IBM Blue Gene

What's New with Blue Gene Job Schedulers

Todd Inglett
tinglett@us.ibm.com

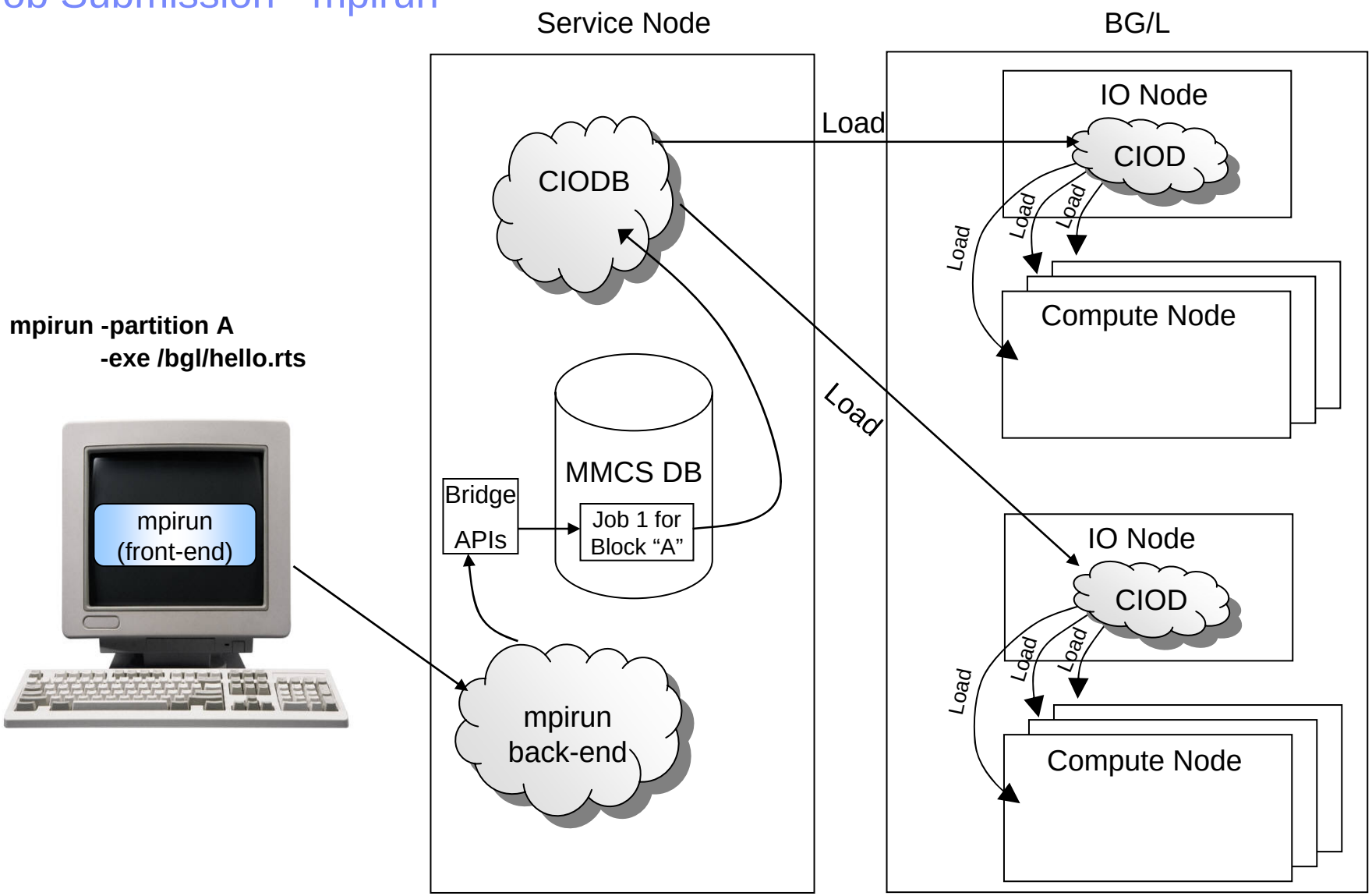
Agenda

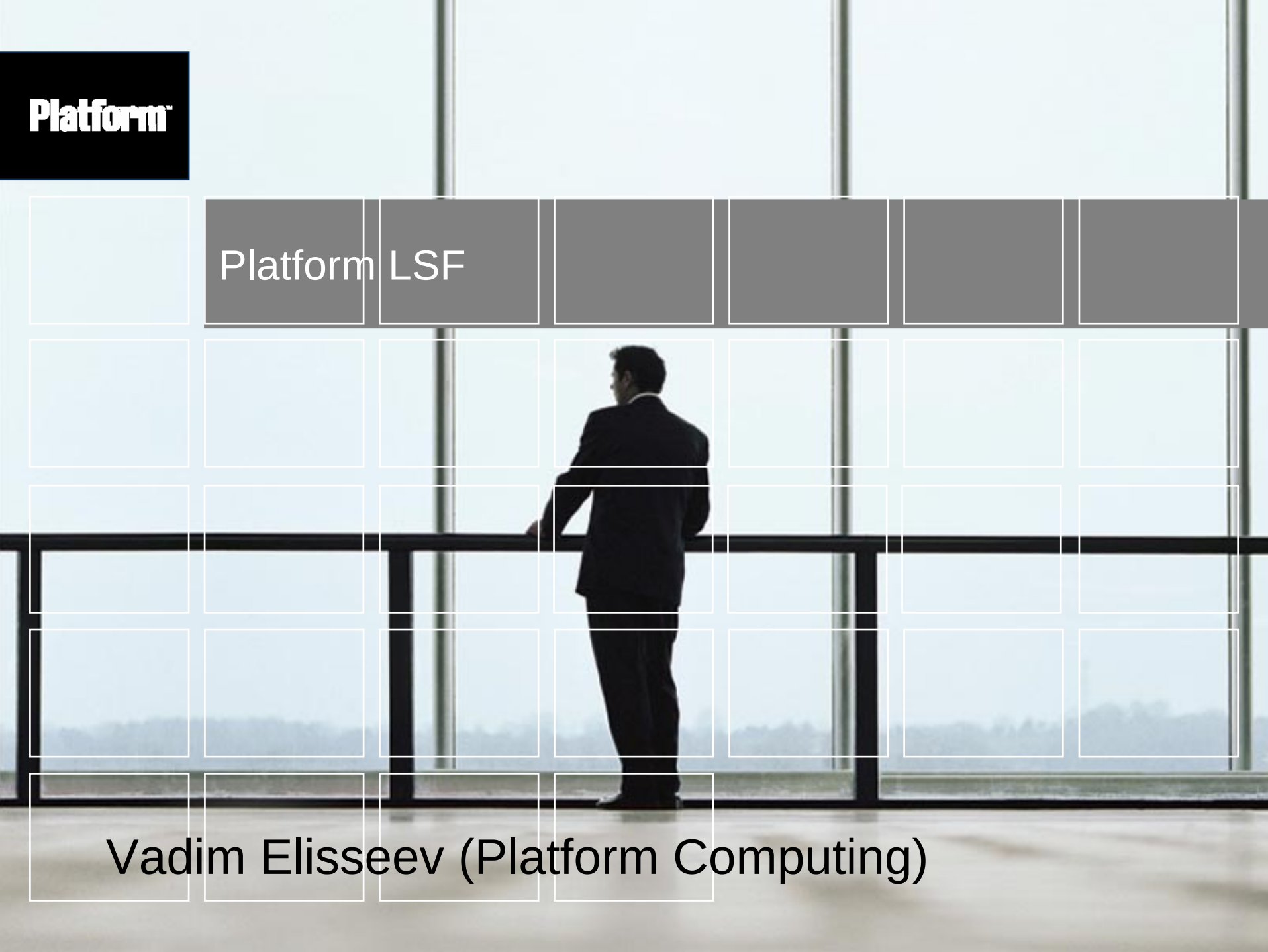
- **Review of Interactive and Batch Mode mpirun**
- **Platform LSF on BG/L Overview**
- **Altair PBS Pro on BG/L Overview**
- **LoadLeveler on BG/L Overview**
- **What's New with mpirun in V1R3**
- **What's New with Bridge APIs in V1R3**
- **Resources**

Review of Interactive and Batch Mode mpirun

- The **mpirun** program runs MPI jobs on BG/L.
- Use **mpirun** as a *stand-alone* program or in the framework of a *scheduling system.*
- You can invoke the **mpirun** program as a shell command. It allows the user to *interact* with the running job via the job's standard input, standard output, and standard error.
- When used by a scheduling system, the scheduler daemons can invoke **mpirun** on behalf of the user who submitted the job.

Job Submission - mpirun




Platform
Platform LSF
Vadim Elisseev (Platform Computing)

Platform

LSF for Blue Gene/L

Service Node runs two customized LSF daemons – LIM and RLA, that get various metrics of the BG/L machine from Bridge APIs:

- LIM gets information about number of nodes in the BG/L machine and sends to the MLIM
- RLA collects topology information about the BG/L machine
- RLA is also responsible for passing block allocate and deallocate requests to the Bridge APIs

LSF master daemons (MLIM and MBD) and scheduler (MBSHCD) can reside on any host designated as the master of the LSF cluster.

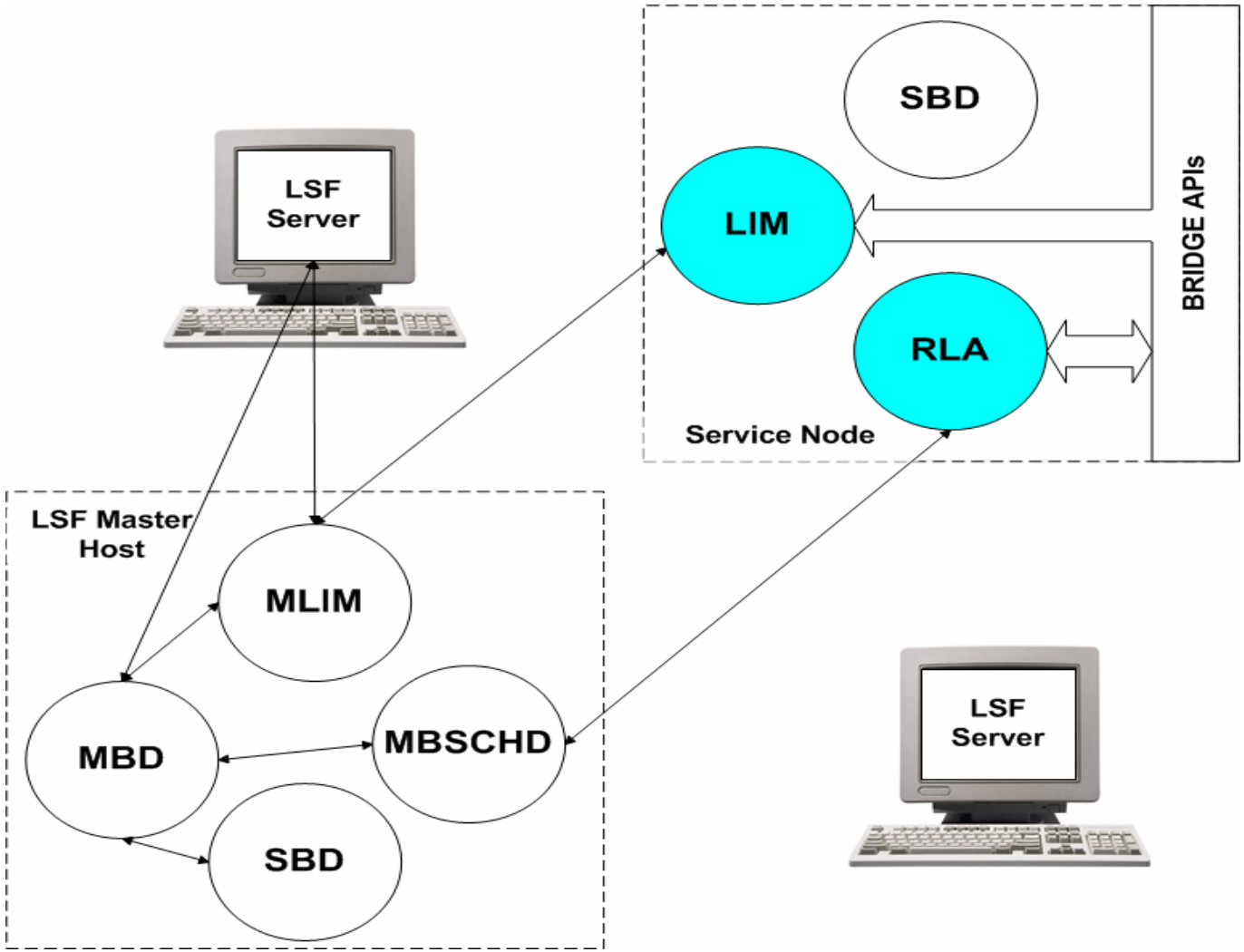
- MBSCHD makes a placement decision for a job based on resource requirements and availability
- MBSCHD talks with RLA daemon to get information about machine and to inform RLA about allocate/deallocate decisions.

Front End Nodes act as regular LSF server hosts.

SBD daemons are responsible for starting and monitoring jobs.

SBD on the master host is also responsible for starting and monitoring MBD.

LSF for Blue Gene/L (cont.)



Supported Scheduling Policies

Main scheduling policies currently supported for Blue Gene/L:

- Backfill scheduling
- Slots/nodes reservation
- Time based slots/nodes reservation
- Limited support for preemption
- Limited support for checkpointing

All of the above policies are working together with the topology aware scheduling:

- Full blocks with MESH connection type and small blocks are supported.

Job submission

User submits a job via ‘bsub’ command from any LSF host.

Job goes to the MBD

MBD passes job information to MBSCHD

MBSCHD makes a scheduling decision based on job resource requirements and resources availability

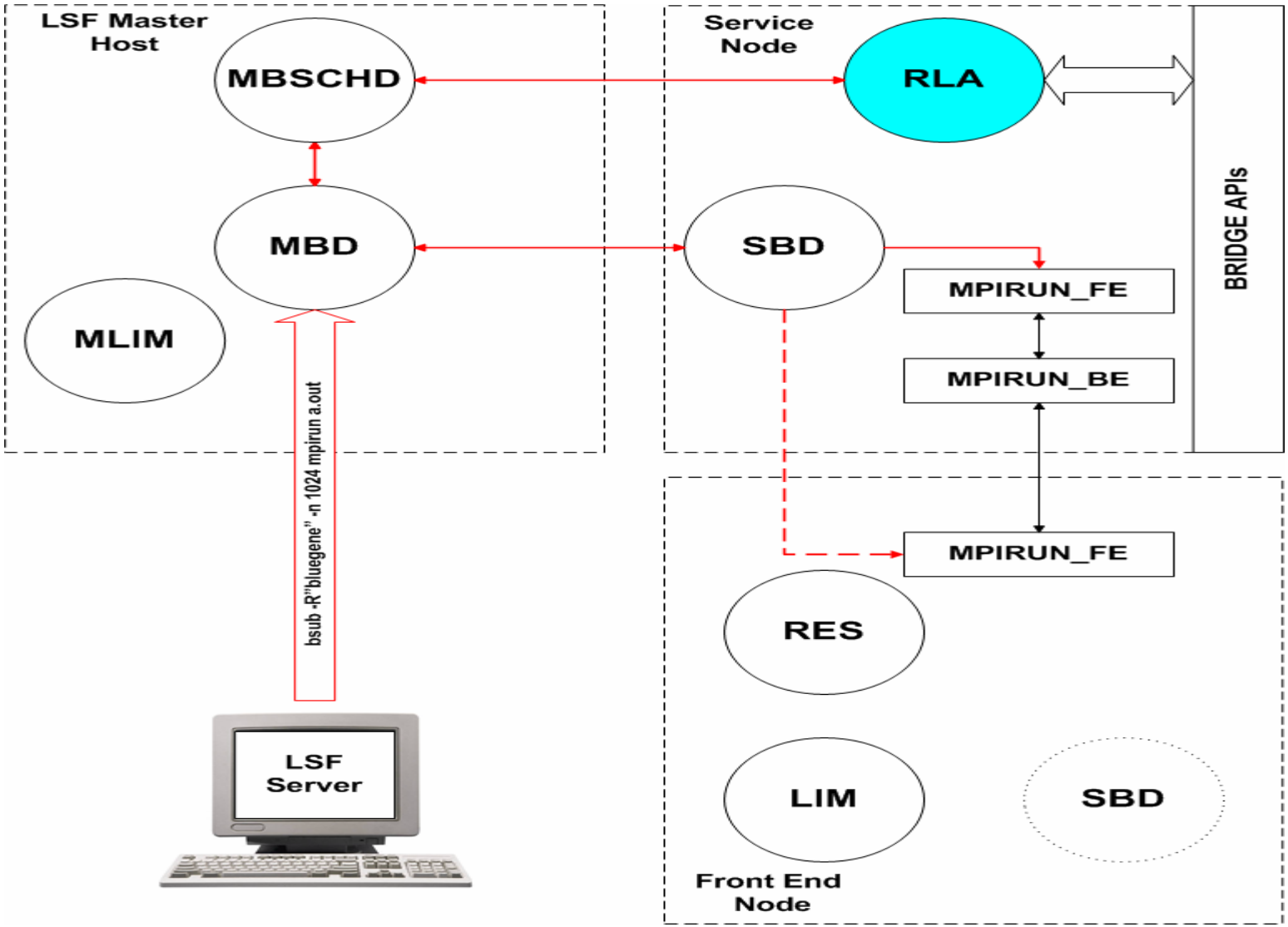
If the job can be dispatched, MBSCHD allocates partition for it via RLA. Partition ID becomes part of the job information.

Job is dispatched to SBD on the Service Node

SBD passes job block ID to the mpirun front end via an environment variable

It is possible to configure LSF in such a way that mpirun front end will be launched from front end nodes instead of the service node.

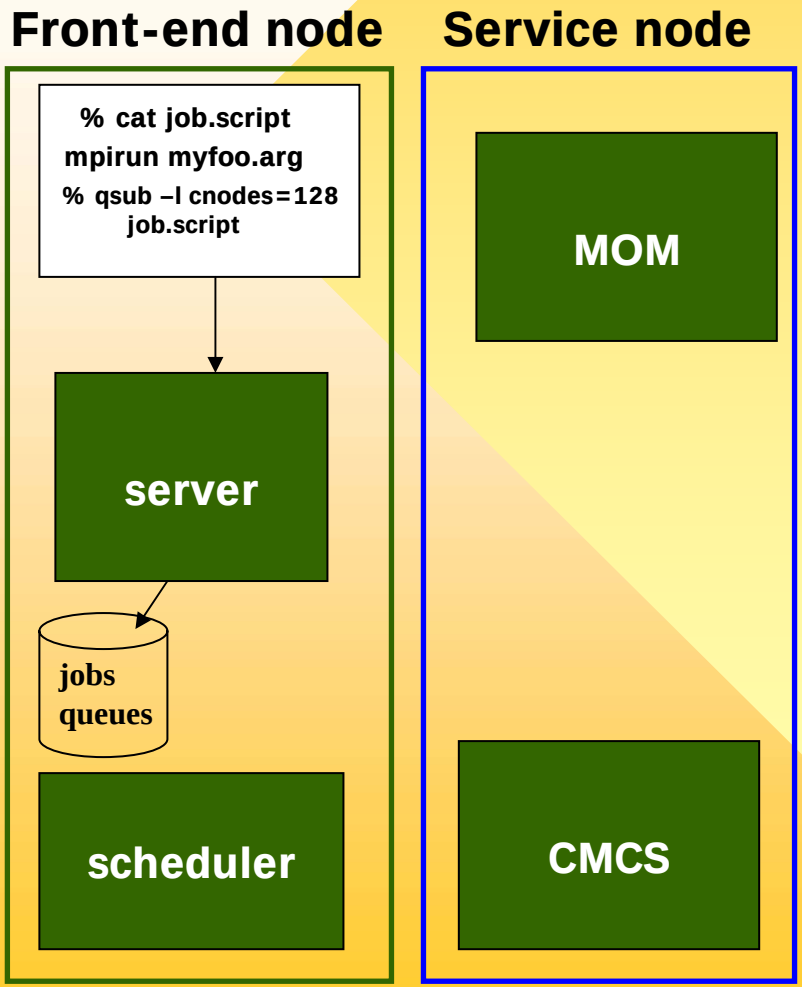
Job Submission (cont.)



Altair PBS Pro on Blue Gene
by
Albeaus Bayucan

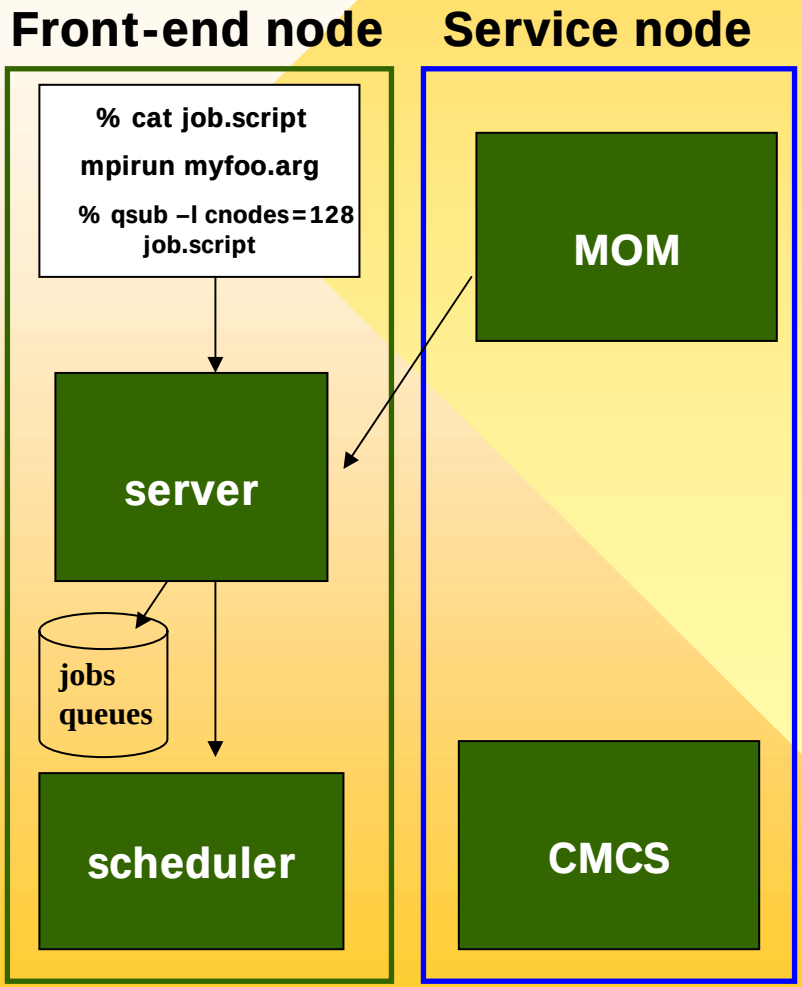
•April 10, 2006

PBS Pro for Blue Gene/L



- A user submits jobs by specifying resources and a job script.
- User requests BGL resources in terms of # of compute nodes via "cnodes" or by "ncpus".
- Within a user's job script is a call to "mpirun".
- PBS server puts the job in some queue, and info saved on disk.

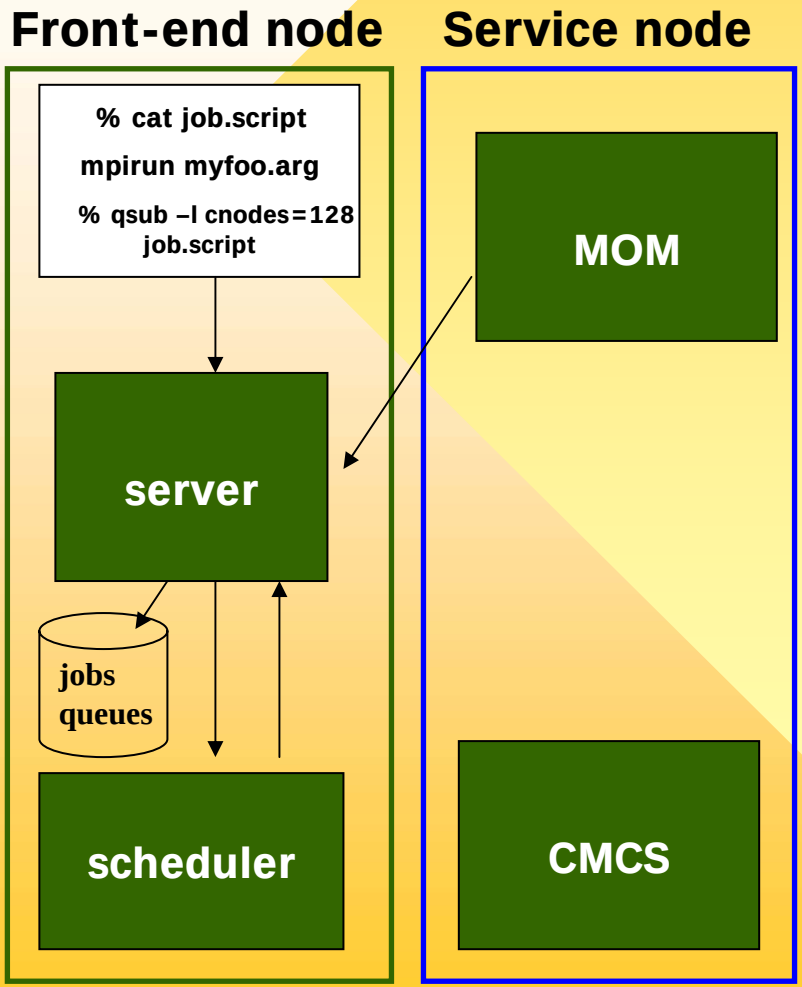
PBS Pro for Blue Gene/L



- Server informs the scheduler of the new job.

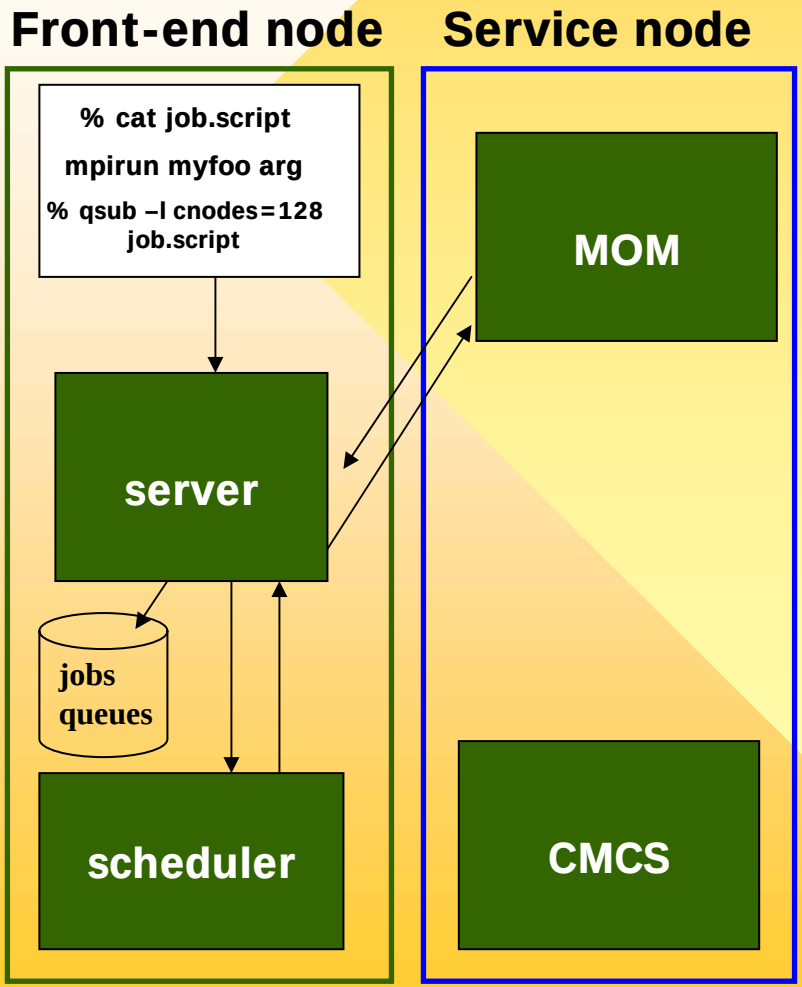
- Scheduler scans the list of available BGL partitions (periodically obtained from the server, as reported by mom) and decides on a partition to assign.

PBS Pro for Blue Gene/L



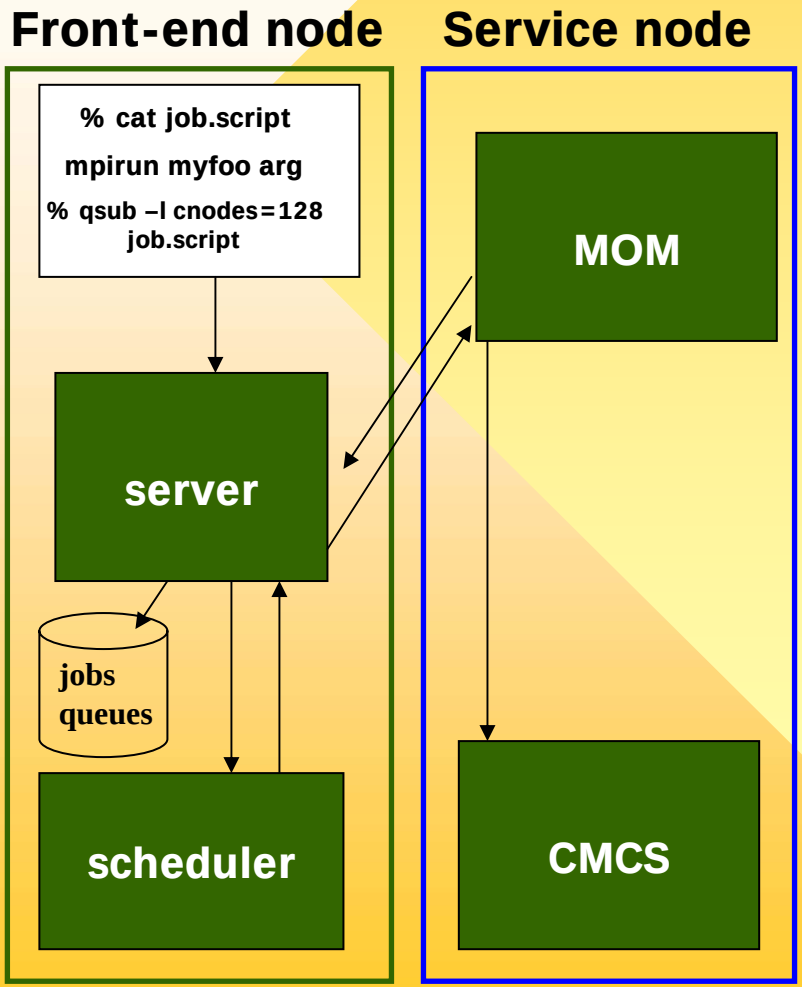
- Scheduler requests the server to run the job with the chosen partition.

PBS Pro for Blue Gene/L



Server sends the PBS job to the MOM on the service node.

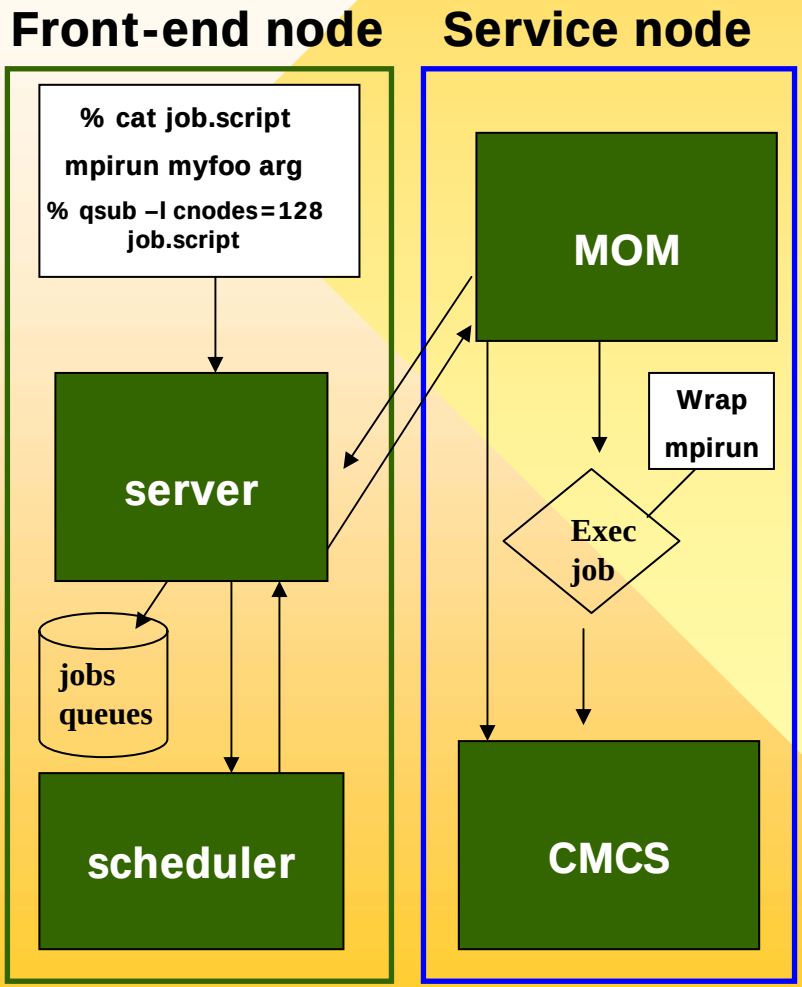
PBS Pro for Blue Gene/L



- Mom occasionally queries the BGL control system for a list of midplanes, nodecards, and predefined partitions, and info sent to the server.

- When mom receives a job from the server, it checks to make sure 1) scheduler-chosen partition is READY, FREE, or CONFIGURING, 2) the partition's midplanes/nodecards are UP, and 3) that no BGL job is running on it.

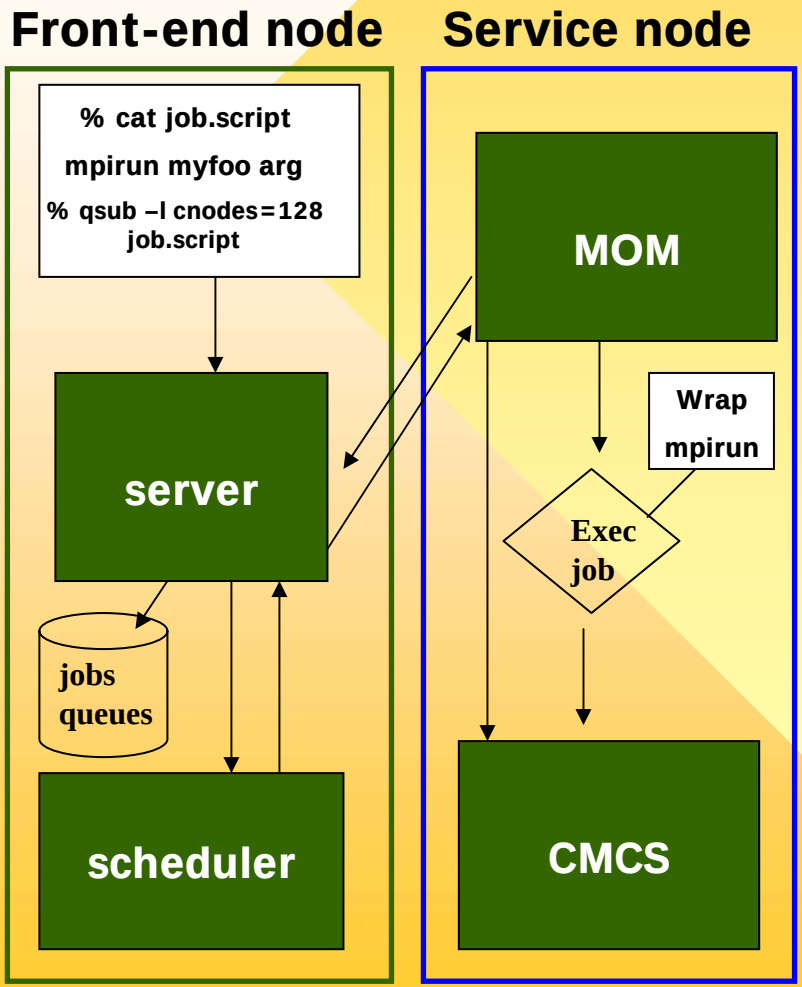
PBS Pro for Blue Gene/L



- Mom executes the job script, passing along the environment variable MPIRUN_PARTITION set to the chosen partition.

- Job script calls “mpirun” which executes the PBS-wrapped front-end mpirun.

PBS Pro for Blue Gene/L



Wrapped front-end mpirun prevents the following options from being specified:

-partition
-connect
-shape
-psets_per_bp
-host
-noallocate
-nofree

They're inconsistent with pre-assigned partition.

•mpirun takes care of booting the partition and instantiating the BGL job.

•When PBS job finishes or prematurely ends, MOM cleans up any BGL job left running on the partition.

Job Submission Examples

1) **User requests 512 cnodes, PBS assigns a partition containing 1 BP, wrapped mpirun passes "-np 512":**

```
% qsub -l select=cnodes=512 job.script
```

where job.script contains:

```
mpirun /scratch/foo/parallel_run 5 23
```

2) **User requests 32 cnodes, PBS assigns a partition containing 1 nodecard, wrapped mpirun passes "-np 32":**

```
% qsub -l select=cnodes=32 job.script
```

where job.script contains:

```
mpirun /scratch/foo/parallel_run 5 23
```

3) **User requests 128 cnodes and mpirun # of ranks of 80, PBS assigns a partition containing 4 nodecards, wrapped mpirun honors "-np 80":**

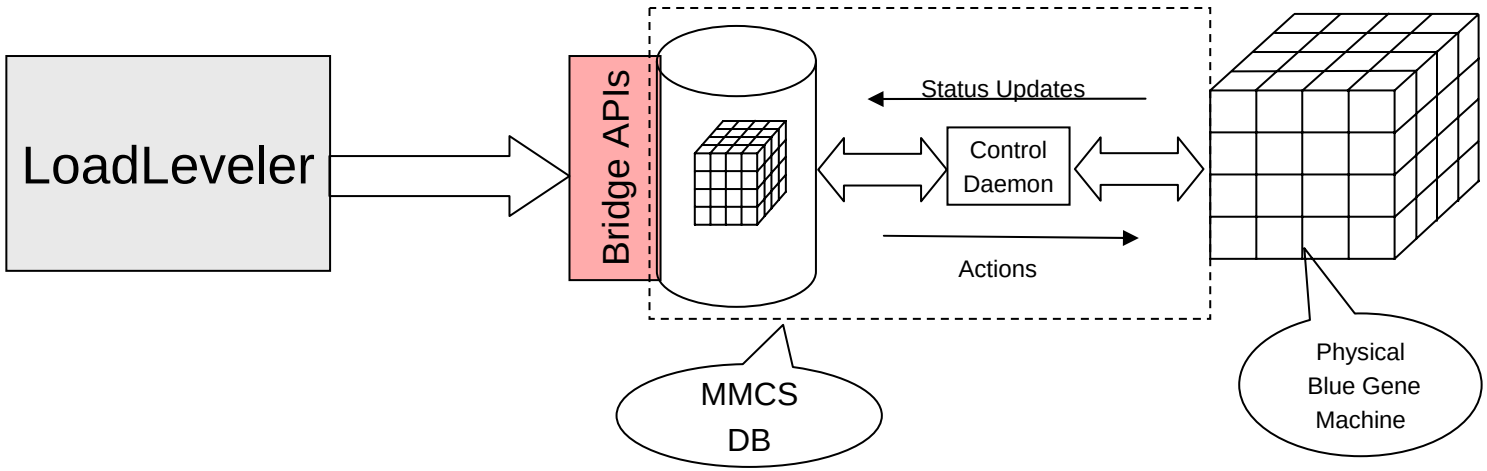
```
% qsub -l select=cnodes=128 job.script
```

where job.script contains:

```
mpirun –np 80 /scratch/foo/parallel_run 5 23
```

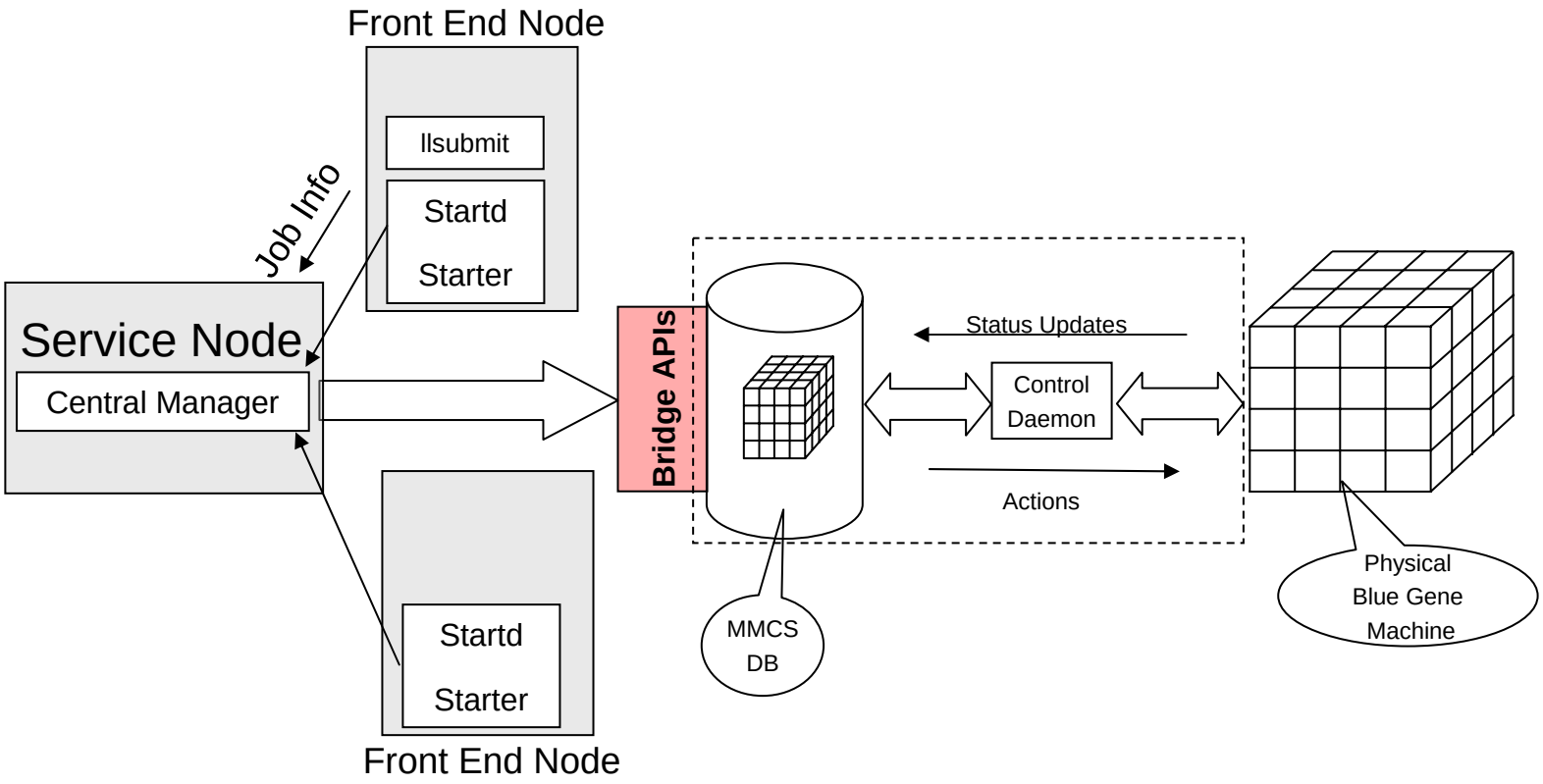
LoadLeveler (LL) on BG/L Overview

- LL multicluster (batch jobs only).
- LL Central Manager resides on Service Node.
- Front End Nodes run schedd. They handle job submission and act as gateways.
- LL does not support pre-emption or checkpoint on Blue Gene currently. The backfill scheduler must be used.
- It 'talks' to Blue Gene using the Bridge APIs and mpirun interface:
  - Gets the machine configuration
  - Decides which nodes to allocate for the jobs (i.e., partitions)
  - Monitors the job until it terminates



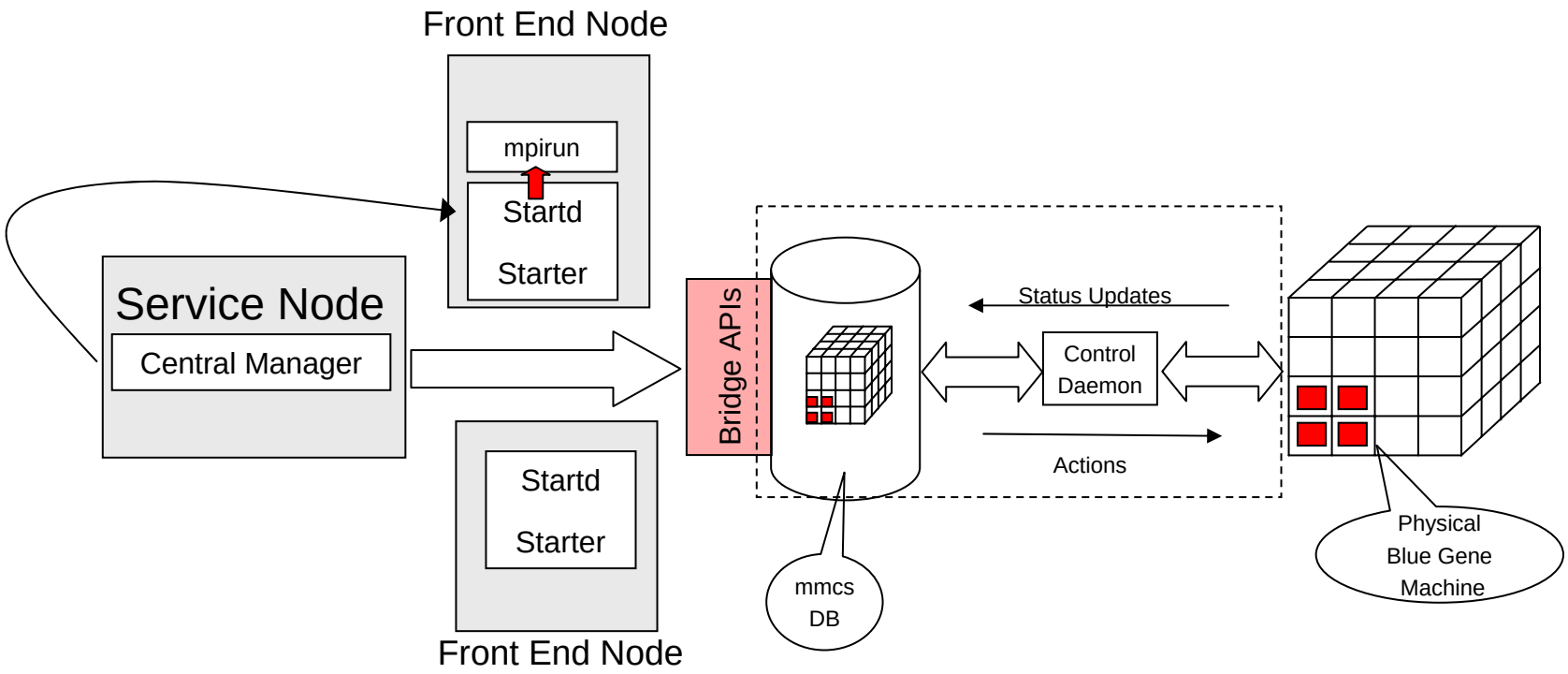
LoadLeveler Job Submission

- User on the Front End Node submits a job via llsubmit
  - llsubmit forwards to the Central Manager
  - Central Manager uses Bridge APIs to get snapshot of Blue Gene
  - Central Manager constructs a matching partition
  - Central Manager uses Bridge APIs to allocate partition



LoadLeveler Job Execution/Termination

- Central Manager chooses one of the Front End Nodes and launches mpirun
- mpirun runs the job
- Central Manager waits for mpirun to complete
- mpirun terminates and Central Manager frees and removes the partition via Bridge APIs



What's New with mpirun in V1R3

- **mpirun enhancements**
  - **Improved parameter checking**. Following rules are enforced:
    - Environment variables are always checked first and used.
    - If a command line parameter is specified when the equivalent environment variable has been specified, the environment variable is overridden without any warning messages.
    - If a command line parameter is passed more than once, it is an error. The only exceptions are the two parameters related to environment variables - those are cumulative and can be specified as many times as necessary to pick up multiple environment variables.
  - **Improved tracing for problem determination**
    - Useful for administrators or those implementing schedulers

What's New with Bridge APIs in V1R3

- **Bridge API enhancements**
  - **New max_psets_per_bp db.properties setting**:
    - New setting max_psets_per_bp can be specified in db.properties to allow a customer to specify a system wide default psets per base partition (midplane) value.
    - Setting will only be used if:
      - the caller of the Bridge API chose to default the value for psets per BP
      - the number provided is lower than that which was extracted from the database query
      - the number is a valid value, such as 32, 16, 8, 4
    - This feature is being done to make it easier for job schedulers to support a customer environment where not all the I/O nodes are attached to the network.
  - **New job metrics for completed jobs**:
    - New rm_get_data specifications for jobs that have completed (jobs that are in the history table). The current rm_get_data spec RM_JobInHist can be used to determine if a job has completed or not.
    - The new rm_get_data specs are:
      - RM_JobStartTime - the start time of the job returned as a char[27] with format of yyyy-mm-dd-hh.mm.ss.nnnnnn
      - RM_JobEndTime - the end time of the job returned as a char[27] with format of yyyy-mm-dd-hh.mm.ss.nnnnnn
      - RM_JobRunTime - an integer value with the run time of the job in seconds (calculated based on difference between start and end time)
      - RM_JobComputeNodesUsed - an integer value with the number of compute nodes that the job used

What's New with Bridge APIs in V1R3 - continued

- **Bridge API enhancements - continued**
  - **New rm_get_data spec for compute node memory size**:
    - RM_BPComputeNodeMemory - will return an enum value with the memory size of the compute nodes for the base partition.
    - We anticipate that a job may have specific memory requirements and this new specification will allow a job scheduler to satisfy the memory requirements in mixed memory environment.
    - The following are the enum values that can be returned. In a normal customer environment the memory options are 512 MB and 1 GB.

    - enum rm_BP_computenode_memory {
      RM_BP_COMPUTENODE_MEMORY_256M,
      RM_BP_COMPUTENODE_MEMORY_512M,
      RM_BP_COMPUTENODE_MEMORY_1G,
      RM_BP_COMPUTENODE_MEMORY_2G,
      RM_BP_COMPUTENODE_MEMORY_4G,
      RM_BP_COMPUTENODE_MEMORY_NAV
      };

Scheduler Features

	LoadLeveler	LSF	PBS Pro	SLURM
Partitioning support	Dynamic and Static	Dynamic and Static	Static	Static Static overlap May/06
Small block support	32/128 small blocks	32/128 small blocks	32/128 small blocks	128 small blocks 32 small blocks May/06
Scheduling options	Backfill Job classes Job priority	Backfill Nodes reservations Time reservations	FIFO Whole pool evaluation Job priority	FIFO Backfill (not available on BG/L)
Preemption	Limited - preempted job is killed	Limited - preempted job is killed	No	No
Checkpoint support	No	Limited on BG/L	No	Plugin API
Heterogeneous Platform support	AIX, Linux, BG/L	AIX, Linux, BG/L, Cray, HP, SGI Altix, Solaris, Windows	AIX, Linux, BG/L, HP, SGI Altix, Cray, Solaris, Windows	AIX, Linux, BG/L, HP (possibly others since Open Source)
Scheduling API	Yes	Yes	Yes	Yes
Open source	No	No	No	Yes

*Cobalt shown in next presentation

Resources

- **mpirun**
  - Chapt. 2 in BG/L Sys Admin redbook: http://www.redbooks.ibm.com/abstracts/sg247178.html
- **LoadLeveler**
  - http://www.ibm.com/servers/eserver/clusters/software/loadleveler.html
  - http://publib.boulder.ibm.com/infocenter/clresctr/index.jsp?topic=/com.ibm.cluster.loadl.doc/loadl331/am2ug30306.html
- **Platform LSF**
  - http://www.platform.com/Products/Platform.LSF.Family
- **Altair PBS Pro**
  - http://www.altair.com/software/pbspro.htm
- **SLURM**
  - http://www.llnl.gov/linux/slurm/bluegene.html
- **Cobalt**
  - http://www-unix.mcs.anl.gov/cobalt/bgl.xml