using slurmdbd lookup
- Make use of 32-bit job ID
  - Embed cluster ID within the job ID
  - Bit 31: Flag for federated cluster job ID
  - Bits 23-30: Cluster ID (0 to 255)
  - Bits 0-22: Job ID (0 to 8,388,607)
- Unique job ID across all clusters
  - Large but unique: 2164339463 (ClusterBit + ClusterID:2 + JobID:78,599)

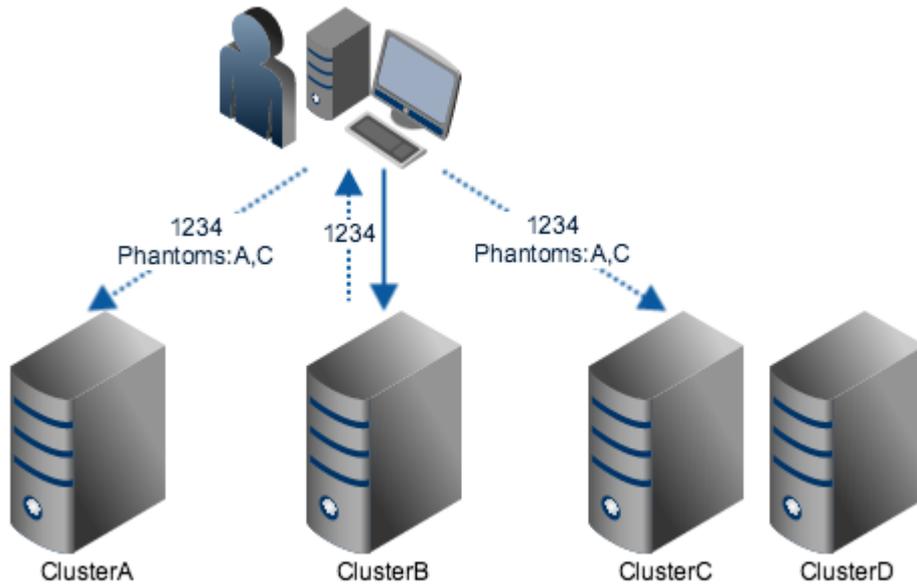
# Job Submission

- sbatch, salloc, srun supported
- Get available clusters (IP address + port) from local slurmctld
  - Local slurmctld keeps a cache from the slurmdbd
  - slurmdbd is the backup for cluster information
- Submit a “master” job to a randomly selected cluster
  - Perform light weight check to verify job can run on cluster
- “phantom” jobs are submitted to a number of clusters
- All jobs contain job ID and the locations of all “phantom” jobs

# Job Submission

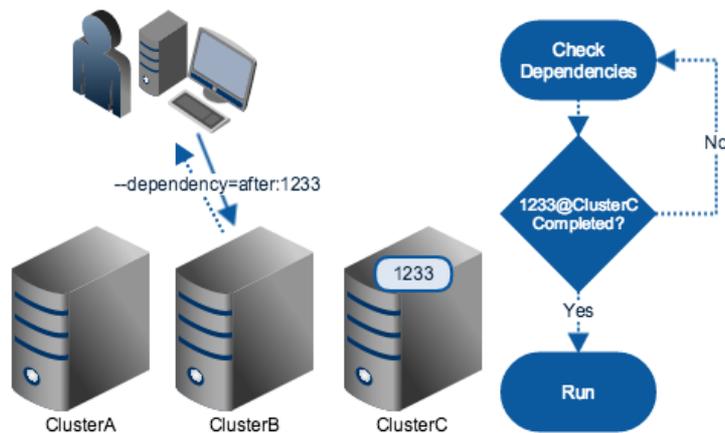


# Job Submission



# Cross-Cluster Job Dependencies

- Dependencies created with standard `--dependency=` syntax
- “Master” controller will check status of remote jobs on other clusters

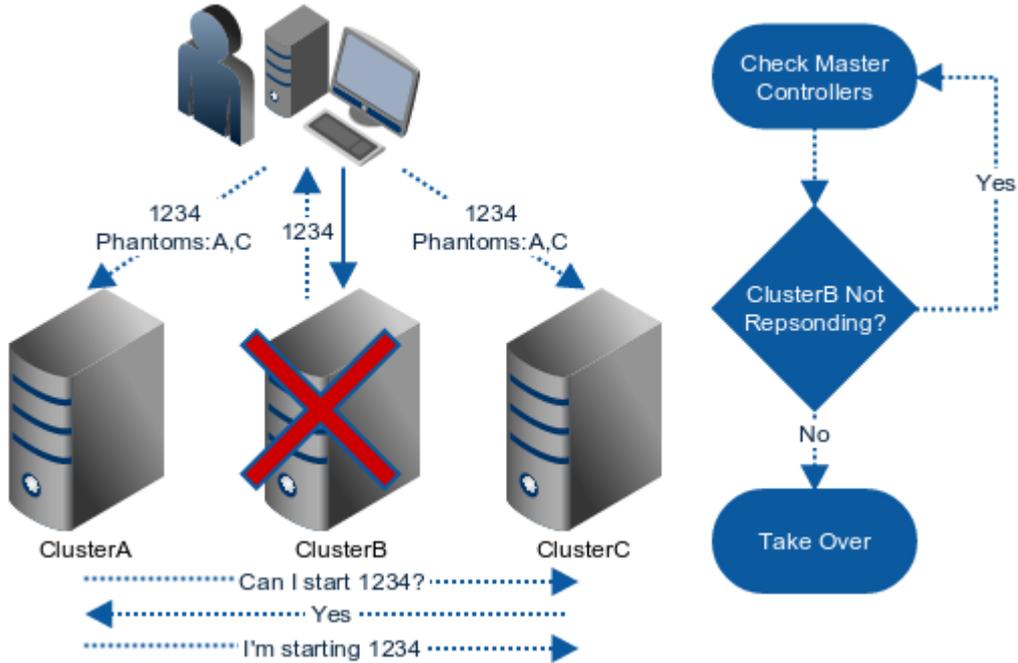


# Fault Tolerance / Job Migration



- Controllers coordinate taking over the job if:
  - “master” controller is down for a period of time
  - Job can be started sooner
- The RPCs for jobs started on another cluster will be re-routed
- Lots of moving parts to consider and to prevent split-brain
  - Don't have two jobs running at the same time

# Fault Tolerance / Job Migration



# Configuration



- Parameter identifying whether cluster is part of a federation
  - Federation=yes|no
- Parameter to control how many “phantom” jobs to spawn
  - PhantomJobs=#
- Possible to move cluster into or out of federation without Slurm restart (scontrol command)

# Unified Views

- Provide unified view of federated clusters by default
- Subset of federated clusters can be requested using -M option
- `queue`, `sinfo`, `sprio` will output cluster name in separate column
  - `queue` can sort by cluster name

CLUSTER	JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST(REASON)
ClusterA	2164339463	debug	wrap	brian	PD	0:00	10	(Resources)
ClusterA	2164329686	debug	wrap	brian	PD	0:00	10	(Priority)
ClusterB	2197893831	power	wrap	brian	PD	0:00	10	(Resources)
ClusterA	2164333731	long	wrap	brian	PD	0:00	10	(Resources)
ClusterC	2265002695	debug	wrap	brian	PD	0:00	10	(Priority)
ClusterC	2265002695	debug	wrap	brian	R	0:36	1	c11
ClusterA	2164333936	debug	wrap	brian	R	0:36	10	a[01-10]
ClusterB	2197893831	power	wrap	brian	R	0:36	1	b180
ClusterC	2265002695	debug	wrap	brian	R	0:36	102	c[1-10,15,51-100,120-140,143-160]
ClusterA	2164333932	short	wrap	brian	R	0:39	1	a52
ClusterA	12345	short	wrap	brian	R	0:39	1	d79
ClusterA	2164333933	short	wrap	brian	R	0:39	1	a113

# Schedule

- Development to begin 4Q 2015
- Included in next major release, version 16.05

# Questions?

