SciDAC Software Infrastructure for Lattice Gauge Theory

Richard C. Brower

All Hands Meeting

FNAL, April 6-7, 2005

Code distribution see http://usqcd.org/usqcd-software

Outline

- Software Participants
- Status of SciDAC-1 API
- SciDAC-2 Extensions to API
- Tasks and near term Milestones

Software Committee

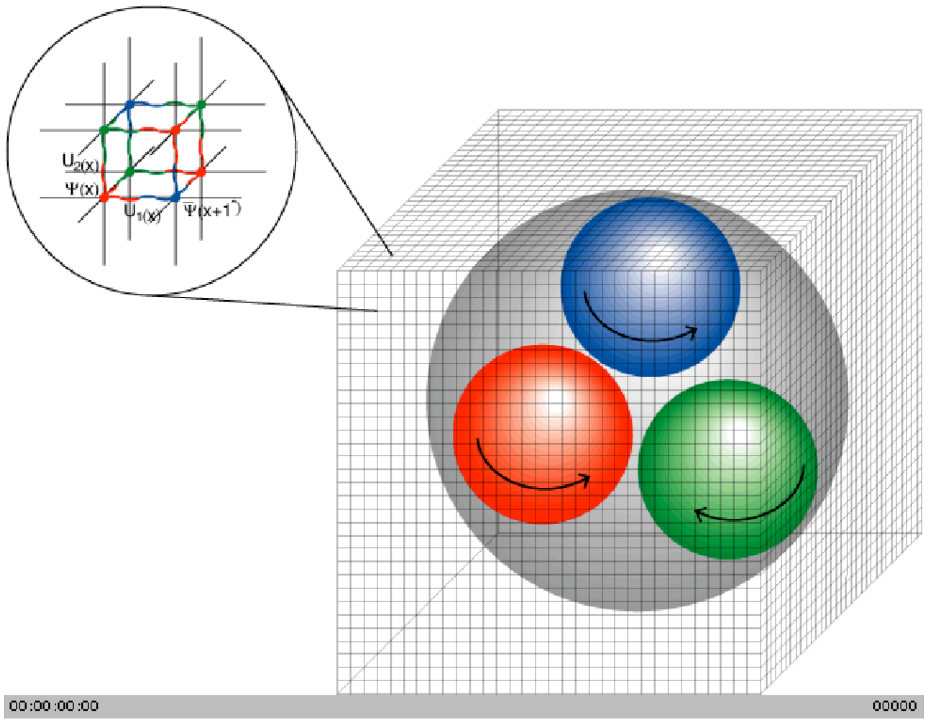
• Rich Brower (chair)	brower@bu.edu
• Carleton DeTar	detar@physics.utah.edu
• Robert Edwards	edwards@jlab.org
• Don Holmgren	djholm@fnal.gov
• Bob Mawhinney	rdm@phys.colmubia.edu
• Chip Watson	watson@jlab.org
• Ying Zhang	zhang@cs.uiuc.edu

Major Participants in SciDAC Project

Arizona	Doug Toussaint	MIT	Andrew Pochinsky
	Dru Renner	North Carolina	Dan Reed
BU	Rich Brower *		Ying Zhang *
	James Osborn	JLab	Chip Watson *
	Mike Clark		Robert Edwards *
BNL	Chulwoo Jung		Jie Chen
	Enno Schloz		Balint Joo
	Efstratios Efstathiadis	IIT	(Xien-He Sun)
Columbia	Bob Mawhinney *	Indiana	Steve Gottlieb
DePaul	(Massimo DiPierro)		(tba)
FNAL	Don Holmgren *	Utah	Carleton DeTar *
	Jim Simone		(Ludmila Levkova)
	Eric Neilsen	Vanderbilt	(Ted Bapty)
	Amitoj Singh	UK	Peter Bolye

* Software Committee: Participants funded in part by SciDAC grant

Lattice QCD – extremely uniform

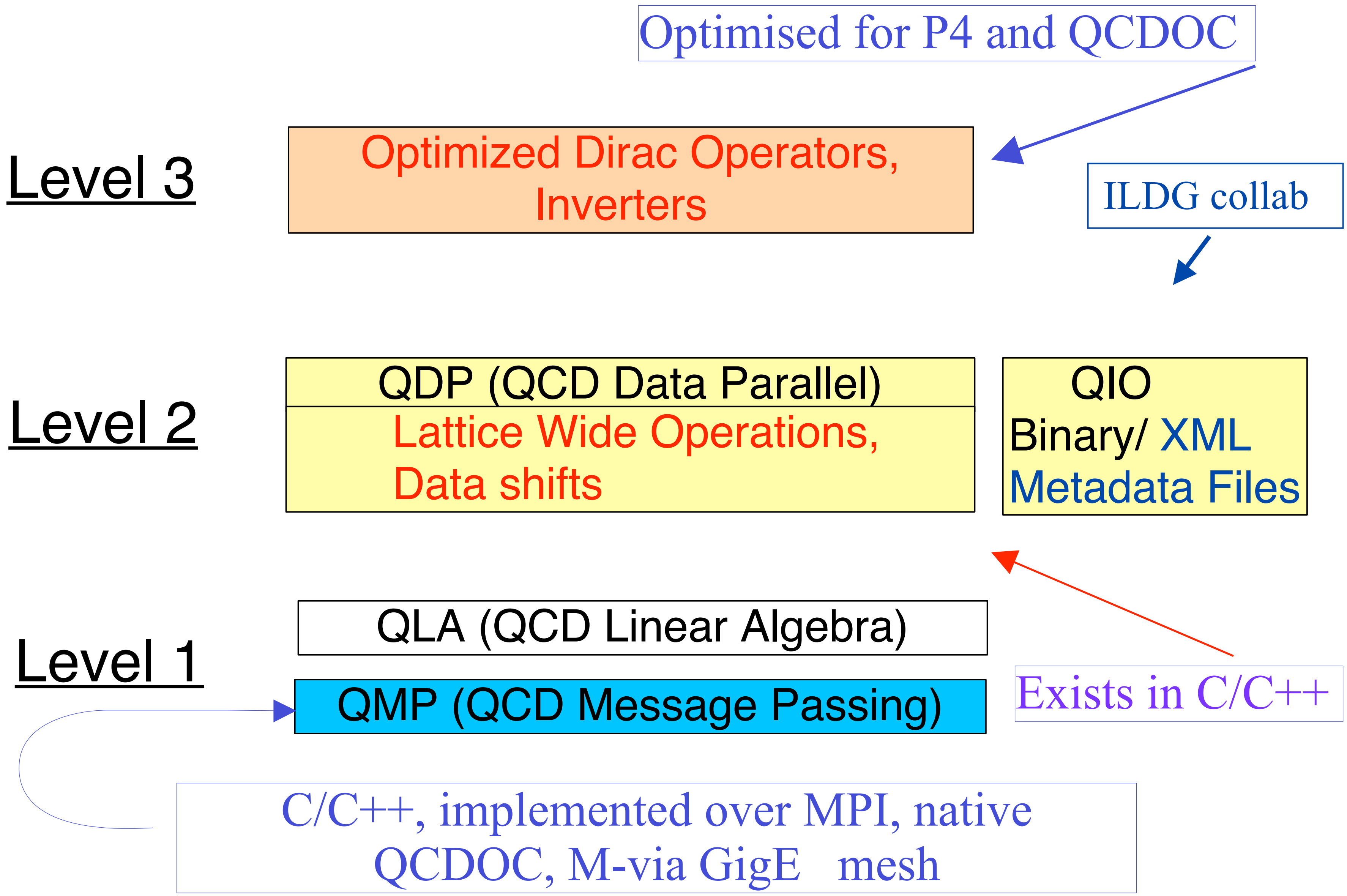


- (Perfect) Load Balancing: *Uniform periodic lattices & identical sublattices per processor.*
- (Complete) Latency hiding: *overlap computation /communications*
- Data Parallel: *operations on Small 3x3 Complex Matrices per link.*
- Critical kernels : *Dirac Solver is ~90% of Flops.*

Lattice Dirac operator:

$$[D\Psi]^i_\alpha(x) = \frac{1}{2a}\sum_\mu [U^{ij}_\mu(x)\gamma^{\alpha\beta}_\mu \Psi^j_\beta(x+\mu) - h.c] + \cdots$$

SciDAC-1 QCD API



Example of QDP++ Interface

Typical for Dirac Operator:

$$\psi^i_\alpha(x) = U^{ij}_\mu(x)\chi^j_\alpha(x+\mu) + 2\phi^i_\alpha(x) \quad \forall\, i, \alpha, x \in \text{even}$$

- **QDP/C++ code:**

```
multi1d<LatticeColorMatrix> u[Nd] ;
LaticeDiracFermion psi, chi, phi ;
int mu;
psi[even] = shift[chi,mu] + 2 * phi ;
```

- Can improved performance, generalize interface for wider use in lattice field theory.

Level 3 on QCDOC

Dirac CG	Local Volume	% of peak
Wilson	2^4	32%
Wilson	4^4	38%
Wilson–Clover	2^4	32%
Wilson–Clover	4^4	47.5%
Domain Wall	2^5	32%
Domain Wall	4^5	42%
Asqtad	2^4	19%
Asqtad	4^4	42%

MILC Asqtad (70% thru put and increasing!)
4 racks 48^3x144 0.09fm 850 lattices at present
2 racks 40^3x96 0.06fm goal 850 lattices on QCDOC
1 rack LHPC DW (Edwards)
1 rack (14 MB used) 468 28^3x96x12 DW prop 140Mflops (Negele)
2 racks Thermo (Petreczky)
2 racks DWF (Christ)

Reality Check: Production can not always at optimal point

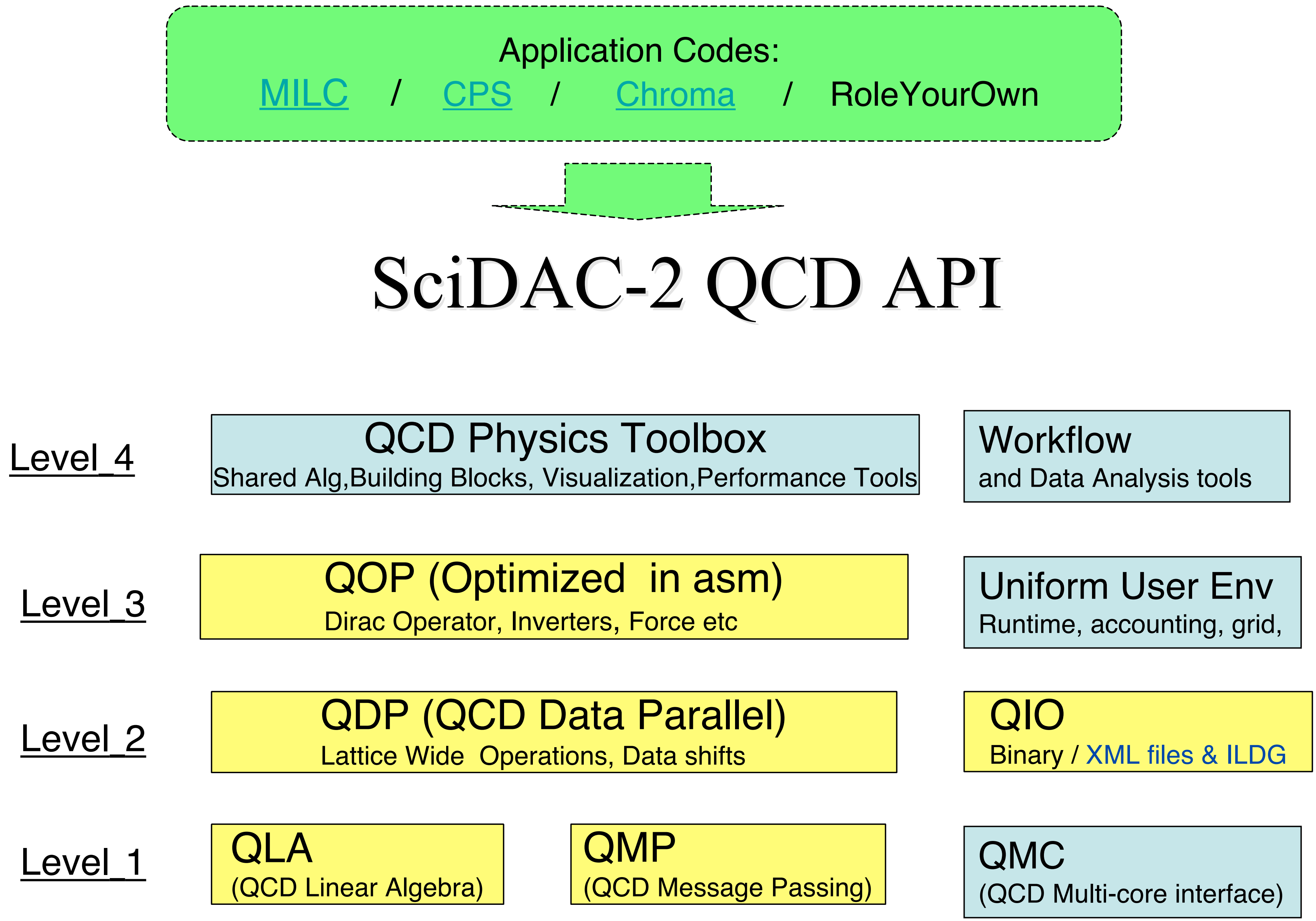
- Peter Bolye's inverter on QCD (invg2 Chroma) as you move from 4^5 to 8^5 performance slows by factor of 0.43
- I/O is stronger on cluster so some of the LHPC structure function work was moved (back) to JLab.

But different machines have different optimal points!

QCDOC/cluster complementarity "principle"

smaller sublattices ➔ faster/slower

larger sublattices ➔ slower/faster



Application Codes:
MILC / CPS / Chroma / RoleYourOwn
SciDAC-2 QCD API
Level_4
QCD Physics Toolbox
Shared Alg,Building Blocks, Visualization,Performance Tools
Workflow
and Data Analysis tools
Level_3
QOP (Optimized in asm)
Dirac Operator, Inverters, Force etc
Uniform User Env
Runtime, accounting, grid,
Level_2
QDP (QCD Data Parallel)
Lattice Wide Operations, Data shifts
QIO
Binary / XML files & ILDG
Level_1
QLA
(QCD Linear Algebra)
QMP
(QCD Message Passing)
QMC
(QCD Multi-core interface)

Major SciDAC-2 Challenges

- ILDG deadline June 2006

  archive/retrieve/query lattice configurations

- Full use of API by entire community

  Documentation, Testing and Distribution

  Timely optimization of (new) Level 3 code

- Exploitation of Multi-core (SMP)

  Multi-core Intel/AMD, BG/L and beyond

  QMC interface for clusters and QCDOC-2

- Common Runtime Env. "Practical Meta-facility"

  File transfer, Batch scripts, Compile targets,

  Libraries, grid access, bug tracking.

- Tool Box (shared algorithms / building blocks)
  - RHMD, eigenvector solvers, etc
  - Visualization and Performance Analysis
  - Rapid prototyping new (multi-scale) algorithms
- Porting API new Machines
  - QMP over Inifinband and BG/L network
  - QLA for Cray XT3 Opteron, 32 bit SSE, etc.
- Workflow and Data Analysis
  - Automate campaign to combine lattices, propagators to extract physical parameters. (ITT)
- Monitor and control large system
  - Automate fault monitoring and mitigation strategies

Tasks and Milestones of Participating Institutions

(Required reading: Appendix A.3 of SciDAC-2 Proposal)

Institutions	*Oversight*
• BNL/Columbia	Mawhinney/Chulwoo Jung
• JLab	Edwards/Watson
• FNAL/ITT/Vanderbuilt	Holmgren/Simone
• BU/MIT	Brower/Pochinsky
• DePaul/NorthCarolina	DiPierro/Zhang
• Arizona/Indiana/Utah	DeTar/Gottlieb/Toussaint

(Mawhinney/Chulwoo Jung)

BNL: BNL will continue to optimize software and implement new algorithms for the QCDOC. It will compile, install and test SciDAC software packages on this machine. BNL will continue the evolution of the Columbia Physics System (CPS) code. It will optimize the CPS for the BlueGene/L, and work on an implementation for the successor to the QCDOC. This work will continue throughout the project, although there will be greater emphasis on QCDOC software during the first three years, and on software for the QCDOC successor in the last two years.

Columbia University: Columbia University will lead an international effort to design and prototype a specialized computer for QCD. During the first year, different design approaches will be studied, and a detailed report prepared describing the results of the study and proposing what is judged to be the best approach. During the second year, this approach will be will be pursued in greater detail, and a proposal will be submitted to the Executive Committee with a specific architecture, cost and schedule for design and construction. If this proposal is accepted, then the design and prototyping work will be pursued in subsequent years.

(Edwards/Watson)

JLab: In each year of this project JLab will carry out research aimed at improving algorithms and producing high performance code for the study of lattice QCD. During the first year of the project, JLab will focus on implementations and optimizations for multi-core processors and for the Intel/SSE3 architecture, and on support for data analysis activities. It will also expand the existing code testing framework, and provide enhanced user support in collaboration with other institutions via workshops, phone and email. JLab will work with FNAL throughout the project to study commodity hardware for lattice QCD. During the first year of the project, JLab will study the Intel dual core "Woodcrest" processor, and double data rate Infiniband fabrics.

(Holmgren/Simone)

FNAL: During the first year of the grant, FNAL will port SciDAC code from the Intel 32 bit to 64 bit environment, and will optimize the code for Opteron processors. During year one, it will also explore the approach for and determine the benefit of a native implementation of QMP over Infiniband, and if warranted, create the implementation. In collaboration with JLab and university researchers, FNAL will provide code to support multi-core processors. With computer scientists at Illinois Institute of Technology it will provide software for automated workflow, and with computer scientists at Vanderbilt it will create software to enhance the reliability of large systems. It will implement and/or deploy software to support the ILDG and other grid activities, and provide software support for the evaluation of new hardware. FNAL will work with JLab throughout the project to study commodity hardware for lattice QCD. During the first year of the grant, it will evaluate AMD Opteron processors and Pathscale Infinipath. Finally, it will assist in the overall project management.

(Holmgren/Simone)

Illinois Institute of Technology: Computer scientists at the Illinois Institute of Technology (IIT) will build a workflow management system for planning, capturing and executing LQCD analysis campaigns. This work will be done in collaboration with FNAL. During the first two years of the grant, a workflow system will be developed and integrated into the existing LQCD computing infrastructure, allowing users to describe their analysis campaign workflow through XML files or graphical interfaces, and submit them for execution. Next a scheduling system capable of interacting with the workflow system and the system performance monitor will be deployed. The final result will be an integrated workflow environment capable of handling multiple campaigns.

Vanderbilt University: Computer scientists at Vanderbilt will develop an automated fault monitoring and mitigation system for the large lattice QCD clusters being built at FNAL and JLab. This work will be done in collaboration with FNAL. During the first year, an integrated monitoring and control system will be designed using existing standards and tools. Also during the first year, a tool will be developed for definition of workflows, monitoring and mitigation actions, based on Vanderbilt's Generic Modeling Environment. This task will be closely coordinated with work at IIT. During the second year, model based generators will be developed to transform the designs into components and configurations for the runtime system. In subsequent years, refined versions of these tools will be developed.

(Brower/Pochinsky)

Boston University: Boston University provides significant leadership for the project as a whole with Richard Brower serving as Software Coordinator and Claudio Rebbi as chair of the Scientific Project Committee. James Osborn of BU has special responsibility to develop the C implementation of QDP and work with collaborators at Arizona, Indiana and Utah to integrate it into the MILC code. He will also work closely with Andrew Pochinsky at MIT to optimize the QCD API for the BlueGene architecture. Brower and Rebbi are leading the physics side of the collaboration with TOPS to study multigrid methods for lattice QCD.

MIT: Andrew Pochinsky of MIT will lead an effort to optimize the QCD API for the BlueGene series of computers. During the first two years, the effort will focus on the BlueGene/L. The gcc compiler will be modified to make efficient use of the two arithmetic units on each processor. The QLA routines will be compiled with this modified compiler, and key routines will be hand optimized as required. A level-3 inverter for domain wall fermions will be written, and in collaboration with James Osborn of Boston University, an optimized version of QMP will be developed. This work will be aided by contract commitments made by IBM as part of the MIT purchase of a BlueGene/L. It is anticipated that in subsequent years these software developments will be extended to later models in the BlueGene line.

(DeTar/Gottlieb/Toussaint)

University of Arizona, Indiana University and University of Utah: The MILC code is an integrated package of some 150,000 lines of scientific application code and a library of generic supporting codes, that is publicly available and widely used. Arizona, Indiana and Utah will work together to carry out a major overhaul of this code to exploit the advantages of the SciDAC software. During the first year of this effort, generic code that supports multiple science-specific applications will be converted to QLA/C to take advantage of its platform-specific optimizations. During the second year, key modules will be rewritten in QDP. Optimization and tuning of the RHMC algorithm, which is currently being incorporated into the code, will be carried out. The first production version of the algorithm will be made available by the end of year one of the grant. Production versions of the code optimized for the Cray XT3 and BlueGene/L will be completed during the first year of the grant, and multi-core and enhanced compiler improvements will be incorporated during the second year. As always, upgrades to the code will be made available to the lattice community as they are completed. Finally, improved documentation for the code will be produced and published on the web by the end of the second year of the grant.

(DiPierro/Zhang)

DePaul University: DePaul University will lead the design and development of a visualization tool for lattice QCD. Work will be done in collaboration with physicists involved in the project and with computer scientists at the University of North Carolina. The goals for the first year of the project are to identify and catalog the types of datasets to be visualized, identify appropriate smoothing and visualization algorithms, and develop a prototype interface. In subsequent years, plugins will be developed to read in the various types of datasets produced in lattice QCD simulations, and tools for manipulating the data in increasingly sophisticated ways will be created.

University of North Carolina: Computer scientists at the University of North Carolina will develop a performance profiling library (PQDP) to analyze the performance of the MILC and Chroma codes during the first year of the project. During the second year, the PQDP will be validated by profiling the MILC code on a variety of HPC platforms, including the QCDOC, clusters, the BlueGene/L and the Cray XT3. In subsequent years the UNC SvPablo performance analysis toolkit will be extended to support analysis of C++ codes so that the PQDP can be used to study Chroma. Performance analysis will be carried out on both codes on a wide variety of HPC platforms, and a web-based performance database will be established. The goal is to optimize the performance of MILC and Chroma based on the collected performance data. Finally, UNC will work with computer scientists at DePaul on the visualization effort discussed above.

More Q&A @ Software Panel tomorrow

moderated by Don Holmgren ("Project Manager Zombie")