Lecture 7 Summary

This summary is about comparison of the IBM Blue Gene/L (2004 GA), BG/P (2007GA) and BG/Q (2012 GA) supercomputer system. Each of these supercomputer systems is one evolution of Blue Gene architecture. From the begin of this series, it was designed to improve cost/performance for a relatively broad class of applications with good scaling behavior. Since the PowerPoint and the pdf file distributed on the course website only mentions BG/L and BG/Q with details, except BG/P generation. From the overall performance: BG/L, the first generation, has the performance of 0.734 PF/s, or 5.7 PF/rack, or 210 MF/W; BG/P, the second generation, has the performance of 3.5 PF/s, or 13.9 PF/rack, or 357 MF/W; BG/Q, the latest generation, has the performance of 53.6 PF/s, or 209 PF/rack, or 2000 MF/w. As for the first generation, BG/L chose dual-core SOC(system-on-chip) with resource of 0.5/1 GB/Node; BG/P chose quad core SOC with DMA(direct memory access), resource of 2/4 GB/Node and SMP support, as well as OpenMP and MPI; as for the latest generation, BG/Q use 16 core/64 thread SOC with resource of 16 GB/Node, speculative execution, sophisticated L1 prefetch, transactional memory, fast thread handoff and compute + IO systems.

As for software comparison: 1. Overall principle for BG/L is scale infinitely, minimal functionality and closed environment, but for BG/L, it is scale infinitely, more functionality and almost all open software environment; 2. For programming model part, BG/L does not have shared memory, and has hybrid of 2 processes 1 thread and managed by software, no low-level general messaging, and support MPI, ARMCI and global arrays, but BG/Q has shared memory, hybrid of 1-64 processes 64-1 threads, PAMI, generic parallel program runtimes and wake-up unit for low-level general messaging, and more support of OpenMP, UPC, Charm++; 3. For kernel part, BG/L uses proprietary system call interface, library of glibc and threading of proprietary, static linking only, CNK for compute node OS and Linux for I/O node OS, while BG/Q uses Linux/POSIX system calls, same library but pthreads for threading, static or dynamic linking, multiple choice of CNK, Linux, and Red Hat as compute node OS, and SMP Linux with SMT or Red Hat as I/O node OS; 4. For control part, BG/L uses generic API for scheduling and run mode of HPC or prototype HTC, while BG/Q uses generic and real-time API and run mode of integrated HPC, HTC, MPMD and sub-blocks, and HA with job count. 5. For the final part tools, BG/L uses HPC toolkit while BG/Q uses HPC toolkit, Dyninst, Valgrind, and PAPI.