Partitions and scheduling, running jobs effectively on Graham and Cedar

SHARCNET™

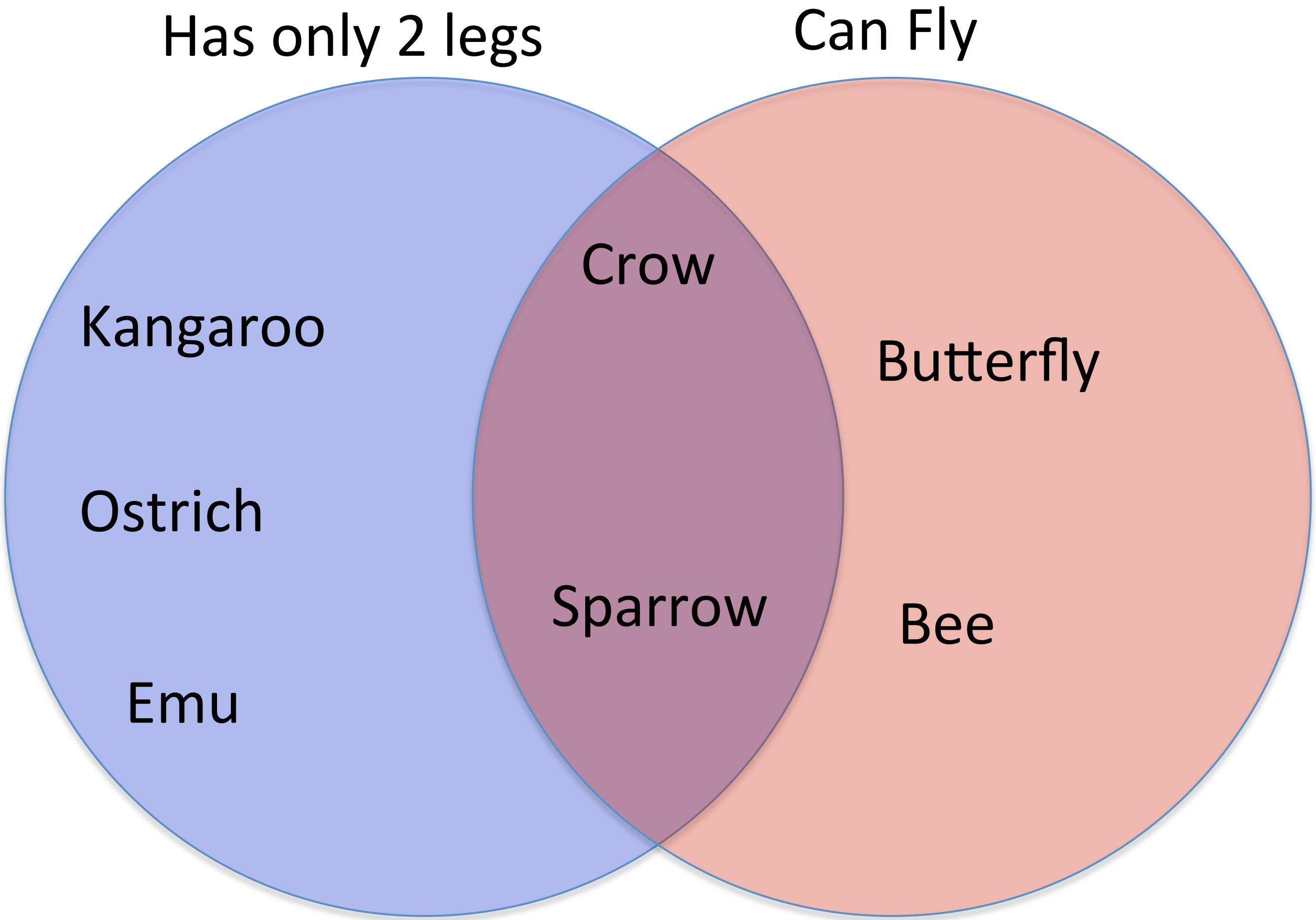
compute canada | calcul canada

WESTGRID

Partitions

- Your job will automatically be assigned
- Somewhat like queues or classes in pbs/ torque and moab.
- A job can be in multiple partitions simultaneously, and can have multiple a per partition priorities.
- A node can be in multiple partitions simultaneously

Venn Diagram



Has only 2 legs

Can Fly

Kangaroo

Ostrich

Emu

Crow

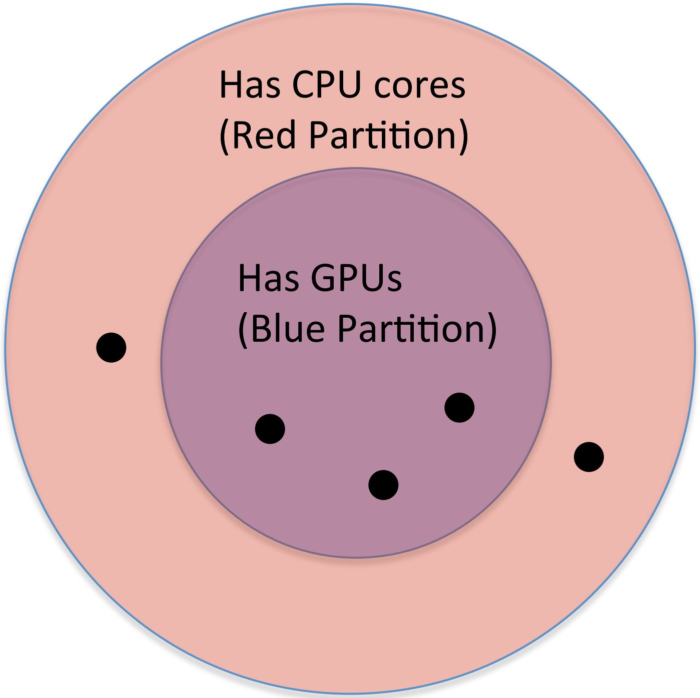
Sparrow

Butterfly

Bee

Partition Venn Diagram

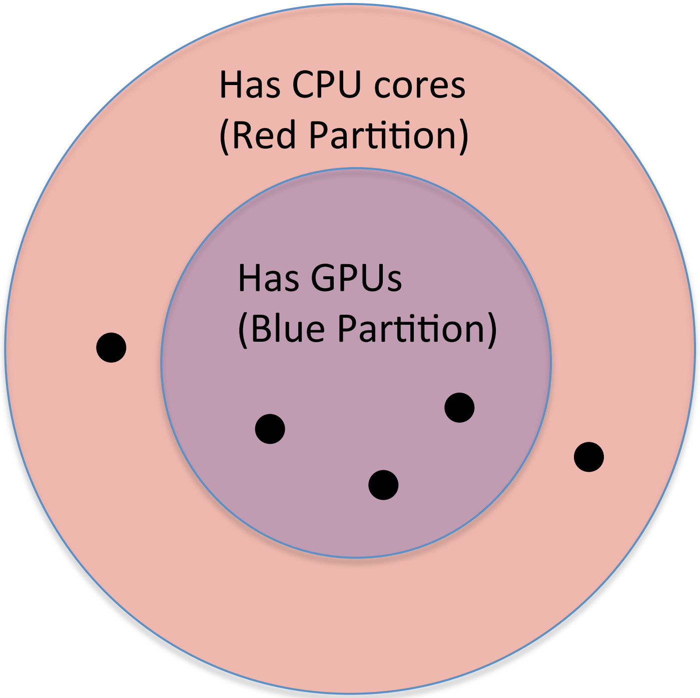
(on a 5 node imaginary cluster)



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
  - The two nodes with no gpu in the red partition may be idle but a job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority blue jobs.

Partition Venn Diagram

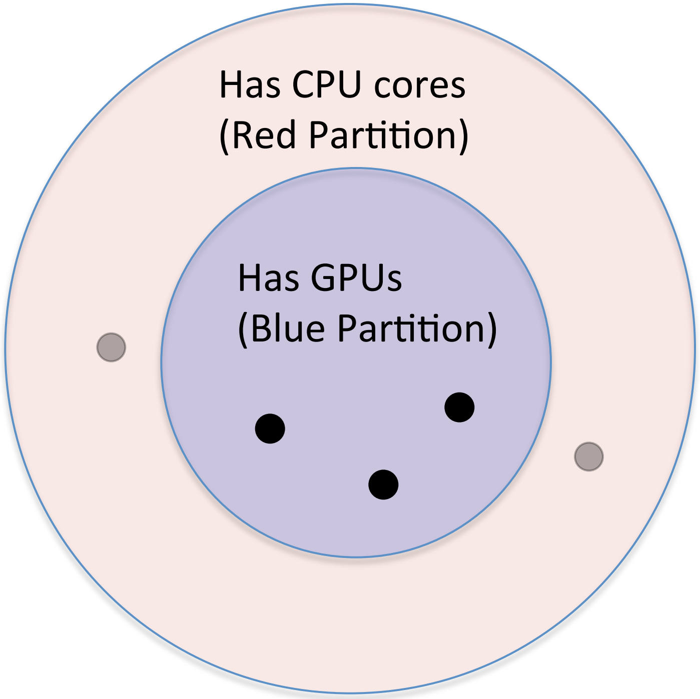
(on a 5 node imaginary cluster)



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
  - The two nodes with no gpu in the red partition may be idle but a job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority blue jobs.

Partition Venn Diagram

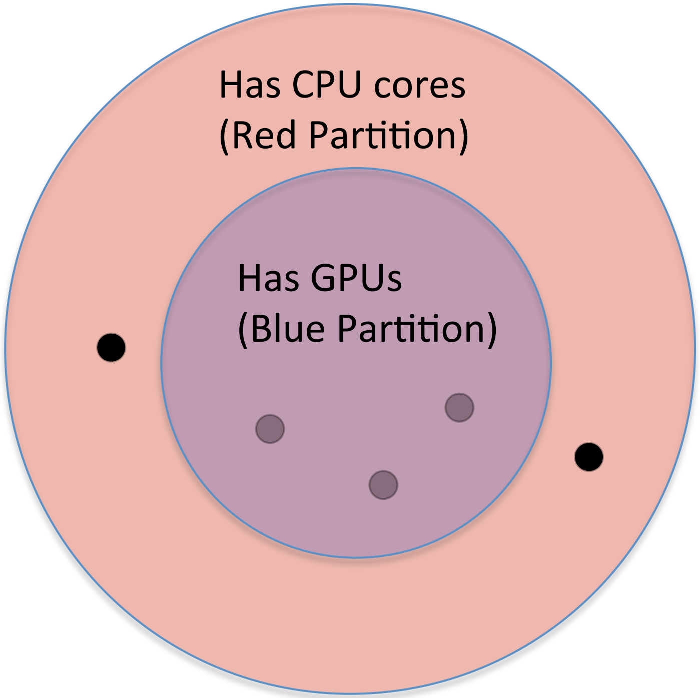
(on a 5 node imaginary cluster)



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
  - The two nodes with no gpu in the red partition may be idle but a job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority blue jobs.

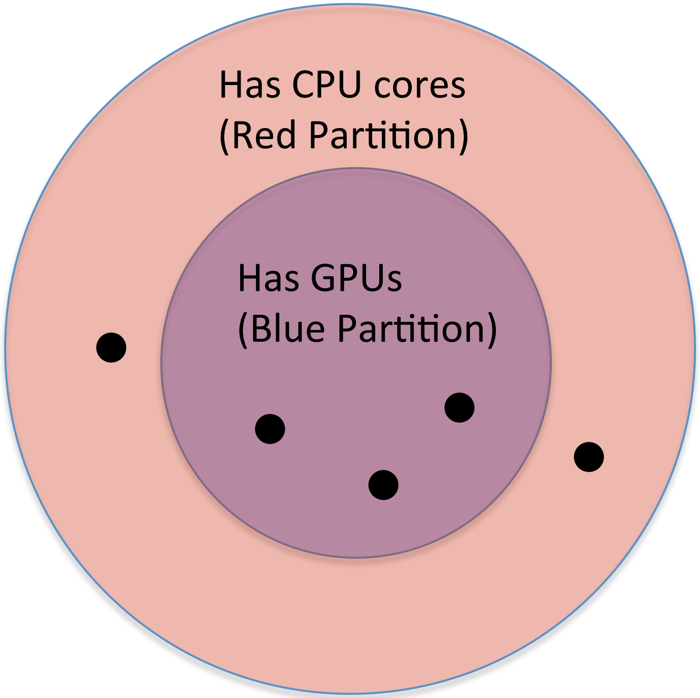
Partition Venn Diagram

(on a 5 node imaginary cluster)



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs (In the red partition but not in the blue)
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
  - The two nodes with no gpu in the red partition may be idle but a job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority blue jobs.

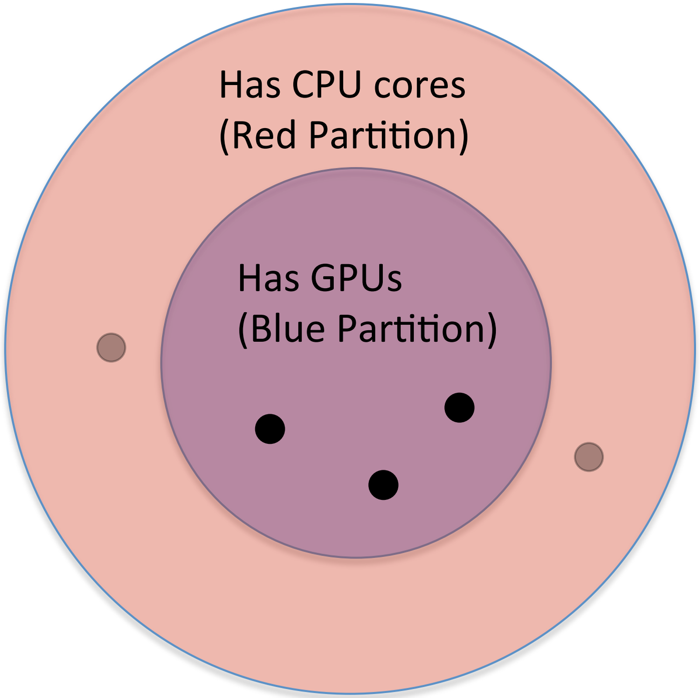
Partition Venn Diagram



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
  - The two nodes with no gpu in the red partition may be idle but a job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority blue jobs.

Partition Venn Diagram

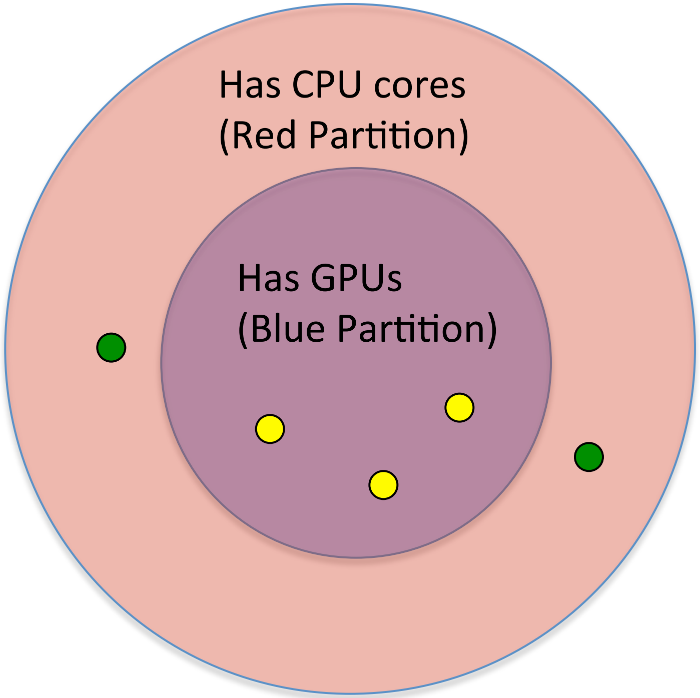
(on a 5 node imaginary cluster)



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
  - The two nodes with no gpu in the red partition may be idle but a job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority blue jobs.

Partition Venn Diagram

(on a 5 node imaginary cluster)



- Black dots are nodes
- In this example we have:
  - 5 nodes with CPUs (Red partition)
  - 3 nodes with GPUs (Blue partition)
  - 2 nodes have CPUs but not GPUs
- A Job that requires CPUs (red partition) can run on any of the 5 nodes
- A job that requires GPUS (blue partition) can run on any of the 3 nodes.
- In the case that the two nodes with no gpus in the red partition may be idle(green) and 3 nodes with gpus may be busy.
  - A job that requires a GPU node (from the blue partition) will be unable to start if no GPU nodes are idle. A job that requires CPUs only (Red partition) will be able to start immediately, even when there are higher priority jobs in the blue partition.

🟢 Idle node

🟡 Busy node

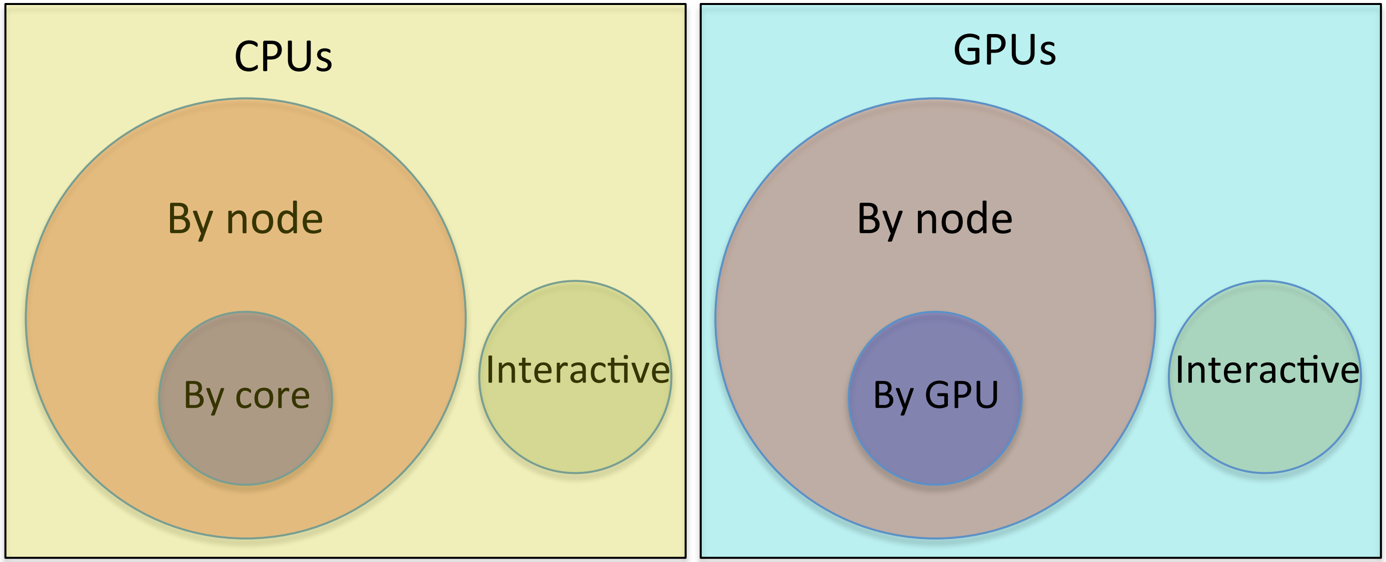
Node types on Cedar

Total Mem TB	Cores	Memory	GPUS	Number of Nodes	Partition type
1/8	32	4GB/core		576	cpubase
1/4	32	8GB/core		182	cpubase
1/2	32	16GB/core		24	cpularge
1.5	32	48GB/core		24	cpularge
3	32	96GB/core		4	cpularge
1/8	24	32GB/GPU	4	114	gpubase
1/4	24	64GB/GPU	4	132	gpularge

Node types on Graham

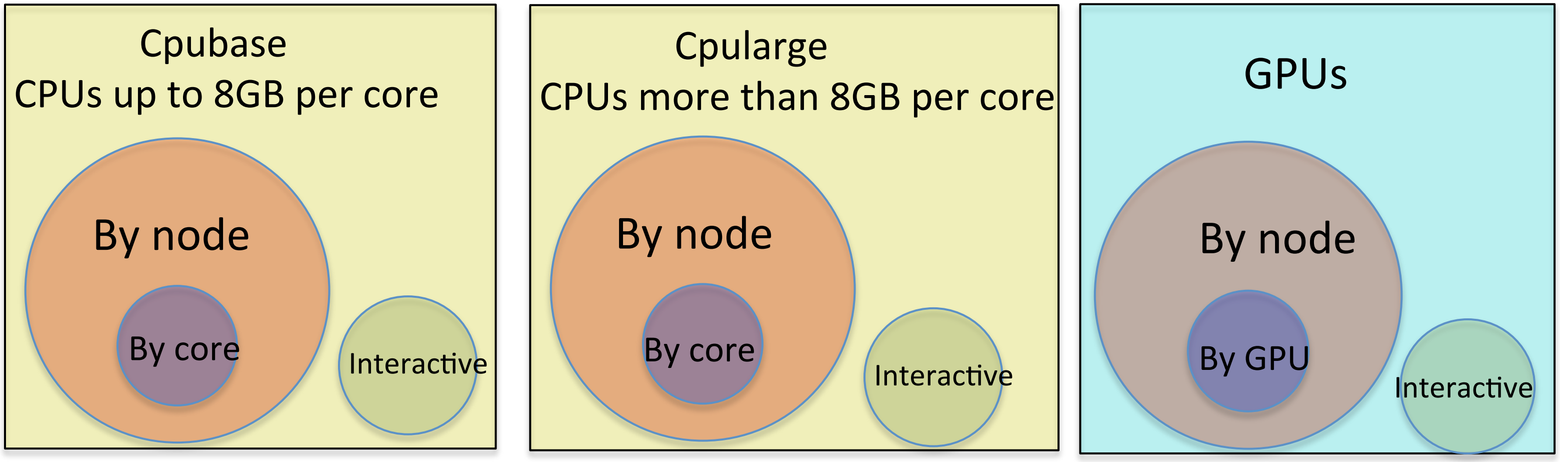
Total Mem TB	Cores	Memory	GPUS	Number of Nodes	Partition Type
1/8	32	4GB/core		800	cpubase
1/4	32	8GB/core		55	cpubase
1/2	32	16GB/core		24	cpularge
3	32	96GB/core		3	cpularge
1/8	32	32GB/GPU	4	114	gpubase

Partitions on Cedar and Graham



- Separate partitions for GPUs and CPU request
- Nodes that are in the by core partition are also in the by node partition, the reverse is not always true.
- There are separate interactive (testing) partitions with dedicated nodes for interactive usage.

Partitions on Cedar and Graham

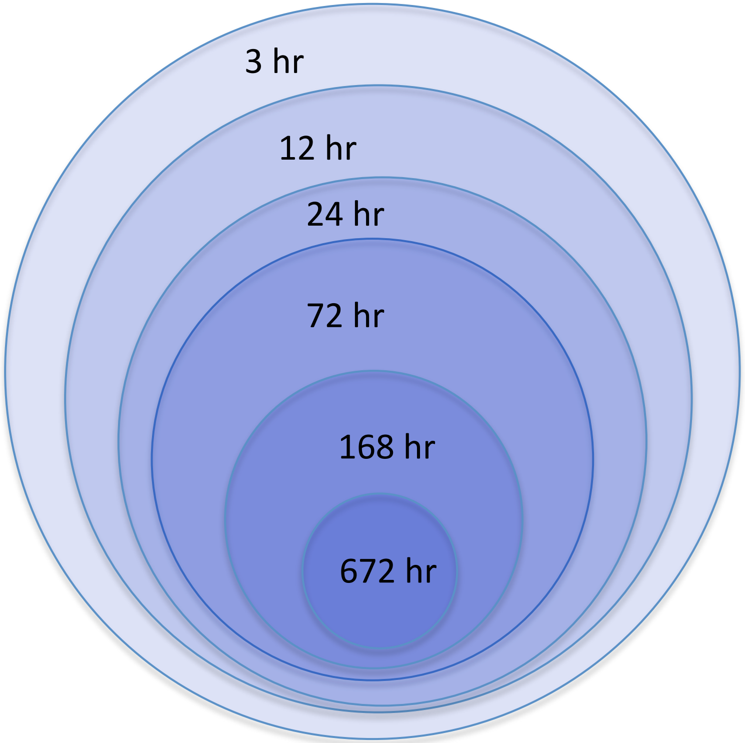


- Separate partitions for large memory Nodes and jobs that have more than 8 GB RAM and smaller memory nodes and jobs.
  - This is done to disallow low memory jobs from stopping a large memory job from running quickly on the few expensive large memory nodes we have.

Partitions why the complexity?

- If we allowed serial jobs to run on all nodes, the chances that there was a node that had all 32 cores not used or coming to an end soon would be very small.
  - if ½ the cluster was empty and the job distributed randomly the chances a any particular node to be empty = $\frac{1}{2^{32}} = \frac{1}{4,294,967,296}$
- As a consequence whole node jobs would in practice all have to wait (max walltime) time to start regardless of priority.
- If the whole cluster only allows allocation to jobs by node jobs by core will not run or people would ask for a node and use a single core.

Partitions on Cedar and Graham



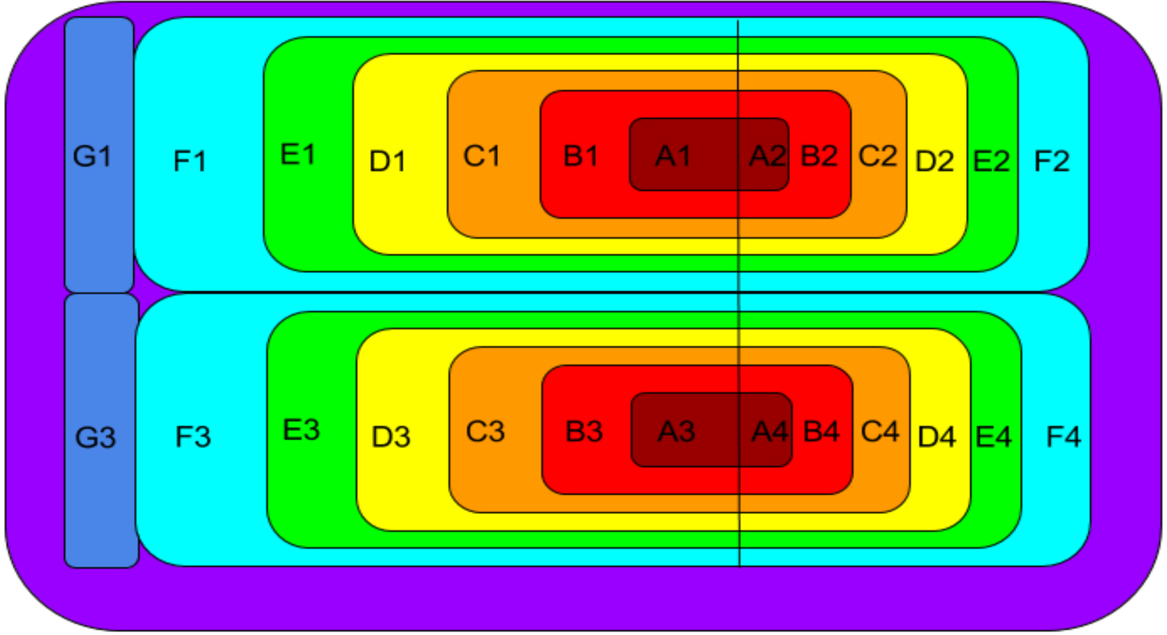
- There are partitions based upon how long the maximum walltime your job has.
- Your job ends up in the shortest walltime partition that has a longer walltime than your job
- The shorter walltime partitions include all the nodes of longer walltime partitions.

Partitions why the complexity?

- Some jobs need to run a long time
  - Commercial code that does not checkpoint
  - Checkpoints can take a very long time
- If we allow all nodes to run long walltime jobs
  - It would take a long time for resources to be come available, researchers that need to run short jobs and analyze the result before running another would find the system unusable.
  - People that can divide their work arbitrarily would run long walltime jobs as they have already waited a long time for their job to start, making the situation worse.
- CC has dealt with the situation in the past by having different cluster each has different walltimes. But there are not enough clusters to do this anymore.
- The solution of concentric partitions on larger cluster allows us to more efficiently address diverse user needs.

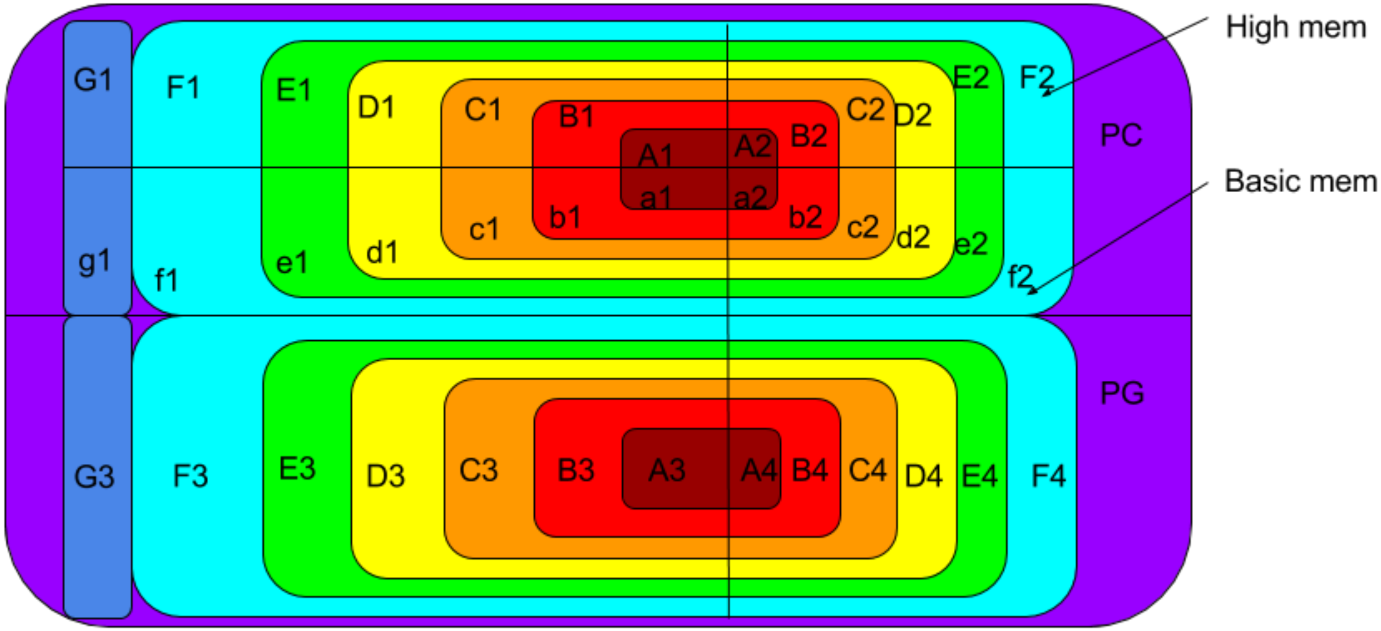
Partitions on Cedar and Graham

Walltime	Whole node cpu	By core cpu	Whole node gpu	By gpu
768 hr	A1 + A2	A2	A3 + A4	A4
168 hr	B1 + B2	B2	B3 + B4	B4
72 hr	C1 + C2	C2	C3 + C4	C4
24 hr	D1 + D2	D2	D3 + D4	D4
12 hr	E1 + E2	E2	E3 + E4	E4
3 hr	F1 + F2	F2	F3 + F4	F4
interactive	G1		G3	
Preemtable				



Partitions on Cedar and Graham

Walltime	Whole node cpu	By core cpu	Whole node gpu	By gpu
768 hr	A1 + A2	A2	A3 + A4	A4
168 hr	B1 + B2	B2	B3 + B4	B4
72 hr	C1 + C2	C2	C3 + C4	C4
24 hr	D1 + D2	D2	D3 + D4	D4
12 hr	E1 + E2	E2	E3 + E4	E4
3 hr	F1 + F2	F2	F3 + F4	F4
Low Priority Backfill Short Jobs	F1 + F2 + f1 + f2		F3 + F4	
interactive	G1 + g1		G3	
Preemptable	PC		PG	

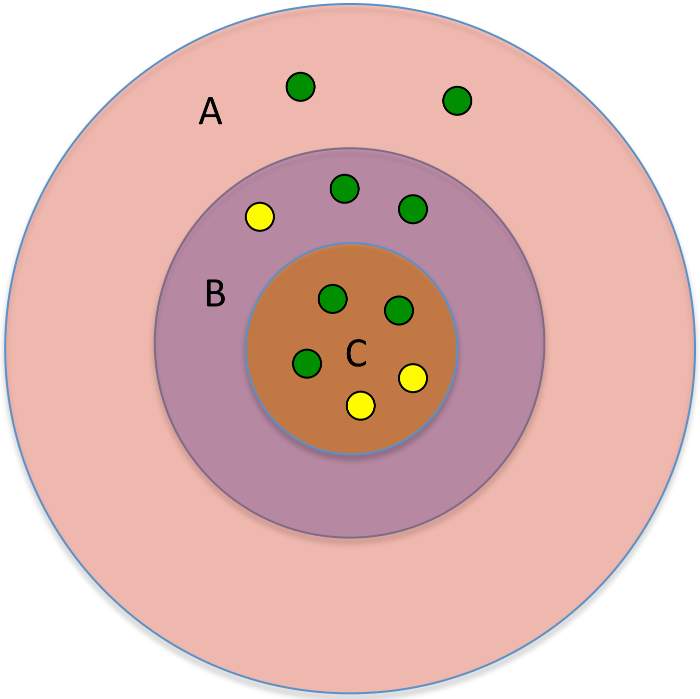


Partition Stats

(CC script)

```
Node type |                     Max walltime
          |   3 hr   |  12 hr  |  24 hr  |  72 hr  |  168 hr |  672 hr |
----------|-------------------------------------------------------------
       Number of Queued Jobs by partition Type (by node:by core)
----------|-------------------------------------------------------------
Regular   |    0:2   |   0:0   |   0:559 |   0:110 |   0:664 |   0:15  |
Large Mem |    0:0   |   0:0   |   0:0   |   0:0   |   0:0   |   0:0   |
GPU       |    6:78  |   6:206 |   4:35  |   7:6   |   3:3   |   2:0   |
GPU Large |    0:-   |   0:-   |   0:-   |   0:-   |   0:-   |   0:-   |
----------|-------------------------------------------------------------
      Number of Running Jobs by partition Type (by node:by core)
----------|-------------------------------------------------------------
Regular   |    0:2   |   1:159 |  16:33  |  13:100 |  17:22  |  51:3   |
Large Mem |    0:0   |   0:0   |   0:0   |   0:0   |   0:0   |   1:0   |
GPU       |   10:0   |   3:0   |   0:0   |   8:0   |   1:0   |   0:0   |
GPU Large |    0:-   |   0:-   |   1:-   |   1:-   |   2:-   |   1:-   |
----------|-------------------------------------------------------------
        Number of Idle nodes by partition Type (by node:by core)
----------|-------------------------------------------------------------
Regular   |  137:30  | 110:18  |  74:11  |  74:11  |  19:1   |  18:0   |
Large Mem |    2:1   |   2:1   |   2:1   |   2:1   |   2:1   |   2:1   |
GPU       |    0:0   |   0:0   |   0:0   |   0:0   |   0:0   |   0:0   |
GPU Large |   13:-   |   9:-   |   7:-   |   5:-   |   0:-   |   0:-   |
----------|-------------------------------------------------------------
       Total Number of nodes by partition Type (by node:by core)
----------|-------------------------------------------------------------
Regular   |  691:317 | 635:285 | 542:223 | 478:191 | 255:95  | 159:39  |
Large Mem |   50:5   |  50:5   |  50:5   |  44:5   |  17:2   |   3:2   |
GPU       |  112:64  |  96:64  |  96:47  |  63:31  |  32:8   |  16:4   |
GPU Large |   32:-   |  28:-   |  24:-   |  20:-   |   8:-   |   4:-   |
----------|-------------------------------------------------------------
```

Partitions and priority example

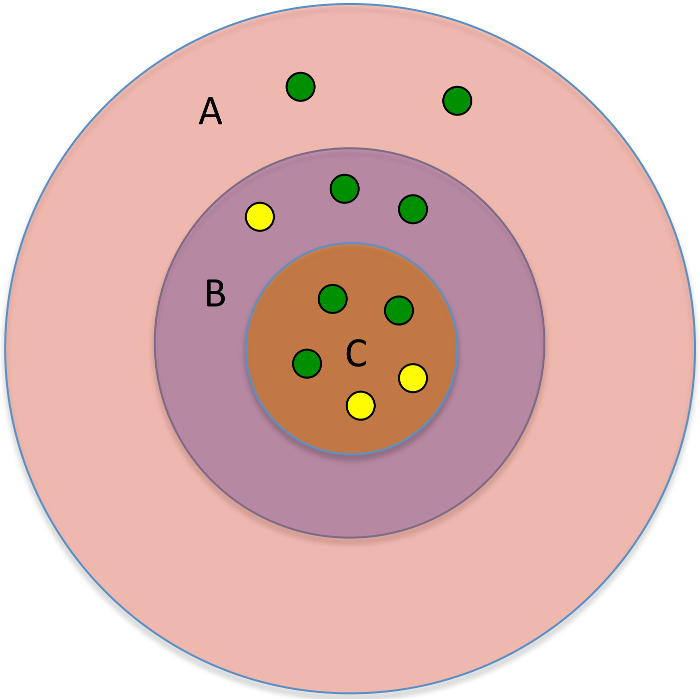


- Partition A has 3 hour walltime and includes all the nodes of this type on the cluster
- Partition B is the largest partition that your job can run in.
- Partition C is a subset of partition B and contains jobs that have a longer walltime and nodes that can run those jobs.
- Each small green circle represents a idle an idle node
- Each small yellow circle represents a busy node

🟢 Idle node

🟡 Busy node

Partitions and priority example



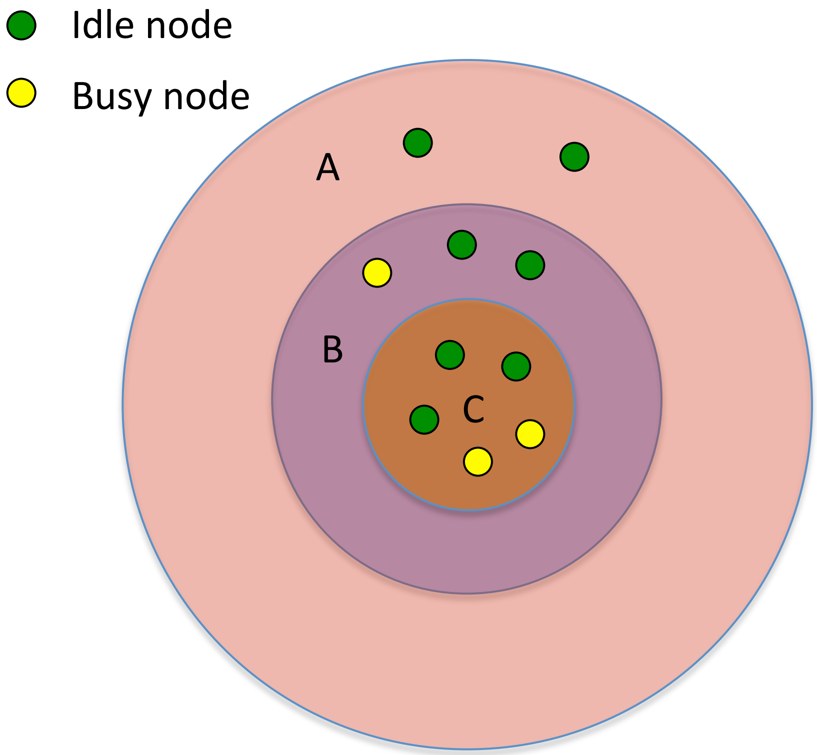
Lets assume we have 3 jobs:

- Highest priority job (1) in partition C that requires 4 nodes.
- 2nd highest job in partition job (2) in partition A that requires 5 nodes.
- Our job in partition B that requires 2 nodes

🟢 Idle node

🟡 Busy node

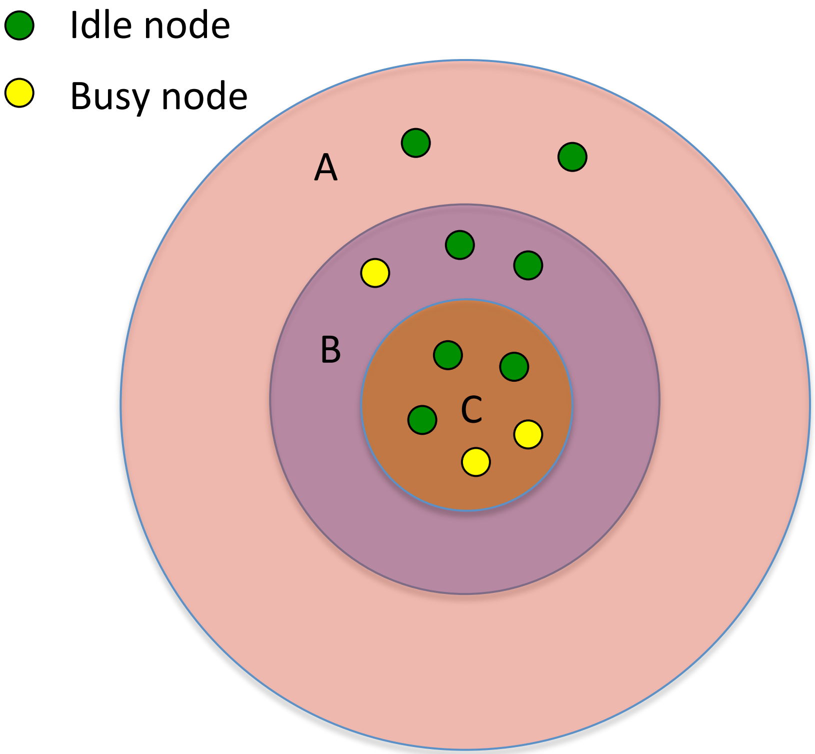
Partitions and priority example



- Highest priority job (1) in partition C that requires 4 nodes.
- 2nd highest job (2) in partition A that requires 5 nodes.
- Our job (3) in partition B that requires 2 nodes

- Job 1 cannot run as there are only 3 idle nodes in partition C.
    - A reservation is created for the idle nodes in partition C and the first of the busy nodes that will become available.
- Job 2 likely cannot run either as it needs one of the nodes reserved by job 1, and unless job 2 can finish before job 1 starts it will not be able to run.
- Job 3 will likely not run as well because it requires resources (nodes) that are reserved by other higher priority jobs.

Partitions and priority example

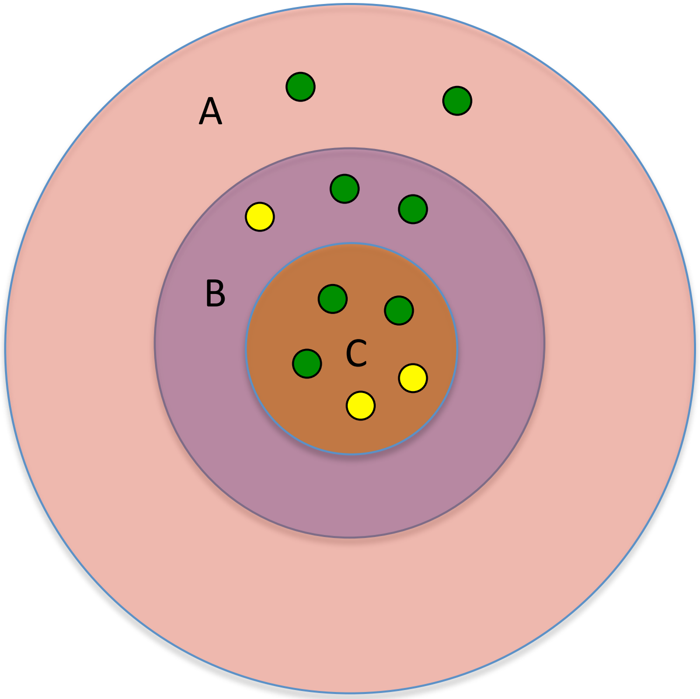


- Highest priority job (1) in partition C that requires 4 nodes.
- 2nd highest job (2) in partition A that requires 5 nodes.
- Our job (3) in partition B that requires 2 nodes

This cluster is 70% idle and and jobs cannot run why?

- The example cluster is small and the jobs are large in comparison
- There are no short single node jobs that can fill in these empty nodes.
- This example was created to show a worse case scenario

Partitions and priority lessons learned



🟢 Idle node

🟡 Busy node

- Submit smaller, shorter jobs
- When looking at priority and why your job is not running, look at the priority of other jobs in the partitions that are either a subset or superset of your job.
- The situation in Compute Canada will get better when Niagara is up as that system is designed for large jobs. The types of jobs on Cedar and Graham will become less diverse and we will be better able to efficiently schedule similar and smaller jobs on Graham and Cedar.

Questions

Other resources and sessions

Invitation to the WestGrid Workshop to be Held

Oct 24-26 1:00 - 3:00 pm MDT (3:00-6:00pm Eastern)

http://goo.gl/rFQv7W

In addition to Scharcnet training you are all welcome in WestGrid training events:

https://www.westgrid.ca/events/westgrid-training-events