

Bigblue: The new cluster at Simula

Xing Cai

with contributions from Åsmund Ødegaard and Wenjie Wei



November 6, 2008

Outline

- Hardware buildup
- A simple user guide
- Software packages
- Parallel performance

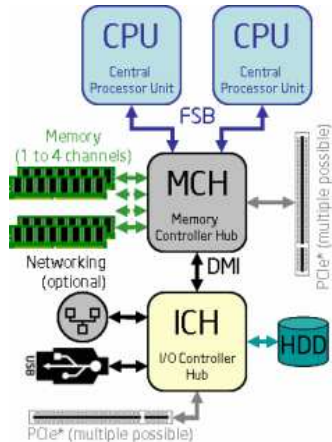
Hardware overview

- IBM multicore-based cluster
- 84 compute nodes, each with 8 cores
- 672 cores in total
- Two separate Gigabit Ethernets as interconnect
- Linux Ubuntu "Hardy Heron" operating system



Compute node specification

- Each compute node has dual Intel quad-core Xeon L5420 / 2.5GHz processors
- 45 nm technology, low power consumption
- L2 cache: 12 MB + 12 MB
- 8 GB shared memory
- 300 GB disk



A simple user guide

- Name of the cluster: `bigblue.simula.no`
- Access by application to `drift@simula.no`
- Serial compilers: `gcc`, `g++`, `f77`, `gfortran`
- Parallel MPI compilers:
 - OpenMPI: `mpicc.openmpi`, `mpicxx.openmpi`,
`mpif77.openmpi`, `mpif90.openmpi`
 - MPICH: `mpicc.mpich`, `mpicxx.mpich`, `mpif77.mpich`,
`mpif90.mpich`
 - MPICH2: under root directory `/simula/arch/mpich2/lib/`
- OpenMP compiler option: `-fopenmp` (for `gcc` and `g++`)

- Must use PBS batch job queue system to run jobs
- Basic PBS commands: qsub, qstat, qdel
- Examples:
 - `qsub -l nodes=16:ppn=8 -l walltime=02:00:00 job_script.sh`
 - `qstat -a`
 - `qdel job_id`
- Typical PBS job script

```
#!/bin/bash
#PBS -j oe
cd $PBS_O_WORKDIR
/usr/local/bin/pmpirun.openmpi ./a.out
```

Useful local commands

- Use `wallcmd` to run a command on all compute nodes
 - example 1: `wallcmd 'ps aux' | grep username`
 - example 2: `wallcmd 'pkill -u username'`
- Use `wallusers` to check users for running processes
- Use `rsh compnode1XX` to log onto a particular compute node
 - very useful when you want to terminate processes there

Software installed (an incomplete list)

- Trilinos:
 - root directory:
`/home/wenjie/local/trilinos-8.0.8/LINUX_OPENMPI`
- Diffpack:
 - root directory: `/simula/arch/NO-bigblue`
 - value of `MACHINE_TYPE` should be chosen as `linux-gcc`
- dolfin
- hypre
- numpy
- py4c
- pycc
- pypar
- ufc
- underworld

Parallel performance (preliminary measurements)

Benchmark: a simple FDM code involving many 2-level for-loops to repeatedly update some plain C arrays

Mesh size=3201 \times 3201

Flat MPI		Mixed MPI & OpenMP	
nodes=1:ppn=8	305.891	1 \times 8 threads	305.375
nodes=2:ppn=8	153.728	2 \times 8 threads	154.829
nodes=4:ppn=8	77.6368	4 \times 8 threads	75.8057
nodes=8:ppn=8	29.5310	8 \times 8 threads	26.9362

wall-clock measurements