Tests and Milestones for SciDAC Software

All Hands Meeting: FNAL Feb. 22, 2003

Richard C.Brower

Three Roles for Tests/Milestones

- Hardware Construction Criteria
  - e.g. Construction Criteria for 10+ Tflops QCDOC
- Verification of Functionality Software Libraries
  - e.g. SciDAC libraries: QMP, QLA, QDP, QIO
- Performance and Optimization of Physics Code
  - e.g Benchmark Level 3 Inverters and HMC MD

Milestones for 1.5 Tflops QCDOC

- **Construction criteria for 10+ Tflops QCDOC**
  1. **Availability and performance of applications codes**
  2. **Availability and reliability of hardware**
  3. **Functionality of operating system**

- **Other Major Milestones**
  4. **Convenient and robust user environment**
  5. **High performance with level three inverters**

- **Achievement of the first three milestones is expected within two months of completion of the 1.5 Tflops machine**

- **Milestones 4. and 5. should be achieved three and six months after completion of the 1.5 Tflops machine, respectively**

Availability and Performance of Application Code

Programs implementing the Hybrid Monte Carlo algorithm to evolve full QCD lattices using staggered, asqtad, Wilson, clover Wilson and domain wall fermions should be available. With the full machine configured as a single partition, evolutions with each of these actions should achieve at least 20% efficiency (0.6 Tflops) with a local volume of 2^4 sites per node (12×16^3 lattice volume) for light quark masses. The ability to run general community software should be demonstrated by showing that standard MILC code using the QMP interface, but without specific optimization for QCDOC, achieves at least 10% efficiency (0.3 Tflops) on an 8^4 local volume (48×64^3 lattice volume). Similarly, MILC code modified to call level-3 inverters should run with at least 20% efficiency (0.6 Tflops) on a 2^4 local volume (12×16^3 lattice volume)

Availability and Reliability of Hardware

The full machine should run for a one week period with no more than one day lost in total for hardware maintenance and debugging on a combination of the application codes described above. (That is, at least an integrated 6 days of physics production running should be achieved. Uptime between the last checkpoint and a machine failure is not counted as production running.) Diagnostic support from the operating system should be sufficient that faults can be diagnosed and repaired without reference to the application code being run. During this week, 50% of the computer time should be spent on reproducibility checks, to verify that there are no undetected numerical errors.

Functionality of the Operating System

Applications running on a node of QCDOC should have a run time environment with support for the standard C/C++ library, standard UNIX I/O routines and QMP. In particular, applications must be able to read and write files on the host computer. The bandwidth to the host computer should be sufficient to permit a double precision $32^3 \times 64$ lattice (1.2 GBytes) to be loaded or unloaded from the host in less than 1 minute (20 Mbytes/sec). The host computer should provide a UNIX-like environment for users to load and run programs.

Convenient and Robust User Environment

A convenient and robust user environment must be provided. This will include the following functionality:

- **A new user can easily register, log on and move code and data to the facility. Procedures are web documented and yield a new account in 3 business days. The ssh and scp protocols are supported and 0.5 Gbyte of RAID disk space per user is routinely provided and backed up weekly. Much more space will be provided as required for particular projects.**

- **SciDAC specified file I/O is implemented allowing files created on a cluster to be used on QCDOC and vice-versa.**

- **MILC and SciDAC QDP (C and C++) codes compile easily.**

- **LQCD SciDAC batch protocols are supported and provide a uniform batch environment across both QCDOC and clusters. The batch queue and status of jobs can be viewed. Batch log files, stderr and stdout files are readily available and there is a defined method (automatic or administrative procedure) to ensure that time allocations work.**

- **An adequate level of personal, user support is provided. This support will be provided uniformly to both cluster and QCDOC users with the actual implementation and delivery shared between BNL, FNAL and JLab.**

Performance with Level Three Inverters

CPS, MILC and QDP++ software, using level-3 inverters should achieve 40% efficiency for full QCD hybrid Monte Carlo evolution on 2^4 local volumes per node using Wilson fermions and with somewhat reduced performance, 30% for staggered (naive or asqtad) fermions or 1.2 and 0.9 Tflops respectively.

Components of SciDAC Software

Tasks	LIBRARIES:
1. Optimize Message Passing	➔ QMP
2. Optimize QCD Linear Algebra	➔ QLA
3. QCD Data Parallel API	➔ QDP
4. I/O, Data Files and Data Grid	➔ QIO
5. Opt. Physics Codes	➔CPS/MILC/LHPC/etc.
6. Execution Environment	➔ unify BNL/FNAL/JLab

QMP: Functionality Tests

- Present Release (C code)
  - QMP_msg.c
  - QMP_loopback.c
  - QMP_qcd_test.c
  - QMP_grid_test.c
  - QMP_MILC_test.c

- Addition Tests (by Celso Mendes)
  - QMP_basics.c - "allocated machine" calls
  - QMP_globals.c - all global operations
  - QMP_barrier.c – barrier calls
  - QMP_strided.c – strided data transfers
  - QMP_iscomplete.c – msg completion test

- Under Development
  - QMP_logicals.c – “logical machine” calls
  - QMP_problem.c – “problem specification” calls
  - QMP_largemach.c – msg transfers in various directions
- QMP Stress Tests are recommended as well.

- Similar Functionality Test Suites for other libraries
  - QLA has a recursive test against C libraries
  - QDP tests not yet developed

QIO, Data Files and Archive

- QIO major topic of Workshop Jan 20 FNAL
- File formats: XML metadata + Binary Data
  - Co-ordinate with UK Data Archive initiative (Richard Kenway)
  - Tools DIME and BINX
- Lattice Archive
  - Replicate data (multi-site), global tree structure.
  - SQL-like data base for storing data and retrieving
- "QIO lite" release in C/ C++ circa May 1,'03.
  - Recognize QDP data type
  - Have name list facility

Software Performance Test for 2003

Perform a uniform set of test on

- QCDOC (asic simulator/prototype)
- Pentium 4 Myrinet/GigE cluster at Jlab
- Pentium 4 duals with Myrinet at FNAL

for 3 Dirac actions

- clover Wilson
- Asqtad staggered
- Domain Wall

with per processor lattice

- 2^4, $2^2\,4^2$, 4^4, $4^2\,8^4$, 8^4, $8^2\,16^2$

for

- Dirac Inverters
- HMC molecular dynamics evolution

to track, test and benchmark software development