Revealing Power, Energy and Thermal Dynamics of a 200PF Pre-Exascale Supercomputer

Woong Shin
Oak Ridge National Laboratory
Oak Ridge, TN, USA
shinw@ornl.gov

Vladyslav Oles
Oak Ridge National Laboratory
Oak Ridge, TN, USA
olesv@ornl.gov

Ahmad Maroof Karimi
Oak Ridge National Laboratory
Oak Ridge, TN, USA
karimiahmad@ornl.gov

J. Austin Ellis
Oak Ridge National Laboratory
Oak Ridge, TN, USA
ellisja@ornl.gov

Feiyi Wang
Oak Ridge National Laboratory
Oak Ridge, TN, USA
fwang2@ornl.gov

ABSTRACT

As we approach the exascale computing era, the focused understanding of power consumption and its overall constraint on HPC architectures and applications are becoming increasingly paramount. Summit, located at the Oak Ridge Leadership Computing Facility (OLCF), is one of the fastest and largest pre-exascale platforms in operation today. This paper provides a first-order examination and analysis of power consumption at the component-level, node-level, and system-level, from all 4,626 Summit compute nodes, each with over 100 metrics at 1Hz frequency over the entire year of 2020. We also investigate the power characteristics and energy efficiency of over 840k Summit jobs and 250k GPU failure logs for further operational insights. To the best of our knowledge, this is the first systematic analysis of power data of HPC system at this scale.

CCS CONCEPTS

• **General and reference → General conference proceedings**; • **Hardware → Enterprise level and data centers power issues**.

KEYWORDS

HPC, GPU, power, energy, reliability, telemetry, data analysis

ACM Reference Format:
Woong Shin, Vladyslav Oles, Ahmad Maroof Karimi, J. Austin Ellis, and Feiyi Wang. 2021. Revealing Power, Energy and Thermal Dynamics of a 200PF Pre-Exascale Supercomputer . In *The International Conference for High Performance Computing, Networking, Storage and Analysis (SC '21), November 14–19, 2021, St. Louis, MO, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3458817.3476188

1 INTRODUCTION

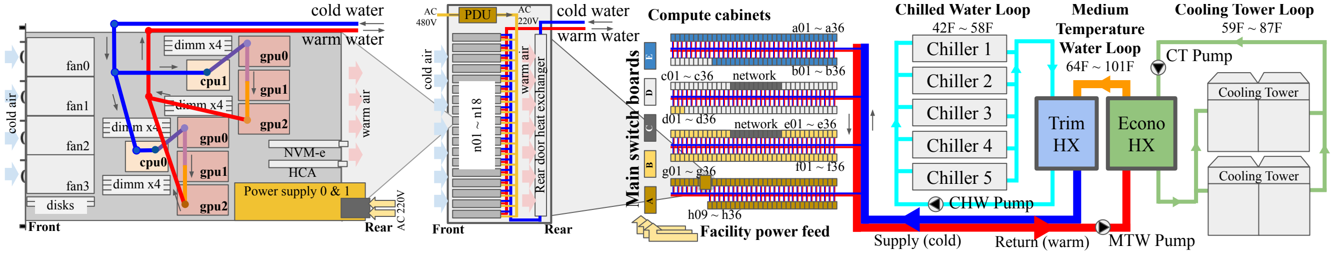
Ever since the end of Dennard's scaling [10], the power and energy footprint of HPC data centers has been increasing. In the pursuit of breaking the exascale barrier, such energy demands present a significant challenge that ultimately requires the need for highly energy-efficient HPC data centers. HPC practitioners must consider issues at both the HPC system-level and the data center-level[4, 36]—both adding additional dimensions to system design and deployment. At exascale, system design, deployment, and operation all require exploring design parameters and operational parameters in uncharted territories. In such an exploration, operational data plays an important role [28]. Insights from data provide the necessary means for system designers or operators to drive efficiency close to the limits; however, gaining such insights can be a challenge.

In this work, we aim to provide such insights necessary for future energy-efficient exascale HPC data centers—with a focus on revealing the power and energy dynamics of an HPC data center both as a whole and as related to its job history. To achieve this goal, we have instrumented the Summit HPC data center[3], the US Department of Energy (DOE) 200PF pre-exascale system at the Oak Ridge Leadership Computing Facility, and have gathered data that covers power and energy data from the Summit system itself and its supporting infrastructure. In particular, we have leveraged an out-of-band telemetry stream, accumulating data for the entire year of 2020 at a 1Hz sampling rate from all 4,626 nodes, 100 metrics per node that cover the power and temperature of individual components within each node. This data was cross analyzed with the job scheduler logs, the power & cooling supply information from the facility, and GPU failure logs. With such comprehensiveness, we are able to deeply characterize the tendencies of Summit's leadership-class jobs down to its smallest scale HPC workloads, and reveal its short-term and long-term impact on HPC data center energy efficiency. The contributions of this work are as the following:

HPC data collection and workload power characterization: We reveal the dynamic power consumption behavior of HPC applications and their impact at unprecedented resolution and scale. Using the high-resolution power measurement data, we have characterized the spatial-temporal behavior of HPC applications at scale. In doing so, we have developed analysis methods that are geared towards understanding dynamic patterns across a long period of time. We were able to quantify the frequency and amplitude of power consumption especially tied to the well-known behavior of HPC applications themselves and the operational policies in place.

SC '21, November 14–19, 2021, St. Louis, MO, USA

ACM ISBN 978-1-4503-8442-1/21/11…$15.00
https://doi.org/10.1145/3458817.3476188



a) IBM AC922 8335-GTX node (Two CPUs & six GPUs watercooled)

b) Cabinet (18 nodes each)

c) Compute Cluster Floor (257 cabinets, 4,626 nodes)

d) Central Energy Plant (20MW)

Figure 1: Architectural overview of the Summit system at Oak Ridge Leadership Computing Facility

HPC data center cross-cutting interactions: We also provide a comprehensive illustration of the dynamic impact of HPC workloads on energy efficiency from a data center holistic point of view. With a special focus on correlating sensor and operational data from multiple domains of the HPC data center, we provide illustrations of cross-cutting dynamics that go beyond the traditional boundaries such as software vs. hardware and HPC platform vs. facilities. We provide understandings of the dynamic interactions between components across layers within the HPC data center that can help to design various control algorithms, methods, systems that are required to enable an energy-efficient HPC data center.

Reliability and variability study at scale: Finally, we provide an analysis reporting the long-term system reliability characteristics of a large-scale dense GPU deployment that leverages medium temperature direct liquid cooling. This study explores the long-term relationship between various GPU failures and the temperature the system yields given the HPC workload and the cooling supplied by the supporting facility. This study leverages high-resolution component level thermal readings across the whole cluster associated with temporal features such as HPC job allocations, and the spatial features such as node locations or component locations.

2 ARCHITECTURE

Power, energy and cooling of Summit. Summit, currently ranked No. 2 on the Nov. 2020 edition of the Top500 list [11] of supercomputers, is a 122.3 petaflops pre-exascale system (200 petaflops theoretical peak) located at the Oak Ridge Leadership Computing Facility (OLCF). Summit caters to the DOE Office of Science's (SC) workload consisting primarily of full-system jobs that solve challenges in research areas of national importance such as advanced scientific computing, basic energy, biological and environmental, fusion energy, high energy physics, and nuclear physics. Summit incorporates Power9 CPUs, NVidia Volta V100 GPUs, and Mellanox Enhanced Data Rate (EDR) InfiniBand (IB) network technologies (Table 1). It has a total of 4,626 IBM AC922 nodes each powered by two CPUs and six GPUs. Summit's peak power consumption is 13 MW and is currently ranked No. 11 on the Green500 list with 14.719 GFlops/watts, supported by a 20MW facility (Figure 1) that provides the necessary power and cooling to the compute cluster floor (Figure 1-(c)).

Summit leverages medium temperature water (MTW) in the secondary loop to maximize cooling efficiency at this scale and density. With a 70°F (20°C) supply temperature from a central energy plant (Figure 1-(d)), this secondary loop first touches each 18 node cabinet's rear-door heat exchangers and the cold plates of the two CPUs and the six GPUs on each node (Figure 1-(a),(b)). Here, MTW minimizes chilled water use by enabling cooling towers based on evaporative cooling in the primary loop (Econo HX - Figure 1-(d) right) when the weather conditions are advantageous (i.e., wet-bulb temperature is below the necessary supply temperature).

Table 1: Summit system specification

OLCF Summit	
Nodes	4,626 IBM AC922C 8335-GTX nodes
Cabinets	Total 257 watercooled cabinets, 18 nodes each
Power consumption	13 Megawatts peak
Econ. Primary loop	8 cooling towers (59°F ~ 87°F)
Trim Primary loop	5 chillers (42°F ~ 48°F)
Secondary loop	supply: 64°F ~ 71°F & return: 80°F ~ 100°F
IBM AC922C 8335-GTX nodes	
Processor	2 x IBM Power9 22C 3.07GHz direct water-cooled
GPU	6 x NVidia Volta GV100 direct water-cooled
Memory	512GB DDR4 + 96GB HBM2 + 1.6TB NVMe
Interconnect	Dual-rail Mellanox EDR Infiniband
Node max power	2,300 Watts (220V ~240V AC)
Thermal output	8,872 BTU/hr max (2,600 Watts)
IBM Power9 22C Processor	
TDP	300 Watts
Frequency	3.07 GHz
Cores	22 per CPU / 4 threads per core
NVidia Volta V100 SXM2	
TDP	300 Watts
Frequency	1335 MHz ~1530 Mhz (boost)
Processors	80 SMs
Memory	16GB 4096-bit HBM2

The facility relies on chilled water only when the cooling towers cannot sufficiently remove heat to drive supply temperature down to the required supply temperature (Trim HX loop - Figure 1-(d) left). This is especially true during the hot and humid Tennessee summer months. With this dual mechanism primary cooling loop, the facility uses chilled water for only about 20% of the year.

Telemetry data for MTW operations: Figure 2 depicts the telemetry system that supports MTW operations. Data center-level electrical and mechanical data is aggregated with power and temperature data emitted from individual nodes and is processed, summarized,

Table 2: Data specification (12 months)

id	Source	Sample Interval	Rows	Footprint	Description
(a)	Summit per node OpenBMC Telemetry	1 sec	134B	8.5TB (compressed)	Per-node, per-component power and temperature
(b)	Central Energy Plant (CEP)	Approx. 15 sec	2M	256MB	Mechanical, electrical and environmental data
(c)	Job scheduler allocation history	End of every job	938K	285MB	Project, user, node count, allocation param. submit, start & end time
(d)	Per-node job scheduler allocation history	End of every job	87M	14GB	Per-node job allocation history, end of job statistics
(e)	NVidia GPU XID error log	At occurrence	256K	50MB	GPU error hardware and software errors

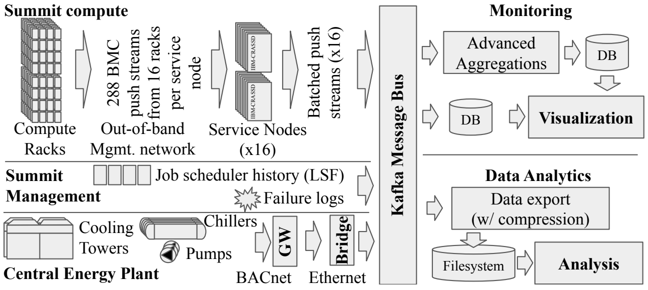


Figure 2: Telemetry system for Summit's power and energy

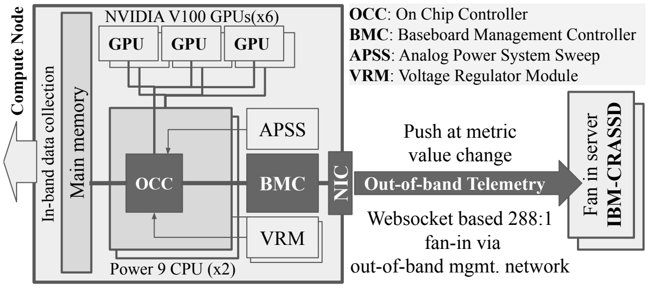

Compute Node
NVIDIA V100 GPUs(x6)
GPU
GPU
GPU
In-band data collection
Main memory
APSS
OCC
BMC
VRM
NIC
Power 9 CPU (x2)
OCC: On Chip Controller
BMC: Baseboard Management Controller
APSS: Analog Power System Sweep
VRM: Voltage Regulator Module
Push at metric value change
Out-of-band Telemetry
Websocket based 288:1 fan-in via out-of-band mgmt. network
Fan in server
IBM-CRASSD

Figure 3: Compute node out-of-band data collection path from the on chip controllers (OCCs) via per-node board management controllers (BMC)

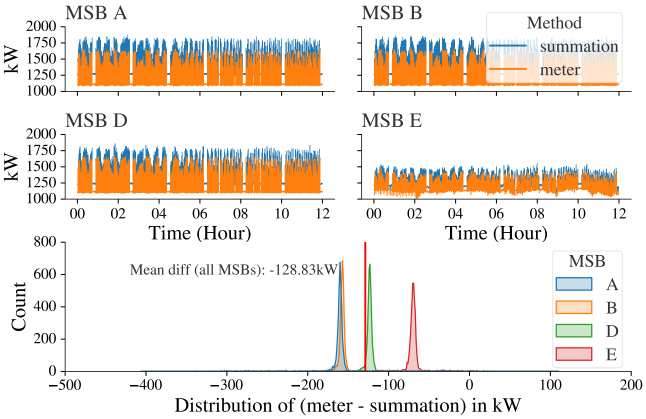


Figure 4: Power meter vs. per-node sensor at scale

and rendered to engineers in near real-time. Given the node allocation and outside weather conditions, the facility cross-checks notable data center control parameters such as MTW supply & return temperature and MTW flow with the histogram-based component-wise temperature distribution summary of the HPC platform (27,756 GPUs and 9,252 CPUs).

This system relies on the out-of-band telemetry streams pushed to the telemetry system at a 1Hz data rate from the baseboard management controllers (BMCs) of each Summit compute node [2] as depicted in Figure 3. At a 460k metrics/sec ingest rate, per-node & per-component power and temperature sensor changes from all Summit compute nodes are propagated to the point-of-analysis with an average 4.1-second delay [32]. Despite the high-resolution and low-latency propagation, no impact occurs on HPC applications due to the method's out-of-band nature.

While the primary use of such telemetry data streams is on near real-time verification of data center state, the data streams are exported and archived for long-term analysis. We have decided to store the high-frequency datasets in their original form despite the projected size when accumulated. By leveraging several lossless data compression methods throughout the telemetry data pipeline, the footprint of an aggregated 460k metrics per second data stream from Summit resulted in a manageable 1MB/s data stream. Accumulation of an entire year data resulted in a moderate 8.5TB footprint on disk per year.

3 METHODOLOGY

Data collection and pre-processing: We take a data-intensive approach to analysis that leverages the comprehensive dataset from the data center as described in Section 1. We have leveraged the near real-time telemetry data stream developed to support MTW operations described in Section 2. The data stream has a high sampling rate at 1Hz and a broad scope that includes the state of individual components in the HPC system under load and the facility's state that supplies the necessary power, energy, and cooling. Table 2 describes the character of each data-stream involved.

Leveraging a parallel data analytics & computing tool Dask [9], we first aggregated the 1Hz data stream down into a manageable size & form. Here, we have coarsened the data to a 10-second window, but we have avoided information loss by storing statistical information such as min., max., mean, and standard deviation values of the samples in each window per time-series from each node. We further collapsed the coarsened per-node time-series data into a cluster level time-series using different aggregation methods we have implemented depending on the analysis. For studies that require job context, we performed the collapse after joining the time series with job scheduler allocation logs. We further joined the resulting datasets with facility mechanical & electrical data from the facility and the GPU failure logs to perform a data center level end-to-end analysis.

Per-node power measurements and error management: The power measurement primarily used in this work is an aggregation of per-node power measurement to approximate the cluster-level power consumption or to perform a per-job breakdown of power consumption. However, there were challenging aspects of the method that can impact its accuracy. 1Hz emits from each node were based on a 1-second interval sampling of a 500μs instantaneous power measurement[1]. The payloads were timestamped later at the aggregation point after an average 2.5-second delay (max. 5 seconds) [32]. Also, due to the number of sensors involved (i.e., two power supplies in total 4,626 nodes), various errors caused by per-node variation could accumulate.

To cope with these errors, we have primarily used 10-second window coarsening (min., max., mean, and standard deviation) before performing the summarization at the cluster level. This method was validated against the measurements from the main switchboards (MSBs) that distribute power to the compute cabinets (Figure 1-(c)). Specifically, we compared the summation of the per-node 10-second mean input power (Figure 1-(a)) that belongs under each MSB with per MSB 10-second mean power consumption at each MSB. Figure 4 shows the result of this comparison.

Overall, our method using the summation of the 10-second mean power of each node was on average 11% from the actual physical MSB measurement. However, we have found that the oscillation and its amplitude were in phase and had the same magnitude, respectively. The distribution of the differences over time, seen in lower portion of Figure 4, lie tightly around their mean values and have low standard deviation. Yet, there were subtle differences between the mean values of the differences across MSBs, indicating an external factor is influencing the MSB discrepancies. With the focus on understanding the dynamics of HPC consumption, the tight sync supports the use of the per-node measurements for arbitrary job level per-node aggregations that follow this section. For brevity, we accept the 11% differences as a caveat and largely rely on the magnitudes of the more readily available on-node sensor measurements.

4 SYSTEM POWER AND ENERGY

4.1 Overview

In this Section, we characterize the system's power and energy consumption and observe that the behavior of HPC applications can best explain the resulting power consumption. In the HPC job context, we have analyzed aspects to characterize the exact workloads common to 840k Summit jobs across various node counts and the total system trends during 2020 and how they ultimately impact facility operations. Leveraging the high-resolution per-node 1Hz power measurement data, we quantify the dynamics of cluster-level power consumption envelopes.

Cluster power and energy: Figure 5 outlines the power and energy trend of the Summit system and the supporting facility during the entire year of 2020. Average power consumption was between 5MW and 6MW with a constant small percentage of extremes that touches both the system idle (2.5MW) and peak (13MW) power consumption throughout the year. Under this power consumption, the average power usage effectiveness (PUE) [2] of the data center is 1.11 thanks to the cheap evaporative-based cooling. However, in summer, the PUE increases to an average of 1.22 due to the chilled water used to trim down the supply temperature (bottom row)[3]. As mentioned in Section 2, data center PUE is primarily impacted by the outside weather condition. Yet, PUE is also impacted by the large extremes of the HPC workloads and is just futher exaggerated in the summertime when the "expensive" cooling is employed.

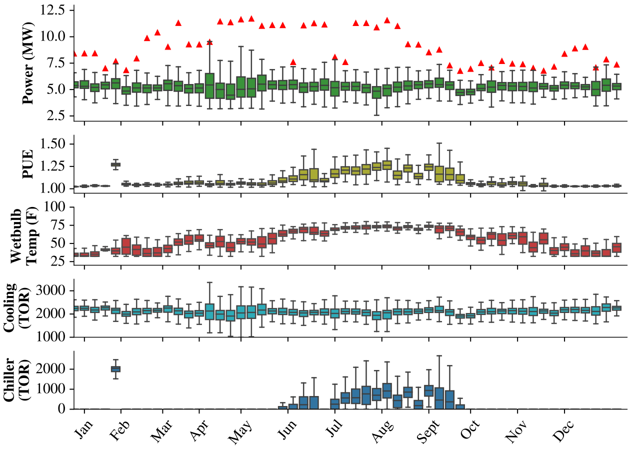


Figure 5: Summit power and energy trends (year 2020)

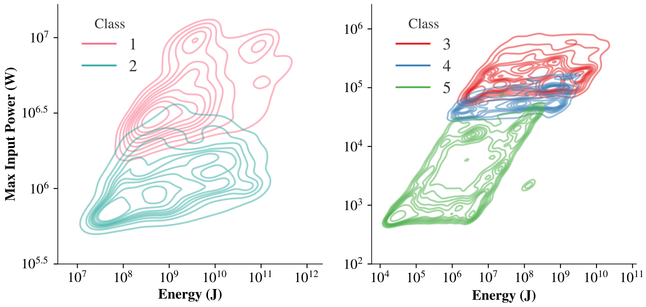


Figure 6: Distribution of the total energy consumed during a job run versus the maximum input power for each Summit scheduling class.

Impact of HPC applications: To understand the extreme differences in power consumption, we must examine the impact of HPC applications. Figure 6 depicts jobs of five different scheduling classes based on the number of nodes as shown in Table 3 using the Gaussian kernel density distribution of input power and total energy consumed. Classes 1 and 2 are large-scale jobs that run on more than 20% of the Summit's nodes, and Classes 3-5 are considered small-scale jobs which run on less than 20% of the nodes. The smaller contour rings show higher data density, and the large outer contour rings represent low-density regions. Plots for Classes 3-5

[1] Energy accumulators were not available from the BMCs.

[2] PUE is a metric used to evaluate data center efficiency. It is the total facility energy divided by the IT equipment energy, where a value close to 1.0 indicates an efficient data center

[3] The high PUE at 1.3 during early February was due to scheduled maintenance of the cooling towers that led the CEP to run at 100% chilled water

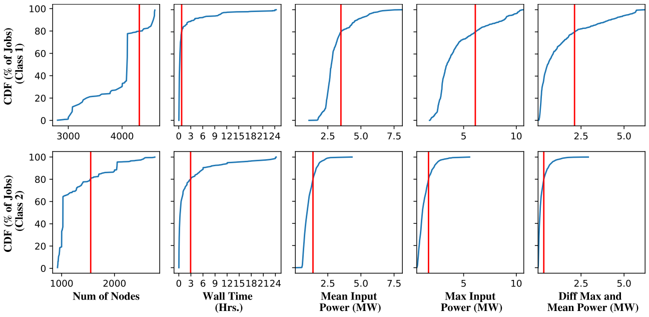


Figure 7: Cummulative distribution function of jobs with respect to various jobs' features.

Table 3: Summit scheduling classes by job node count.

Summit scheduling policy		
Class	Node Range	Max Walltime (hours)
1	2765 - 4608	24
2	922 - 2764	24
3	92 - 921	12
4	46 - 91	6
5	1 - 45	2

have many small contour rings showing that the joint distribution of maximum input power and energy has a multi-modal pattern with several high-density regions. In contrast, the large-scale classes have few smaller contour rings showing most of the points are concentrated in fewer peaks. Overlap of maximum input power is minimal across different groups showing that the maximum power is strongly correlated with job class, while the energy value of jobs across classes has a more extended overlap range.

Figure 7 shows the cumulative distribution functions of jobs with respect to job node count, job wall time, mean power, maximum power, and the maximum and mean power difference. The above row considers jobs of class 1, while the second row considers jobs of class 2. The red vertical lines correspond to 80 percent in the cumulative densities. Over 60% of Class 1 jobs use node counts in the upper band region of over 4,000 nodes with a maximum frequency at 4096 nodes. However, 80% of Class 2 jobs run on less than 1,500 nodes, with most jobs running on 1,024 or 1,000 nodes. Class 2 jobs have a longer run time than class 1; 80% of the Class 2 jobs take almost up to 3 hours, whereas 80% of jobs in Class 1 take less than 43 minutes. The Class 1 and Class 2 mean power curve is similar, but the magnitude of the power values is higher for Class 1 due to the higher node counts. Also, 80% of the jobs consume less mean power because only 20% of the jobs lie beyond the red line or are in a higher power region. We see a similar trend for the maximum power in two classes, 80% of Class 1 and Class 2 jobs consume less than 6.6 MW and 1.6 MW, respectively, but their largest values are 10.7 MW in Class 1 and 5.6 MW in Class 2. The difference between maximum and mean values show significantly more variation in Class 1 than in Class 2.

Figure 8 shows the distribution of maximum input power and energy across different domain sciences for Class 1 and Class 2. We can learn various features of jobs from different sciences based on domain-specific power and energy plots. Variation in peak power values can be due to different applications and kernels used across various disciplines. Also, certain codes and algorithms may dominate certain disciplines, so their power and energy profile outweighs contributions from less popular cases. High power peaks near 10 MW in Class 1 show a high degree of parallelism, and the significant variation in energy consumption is an artifact of job run time.

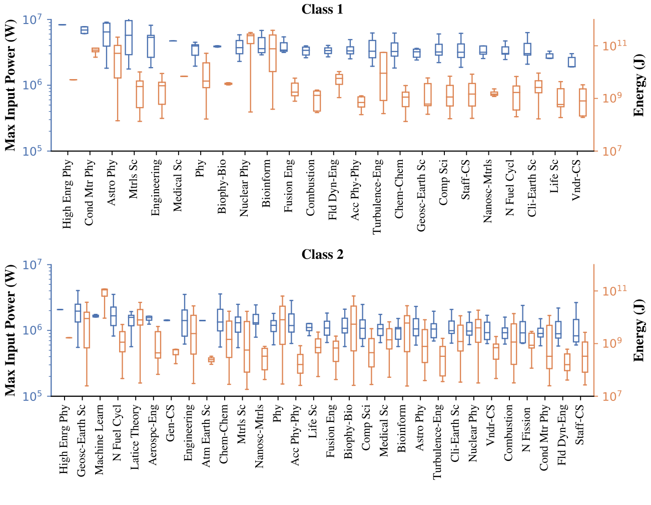


Figure 8: Job level power and energy consumption breakdown by science domains

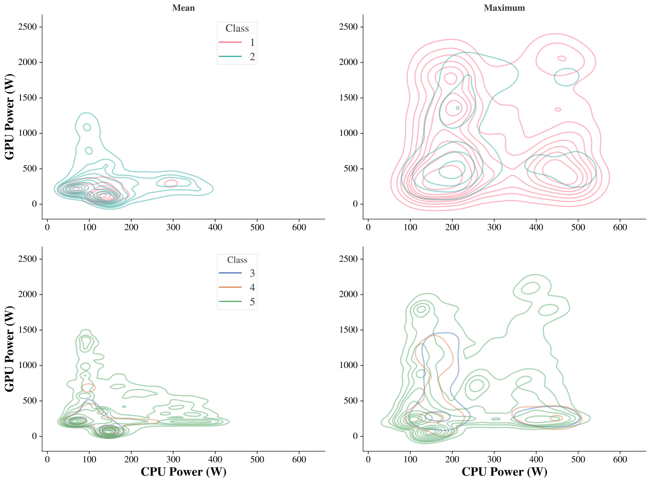


Figure 9: Distribution of CPU and GPU per-node power consumption for mean values and maximum values for all the jobs across five different classes.

Component level power consumption: The component-level power consumption characterization per-node provides insight into the CPU and GPU power usage profile. The joint density distribution of power consumed by CPUs and GPUs per-node for each job is shown in Figure 9. Each compute node in the Summit machine has 6 GPUs and 2 CPUs. The regions having smaller rings in the plots represent higher density, and we can observe that the spread of data density is mainly near the x-axis or y-axis. We infer from the mean plots that the jobs concentrated into dense regions near

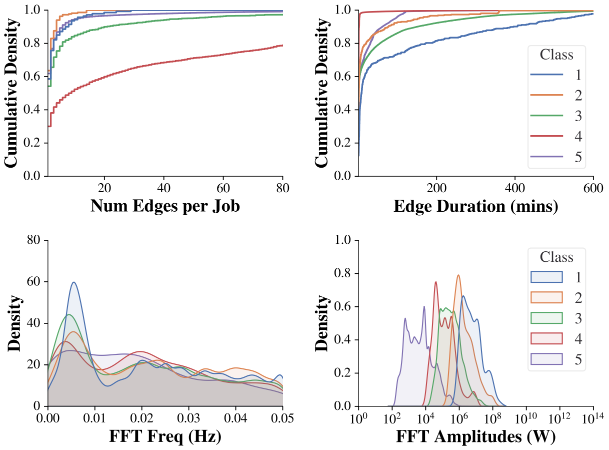


Figure 10: Power consumption dynamics (overview)

the x-axis represent CPU-intensive jobs, and the set of jobs aligned near the y-axis are primarily GPU focused. A broader spread along the y-axis of maximum value plots shows that there are quite a few jobs that utilize GPU resources for a certain period. Fewer contour rings near the upper right corner of maximum power plots show jobs do not heavily use both CPU and GPU resources together.

Summary: *Power consumption of the system exhibits two very distinct modes of power consumption: average and peak power consumption. The difference is notable, and applications within distinct scheduling classes play a significant role on these extremes. Scheduling classes delineated well the separate modes. Specifically, just a few large-scale jobs define the peak power consumption, while smaller jobs define the average power consumption. Decreasing per job node count, GPUs are indeed the main workhorses that define the peak, where as CPUs mostly define the average.*

4.2 Power Consumption Dynamics

Peaks, valleys, and spikes of power consumption: Amongst the features common to power and energy profiles, swings with large amplitudes and short time intervals are the most impactful to the power systems. Massive swings in an HPC data center work load can cause the power system to experience both frequent and sustained pressure. Figure 10 contains an analysis of rising and falling *edges* in the power profile. We define a rising or falling edge as a change in the power usage of greater than 868 W averaged across nodes in the job. Edges for jobs at 4,608 nodes, or full system-scale, therefore need a change of at least 4MW to be defined as an edge in this analysis. The upper left plot shows the cumulative distribution of edge counts per job for each Summit job category. The upper right plot contains the cumulative distribution of durations for each edge in the upper left plot. The *edge duration* is defined as the time from the start of the rising edge to the end time where power has returned back 80% from its peak to its initial power.

We report that the large majority of jobs, 96.9%, experience neither rising nor falling edges during their lifetime, so we cumulate the distribution of those with greater than one swing. Class 4 jobs experience the most edges and the durations of each edge is incredibly short relative to other categories. This would appear as higher frequency swings of at least 40 to 79 kW depending on the job size. In contrast, the leadership Class 1 jobs experience relatively fewer edges per job with 95% of jobs with an edge experiencing less than 15 edges (still quite substantial operationally), but the duration distribution indicates that these edges are relatively more sustained for the life of the job. Whereas 60% of Class 1 job edges last less than 25 minutes, 20% of the leadership class edges last longer than 200 minutes. Another observation is that the Class 5 jobs have a job-scheduler dictated wall-limit of 120 minutes that is confirmed here by the non-differentiable point at the maximum cumulative density.

To further characterize the largest swings, we employ Fourier analysis to find the most critical frequencies and their amplitudes. The power data for each job is differenced due to its auto-correlated nature and then an FFT is applied to discover the maximum amplitude and its corresponding frequency. Distributions for both most important frequencies and amplitudes are found in the lower portion of Figure 10. Certain frequencies are common across multiple classes, such as .005Hz, or 200 second intervals, which indicates that the largest shifts in power consumption are occurring most commonly with periods of 200 seconds, irrespective of job category. Of the five categories, leadership Class 1 jobs are the most evenly distributed in terms of important frequencies with only a small taper towards .05Hz. Amplitude distributions skew toward lower amplitudes but with a conspicuous stair-stepping pattern towards the higher amplitudes. The location of the stairs could be a artifact of popular node counts, such as 3,000 or 4,096, or component level power consumption differences, such as different arithmetic engines being used.

In Figure 11, snapshots of power consumption and PUE around rising edges at various MW levels have been summarized by superimposing the snapshots aligned at the edge. For example, the 7MW rising edge on the top-rightmost figure is a summary of four 7MW rising edges detected during summer. Shading indicates the surrounding 95% confidence interval among the detected snapshots over time. The power consumption versus PUE plots are noticeably symmetric and inversely proportional, and optimal PUE (closest to 1.0) favors the largest 7MW swings in power consumption where workload is highest. In the 4MW rising edge snapshots, we notice common, but brief spikes that occur on roughly 60 second intervals, where as the 7MW snapshots the duration is longer and the average edge begins to fall after 120 seconds. Interestingly, the dynamic power consumption behavior exhibits largely similar patterns across similar magnitudes of edges.

Summary: *Within a small percentage of large class jobs, only a short window of time actually contributes to the peak power consumption. However, transition to this peak power consumption is rather violent and can happen within tens of seconds. Analysis on frequency and amplitude suggests that we can characterize these jobs—possibly for potential predictive model based on categorization; however, power prediction is most likely improbable with the power consumption history alone. We find that sufficient information about the HPC applications and job scheduling, such as job allocations and job operations may best explain future behavior.*

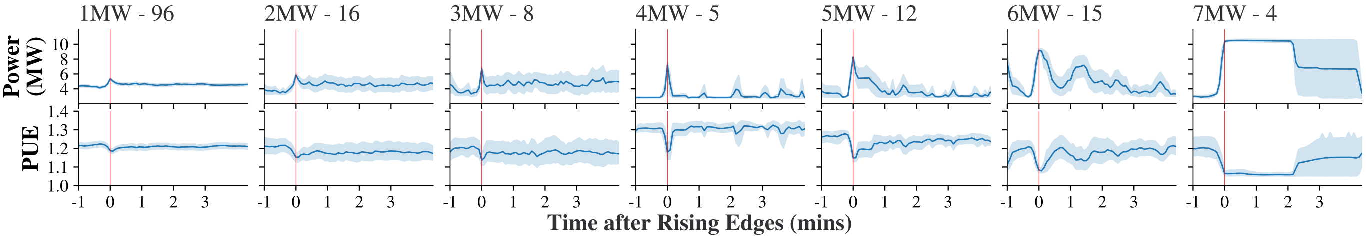


Figure 11: Time series snapshots of the rising edges detected during the summer (July 24 to Sept. 30, 2020). Per amplitude class (1MW bins left to right), multiple snapshots are superimposed and aligned at their rising edges ("0 mins" in the X-axis). Title per amplitude format: "rise amplitude class - snapshot count".

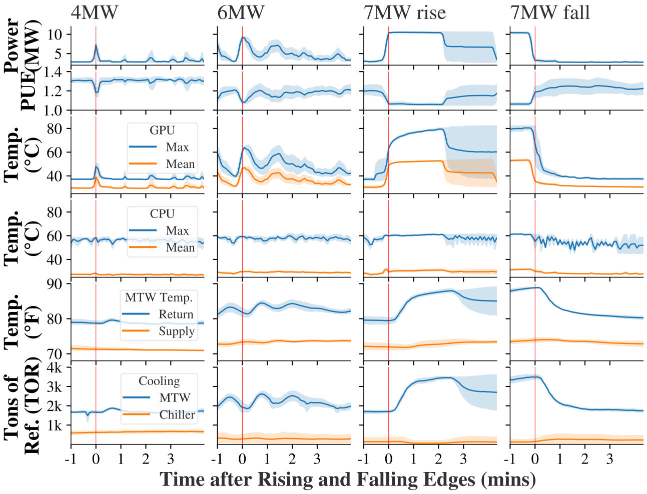


Figure 12: Component-wise temperatures and thermal response of Summit's water cooling system during the summer (July 24 to Sept. 30, 2020). Multiple time series snapshots are superimposed and aligned at their rising or falling edges ("0 mins" in the X-axis).

5 RESPONSE OF THE SYSTEM

Large-scale synchronous parallel behavior of HPC applications induces the dynamic power consumption, as seen in Section 4. Then, how Summit's system responds to intense changes in workloads.

Power and thermal response of the system: To provide a clear look into the facilities overall dynamics under load, we extend the edge detection and summarization method used in Section 4.2. A pulse from an increase in overall workload is followed by reciprocal response from Summit's water cooling system. Figure 12 shows the power consumption and PUE along with node component statistics and cooling system measurements during three types of rising edges and one falling edge. Rows 2 and 3 contain node component statistics, such as the mean and maximum temperatures for both the V100 GPUs and Power9 CPUs. Alternatively, Row 4 contains the temperatures of the incoming and outgoing MTW supply and Row 5 has the actual supply tons of refrigeration from the MTW supply and the higher cost chillers. Again, PUE is mostly symmetric and inversely proportional to the power consumption, except that after the large 7MW falling edge, there are noticeable oscillations in the PUE's steady state behavior. GPU temperatures themselves are tightly responding to power swings, with the maximums continuing to rise (in the 7MW case) after the rising edge and temperature means follow the power envelope. CPU temperatures in contrast remain relatively fixed throughout the rising and falling edges. Much of the component-wise temperature variances are potentially due to physical variances in spatial location and manufacturing processes.

The cooling system's response is triggered by the measured temperature of the return MTW supply. The data indicates a roughly one minute delay before the tons of refrigeration and supply temperatures increase. Furthermore, comparing the rising and falling edges shows that the attenuation of the return MTW temperature and tonnage is much slower during decreases than increases.

Energy efficiency: Overall, the PUE of Summit is maximized and stable for long leadership class jobs in which workload is constant and machine utilization is highest. Constant context switching that comes from serving smaller jobs, though absolutely necessary, greatly increases the variance and overall magnitude of the PUE. The large differences between peak and average power consumption, ultimately, have an impact on HPC data center energy efficiency. Leveraging live telemetry data mentioned in Section 2, the facility allows GPU and CPU temperatures to rise as high as possible but under the threshold where the system can operate without adverse effects such as thermal-induced throttling or even device shutdowns. However, even though peak power is observed in few job allocations, the cooling plant is tuned to safely handle peak power consumption at any moment. The power swings in Figure 11 and 12 can be faster than the cooling mechanical systems, and so abundantly safe precautions are maintained. Such practices result in a general overcooling of the system, but the difficulty in responding to such dynamic power movements makes it unavoidable. For the Summit system, the impact can be noticeable when running on expensive cooling during summer.

Summary: *Energy-efficient HPC systems require an end-to-end view of the data center. Traditionally, there was an information blockage between the HPC platform and the underlying facility; however, this may not be the case in the future. Advanced cooling technologies, such as medium temperature water direct liquid cooling, already require facility engineers to monitor HPC platform CPU & GPU thermal responses. Further, the two separate control domains could benefit by*

Table 4: GPU failure composition. The double-ruler separates failure types by those that can be associated with user applications (top) and those that cannot (bottom).

GPU error	Count	Max. count per node
Memory page fault	186,496	1,189 (0.6%)
Graphics engine exception	32,339	259 (0.8%)
Stopped processing	22,649	118 (0.5%)
NVLINK error	8,736	8,462 (96.9%)
Page retirement event	851	37 (4.3%)
Page retirement failure	210	89 (42.4%)
Double-bit error	179	33 (18.4%)
Preemptive cleanup	162	34 (20.1%)
Internal microcontroller warning	74	33 (44.6%)
Graphics engine fault	44	5 (11.4%)
Fallen off the bus	31	8 (25.8%)
Internal microcontroller halt	29	4 (13.8%)
Driver firmware error	26	2 (7.7%)
Driver error handling exception	21	21 (100%)
Corrupted push buffer stream	11	9 (81.8%)
Graphics engine class error	1	1 (100%)

having information flow between each other. Making the large power consumption visible or deterministic enough to be predictable by the cooling plant can open additional energy savings opportunities. We believe safe and robust data-driven algorithmic approaches can claim these opportunities.

6 RELIABILITY

We investigate the long-term impact of the overall thermal state that results from the dynamics of power consumption (Section 4) and the response of the system (Section 5). Motivated by prior research [22, 26, 33] on Titan [1], the predecessor of Summit, we focus on the impact on GPUs, especially under the influence of high temperatures.

6.1 Failures

General trend: GPUs are the most power-consuming component of a node, which is associated with additional challenges such as reliability and overheating. At the same time, GPU reliability is paramount for many scientific applications, where a failure can potentially obliterate an existing compute effort with tens of thousands of node-hours already invested.

NVIDIA GPU XID error logs (Table 2-(e)) indicate that Summit saw a total of 251,859 GPU errors in 2020, whose composition by type is shown in Table 4. Only a tiny fraction is constituted by driver- and hardware-failures, such as double-bit or off-the-bus errors, while the vast majority can be associated with user applications. The presence of nodes accounting for a disproportionate share of non-software errors of each type heavily suggests the presence of manufacturing defects (e.g. [38]).

To better understand the co-occurrence between different types of GPU failures, we counted them separately for every Summit node, and computed the Pearson correlation between the resulting 4,626-dimensional vectors for every pair of failure types. Figure 13 shows the correlation coefficients significant at 0.05 after applying the Bonferroni correction to account for the number of pairs. Beside the expected co-occurrences such as between double-bit errors, preemptive cleanups, and page retirement events, the analysis shows an extremely strong correlation between internal micro-controller warnings and driver errors handling GPU exception. The latter suggests that soft errors such as micro-controller warnings can be efficient for early diagnostics and ultimately prevention of fatal driver errors.

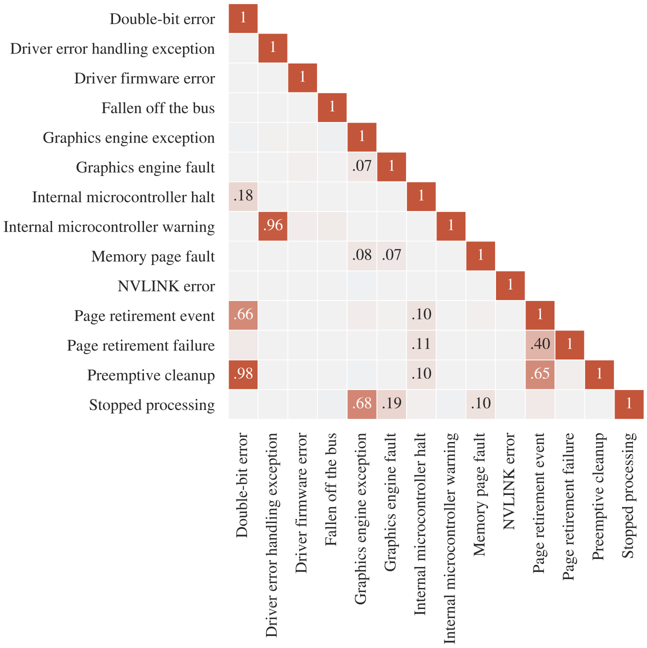
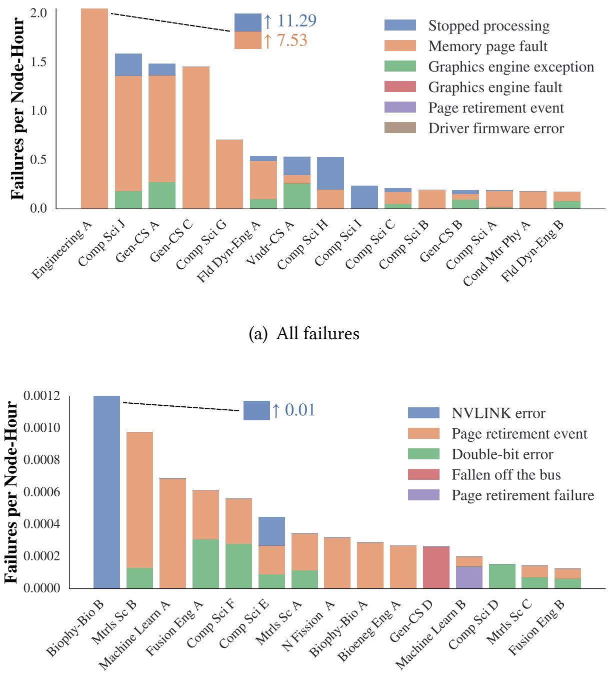


Figure 13: GPU failure co-occurrence

GPU failure frequency per node-hour of computation in a job depends significantly on the application domain and project it belongs to. Figure 14-(a) shows top-15 projects in terms of failure frequency, and decomposes their error occurrences by type. Figure 14-(b) shows an analogous breakdown for the subset of GPU failures that are not associated with user application, and the corresponding top-15 projects. High variability of hardware errors across application domains and individual projects within them indicates that distinct workload patterns are a major factor affecting GPU reliability.

High-temperature and failures: [22, 26, 33] suggest that GPU overheating is likely to be a contributing factor to off-the-bus and double-bit error occurrences. To account for workload specificity of a job encountering an error, we considered temperature at the offending GPU core in the context of temperature distribution across all GPUs within the job at the moment of failure. We used the z-score, the number of standard deviations above the mean, as a metric of thermal extremity that is independent of the associated workload. Figure 15 shows how the frequency of failure occurrences depends on their thermal extremity. Due to missing temperature data[4], only a part of GPU failures is represented by the plot. To make the analysis more informative, we removed the data for a "super-offender" node accounting for 97% of all the NVLink errors, which suggests a permanent chip malfunction. Almost no distributions

[4] Due to software issues in the data aggregation path, there were significant loss in temperature data during spring and early summer of 2020



(a) All failures

(b) Hardware failures

Figure 14: GPU failure overview

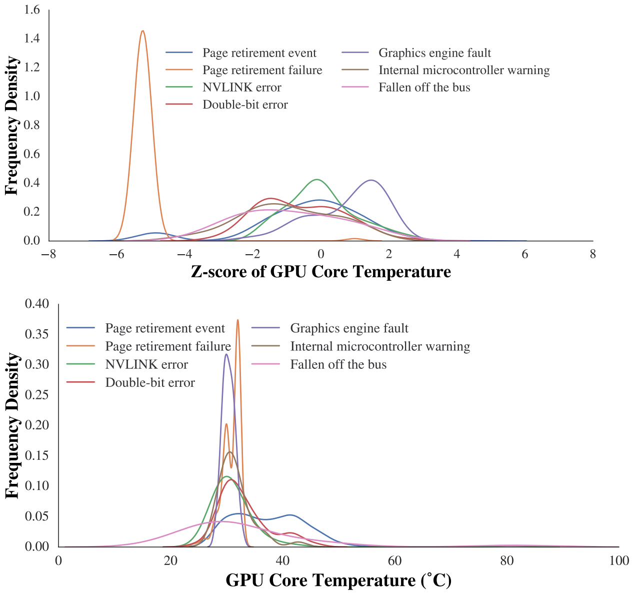


Figure 15: Frequency of GPU failures based on their thermal extremity

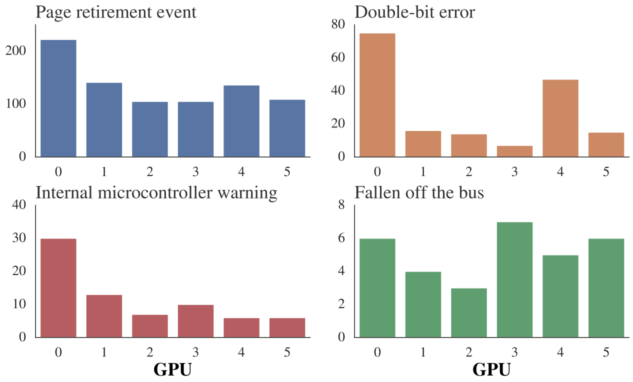


Figure 16: Counts of GPU failures per component placement

exhibit left skewness, indicating that overheating is not a significant aspect of GPU failures of any type except for, potentially, graphics engine faults. Somewhat surprisingly, thermal distributions for double-bit and off-the-bus errors, as well as for internal microcontroller warnings and page retirement failures, are right-skewed. This may potentially suggest that these errors tend to occur more often on the GPUs that did not yet warm up from a task. In terms of the absolute temperature, the only failures taking place at 60°C or above were 1.4% of the NVlink errors and 5.2% of the off-the-bus errors — in particular, the highest known temperature for double-bit errors was 46.1°C.

GPUs getting cold water that was already used for cooling other cards are potentially more susceptible to overheating and ultimately to failures. Given the cooling order of node components (see Figure 1-(a)), we would expect to see an increase in failures from GPU 0 to GPU 1 to GPU 2 (for CPU 0) and from GPU 3 to GPU 4 to GPU 5 (for CPU 1), if this were the case. However, the observed trend in Figure 16, showing the breakdown of failures by placement of the offending GPU, is close to the reverse. Although "second-hand" GPU water-cooling does not seem to be an issue, the plot demonstrates other peculiar trends of GPU reliability. While frequent failures on GPU 0 and low failure count for CPU 1-connected GPUs can be attributed to the presence of single-GPU and single-CPU jobs, the reasons for heightened frequency of double-bit errors and page retirement events on GPU 4 are not immediately clear. At the same time, high off-the-bus failure count on these three GPUs may indicate that irregularity of HPC tasks makes graphics cards more susceptible to falling off the bus.

Summary: *Compared to the prior generation system Titan[1], the GPUs are not the same. Different architecture and cooling mechanisms introduce different outcomes. While high-temperature was a reason for the major errors in the case of Titan, its direct effect on GPU failures in the current system is not significant. However, given the influences of applications and the highly dynamic nature of HPC workloads we have observed so far, temporal characteristics in temperature changes may have high impact. The reason for errors being associated with lower-temperature GPUs is inconclusive. A potential explanation may be related to the 10-second aggregations used in the paper. If failure of a GPU makes it go idle during a large-scale load, its average temperature*

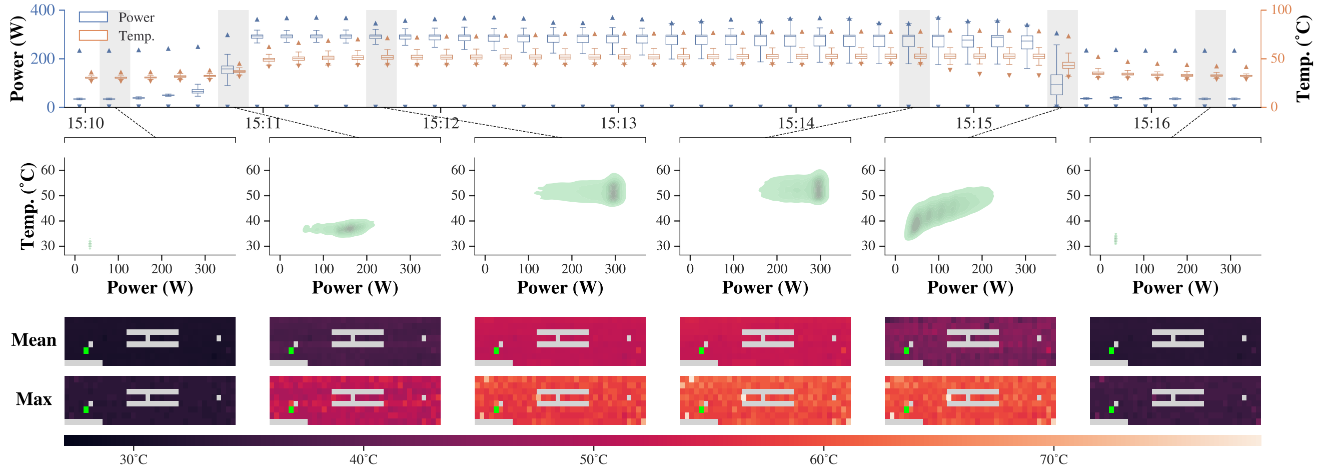


Figure 17: Variation of GPU and power consumption during peak system load

over the 10-second interval is expected to be lower compared to the rest of the GPUs.

6.2 Variability

Impact of node variability: Figure 17 shows variability in power consumption and temperature of individual GPUs during the compute-intense part of an exemplar large-scale job spanning 4608 Summit nodes and lasting around 21.5 minutes in total. The job was part of a material science application BerkeleyGW [6], which computed quasiparticle eigenenergies using density functional theory. The criterion for selecting this job was its near-full utilization of Summit compute resource. In particular, the insignificant variability in GPU power consumption exhibited by the job at certain time intervals allows for studying other factors of thermal behavior of the system.

Distributions of 10-second averages of power consumption and core temperature for each of the 27,648 GPUs is represented by the blue and orange boxplots, respectively. The system transitions between near-idle and maximum capacity in less than half a minute, and component temperature follows the power consumption trends in a matter of seconds.

The second row of the plot shows the power-temp relation in each of the 27,648 GPUs at six selected time intervals. The plots illustrate that GPU core temperature depends on its power consumption in a monotonic, near-linear way. However, power consumption is not the only factor in the thermal response of the system, as can be seen from the third plot. Despite the spread of non-outlier (defined according to the 1.5 interquartile range rule) power consumption in individual GPUs being as narrow as 62W, the spread of their non-outlier temperatures is 15.8°Celsius. It suggests that a part of the temperature variability can be attributed to manufacturing variation in the chips, and to uneven impact of the cooling system due to their location. Despite the scale of the workload, the vast majority of the GPUs do not exceed 60°C, which speaks to the efficiency of Summit's cooling.

Impact of spatial locality: The two bottom rows of Figure 17 show GPU temperatures in individual Summit cabinets at the same time intervals as above. The "Mean" row shows average GPU temperature within each cabinet, while the "Max" row shows their highest GPU temperatures achieved within the 10-second intervals. Grey rectangles denote that there are no Summit nodes involved with the job in the corresponding location. The bright green rectangle represents the cabinet for which no telemetric data for the duration of the job is available [5]. Spatial distribution of heat during the peak loads on Summit (time intervals 3 and 4) remains quite even. Darker areas at both top and bottom of the "Mean" plot for the 5th time interval suggest that heat dissipation on Summit exhibits a slight spatial locality. The presence of standalone light yellow cells in the "Max" plots is explained by the cold water outake points towards other systems that are located by these cabinets.

Summary: *Given the numerous Summit components, the system is influenced by subtle differences that are introduced by spatial features within the node and on the actual Summit floor. We show that such features can be observed and monitored to great benefit. In an energy-efficiency context, continuous component-level temperature readings are useful to monitor and control this aspect of the system in the face of reliability issues. The tight thermal response of components to power consumption dynamics may require higher resolution. Sheer number of components can impose a challenge, but the advances in modern HPC data collection systems makes such practices possible.*

7 RELATED WORKS

Energy-efficient HPC: Power and energy challenges of exascale HPC drove various efforts at multiple areas in the HPC data center towards energy efficiency [5, 36]. To achieve performance goals within the power constraints, HPC practitioners employ energy-efficient hardware components and infrastructure [7, 27]. Also, software-driven fine-grained application task management [13, 39],

[5] There are missing values in telemetrics coming from 24 nodes associated with the job, of which 18 constitute the bright green cabinet and the other 6 belong to distinct cabinets.

job scheduling [14, 19, 21, 35, 40], are employed to pursue additional energy efficiency.

Power and energy analysis: Studies on the power and energy of HPC systems help to design and develop techniques and strategies that improve energy efficiency by providing insights of HPC workload characteristics [12, 17, 30, 31, 37], hardware impact [8, 16], energy-performance trade-offs [12, 18], infrastructure power and energy [25, 29], or manufacturing variability [15]. However, such studies are often limited to specific benchmarks, periods, or specific non-production HPC systems.

Patel et. al, [30, 31] addresses this issue by enabling continuous power consumption monitoring on a production system and providing an in-depth analysis on power consumption across many jobs. Our work shares a similar purpose and subject however, this work differs in its comprehensive scope that aims to provide a cross-cutting view of an HPC data center as a single system revealing cross-layer energy optimization opportunities throughout the year. Also, in terms of power measurement and handling, this work use out-of-band methods [20] that benefits from no-impact high-frequency power measurement from the HPC system. Compared to [20], this work uses a different similar out-of-band technology [32] and relies on a relatively low-frequency sample rate at 1Hz, but fully captured the power measurements without a loss for a long period of time and aimed to deliver a full analysis.

GPU reliability studies: [34], [33] and [22] used exploratory data analysis to investigate GPU errors in connection with temperature, workload, location, and intrinsic characteristics of individual GPUs. Ostrouchov et al. conducted survival analysis of GPUs based on their inventory times and physical location in [26]. [23] and [24] employed linear regression, support-vector machine, random forest, and neural networks to predict GPU failures in HPC and data center systems.

8 CONCLUSION

Considering the breadth of the users and science domains that Department of Energy owned and operated open science HPC data centers must serve, and due to the increasing power and energy footprint of HPC systems, HPC data center operational data becomes uniquely invaluable and enlightening as we move towards the exascale era. In this work, a comprehensive analysis of a heavily instrumented pre-exascale supercomputer reveals the dynamic nature of HPC power consumption with respect to HPC applications, the cooling system, and its overall energy efficiency. After close examination of the impact of scale on the data center, we reveal that relatively high-frequency & high-amplitude transitions between two vastly different power consumption modes (peak and average) introduce non-trivial issues at the data center level resulting in increased operational cost caused by overcooling. This suggests aggressive power and energy aware application optimizations and scheduling policies can have impact even on HPC deployments like Summit that impose no power constraints on its jobs. However, such pursuits should be backed by data that provide a good understanding of the bounds on system reliability. We consider the importance of the operational data comprehensiveness for HPC data centers as a single system, and such comprehensiveness, combined with the tools and techniques that enable cross layer understandings, will open opportunities for more energy efficient HPC data centers. We believe that purpose-driven end-to-end instrumentation across the data center should be motivated by solid use-cases and opportunities. To this end, we believe this work contributes marching towards such activities.

9 FUTURE WORK

Operational impact: Analysis of the dataset resulted in new and deeper understanding of the workload itself and the response of the system. On top of the understanding of the system from day-to-day observation of the short-term monitoring activities, the longer range, high-resolution, multi-datastream analysis revealed new knowledge. Such knowledge was not readily applied to any operational decisions yet, but the knowledge of such behavioral dynamics have set the foundation for improvement and facility oversight.

While confirming the well known swinging behavior of HPC applications, this work revealed the magnitude, swing-frequency and occurrence throughout the year. Large power swings that range from 4MW to 7MW do happen in a few tens of seconds, but accounts for rather low occurrence only with larger classes of jobs throughout the year. With the understanding of such parameters, the long-term cooling system response analysis revealed potential avenues for facility improvements. For example, the higher PUE experienced on the high-magnitude falling edges revealed potential parameter tunings that can be made to the control system that stages and de-stages cooling capacity (i.e., cooling towers) based on the load. Also, with the first direct-liquid cooled GPU system for OLCF, the tight thermal response in the lower band temperature that closely follows the power envelope has introduced new aspects of the system to follow-up in terms of energy-efficiency and resilience.

Tangentially, the result of this analysis influences how OLCF approaches monitoring and operational data analytics. Towards higher-fidelity data-center wide comprehensive approaches investments and vendor engagements are made towards enabling such data intensive facility and system operations in the continuum of generations of OLCF systems to come.

In-depth analysis and modeling of system-wide application job power profiles: In continuation, we plan to develop a combined user and job power-profile *fingerprinting* capability that will aid in predictive analysis of system- and node-level power consumption. The drastic and immediate power swings show that using the power consumption histories alone will most likely be insufficient. A model that leverages both power consumption histories and the streaming job queue data mediated by the fingerprints may lead to more accurate predictions. From the existing 2020 Summit job power dataset, we create fingerprints as vector representations that describe user job power consumption at the OLCF. Fingerprints are then clustered and user-portraits are generated. Queued jobs will assume the average power portrait of the user given job size, job launch arguments, and project ID. A default measure of uncertainty would be associated with the queued-job fingerprint, and as the job runs, the uncertainty in the fingerprint would converge. Simultaneously, reliance on the fingerprint may or may not wane depending on how well the actual power consumption is matching the predictions. With these features, we believe that predictive power analytics would become more feasible.

Further, the analysis of job's power consumption versus the scientific domain motivates a more in-depth jobs and power consumption study. As the percentage of machine learning and artificial intelligence workloads has increased considerably, we will also be focusing on analyzing the power usage patterns of ML/AI applications and how they differ from traditional modeling and simulation jobs on OLCF leadership systems. We will expand our analysis of CPU and GPU power consumption analyzing the correlation between AI/ML jobs and GPU across different science domains and how these jobs affect the HPC power profile.

ACKNOWLEDGMENTS

We are grateful to the technical staff members at ORNL, James Rogers, David Grant, Scott Atchley, Don Maxwell, Rick Griffin, and Saeed Ghezawi who provided insights in many aspects of this work that greatly assisted this work. We also express special thanks to Todd Rosedahl for providing technical support in understanding the telemetry data streams from the IBM Power9 system.

This work was supported by, and used the resources of, the Oak Ridge Leadership Computing Facility, located in the National Center for Computational Sciences at ORNL, which is managed by UT Battelle, LLC for the U.S. DOE (under the contract No. DE-AC05-00OR22725). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

REFERENCES

[1] 2012. *Titan.* Retrieved April 2, 2021 from https://www.olcf.ornl.gov/olcf-resources/compute-systems/titan/

[2] 2015. *OpenBMC Event subscription protocol.* Retrieved April 2, 2021 from https://github.com/openbmc/docs/blob/master/rest-api.md

[3] 2018. *Summit.* Retrieved April 2, 2021 from https://www.olcf.ornl.gov/olcf-resources/compute-systems/summit/

[4] Axel Auweter, Arndt Bode, Matthias Brehm, Herbert Huber, and Dieter Kranzlmüller. 2011. Principles of Energy Efficiency in High Performance Computing. In *Information and Communication on Technology for the Fight against Global Warming*, Dieter Kranzlmüller and A. Min Toja (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 18–25.

[5] Natalie Bates, Girish Ghatikar, Abdulla Ghaleb, Gregory A. Koenig, Sridutt Bhalachandra, Mehdi Sheikhalishahi, Tapasya Patki, Barry Rountree, and Stephen Poole. 2014. The Electrical Grid and Supercomputing Centers: An Investigative Analysis of Emerging Opportunities and Challenges. *Energiinformatik* 38 (2014), 111–127.

[6] Mauro Del Ben, Charlene Yang, Zhenglu Li, Felipe H. da Jornada, Steven G. Louie, and Jack Deslippe. 2020. Accelerating Large-Scale Excited-State GW Calculations on Leadership HPC Systems. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (SC'20).*

[7] Henry Coles, Michael Ellsworth, and David J Martinez. 2011. Hot for warm water cooling. In *International Conference for High Performance Computing, Networking and Storage (SC'11).*

[8] Jared Coplin and Martin Burtscher. 2016. Energy, Power, and Performance Characterization of GPGPU Benchmark Programs. In *2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW).* 1190–1199. https://doi.org/10.1109/IPDPSW.2016.164

[9] Dask Development Team. 2016. *Dask: Library for dynamic task scheduling.* https://dask.org

[10] Robert H. Dennard, Fritz H. Gaensslen, Hwa-Nien Yu, V. Leo Rideout, Ernest Bassous, and Andre R. LeBlanc. 1974. Design of ion-implanted MOSFET's with very small physical dimensions. *IEEE Journal of Solid-State Circuits* 9, 5 (1974), 256–268.

[11] Jack J. Dongarra, Hans W. Meuer, and Erich Strohmaier. [n.d.]. *Top500.* Retrieved May 7, 2019 from https://www.top500.org/

[12] Mark Endrei, Chao Jin, Minh Ngoc Dinh, David Abramson, Heidi Poxon, Luiz DeRose, and Bronis R de Supinski. 2018. Energy efficiency modeling of parallel applications. In *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'18).* IEEE, 212–224.

[13] Neha Gholkar, Frank Mueller, Barry Rountree, and Aniruddha Marathe. 2018. PShifter: Feedback-Based Dynamic Power Shifting within HPC Jobs for Performance. In *Proceedings of the 27th International Symposium on High-Performance Parallel and Distributed Computing (HPDC'18).* Association for Computing Machinery, 106–117.

[14] Hiroaki Imade, Takahiro Kagami, Tomohiro Otawa, Kouichi Hirai, Yoshio Sakaguchi, and Naoyuki Fujita. 2019. Automatic Power Saving Method by Energy Aware Job Scheduler. In *2019 IEEE International Conference on Cluster Computing (CLUSTER).* 1–2.

[15] Yuichi Inadomi, Tapasya Patki, Koji Inoue, Mutsumi Aoyagi, Barry Rountree, Martin Schulz, David Lowenthal, Yasutaka Wada, Keiichiro Fukazawa, Masatsugu Ueda, et al. 2015. Analyzing and mitigating the impact of manufacturing variability in power-constrained supercomputing. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'15).* IEEE, 1–12.

[16] Yang Jiao, Heshan Lin, Pavan Balaji, and Wu-chun Feng. 2010. Power and performance characterization of computational kernels on the gpu. In *2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing.* IEEE, 221–228.

[17] Stephanie Labasan, Matthew Larsen, Hank Childs, and Barry Rountree. 2019. Power and Performance Tradeoffs for Visualization Algorithms. In *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS'19).* 325–334.

[18] James H Laros III, Kevin T Pedretti, Suzanne M Kelly, Wei Shu, and Courtenay T Vaughan. 2012. Energy based performance tuning for large scale high performance computing systems. In *Proceedings of the 2012 Symposium on High Performance Computing (HPC'12).* 1–10.

[19] Savoie Lee, David K. Lowenthal, Bronis R. De Supinski, Tanzima Islam, Kathryn Mohror, Barry Rountree, and Martin Schulz. 2016. I/O Aware Power Shifting. In *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS'16).* 740–749.

[20] Antonio Libri, Andrea Bartolini, and Luca Benini. 2018. DiG: Enabling out-of-band scalable high-resolution monitoring for data-center analytics, automation and control (extended). *Cluster Computing* (2018), 1–12.

[21] Matthias Maiterth, Gregory Koenig, Kevin Pedretti, Siddhartha Jana, Natalie Bates, Andrea Borghesi, Dave Montoya, Andrea Bartolini, and Milos Puzovic. 2018. Energy and Power Aware Job Scheduling and Resource Management: An In-depth Analysis. In *Workshop on Data-center Automation, Analytics, and Control (DAAC'18).*

[22] Bin Nie, Devesh Tiwari, Saurabh Gupta, Evgenia Smirni, and James H Rogers. 2016. A large-scale study of soft-errors on GPUs in the field. In *IEEE International Symposium on High Performance Computer Architecture (HPCA'16).* IEEE, 519–530.

[23] Bin Nie, Ji Xue, Saurabh Gupta, Christian Engelmann, Evgenia Smirni, and Devesh Tiwari. 2017. Characterizing temperature, power, and soft-error behaviors in data center systems: Insights, challenges, and opportunities. In *IEEE 25th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'17).* IEEE, 22–31.

[24] Bin Nie, Ji Xue, Saurabh Gupta, Tirthak Patel, Christian Engelmann, Evgenia Smirni, and Devesh Tiwari. 2018. Machine learning models for GPU error prediction in a large scale HPC system. In *48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'18).* IEEE, 95–106.

[25] Jorji Nonaka, Toshihiro Hanawa, and Fumiyoshi Shoji. 2020. Analysis of Cooling Water Temperature Impact on Computing Performance and Energy Consumption. In *IEEE International Conference on Cluster Computing (CLUSTER'20).* 169–175.

[26] George Ostrouchov, Don Maxwell, Rizwan Ashraf, Christian Engelmann, Mallikarjun Shankar, and James Rogers. 2020. GPU lifetimes on Titan supercomputer: Survival analysis and reliability. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (SC'20).* 15–20.

[27] Michael Ott and Dieter Kranzlmüller. 2018. Best Practices in Energy-Efficient High Performance Computing. In *48. Jahrestagung der Gesellschaft für Informatik, Architekturen, Prozesse, Sicherheit und Nachhaltigkeit, INFORMATIK 2018 - Workshops, Berlin, Germany, September 26-27, 2018 (LNI, Vol. P-285).* GI, 167–176.

[28] Michael Ott, Woong Shin, Norman Bourassa, Torsten Wilde, Stefan Ceballos, Melissa Romanus, and Natalie Bates. 2020. Global Experiences with HPC Operational Data Measurement, Collection and Analysis. In *IEEE International Conference on Cluster Computing (CLUSTER'20).* 499–508.

[29] Scott Pakin, Curtis Storlie, Michael Lang, Robert E. Fields III, Eloy E. Romero Jr., Craig Idler, Sarah Michalak, Hugh Greenberg, Josip Loncaric, Randal Rheinheimer, Gary Grider, and Joanne Wendelberger. 2016. Power usage of production supercomputers and production workloads. *Concurrency and Computation: Practice and Experience* 28, 2 (2016), 274–290.

[30] Tirthak Patel, Zhengchun Liu, Raj Kettimuthu, Paul Rich, William Allcock, and Devesh Tiwari. 2020. Job Characteristics on Large-Scale Systems: Long-Term Analysis, Quantification, and Implications. In *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'20)*. 1–17.

[31] Tirthak Patel, Adam Wagenhäuser, Christopher Eibel, Timo Hönig, Thomas Zeiser, and Devesh Tiwari. 2020. What does Power Consumption Behavior of HPC Jobs Reveal? : Demystifying, Quantifying, and Predicting Power Consumption Characteristics. In *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS'20)*. 799–809.

[32] Justin Thaler, Woong Shin, Steven Roberts, James Rogers, and Todd Rosedahl. 2020. Hybrid Approach to HPC Cluster Telemetry and Hardware Log Analytics. In *IEEE High Performance Extreme Computing Conference (HPEC'20)*.

[33] Devesh Tiwari, Saurabh Gupta, George Gallarno, Jim Rogers, and Don Maxwell. 2015. Reliability lessons learned from gpu experience with the titan supercomputer at oak ridge leadership computing facility. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'15)*. IEEE, 1–12.

[34] Devesh Tiwari, Saurabh Gupta, James Rogers, Don Maxwell, Paolo Rech, Sudharshan Vazhkudai, Daniel Oliveira, Dave Londo, Nathan DeBardeleben, Philippe Navaux, et al. 2015. Understanding GPU errors on large-scale HPC systems and the implications for system design and operation. In *IEEE 21st International Symposium on High Performance Computer Architecture (HPCA'15)*. IEEE, 331–342.

[35] Sean Wallace, Xu Yang, Venkatram Vishwanath, William E. Allcock, Susan Coghlan, Michael E. Papka, and Zhiling Lan. 2016. A Data Driven Scheduling Approach for Power Management on HPC Systems. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'16)*. 656–666.

[36] Torsten Wilde, Axel Auweter, and Hayk Shoukourian. 2014. The 4 Pillar Framework for energy efficient HPC data centers. http://dx.doi.org/10.1007/s00450-013-0244-6.

[37] Xingfu Wu, Valerie Taylor, Justin M Wozniak, Rick Stevens, Thomas Brettin, and Fangfang Xia. 2019. Performance, energy, and scalability analysis and improvement of parallel cancer deep learning candle benchmarks. In *Proceedings of the 48th International Conference on Parallel Processing*. 1–11.

[38] Edward Wyrwas. 2018. Body of knowledge for graphics processing units (GPUs). *Lentech Inc* (2018).

[39] Huazhe Zhang and Henry Hoffmann. 2019. PoDD: Power-Capping Dependent Distributed Applications. In *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'19)*. 1–22.

[40] Qi Zhu, Bo Wu, Xipeng Shen, Li Shen, and Zhiying Wang. 2017. Co-Run Scheduling with Power Cap on Integrated CPU-GPU Systems. In *2017 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 967–977.

Appendix: Artifact Description/Artifact Evaluation

SUMMARY OF THE EXPERIMENTS REPORTED

1 OVERVIEW

This work was focused on analyzing telemetry data and logs from a pre-exascale supercomputer. Due to the extreme size (8 TB) and the operational nature of the dataset, we were not able to publish the dataset for review. Instead, we submit the description of the datasets involved and the related data processing scripts and analysis files for review. If necessary, access to the telemetry dataset can be obtained via contacting ORNL staff members (shinw@ornl.gov).

2 TOOLS AND PACKAGES USED FOR DATA PRE-PROCESSING AND ANALYSIS

- Dask (https://dask.org/)
- Pandas (https://pandas.pydata.org/)
- PyArrow (https://arrow.apache.org/docs/python/)
- Numpy (https://numpy.org/)
- Scipy (https://www.scipy.org/)
- Matplotlib (https://matplotlib.org/)
- Seaborn (https://seaborn.pydata.org/)

3 RAW DATASET DESCRIPTION

3.1 Dataset A

- Name: Summit per node OpenBMC telemetry Per-node per component power and temperature measurements
- Files (type and quantity): one tar file per day that archives 1,440 parquet files per day, total 399 files (2019-12-27 to 2021-1-31) except missing dates due to maintenance periods
- Memory Footprint: 8.5 TB
- Source: Per-node OpenBMC data from Summit archived via the telemetry system for MTW operations
- Frequency: 1 sec
- Data of Creation / Last Update: one file per day from 2019-12-27 to 2021-1-31

3.2 Dataset B

- Name: Central energy plant (CEP) data
- Files (type and quantity): One parquet per month, 12 parquet files
- Memory Footprint: 256 MB
- Source: Control system of Summit's central energy plant via the telemetry system for MTW operations
- Frequency: Approx. 15 second interval
- Data of Creation / Last Update: One file per month, 2020-1-31 2021-01-31

3.3 Dataset C

- Name: Job Scheduler allocation history
- Files (type and quantity): Single csv file
- Memory Footprint: 285 MB
- Source: IBM CSM system via telemetry data store for Summit
- Frequency: At occurence
- Data of Creation / Last Update: 2021-2-28

3.4 Dataset D

- Name: Per node job scheduler allocation history
- Files (type and quantity): Single csv file
- Memory Footprint: 14 GB
- Source: IBM CSM system via telemetry data store for Summit
- Frequency: At occurence
- Data of Creation / Last Update: 2021-2-27

3.5 Dataset E

- Name: Nvidia GPU XID error log
- Files (type and quantity): Single csv file
- Memory Footprint: 50 MB
- Source: Per-node syslog data via telemetry data store for Summit
- Frequency: At occurence
- Data of Creation / Last Update: 2021-2-12

4 PRE-PROCESSED DATASET DESCRIPTION

4.1 Dataset 0

- Name: Summit per node OpenBMC telemetry 10-second aggregates
- 10-second aggregation of min, max, mean, std per-node OpenBMC telemetry data that measures node-wise, component-wise power and temperature.
- Script: andes-load-summit-power-temp-openbmc-init10s-agg.py
- Input: 1 sec interval Summit per node OpenBMC telemetry data
- Output: 10 second aggregates of Summit per node OpenBMC telemetry data - one parquet file per day
- Memory Footprint: 5.5 TB
- Index used: timestamp
- Key Columns: timestamp, input_power.[count, min, max, mean, std], p[0,1]_power.[count, min, max, mean, std], p[0,1]_gpu[0,1,2]_power.[count, min, max, mean, std], gpu[0,1,2,3,5]_[core,mem]_temp.[count, min, max, mean, std]

4.2 Dataset 1

- Name: Cluster-level power time-series
- The cluster-level power time-series data has aggregated cluster-level aggregated power values at every 10 seconds. For each timestamp, the power values are calculated by taking the sum of input power from all the nodes at that instance.
- Script: power_ts_job_ignorant.py

- Files (type and quantity): Power time series dataset with 10 seconds frequency. 1 parquet file for a day with 1 minute partition.
- Memory Footprint: 1.5 GB
- Index used: timestamp
- Key Columns: timestamp, count_inp, sum_inp, mean_inp, max_inp

4.3 Dataset 2

- Name: Cluster-level CPUs and GPUs component power time-series
- Cluster level CPU and GPU components are calculated by aggregating power values for every CPU and GPU component in a node.
- Script: power_ts_job_ignorant_component.py
- Files (type and quantity): CPU and GPU component power time series dataset with 10 seconds frequency. 1 parquet file for a day with 1 minute partition.
- Memory Footprint: 0.5 GB
- Index used: timestamp
- Key Columns: timestamp,mean_cpu_power, std_cpu_power,min_cpu_power, max_cpu_power,mean_gpu_power, std_gpu_power,max_gpu_power

4.4 Dataset 3

- Name: Job wise power time-series
- The dataset has a time-series of power values for every job. It is generated by combining node-level power consumption data and the job scheduler data, which contains the list of nodes on which job has run.
- Script: power_ts_job_aware.py
- Files (type and quantity): Power time-series and job scheduler time-series dataset each having one parquet file for a day.
- Memory Footprint: 49 GB
- Index used: allocation_id,timestamp
- Key Columns: allocation_id, timestamp, count_hostname, sum_inp,max_inp, mean_inp

4.5 Dataset 4

- Name: Job wise CPU and GPU components power time-series
- The data has time-series of CPU and GPU power consumption usage for every jobs. It is generated by combining node-level power consumption data and the job scheduler data, which contains the list of nodes on which job has run.
- Script: power_ts_job_aware_component.py
- Files (type and quantity): CPU and GPU components power time-series and job scheduler time-series dataset each having one parquet file for a day.
- Memory Footprint: 45 GB
- Index used: allocation_id
- Key Columns: allocation_id, timestamp, count_hostnamemean_cpu_power, std_cpu_power, max_cpu_power, cpu_nans, mean_gpu_power, std_gpu_power, max_gpu_power, gpu_nans

4.6 Dataset 5

- Name: Job-level power data
- The per-node job-level power allocated data contains aggregated power values for a job across its run-time.
- Script: power_job_aware.py
- Files (type and quantity): Aggregating power time-series data over its job run. The input dataset has csv files for each day and the output dataset also has csv files for each day.
- Memory Footprint: 14 GB
- Index used: allocation_id
- Key Columns: allocation_id, max_sum_inp, mean_sum_inp, begin_time, end_time

4.7 Dataset 6

- Name: Job-level CPU and GPU components power data
- The job-level aggregated power values for per-node CPU and GPU components for a job across its run-time.
- Script: power_job_aware_component.py
- Files (type and quantity): Aggregating CPU and GPU components power time-series data over its job run. The input dataset has csv files for each day and the output dataset also has csv files for each day.
- Memory Footprint: 200 MB
- Index used: allocation_id
- Key Columns: allocation_id, mean_mean_cpu_pwr, max_cpu_pwr, mean_mean_gpu_pwr, max_gpu_pwr, begin_time,end_time

4.8 Dataset 7

- Name: Job-level energy data
- The job-level energy data is calculated by aggregating the energy values consumed by each node of a job.
- Script: job_energy.py
- Files (type and quantity): The dataset has one parquet file for each day. We sum up energy values across the nodes on which job has run.
- Memory Footprint: 100 MB
- Index used: allocation_id
- Key Columns: allocation_id, energy, gpu_energy, num_nodes, num_gpus, begin_time, end_time, job_domain, account

4.9 Dataset 8

- Name: Thermal cluster-level time-series
- Each row corresponds to a 10-second time interval and contains the number of nodes with thermal measurements, the list of nodes and their GPUs that were hot, and the number of nodes in each temperature band, together with telemetrics for the cooling plant.
- Script: andes-thermal-cluster.py
- Files (type and quantity): 1 CSV file for each day.
- Memory Footprint: 1 GB
- Index used: timestamp
- Key Columns: hostname, any_nan

4.10 Dataset 9

- Name: Thermal cluster-level time series for component types
- Each row corresponds to a 10-second time interval and contains information about component temperature distribution across Summit, together with telemetrics for the cooling plant.
- Script: thermal-cluster-comptype.py
- Files (type and quantity): 1 CSV file for each day.
- Memory Footprint: 2 GB
- Index used: timestamp
- Key Columns: gpu_core.mean

4.11 Dataset 10

- Name: Thermal per-node job-level time series
- Each row corresponds to a 10-second time interval in a job and contains its number of nodes with thermal measurements, the list of nodes and their GPUs that were hot, and the number of nodes in each temperature band, together with telemetrics for the cooling plant.
- Script: andes-thermal-perjob-time.py
- Files (type and quantity): 1 CSV file for each day.
- Memory Footprint: 167 GB
- Index used: timestamp, allocation_id
- Key Columns: hostname, any_nan

4.12 Dataset 11

- Name: Thermal job-level time series for component types
- Each row corresponds to a 10-second time interval in a job and contains information about component temperature distribution across the job at this time, together with telemetrics for the cooling plant.
- Script: thermal-perjob-comptype.py
- Files (type and quantity): 1 CSV file for each day.
- Memory Footprint: 268 GB
- Index used: timestamp, allocation_id
- Key Columns: gpu_core.mean

4.13 Dataset 12

- Name: Summit cooling system and weather time-series
- Each row corresponds to a 10-second time interval and contains telemetrics for the cooling plant.
- Files (type and quantity): single parquet file
- Memory Footprint: 350 MB
- Index used: timestamp
- Key Columns: mtwst, mtwrt

4.14 Dataset 13

- Name: Main switch board meter data
- Power measurements at the main switch boards depicted in Figure 1-(c) in the period of 2021-01-14 2021-01-15
- Files (type and quantity): total 5 csv files, each per main switch board
- Dimension: 172,800 x 2
- Memory Footprint: 7.2MB x 4
- Index used: timestamp
- Key Columns: B5600_MSBMSB_ID_MTRs

5 ANALYTICS DESCRIPTIONS

5.1 Figure 3

- Filename: validation.ipynb
- Datasets: Main switch board meter data (Dataset 12), Per-node 10-second time series data
- Tools Used: pandas, dask, matplotlib, seaborn
- Primary Calculations Performed: Per-node 10-second time series data is joined with a node to MSB mapping that has been manually created from the floormap. Then a groupby summation per MSB was performed to produce 10-second mean power time-series data. This data was compared with 10-second averages of the MSB level measurements.
- Other Complimentary Calculations: N/A

5.2 Figure 4

- Filename: summit-pue-plot-clean.ipynb
- Datasets: Summit cooling system and weather time-series (Dataset 11)
- Tools Used: pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: 5 columns of the Summit cooling system and weather time-series data are summarized into weekly box plots over the year 2020. For the weekly power summaries, we also plot the maximum cluster-level power seen that week.
- Other Complimentary Calculations: We calculate the average PUE of 2020 and the average PUE during just the summer with Summit's chillers active.

5.3 Figure 5

- Filename: input_power_total_energy.ipynb
- Dataset: Job-level power data (Dataset 5)
- Tools Used: pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: The energy consumption of the jobs and maximum input power is an artifact of profiling jobs. The Gaussian kernel density plots show the distribution of input power and total energy across five classes.
- Other Complimentary Calculations: N/A

5.4 Figure 6

- Filename: boxplot_input_power_total_energy.ipynb
- Datasets: Job-level power data (Dataset 5), Job-level energy data (Dataset 7)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: The two leadership node count classes are compared over a variety of metrics: Number of Nodes in Job, Walltime of Job, Mean Power, Max Power, and (Mean - Max) Power Difference. Each of those are shown are cumulative density functions with the 80
- Other Complimentary Calculations: N/A

5.5 Figure 7

- Filename: boxplot_input_power_total_energy.ipynb
- Datasets: Job-level power data (Dataset 5), Job-level energy data (Dataset 7)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn

- Primary Calculations Performed: The two leadership node count classes are compared in both energy and max power. Results are further divided by OLCF project science domains and presented as boxplot distributions.
- Other Complimentary Calculations: N/A

5.6 Figure 8

- Filename: cpu-gpu.ipynb
- Datasets: Job wise CPU and GPU components per-node power (Dataset 6)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: Partition jobs into the 5 node count classes then produce four 2-dimensional kdeplots based on mean and maximum CPU power as 1 dimension, and GPU Power for the two leadership classes and the three smaller classes.
- Other Complimentary Calculations: N/A

5.7 Figure 9

- Filename: summit-edges-plot-clean.ipynb
- Datasets: Job wise power time-series (Dataset 3)
- Tools Used: pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: For each Summit job node-count class, we calculate 1.) The number of rising and falling edges per job where a rising/falling is at least a 4 MW change in power over a 10 second interval at full system scale. Jobs with fewer than 4626 nodes have the appropriately weighted power change threshold (e.g. a job with 2313 nodes requires a 2 MW change). A cumulative density function is then created. 2.) The duration of each previously identified rising or falling edge. A duration is defined as the time from the beginning of the edge till the return back 80% from its peak to its initial power. A cumulative density function is then created. 3 4.) The job-level power time-series that contain a rising or falling edges are differenced to find their 10 second power changes. This differenced job power time-series then has an FFT applied to it. The maximum amplitude and its corresponding frequency are then collected and a density function is created. Each job contributes a single amplitude and a single frequency.
- Other Complimentary Calculations: Section 1 determines the rising and falling edges over 4MW and presents the snapshots surrounding the edge. The steepest rise and fall over 10 seconds is determined to be 5.79 MW and -5.89 MW, respectively. There are 165 rising edges and 81 falling edges of greater than 4MW.

5.8 Figure 10

- Filename: power_dynamics_per_amp.ipynb
- Datasets: Job wise power time-series (Dataset 3), Summit cooling system and weather time-series (Dataset B11)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: For each Summit job, we find the rising edges (same definition as Figure 9 edges) for various amplitudes and the surrounding 5 minute time-series snapshots (1 minute before and 4 minutes following). We calculate the 95% confidence interval around the mean and display each amplitude sequentially. The numbers of various rising edge amplitudes are tallied and provided at the top of the figure. The snapshots are also used to find the corresponding PUE data within the Summit cooling system and weather time-series dataset. Those are also plotted with their 95
- Other Complimentary Calculations: A similar analysis is completed but only for the summer months.

5.9 Figure 11

- Filename: thermal_response.ipynb
- Datasets: Job wise power time-series (Dataset 3), Summit cooling system and weather time-series (Dataset B11)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: Taking just the 4MW, 6MW, and 7MW rising edges and adding the 7MW falling edges, we plot the corresponding mean and max component temperatures (both CPU and GPU), the supply and return temperatures of the MTW cooling system, and the tons of refridgeration (TOR) of the MTW system and the chillers. All data is taken from the same 5 minute snapshot (1 minute before and 4 minutes following) that surrounds the rising or falling edge.
- Other Complimentary Calculations: N/A

5.10 Figure 12

- Filenames: gpu-failures-per-project.ipynb, gpu-failures-correlation.ipynb
- Datasets: Per node job scheduler allocation history (Dataset D), Nvidia GPU XID error log (Dataset E)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: 1.) We cross examine the nodes of all GPU failure logs and their OLCF project. Different GPU failure types are tallied and the 15 most error prone projects are listed 2.) We cross examine the nodes of only the GPU hardware failure logs and their OLCF project. Different GPU failure types are tallied and the 15 most error prone projects are listed 3.) We count the GPU failures separately for every Summit node, and compute the Pearson correlation between the resulting 4,626-dimensional vectors for every pair of failure types. We show the correlation coefficients significant at 0.05 after applying the Bonferroni correction to account for the number of pairs.
- Other Complimentary Calculations: N/A

5.11 Figure 13

- Filename: gpu-failures-thermal.ipynb
- Datasets: Nvidia GPU XID error log (Dataset E), Thermal per-node job-level time series (Dataset 10)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: For each error type, we calculate the Z-score of the rank 0 GPU core temperature when the error occurs and density functions are created. We then plot the actual GPU core temperatures as density functions.

- Other Complimentary Calculations: N/A

5.12 Figure 14

- Filename: gpu-failures-spatial.ipynb
- Datasets: Nvidia GPU XID error log (Dataset E)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: GPU failure logs have their PCI addresses mapped to their physical locations in the node slot (0 - 5). Four error types are tallied for each of the 6 GPU slots and presented as a histogram.
- Other Complimentary Calculations: All error types are tallied for each of the 6 GPU slots. Distribution of NVlink errors for each of the 6 GPU slots and the outgoing link of the error composes each historgram bar. Distribution functions are created for each error type and their physical location in the Summit machine. Physical locations have three coordinates: Row on the Summit floor, cabinet within a row, and node height within a cabinet.

5.13 Figure 15

- Filename: component_variation.ipynb
- Datasets: Thermal per-node job-level time series (Dataset 10)
- Tools Used: dask, pandas, numpy, matplotlib, seaborn
- Primary Calculations Performed: We identify one key 4608 node job that lasts 7 minutes long. We show a play-by-play snapshot that present boxplots of both the individual GPU powers and temperatures along with their maximums. Six instants are further examined to visualize the distribution of GPU powers versus the temperatures for all GPUs participating in the job. Lastly, GPU core temperatures at the six instants are aggregating into racks and displayed as a heatmap looking down on the Summit floor layout. Both mean and maximum GPU temperatures are plotted. Missing racks are plotted in grey and racks not participating are plotted in bright green.
- Other Complimentary Calculations: Spread of the GPU core temperatures is 15.8 degrees C and the spread of the GPU power is 62.2 Watts.

Author-Created or Modified Artifacts:

```
Persistent ID: https://github.com/at-aaims/sc21\_sum⌋
↪  mit\_power\_analysis\_artifacts
Artifact name:
↪  sc21\_summit\_power\_analysis\_artifacts
```