FY19 Joint Acquisition Evaluation Committee Report
for the
Lattice QCD Computing Project Extension II (LQCD-ext II)

Unique LQCD-ext II Project (Investment) Identifier: 019-20-01-21-02-1032-00

Operated at
Brookhaven National Laboratory
Fermi National Accelerator Laboratory

for the
U.S. Department of Energy
Office of Science
Offices of High Energy and Nuclear Physics

Version 0.6

September 12, 2018

PREPARED BY:
The FY19 Acquisition Evaluation Committee

Robert Edwards, James Osborn, Chris Jones, Bob Mawhinney, Gabe Perdue, Amitoj Singh (Chair)

FY19 Joint Acquisition Report Change Log

Version	Description	Date
0.5	Initial draft for review by Committee and Project Manager	9/10/2018
0.6	Edited section 4.6 to correct target goal.	9/12/2018
0.9	Revised draft ready for public consumption	
0.95	Revised draft for review by Committee and Project Manager	
1.0	Final report	

Table of Contents

1. Introduction and Background

The purpose of the FY19 Joint Acquisition Evaluation Committee was to understand user needs, review existing computing resources and make a recommendation to the LQCD-ext II Project Manager and the Fermilab CIO on the design and specifications for a new Institutional Compute cluster at Fermilab (FNAL-IC).

LQCD participation in the FNAL-IC will give USQCD access to a wide range of hardware offerings via the Fermilab HEPCloud science gateway but may impose some constraints on delivered performance. Since the selection of the most cost-effective hardware depends on negotiations between the LQCD project manager and FNAL, the purpose of this evaluation is to determine which option could meet both USQCD and FNAL computing needs and to set forth factors that should be considered in making the hardware selection.

2. Recommendation on hardware acquisition

- FNAL-IC should be Intel "Skylake" based with powers of two core counts per socket, 4GB/core memory and either EDR (mandatory)/HDR (upgrade) Infiniband or Intel OmniPath for inter-node communication.

- Lattice QCD has sufficient GPU capacity with the recent addition of said resources to the portfolio of LQCD hardware available at JLab and BNL. CMS requires GPUs for software development. Machine Learning efforts within the Intensity Frontier experiment community are starved for the availability of quick turnaround GPU cycles. We strongly recommend dual GPUs per host on a fraction of the worker nodes relative to funding. The addition of GPUs should not add a significant overhead cost per node, other than for a chassis with sufficient cooling and power to support GPUs. This will allow the flexibility of installing additional GPUs per worker node when additional funding is available.

- To accommodate the disparate FNAL-IC batch system requirements for both LQCD and non-LQCD workloads, we propose the following:

    - A Submit host which provides LQCD both "interactive" access (ability to get a shell prompt on a worker node) and the ability to submit batch jobs directly to the FNAL-IC. SLURM is a batch queuing system that can satisfy both these requirements. The Submit Host must be well connected to various storage systems that are also directly accessible to batch jobs running on worker nodes.

    - Majority non-LQCD workloads request compute resources via Condor. Each worker node on the FNAL-IC will have outbound network access to the WAN, thus allowing Condor to track and allocate resources.

    - HEPCloud can and has been tested to successfully interface with both the above-mentioned setups.

3. Summary of Committee Activities:

3.1. Gather and review computing needs of the LQCD, CMS, neutrino program user groups.

Below is a summary of requirements gathered from CMS, LQCD and IF experiments. Appendix A contains a template of the requirements document and responses received from each user community. GPU requirements for CMS and the IF are based on Machine Learning (ML) workloads. On the FNAL-IC CMS is targeting running production and analysis workflows. The analysis workflows, which are the most stressful on the I/O system, include the following three steps: data ingest at the rate of 10MB/s/thread, computation at the rate of 1000 events/s and then output which is minimal.

Requirements	CPU			Requirements	GPU		
	CMS	LQCD	Intensity Frontier		CMS	LQCD	Intensity Frontier
Architecture	X86	X86	X86	Architecture	NVIDIA	NVIDIA	NVIDIA
Memory	2GB/thread	4GB/core	4GB/thread	Memory	More is better	More is better	More is better
Interface	Condor	SLURM	Condor	Interface	Condor	SLURM	Condor
High Speed Storage	Experimental	Capacity: 0.5PB Speed: 25GB/s aggregate	Experimental	High Speed Storage	Experimental	Capacity: 0.5PB Speed: 25GB/s aggregate	Experimental
Job Sizes	1-node	1-32 node	1-node	Job Sizes	1-node	1-16 Node	1-node
Intra-node networking	None	None	None	Intra-node networking	NVLINK	NVLINK	NVLINK
Inter-node networking	None	Infiniband or OmniPath	None	Inter-node networking	None	Infiniband or OmniPath	None
Local Disk	20GB/thread	SATA, >1TB	20GB/thread	Local Disk	NA	SATA, >1TB	20GB/thread
IO rates to local disk	10MB/s/thread	NA	10MB/s/thread	IO rates to local disk	NA	NA	10MB/s/thread
Interactive access to nodes	Not required but useful	Yes	Not required but useful	Interactive access to nodes	Yes	Yes	Yes
Software	CVMFS	MPI	CVMFS	Software	CUDA	CUDA	CUDA, CVMFS, Singularity
Archival Storage	Outbound internet access required	1PB-yr	NA	Archival Storage	NA	1PB-yr	NA
WAN access from Worker nodes	Yes	None	Yes	WAN access from Worker nodes	Yes	None	Yes
File System	POSIX	POSIX	POSIX	File System	POSIX	POSIX	POSIX

Table 1: Summary of requirements gathered from CMS, LQCD and Intensity Frontier (IF) experiments. Green indicates matching requirements and red indicates mismatch between requirements across the three user bases.

Cosmic Frontier experiment requirements mostly for cosmologists using analysis programs like CosmoSIS were as follows:

1. Ability to support multi-node MPI jobs, using up to ~500 core simultaneously.
2. Global file system accessible to all compute nodes; parallel filesystem would be preferable.

3. Ability to gain access to at least two nodes interactively, for interactive testing/debugging of MPI programs.
4. The MPI programs in question do not stress the interconnection severely, so extreme low-latency networking isn't required.

Most of the requirements noted above are already being met by the proposed FNAL-IC design.

3.2. Understand the capabilities of the existing hardware portfolio available to LQCD

The following table lists all "allocated" resources available to LQCD across the three sites: Jefferson Lab, Fermilab and Brookhaven.

Nick Name	Nodes	CPU Cores	CPU Type	GPU	GPU Type	KNL cores	Memory (GB)	Fabric (Gbps)	
18p	180		Knights Landing			12,240	16 HBM + 92	OPA 100	JLAB
16p	264		Knights Landing			16,896	16 HBM + 192	OPA 100	
pi0	314	5,024	Ivy Bridge				128	QDR 40	FNAL
pi0g	32		Ivy Bridge	128	K40		128	QDR 40	
BNL-IC	108		Broadwell	432	K80		256	EDR 100	BNL
BNL-IC	54		Broadwell	108	P100		256	EDR 100	
BNL-KNL	142		Knights Landing			9,088	192	OPA 100	
BNL-SKL	64	2,304	Skylake				192	EDR 100	
		7,328		668		38,224			

Table 2: Portfolio of existing LQCD hardware across the three sites.

The portfolio of hardware is rich, and time is allocated through a call for proposals that starts in April, with annual allocations announced in June for a start date of July 1st and end date of June 30th. Time on the dedicated LQCD clusters and institutional clusters is allocated in a standard candle of Jpsi-core-hours. *Jpsi*, a now decommissioned cluster, consisted of 8-core AMD Opterons with DDR (20 Gb/s) speed Infiniband. Cluster ratings are based on appropriate averages of asqtad, DWF fermion, and Clover inverters, representative of a majority of "production" quality LQCD codes.

Following is a summary of available capacity at each site:

- 222 M Jpsi-core-hours on CPU clusters at FNAL and BNL,
- 470 M Jpsi-core-hours on KNL clusters at JLAB and BNL, and
- 7.4 M GPU-hours on GPU clusters at FNAL and BNL

3.3. Assess the vendor landscape for viable architecture options

Conventional CPU architectures are not on the LCF roadmaps, but clearly will continue to be evolved by the vendors. The current Intel Skylake architecture itself is of interest because it supports the AVX-512 instruction set and higher memory bandwidth. At this time only LQCD codes are optimized to exploit the AVX-512 vector extensions. Future Xeon architectures are expected to boost memory bus speed, provide additional memory lanes and support for a tiered memory architecture. Intel is expecting to release the next Skylake successor in early 2019. Benchmarking of said architecture might be available late 2018.

The AMD EPYC CPUs appear to be a worthy competitor to the Intel line, but lack AVX-512 instruction set. Performance on AMD is brittle as processes require processor and memory affinity "guru" level tricks to reduce the side effects of a Non-Uniform Memory Architecture (NUMA). CMS production code provides perfect scaling on AMD with larger core counts per node compared to similar priced Intel processors with lower core counts. But CMS code is not memory bandwidth bound like a majority of LQCD codes. KNL is currently end of life on Intel's roadmap. We do expect to see some feature mixing and convergence between conventional Xeon and Xeon Phi line over time.

The next generation in the NVIDIA Tesla series, the Volta, is in general production. LQCD performance on Volta relative to K20 is 9X for compute and 5X for memory bandwidth.

3.4. Alternative Analysis of viable options

Other options include IBM OpenPower, ARM/ARM64, and FPGAs.

CMS experience on these architectures is as follows:

- <u>IBM OpenPower</u>: incompatibility with ROOT. CMS ran tests on an evaluation "Minsky" system last year, resources available at ANL and Princeton to draw the before mentioned conclusion.
- <u>ARM/ARM64</u>: Nightly builds are already available for ARM but code is not validated yet.
- <u>FPGA:</u> R&D programs in CMS are looking at FPGAs primarily for Data Acquisition.

For LQCD, since the existing portfolio of hardware available to the project offers various viable computing choices, there is no compelling need to consider these alternative options further.

3.5. Recommendation, with technical design and cost estimate for the FNAL-IC

There are five main components of the FNAL-IC machine: core, networking, storage, bridge servers and interface.

- **Core**

Consists of the worker nodes which should be x86_64 based, with powers of two core counts per worker node in order to map input lattices evenly across the four dimensions. As of now Intel Skylake is the only processor that supports the AVX512 vector extensions. A large fraction of production LQCD code now use AVX512 which gives a boost in performance compared to AVX2 or SSE. CMS production code does not use the AVX instruction set.

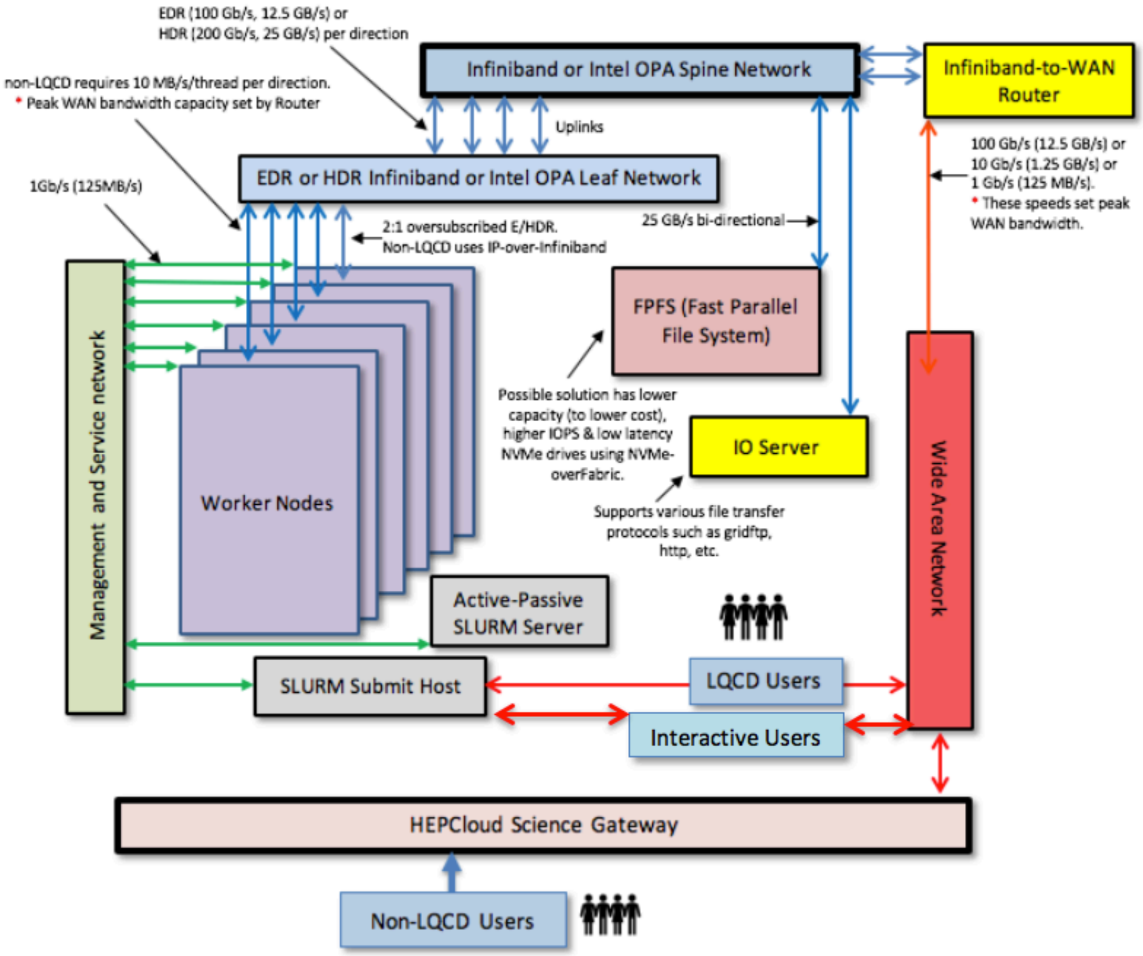


Figure 1: Fermilab Institutional Cluster (FNAL-IC) technical design

	Worker Node Features
CPU	x86, Intel Xeon "Skylake" Gold, powers of 2 core counts
Memory	LQCD & IF: 4GB/core, CMS: 2GB/thread
Local Disk	30GB/core, SATA (standard), SSD (optional)
Software	CVMFS, Singularity, MPI, Intel (license required) or PGI Compilers
GPU	NVIDIA V100, NVLINK between GPUs preferred
Network	1 Gb/s (standard) 10 Gb/s (optional) Ethernet, EDR (standard) HDR (optional) Infiniband or Intel Omni-Path
Management	IPMI

Table 3: FNAL-IC worker node feature set.

Performance gained by using hyperthreading i.e. the ability to run more than one thread per core depends on the workload i.e. CMS, LQCD or IF. Table 4 summarizes the effects of running more than a single thread per socket for the various benchmarks which are representative of "production" codes for each user community.

	AMD EPYC % boost	**Intel SKL % boost**
CMS	+25	+80
LQCD	0	0
Mu2e	-35	-45
ProtoDUNE	+10	+50

Table 4: Effects of Hyper Threading on various benchmark codes

It is apparent that CMS and ProtoDUNE benefits from hyperthreading. But running more than one thread per core will double the CMS WAN IO requirement and mostly probably end up saturating the FNAL-IC WAN network links. On latest Scientific Linux 7 systems, hyperthreading can be enabled or disabled, without requiring a node reboot, via the */sys/devices/system/cpu* interface. This provides the flexibility to allow multiple threads to run per core for codes that benefit from it and restrict a single thread per core for codes that perform poorly under hyperthreading. If a particular machine does not support the before mentioned /sys interface, then hyperthreading can only be configured through the BIOS which requires a node reboot, a disruptive procedure.

- **Networks**

The FNAL-IC comprises of two main private networks: There is a 1 Gb/s (125 MB/s) service and management network. This network is used by SLURM for client server communications and for remote node management using IPMI.

There is an Infiniband based network for inter-node communication for MPI-based jobs. This network should at a minimum be EDR-based (100 Gb/s, 12.5 GB/s) with option to upgrade to HDR (200 Gb/s, 25 GB/s). The highest node count of MPI jobs being run on existing Fermilab LQCD machines are 32 nodes. Most 32-node jobs would fit within nodes connected to the same switch thus providing full bi-sectional bandwidth for such jobs. In such a configuration using 2:1 oversubscription on uplinks from the leaf to spine switches will reduce the number of spine switches and provide a cost savings without hurting performance for multi-node MPI jobs. There is tremendous potential for non-LQCD workflows to benefit from the low latency, high bandwidth Infiniband network using IPoIB. IPoIB (IP-over-InfiniBand) is a protocol that defines how to send IP packets over Infiniband; Linux has a driver that implements this protocol. This driver creates a network interface for each InfiniBand port on the system, which makes an Infiniband or OPA HCA act like an ordinary NIC.

- **Storage**

LQCD jobs require access to a disk storage system that provides read and write capability from hundreds of processes in parallel with an aggregate sustained bandwidth of at least 25 GB/s or better. A Fast Parallel File System (FPFS) meets this requirement.

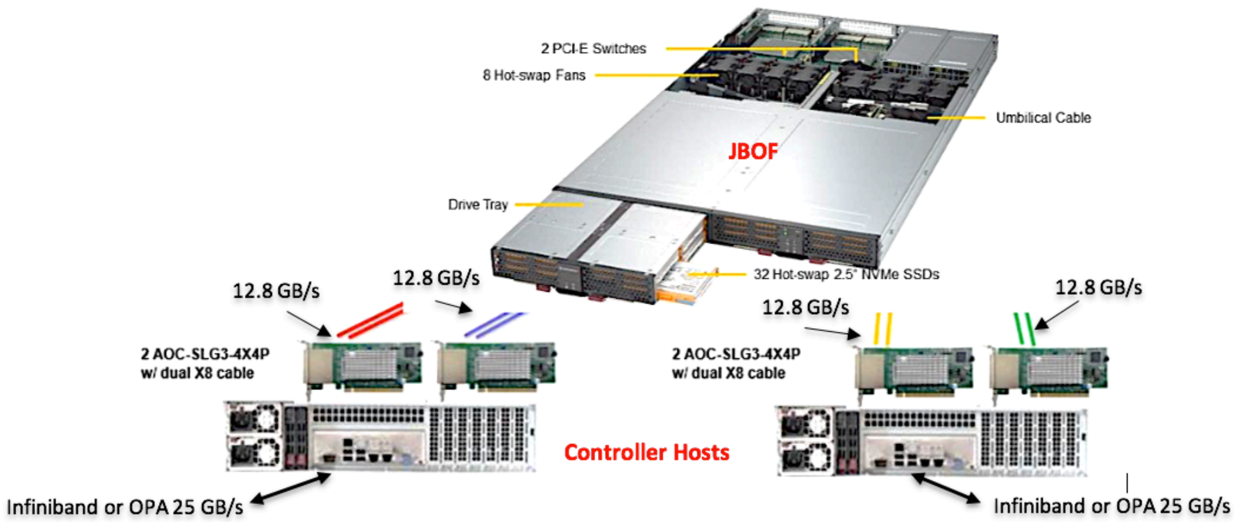


Figure 2: Possible FPFS design

For Machine Learning based workflows there might be a need for a FPFS (Fast Parallel File System). For regular IF production workflows there is no such extreme requirement. Cost savings could be achieved by provisioning less capacity and low latency high throughput storage (such as NVMe based SSDs) and dual rail Infiniband connections per storage server.

A possible scalable FPFS solution is a JBOF (Just a Bunch of Flash) connected to multiple controller hosts using PCIe extension cables (Figure 2). The JBOF, a possible Supermicro solution, is capable of providing up to 1 PB of capacity in a 1U using 128 U.2 based NVMe SSD slots. The largest U.2 NVMe SSD from Intel is 8 TB and 11 TB from Micron. Sometime next year, 16 and 32 TB drives will become available thus doubling capacity, but costs are not available for these at this time. The JBOF with 32 Intel DC P4510 8TB NVMe PCIe 3.0 2.5" SSDs, thus only partially populated, with a total capacity of 256 TB, including a pair of controller hosts would cost about $488/TB. Replacing the flash media with 32 Micron 9200 11TB NVMe PCIe3.0 2.5" SSDs with a total capacity of 352 TB would cost $454/TB.

Regular Hard Disk Drives instead of NVMe based SSDs would cost about $90/TB, far cheaper than SSD based file-system. This would come at the cost of lowered aggregate bandwidth and IOP/s to the disk store.

GPFS, BeeGFS or Lustre on the controller hosts will provide a low latency, high IOPs Fast Parallel File System. GPFS is a closed source commercial product requiring a license. BeeGFS is a closed source commercial product with a free version. Lustre is an open source, community supported product available for use free of charge. The HPC department at Fermilab has several years of experience supporting Lustre on the LQCD dedicated clusters.

- **Bridge Servers**

  Bridge servers consists of an IO server and an Infiniband-to-WAN router server. The IO server supports various file transfer protocols for off-site data transfer. The Infiniband-to-WAN router server provides access to Wide Area Networking for the FNAL-IC worker nodes. The speed of the WAN-connected interface on the Infiniband-to-WAN router sets the peak aggregate WAN bandwidth available to FNAL-IC private network connected nodes.

- **Interface**

  Current LQCD workflows being run at HPC sites are primarily SLURM based. The SLURM Workload Manager (Simple Linux Utility for Resource Management or SLURM), is a free and open-source job scheduler for Linux and Unix-like kernels, used by many of the world's supercomputers and computer clusters. CMS and Intensity Frontier experiments interface with Condor. Condor is an open-source high-throughput computing software framework for coarse-grained distributed parallelization of computationally intensive tasks. HEPCloud can overlay a virtual Condor cluster over any cluster type (for e.g. SLURM) as long as each cluster worker node has outbound WAN access. HEPCloud, extends the current Fermilab computing facility to transparently provide access to disparate resources including commercial and community clouds, grid federations and HPC centers. The outbound WAN access on FNAL-IC will be provided by the Infiniband-to-WAN router.

  The FNAL-IC shall run SLURM as the primary workload manager and provide outbound WAN access via the Infiniband-to-WAN router. HEPCloud will be the user interface to FNAL-IC for non-LQCD subscribers. LQCD will submit jobs directly via the SLURM Submit Host. This will allow all three stakeholders with disparate interface requirements to access the FNAL-IC resources with minimal to no overhead in translating their current workflow submission scripts.

Cost summary of FNAL-IC is as follows. NOTE: It is possible to reduce or increase quantities to fit the total FNAL-IC cost within a set budget envelope.

	Resource	Quantity	Unit	Per Unit Cost	Total
Compute	Skylake-based worker nodes with Infiniband or OPA networking	96	Nodes	$13,000	$1,152,000
	GPU – V100	12	GPU	$11,000	$132,000
Storage	FPFS-Intel SSD	512	TB	$488	$249,856
	FPFS-Micron SSD	704	TB	$454	$319,616
	FPFS-SATA	1000	TB	$90	$90,000
Networking	Management and Service network	128	Ports	$100	$12,800
Storage	IO Server	1	Each	$10,000	$10,000
Networking	Infiniband-to-WAN router	1	Each	$10,000	$10,000
Compute	SLURM servers	2	Each	$5,000	$10,000

FNAL-IC cost w/Intel based FPFS	**$1,576,656**
FNAL-IC cost w/Micron based FPFS	**$1,646,416**
FNAL-IC cost w/SATA based FPFS	**$1,416,800**

Table 5: Cost summary of FNAL-IC.

4. Deliverables:

4.1. USQCD-Specific Software Benchmarks

A set of USQCD-specific software benchmarks have been developed, and performance data has been collected to assist in evaluating the FNAL-IC hardware options. There benchmarks were written by Peter Boyle and other authors (https://github.com/paboyle/Grid) and packaged into a singularity image, for portability, by Jim Simone.

The following codes have been used to provide a somewhat "portfolio" view of the performance. In all cases the most heavily used sparse matrix solver provides the quantitative measure of performance. We take this performance measure only as a rough indication of hardware effectiveness, since some calculations may emphasize other algorithms, or they may depend heavily on system infrastructure, such as I/O bandwidth.

1. DWF using Grid, 32^4 local volume, run on single node. This highly optimized code represents a significant fraction of USQCD computation.
2. MILC code with optimizations, 32^4 local volume, run on single node. Optimized single-node code is representative of a significant fraction of USQCD computation.
3. MILC code with generic C code, 32^4 local volume, run on single node. The code without optimizations is considered to be representative of all non-optimized USQCD code.

The MILC staggered fermion dslash kernel has little computation with lots of data, this is quantified by the arithmetic intensity (or computational intensity) given by flops per byte of data. This means that the staggered-fermion flop rate is limited, not so much by the processor and code, but by, first, the bandwidth to local memory, and second, the network. The measurements that we have made typically show that we saturate both the memory and network bandwidths.

Below are Intel Skylake numbers for the performance of the inverters, comparing one of the Grid multiple-right-hand side solvers with QPhiX on a small lattice with from 1 to 4 nodes.

```
32^3 x 48 strong scaling

BNL Skylake 2 MPI ranks per node, 16 threads per rank
Compiled with Intel MPI
Grid 5D CG with 16 sources
QPhiX single-mass with 16 separate solves
double precision
CG iteration count > 2500
rates in GF/s/node includes remapping, but remapping is negligible
```

Nodes	GFlop/s	
	Grid	QPhiX
1	152	74
2	87	60
4	62	51

Table 6: LQCD inverter performance using Grid and QPhiX.

You can see that on one node, the Grid yields more than double the performance of QPhiX, which has to do a separate solve for each source. The reason this happens is that the multiple rhs computation has a higher arithmetic intensity, with 5D CG the gauge field in dslash needs to be brought into the CPU only once for each of the rhs color vectors, whereas with QPhiX, it is brought in repeatedly for each rhs. The Grid/QPhiX ratio drops closer to 1 as we run on more nodes. This happens because of network limitations. This test ran with 2 MPI ranks per node and 16 threads per rank. Credit goes to Carleton DeTar a USQCD collaborator who ran the above benchmarks.

The next set of benchmarks were run on an AMD system on loan from integrator KOI Computers. The AMD was an EPYC 7601 2-socket 32-core 64-thread 2.2GHz, 2666 MHz memory. The Intel was a Xeon Skylake 8170 2-socket 26-core 52-thread 2.1GHz, 2666 MHz memory, Turbo off and was provided as part of Intel's remote cluster access. On both machines no NUMA tricks were used to provide processor and memory affinity to multi-threaded runs. It has been observed that NUMA side effects are worse on AMD than on Intel. AMD's offering of larger core counts per socket compared to Intel translates to additional NUMA domains, thus increasing the complexity of laying compute and data in the associated processor and memory NUMA regions.

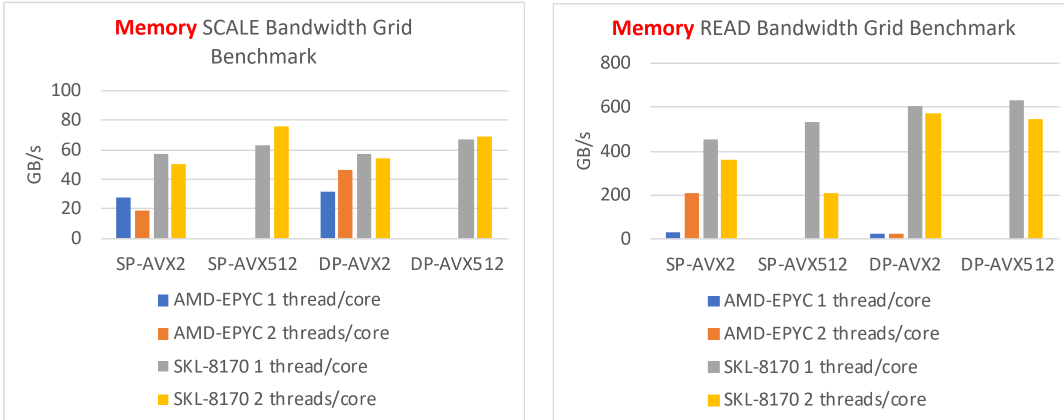


Figure 3: Memory bandwidth LQCD Grid benchmarks. SP = Single Precision, DP = Double Precision.

Memory bandwidth of primary interest to LQCD, is faster on Intel Skylake compared to AMD. Performance drops when running multiple threads per core due to memory contention.

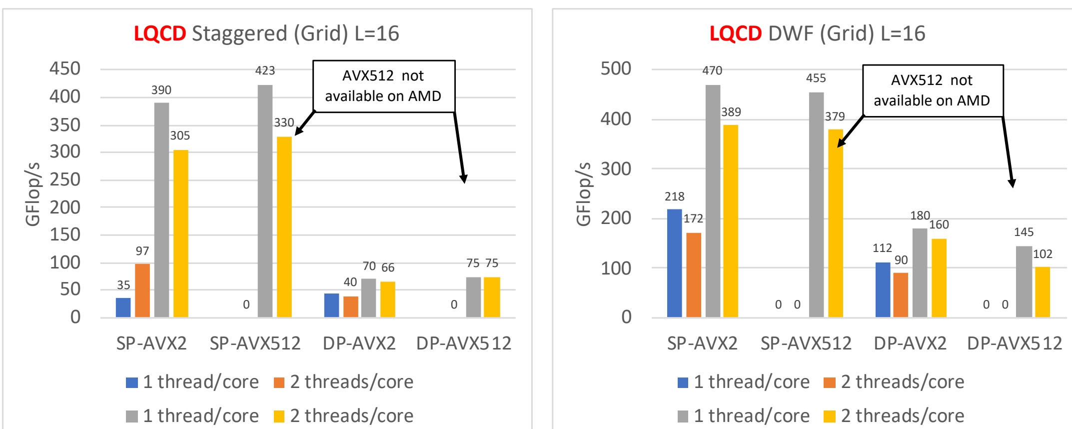


Figure 4: LQCD Grid based Staggered and DWF benchmarks for L=16. SP = Single Precision, DP = Double Precision.

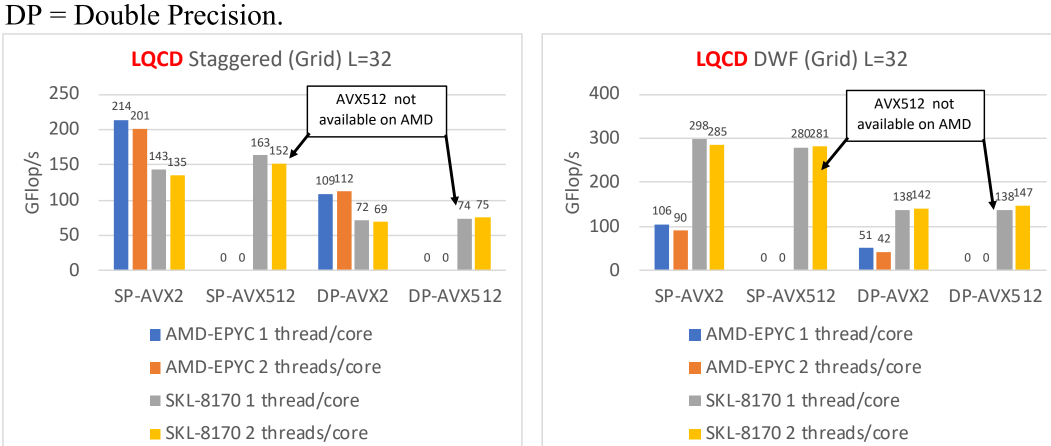


Figure 5: LQCD Grid based Staggered and DWF benchmarks for L=32. SP = Single Precision, DP = Double Precision.

For L=16 LQCD Grid-based staggered benchmark is 3X faster on Intel and LQCD Grid-based Domain Wall Fermion is 2X faster on Intel than AMD. For L=32. LQCD Grid-based staggered benchmark is 1.4X faster on AMD but LQCD Grid-based Domain Wall Fermion is 3X faster on Intel. AVX2 and AVX512 performance numbers are the same as on Intel because the benchmark suite is picking the “best available” vector registers, AVX512 on Intel and AVX2 on AMD. Running multiple threads per core provides zero scaling and hurts performance in some cases.

4.2. FNAL-specific Software Benchmarks

- The CMS TTBar benchmark does the simulation of the collision and the interaction of the collision results with the CMS detector. Performance is recorded in events/second.
- The Mu2e benchmark processes 1500 events per node and performance is recorded in events/second.
- The ProtoDUNE benchmark processes 100 events per thread and performance is recorded in events/second
- MicroBooNE - GPU - Walltime minutes. Training a neural net.

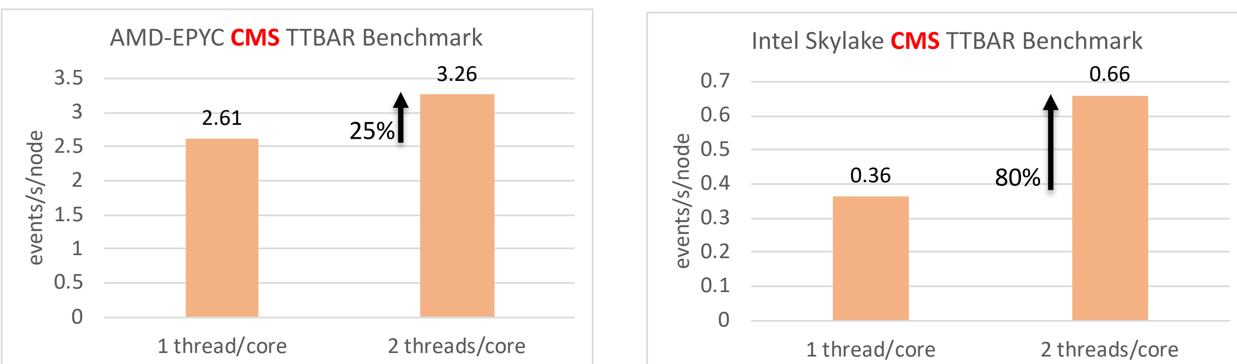


Figure 6: CMS TTBar benchmark on AMD EPYC 7601 2-socket 32-core 64-thread 2.2GHz and Intel Skylake 4116 1-socket 12-core 24-thread 2.1GHz.

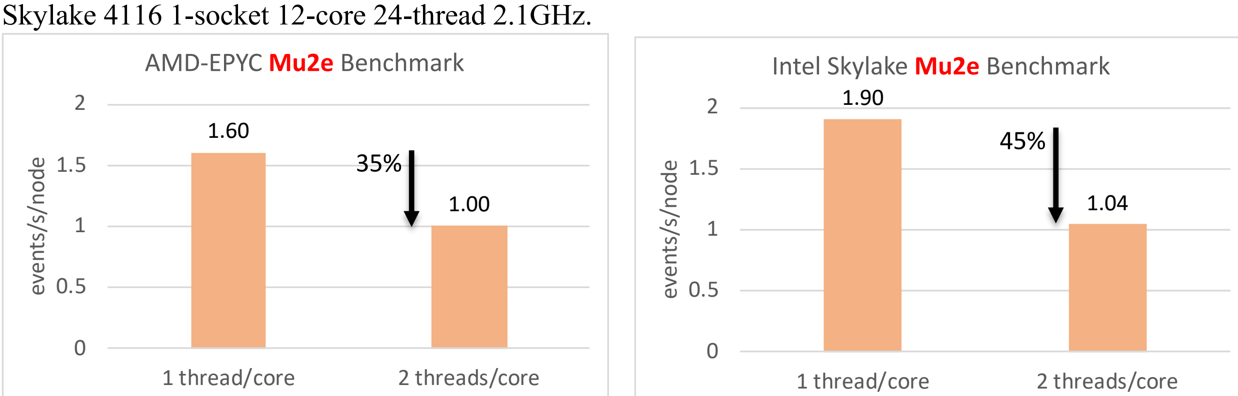
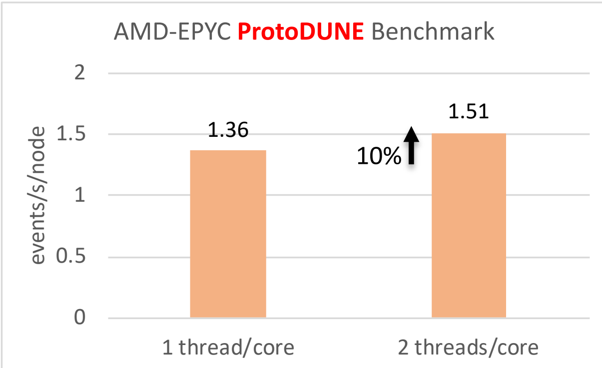


Figure 7: Mu2e benchmark on AMD EPYC 7601 2-socket 32-core 64-thread 2.2GHz and Intel Skylake 8170 2-socket 26-core 52-thread 2.1GHz.



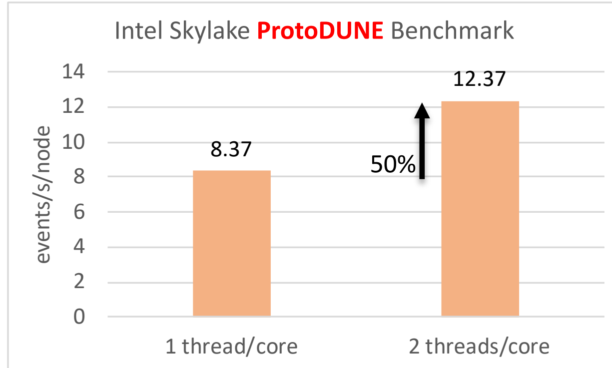


Figure 8: ProtoDUNE benchmark on AMD EPYC 7601 2-socket 32-core 64-thread 2.2GHz and Intel Skylake 8170 2-socket 26-core 52-thread 2.1GHz.

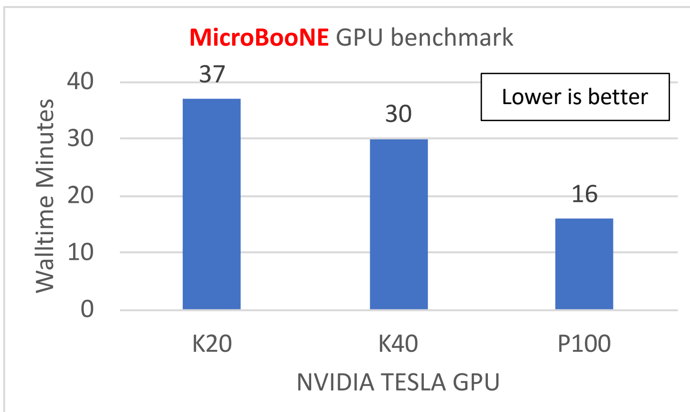


Figure 9: MicroBooNE GPU benchmark on host with Intel Xeon processors and NVIDIA Tesla class GPUs.

MicroBooNE GPU benchmark scales perfectly between GPU generations, i.e. Kepler vs Pascal. We could not get access to a Volta-based machine in a timely manner to run the MicroBooNE GPU benchmark, but on paper Volta can deliver 5X the performance of a Pascal and is a formidable option for GPU-based hosts.

4.3. Near- and long-term demand for each hardware architecture

Near-term Demand for LQCD: Aida El-Khadra, of the USQCD Software Program Committee, states that in the 2018 allocation year proposals, the architecture demand was as follows:

- GPUs are over-requested by a factor of 1.8
- CPUs are over-requested by a factor of 1.4
- KNLs are over-requested by a factor of 1.2

It is worth noting here that Intel has initiated a product discontinuance plan for its Xeon Phi 7200-series processors codenamed Knights Landing (KNL). Given that the existing portfolio of LQCD machines contains a significant fraction of KNL based hardware (table 2) the SPC will continue to allocate this resource as long as it is supported by the OEMs and system integrators warranty and service plans. At the same time, Intel will keep offering its codenamed Knights Mill (KNM) solutions for Deep Learning.

ML workflows require a substantial increase in GPU availability at Fermilab in the near term. There is a dearth of resources at scales in between very small clusters and LCFs. In order to properly leverage LCF resources, they require intermediate scale facilities that do not require codes to utilize 1,000s of GPU nodes with maximum efficiency to properly develop this capability. Additionally, the LCF model disfavors long runs on small numbers of nodes, which is a more typical ML workflow than massive numbers of nodes for short periods. To develop and run multi-node code, ML programmers additionally need high-speed interconnect between nodes and the ability to flexibly provision different numbers of nodes.

Long-term Demand for LQCD: In the long-term we see:

- Continued demand for CPU technologies,
- A significant fraction of conventional CPU demand metamorphosing into GPU demand, when higher performance is available,
- Continued demand for GPU technology, and
- Technologies that appear likely to be the most cost-effective for both LQCD and Fermilab going forward are:
    - CPU (example today is Intel Skylake)
    - GPU (example today is NVIDIA Volta)

Near-term demand for CMS:

- 25k jobs simultaneously around the world
- Analysis only jobs will most probably run on FNAL-IC.
- GPUs for similar workloads as IF experiments but currently only for software development.

Long-term demand for CMS:

- GeantV which exploits the AVX vector extensions.
- LHC-High Luminosity which is a 100x compared to current requirements.

Demand for Machine Learning at Fermilab: Given the explosion of interest in Machine Learning, long-term demand for GPU resources looks to be very robust. GPU life cycles are long enough to merit investment (see, for example, how long Titan (OLCF) has successfully used K20 GPUs). Additionally, there is significant value in having a stable, accessible platform for optimizing code and software architecture.

4.4. Alternate computing architectures

See section 3.4

4.5. Availability of production software to utilize capabilities of proposed IC machine

A large fraction of USQCD computing requires high bandwidth, low latency networking, such as Infiniband or Omni-Path. Some applications require high I/O bandwidth, and some high memory capacity. Thus far all machines designed by the project include networking and memory suitable to lattice QCD calculations.

CPUs: all USQCD software can run on CPUs, provided it is not optimized to some other architecture, though it may not necessarily run at peak performance. All CMS software can run on CPUs, it is not optimized for AVX extensions but with use of GeantV in production that may change.

KNL: most USQCD software can run on KNL, provided it is not optimized to some other architecture. This may require running in a backward-compatibility memory mode, but that can be arranged with the local site administrators for specific jobs.

GPU: some USQCD software can run on GPUs provided it is optimized for that architecture. Large-memory algorithms however are not as effective on GPUs due to limited total memory per host in the packaging that we could afford. Intensity Frontier experiments use GPUs primarily for Machine Learning workloads such as training neural nets.

4.6. Ability to meet time-based performance goals for the LQCD project

	Target Goals (DWF + HISQ averages used). Integrated performance figures use an 8000-hr year.				
	FY15	**FY16**	**FY17**	**FY18**	**FY19**
Planned computing capacity of new deployments (Tflop/s)	0	49	66	134	172
Planned delivered performance (TFlop/s-yr)	180	135	165	230	370

Table 7. Performance of New System Deployments, and Integrated Performance (DWF+HISQ averages used). Integrated performance figures use an 8000-hour year. The capacity and delivered performance figures shown in each year sum the conventional (TFlop/s and TFlop/s-yr) and accelerated (effective TFlop/s and effective TFlop/s-yr) resources deployed and operated. All deployment figures assume that 50% of the annual hardware budget is used to purchase accelerated hardware, and 50% to purchase conventional hardware.

Because more USQCD code already runs on CPU systems, predicting portfolio performance on this architecture is easy. Planned acquisition for FY19 is for a cluster rated at 172 TFlop/s (table 7). The average of Single Precision AVX512 based DWF and Staggered performance is about 439 GFlop/s (Figure 4) and on NVIDIA V100 GPU is 1.413 TFlop/s. With 96 Skylake nodes and 12 V100 GPUs (see price estimate table 5) we would deploy 59 TFlop/s or 34% of planned computing capacity. It is worth mentioning here that the LQCD project planned deployment for FY19 assumed a 50% conventional and 50% accelerated hardware. For accelerated clusters, the figure is based on the USQCD rating of an NVIDIA K20 model GPU rated at 157 effective GFlop/s. The next generation in the NVIDIA Tesla series, the Volta, relative to K20 is 9X faster for compute and 5X for memory bandwidth. Thus, a single Volta GPU would be rated at 1,413 GFlop/s. A 50-50 conventional-accelerated deployment would then surpass the 172 TFlop/s target goal for FY19.

Acquisition of Intel Skylake CPUs can meet time-based performance goals better than AMD, subject to available funding. Both the LQCD project and Fermilab have experience successfully operating Skylake based clusters which have delivered solid performance and good reliability.

Performance depends on the degree to which users are able to optimize their code, as with the Grid and QPhiX software strategies. Network bottlenecks have limited the performance of even these highly optimized software strategies for multi-node applications. Although Intel is working on solutions, a prudent cost-performance calculation should be based on present-day performance.

GPUs can meet the time-based performance goals, subject to available funding and provided enough of the software portfolio can run on GPUs at scale.

4.7. Alignment of IC configuration with vendor technology roadmaps and with LCFs

See section 3.3 for alignment with vendor technology roadmaps. The following is a list of available ASCR facilities:

1. Argonne Leadership Computing Facilities (ALCF)
    a. Aurora (Online in 2021)
    b. Mira (10-petaflops IBM Blue Gene/Q)
    c. Theta (11.69 petaflops system based on the second-generation Intel® Xeon Phi™ processor)
2. Oak Ridge Leadership Computing Facilities (OLCF)
    a. Summit – (IBM Power9, NVIDIA Volta)
    b. Titan – (Cray XK7, NVIDIA K20)
3. National Energy Research Scientific Computing (NERSC) center at Berkeley Lab.
    a. Cori – (Cray Intel Haswell and KNL)
    b. Edison – (Cray Intel Ivy Bridge)

4.8. Recommendation to the Fermilab CIO and LQCD Project Manager on how best to proceed with hardware acquisition

See section 2.

4.9. Bi-weekly status reports.
http://www.usqcd.org/fnal/acquisition

5. Suggestion for Future Acquisition Evaluations

There was a need for more lead time in order to package benchmarks that are true representatives of "production" running and gather requirements. This especially impacted Intensity Frontier and having additional lead time could have allowed better benchmarks. Also given the short deadline, several committee members had scheduling conflicts and thus less time to actively participate in the committee activities.

Thus, the suggestion for future is to provide sufficient lead time to fulfil committee activities and a time frame where majority of committee members are available or have minimum schedule conflicts.

Appendix A: Template of Requirements for the Joint Fermilab & LQCD Institutional Cluster

Author:
Experiment / Project:

1. Introduction
    a. Current science drivers for your field of research.
    b. Science challenges expected to be solved in the next 5 years' time frame using extant computing ecosystems.
    c. Scope of the software.
    d. Definitions, Acronyms and Abbreviations.
    e. References.
2. Specific Requirements
    a. Functional.
    b. Performance.
    c. Interface.
    d. Operational.
    e. Resource.
    f. Software.
    g. Verification.
    h. Acceptance Testing (Validation Requirements).
    i. Documentation.
    j. Security.
    k. Portability.
    l. Quality.
    m. Reliability.
    n. Maintainability.
    o. Safety.
3. General Constraints
4. Compute resources being currently used
    a. Resource list (e.g. LCFs, Fermigrid, OSG, dedicated clusters, etc.)
    b. Limitations or issues experienced on said resources.
5. What top three computing ecosystem aspects will accelerate or impede your progress in the next 5 years? Why? (aspects listed below are examples, feel free to edit as needed)

Accelerate	Why?
1. Hardware resources	

2. Software frameworks	
3. Intra and inter node communication fabric	

Impede	Why?
1. Process for allocation of computational resources	
2. Application optimization/ development support	
3. Persistent archival data storage	

Requirements for the Joint Fermilab & LQCD Institutional Cluster
Author: Chris Jones
Experiment / Project: CMS

1. Introduction
2. Specific Requirements
    a. Functional.
        i. Worker nodes need access to local or network-based storage: 20 GB/core.
        ii. No requirements on networking between the nodes.
        iii. Each host needs to be able to get to the internet or indirectly through a proxy. (CVMFS provides code, Squids provide conditions [inputs for job, calibration]), Condor may want full access over internet, but CMS jobs have workaround if not connected to the internet.
    b. Performance.
        i. The local or network-based storage needs to be scalable to handle order of 0.1 - 10MB/s read/writes per Core.
    c. Interface.
        i. Be able to schedule CMS jobs into the nodes, this requires an interface compatible with Condor.
    d. Operational.
        i. Be able to log into some (10 either CPU or GPU very important for GPU-based hosts) of the nodes to allow interactive access (for debugging, testing, and performance analysis)
    e. Resource.
        i. 2GB/core (when using multiple threads, CMS jobs can work with less memory)
        ii. x86-64 compliant architecture.
3. General Constraints
    a. Condor
    b. 20GB/core storage
4. Compute resources being currently used
    a. Resource list (e.g. LCFs, Fermigrid, OSG, dedicated clusters, etc.)
        i. WLCG
        ii. Tier-1 and Tier-2 compute
        iii. Just starting to use Stampede
        iv. Cori at NERSC
    b. Limitations or issues experienced on said resources.
        i. CVMFS access an issue (access to code)
        ii. Allocation for computation resources is an issue as CMS requires steady state resource allocation.

iii. Other LCFs not used because of lack of internet access, not x86 based, not used to having HTC jobs and not built for streaming inputs.

Requirements for the Joint Fermilab & LQCD Institutional Cluster
Author: Kazuhiro Terao
Experiment / Project: LAr-ML, ArgonCUBE (Pixel LArTPC R&D)

1. Introduction
    a. Current science drivers for your field of research.

       Sterile neutrino (Short Baseline Neutrino program, SBN), Neutrino mass hierarchy and CP violation (Deep Underground Neutrino Experiment, DUNE)

    b. Science challenges expected to be solved in the next 5 years' time frame using extant computing ecosystems.

       High quality data reconstruction of neutrino events in liquid argon time projection chamber (LArTPC) is necessarily for successful physics measurements in SBN and for reaching proposed goals for DUNE experiments. In next 5 years we address this challenge for SBN using machine learning based data reconstruction.

    c. Scope of the software.

       We use open-source ML software for data reconstruction and analysis. Candidates include Tensorflow and Pytorch with many common scientific python modules such as scipy, numpy, pytables, and h5py. We use Geant4 for particle simulations and our custom C++/Python code for running simulation.

    d. Definitions, Acronyms and Abbreviations.

       LArTPC = Liquid argon time projection chamber SBN = Short Baseline Neutrino program
       DUNE = Deep Underground Neutrino Experiment ND = near detector
       FD = far detector
       DNN = deep neural network
       CV = computer vision
       ML = machine learning
       DLP = DeepLearnPhysics (organization, deeplearnphysics.org)

2. Specific Requirements

    I have very poor ideas on what kind of functional, interface, performance, etc. to expect in HPC. So, I leave most of items below empty but if someone could ping me I can fill with someone's help.

    a. Software.

       Singularity is all required, or alternative software/data distribution mechanism.

    b. Documentation.

       To be delivered as a publication (later summer 2018).

    c. Portability.

       Singularity container to distribute our software to workers, or alternative distribution mechanism.

3. General Constraints

4. Compute resources being currently used
    a. Resource list (e.g. LCFs, Fermigrid, OSG, dedicated clusters, etc.)
    Local GPU cluster @ SLAC equipped with NVIDIA GPUs including x6 GV-100 (32GB) and x10 GTX 1080Ti (11GB).
    b. Limitations or issues experienced on said resources.
    None on VG-100, memory is lacking to fit the whole data reconstruction algorithm (DNN) for 1080Ti.

5. What top three computing ecosystem aspects will accelerate or impede your progress in the next 5 years? Why? (aspects listed below are examples, feel free to edit as needed)

Accelerate	Why?
1. GPU hardware	One choice to accelerate DNN efficiently
2. ML Software frameworks	Implement code to benefit from GPU acceleration
3. GPU-to-GPU bandwidth (NVLink, etc.)	Helps for DNN algorithm parallelism on multi-GPU platform

Impede	Why?
1. Process for allocation of computational resources	Less familiar with how this process works, how much overhead time to expect, etc.
2. Application optimization/ development support	Our application heavily depends on GPUs, and we do not possess expertise to best benefit from specific hardware configurations at HPC sites.
3. Data transfer and storage	Getting data in and out to HPC sites may be proven to be problematic for LArTPC where data tends to be huge (yet sparse).

Requirements for the Joint Fermilab & LQCD Institutional Cluster
Authors: Robert Edwards, James Osborn, Bob Mawhinney
Experiment / Project: LQCD

Types of workflows expected to run on the FNAL-IC

- Similar to BNL-IC. Compute nodes do not need to talk to outside world.
- I/O requirements are the same as CMS. Single instance jobs Double precision, complex, multiple cores, batch across multiple cores.
- 3 basic parts of overall workflow: generating the **gauge generators** (LCFs), **propagators** (16-32 GPUs solving large sparse matrix equations, suited for V100, not cost effective though, gaming cards for V100 not yet available, gamer P100 available but no-ECC is an issue, LQCD code already have ECC built into the code but others may see memory errors), **contractions** (input file, list of correlation functions, graph evaluation for tensor contractions, single node jobs). Contractions consume about 300GB not over a long period of time, are serialized, read compute repeat.
- Gauge generators and propagators mix them together and generate large number of configurations, ensemble jobs with lots of different parameter sets, BNL using Globus Online using Panda for getting input data. Whole QCD (gauge gen at 25%, propagators 75%, contractions fixed cost can keep coming back using different inputs, 15-20%)

Memory requirement

- 100GB/node – lesser core counts will lead to smaller dataset, there is a loose correlation between core counts and memory.

Local Scratch disk

- 1TB or ½ TB spinning disk
- Paging into Lustre is not efficient because of shared resource.
- With local scratch one node controls it all and wipes out when job done

Alternate computing architectures

- IBM OpenPower – not worth for propagators because GPUs very fast than Power CPU.
- ARM/ARM64 – no
- FPGA – not looked at FPGAs in the traditional sense.

Compute resources being currently used

- Limitations
  - Smaller number of GPUs being used to avoid strong scaling. Gauge generation we want strong scaling, inter-node fabric is a limitation but intra-node memory channels an issue.
- Moving away from using Lustre as scratch and local disk preferred (not all but a fraction of UQCD is headed this way)

Requirements for MILC

- Hyper Threading: this generally helps a little but isn't a big deal.
- MILC memory and disk usage is typically less than the others.
- Job sizes are sometimes larger, 64 and 128 nodes, but that also depends on the speed of a node and the jobs running. It also affects what jobs would run at FNAL vs. NERSC or XSEDE.
- MILC does not need interactive access to node, but it can sometimes be useful.

Appendix B: Joint LQCD & FNAL Acquisition Planning Committee Charge

Liz Sexton-Kennedy, CIO, Fermilab
Bill Boroski, Project Manager, LQCD-ext II Computing Project

Revision 3
July 27, 2018

Fermilab and the LQCD Computing Project are collaborating on the design, procurement, and installation at Fermilab of a high-performance computing cluster that will 1) meet the computing needs of LQCD and the Fermilab scientific community; and 2) be operated as an institutional cluster. The purpose of this committee is to understand user needs and existing computing resources and make a recommendation on the design and specifications for a new institutional compute cluster at Fermilab.

Background

On an annual basis, the LQCD-ext II Computing Project (herein LQCD) has typically executed one or more large purchases of computing hardware to augment the existing hardware portfolio operated by the project. The hardware portfolio is used by the U.S. lattice gauge community (USQCD) in support of its scientific program.

In fall 2017, LQCD began transitioning from a dedicated compute cluster model to a new operating model under which project funds will be used to purchase computing cycles from institutional clusters (ICs) operating at BNL and FNAL. BNL operates three institutional clusters and LQCD began purchasing compute cycles from BNL in January 2017.

FNAL intends to implement an institutional cluster within the next 3-4 months and LQCD is interested in committing FY18 hardware funds in exchange for compute cycles on the new system in FY19. To ensure a good outcome, LQCD and FNAL are working together on the design and procurement of the initial institutional cluster deployment.

Because the Fermilab scientific community is broad, the initial customers for the Fermilab IC are envisioned to be LQCD, CMS and portions of the Neutrino program.

Building on an acquisition strategy and annual acquisition planning process used by LQCD for many years, a joint committee has been formed to understand computing needs, create viable options, and supply a recommendation to LQCD and FNAL management regarding a preferred solution for the initial Fermilab institutional cluster procurement.

Charge

The Acquisition Planning Committee is asked to review the work completed by the LQCD FY17 Acquisition Evaluation Committee, consider changes that have occurred in the hardware architecture landscape since November 2017, and provide input into the computing hardware planning process. The intent is to help ensure that LQCD and FNAL are making the most effective use of computing hardware funds to support and advance their respective scientific programs. Specific activities include the following:

1. Gather and review computing needs of the LQCD, CMS, neutrino program user groups;
2. Understand the capabilities of the existing hardware portfolio available to LQCD;
3. Assess the vendor landscape for viable architecture options;
4. Prepare an Alternatives Analysis of viable options;
5. Present a recommendation, with technical design and cost estimate, to LQCD and FNAL computing leadership on the most cost-effective, preferred hardware solution.

Completing these tasks will provide a continued strong alignment of the LQCD hardware portfolio with the anticipated computing needs of the USQCD scientific program. The committee must also consider the alignment of a new FNAL institutional cluster with the anticipated computing needs of the FNAL scientific program.

Each committee member is asked to review supporting materials, provide input, and actively participate in committee discussions. The committee is asked to consider:

- The near- and long-term demand (at a high level) for each hardware architecture in the existing portfolio and how the proposed procurement of additional compute cycles at FNAL will augment or complement the existing hardware portfolio.

- Alternate computing architectures that may better meet USCQD and FNAL needs, considering compatibility with the existing hardware and software portfolios and infrastructure.

- The availability of production software for use by the USQCD collaboration, and the FNAL user community, to effectively utilize the capabilities of the proposed procurement of additional compute cycles.

- The ability of the proposed acquisition, along with the existing hardware portfolio, to meet the established time-based performance goals for the LQCD project.

- The alignment of the computing hardware in the existing portfolio with vendor technology roadmaps; and with the technology roadmaps of leadership-class facilities at which USQCD collaboration members run scientific software codes.

- Which USQCD and FNAL software benchmarks should be used in making the best-value assessment during the cluster evaluation process.

Hardware Budget

In considering design options, the committee should consider the following assumptions:

1. The total FY18 hardware acquisition budget is $1.545 million. LQCD will contribute $1.23 million and FNAL will contribute $315K.

2. It is possible that LQCD and FNAL will make available FY19 funds to augment the initial system purchase. Therefore, planning should consider the possibility of an expansion option on the initial procurement.
3. In addition to compute nodes, hardware funds may be required to procure associated items such as high-speed network hardware, storage hardware, etc.
4. Allocations on the initial institutional cluster deployment will be proportional to the funding contributions. On the initial system, 75% of available node-hrs will be allocated to LQCD and 25% will be available for FNAL programs. FNAL will be responsible for determining the process to allocate time to FNAL programs.

Deliverables

- A review for completeness of existing USQCD-specific software benchmarks that were used in 2018 to evaluate the performance of candidate computing architectures.
- A set of FNAL-specific software benchmarks that can be used to evaluate the performance of candidate computing architectures.
- A brief, written report summarizing the review committee's analysis of potential hardware architectures, an assessment of how effectively each potential architecture will meet the computing needs of the LQCD and FNAL scientific programs and augment existing hardware, an assessment of existing and required software benchmarks, and recommendation with conceptual design for the preferred solution.
- Recommendation(s) to the Fermilab CIO and LQCD Project Manager on how best to proceed with the hardware acquisition.
- Bi-weekly status reports created by the chair and sent to the FNAL and LQCD managers.

Timeline

The review committee should complete its full analysis and provide a final written report with recommendations to Liz Sexton-Kennedy and Bill Boroski no later than August 23, 2018.

Membership

The review committee comprises members of the LQCD project, USQCD collaboration, and FNAL scientific community with an appropriate mix of relevant technical and scientific expertise to effectively evaluate the merits of the proposed acquisition plan.
The Chair of the committee is Amitoj Singh. The membership of the committee is shown in the following table.

NAME	PROJECT ROLE	AFFILIATION	EMAIL

Robert Edwards	USQCD Representative (Chroma)	JLab	edwards@jlab.org
James Clifton Osborn	USQCD Representative (MILC)	Argonne National Laboratory	osborn@alcf.anl.gov
Chris Jones	FNAL Representative (CMS)	FNAL	cdj@fnal.gov
Bob Mawhinney	BNL Site Architect & USQCD Representative (CPS)	BNL/Columbia University	rdm@phys.columbia.edu
Gabe Perdue	FNAL Representative (Neutrinos)	FNAL	perdue@fnal.gov
Amitoj Singh, Chair	FNAL Site Architect	FNAL	amitoj@fnal.gov

Supporting Documentation

The following documentation will be provided to the review committee:

- LQCD-ext II Acquisition Strategy *(04/17/2017)*
- LQCD-ext II FY17 Alternatives Analysis *(11/14/2017)*
- FY17 Acquisition Evaluation Committee Report *(11/15/2017)*
- Anticipated Computing Needs of the Scientific Program (2017-2021)
- Table of existing systems, with capacities, in the current LQCD hardware portfolio
- Performance data on USQCD-specific software benchmarks, as summarized in the Alternatives Analysis document listed above.

Requests for additional information should be made to the committee chairperson.

Revision History

Revision #	Description of Change	Date	Author
0	Original version	07/06/18	W. Boroski
1	Added deliverable and updated membership table	7/12/18	J. Fazio
2	Replaced FY18 with initial procurement.	7/23/18	A.Singh
3	Replaced Steve with James Osborn	7/26/18	J. Fazio