Lessons Learned from Memory Errors Observed Over the Lifetime of Cielo

Scott Levy*, Kurt B. Ferreira*, Nathan DeBardeleben†, Taniya Siddiqua‡, Vilas Sridharan‡, Elisabeth Baseman†
*Center for Computing Research, Sandia National Laboratories
{sllevy,kbferre}@sandia.gov
† Ultrascale Systems Research Center, Los Alamos National Laboratory
{ndebard,lissa}@lanl.gov
‡ RAS Architecture, Advanced Micro Devices, Inc.
{taniya.siddiqua,vilas.sridharan}@amd.com

***Abstract*—Maintaining the performance of high-performance computing (HPC) applications as failures increase is a major challenge for next-generation extreme-scale systems. Recent work demonstrates that hardware failures are expected to become more common. Few existing studies, however, have examined failures in the context of the entire lifetime of a single platform. In this paper, we analyze a corpus of empirical failure data collected over the entire five-year lifetime of Cielo, a leadership-class HPC system. Our analysis reveals several important findings about failures on Cielo: (i) its memory (DRAM and SRAM) exhibited no aging effects; detectable, uncorrectable errors (DUE) showed no discernible increase over its five-year lifetime; (ii) contrary to popular belief, correctable DRAM faults are not predictive of future uncorrectable DRAM faults; (iii) the majority of system down events have no identifiable hardware root cause, highlighting the need for more comprehensive logging facilities to improve failure analysis on future systems; and (iv) continued advances will be needed in order for current failure mitigation techniques to be viable on future systems. Our analysis of this corpus of empirical data provides critical analysis of, and guidance for, the deployment of extreme-scale systems.**

I. Introduction

Maintaining the performance of high-performance computing (HPC) applications as failures become more frequent is a major challenge that needs to be addressed for next-generation extreme-scale systems. Recent studies have demonstrated that hardware failures are expected to become more common [1]. Increasing the scale of HPC systems requires the aggregation larger numbers of individual components. More components means more frequent failures. Current systems use powerful error-correcting codes (ECC), e.g., chipkill-correct, to protect against DRAM errors. However, chipkill-correct (and other similar techniques) require the activation of a large number of memory devices (four times more than less-protective techniques such as single error correct double error detect (SECDED)) [2]. Activating more memory devices requires more power for each memory access. However, because of tightening power budgets on next-generation systems [1], it is not yet clear that chipkill-correct will continue to be viable. Reduced device-feature sizes also have the potential to result in more frequent failures. Understanding the implications of these trends requires detailed knowledge of how failures affect current leadership-class systems.

In this paper, we analyze a corpus of empirical failure data collected over the entire five-year lifespan of Cielo, a leadership-class computing system. Our dataset consists of: (i) resource management logs that describe when nodes in the system go down and when they are brought back into service; (ii) detailed system logs containing information about all of the detected memory failures (DRAM and SRAM) that occurred in the system; (iii) logs of each kernel panic in the system; and (iv) detailed logs of the hardware devices installed on each compute node. Unlike existing studies of empirical failure data [3], [4], [5], [6], [7], [8], [9], [10], we analyze failures in the context of the entire lifetime of a single platform. Studying the entire lifetime of this machine allows us, for the first time, to answer the question of how age affects system reliability.

Given this corpus of failure data, we use several statistical techniques to study how failures occurred on Cielo. Our analysis reveals several important findings about failures on current and future systems:

- Cielo's memory (DRAM and SRAM) exhibited no aging effects: the rate of detectable, uncorrectable errors (DUE) showed no discernible increase over its five-year lifetime; and the correctable DRAM FIT rate showed a modest *decrease* over its lifetime (§III-D).
- Contrary to popular belief, correctable DRAM faults are not predictive of future uncorrectable DRAM faults. No correlation between these two fault modes was found over the entire lifetime of the system (§III-E).
- The majority of the system down events have no identifiable hardware root cause, highlighting the importance of developing more comprehensive and tightly integrated logging on future machines (§III-B).
- Important system design trade-offs will need to be made on next-generation systems for current failure mitigation techniques to remain efficient (§III-F).

This paper is, to the best of our knowledge, the first detailed analysis of correctable and uncorrectable memory errors over the entire lifetime of a leadership-class system. Based on our analysis, we provide insight into, and guidance for, the deployment of extreme-scale systems.

SC18, November 11-16, 2018, Dallas, Texas, USA
978-1-5386-8384-2/18/$31.00

II. Methodology

A. System Description

Cielo was a leadership-class HPC system located in Los Alamos, New Mexico. It was a Cray XE6 system running Linux that was operated from March 2011 to May 2016. At the time of its decommissioning, it was comprised of approximately 8,500 compute nodes. Each compute node contained 32 GB of DRAM and two processor sockets, each occupied by an AMD Opteron™ 8-core processor.

Cielo consisted of 96 *racks* of compute nodes arranged in 6 rows. Each rack contained 96 compute nodes arranged in a three-level hierarchy. Each rack was composed of three *chassis*. Each chassis was composed of eight *slots*. Each slot hosted four compute nodes.

B. Terminology: faults and errors

Throughout this paper, we distinguish between faults and errors, *cf.* [11]. A **fault** is the underlying cause of an error (e.g., stuck-at bits or high-energy particle strikes). An **error** is incorrect system state due to an active fault. Errors are *detected* and possibly *corrected* by higher-level mechanisms such as parity or error correcting codes (ECC). They may also be *uncorrected* or, in the worst case, *undetected*.

C. System Logs

Our analysis is based on information captured by Cielo over its lifetime. This subsection describes the four principal data sources we obtained from Los Alamos National Laboratory.

1) Resource Management Logs: The Application Level Placement Scheduler (ALPS) is designed to provide resource management services on Cray supercomputers. Effective management of computational resources requires ALPS to have accurate and up-to-date information about which compute nodes are *operational* (i.e., are available for use) and which are *down* (i.e., are unavailable for use). Each time that a compute node transitions into or out of the operational state, ALPS writes an entry to its log file. The corpus of ALPS logs collected over the lifetime of Cielo provides us with a detailed picture about the state of its compute nodes. We analyzed approximately two years, June 2014 to May 2016, of Cielo's resource management logs.

Compute nodes can enter the down state unexpectedly when an error occurs (e.g., an uncorrectable memory error, a kernel panic) or for reasons that are not directly related to machine reliability (e.g., scheduled downtime). The ALPS logs do not explicitly distinguish between these cases. We are primarily interested in identifying instances where compute nodes entered the down state because of uncorrectable errors. We assume that errors that cause a compute node to enter the down state are independent events. In other words, an error that causes a compute node to crash is independent of errors that cause other compute nodes to crash. As a result, when multiple compute nodes are in the down state simultaneously it is likely due to the failure of a shared resource (e.g., parallel file system, rack-level power supply). When large portions of the machine are simultaneously in the down state, it is likely due to administrative or facility-related issues.

The objective of our analysis is to examine the reliability of Cielo's compute nodes. Therefore, we want to isolate node down events in the ALPS logs that are due to node-level hardware failures and exclude node down events that are due to administrative or facility-related causes. We begin by temporally clustering down events in the ALPS logs. However, when multiple compute nodes fail due to a single event, the timestamps of the events in the ALPS logs are unlikely to be exactly the same. To address this issue, we cluster events that are temporally close together: within 60 seconds of each other. To reduce down events in our dataset that are due to system-level causes, we exclude clusters that contain more than five of the system's compute nodes since it is unlikely that they represent node-level failures. We believe that these efforts result in a dataset that is a more accurate representation of failures on Cielo.

2) Memory Failure Logs: All of the DRAM on Cielo is protected by chipkill-correct ECC. When the memory controller detects a memory error, it is designed to use ECC to correct the error. If it is able to correct the error, the error is recorded as a *correctable error* (CE). If it is unable to correct the error, the error is recorded as a *detected, uncorrectable error* (DUE). Correctable errors are recorded in registers provided by the x86 Machine Check Architecture (MCA) [12]. The contents of these registers are polled periodically and written to the console log. Uncorrectable errors are recorded in an event log after the node is rebooted. For both correctable and uncorrectable errors, detailed information about each error is recorded. This information includes the physical address where the error occurred and ECC syndrome data that describes the cause of the error. Decoding the recorded information about each error allows us to identify the physical location of each logged error. We examined the memory error logs collected on Cielo from May 2011 to May 2016.

3) Kernel Panic Logs: A Linux kernel panics when it encounters conditions that indicate that its internal state has been corrupted and continued correct operation of the kernel cannot be guaranteed. Causes of kernel panics include software bugs, device driver errors, and undetected hardware errors. Information gathered by the kernel due to a kernel panic is written to the system log. We analyzed kernel panic logging data collected from June 2014 to May 2016.

4) Hardware Inventory Logs: Hardware inventory logs record details about the hardware that is in use on the system at any given moment in time. They include detailed information about the configuration of each node's hardware, including the memory device manufacaturer of each DIMM. These logs allow us to compare the DRAM reliability across manufacturers. However, they lack sufficient detail to allow us to track the movement of individual DIMMs. System administrators may replace (or swap) DIMMs when they experience a DRAM DUE or when they consistently experience CEs.

D. A note on data presentation

Because some of the logs we analyzed contain confidential information, figures in this paper have been modified to avoid disclosure of protected information. Figures 7 and 8 have their axis values removed, the MTBF used in Figures 16a and 16b and TABLE III is not disclosed, and DRAM vendor names are anonymized. We therefore focus our analysis on temporal and spatial trends in the data rather than on the absolute values of performance metrics.

III. Results and Analysis

A. Analysis of DRAM Device Usage

The daily usage of Cielo's DRAM is shown in Fig. 1. The DRAM devices used on Cielo were produced by three manufacturers. Throughout this paper, we refer to them as Manufacturers A, B, and C. Each point on this figure shows how many millions of device-hours were recorded for a particular manufacturer on a single day. Days when the system was not operational are excluded. The periodic outliers above and below the bulk of the data for each manufacturer is a consequence of transitions into and out of daylight saving time. When daylight saving time begins in the spring it results in a day that is effectively 23 hours long. Similarly, when daylight saving time ends in the fall it results in a day that is effectively 25 hours long. As a result, we record more or fewer device-hours on these days. Additionally, the minimum device usage in our dataset is a single outlier in September 2013. This is due to an anomaly in the hardware inventory logs: for several hours on September 17, 2013, the hardware inventory recorded only two operational nodes.

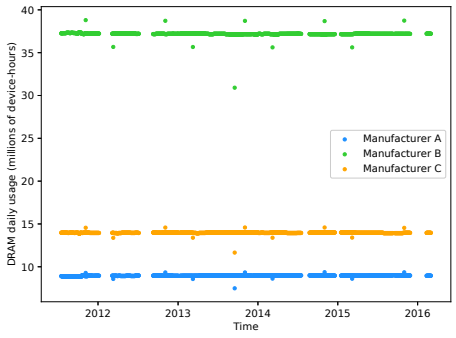


Fig. 1. **Daily DRAM Usage by Manufacturer**. Each point represents the number of device hours on a single day for each manufacturer.

The data in this figure show that per-manufacturer device usage was extremely stable over the lifetime of Cielo. For the purposes of this paper, temporal trends in machine behavior cannot be attributed to variations in DRAM device manufacturers. The total per-manufacturer share of DRAM device-hours is shown in Fig. 2. Although the per-day variation in device usage is low, the total DRAM device usage varies significantly per calendar year. Fig. 3 shows how many millions of DRAM device-hours were recorded over each calendar year. Nearly 70% of the device usage occurred during 2013, 2014, and 2015. Because the machine experienced several outage periods in early 2016 and was decommissioned in May 2016, the DRAM device usage for 2016 is a small fraction (less than 1.5%) of the total. As a result, variability in data collected during 2016 may be because it is based on a much smaller sample than the other years in our dataset.

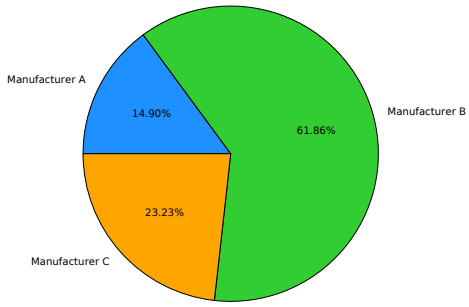


Fig. 2. **Total DRAM Usage by Manufacturer**.

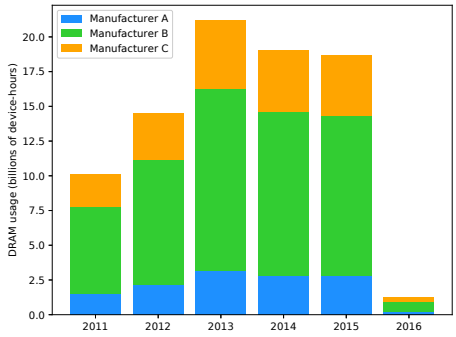


Fig. 3. **Total DRAM Usage by Year and Manufacturer**. Each colored region corresponds to the number of DRAM device-hours of operation for each of the three DRAM manufacturers.

B. Analysis of memory DUEs

To better understand the role that memory DUEs play in overall system reliability, we examined the resource manager logs and correlated node down events with the occurrence of memory DUEs.[1] The process of matching node down events to DUEs is inexact. Fig. 4 shows the results of our analysis. This figure divides the node down events into three categories based on their identified root cause. Memory DUEs were responsible for only a modest fraction ($\approx 27\%$) of all node down events. For the majority of the recorded node down events we were unable to definitively identify a root cause. This experience is not unique to Cielo; many studies have documented similar challenges with root cause analysis. Our experience with this analysis highlights the need for more tightly integrated logging infrastructure on future leadership-class systems.

To understand memory reliability on Cielo, we examined the temporal distribution of memory DUEs observed over the

[1]As described in Section II-C, the resource manager logs that we analyzed covered a fraction of Cielo's lifetime. As a result, we excluded memory errors that occurred outside of the interval covered by the resource manager logs from the root cause analysis described in this section.

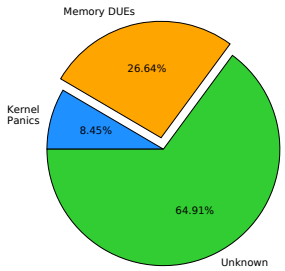


Fig. 4. **Root Causes of Node Down Events**.

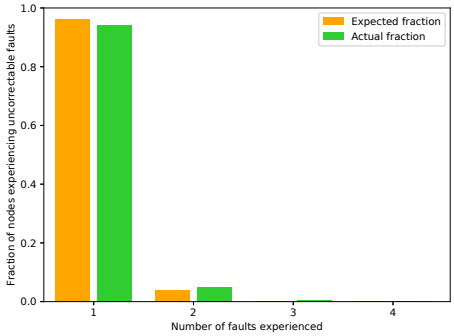


Fig. 5. **Uncorrectable Faults per Node**. This figure shows how DUEs are distributed among compute nodes that experienced one or more DUEs.

Distribution	Memory type		
	All memory	DRAM	SRAM
Exponential	-8852.1	-1397.6	-7704.4
Weibull	**-8839.3**	**-1378.3**	**-7698.4**
Gamma	-8842.2	**-1379.0**	-7706.3

TABLE I
VALUE OF THE BAYESIAN INFORMATION CRITERION (BIC) FOR THREE GROUPINGS OF DUE DATA. VALUES THAT CORRESPOND TO THE SELECTED MODEL (OR MODELS) SELECTED ARE HIGHLIGHTED IN BOLD.

lifetime of Cielo. We discovered several instances in which a long period of time elapsed between recorded errors. It is possible that these outliers represent periods of exceptionally reliable operation. However, we believe that it is more likely that they represent outages: periods when the machine was taken down for administrative reasons (e.g., software/hardware upgrades or repairs). We observe congruent episodes in the Resource Manager logs, however, we do not have access to appropriately annotated data that would allow us to definitively identify the administrative state of the machine during these long intervals when no errors were recorded. Therefore, we eliminated all of the fault-free intervals from our dataset that are more than three standard deviations from the mean. We identified three such periods. We have also consolidated uncorrectable faults that occurred multiple times within 30 seconds of each other on the same node into a single fault.[2] In the remainder of this paper, all of our analysis of the time between memory DUEs is performed on this reduced dataset.

To determine the temporal independence of these DUE events, we used the technique developed by Aupy *et al.* [13] for detecting failure cascades: periods when the density of failures is statistically unlikely. To determine the presence of failure cascades, the authors' approach computes the *lag plot ratio* of the sequence of observed failures. If the lag plot ratio is high ($\geq$ 4), the authors conclude that failure cascades are present. If the lag plot ratio is low ($\leq$ 2), then failure cascades are not present. Intermediate values indicate that failure cascades *may* be present. Using this approach, we calculated the lag plot ratio of three subsets of our DUE data. The lag plot ratios of the DRAM and SRAM faults, 1.82 and 1.31, respectively, indicate that there are no signs of failure cascades. However, the lag plot ratio of the combined dataset, 2.13, means that we cannot rule out the presence of failure cascades, i.e., there may be some temporal dependence in this dataset.

Given this dataset, Fig. 5 shows the distribution of memory DUEs across the nodes of the system that experienced one or more memory DUEs. This figure compares the number of faults that were observed on Cielo's compute nodes to the expected distribution. The expected distributed is computed by assuming that errors occur uniformly at random across all nodes of the system. The actual results match the expected results closely: the vast majority of compute nodes (94%) that experienced memory DUEs never experienced more than one.

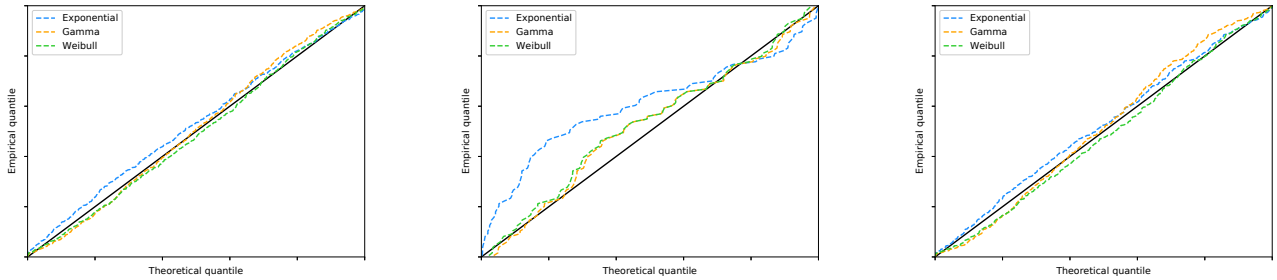
C. Distribution fitting of memory DUEs

In this subsection, we attempt to fit the time between memory DUEs to a statistical distribution. Although distribution fitting is inexact, identifying a mathematical model of how errors occur facilitates important modeling and forecasting research on how next-generation systems will perform.[3]

Fig. 6 shows quantile-quantile (Q-Q) plots that compare the distribution of the intervals between memory DUEs and three statistical distributions that are commonly used to model failures in large-scale systems: exponential, gamma, and weibull. In addition to fitting a distribution to the entire memory DUE dataset (Fig. 6a), we also consider the intervals between DRAM DUEs (Fig. 6b) and SRAM DUEs (Fig. 6c). The data in Figures (a) and (c) show that our empirical data fits all three distributions well; the differences between the distributions is small. This is because the mean and standard deviation of our empirical data are very nearly equal. Gamma and weibull distributions degenerate to an exponential distribution when the mean and standard deviation are equal. The DRAM DUE data indicate different behavior. The fit between our empirical data and the three distributions is noticeably different across the distributions. This is because the standard deviation of our empirical DRAM DUE data is much larger than the mean. The data in this figure suggest that the gamma distribution is the best fit of the DRAM inter-occurrence interval.

[2]Because uncorrectable faults generally require nodes to be rebooted, these faults almost certainly represent a single node-down event.

[3]Identifying a distribution that fits error data can have significant benefits. However, fine-grained decisions between similar statistical distributions may not be necessary to accurately model application performance [14].



(a) All Memory (b) DRAM (c) SRAM

Fig. 6. **Fitting Memory DUEs to Known Statistical Distributions.** Quantile-quantile (Q-Q)plots comparing the distribution of the inter-occurrence periods of memory DUEs on Cielo to well-known statistical distributions that are frequently used to characterize failures.

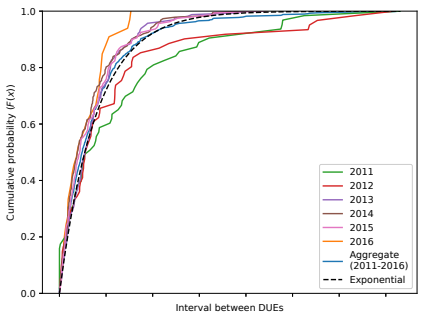


Fig. 7. **Empirical cumulative distribution function of memory DUEs over the lifetime of Cielo**. These data demonstrate that the statistical distribution of failures was relatively stable over the life of the system.

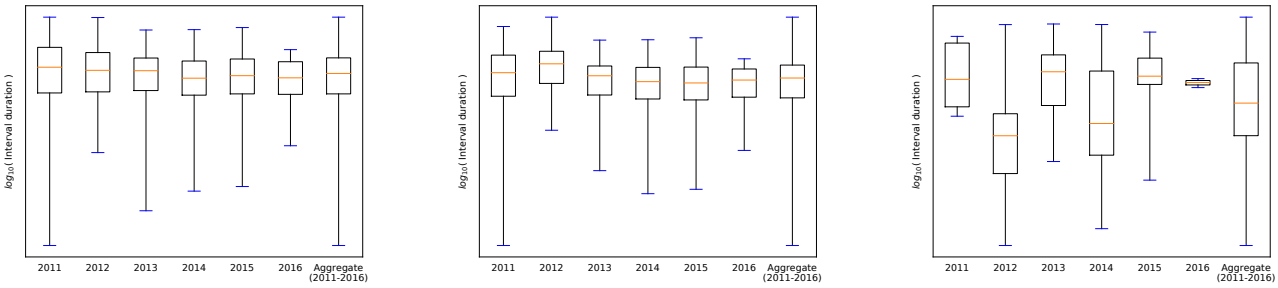
The Bayesian Information Criterion (BIC) provides a criterion for choosing between statistical models [15]. The value of the BIC for the three subsets of DUE data shown in Fig. 6 are shown in TABLE I. This approach selects the model with the largest BIC. For each of our DUE datasets, the Weibull distribution has the largest BIC value. To evaluate the relative strength of the evidence for choosing between these distributions, we apply the rules of thumb described by Raftery [16]. This approach considers the absolute difference between BIC values for different statistical models. For all memory DUEs, the evidence that the Weibull distribution is a better fit than the gamma is not particularly strong. However, there is very strong evidence that both are superior to the exponential distribution. For SRAM DUEs, there is strong evidence that Weibull is a better fit than either of the other distributions. For DRAM DUEs, there is not good evidence to support a choice between the gamma and Weibull distributions. However, there is strong evidence that both are superior to the exponential distribution.

D. Temporal stability of memory DUE events

Given that we have detailed logs collected over the entire lifetime of Cielo, we can study the impact of device age on memory DUEs. Fig. 7 shows the empirical cumulative density function (CDF) for data collected over each calendar year of Cielo's operation. This figure also includes the empirical CDF of the entire corpus of data and the theoretical distribution of the exponential distribution used in Fig. 6a. We observe that the memory DUE data that was collected from 2013-2015 fits the theoretical exponential very closely. In 2011 and 2012, the distribution was slightly more heavily-tailed than the exponential distribution. In other words, in these two years there were periods when the interval between memory DUEs was longer than the overall average. In 2016, for which we have only a partial year of data (*cf.* Fig. 3) the tail of the distribution is almost non-existent.

The distribution of the time intervals between memory DUEs is shown in Fig. 8. The boxplots in this figure describe the statistical distribution of these intervals for all memory structures (Fig. 8a), SRAM (Fig. 8b), and DRAM (Fig. 8a), for each year of Cielo's lifetime. Fig. 8a is another representation of the data shown in Fig. 7. The distribution of inter-occurrence intervals for SRAM DUEs is very stable from year-to-year. In contrast, the intervals between DRAM DUEs are much less stable. While there is fluctuation in these data, there is no discernible evidence to suggest that Cielo's DRAM was becoming less reliable over time. In fact, the median inter-occurrence interval was longer than average in 2015 and 2016.

Fig. 8 also shows that the intervals between DRAM DUEs were, on the whole, shorter in 2012 and 2014. Additional detail about the occurrence of DRAM faults is shown in Fig. 9. The data in this figure show the average fraction of DRAM faults per day over the lifetime of Cielo. We compute this fraction by dividing the number of DRAM faults that occurred on a single day by the total number of DRAM faults. To reduce noise and highlight temporal trends in the data, the daily average is computed using an exponentially weighted moving average [17, §6.4.3]. We selected the value of the *smoothing parameter* (α) for the moving average to minimize the sum of the squared errors. The orange line represents uncorrectable DRAM faults. The blue line represents correctable DRAM faults. These data show that the abnormally short intervals between DRAM DUEs observed in 2012 and 2014 were principally due to two short periods of time (one early in 2012 and one in the middle of 2014) in which DRAM DUEs were abnormally high. Based on the data that is currently available



(a) All Memory (b) SRAM (c) DRAM

Fig. 8. **Variation in the observed intervals between memory DUEs over the lifetime of Cielo**. The top and bottom of the boxes are the third and first quartiles of the data, respectively. The orange line represents the median and the whiskers range from the minimum to the maximum values in the dataset.

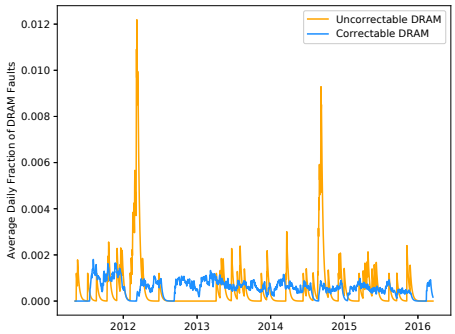


Fig. 9. **DRAM Faults Trend**. This figure shows the daily average fraction of DRAM faults over Cielo's lifetime. The daily average is computed by calculating the exponentially weighted moving average of the raw daily fraction of faults. The orange line represents the average fraction of uncorrectable DRAM faults per day. The blue line represents the average fraction of correctable DRAM faults per day.

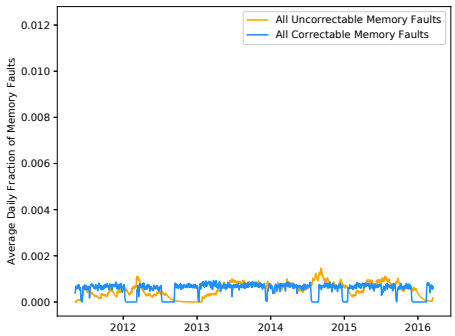


Fig. 10. **All Memory Faults Trend**. This figure shows the daily average fraction of all memory faults. To reduce noise and highlight long-term temporal trends, the daily average is computed by calculating the exponentially weighted moving average of the raw daily fraction of faults. The orange line represents the average fraction of uncorrectable faults that occurred per day. The blue line represents the average fraction of correctable faults per day.

to us, we cannot identify the root cause of these isolated spikes in DRAM DUEs. By way of comparison, Fig. 10 shows the same set of data for all memory faults. The orange line represents all uncorrectable memory faults. The blue line shows the monthly trends for all correctable memory faults. These data show that uncorrectable faults are more or less

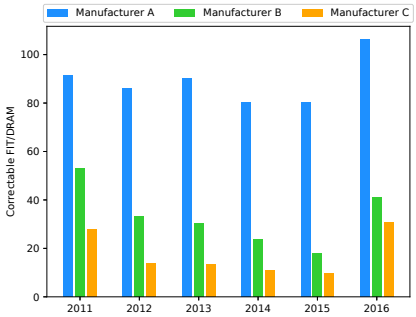


Fig. 11. **Correctable FIT Rate per DRAM Device**.

evenly distributed over the lifetime of Cielo.

Examining the occurrence of correctable DRAM faults provides additional information about how device aging impacted the reliability of Cielo's memory. The data in Fig. 11 show the failures-in-time (FIT)[4] rate for correctable DRAM faults for each DRAM device manufacturer over the lifetime of Cielo. These data show that, with the exception of 2016, there is an overall downward trend in the FIT rate; Cielo experienced fewer failures per hour of device operation near the end of its operational life than it did at the beginning. The data from 2016 *may* represent a change in this trend, but they should be carefully considered because they represent many fewer device-hours than the data from the other five years (*cf.* Fig. 3).

On the whole, these data indicate that there is no discernible trend that would indicate that Cielo's memory was becoming less reliable over its lifetime. This result is unexpected; the lifetime of processors is typically between five and seven years [18]. As a result, we would have expected to see aging effects on Cielo. Although decisions about machine decommissioning are complex and multi-factored, this result suggests that Cielo's memory may have had additional years left in its operational life.

[4]Failures-in-time is commonly used to describe hardware device reliability. It represents the number of failures, on average, that would be expected to occur in one billion (10^9) hours of device operation.

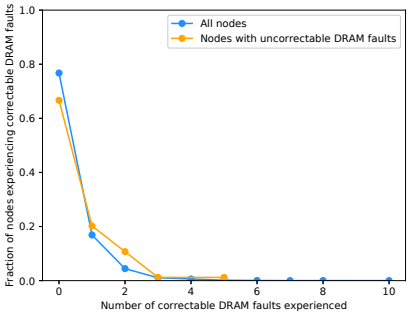


Fig. 12. **Correctable DRAM Faults per Node**. The blue points represent the fraction of all compute nodes that experienced a given number of correctable DRAM faults. The orange points represent the fraction of the compute nodes with one or more uncorrectable DRAM faults that experienced a given number of correctable DRAM faults. The lines connecting the points are provided only for readability.

	All Correctable Faults per Node		Correctable DRAM Faults per Node	
	All nodes	Nodes with DUE	All nodes	Nodes with DUE
1st quartile	6	6	0	0
Median	7	8	0	0
3rd quartile	9	10	0	1
Mean	7.54	7.74	0.33	0.54

TABLE II
STATISTICS COMPARING THE NUMBER OF CORRECTABLE FAULTS EXPERIENCED BY NODES THAT EXPERIENCED ONE OR MORE DUES TO THE NUMBER OF CORRECTABLE FAULTS EXPERIENCED BY ALL NODES.

E. Relationship between correctable and uncorrectable DRAM faults

Correctable DRAM faults may signal the initial stages of device failure and may presage a future uncorrectable DRAM fault.[5] In this subsection, we examine the relationship between correctable and uncorrectable DRAM faults on Cielo.

Fig. 12 shows how correctable DRAM faults are distributed across the compute nodes. The blue points represent the distribution for all compute nodes; the orange points represent the distribution for all nodes that have experienced one or more uncorrectable DRAM faults. Summary statistics comparing the distributions of these datasets are presented in TABLE II. These data show that the distribution of correctable DRAM faults across nodes that have experienced an uncorrectable DRAM faults is very similar to the distribution over all nodes. The average number of correctable DRAM faults is slightly higher (0.33 vs. 0.54) on nodes with uncorrectable DRAM faults, but the difference is small. The upshot is that controlling for the occurrence of uncorrectable DRAM faults does not skew the distribution of correctable DRAM faults; the data show that these nodes experience approximately the same number of correctable DRAM faults as if they were selected uniformly at random from all compute nodes. In other words, more frequent correctable DRAM faults do not appear to be correlated with the occurrence of uncorrectable DRAM faults.

[5]This is commonly the rationale used to support page offlining [19], [20]

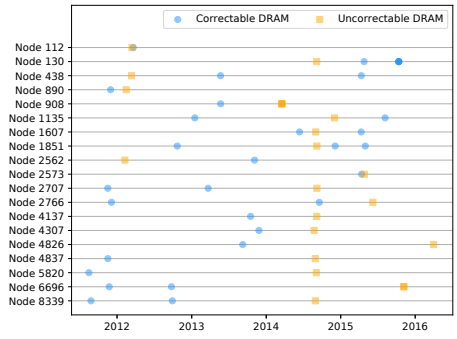


Fig. 13. Temporal arrangement of compute nodes that experienced uncorrectable DRAM faults preceded by by one or more correctable DRAM faults.

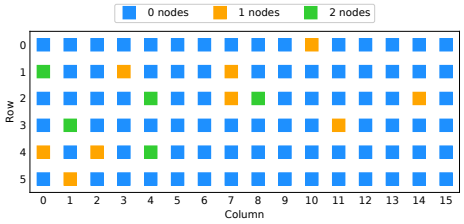


Fig. 14. Spatial arrangement of compute nodes that experienced uncorrectable DRAM faults preceded by by one or more correctable DRAM faults. The color of the squares represents the number of compute nodes from Fig. 13 that belong to each *rack* of Cielo.

If correctable DRAM faults presaged uncorrectable DRAM faults, we would expect to observe a sequence of correctable DRAM faults to occur in close temporal proximity to uncorrectable DRAM faults. Fig. 13 shows the temporal relationship between correctable and uncorrectable DRAM faults for all of the compute nodes that experienced one or more uncorrectable DRAM faults and one or more correctable DRAM faults. We have excluded those nodes for which all of the uncorrectable DRAM faults occurred before any of the correctable DRAM faults. Additionally, the majority of nodes that experienced uncorrectable DRAM faults experience zero correctable DRAM faults, *cf.* Fig. 12. These data show that were no instances on Cielo where a correctable DRAM fault is temporally related to a subsequent uncorrectable DRAM fault. In those cases where uncorrectable DRAM faults are preceded by correctable DRAM faults, the temporal lag is significant. Even the shortest interval (i.e., Node 2573) is nearly two weeks long.

Fig. 14 depicts the spatial arrangement of the nodes shown in Fig. 13. Each square represents one of the 96 racks that comprised Cielo (*see* Section II-A). The squares are color-coded to indicate the number of nodes in each rack that experienced an uncorrectable fault that was preceded by a correctable fault. Given the small number of nodes under consideration, it is difficult to draw statistically meaningful conclusions from this figure. However, it does appear that these nodes are disproportionately located in the first eight columns.

The data in Fig. 9 provides additional evidence that there is not a strong correlation between correctable and uncorrectable DRAM faults. This figure shows the average fraction of faults per day on Cielo. As discussed in Section III-D, there were

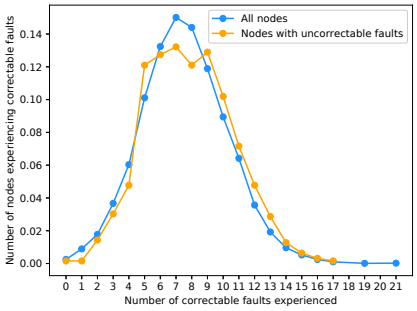


Fig. 15. **All Correctable Memory Faults per Node**. The blue points represent the fraction of all compute nodes that experienced a given number of correctable faults. The orange points represent the fraction of the compute nodes with one or more uncorrectable faults that experienced a given number of correctable memory faults. The lines connecting the points are provided only for readability.

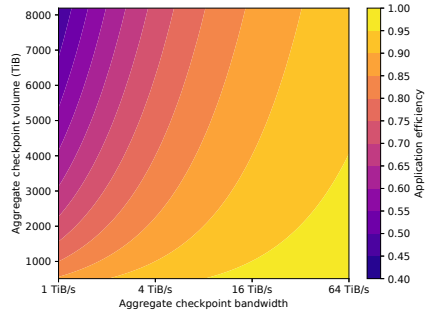
two spikes in uncorrectable DRAM faults: one in early 2012, and one in the middle of 2014. However, the data in Fig. 9 show that there was no corresponding spike in the number of correctable DRAM faults. Statistical analysis bears this out: the Pearson correlation coefficient [21] for the number of correctable and uncorrectable DRAM faults per day is: $r = -0.05$ [6]. Similarly, the data in Fig. 9 also demonstrate that there is not a strong correlation between correctable and uncorrectable faults from all sources. The Pearson correlation coefficient for these two temporal sequences is: $r = 0.03$.

Performing the same analysis on all correctable faults yields a similar conclusion. Fig. 15 shows how many correctable faults occurred on each compute node. The blue points represent the number of correctable faults detected on all compute nodes. The orange points represent the number of correctable faults that were detected on compute nodes that also experienced one or more correctable faults. This figure shows that the shapes of the distribution are very similar. A more detailed comparison of the distributions of these two datasets is presented in TABLE II. This table contains summary statistics for the two distributions. The quartiles of the distributions are very similar; these data suggest that nodes that have experienced one or more uncorrectable faults may experience slightly more correctable faults, but the mean number of correctable errors (7.54 for all nodes vs. 7.74 for nodes that have experienced correctable faults) show that any difference between the two datasets is small. These data suggest that the rate that nodes with uncorrectable faults experience correctable faults is essentially indistinguishable from the rate that all nodes experience correctable faults; the number of correctable faults that occur on a node is not strongly correlated with the occurrence of uncorrectable faults.

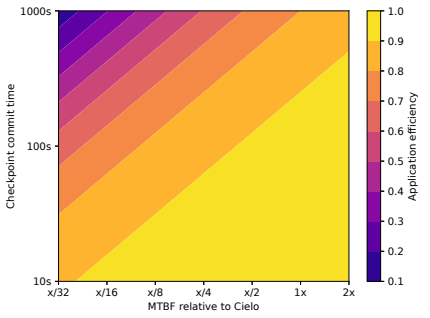
Although we have shown that there was not a strong correlation between correctable and uncorrectable faults, our analysis is subject to the DIMM replacement policy on Cielo, *see* Section II-C4. The replacement policy means that it is possible that a DIMM that was replaced in response to a correctable fault avoided a subsequent uncorrectable fault, thus concealing correlated faults from our analysis. Although we do not have precise data on the occurrence of DIMM replacements, we can establish on upper bound on the number of DIMMs that were replaced due to correctable faults. We know that correctable faults led to replacement only when the fault was the result of multiple errors. As a result, we know that no more than 1.9% of Cielo's DIMMs could have been replaced in response to a correctable fault.

Based on this analysis, we can draw one of two possible conclusions: either (i) the replacement policy was perfect, all correctable faults that signaled the imminent occurrence of an uncorrectable fault led to the replacement of the DIMM before the uncorrectable fault occurred; or (ii) correctable faults are not strongly correlated with the occurrence of subsequent uncorrectable faults. In either case, we can say that correctable faults were not reliable predictors of the uncorrectable faults that actually occurred during Cielo's lifetime.

[6]Coefficients near 1 denote a total positive linear correlation, 0 is no linear correlation, and -1 is total negative linear correlation.



(a) Application efficiency vs. checkpoint bandwidth and checkpoint volume



(b) Application efficiency vs. system MTBF and checkpoint commit time

Fig. 16. Implications of Cielo's reliability on the application efficiency of future systems.

F. Implications for fault tolerance

Understanding the reliability of Cielo's memory over its lifetime allows us to consider how fault tolerance may impact performance on next-generation systems. This section examines the performance of a hypothetical HPC application on a system with a failure rate equal to the rate measured over Cielo's lifetime. Our idealized application uses a state-of-the-

art coordinated checkpoint/restart library [22], [23] to ensure application progress across DUE. We use the Young/Daly model of application execution [24], [25] with failures and checkpointing, to calculate application efficiency on a next-generation system. This model assumes that uncorrectable faults are derived from an exponential distribution. In Fig. 6, we demonstrate that this is a reasonable assumption for Cielo. Application efficiency is the fraction of an application's runtime that is used to perform useful work. Therefore, an efficiency of 90% means only 10% of an application's time-to-solution is used for fault tolerance activities (*e.g.* checkpointing, restarting). Hard targets for application efficiency have not been established, but we believe that checkpoint/restart will remain viable if application efficiency remains above 80%.

In Fig. 16a, we examine how the relationship between aggregate checkpoint volume and the aggregate checkpoint bandwidth affect application efficiency. For the purposes of this figure, we examine the (optimistic) case where a next-generation machine will experience failures at the same rate as Cielo. Because next-generation machines will be much larger, this assumption means that the decrease in reliability due to scale will be entirely offset by advances in device reliability. For reference, Cielo supported approximately 256 TiB of DRAM and its aggregate filesystem bandwidth was 160 GB/s. Additionally, the parallel filesystem bandwidth of current leadership-class machines is greater than 1 TB/s (*see e.g.*, Trinity parallel filesystem bandwidth, 1.45 TiB/s, and aggregate burst buffer bandwidth, 3.3 TiB/s [26]). This figure demonstrates that if we can maintain the memory failure rate observed on Cielo and filesystem bandwidths continue to grow, application efficiency would exceed 80% even if the total checkpoint volume exceeds 1 PiB.

Assuming that the decrease in reliability associated with larger machines will be entirely offset by increased reliability due to technological advances in device design and fabrication is perhaps too optimistic. In Fig. 16b, we examine how the relationship between checkpoint commit time and system reliability affects application efficiency. In this figure, reliability is expressed in relation to Cielo. On the far right of this figure we consider the most optimistic case: a system that is even more reliable than Cielo despite its increased size. The rest of the figure shows how an application might perform on a next-generation system that is less reliable than Cielo. To keep application efficiency above 80%, the checkpoint commit time needs to be kept below approximately one minute or we need to develop technology that will offset a significant portion of the reliability impact of building larger and larger systems.

In TABLE III, we further consider the implications of Cielo's reliability for next-generation systems. For this analysis, the relevant parameters for these systems are: total volume of system memory, checkpoint volume, and memory reliability. We express the checkpoint volume as a fraction of system memory and base our values on the work of Lujan et al. [27]. We consider three different memory reliability scenarios: (i) *fixed system reliability*, an optimistic scenario in which a future system is as reliable as Cielo despite being much larger; (ii) *fixed device reliability*, a more pessimistic scenario in which the number of failures per byte of memory does not change, but the total volume of memory increases significantly; and (iii) *SEC-DED ECC*, a scenario in which the decrease in reliability of a future system is due to the fact that memory is protected by SEC-DED ECC instead of Chipkill-correct. The rate of uncorrected errors in memory protected by SEC-DED ECC has been shown to be 42× greater than for Chipkill [28]. To highlight the impact of memory reliability, we fix the reliability of all other system components.

	Optimistic Strawman	Pessimistic Strawman
System memory	4 PiB	8 PiB
Checkpoint volume (fraction of total memory[27])	0.25	0.75
Viable Checkpoint Bandwidth (fixed system reliability)	> 944 GiB/s	> 5 TiB/s
Viable Checkpoint Bandwidth (fixed reliability per byte)	> 6 TiB/s	> 140 TiB/s
Viable Checkpoint Bandwidth (SEC-DED ECC)	> 64 TiB/s	> 384 TiB/s

TABLE III
CONSIDERING THE IMPLICATIONS OF CIELO'S RELIABILITY ON NEXT-GENERATION SYSTEMS. VIABLE CHECKPOINT BANDWIDTH IS THE BANDWIDTH TO STORAGE REQUIRED TO KEEP APPLICATION EFFICIENCY ABOVE 80%. THE FEASIBILITY OF EACH BANDWIDTH IS INDICATED BY COLOR: GREEN (ACHIEVABLE IN THE NEAR FUTURE); ORANGE (WITHIN EXASCALE PROJECTIONS); AND RED (BEYOND CURRENT PROJECTIONS).

Given these parameters, we consider two possible configurations of a next-generation system: a *pessimistic* configuration in which the volume of system memory is large and the volume of checkpoint data is a large fraction of total system memory; and an *optimistic* configuration in which the volume of system memory is relatively small and the volume of checkpoint data is a modest fraction of total system memory. Although these targets represent a significant increase over Cielo, they still fall short of what is projected for exascale systems, *cf.* [29]. We then determine the checkpoint bandwidth[7] necessary to achieve at least 80% application efficiency for our three scenarios. The results in TABLE III show that in the optimistic case, it may be possible to offset decreases in memory reliability with increases in checkpoint bandwidth for all of the reliability scenarios. In the SEC-DED ECC scenario, achieving an aggregate checkpoint bandwidth of 64 TiB/s may be challenging, but some have forecasted that it may be possible to deploy a parallel file system with this performance in the relatively near future, *cf.* [30] (projecting that an exascale system might include a file system with ≈ 60TiB/s of aggregate bandwidth). For the pessimistic strawman, even relatively modest declines in memory reliability would require checkpoint bandwidths that are likely infeasible in the near future. As a result, the viability of checkpoint/restart on next-generation systems will likely depend on minimizing the volume of checkpoint data that applications require for restart.

[7]Checkpoint bandwidth is the rate at which checkpoints can be written to some form of stable storage (e.g., a parallel filesystem, burst buffers).

IV. Lessons learned

In this section, we share the lessons that we have learned from analyzing the failure data that was collected over the lifetime of Cielo and discuss the implications of the data for next-generation systems.

No discernible aging effects were observed*:* As components on Cielo aged, we expected them to become less and less reliable [18]. However, we observed no discernible aging effects on Cielo. Our data shows that the memory structures encountered memory DUEs at roughly the same rate at the end of the system's life as they did at the beginning.

Correctable DRAM faults were not reliable predictors of uncorrectable DRAM faults*:* Correctable DRAM faults have long been thought to be indicators of DRAM device defects that will manifest as uncorrectable DRAM faults in the future, *cf.* [19], [20]. However, the analysis in this paper shows that there is no meaningful relationship between correctable and uncorrectable DRAM faults. On Cielo a significant fraction of uncorrectable DRAM faults occurred on nodes that had not yet experienced a single DRAM fault. Even in those few instances where an uncorrectable DRAM fault was preceded by a correctable DRAM fault, the temporal distance between the two events was large, in most cases many months.

Failure analysis is challenging*:* We made several observations about the challenges we faced in trying to analyze failures on Cielo. For example, we lacked sufficient information to identify the cause of node down events in the Resource Manager logs. Moreover, to the extent that we were able to identify the root cause, the process of correlating events across different log files tedious and challenging. Based on our experience with Cielo, we believe that effective failure analysis requires the following.

Unified data storage. On Cielo, failure data was collected in different log files that were maintained by different software entities. A unified data storage service would reduce (or eliminate) some of the challenges associated with reconciling data stored in different files and formats.

Proactive RAS services. When software component detects a failure (e.g., the scheduler determines that one or more nodes are no longer available), it is important to gather information from other system components to provide context even if those components have not yet detected abnormal behavior. Relying on individual components to detect abnormal behavior may not be sufficient to understand the source of a failure.

Data Analytics Tools. The volume of failure data that is collected over the lifetime of a leadership-class machine can be significant. Tools that are designed to extract meaning from large data sets (e.g., Apache Spark) may provide additional insight into how leadership-class systems fail.

Efficient failure mitigation through checkpoint/restart will require continued advancement*:* If system reliability remains constant, current theoretical checkpoint bandwidths are sufficient to achieve greater than 80% application efficiency. However, if reliability decreases as expected (e.g., due to increased scale and/or power constraints), the viablity of checkpoint/restart will depend on advancements to minimize the time required to commit a checkpoint.

V. Related Work

The study of failures in production systems has been an active research topic for over a decade [31], [32], [33], [34], [35], [36], [37], [38], [39]. Failures have been studied in HPC systems [3], [10], [7] and commercial data centers [5], [4], [40], [19]. The circumstances under which DRAM devices fail have also been studied [41], [40], [9].

Siddiqua *et al.* [42] presented a study demonstrating that the incidence of each DRAM correctable fault *mode* on Cielo was stable over time. Gupta *et al.* [43] studied five vastly different systems of varying sizes and hardware and software configurations to discover common failure trends that are across HPC systems. The data set covering the longest period of operation that they considered was collected on the Jaguar XT4 system from 2008-2011.

Our work is distinct from these existing studies in several important ways. First, we analyze data from a recently decommissioned system; because it is a recent system it more accurately represents current systems than older systems that have been studied. Second, we examine the corpus of failure data from the *entire* lifetime of a leadership-class HPC system. This allows us to provide a detailed study of hardware aging effects on both SRAM and DRAM. It also allows us to analyze the performance of failure mitigation, in this case checkpoint/restart, to gain insight on current and future systems. Finally, to the best of our knowledge, this is the first study to examine the relationship between correctable and uncorrectable DRAM faults.

VI. Conclusion

Deploying and using the first Exascale system will require a detailed understanding of how failures occur. In this paper, we provide a detailed analysis of failure data collected over the entire lifetime of Cielo, a recent leadership-class HPC system. The results of our analysis provide novel insight into how failures and fault tolerance will affect application performance on current and future systems.

Acknowledgment

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

AMD, the AMD Arrow logo, AMD Opteron, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

REFERENCES

[1] K. Bergman, S. Borkar, D. Campbell, W. Carlson, W. Dally, M. Denneau, P. Franzon, W. Harrod, J. Hiller, S. Karp, S. Keckler, D. Klein, R. Lucas, M. Richards, A. Scarpelli, S. Scott, A. Snavely, T. Sterling, R. S. Williams, and K. Yelick, "Exascale computing study: Technology challenges in achieving exascale systems, Peter Kogge, editor & study lead," 2008.

[2] X. Jian, H. Duwe, J. Sartori, V. Sridharan, and R. Kumar, "Low-power, low-storage-overhead chipkill correct via multi-line error correction," in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, ser. SC '13. New York, NY, USA: ACM, 2013, pp. 24:1–24:12. [Online]. Available: http://doi.acm.org/10.1145/2503210.2503243

[3] B. Schroeder and G. A. Gibson, "A large-scale study of failures in high-performance computing systems," in *Dependable Systems and Networks (DSN 2006)*, Philadelphia, PA, June 2006.

[4] X. Li, K. Shen, M. C. Huang, and L. Chu, "A memory soft error measurement on production systems," in *2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference*, ser. ATC'07. Berkeley, Calif., USA: USENIX Association, 2007, pp. 21:1–21:6. [Online]. Available: http://dl.acm.org/citation.cfm?id=1364385.1364406

[5] B. Schroeder, E. Pinheiro, and W.-D. Weber, "DRAM errors in the wild: a large-scale field study," *Commun. ACM*, vol. 54, no. 2, pp. 100–107, Feb. 2011. [Online]. Available: http://doi.acm.org/10.1145/1897816.1897844

[6] X. Li, M. C. Huang, K. Shen, and L. Chu, "A realistic evaluation of memory hardware errors and software system susceptibility," in *Proceedings of the 2010 USENIX conference on USENIX annual technical conference*, ser. USENIXATC'10. Berkeley, Calif., USA: USENIX Association, 2010, pp. 6–20. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855840.1855846

[7] A. A. Hwang, I. A. Stefanovici, and B. Schroeder, "Cosmic rays don't strike twice: understanding the nature of DRAM errors and the implications for system design," in *Proceedings of the 17th international conference on Architectural Support for Programming Languages and Operating Systems*, ser. ASPLOS XVII. New York, NY, USA: ACM, 2012, pp. 111–122. [Online]. Available: http://doi.acm.org/10.1145/2150976.2150989

[8] T. Siddiqua, A. Papathanasiou, A. Biswas, and S. Gurumurthi, "Analysis of memory errors from large-scale field data collection," in *Silicon Errors in Logic - System Effects (SELSE), 2013 IEEE Workshop on*, 2013.

[9] V. Sridharan, J. Stearley, N. DeBardeleben, S. Blanchard, and S. Gurumurthi, "Feng shui of supercomputer memory: Positional effects in DRAM and SRAM faults," in *Proceedings of SC13: International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '13. New York, NY, USA: ACM, 2013, pp. 22:1–22:11. [Online]. Available: http://doi.acm.org/10.1145/2503210.2503257

[10] C. Di Martino, Z. Kalbarczyk, R. K. Iyer, F. Baccanico, J. Fullop, and W. Kramer, "Lessons learned from the analysis of system failures at petascale: The case of Blue Waters," in *International Conference on Dependable Systems and Networks*, 2014.

[11] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *Dependable and Secure Computing, IEEE Transactions on*, vol. 1, no. 1, pp. 11–33, 2004.

[12] "AMD64 architecture programmer's manual volume 2: System programming, revision 3.23," http://developer.amd.com/wordpress/media/2012/10/24593_APM_v21.pdf, 2013.

[13] G. Aupy, Y. Robert, and F. Vivien, "Assuming failure independence: Are we right to be wrong?" in *2017 IEEE International Conference on Cluster Computing (CLUSTER)*, Sept 2017, pp. 709–716.

[14] S. Levy and K. B. Ferreira, "An examination of the impact of failure distribution on coordinated checkpoint/restart," in *Proceedings of the ACM Workshop on Fault-Tolerance for HPC at Extreme Scale, FTXS@HPDC 2016, Kyoto, Japan, May 31, 2016*, 2016, pp. 35–42. [Online]. Available: http://doi.acm.org/10.1145/2909428.2909430

[15] G. Schwarz *et al.*, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[16] A. E. Raftery, "Bayesian model selection in social research," *Sociological methodology*, pp. 111–163, 1995.

[17] N. Sematech, "Nist/sematech e-handbook of statistical methods," *NIST SEMATECH*, 2013. [Online]. Available: https://www.itl.nist.gov/div898/handbook/index.htm

[18] P. Ramachandran, S. V. Adve, P. Bose, and J. A. Rivers, "Metrics for architecture-level lifetime reliability analysis," in *ISPASS 2008 - IEEE International Symposium on Performance Analysis of Systems and software*, April 2008, pp. 202–212.

[19] J. Meza, Q. Wu, S. Kumar, and O. Mutlu, "Revisiting memory errors in large-scale production data centers: Analysis and modeling of new trends from the field," in *2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, June 2015, pp. 415–426.

[20] D. Tang, P. Carruthers, Z. Totari, and M. W. Shapiro, "Assessment of the effect of memory page retirement on system ras against hardware faults," in *International Conference on Dependable Systems and Networks (DSN'06)*, June 2006, pp. 365–370.

[21] K. Pearson and L. N. G. Filon, "Mathematical contributions to the theory of evolution. IV. on the probable errors of frequency constants and on the influence of random selection on variation and correlation," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 191, pp. 229–311, 1898. [Online]. Available: http://www.jstor.org/stable/90745

[22] A. Moody, G. Bronevetsky, K. Mohror, and B. R. d. Supinski, "Design, modeling, and evaluation of a scalable multi-level checkpointing system," in *2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, Nov 2010, pp. 1–11.

[23] L. Bautista-Gomez, S. Tsuboi, D. Komatitsch, F. Cappello, N. Maruyama, and S. Matsuoka, "Fti: High performance fault tolerance interface for hybrid systems," in *2011 International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, Nov 2011, pp. 1–12.

[24] J. T. Daly, "A higher order estimate of the optimum checkpoint interval for restart dumps," *Future Gener. Comput. Syst.*, vol. 22, no. 3, pp. 303–312, 2006.

[25] J. W. Young, "A first order approximation to the optimum checkpoint interval," *Communications of the ACM*, vol. 17, no. 9, pp. 530–531, 1974.

[26] LANL, "Trinity Technical Specifications," http://www.lanl.gov/projects/trinity/specifications.php, Jan. 10 2017.

[27] J. Lujan *et al.*, "Apex workflows," Technical report, LANL, NERSC, SNL, Tech. Rep. LA-UR-15-29113, 2015. [Online]. Available: https://www.nersc.gov/assets/apex-workflows-v2.pdf

[28] V. Sridharan and D. Liberty, "A study of DRAM failures in the field," in *High Performance Computing, Networking, Storage and Analysis (SC), 2012 International Conference for*. IEEE, 2012, pp. 1–11.

[29] V. Sridharan, N. DeBardeleben, S. Blanchard, K. B. Ferreira, J. Stearley, J. Shalf, and S. Gurumurthi, "Memory errors in modern systems: The good, the bad, and the ugly," *ACM SIGARCH Computer Architecture News*, vol. 43, no. 1, pp. 297–310, 2015.

[30] J. Shalf, S. Dosanjh, and J. Morrison, "Exascale computing technology challenges," in *International Conference on High Performance Computing for Computational Science*. Springer, 2010, pp. 1–25.

[31] N. El-Sayed and B. Schroeder, "Reading between the lines of failure logs: Understanding how HPC systems fail," in *2013 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, June 2013, pp. 1–12.

[32] D. Tiwari, S. Gupta, G. Gallarno, J. Rogers, and D. Maxwell, "Reliability lessons learned from gpu experience with the titan supercomputer at oak ridge leadership computing facility," in *SC15: International Conference for High Performance Computing, Networking, Storage and Analysis*, Nov 2015, pp. 1–12.

[33] D. Tiwari, S. Gupta, J. Rogers, D. Maxwell, P. Rech, S. Vazhkudai, D. Oliveira, D. Londo, N. DeBardeleben, P. Navaux, L. Carro, and A. Bland, "Understanding gpu errors on large-scale hpc systems and the implications for system design and operation," in *2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)*, Feb 2015, pp. 331–342.

[34] A. Gainaru, F. Cappello, and W. Kramer, "Taming of the shrew: Modeling the normal and faulty behaviour of large-scale HPC systems," in *2012 IEEE 26th International Parallel and Distributed Processing Symposium*, May 2012, pp. 1168–1179.

[35] Y. Liang, Y. Zhang, A. Sivasubramaniam, M. Jette, and R. Sahoo, "Bluegene/l failure analysis and prediction models," in *International Conference on Dependable Systems and Networks (DSN'06)*, June 2006, pp. 425–434.

[36] Y. Liang, Y. Zhang, A. Sivasubramaniam, R. K. Sahoo, J. Moreira, and M. Gupta, "Filtering failure logs for a BlueGene/L prototype," in *2005 International Conference on Dependable Systems and Networks (DSN'05)*, June 2005, pp. 476–485.

[37] B. Nie, D. Tiwari, S. Gupta, E. Smirni, and J. H. Rogers, "A large-scale study of soft-errors on GPUs in the field," in *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, March 2016, pp. 519–530.

[38] A. Patwari, I. Laguna, M. Schulz, and S. Bagchi, "Understanding the spatial characteristics of DRAM errors in HPC clusters," in *Proceedings of the 2017 Workshop on Fault-Tolerance for HPC at Extreme Scale*, ser. FTXS '17. New York, NY, USA: ACM, 2017, pp. 17–22. [Online]. Available: http://doi.acm.org/10.1145/3086157.3086164

[39] S. Gupta, D. Tiwari, C. Jantzi, J. Rogers, and D. Maxwell, "Understanding and exploiting spatial properties of system failures on extreme-scale HPC systems," in *Proceedings of the 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, ser. DSN '15. Washington, DC, USA: IEEE Computer Society, 2015, pp. 37–44. [Online]. Available: http://dx.doi.org/10.1109/DSN.2015.52

[40] N. El-Sayed, I. A. Stefanovici, G. Amvrosiadis, A. A. Hwang, and B. Schroeder, "Temperature management in data centers: why some (might) like it hot," in *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '12. New York, NY, USA: ACM, 2012, pp. 163–174. [Online]. Available: http://doi.acm.org/10.1145/2254756.2254778

[41] V. Sridharan and D. Liberty, "A study of DRAM failures in the field," in *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, ser. SC '12. Los Alamitos, Calif., USA: IEEE Computer Society Press, 2012, pp. 76:1–76:11. [Online]. Available: http://dl.acm.org/citation.cfm?id=2388996.2389100

[42] T. Siddiqua, V. Sridharan, S. E. Raasch, N. DeBardeleben, K. B. Ferreira, S. Levy, E. Baseman, and Q. Guan, "Lifetime memory reliability data from the field," in *2017 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, Oct 2017, pp. 1–6.

[43] S. Gupta, T. Patel, C. Engelmann, and D. Tiwari, "Failures in large scale systems: Long-term measurement, analysis, and implications," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '17. New York, NY, USA: ACM, 2017, pp. 44:1–44:12. [Online]. Available: http://doi.acm.org/10.1145/3126908.3126937