# 2019
# High Performance Computing Intern Mini-Showcase

*Highlighting Intern Research*

August 1, 2019

Los Alamos
NATIONAL LABORATORY
EST. 1943

LA-UR-19-29011


CSCNSI 2019 Summer Students


HPC 2019 Summer Students

# Table of Contents

# Agenda

## Los Alamos Study Center
## Jemez and Cochiti Rooms (2nd Floor)

## Unclassified

## August 1, 2019

### Welcome Session: 7:30 - 8:10

7:30 Light Refreshments.......................................................................................All

8:00 HPC and Mini-showcase Welcome...... Gary Grider, HPC Division Leader

### Presentation Session I: 8:10 - 9:40
**Facilitator:** *Julie Wiens, Intern Liaison*

8:10 Exploring Computational System Health Monitoring and Reporting Solutions ..................................Jose Franco Baquera, Ethan O'Dell, and Daniel Perry

8:25 Algorithm Learning with Diagonal Neural GPUs......................................Vanessa Job

8:40 Using Containers to Build Complex Software Applications...........Calvin Seamons

8:55 An analysis of the effects of the Spectre and Meltdown patches on the Lustre parallel file system...........Tevor Bautista, Adam Good, and Amaris Velez-Candelaria

9:10 Performance characterization of a DRAM-NVM hybrid memory architecture for HPC applications using Intel Optane DC Persistent Memory Modules.....................Onkar Patil

9:25 Differential Privacy for Supercomputer Sensor Data..........................Spencer Ortega

# Agenda
# (cont.)

## Break: 9:40 - 9:50

## Presentation Session II: 9:40 - 11:25
**Facilitator:** *Julie Wiens, Intern Liaison*

9:50 sFlow Monitoring for Security and Reliability...........Xava Grooms, Robby Rollins, and Collin Rumpca

10:05 Improving SaNSA: Spark Integration and Anomaly Detection in HPC State Analysis ............................................................Dakota Fulp and Megan Hickman Fulp

10:20 Shortening Hamming Codes to Better Correct 2-Bit Errors....................Cannon McIntosh

10:35 LayerCake Workflow: Incorporating Containers into an HPC Environment ..........Logan Caraway, Donald Elrod, and Timothy Goetsch

10:50 Examining Contextual-based Error Correction Techniques in CLAMR.......Dylan Wallace

11:05 Improvements Towards the Release of the Pavilion 2.0 Test Harness ...................................Kody Everson and Francine Lapid

11:20 HPC Wrap-up Comments & Invite to Poster Session...........................Catherine Hinton, CSCNSI Co-Director

## Poster Session: 11:25 - 1:30

11:25 Poster Session (Light lunch will be provided).......................................................All

# Posters

| | |
|---|---|
| Bringing Workflows to Life Using a Graph Database with BEE | **Author:** Steven Anaya<br>**Affiliation:** New Mexico Tech<br>**Mentors:** Tim Randles and Patricia Grubel |
| Exploring Computational System Health Monitoring and Reporting Solutions | **Authors:** Jose Franco Baquera, Ethan O'Dell, and Daniel Perry<br>**Affiliations:** New Mexico State University, Missouri University of Science and Technology, Cornell University<br>**Mentors:** Kierstyn Brandt, Travis Cotton, and Graham Van Heule |
| KrakenBoot: Firmware-Level Cluster Provising via UEFI Surgery | **Author:** Devon Bautista<br>**Affiliation:** Arizona State University<br>**Mentors:** Lowell Wofford and Cory Lueninghoener |
| LayerCake Workflow: Incorporating Containers into an HPC Environment | **Authors:** Logan Caraway, Donald Elrod, and Timothy Goetsch<br>**Affiliations:** Montana State University, Texas A&M, and New Mexico Tech<br>**Mentors:** Cory Lueninghoener and Lowell Wofford |
| Revere: HPC Job Failure Early Alert | **Author:** Alexandra DeLucia<br>**Affiliation:** Johns Hopkins University<br>**Mentor:** Lissa Moore |

## Posters (cont.)

| | |
|---|---|
| Investigating the use of BlueField with OpenSHMEM and MPI over OpenUCX distributed applications | **Author:** Liliana Aguirre Esparza<br>**Affiliation:** New Mexico State University<br>**Mentors:** Wendy Poole and Steve Poole |
| Improvements Towards the Release of the Pavilion 2.0 Test Harness | **Authors:** Kody Everson and Francine Lapid<br>**Affiliations:** Dakota State University and Los Alamos National Laboratory<br>**Mentors:** Paul Ferrell, Nicholas Sly, and Jennifer Green |
| Improving SaNSA: Integration with Spark and Tivan | **Author:** Dakota Fulp<br>**Affiliation:** Coastal Carolina University<br>**Mentors:**Nathan DeBardeleben and William M. Jones |
| An analysis of the effects of the Spectre and Meltdown patches on the Lustre parallel file system | **Authors:** Adam Good, Trevor Bautista, and Amaris Velez-Candelaria<br>**Affiliations:** Arizona State University, Dakota State University, and Polytechnic University of Puerto Rico<br>**Mentors:** Garrett Ransom, Christian Storer, and Scott White |
| Porting mini-apps to ARM HPC systems | **Author:** Brian Gravelle<br>**Affiliation:** University of Oregon<br>**Mentors:** Dave Nystrom and Howard Pritchard |

## Posters (cont.)

| | |
|---|---|
| sFlow Monitoring for Security and Reliability | **Authors:** Xava Grooms, Robby Rollins, and Collin Rumpca<br>**Affiliations:** University of Kentucky, Michigan Tehcnological University, and Dakota State University<br>**Mentors:** Nicolas Jones, Michael Mason, and Marc Santano |
| HPC State Anomaly Detection and Visualization with SaNSA | **Author:** Megan Hickman-Fulp<br>**Affiliations:** Coastal Carolina University<br>**Mentors:** Nathan DeBardeleben and William M. Jones |
| Progression of Live Operation Monitoring | **Author:** David Huff<br>**Affiliation:** New Mexico Institute of Mining and Technology<br>**Mentors:** Daniel Illescas and Mike Mason |
| Falling STAR: How Livermore's Mistake Brought the Cray-1 to Los Alamos | **Author:** Nicholas Lewis<br>**Affiliation:** University of Minnesota<br>**Mentors:** Carolyn Connor, Jeff Johnson, Gary Grider, and Fredie Marshall |
| Shortening Hamming Codes to Better Correct 2-Bit Errors | **Authors:** Cannon McIntosh and Willy Kim<br>**Affiliations:** Coastal Carolina University and Florida State University<br>**Mentor:** Laura Monroe |

## Posters (cont.)

| | |
|---|---|
| The GUTS of HPC's Historic Archive: Grand Unified Text Search (GUTS) | **Author:** Wyatt Merians<br>**Affiliation:** Seattle University<br>**Mentor:** Amanda Bonnie |
| Differential Privacy for Supercomputer Sensor Data | **Author:** Spencer Ortega<br>**Affiliation:** University of Southern California<br>**Mentors:** Nathan DeBardeleben and Claire Bowen |
| The Symlink Ranch<br>An Unprivileged Overlayfs For All Distributions | **Author:** Elexis Panas<br>**Affiliation:** University of New Mexico<br>**Mentors:** Reid Priedhorsky and Jordan Ogas |
| Using Kraken for HPC Cluster Management | **Author:** Kevin Pelzel<br>**Affiliation:** Los Alamos National Laboratory<br>**Mentors:** Lowell Wofford and Alfred Torrez |
| Profiling HPC Application Resilience using DisCVar | **Author:** Stephen Penton<br>**Affiliation:** Los Alamos National Laboratory<br>**Mentors:** Nathan DeBardeleben and Terry Grové |

## Posters (cont.)

| | |
|---|---|
| The Development of LDMS on Cray Systems | **Author:** Deborah Rezanka<br>**Affiliation:** Los Alamos National Laboratory<br>**Mentors:** Mike Mason, Daniel Illescas and Howard Pritchard |
| Building Complex Software Applications Inside of a Container | **Author:** Calvin Seamons<br>**Affiliation:** Los Alamos National Laboratory<br>**Mentors:** Jordan Ogas and Jennifer Green |
| Developing a Website for High-Level Visualization of Modulefile Usage Data | **Author:** Trent Steen<br>**Affiliation:** Dakota State University<br>**Mentors:** Dan Magee and Jennifer Green |
| Examing Contextual-based Error Correction Techniques in CLAMR | **Author:** Dylan Wallace<br>**Affiliation:** Coastal Carolina University<br>**Mentors:** Nathan DeBardeleben, Terry Grové, and William Jones |
| Exploring Mellanox Bluefield SmartNICs as Accelerators for Heterogeneous Architectures | **Authors:** Brody Williams and Liliana Aguirre-Esparza<br>**Affiliations:** Texas Tech University and New Mexico University<br>**Mentor:** Wendy Poole and Steve Poole |

# Abstracts

## Exploring Computational System Health Monitoring and Reporting Solutions

**Authors:** Jose Franco Baquera, Ethan O'Dell, and Daniel Perry
**Mentors:** Kierstyn Brandt, Travis Cotton, and Graham Van Heule

At Los Alamos National Laboratory (LANL), the High Performance Computing (HPC) software stack is undergoing a complete overhaul. With this opportunity for change, we looked at the available node health monitoring software and compared their viable reporting solutions to find the tool best suited for LANL's needs. The initial programs were Ganglia, Nagios, Zabbix, Lawrence Berkeley National Laboratory's Node Health Check (NHC), and Node Cluster Check (nodediag), but due to the complex nature of LANL's systems, NHC and nodediag were the only viable solutions since they already exist within HPC's software stack and are extremely lightweight. The High-Performance Linpack (HPL) benchmarking software was then used to measure the impact of node health monitoring software and its corresponding "out-of-the-box" default tests on a cluster. After the results were recorded, we then created a unified test suite for both solutions and compared the performance results. It was found that even though NHC and nodediag have similar performance impacts, NHC is still more intuitive, has better documentation than nodediag, and would be the overall better tool to implement across HPC's software stack.

---

## Algorithm Learning with Diagonal Neural GPUs

**Author:** Vanessa Job

Algorithm learning, i.e. synthesizing an algorithm from input/output examples, is an important area of artificial intelligence. We investigate the performance of Diagonal Neural GPUs (DNGPUs), a deep learning framework for algorithm learning developed by Freivalds and Liepins. DNGPU models have been shown to be capable of learning string copying, string reversal, binary addition, binary multiplication, and decimal multiplication. We explore the performance of DNGPU models on modular arithmetic and binary subtraction.

## Using Containers to Build Complex Software Applications

**Author:** Calvin Seamons

High performance computing (HPC) scientific applications require complex dependencies, many of which are not supplied by the Linux operating system. Typically, HPC centers offer these dependencies through environment module-files that when loaded, modify the user environment to provide access to software installations. If a package doesn't exist on a system, customers must request them through system administrators or find alternatives. It is unrealistic for HPC centers to provide every unique dependency requested, thus the interest for user defined software stacks and containers are increasing. By building Model for Prediction Across Scales (MPAS) and its dependencies inside a Debian GNU/Linux 9 container image, we demonstrate that a common atmospheric simulation runs nearly identically on a Red Hat Enterprise Linux 7 Commodity Technology System (CTS1) cluster with the Intel® Core Broadwell™ architecture and a Cray System with the SuSE Enterprise Linux 12 + Cray Linux Environment (CLEv6.0) and an Intel® Core Haswell™ architecture, with minimal modifications to the container. This shows that it is possible to build complex software applications inside an unprivileged container and run it successfully across various super computers with different hardware components, Linux operating systems, and environments. Furthermore, the application computational results from their execution are essentially identical with 28 bytes differing between 2.1GB output files. Containers offer customers versatility with nominal dependencies on the system they run on. This advancement potentially allows for tremendous portability across Linux HPC systems; by encapsulating complex dependencies it gives scientists the ability to run large scale simulations on HPC resources with their own preferred software.

---

## An analysis of the effects of the Spectre and Meltdown patches on the Lustre parallel file system

**Authors:** Trevor Bautista, Adam Good, and Amaris Velez-Candelaria

In a High-Performance Computing (HPC) environment it is imperative to ensure that systems perform securely while maintaining the best performance possible. To meet these performance requirements, it is common to use a parallel file system such as Lustre. However, these performance requirements also conflict with the Spectre and Meltdown patches that are notorious for reducing performance. This research focuses on evaluating the impact these patches have on the Lustre parallel file system while assuming that the client side is always patched. The performance of the Lustre file system was evaluated before and after file system servers were patched for the Spectre and Meltdown vulnerabilities to measure the effect concerning server and client interaction. According to past research, the Spectre and Meltdown patches were expected to have a slight effect on the overall Lustre file system and a significant effect on the metadata server (MDS). Our results partially agreed with the initial expectations. The overall Lustre file system had a 15% decrease in performance due to the patches, one of the most significant effects observed. Among Lustre servers, the MDS suffered the greatest decrease in performance. Consequentially, it has been determined that Spectre and Meltdown patches present overall a significant effect on the Lustre file system. Finally, the effect of the patches on Lustre were more clearly observed as the number of nodes was increased. Due to this observation, future research could focus on executing the same benchmarking processes with a greater number of nodes. Additionally, it may be beneficial to look into the cause of the increase in performance for small-scale patched systems.

## Performance characterization of a DRAM-NVM hybrid memory architecture for HPC applications using Intel Optane DC Persistent Memory Modules

**Author:** Onkar Patil

Non-volatile, byte-addressable memory (NVM) has been introduced by Intel in the form of NVDIMMs named Intel® OptaneTM DC PMM. This memory module has the ability to persist the data stored in it without the need for power. This expands the memory hierarchy into a hybrid memory system due the differences in access latency and memory bandwidth from DRAM, which has been the predominant byte-addressable main memory technology. The Optane DC memory modules have up to 8x the capacity of DDR4 DRAM modules which can expand the byte-address space up to 6 TB per node. Many applications can now scale up the their problem size given such a memory system. We evaluate the capabilities of this DRAM-NVM hybrid memory system and its performance impact on High Performance Computing (HPC) applications. We characterize the Optane DC in comparison to DDR4 DRAM with a STREAM-like custom benchmark and measure the performance for HPC mini-apps like VPIC, SNAP, LULESH and AMG under different configurations of Optane DC PMMs. We find that Optane-only executions are slower than DRAM- only and Memory-mode executions by minimum of 2 to 16% for VPIC and maximum of 6x for LULESH.

---

## Differential Privacy for Supercomputer Sensor Data

**Author:** Spencer Ortega

One of the biggest challenges for any data-driven research is protecting and disclosing data that contains sensitive information, such as healthcare and social media data. From collaborating with colleagues to publishing new findings, there is certainly a demand for a secure way to release data to the public without compromising its privacy. Differential Privacy (DP) is a relatively new data protection method that aims to solve this problem, as opposed to its predecessors which have been proven to be vulnerable to data intrusion. For this project, we focused on implementing DP to protect supercomputer sensor data. To the best of our knowledge, DP has never been applied to this domain of data before. We ran multiple experiments on CPU temperature data, where we tested the inherit privacy/accuracy trade-off of DP by adjusting its privacy parameter (ε) and performed metrics to determine how well it protected our data. Our results from applying DP in this new domain is foundational work that explores what privacy means in the context of supercomputer sensors. This research attempts to quantify how much protection is necessary to confidently release this information to the public.

# Abstracts (cont.)

## sFlow Monitoring for Security and Reliability

**Authors:** Xava Grooms, Robby Williams, and Collin Rumpca

In the past ten years, High Performance Computing (HPC) has moved far beyond the terascale performance, making petascale systems the new standard. The drastic improvement in performance has been largely unmatched with insignificant improvements in system monitoring. Thus, there is an immediate need for practical and scalable monitoring solutions to ensure the effectiveness of costly compute clusters. sFlow is a network protocol that collects network traffic information using a packet sampling method. Los Alamos National Laboratory (LANL) possesses switches capable of utilizing sFlow which are not currently being utilized in its HPC production system. This project aims to explore the viability and impact of sFlow enabled switches on cluster network monitoring. By identifying abnormalities, cluster security and reliability can be improved through network monitoring through sFlow.

To explore the viability of monitoring with sFlow we conducted testing on a custom network consisting of a nine-node stateless compute cluster connected via a sFlow enabled switch. In addition, we had one master node to provision the cluster and one stateful node external from the cluster. The series of tests performed targeted specific network abnormalities such as cluster boot issues and services that communicate beyond secure network boundaries. Splunk is a computer software used to collect and analyze high-volume machine data within a network. Using Splunk, the data collected via sFlow was visualized to display network abnormalities during cluster boot and network activity. Dashboards were created in Splunk with simple alerts of potential network abnormalities. This gives system administrators an automated data visualization tool to easily detect abnormalities to ensure security and reliability.

---

## Improving SaNSA: Spark Integration and Anomaly Detection in HPC State Analysis

**Authors:** Dakota Fulp and Megan Hickman Fulp

Supercomputers and HPC systems consist of numerous individual nodes that synchronize to solve complex problems. Throughout its lifetime, a given node will cycle through a multitude of different hardware and software events. To coordinate all nodes in a system, a scheduler keeps information on all nodes and whether or not they are available to run jobs. However, this is only a small subset of events and administrators need to know more about every node to keep machines healthy. This poses a problem, due to the fact that these systems are large and complex, making their scheduler and system logs difficult to read and manage. Here we demonstrate SaNSA, or Supercomputer and Node State Architecture, to assist administrators in visualization of HPC states and detection of anomalies. Through the utilization of Apache Spark and Elasticsearch, SaNSA is able to run in a fraction of the time that it took in its previous iteration. Included in this improved run time is the addition of 14 new event messages that provide more clarity on the systems status. An analytics suite leveraging Apache Spark was also developed to provide more control as well as assist in the detection of anomalous nodes. Through the use of SaNSA, the average time and percentage of time each node spent in a state was revealed. Using these metrics and other calculations, anomalous nodes within the system are able to be detected so that an administrator can determine the cause and solution. We anticipate our improvements to SaNSA will be a starting point for more graphs and calculations, thus becoming more useful to administrators and users.

## Shortening Hamming Codes to Better Correct 2-Bit Errors

**Author:** Cannon McIntosh

Two-bit errors are occurring frequently. However, DECTED codes require more memory and time. We want two-bit error correction with SECDED codes. Prior work has shown a method to do this using coding theory techniques and contextual information from the application. We wish to build on this prior work to create a new class of SECDED codes with better two-bit error correction. We achieve this via a greedy shortening of extended Hamming codes and analyze our success by exploiting the mathematical structure of extended Hamming codes, such as block designs and combinatorics.

---

## LayerCake Workflow: Incorporating Containers into an HPC Environment

**Authors:** Logan Caraway, Donald Elrod, and Timothy Goetsch

Today, HPC deployment systems distribute programs onto tens to tens of thousands of compute nodes to run a job. Specifically, using Slurm, a job typically shares a scientist's NFS-mounted home directory and runs the program from there on all of the nodes allocated to the job. All programs to be executed are compiled against a base operating system, TOSS. If a user needs libraries not installed on TOSS, they must compile them into the executable or convince the HPC systems team that their libraries should be included in TOSS. This has worked well for the past 15 years, but is now an antiquated way of doing things. The goal of our project was to create a workflow that allows scientists to use Slurm as usual, as well as be able to bring custom software stacks, tools, and entire environments to any cluster using Slurm. Our vision is to transparently integrate Charliecloud into the Slurm workflow to provide this much needed functionality.

With the LayerCake workflow, we allow users to specify, at job submission time, a precrafted image to run their program. By bringing their own containers, users can preinstall any libraries or tools they need, run those preconfigured environments at scale, and use other scientists' environments. If not taking advantage of these features, the adoption of this workflow will not change anything from the user's perspective: if no options are specified, Slurm will use a default TOSS container. By containerizing jobs, nodes are safer and much less likely to be disturbed by running programs, and with unprivileged namespaces, no administrator overhead will be needed.

## Examining Contextual-based Error Correction Techniques in CLAMR

**Author:** Dylan Wallace

As we approach exascale, extreme-scale systems grow in scope and complexity. This growth is associated with an increase in hardware failure rates, so detectable but uncorrectable errors (DUEs) are increasingly problematic in extreme-scale systems as they are traditionally viewed as unrecoverable and the down time they cause is costly. Correction techniques that address these errors are of great importance and we've taken a contextual-based approach to correcting DUEs inside CLAMR, a cell-based adaptive mesh refinement (AMR) mini-app. Extending previous work in which 2-bit DUEs were applied to every cell in a static CLAMR mesh before applying two correction methods (average neighbor and conservation of mass), we've resumed the simulation to see how these changes would manifest in the mesh. We discuss the methods of running 75k experiments to analyze the results of these correction methods so we can determine which method is the most effective.

---

## Improvements Towards the Release of the Pavillion 2.0 Test Harness

**Authors:** Kody Everson and Francine Lapid

High performance computing production support entails thorough testing in order to evaluate the efficacy of a system for production-grade workloads. There are various phases of a system's life-cycle to assess, requiring different methods to accomplish effective evaluation of performance and correctness. Due to the unique and distributed nature of an HPC system, the necessity for sophisticated tools to automatically harness and assess test results, all while interacting with schedulers and programming environment software, requires a customizable, extensible, and lightweight system to manage concurrent testing. Beginning with the recently refactored codebase of Pavilion 1.0, we assisted with the finishing touches on readying this software for open-source release and production usage. Pavilion 2.0 is a Python 3-based testing framework for HPC clusters that facilitates the building, running, and analysis of tests through an easy-to-use, flexible, YAML-based configuration system. This enables users to write their own tests by simply wrapping everything in Pavilion's well-defined format.

We took advantage of this system by writing three new commands as well as adding more functionality to two existing commands. Our contributions to the test harness also included improving and adding three plugins to facilitate automatic parsing of test output files to interactively display and convert results to json format for logging. We improved Pavilion to support advanced configuration capabilities including better variable handling and allowing users to add environment commands to their kickoff scripts. In addition, we developed better table displays and integrated four parallel benchmarks tests to demonstrate these features. As Pavilion approaches production and community readiness, the improvements this project accomplished aided in the meeting of production-level testing requirements.


Los Alamos National Laboratory

## About the High Performance Computing Mini-Showcase

The HPC mini-showcase provides a unique opportunity for interns in the high performance computing field to present their research. The intent is to broaden expertise and prepare our interns for careers in their field. This is an excellent forum for interns to network and make professional contacts.

## About the Sponsors

This year's mini-showcase was sponsored by:

- The High Performance Computing (HPC) Division, including the division office, HPC staff, and
- administrative teams.
- The Information Science and Technology Institute (ISTI)
- The New Mexico Consortium (NMC)
- The Theory, Simulation, and Computation Directorate (ALDSC)
- The National Security Education Center (NSEC)

## About Los Alamos National Laboratory

Los Alamos National Laboratory, a multidisciplinary research institution engaged in strategic science on behalf of national security, is operated by Los Alamos National Security, LLC, a team composed of Bechtel National, the University of California, The Babcock & Wilcox Company, and URS for the Department of Energy's National Nuclear Security Administration. Los Alamos enhances national security by ensuring the safety and reliability of the U.S. nuclear stockpile, developing technologies to reduce threats from weapons of mass destruction, and solving problems related to energy, environment, infrastructure, health, and global security concerns.

## About The New Mexico Consortium (NMC)

The New Mexico Consortium (NMC) is a non-profit corporation formed by the three New Mexico Universities under a teaming agreement with The University of California (UC) to partner with Los Alamos National Laboratory (LANL) to advance scientific research and education in New Mexico. The NMC leverages the strengths of three research universities to achieve common goals, builds joint programs in support of common interests, develops strategic partnerships with government, industry and other universities insupport of the partnership, and provides common organization and facilities to support these initiatives.

Los Alamos NATIONAL LABORATORY EST.1943

New Mexico CONSORTIUM

ISTI INFORMATION SCIENCE & TECHNOLOGY INSTITUTE

LA-UR-19-29011