Final Report

for the

Lattice QCD ARRA Computing Project

at the

Thomas Jefferson National Accelerator Facility
Newport News, Virginia

For the U.S. Department of Energy
Office of Science
Office of Nuclear Physics

May 2013

Table of Contents

Lattice QCD ARRA Computing Project 1
1 SUMMARY 3
2 PROJECT MANAGEMENT AND BUDGET 4
2.1 Management 4
2.2 Project Budget 4
2.3 Change Control 6
2.4 Project Milestones 8
3 TECHNICAL PERFORMANCE 9
3.1 Technical Requirements 9
3.1.1 Computational Requirements 9
3.1.2 I/O and Data Storage Requirements 9
3.1.3 Network Requirements 9
3.2 Computational System Procurements & Performance 10
3.2.1 GPU Selection 10
3.2.2 Risk Management 10
3.2.3 Phase 1 Systems 11
3.2.1 Phase 2 Systems 12
3.2.2 Benchmarking Adjustments: Effective Performance 14
3.2.3 GPU Expansion and Upgrades 15
3.2.4 Xeon Phi Cluster 15
3.2.5 Operations 16
3.2.6 Price Performance 16

1 SUMMARY

The LQCD ARRA Computing project was funded in 2009 by the American Recovery and Reinvestment Act (ARRA) for a total of $4.965 million to procure and operate significant computational resources for the field of Lattice QCD (Quantum ChromoDynamics) in support of the science mission of the USQCD collaboration, with a goal of expanding the understanding of the fundamental forces of nature and the basic building blocks of matter.

The project, at the Thomas Jefferson National Accelerator Facility (Jefferson Lab), was intentionally complementary to the LQCD-ext Project, a 5 year extension to an earlier Lattice QCD Computing project. In this context it was able to leverage existing expertise at the lab, and to re-use existing advisory bodies so as to enable a quick start to the project.

The original computing performance deployment goal was 16 TFlops as measured by key Lattice QCD kernels, specifically the matrix inversion routines for 3 numerical approaches for solving LQCD: asqtad (a-squared-tadpole), clover, and domain wall. The LQCD ARRA resources would approximately double then existing USQCD resources (17 TFlops at that time).

The project was modified prior to the first procurement in order to include GPU (Graphics Procesing Unit) accelerated nodes in the design. With that change and a subsequent small schedule adjustment the performance goal was increased to 60 Teraflops, and a goal for 180 Teraflops-years of running was fixed. The details of these goals are defined in the Project Execution Plan (PEP) which covers the technical scope, schedule, cost, management organization, and control processes for the project.

In 2009 and 2010, the project deployed significant conventional and GPU resources, carefully setting the balance between conventional x86 clusters and GPU accelerated clusters to match and somewhat anticipate the ability of USQCD software to exploit this new computer architecture. The hardware procurements were build-to-cost, thus for 80% of the project contingency was in deployed performance and not cost.

As a consequence of this careful balancing act, the project was able to deploy a total of 85 TFlops of GPU accelerated nodes while still deploying 10 TFlops of conventional nodes. The GPU performance enabled a significant acceleration in the progress towards long term computational science goals, such as the calculation of the spectra of excited states of nucleons and enabled the project to achieve its integrated performance goal of 180 TFlops-years one year early.

An additional deployment of 10 TFlops of Intel Xeon Phi accelerated nodes provided a cost effective scalable platform to support software development and testing for applications that would be difficult to port to GPUs, bringing the total deployed capacity to 10+85+10 = 105 TFlops across the three architectures.

With all goals met, the project was completed in the Fall of 2012, whereupon all hardware was transferred to the LQCD-ext project for additional running until the hardware's useful end of life.

The project completed in November of 2012 on budget and ahead of schedule, with all performance targets exceeded.

2 PROJECT MANAGEMENT AND BUDGET

This section reports upon the management of the project, including budget, change control, and milestones. The following section discusses the technical performance of the project, describing the procured hardware, its benchmark performance, and operations performance.

2.1 Management

The project was headed by Chip Watson as Contractor Project Manager, and was advised by the USQCD Executive Committee, the USQCD Scientific Program Committee, and a Change Control Board. This structure mirrors the LQCD-ext Computing Project's structure, and re-uses the first two of these committees. The Change Control Board had significant overlap with the larger LQCD-ext CCB.

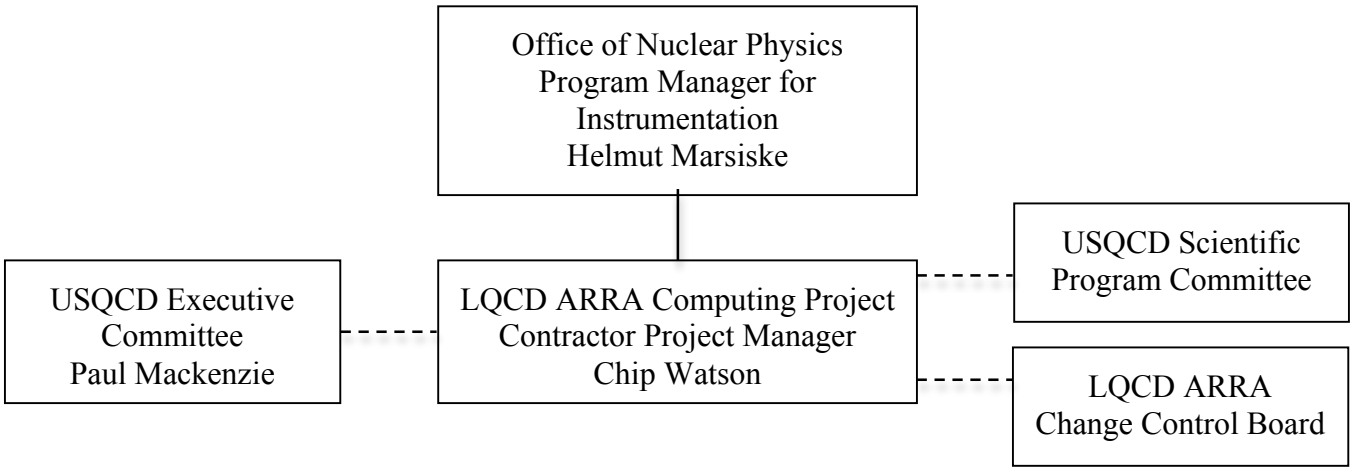


Figure 1. Management Organization Chart for the LQCD Computing Project. Vertical lines indicate reporting relationships. Horizontal lines indicate advisory relationships.

The USQCD Executive Committee consists of collaboration members representing the various scientific areas, and provides overall scientific direction for multiple projects. The Scientific Program Committee is responsible for allocating USQCD resources, including the resources deployed by this project. Both of these committees' input was used to help guide the selection of hardware, particularly with respect to guiding the balance between GPU and conventional resources. Additional details are in the Project Execution Plan.

Monthly conference calls were held jointly among the project manager, the LQCD-ext project management team, the chair of the Executive Committee, and with the DOE program managers, reflecting the close cooperation of the two complementary projects.

2.2 Project Budget

The total project budget for the LQCD ARRA Computing Project was $4.965 million. Equipment costs included system acquisitions (computers, networks), storage (disk), and

power provisioning / conditioning (UPS, circuits). Labor costs include system administration, engineering and technical labor, and project management.

The project was organized using a WBS (Work Breakdown Structure) for purposes of planning, managing and reporting project activities. Management activities included oversight of those working on the project as well as the process of interacting with stakeholders and shaping and optimizing the system architecture. Milestones for the project consisted of procurement related milestones (request for proposal, contract award) and operational milestones (early use of hardware, production running). A second level rollup of the WBS is shown below.

WBS	**Name**	**Total Cost K$**
1.	Project Planning and Management	101
2.	Deployment	
2.01	Site preparations	266
2.02	Phase 1 deployment	1,970
2.03	Phase 2 deployment	1,816
3.	Operations	
3.01	Year 1	127
3.02	Year 2	239
3.03	Year 3	229
3.04	Year 4	217
	Total Project Cost	**4,965**

Table 3 – Cost Summary by WBS including contingency

The hardware procurements were "build to cost" and so no contingency was held for those portions of the project. Because there was considerable experience in procuring and running dedicated clusters for LQCD within the prior projects, contingency was set low on the remaining items (5%); the final year of operations also served as labor burn rate contingency. Ultimately the real contingency in the project was in deployed Teraflops and in integrated Teraflops (operations). As the project evolved, contingency was reallocated based upon remaining risk. The chart below shows a snapshot of costs and remaining contingency for FY2010 Q2.

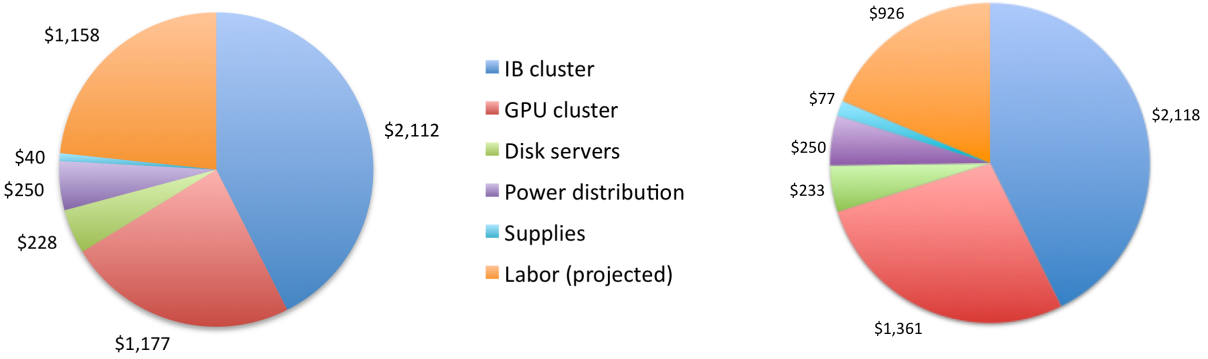
WBS	Item	Baseline Total Cost (AY$)	Costed & Committed (AY$)	Estimate To Complete (AY$)	Estimated Total Cost (AY$)	Baseline Contingency (AY$)	Remaining Contingency (AY$)
1.01	FY09 Mgmt	26	26	0	26	0	0
1.02	FY10 Mgmt	25	15	10	25	1	1
1.03	FY11 Mgmt	15	-	13	13	1	1
1.04	FY12 Mgmt	15	-	14	14	1	1
1.05	FY13 Mgmt	16	-	14	14	1	1
2.01	Site Prep	250	195	50	245	16	6
2.02	Phase 1	1,970	1,967	0	1967	0	0
2.03	Phase 2	1,816	864	952	1816	0	0

3.01	FY10 Operations	121	53	68	121	6	12
3.02	FY11 Operations	228	-	228	228	11	12
3.03	FY12 Operations	218	-	218	218	11	12
3.04	FY13 Operations	207	-	207	207	10	12
	Totals	4907	2185	2727	4912	58	58

This early in the project, no contingency had been consumed, just re-allocated, thereby increasing the % contingency on remaining costs. The move to GPUs greatly increased system performance (by 500%) but also increased the labor costs 15%-20%, and eventually all contingency was consumed as operations labor.

The chart below left gives the budget after incorporating GPUs by category, separating deployment hardware purchases from deployment labor expenses, and rolling up all labor.

When the project completed ahead of schedule, the remaining FY13 operations budget was reallocated to FY12 labor plus two small system upgrades, committed in FY12 (described in the Technical Performance section below). The final cost distribution is shown below right:



2.3 Change Control

The most significant change to the project was the inclusion of GPUs. This particular change did not technically trigger any thresholds in that it did not shift costs from one WBS element to another, nor did it result in any missed or late milestones. Nevertheless, all members of the CCB were consulted, and an internal informal procurement review was organized by the USQCD Executive Committee, Sept 3, 2009 to provide advice on how to move forward with including GPUs into the system design.

In late 2009, one desirable change in the project schedule did have the potential to cause the project to miss the milestone of deploying the second phase clusters into production. The project determined that the best system design would incorporate the anticipated NVIDIA Fermi GPU, but at that time there was some uncertainty in the delivery date for that GPU. To give the project more flexibility, a change in the GPU deployment milestone by 4 months was desirable, but that triggered the threshold for approval by the CCB and DOE project manager (see table, next page). The change was taken to the CCB and the monthly call with DOE, and both approved formally shifting the milestone back to the end of FY2010 (4 months). This change also raised the deployment performance goal from 35 TFlops (the original estimate of what could be procured using originally available GPUs) to 60 TFlops.

Change Control Thresholds:

Level	Cost	Schedule	Technical Scope
DOE Program Manager (Level 0)		> 1-month delay of a Level 1 milestone date	Change of any WBS element that could adversely affect project performance specifications
LQCD ARRA CCB (Level 1)	A cumulative increase of more than $200K in WBS Level 2	> 1-month delay of a Level 1 milestone date	Any deviation from technical deliverables that does not affect expected project performance specifications.
LQCD ARRA Contractor Project Manager (Level 2)	Any increase of > $50K in the WBS Level 2	> 1-month delay of a Level 2 milestone date	Technical design changes that do not impact technical deliverables.

An additional small change was made in the project in FY2012 that involved replacing the 2009 GPU gaming cards (GTX-285) with cards using the newer Fermi GPU (GTX-580). This very cost effective upgrade of around $60K did not exceed the threshold for bringing items to the Change Control Board, but was again informally discussed and was endorsed.

The final change to the project was made when the delivered integrated performance reached the goal of 180 TFlops-years. At the request of DOE to complete all ARRA projects expeditiously, the project worked with DOE and the LQCD-ext project, and transferred ARRA hardware to the LQCD-ext project for ongoing operations. The final 3% of ARRA funds were used to make one final strategic investment in a small cluster of Intel Xeon Phi accelerated nodes (described in the Technical Performance section below). That system was ordered at the end of FY2012 and delivered and commissioned in Oct-Nov. The project ended November 30, 2012.

2.4 Project Milestones

The original Level 1 project milestones are shown in Table 2 on the next page, along with the milestones as modified by the change control process described above.

(Changes are shown in blue.)

Milestones	Date	Modified
Project Start	6/2009	6/2009
Issue Request for Proposal (RFP) for disk and compute clusters	8/2009	8/2009
Place order for disk cluster and first compute cluster	9/2009	9/2009
Place order for Phase 2a Infiniband cluster	11/2009	1/2010
Begin early use on first cluster	12/2009	12/2009
Production running on first cluster	1/2010	1/2010
Place order for Phase 2b next generation GPU cluster	1/2010	4/2010
Begin early use of Phase 2a Infiniband cluster	2/2010	4/2010
Production running on Phase 2a Infiniband cluster	3/2010	5/2010
Begin early use of Phase 2b next generation GPU cluster	4/2010	8/2010
Production running on all resources	5/2010	9/2010
Complete Annual Peer Review of LQCD program (June of each year)	6/2010	6/2010

Table 2 – Level 1 Milestones

3 TECHNICAL PERFORMANCE

3.1 Technical Requirements

The following sub-sections summarize the technical performance requirement. All of these goals were significantly exceeded by the end of the project.

3.1.1 Computational Requirements

The project had as a major technical goal to deploy systems with an aggregate performance on LQCD kernels of 60 Tflops while running an ensemble of jobs with performance up to several hundred GFlops per job. I.e. the system was targeted towards high end capacity computing.

The second major technical goal was to achieve an integrated 180 TFlops-years of running.

3.1.2 I/O and Data Storage Requirements

Disk systems must cache an adequate volume of data for input to and output from the computational systems. The file system must support files 10's of GBytes in size, with an aggregate I/O rate of hundreds of MBytes/sec.

3.1.3 Network Requirements

For multi-node jobs, the cluster inter-node bandwidth must be adequate to provide good scaling to the performance level specified above.

Configuration files will be generated at supercomputing centers, and transferred to Jefferson Lab. This represents a very modest network bandwidth requirement, less than 1 Gbit/s. The larger propagator files will typically be generated and consumed at the same site (Jefferson

Lab in this case) and so do not represent a large wide area network bandwidth requirement., but do require adequate bandwidth to and from storage.

3.2 Computational System Procurements & Performance

At the start of the project, there was not yet any production GPU software in use by the USQCD community. A Lattice QCD group in Germany was already using GPUs, and in the US, two postdoctoral fellows at Boston University (Mike Clark and Ron Babbich) had prototype code running for a basic Wilson inverter. In 2009, dual socket conventional x86 compute nodes could achieve of order 20 GFlops on these key kernels. The prototype code showed performance of order 80-100 GFlops for a single GPU.

As the inverter kernels in the production codes had already been optimized and/or re-written for a number of prior architectures, the approach for integration of new versions was well understood, and it was deemed feasible to exploit GPUs to reach much higher performance per dollar. Nevertheless, in light of the relative immaturity of the software, only a small portion of the budget was initially allocated to GPUs.. In fact, one of the reasons for splitting the procurement into two phases was to allow the software maturity for exploiting GPUs to grow, and allow for an further expansion into GPUs with Phase 2.

3.2.1 GPU Selection

There were two important decisions to take in selecting the GPUs. The first involved software and the development environment. NVIDIA's CUDA language and tools was far more mature than alternatives, and because of the complexity of developing GPU software, a decision was taken to limit the procurement to NVIDIA GPUs and CUDA.

The second choice was between the professional line of GPUs and the GPUs aimed at computer games (gaming cards). The professional cards were built more conservatively, and came with higher guarantees and significantly higher prices. The gaming cards were clocked higher, and so had better performance, but could potentially exhibit random memory errors. A key factor in the final decision had to do with the fault tolerance of the inverter kernels: these kernels were essentially inverting a very large matrix (a high computing cost operation), but the answer could be easily and cheaply checked: matrix times inverse should equal one within some residual which is never zero. Checking that residual (very fast) could confirm that the inverse was correct to the desired accuracy. Because of the fault tolerance of the algorithm, a decision was made to procure mostly gaming cards, specifically the GTX-285. This card was the second version card using the GT200 architecture, and was reported to be of much higher reliability than the predecessor GTX-280.

3.2.2 Risk Management

The project needed to balance computational performance with software availability and risk – would the GPUs deliver on the promise? A GPU workshop was held to explore software maturity and evolution, and to discuss possible configurations for GPU accelerated nodes.

With this workshop providing input, a decision was taken to procure a mix of configurations: (1) conventional non-accelerated cluster nodes, (2) nodes with 2 GPUs, and (3) nodes with 4 GPUs. The host systems for the two GPU systems would be identical to those with 4 GPUs, and thus would allow all systems to be downgraded to 2 GPUs by buying additional empty nodes in Phase 2, or all systems to be upgraded to 4 GPUs by buying additional GPUs if one

solution was later determined to be superior. This approach helped to control risk at the beginning of the GPU deployments.

An additional risk mitigation was to select host systems that had identical CPUs as the non-accelerated nodes. At worst, the GPUs could be ignored. Since the gaming GPUs were so inexpensive compared to the host systems, the financial risk would be small. 196 cards were procured to support 16 dual GPU nodes and 40 quad GPU nodes with some remaining as interactive and spares. The gaming cards represented only 5% of the cost of the Phase 1 systems, and so in fact presented low risk to the project.

3.2.3 Phase 1 Systems

Conventional Cluster

Two RFPs (request for proposals) were issued in early August of 2009, one for a conventional Infiniband cluster, and one for a GPU accelerated cluster.

The conventional system was awarded to Dell for $1.216M for a 320 node cluster plus two interactive nodes and four cold spares. The nodes each included:

Dual quad core Xeon 2.4 GHz 5530 CPUs
24 GB DDR3 memory
Mellanox QDR Infiniband HCA

Per node performance (average of the three inverters) was 20 GFlops, so this system was 6.4 TFlops (consistent with the original performance target for non-accelerated nodes).

Six of the 10 racks of 32 nodes were configured as separate partitions connected to a 36 port QDR Infiniband switch, yielding the best bandwidth per node for jobs of up to 32 nodes (~600 GFlops). The remaining 4 racks were configured as a single partition, with 20 or 24 nodes per leaf switch, 6 leaf switches and 2 core switches in a 2:1 over subscription configuration. This allowed for jobs of up to 128 nodes or 1024 cores, well over 1 TFlops of single job performance with a memory footprint of 2 TBytes (comfortably above requirements).

GPU Cluster

The phase 1 GPU cluster specified hosts capable of 4 GPUs, and the most cost effective solution was a SuperMicro chassis and motherboard which had 4 slots of PCIgen2 x16 and one x4 slot. An award for $384K was made to Koi for a cluster of 58 nodes. All nodes included:

Dual quad core Xeon 2.4 GHz 5530 CPUs
24 GB DDR3 memory for most nodes, 48 GB for 12 of the nodes

40 of the nodes were configured with 4 GTX-285 cards and no Infiniband, and 16 of the nodes were configured with 2 GTX-285 cards and QDR Infiniband. One additional node was connected to an existing external 4 GPU NVIDIA S1070 server, and one was used to hold 4 C1060 GPUs (professional line), giving users 2 systems with professional cards for developments needing to do more than just matrix inversions.

We were fortunate to be able to procure the GTX-285 cards each with 2 GB of memory, a less common configuration found by our vendor after award. The larger memories allowed more applications to run without initially having to move to multi-GPU software.

The quad GPU nodes were upgraded by JLab with existing SDR Infiniband cards (recycled from a 2006 cluster) installed into the x4 slot, effectively giving half of SDR performance, suitable for 2 node running and for file I/O. The QDR cards were installed into a x16 slot.

These two configurations thus allowed 2 types of jobs: (1) small GPU count jobs, 1-4 or 8 GPUs, and (2) larger GPU count jobs, up to 32 GPUs.

GPU software was originally only available for the Clover inverter, heavily used by one research group at Jefferson Lab, and for a thermodynamics code (re-using software from the German group). The Clover inverter performance started at slightly under 100 GFlops, but was eventually tuned up to 130 GFlops per card for a mixed half precision (16 bit)/ single precision inverter. This GPU cluster thus represented a resource of 25 TFlops.

File System

To accompany this significant increase in computational capacity at Jefferson Lab, a greatly expanded file system was required. Many alternatives were considered, and the winning solution was to adopt open source Lustre to aggregate a set of conventional file server nodes into a single system and a single namespace.

A best value procurement for disk storage was won by AMAX for $132K for a total of 14 SuperMicro systems each with 24 1 TB disks. These were configured as 2 sets of RAID-6 8+2, and the remaining disks were used as a mirrored system disk and a mirrored journal disk. The Object Storage Servers (OSS) were connected by DDR Infiniband to the MetaData Server (MDS), and that DDR switch in turn was connected by fiber optic links to the computational clusters. This system provided nearly 200 TB of usable space, and nearly 2 GB/s of bandwidth.

3.2.1 Phase 2 Systems

As already described, the Phase 2 procurement timeline was adjusted so as to allow for the possibility of procuring NVIDIA's Fermi generation of GPU, which was expected at that time to give performance roughly a factor of two higher. The delay also allowed the user community to gain further operational experience with GPUs.

Early running on the Phase 1 GPU systems was heavily single GPU jobs, and host memory to hold 4 jobs was an operational constraint. As multi-GPU software became available, it was clear that quad GPU running would be most cost effective.

The GPUs were critical in speeding up the science programs for the two research groups with software in production at that time, and proved that USQCD could exploit GPUs in a production environment. With that success, a decision was taken to increase the fraction of the funds going into GPU systems to as much as 50%.

Even with that split, the project still chose to allow for a further growth in GPU accelerated systems by specifying that the conventional system servers must be capable of a later upgrade to a single GPU per system. This increased the cost per node by a few percent, but created additional flexibility that was later used.

Conventional Cluster

The procurement of the conventional nodes yielded a system that was very similar to the Phase 1 system (other that being upgradeable to include a GPU). An award was made to Koi

for $858K for 224 nodes (7 racks) in a SuperMicro 2-in-2U configuration. Each chassis held 2 nodes and 2 power supplies (1+1 fault tolerant), and each node was configured with

Dual quad core Xeon 2.53 GHz Westmere 5630 CPUs
24 GB DDR3 memory
Mellanox QDR Infiniband HCA (on motherboard)

All racks were configured as separate partitions, 32 nodes (16 chassis) per rack, with a 36 port QDR switch per rack. Performance was only slightly higher than the Phase 1 Nehalem systems, about 21 GFlops on the ARRA project benchmark, yielding a total of 4.7 TFlops.

This procurement also included extra memory sufficient to upgrade all Phase 1 GPU systems to 48 GB, which helped to improve the utilization of those GPU systems. The award also included 4 additional node pairs to use for system infrastructure (e.g. two were used as interactive GPU development systems, each with one GPU installed).

GPU Cluster

In addition to higher performance, the Fermi GPU had one additional feature of interest to the collaboration: ECC (Error Correcting Code) memory. By the time of the Phase 2 GPU award, the mix of GPU software included some calculations that were doing more than just matrix inversions, and so would benefit from the ECC memory capability of the professional line of GPUs. Consequently a decision was made to procure a mix of gaming cards (as demand for high capacity continued to grow), and Tesla (professional) cards (as the software grew in breadth).

Koi again won the GPU procurement, with a similar Supermicro 4U chassis as in phase 1, but with the Westmere CPUs. The $661K procurement included

32 nodes quad C2050 (Tesla Fermi) GPUs, 16 with QDR Infiniband in x4 slot
20 nodes of quad GTX-480 (Fermi gaming GPU)
130 additional GTX-480 gaming cards

The additional gaming cards were used to upgrade the Phase 1 dual GPU nodes to quad GPU, and to move 32 GPUs into one rack of the Phase 2 Infiniband cluster (one GPU per node).

The QDR cards from Phase 1 were moved into the C2050 nodes to make a cluster of 32 quad C2050 with (half-speed) QDR, allowing scaling to 128 GPUs (with bandwidth constraints). All other nodes were then equipped with the re-cycled SDR cards from 2006 for file system access.

The C2050 cards achieved 176 GFlops in mixed half/single precision, and the GTX-480 cards achieved 273 GFlops in mixed precision. This large disparity was due to the C2050 using more strongly bin selected parts that had only 448 cores and 5 memory I/O systems operational, while the GTX-480 had 480 cores and 6 memory I/O systems operational. Furthermore, the C2050 ECC capabilities came at the expense of memory bandwidth, as part of the bandwidth was carrying ECC bits.

The GTX-480 cards were of high performance, but not as high a quality as the GTX-285 parts with respect to memory errors. Of the 196 GTX-285 cards put into production, only one had failed on memory errors. With the 480s the failure rate was much higher. Nearly 20 were non-functional at delivery and were replaced, and another nearly 20 had memory error rates deemed unacceptable. In the end Jefferson Lab wrote a memory test procedure that ran a

memory test for 2 hours. Any GPU showing more than 10 errors was pulled from operation. With this bin selection on memory errors, all nodes were able to be upgraded to quad GPU, but a planned replacement of some 285s with 480s was abandoned. The memory test is now run weekly, and after 3 years of operation, only 107 of the GTX-480s remain in operation. Even with shorter lifetimes, their high performance yielded an excellent return on investment, and maintenance funds were used to replace failed cards.

It is important to note that the users are for the most part unaware of the memory errors. A fault most of the time probably slows down the convergence of the matrix inverse solver. The algorithm is truly fault tolerant. We could probably have set the rejection threshold a bit higher, but the cost of discarding a $500 card every week or two was deemed appropriate.

In aggregate, then, the Phase 2 GPU system had a pure inverter performance of 192*273 + 128*176 = 75 TFlops for Clover half/single.

File System Expansion

In Phase 2, the Lustre system was expanded by a second award to AMAX for 4 additional nodes, each with 3 sets of RAID-6 8+2, 2 TB disks (44 TB useable per server), a 176 TB upgrade.

3.2.2 Benchmarking Adjustments: Effective Performance

One problem with using just the inverter performance is that it didn't reflect the nodes performance on the non-inverter code. While on a homogeneous system such as the conventional clusters the inverters were a good representation of performance on all code (i.e. everything scaled with the inverter), on accelerated nodes the non-accelerated code continued to run at normal CPU speeds, and Amdahl's Law effects limited the actual acceleration of the job as a whole.

Consequently the project switched to a modified metric for tracking the performance of an accelerated node. The "effective performance" was the performance of a conventional resource that would yield the same job clock time as the GPU system produced. E.g. if 60% of the code is accelerated by 6x, then the clock time is only accelerated by 2x, and so the effective performance is twice that of an un-accelerated node. This de-rating of the GPU performance from 6x to 2x would thus be a more valid measure of the performance that the ARRA project was delivering to its users.

The table on the next page shows the rating of GPUs for the mix of applications running in 2010. Users were asked to compare two identical jobs, one running on a pure CPU system, and one using GPUs, in order to extract a job performance conversion factor (something which is of course job dependent).

The table shows that a quad GPU node varied in performance from 47 to 180 "Jpsi cores" (AMD cores in the Jpsi cluster at Fermilab, the benchmark core for that year), with the project with the largest allocation, Spectrum, getting the highest performance. Cores were then converted into GFlops based upon the measured performance of those cores on the inverter kernels (in GFlops/core).

Project	2010-2011 Hours	#GPUs, nodes	Jpsi core hours /GPU hour (job time)	Effective Performance Gflops/node	GPU used
Spectrum	1,359,000	4, 1	180	800	(average)
thermo	503,000	4, 1	90	400	(average)
disco	459,000	4, 1	92	410	C2050
Tcolor	404,000	4, 1	40	175	GTX285
emc	311,000	4, 1	80	350	(average)
gwu	136,000	32, 32	47	50	GTX285

A usage weighted average was then used to derive the GPU cluster performance. This metric achieved a high of 85 TFlops for most of the Spring of 2011 after some GPU upgrades described below, and ranged from as low as 61 TFlops to 73 TFlops during the first year of Phase 2 GPU running.

This "Effective Performance" was used to measure progress towards the integrated delivered goal of 180 TFlops-years, as it provided a better measure of the performance seen by science applications.

3.2.3 GPU Expansion and Upgrades

In March 2011, a $62K system was purchased that used motherboards that had 8 full x16 slots. These nodes were provisioned with 4 GPUs and dual QDR Infiniband (dual rail), thus giving 4 times as much bandwidth per node as the Phase 2 systems. These nodes were used both for software and performance R&D and for additional capacity running.

In November 2011, $63K was used to purchase 130 GTX-580 GPUs. These were identical to the GTX-480 except in having 512 operational cores instead of 480, and were about 10% faster than the 480s. These were used to replace failed 480s, and to partially replace some of the 285s, yielded a cost effective performance boost. As with the 480s, we had to do some bin selection to remove borderline parts, rejecting about 20% of the total.

With these two expansions, aggregate performance reached 85 TFlops for the GPUs, and 10 TFlops for the conventional systems.

3.2.4 Xeon Phi Cluster

In 2012, Jefferson Lab began investigating the Intel Xeon Phi co-processor cards (compute accelerator cards) as an alternative to GPUs, with a potential for addressing the high software cost of porting code to the GPU. This was not to compete with the GPUs, but rather to enable more of the total USQCD requirements to be met by more cost effective accelerated systems.

When the decision to bring the project operations to a close one year early was made, the remaining funds were then used to procure a small Xeon Phi R&D cluster, both to support software development and performance optimizations, and to support initial test running on this architecture for USQCD.

12 nodes each with 4 Xeon Phi 5110P cards, dual Sandy Bridge 2.0 GHz 8 core CPUs, and 64 GB memory were purchased from Seneca Data for $144K. Preliminary studies indicated that each card would have comparable performance to the latest generation NVIDIA K20 GPU. Subsequent software development and testing bears this out. In a multi-accelerator mode each card should exceed 200 GFlops effective performance, raising the total deployed performance for the LQCD ARRA project to approximately 10+85+10 = 105 TFlops (application mix dependent), well above the late 2009 adjusted target of 60 Tflops.

3.2.5 Operations

Similar to the LQCD-ext project, the ARRA project had a goal of 90% uptime for all computing systems. This goal was met for nearly all months of the project, and was comfortably exceeded when averaged over a year.

The GPU nodes have proven to require more labor per node than a conventional system, but in light of the gains of order 5x in performance, much less labor per TFlops-years than conventional systems.

3.2.6 Price Performance

The GPU nodes have yielded as much as 12x the performance per dollar of contemporary conventional nodes, depending upon job and configuration. This 12x gain was achieved by the excited state spectrum calculations where the workflow was partitioned into a GPU-heavy piece and a conventional piece.

The GPU-heavy piece was purely matrix inversions, and was able to exploit the most cost effective gaming cards. The inversions themselves ran 24x faster per node with 4 GPUs; this was diluted by Amdahl's Law to an 18x job time acceleration. The nodes were 50% more expensive than conventional nodes, yielding a performance per dollar gain of 12x.

Exploiting the GPUs was a disruptive challenge, and the success of the LQCD ARRA project is significantly due to the efforts within the community to make the software investments necessary to match the calculations and workflow to the architecture.