Using LLNL’s BlueGene/Q Systems (Sequoia, rzUSEq)

**Useful Batch & Info Commands (from logins)**

- `sbatch/msub scriptfile` - submit batch script job
- `checkjob [-v] job-id` - see detailed job state/info
- `mshow [-a] [[job-id] - partition/job(s) status
- `mdiag -j[-b][f][-P][t][-c] [-v] - batch resource info
- `showstart job-id|procs@secs` - when can job run
- `showstate - state showstats [-q] - resources
- `showbf [-r procs] [-n nodes] [-d secs]` - resource check
- `mpdb` - execute commands on remote nodes
- `scontrol show job` - view SLURM configuration
- `alloc -N nodes -p pdebug` - interactive allocation
- `mxterm nodes 0 mins -q pdebug` - like sxterm

**Interactive Sessions & TotalView Usage**

(Can’t do much from the compute nodes since uses lightweight kernel)

To grab cpus in interactive xterm: `sxterm num_nodes #tasks minutes [-p pool <other_msub_args>]` (or `mxterm`)

Then execute your program in parallel using `srun`: (for available partitions: `smap -c -Db`, use READY & slurm)

- `srun [-N nodes] -n #MPITasks <executable> <args>`
- `srun [-N nodes] -n #MPITasks <executable> <args>`

**Batch Scripts (MOAB)**

- `#!/bin/csh -f` # Sets your shell (`<shell>`, ksh uses export `<var|=><val>`)  
- `#MSUB -V -j oe` # Export env & join out+err; Also: `-h [or -a hhmm]` holds run [until time]
- `#MSUB -q pdebug` # Dont use -l partition=, list by `smap -c -Db`; also -l qos=[standby/expedite]
- `#MSUB -l nodes=# [ppn=]` Nodeblock [...][12k][16k][24k][32k][36k][48k][64k][72k][96k] : tasks/node=16/64]
- `#MSUB -l walltime=3600` # Set duration limit in seconds; 1 hour, or use format 01:00:00 or 00:60:00
- `#MSUB -A bdiv` # Sets bank account to use (or bdev); (mdiag -u userid gives bank info [-a])
- `setenv OMP_NUM_THREADS 4` # For OpenMP threaded runs, set this env var
- `date ; cd /p/lscratchrza/collette # Change into appropriate directory to execute code in & store output
- `srun [-N nodes] -n #MPITasks <exec> <args>`

**Compiling (from front-end)** - Cross-compile code from LAC login nodes (front-end) `[objdump -d file.o - asmlr]

Use IBM’s compilers `mpixlc, mpixlxx, or mpixlf90` (for OpenMP, use their `-r` threaded versions & `-qomp=omp`)

Tuning: `-qhot=novector -qsimd=auto [-qdebug=diagnostic for simd success] [-O3 -qfloat=maf -qstrict=precision]

Useful options: `-qipa, -qhot` - `O[0][2][3][4][5], -g[#,] -qlist[opt], -qreport, -qsource, -qlistfmt=html, -qnostaticlink` Use `-qtm` (-qthreaded) for transactional memory; Use `-qomp=speculative for thread-level speculative execution

For GNU compilers: `mpicc, mpicxx, or mpif90` (or `/gysys/drivers/pcpfloor/gnu-linux/powerpc64-bgg-linux/bin`)

Available libraries: ESSL/BLAS/LAPACK/SCALAPACK/FFTW numerical libs are in `/ust/local/tools/<libname>`

For OpenMP programs with TotalView compiled with IBMs compiler, use options: `-qsmp=omp:noauto:noopt

For batch jobs (use #msub inside a script, or instead these arguments can be given after msub submit command)

- `#MSUB -A bdiv # Sets bank account to use (or bdev); (mdiag -u userid gives bank info [-a])
- `#MSUB -l walltime=3600 # Set duration limit in seconds; 1 hour, or use format 01:00:00 or 00:60:00
- `#MSUB -l nodes= [pool name [i.e. pbatch (or pdebug or others listed by the sinfo command), for partition usage: `smap -c -Db`, use READY & slurm]

Each core can employ 4 hardware threads (but has only 2 floating-point units) which can help with latency issues

All node allocations must be consecutive in the matrix (torus), and only whole partitions of them can be obtained

For batch jobs (use `msub` inside a script, but these arguments can be given after `msub` submit command)

- `#msub -l nodes= num_nodes[:ppn=#]` (MPITasks is set on the srun execution line using its `-n` option)
- `num_nodes` = a full partition [...][16384][18432][24576][27684][32768][36864][41472][49152][55296][65536][73728][98304]

- `procs_per_node# = tasks per node [16|32|64] there are 16 cores, each with up to 4 hardware threads available

For OpenMP programs with TotalView compiled with IBMs compiler, use options: `-qsmp=omp:noauto:noopt

Other options: LLNL_TV_PATCH_SPACE=Add To attach to a running job, use totalview -pid <srunPid> srun

---

Summary sheet by Mike Collette 12/2012
**BGQ Environment Variables**

- **BG_SHAREDMEMSIZE**=[32|64] MB for shared mem pool
- **BG_PERSISTMEMSIZE**=[ ] MB for persistent mem pool
- **PAMID_VERBOSE**=[0|1] For some execution debug info
- **PAMID_CONTEXT_MAX**=[1]
- **BG_COREDUMPDISABLED**=[0|1] allow cores
- **BG_COREDUMPFILEPREFIX**=\textit{str} name of core
- **BG_LTMDISABLE**=[ON|OFF] Livermore thread model
- **BG_MAPCOMMONHEAP**=[0|1] bigger (but shared) task mem
- **BG_SMP_FAST_WAKEUP**=[YES] reduce nonthreaded impact
- **BG_THREADS**=[1|2] round-robin|fill
- **BG_MAPNOALIASES**=[1] disables alias mode
- **OMP_WAIT_POLICY**=[ACTIVE] faster thread performance
- **OMP_STACKSIZE**=[16MB]

**Login Node Info**

- 	exttt{stat} - display file status
- 	exttt{ps} - process status
- 	exttt{pstat} - display process tree
- 	exttt{uname} - machine name and info
- 	exttt{netstat} & 	exttt{vmstat} - net & memory info
- 	exttt{free} - used, swap, and free memory
- 	exttt{pgrep} - find processes by name/attributes
- 	exttt{printenv, env} - see environment variables
- 	exttt{man, apropos, info, whatis} - command info
- 	exttt{cpio - copy to archive}
- 	exttt{nl, wc} - list with line numbers / count lines
- 	exttt{csh} - like ftp but faster
- 	exttt{tar} - archive files
- 	exttt{cpio - copy to archive}

**Debug & Optimization Tools**

- 	exttt{gdb, addr2line, core_stack_merge, locate_rank}
- 	exttt{Tracing: STATview, mpitrace}
- 	exttt{corefile debugging: gdb, addr2line, core_stack_merge, locate_rank}
- 	exttt{TotalView GUI: totalview srun -a \langle core\rangle}
- 	exttt{CLI: tvcli \langle core\rangle} [then ‘dwhere’ traceback cmd]
- 	exttt{profiling compiler options: -p, -pg}
- 	exttt{profiling tools: gprof, cprof, mpiP}
- 	exttt{memory tools: memP}
- 	exttt{hardware performance & others: PAPI (use v5), TAU, valgrind, Open|SpeedShop}

**Architecture**

- 98,304 IBM BlueGene/Q nodes, each with a 16-core 1.6MHz PowerPC A2 64-bit processor
- 1,572,864 total cores with 4 hardware threads & 2-FPUs each; Memory/node:16GB, Total:1.57PB
- Peak 20PFlops; Byte order: Big-Endian (high-order byte is lowest address). CKN liteweight kernel
- 5D-torus interconnect, Lustre&SLURM. Has 4 login front-end/Power7/24core/4GHz/64GB nodes
- 6,291,456 MPItasks

**Help/Info**

- 	exttt{lc-hotline@llnl.gov (925) 422-4531 http[s]://[www|lc].llnl.gov/computing}
- 	exttt{docs /usr/local/docs/[rzuseq.basics] https://lc.llnl.gov/confluence/display/BGQ}

**Summary by Mike Collette 12/2012**