

BGLMapping

Mapping processes on Blue Gene

Order of assignment

The BGLMPI_MAPPING variable can control the order in which tasks are assigned to nodes. The default ordering by dimension is *XYZT*. The *T* dimension is for the first or second processor within a node. So for instance, submitting a job with the default mapping (`cqsub -m vn -n 1 ...`) will give you two processes on two **different** nodes.

If you'd like two tasks on the **same** node, setting the variable (`BGLMPI_MAPPING=TXYZ`) will accomplish that.

Mapping example

```
cqsub -e BGLMPI_MAPPING=TXYZ -m vn -n 32 ...
```

In the above example, because virtual-node mode is specified, the tasks would be assigned first to the processors within a node, then in the x dimension, then the y dimension, then the z direction.

Task	Location
0	<0,0,0,0>
1	<0,0,0,1>
2	<1,0,0,0>
3	<1,0,0,1>
4	<2,0,0,0>
5	<2,0,0,1>
6	<3,0,0,0>
7	<3,0,0,1>
8	<0,1,0,0>
9	<0,1,0,1>
...	...

Mapfile

You can also use a mapfile that defines the coordinates of the torus to which each process is assigned.

```
cqsub -e BGLMPI_MAPPING=/home/voran/sample.map -n 32 ...
```

The mapfile format is a text file with each line specifying the *x, y, z, t* coordinates of each process (*t* is processor 0 or 1 of each node in virtual node mode.) For example:

```
0 0 0 0
0 0 2 0
0 2 0 0
0 2 2 0
...
```

So with this map MPI process 0 is placed on the node at 0,0,0,0, process 1 is at 0,0,2,0, etc.

Note that the mapfile must define the coordinates for the full partition where your job is running.

Check Placement

You can verify where your processes are being placed, by checking the string returned by the [MPI_Get_processor_name](#). It will look something like this:

Processor <0,0,0,0> in a <4, 4, 2, 2> mesh