From BigDog to BigDawg

Transitioning an HPC Cluster for Sustainability

Chet Langin
Office of Information Technology, Southern Illinois University, Carbondale, IL 62901, USA, clangin@siu.edu

ABSTRACT

This paper relates the experiences of managing the transition of a high performance computing (HPC) cluster from Rocks, SGE, and Cisco to OpenHPC, SLURM, and Dell. This transition was made because of sustainability issues related to security, the software, and the hardware. The BigDog HPC cluster was placed in production at Southern Illinois University Carbondale (SIUC) December, 2015, with 40 Cisco servers containing 800 CPU cores running Rocks management software and the SGE job scheduler. Sustainability issues of BigDog were encountered, especially related to keeping software updated for security reasons, and the decision was made to update the cluster for sustainability. This was done by replacing some Cisco servers with Dell hardware, and replacing Rocks with OpenHPC, and replacing SGE with SLURM. The name of the cluster was changed to BigDawg in keeping with the southern pronunciation and the spelling of the names of the university mascots Grey Dawg and Brown Dawg.

CCS CONCEPTS

• Applied computing~IT governance

KEYWORDS

Cluster Management, GPU, HPC, Long Tail of Science, OpenHPC, Rocks, SGE, SLURM, Training, XSEDE

ACM Reference format:

Chet Langin. 2019. From BigDog to BigDawg: Transitioning an HPC Cluster for Sustainability. In *Proceedings of Practice and Experience in Advanced Research Computing 2019 (PEARC'19). ACM, New York, NY, USA, 7 pages.* https://doi.org/10.1145/3332186.3332188

1. INTRODUCTION

A high performance computing (HPC) cluster like any computing system should not remain unchanged indefinitely because the hardware and software becomes outdated. Security vulnerabilities are discovered over time in nearly all software, which then needs to be periodically updated. Especially for research activities, hardware and software should remain current in order to keep up with the latest scientific developments.

However, a carefully constructed HPC cluster can be regarded by some as being like a house of cards which is carefully balanced and should not be touched in case it might all *fall down*. In which case, the administrators might need valuable time to meticulously set it up correctly, again—since an accidental misconfiguration might take days to find. Nevertheless, in spite of the desire to maintain sufficient operating capability indefinitely, eventually hardware starts to fail and security vulnerabilities are discovered in the software, and the cluster needs to be updated. One solution is to keep the cluster running as long as possible, and then completely replace it with a new cluster. This solution may be best for facilities with sufficient resources to purchase a completely new cluster when needed. Another solution, the one described in this paper, is to break the integrity of the cluster and upgrade it incrementally.

Southern Illinois University Carbondale (SIUC) is a Carnegie High Research Activity Doctoral University. The BigDog high performance computing (HPC) cluster was placed into production at SIUC during October, 2015, and was described at PEARC17 [1]. It was a 40-node cluster comprised of Cisco servers with Haswell processors containing a total of 800 CPU cores. It was provided by the university, managed by the Office of Information Technology (OIT), and made freely available to university faculty, staff, and student researchers. After extensive promotion and training of faculty, staff, and student researchers over the course of several months, usage of BigDog started consistently maxing out starting in June of 2017. By then, the hardware was 2.5 years old and a replacement plan was needed for the cluster. Also, software updates were needed, especially for security reasons, and a sustainability plan was needed. It was time to touch the *house of cards*.

Section 2 sets up the scenario of the BigDog cluster and the rationale for changing it. Section 3 describes the transition to the BigDawg cluster. Section 4 is the conclusion.

2. THE SCENARIO

Although the BigDog cluster went into production October, 2015, the hardware for it had arrived January, 2015, which is the date that became the starting point for determining expected hardware life spans. A new administrative unit, OIT Research Computing, was created to manage the cluster with the

PEARC'19, July 28-August 1 2019, Chicago, IL, USA

https://doi.org/10.1145/3332186.3332188

assistance of other OIT units such as Enterprise Systems. OIT Research Computing later in 2015 consisted of one OIT staff, the OIT Research Coordinator (the author), and a varying number of graduate assistants (GAs, currently two). The Research Coordinator wore multiple *hats*, being involved in architecture planning, system administration configuration, facilitating research, software installation, and other activities. The hands-on system administrators were the Enterprise Systems Linux Team which were also responsible for all of the other Linux servers in the OIT Data Center. Compared to other universities with HPC clusters, SIUC was a long tail science [2] facility (i.e. a small HPC center). While the original long tail reference was to a statistical distribution, it was also nice to think of it as being like the tail on an animal that is being whipped around—the supercomputing centers comprising the body of the animal while the small centers are like the tail that is being whipped around by the body. We tried to keep up with the large centers and imitate them as best as we could with our much smaller resources. Sometimes this sent us off in directions where we had to snap back to what was feasible for us.

BigDog was created with the assistance of Campus Bridging [3] (now called XSEDE Cyberinfrastructure Resource Integration--XCRI) and was an XSEDE [4] Compatible Basic Cluster (XCBC) [5]. The management software was Rocks [6] and the job scheduler was SGE. BigDog had a GPU node and two large memory nodes with 768 GB of memory each.

In more technical terms, BigDog was a 40-node Cisco cluster including a head node, a login node, and 38 compute nodes. BigDog consisted of 36 UCS C-220 servers with 64 GB of memory and 1 TB of storage; 2 UCS C-220 servers with 768 GB of memory and 4 TB of storage; and 2 UCS C-240 servers with 64 GB or memory, 1 TB of storage, and 2 NVIDIA Graphical Processing Units (GPUs). (Although two servers had GPUs, only one of them could be used as a GPU node because the other one had been installed as the head node.) The servers had Intel Haswell Xeon CPU E5-2650 v3 @2.30GHz 10-core chips. The head node acted as a file server with 48 TB of usable storage. The GPUs are NVIDIA Tesla K40m with 2,880 CUDA cores. The internal cluster network uses a Nexus 9300 switch with 48 port 10G & 6 port 40G SFP_ Twinax Copper Connections.

The cluster was running at approximately 1 percent of CPU potential because of low usage when the author was assigned to Research Computing on December 15, 2015. Almost no one locally knew how to write a job script and submit a job. Extensive training began in 2016 with workshops, group presentations, and many one-on-one sessions. A requirement was made that every new user have personal training of 30-60 minutes each on how to submit a job before they were given an account. In a few months, the BigDog cluster was properly running jobs at maximum CPU usage for days at a time with job queue wait times of up to seven days.

In 2017, hardware and software issues became apparent as we started planning on how to sustain the cluster. In January of 2017 the BigDog hardware was two years old and we needed to plan for its eventual replacement because we knew that we would not be able to budget a complete replacement at one time. We anticipated the practical life span of the hardware to be five years and we knew that we would not be able to completely replace the cluster in January of 2020, so we started researching how to incrementally upgrade the cluster. Our enterprise IT office was moving away from Cisco products to Dell, and this caused us to consider Dell products in our planning. By starting our planning three years in advance, we avoided time pressures that would have occurred if we had waited longer.

By the summer of 2017, there were also software issues that necessitated planning for changes. Software should generally be kept up to date because the longer that software goes without being updated, the more likely that security vulnerabilities will be found in it. This included the underlying CentOS software; the free version of the SGE job scheduler which was no longer being officially supported; and, many of the installed programs (which also needed to be updated per researcher requests). However, updating CentOS relied on also updating Rocks. Updating installed programs involved updating Rocks Rolls [7]. We did not have the local staff resources to create our own workarounds with SGE, Rocks, and Rolls; and, updates for them were not readily available.

3. THE TRANSITION

The transition was begun by purchasing a new Dell management server and verifying that it could manage the existing Cisco compute nodes because it was unknown to us if these different servers would work together as a cluster or not. We obtained quotes from both Cisco and Dell, and the Dell management server was less expensive. Plus, our system administrators preferred Dell. We interacted with listservs and studied features of cluster management and job management software. We selected OpenHPC [8] largely because it was in wide use and was supported by multiple corporations in the industry. We selected SLURM largely because it was in wide use by university supercomputing centers in the United States. The cluster and job management software was installed on the new Dell management server. Then, multiple Cisco servers were temporarily connected to it to verify that the Dell hardware could manage the Cisco nodes.

A transition plan was created afterwards which took the management of users into account as well as hardware and software changes. Finally, the transition plan was smoothly carried out with minimal job interruptions over a time frame of several months.

The terminology for the cluster master server changed from "Head Node" for the old Cisco server to "System Management Server," or "SMS," for the new OpenHPC Dell server. Also, whereas the old BigDog had a login node, the new BigDawg was going to handle logins through the SMS. Both the old Head Node and the new SMS also acted as file servers for shared directories such as home and scratch. Disk quotas on the old cluster started at 10 GB and were then custom set after a user request. Disk quotas on the new cluster were determined to be permanently fixed at 10 GB for home directories and unlimited on the scratch directory.

The new Dell SMS arrived December 20, 2017, and the Enterprise Systems Linux Team began verifying that it could connect with and manage the existing BigDog Cisco nodes. This was done by connecting the SMS to the same internal switch as the BigDog cluster and temporarily accessing sample Cisco nodes. Once it was verified that the Dell SMS could manage the Cisco nodes, the transition plan was finalized and approved.

In more technical terms, the SMS was a Dell PowerEdge R740XD 2U server, with 512 GB memory and 140 TB of storage (102 TB usable). Dual Intel Skylake processors, 12-core, 2.6 GHz (Intel Xeon Gold 6126 2.6G, 12C/24T/s 2UPI, 19.25M Cache, Turbo, HT (125W) DDR4-2666)—24 CPU cores total. This was sufficient for this size of cluster for an SMS that was also a gateway and file server.

Installing OpenHPC on the new Dell management server and creating images for the compute nodes was not especially challenging. The online OpenHPC documentation was thorough plus we had support from listservs and the XSEDE Cyberinfrastructure Resource Integration (XCRI) group (formerly called Campus Bridging). Noticeable differences between Rocks and OpenHPC were that OpenHPC more readily allowed flexibility in the configuration and types of software installed. OpenHPC more readily supported modules. Once OpenHPC was installed, we were surprised to learn that we needed to choose and learn a package and environment manager. I queried the XSEDE Campus Champions listserv on this and the posts appeared to favor Spack, so that is what we went with. However, we still had a learning curve before we could substantially install programs on the new cluster.

The transition plan consisted of five phases with a goal of having temporary dual clusters with a minimum of down time. By doing this in a stepwise fashion, we reduced our risk of a major failure, plus we could reverse the process in case a major problem developed. Also, there would be no cluster-wide downtime—either BigDog, BigDawg, or both would be available at all times. The hardware and users would be systematically *moved* from the old Rocks/SGE/BigDog Cisco Head Node to the new OpenHPC/SLURM/BigDawg Dell SMS. A node *move* really just meant that a network routing configuration change was made in the internal cluster switch. Server firmware was updated on each node during the transition. Since there was no urgency, the plan could be used as a guideline instead of as a hard schedule. However, since the heaviest use time of the cluster was during the summer, a general goal was made to have the transition completed before the end of Spring Semester, 2018. The transition plan was discussed with the enterprise system administrators and agreed upon on January 9, 2018.

Over 260,000 jobs had been run on the old BigDog cluster when we began dismantling it late in January, 2018. The head node had never been rebooted in over two years of production, and—even though usage was initially very low--there was never a time that we were aware of when there was not at least one job running.

With a listserv of all of our BigDog users, we notified them of our general transition plan. We advised them to backup all of their data from BigDog by April 27, 2018, because then it would be permanently shut down. We reminded them of this with periodic announcements. We made a list of all users who had not run jobs in the last 12 months and advised them that we were not renewing their accounts on the new cluster unless they specifically asked us to. We divided the active users into four types: 1) friendly adept users who could adapt to the new cluster with minimal assistance from us; 2) most users; 3) a particular high volume user with special needs; and, 4) bioinformaticians who needed the large memory nodes. We encouraged users to switch to the new cluster when they got new accounts, although we left their old accounts active until the April 27, 2018, shutdown.

We closely monitored usage of the cluster with Ganglia, SGE, and Linux commands, and remained in close contact with our users, keeping a data base of all users with information such as their academic unit, their status (undergraduate student, masters student, Ph.D. student, faculty, or staff), and the type of research that they were performing on the cluster. This is how we knew who was not using the cluster, who our friendly adept users were, who had special needs, and who the bioinformaticians were.

Generally speaking, in terms of *moving* (changing the routing configurations of) compute nodes, the five phases were 1) move 5 normal nodes, 2) move 10 normal nodes, 3) move 10 normal nodes, 4) move 10 normal nodes, and 5) move the GPU and large memory nodes. The 1-4 user types were added to the new cluster in phases 2-5, respectively, as indicated below. At some later date, after the integrity of the cluster was re-established under the SMS, Cisco compute nodes could gradually be replaced with Dell compute nodes.

Phase 1

The target date for Phase 1 was February 2, 2018, with the following specific objectives.

- Move five normal nodes. Five existing standard BigDog Cisco nodes would be rerouted to and rebuilt on BigDawg. At the end of Phase 1, BigDog would have 35 nodes with 700 CPU cores and BigDawg would have 6 nodes with 124 CPU cores.
- Reset the internal network switch. OIT Network Engineering would reconfigure the internal network switch for the five nodes which were being moved.
- Create accounts for the research coordinator and GAs. This way the facilitators could start learning OpenHPC, SLURM, and other appropriate BigDawg information and be ready to test jobs on it.
- Give notice to inactive users. They would not be given accounts on the new cluster unless they specifically requested it.
- Give notice to all users. A general description of the plan was given to all users via a listserv and they were given a general idea, per user type, of when they would be given a new account on BigDawg.
- Stop new BigDog requests. We quit making changes to the old cluster.

- Basic testing of BigDawg jobs. The research coordinator and GAs would verify that basic jobs would run on the new cluster.
- Verify Python version. Make sure the cluster had the appropriate researcher-requested versions.
- Verify R version. Make sure the cluster had the appropriate researcher-requested version.
- Implement modules. Make sure that all new programs from the start were accessible when feasible with modules (something not done on the previous cluster).
- Create a research computing Linux group. So that Research Computing could set up documentation and training areas in the file structure.

The Phase 1 BigDawg build started on January 29, 2018, and was concluded on February 19, 2018, when the basic testing of BigDawg jobs was accomplished. A normal SLURM partition was set up for the normal nodes. Other specialty nodes would have their own SLURM partitions. The 17-day delay was caused primarily by two issues. The first issue was overconfidence that caused the Phase 1 target date to be set too early. The second issue was that the cluster system administrators were also responsible for all of the Linux servers in the Data Center and also had to respond to other priorities. This 17-day delay carried through to all of the remaining phases.

Phase 2

The target date for Phase 2 was February 16, 2018, with the following specific objectives:

- Move 10 normal nodes. Ten existing standard BigDog Cisco nodes would be rerouted to and rebuilt on BigDawg. At the end of Phase 2, BigDog would have 25 nodes with 500 CPU cores and BigDawg would have 16 nodes with 324 CPU cores—15 Cisco nodes with 20 cores each plus the Dell SMS with 24 cores.
- Create new accounts for friendly adept users. Faculty and graduate students in one particular academic department had previous experience using SLURM on other XSEDE resources. We gave them the first researcher accounts on our new BigDawg cluster because they could take the most advantage of it with the least amount of our assistance.

Phase 2 was concluded on February 26, 2018. BigDawg was in production on February 27, 2018, when a Ph.D. student began running her jobs on the new cluster, and other researchers gradually followed her lead. The delay for this phase was carried over from the previous phase.

Phase 3

The target date for Phase 3 was March 2, 2018, with the following specific objectives:

- Move 10 normal nodes. Ten existing standard BigDog Cisco nodes would be rerouted to and rebuilt on BigDawg. At the end of Phase 3, BigDog would have 15 nodes with 300 CPU cores and BigDawg would have 26 nodes with 524 CPU cores.
- Create most user accounts. All users not scheduled for other phases got new accounts in Phase 3. This generally meant almost all users except those needing the large memory nodes.
- Schedule SLURM workshops. We scheduled SLURM workshops for April 4 and 5, 2018, in a library computer lab to train our users on how to migrate from SGE to SLURM. The was almost trivial substituting SLURM arguments and commands for SGE arguments and commands.
- Create training accounts. We created training accounts for the SLURM workshops.
- Set up email alerts. We configured the cluster so that SLURM could send email alerts.
- Install subversion. This software was installed for a particular user.
- Install TauDEM. This software was installed for users.
- Install MKL. Math Kernel Library was set to be installed.
- Update software on internal switch. Scheduled maintenance was set to update the software on the internal switch, which just took a few minutes. However, jobs kept running while the internal network was updated; any network requests were just delayed until the network came back.

Various objectives of Phase 3 were concluded at various times. The date of March 15, 2018, was either met or dates were set, such as for the workshops, for when completion would be made. The delay for this phase was carried over from previous phases.

Phase 4

The target date for Phase 4 was pushed back from March 16, 2018, to March 30, 2018, because the earlier phases were late. Phase 4 had the following specific objectives:

- Move 10 normal nodes. At the end of Phase 4, BigDog would have 5 nodes with 100 CPU cores and BigDawg would have 36 nodes with 724 CPU cores.
- Create a particular high-intensity user account.
- Set up a parallel environment for a particular user.

Phase 4 was concluded April 2, 2018. The delay for this phase was carried over from previous phases.

Phase 5

The target date for Phase 5 was April 20, 2018, and it had the following specific objectives:

- Move the GPU, large memory, and login nodes. The GPU node was problematic and was successfully installed November 11, 2018. The large memory nodes were moved on April 16, 2018. The login node was repurposed as a debug node. At the end of Phase 5,

BigDog would have 1 node with 20 CPU cores, and BigDawg would have 40 nodes with 804 CPU cores.

- Create accounts for SIU bioinformaticians and an Indiana University support team who relied on the large memory nodes. The Indiana University accounts were for managing the SIU Science Gateway [9], which provided access to bioinformatics software which ran on the large memory nodes.
- Install MaSuRCA [10]. This is bioinformatics software which relied on the large memory nodes and was used by the SIU Science Gateway.
- Set up a bioinformatics Linux group to allow the bioinformaticians to share files while using the large memory nodes.
- Set up the SIU Science Gateway. With assistance from Indiana University, this would provide web-based access to MaSuRCA on the large memory nodes.

Partitions were set up for the GPU node, the large memory nodes, and the debug node. The target date for Phase 5 was met except for the GPU node, which took over six months to properly set up, test, and install. Much of this GPU time was spent with staff on vacation and other types of absences. (In a small HPC center, just one person absent can stall a project.) XCRI assistance was critical for us in installing the GPU node. Plus, the system administrators had other work priorities, as well. Meanwhile, the rest of the cluster was running at full production.

Cleanup

The deadline for the hard shut down was April 27, 2018, and these two specific objectives were planned for the transition cleanup:

- Re-utilize Head Node. We had no immediate usage for the old Head Node. At first, we disabled all of the researcher accounts and left it running. One researcher needed assistance in backing up data files. The old Head Node was finally turned off on May 21, 2018, which symbolized the official termination date of BigDog. The old Head Node was eventually repurposed as a GPU node on a development cluster after its array of hot plug disks were moved to another server.
- Install FDTD? We did not know if Finite Difference Time Domain software needed to be installed, so it was last. As it turned out, it did not need to be installed.

One action item was accidentally omitted from the transition plan: to move access to Compellent storage from the old BigDog to the new BigDawg. Approximately 50 TB of storage on Compellent had been mounted for a research group on BigDog. This mount was moved to BigDawg on April 13, 2018.

Final Deadline

Our final deadline had been set to May 11, 2018, the start of summer break, when we expected the new cluster to be in maximum demand. We began formal record-keeping on April 24, 2018, at which time the overall CPU usage was 87 percent with 32 jobs waiting in queue and an estimated wait time of two days. This started a run of 31 continuous days with BigDawg running at max with wait times of up to three days. On June 12, 2018, a 57-day run began with maximum cluster usage and wait times of up to seven days. BigDawg has been in heavy usage most of the time since then.

When comparing figures 1 and 2 below, note that the data was collected differently between the old cluster and the new cluster, changing the metric by necessity in the process. The metric for the old BigDog cluster in Figure 1 was percentage of CPU cores in use, while the metric for the new BigDawg cluster in Figure 2 was the count of CPU cores in use. This metric changed during the move from SGE on the old cluster to SLURM on the new cluster because the percentage metric was misleading, largely because it included CPU cores which were not actually available for jobs.

Figure 1 below shows the usage of the old BigDog cluster at noon snap shots during the transition period between February 1 and March 12, 2018, while nodes were being removed. Users had been warned weeks in advance of the shutdown, and their jobs were expiring on the old cluster during the time period of this chart. While it looks like the cluster was running at 50 percent at the time of shutdown, most of the nodes had already been removed and all of the jobs were allowed to run to completion.

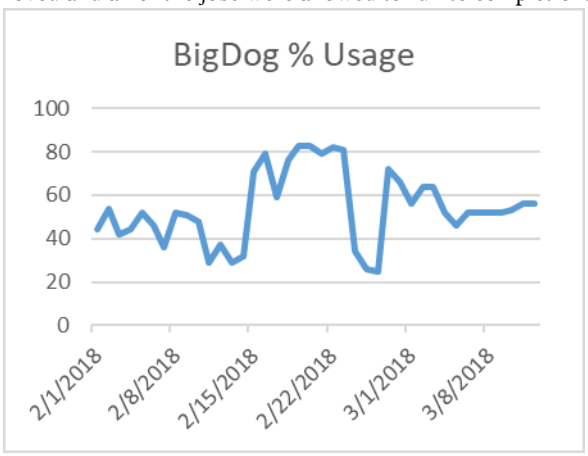

Figure 1: The above chart shows the usage of the old BigDog cluster taken in noon snap shots during the transition period in which it was being disassembled. The chart shows the percentage of CPU cores in use on BigDog from February 1, 2018, to March 12, 2018.

Figure 2 shows usage of BigDawg at noon snap shots during the transition period between April 24 and May 11, 2018. The chart shows BigDawg after all of the normal compute nodes had been added, and while the large memory and debug nodes were being added. Jobs were backed up in queue during this entire time period on BigDawg. Although the first production job began on BigDawg on February 27, 2018, noon snap shots of usage were not implemented until April 24, 2018, so snap shot

data prior to this time was not available. A time gap from March 13 to April 23, 2018, occurred between Figure 1 and Figure 2 when data was not available from either cluster.

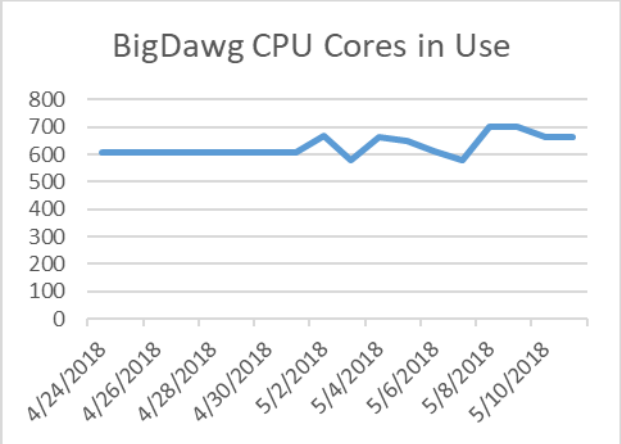

Figure 2: The above chart shows the usage of the new cluster taken in noon snap shots during the transition period when BigDawg was being built between April 24 and May 11, 2018, when all of the normal nodes had been added. The large memory and debug nodes were in the process of being added during this time period. On May 11, 2018, all of the nodes had been added except for the GPU node.

Concerning lessons learned, the overall strategy of long-term planning worked excellently. Production research jobs ran to conclusion on the old cluster while they were being initiated on the new cluster with no jobs being lost. Our overall deadline was beat by 17 days, except for the GPU node. The remounting of Compellent should have been included in the transition plan. SLURM had much better monitoring capabilities than SGE and more widespread community support. The need for Spack once the cluster was installed was a surprise. One user did not backup his files on time from the old cluster, a minor issue.

The GPU node required extensive labor to set up considering how little it was used compared to the remainder of the cluster. A catastrophe was barely averted during the installation of the GPU node. The /scratch directory was mounted during the creation of the image for the GPU node and should have been manually unmounted before the GPU node was added to cluster. However, this unmount of /scratch was not done, and the installation process began deleting files in /scratch during the installation of the GPU node. An alert system administrator caught this while it was happening and stopped the process. A couple of terabytes of files had been deleted which was inconvenient although no jobs were destroyed.

In the future, more time should be allowed in the transition plan to build the initial cluster and much more time allowed to get the GPU node into production.

This method of incremental upgrading decreased costs over buying a new turnkey cluster and gave us more control over our budget. Although we had interim delays on aspects of the upgrade, this was not a concern for us because we had started our planning so far in advance. Also, except for the GPU node, the final deadline was easily met. If time urgency was an issue, then costs would have gone up substantially.

4. CONCLUSION

Based on our previous experience installing the GPU node on BigDog, we expected difficulties in moving the GPU node over to BigDawg. Although we had verified with extensive testing that the GPUs worked on BigDog, we were not aware of any production research being done on the GPU node in over two years of operation on the old cluster. While moving the GPU node to BigDawg, we began to encounter pressure from researchers to get the two GPUs working, although when the GPU node was re-installed in the new cluster, other than testing, no one used it for production research in 2018 that we are aware of. Going into 2019, researchers were starting to use the GPU node for Deep Learning with TensorFlow.

Our only surprise *speed bump* in the overall cluster transition was how to do software installation and management in an OpenHPC environment. We settled on Spack as a package manager and found a significant learning curve in getting software installed functionally on the new cluster. We are still struggling with some software dependencies.

The transition was overall a great success. (The *house of cards* did not fall.) The dual cluster system allowed users to run jobs to completion on the old cluster while starting new jobs on the new cluster. Users could be nudged over to the new cluster gradually without sudden job-stopping deadlines. Although early interim target dates were missed by up to a couple of weeks, the overall deadline was easily met. Strategically, while the old BigDog cluster was an XSEDE Compatible Basic Cluster (XCBC), the new BigDawg cluster is an XSEDE Level 3 resource. It is enough like the supercomputers that once our researchers learn how to submit jobs on BigDawg, it is relatively easy for them to move up to the larger clusters.

Going forward, the summer of 2018 we replaced six old Cisco nodes containing 20 CPU cores each (120 total) with five new Dell compute nodes containing 24 CPU cores each (120 total). After this upgrade (and the addition of the GPU node), the cluster had six Dell computers and 33 Cisco computers. The cluster has been running successfully with this configuration, and for sustainability we can update software and continue to replace compute nodes—or add more—as needed.

References

[1] Chet Langin. 2017. We Have an HPC System—Now What? In Proceedings of Practice & Experience in Advanced Research Computing (PEARC17), New Orleans, LA, USA. July 2017, 8 pages. DOI: 10.1145/3093338.309.3341

[2] Peter Murray-Rust. 2008. Big Science and Long-tail Science. Attributed to Jim Downing. Retrieved April 2, 2019 from https://blogs.ch.cam.ac.uk/pmr/2008/01/29/big-science-and-long-tail-science/

[3] Craig A. Stewart, Richard Knepper, James Ferguson, Felix Bachmann, Victor Hazlewood, Ian Foster, Andrew Grimshaw, and David Lifka. What is Campus Bridging and What is XSEDE Doing About It? XSEDE12, July, 2012. Chicago, IL.

[4] John Towns, Timothy Cockerill, Maytal Dahan, Ian Foster, Kelly Gaither, Andrew Grimshaw, Victor Hazlewood, Scott Lathrop, Dave Lifka, Gregory D. Peterson, Ralph Roskies, J. Ray Scott, Nancy Wilkins-Diehr, "XSEDE: Accelerating Scientific Discovery", Computing in Science & Engineering, vol.16, no. 5, pp. 62-74, Sept.-Oct. 2014, doi:10.1109/MCSE.2014.80.

[5] Eric Coulter, Jeremy Fischer, Barbara Hallock, Richard Knepper, and Craig Steward. 2016. Implementation of Simple XSEDE-Like Clusters: Science Enabled and Lessons Learned. XSEDE16 (July 17-21, 2016), Miami.

[6] Philip M. Papadopoulos, Mason J. Katz, and Greg Bruno. NPACI Rocks: Tools and Techniques for Easily Deploying Manageable Linux Clusters. In Concurrency, Practice and Experience, 15(7-8):707-725, 2003.

[7] Greg Bruno, Mason J. Katz, Frederico D. Sacerdoti, and Phillip M. Papadopoulos. Rolls: Modifying a Standard System Installer to Support User-Customizable Cluster Frontend Applicances. In IEEE International Conference on Cluster Computing, Washington, D.C., September, 2004.

[8] Karl W. Schulz, C. Reese. Baird, and Eric Van Hensbergen. Cluster Computing with OpenHPC. Inaugural HPC Systems Professionals Workshop (HPCSYSPROS16). In Supercomputing 16 (SC16), Salt Lake City, Utah, USA. November 14, 2016, 6 pages.

[9] Marlon Pierce, Suresh Marru, Eroma Abeysinghe, Sudhakar Pamidighantam, Marcus Christie, and Dimuthu Wannipurage. 2018. Supporting Science Gateways Using Apache Airavata and SciGaP Services In Proceedings of Practice & Experience in Advanced Research Computing (PEARC18), Pittsburgh, PA, USA. July 2018. DOI: 10.1145/3219104.3229240

[10] A. Zimin, et al. The MaSuRCA Genome Assembler. Bioinformatics (2013). DOI: 10.1093/bioinformatics/btt476