

# LSF quick reference

Fabio Viola

## #BSUB directives

Set the name of the job:

```
#BSUB -J test_run
```

Set the number of cores/threads (total across the nodes):

```
#BSUB -n 1
```

Run all the processes on the same host:

```
#BSUB -R "span[hosts=1]"
```

Set the number of processors on each host:

```
#BSUB -R "span[ptiles=4]"
```

Exclusive access to nodes (in case of high RAM requirements):

```
#BSUB -x
```

Set the required memory (in MB):

```
#BSUB -M 16000 directive.
```

Set the walltime (hh:mm):

```
#BSUB -W 00:10
```

Set the file for the standard output (%J replaced by the job ID.)

```
#BSUB -o test_%J.out
```

Set the file for the standard error (%J replaced by the job ID.)

```
#BSUB -e test_%J.err
```

Select the queue to put the job in:

```
#BSUB -q queue_name
```

Specify the project name (for accounting):

```
#BSUB -P projectName
```

Specify the service class:

```
#BSUB -sla serviceClass
```

Impose dependencies (use bjdpinfo to show them):

```
#BSUB -w "ended(JOB.ID)"
```

## Queues

Show the queues available to me/all:

```
$ bqueues
$ bqueues -u all
```

Show the queue of the jobs submitted by me/all:

```
$ bjobs
$ bjobs -u all
```

Queues starting with p\_ are parallel queues, while those starting with s\_ are serial. short, medium and long define the duration of the queue.

## History

Show the history of jobs:

```
$ bhist
```

Show the queue of the jobs submitted by me/all:

```
$ bjobs
$ bjobs -u all
```

## Checkpoints and migrations

Start a checkpointable job with auto checkpoint every 10 min:

```
$ bsub -k "io.chkdir_10" io_job
```

Checkpoint existing job:

```
$ bchkpnt -f -p 15
```

Restart a job:

```
$ brestart io.chkdir JOB_ID
```

Migrate Job:

```
$ bmig JOB_ID
```

## Resources

- [🔗 LSF User Manual](#)
- [🔗 LSF Terminology](#)

## Submitting/Moving/Killing the job

Submit the job:

```
$ bsub < myprogram.sh
```

Move a job to another queue:

```
$ bmod -q queue_name JOB_ID
```

Suspend and resume a job:

```
$ bstop JOB_ID
$ bresume JOB_ID
```

Kill the job:

```
$ bkill JOB_ID
```