Case Study of Error Recovery and Error Propagation on Ranger

Edward Chuah[||]*, Arshad Jhumka*, Samantha Alt[††], Theo Damoulas[||]*, Nentawe Gurumdimma**, Marie-Christine Sawley[††], William L. Barth[¶], Tommy Minyard[¶], James C. Browne[‡]
[||]The Alan Turing Institute, 96 Euston Road, London NW1 2DB, UK. Email: {echuah, tdamoulas}@turing.ac.uk
*University of Warwick, Coventry CV4 7AL, UK. Email: {E.Chuah, H.A.Jhumka, T.Damoulas}@warwick.ac.uk
[††]Intel Corporation, USA and France. Email: {samantha.alt, marie-christine.sawley}@intel.com
**University of Jos P.M.B 2084 Jos, Plateau State Nigeria, Post code: 930001. Email: yusufn@unijos.edu.ng
[¶]Texas Advanced Computing Center, Texas 78758. Email: {bbarth, minyard}@tacc.utexas.edu
[‡]University of Texas at Austin, Texas 78712. Email: browne@cs.utexas.edu

***Abstract*—We give the details of two new dependability-oriented use cases on recovery attempt and error propagation on the Ranger supercomputer. The use cases are: (i) Error propagation between the Lustre file-system I/O and Infiniband, and (ii) Recovery attempt and its impact on the chipset and memory system.**

***Index Terms*—Large cluster system; Lustre file-system I/O and Infiniband; Chipset and memory system; Case study; Diagnosis**

I. Capturing Error Propagation: Lustre file-system I/O and Infiniband

In this section, we show an example of the process of error propagation inferred through the correlations of Infiniband and Lustre I/O resource use counters and Lustre file-system and communication error events.

1) Phase 1: Correlated Infiniband & Lustre file-system counters: The Infiniband and Lustre file-system resource use counters can be used to see what happens when the network and file-system are under heavy use. The `net ib0 tx_dropped` counter records the amount of dropped network packets, and the `net ib0 tx_packets` counter records the amount of transmitted network packets. The `llite /share read_bytes` counter records the amount of bytes read in the Lustre file-system's share partition, and the `llite /share write_bytes` counter records the amount of bytes written to the Lustre file-system's share partition.

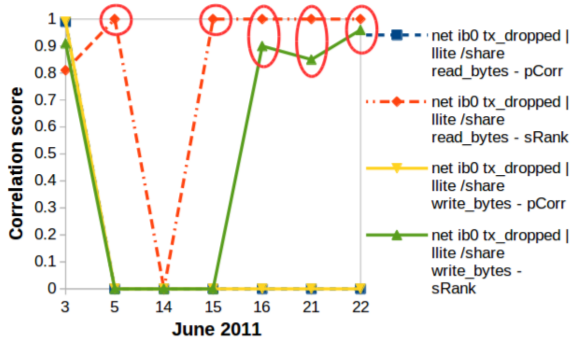
From Fig. 1, we observed that `net ib0 tx_dropped` is strongly correlated to `llite /share read_bytes` with scores that range between 0.81 to 1 on 20 dates, and `net ib0 tx_dropped` is strongly correlated to `llite /share write_bytes` with scores that range between 0.82 to 0.99 on 12 dates. From Fig. 2, we observed that `net ib0 tx_packets` is strongly correlated to `llite /share read_bytes` with scores that range between 0.94 to 1 on 22 dates, and `net ib0 tx_packets` is strongly correlated to `llite /share write_bytes` with scores that range between 0.80 to 0.99 on 16 dates. We observed that only Pearson correlation [1] identified the correlated `net ib0 tx_dropped` and `llite /share read_bytes`, `net ib0 tx_dropped` and `llite /share write_bytes`, and `net ib0 tx_packets` and `llite /share write_bytes` counters on five dates. However, we observed that only Spearman-Rank correlation [1] identified the correlated `net ib0 tx_packets`, `net ib0 tx_dropped`, `llite /share read_bytes` and `llite /share write_bytes` counters on 18 dates. If Pearson correlation is used as the only correlation method, the correlated Infiniband & Lustre file-system counters on these 18 dates would not be identified. However, if Spearman-Rank correlation is used as the only correlation method, the correlated Infiniband & Lustre file-system counters on the five dates would not be identified. Our results show that:

- There is a strong relationship between Infiniband and Lustre I/O activities on 24 dates.
- Pearson correlation and Spearman-Rank correlation are suitable methods. Pearson correlation identified Infiniband & Lustre I/O activities that follow a linear pattern and Spearman-Rank correlation identified Infiniband & Lustre I/O activities that follow a monotonically increasing function.

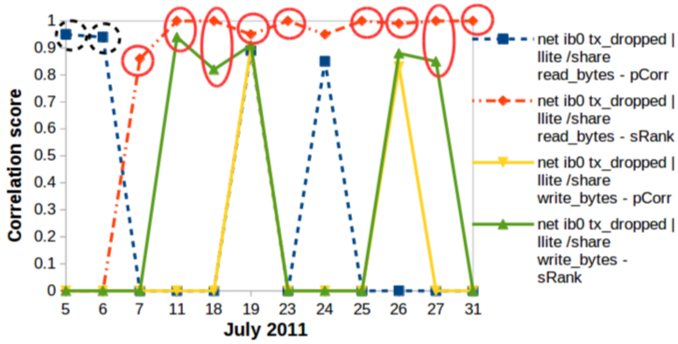
In Section I-2, we will show how error propagation can be inferred between Lustre I/O and Infiniband activities through the correlations of two different groups of error events.

2) Phase 2: Correlated communication & file-system errors: Communication errors can be identified from `error occurred while communicating` events. Errors in the file-system can be identified from the `failure inode` and `error reading dir` events. From Fig. 3(a) and Fig. 3(b), we observed that the `error occurred while communicating` events are strongly correlated to `failure inode` events with scores that range between 0.81 and 1 on 10 dates. From Fig. 3(c), we observed that the `error occurred while communicating` events are strongly correlated to `error reading dir` events with a score of 1 on one date. We observed that Pearson correlation identified the correlated communication and file-system errors on 11 dates but Spearman-Rank correlation identified the correlated communication and file-system errors on six of the 11 dates. Our results show that Pearson correlation identified all the dates when communication and file-system errors are correlated on the given dates on Ranger.

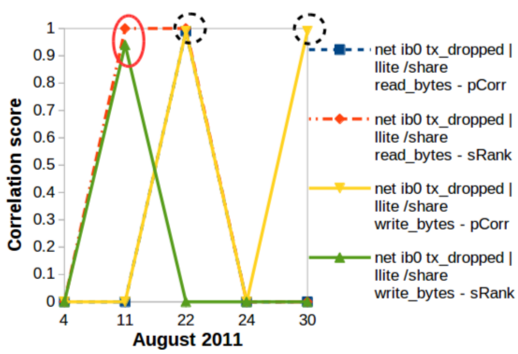
An inode is a data-structure in a Unix-style file-system that stores attributes and disk-block locations about a file. It provides clients the information needed to access files stored on multiple storage servers. However, the information provided by an inode can become lost due to on-disk corruption or failing hard-drives. If a corrupted inode is



(a) June 2011.

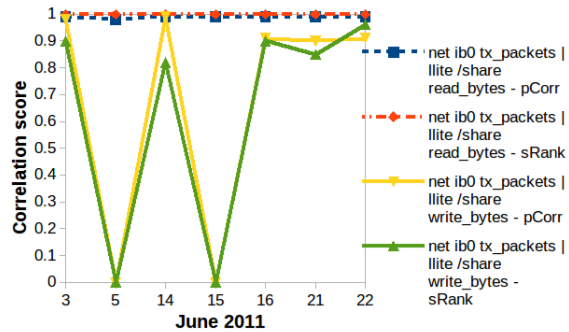


(b) July 2011.

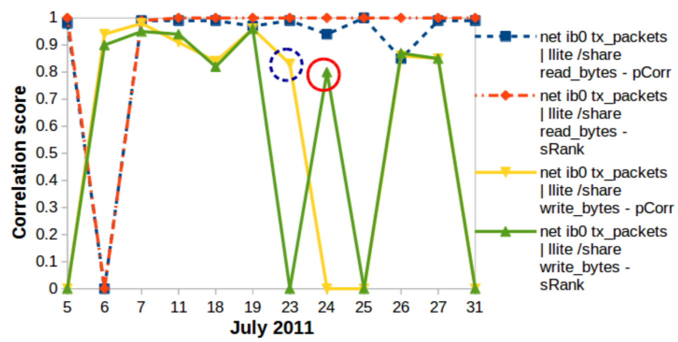


(c) August 2011.

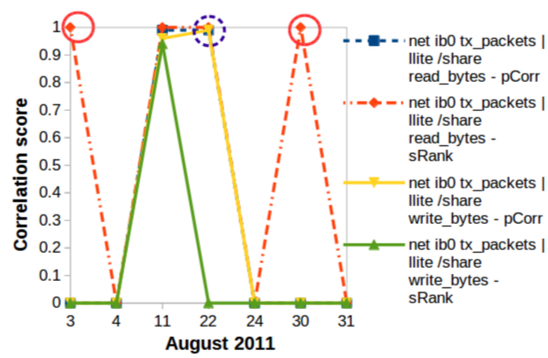
Fig. 1. Correlations between "net ib0 tx_dropped", "llite /share read_bytes" and "llite /share write_bytes" counters. The full-circled counters were identified by Spearman-Rank correlation only. The dot-circled counters were identified by Pearson correlation only.



(a) June 2011.



(b) July 2011.



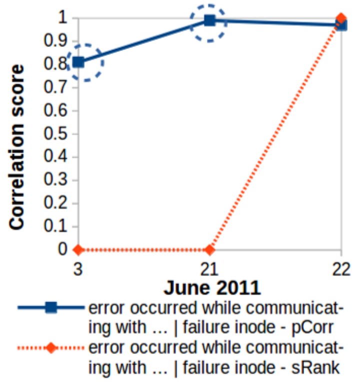
(c) August 2011.

Fig. 2. Correlations between "net ib0 tx_packets", "llite /share read_bytes" and "llite /share write_bytes" counters. The full-circled counters were identified by Spearman-Rank correlation only. The dot-circled counters were identified by Pearson correlation only.

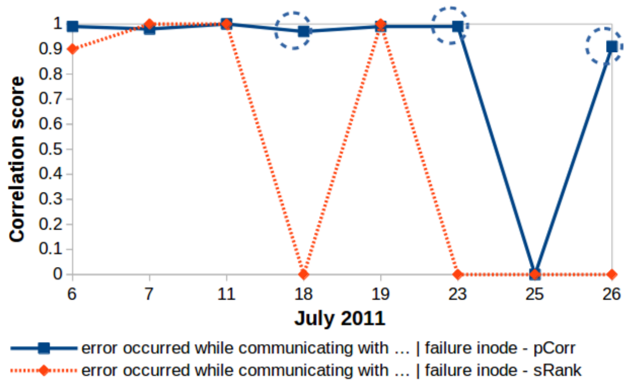
accessed, the information that the client needs is lost and the client is unable to access the file. We implemented a function that scanned the `error occurred while communicating` message and identified the words `failed with Lustre` and `failed with client.c` in all the `error occurred while communicating` messages which are correlated to `failure inode` on all the 10 dates. As we conjectured in Section I-1, our results show that there is indeed error propagation between the Lustre file-system and Infiniband.

Correlations with failures: Next, we scanned the list of correlated events to determine the correlation strength between `error occurred while communicating` and `soft lockup` events, and `failure inode` and `soft lockup` events. A summary of the strongly correlated events is given in Table I. From Table I, we observed that the communication errors are strongly correlated to soft lockup events, and the failure inode events are strongly correlated to soft lockup events on June 21 and July 23.

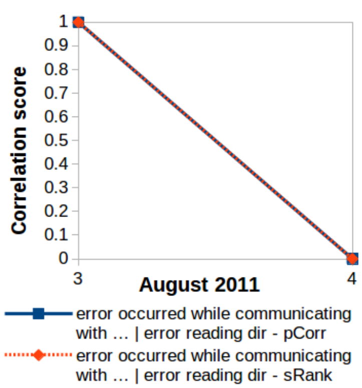
Detailed diagnosis: When a client requested access to data stored in the file-system, the information needed to retrieve the data was lost because it is stored on a faulty inode. This led to a communication error being generated and sent to the client. The client repeated its request but the file-system failed to recover the information, which led the client to hang. The



(a) June 2011.



(b) July 2011.



(c) August 2011.

Fig. 3. (a) and (b) Correlations of "error occurred while communicating" and "failure inode", (c) Correlations of "error occurred while communicating" and "error reading dir". The dot-circled events were identified by Pearson correlation only.

propagation of inode failures (Lustre error) to communication errors led to compute node hang-ups on two of eleven dates, representing a failure rate of 18%.

Further, we found that correlated communication errors and inode failures on the nine other dates are weakly correlated to soft lockup events. Our results suggest that the following had occurred: When a client requested access to data stored in the file-system, the information needed to retrieve the data was lost because it is stored in a faulty inode. This led to the communication error being generated and sent to the client. The client then repeated its request for the data and the file-system was able to recover the information. Recovery from propagation of inode failures occurred on nine out of eleven dates, representing a recovery rate of 81%.

I/O errors can be reported when a client reads a directory from the file-system. We identified the words `failed with Lustre` and `failed with client.c` in the `error occurred while communicating` message which is correlated to `error reading dir` on one date. We manually scanned the list of correlated events and found that `error occurred while communicating` and `error reading dir` events are weakly correlated to soft lockups.

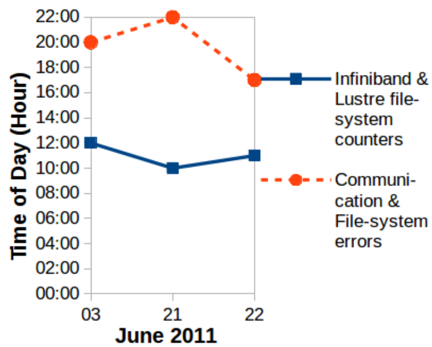
Detailed recovery path: When a client requested access to a directory on the file-system, a directory read error was generated and sent to the client. The client repeated its request however, the file-system recovered from the error and complete the client request.

The benefit of combining analysis of Lustre I/O & Infiniband resource use counters and Lustre file-system & communication error events is as follows: When correlations of Lustre I/O & Infiniband resource use and correlations of Lustre file-system & communication errors occur on the same date, it shows that Lustre I/O & Infiniband activities are associated with the generation of Lustre file-system & communication errors. Therefore, these correlations can be used to track errors between the Lustre file-system and Infiniband.

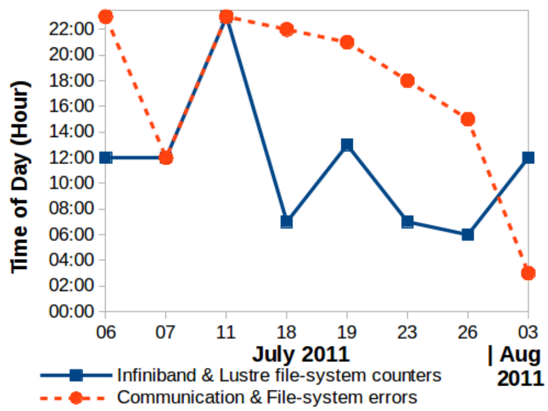
TABLE I
SUMMARY OF CORRELATED "ERROR OCCURRED WHILE COMMUNICATING WITH" AND SOFT LOCKUP, AND CORRELATED "FAILURE INODE" AND SOFT LOCKUP.

Error event	Failure event	Date	pCorr	sRank
error occurred while communicatihg with	soft lockup	June 21	1	-
failure inode	soft lockup	June 21	1	-
error occurred while communicating with	soft lockup	July 23	0.99	-
failure inode	soft lockup	July 23	0.99	-

3) Phase 3: Earliest times of change: From Fig. 4, we observed that the times of change in the correlated Infiniband & Lustre file-system counters and correlated communication & file-system errors on each day are different. The times of change: (i) occurred first in the correlated Infiniband & Lustre file-system counters on eight dates, (ii) occurred first in the correlated communication & file-system errors on one date, and (iii) occurred in both the correlated counters and correlated errors at the same time on two dates. If the correlated errors were used as the only source, the earliest times of change on eight dates would not be identified. Having said that, if the correlated resource use counters were used as the only source, the earliest times of change on one date would not be identified. Our results show that both the correlated resource use counters and correlated errors are required to identify the earliest times of change in the system behaviour on all dates. Further, we observed there are different time-windows between the times of change identified on all dates. The time-windows range from one-hour to 15-hours.



(a) June 2011.



(b) July & August 2011.

Fig. 4. Times of change in the correlated Infiniband & Lustre file-system counters and correlated communication & file-system errors.

4) Validation: Next, we test the significance of: (i) the correlation coefficient of the strongly *positive* correlated resource use counter groups, and (ii) the correlation coefficient of the strongly *positive* correlated error groups. We test all the correlation coefficients against the null hypothesis and obtained the z-scores for all the correlation coefficients and a summary is given in Table II. From Table II, we observed

TABLE II
SUMMARY OF z-SCORES. n CONTAINS THE NUMBER OF HOURLY TIME-BINS IN ONE DAY OF LOGS.

Correlated groups	June 2011	July 2011	Aug 2011
Infiniband & Lustre file-system resource counters ($n = 24$)	$3.71 \leq z_r \leq 10.68$	$3.58 \leq z_r \leq 10.68$	$6.51 \leq z_r \leq 10.68$
Communication & File-system errors ($n = 24$)	$3.71 \leq z_e \leq 10.68$	$5.29 \leq z_e \leq 10.68$	$z_e = 10.68$

that the z-scores for all the correlation coefficients range from 3.58 to 10.68. At the 99% confidence level, under the null hypothesis $z_{0r} = 2.64$ and $z_{0e} = 2.64$. Hence, we reject the null hypothesis in favour of the alternate hypothesis.

Next, we determine the probability of rejecting the null hypothesis when it is true. We apply a one-sided test and use the significance level, $\alpha = 0.01$ for all given hypothesis tests to obtain a P-value. From Table II, we observed that the lowest z-score is 3.58. Since this is a one-sided test, the P-value is equal to the probability of observing a value greater than 3.58 in the standard normal distribution, or $P(Z > 3.58) = 1 - P(Z \leq 3.58) = 1 - 0.999828 = 0.000172$. Using the Bonferroni correction to counteract the problem of inflation in false positive due to multiple hypothesis tests [2], we obtained the adjusted P-value $0.000172 \times 24 = 0.0041$ where 24 is the number of dates. The P-value is less than 0.01, indicating it is highly unlikely this result would be observed under the null hypothesis. All the z-scores in Table II are greater than or equal to 3.58 and all the adjusted P-values are less than 0.01, indicating it is highly unlikely these results would be observed under the null hypothesis.

II. CAPTURING RECOVERY ATTEMPT AND ITS IMPACT: CHIPSET AND MEMORY SYSTEM

In this section, we explain how correlations between CPU and memory resource use counters and correlations between chipset and ECC[1] errors can be used to first infer error recovery, and then assess the impact of the ECC recovery mechanism on the system reliability.

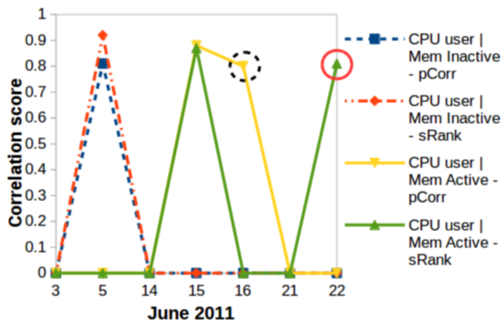
1) Phase 1: Correlated CPU & Memory counters: The CPU and memory resource use counters can be used to see what happens when CPU and memory activities are occurring in the cluster system. The `CPU user` counter records CPU usage by user, and the `CPU system` counter records CPU usage by the system. The `MEM Inactive` counter records the amount of pages that were not accessed recently in main memory, and the `MEM Active` counter records the amount of pages that were accessed recently in main memory.

From Fig. 5, we observed that `CPU user` is strongly correlated to `MEM Inactive` with scores that range between 0.81 to 0.93 on five dates, and `CPU user` is strongly correlated to `MEM Active` with scores that range between 0.80 to 0.97 on six dates. From Fig. 6, we observed that `CPU system` is strongly correlated to `MEM Inactive` with scores that range between 0.84 to 0.97 on eight dates, and `CPU system` is strongly correlated to `MEM Active` with scores that range between 0.81 to 0.97 on seven dates. We observed that only Pearson correlation [1] identified the correlated counters on June 16, July 23 and August 04 and 11. We observed that only Spearman-Rank correlation [1] identified the correlated counters on seven different dates. If Pearson correlation is used as the only correlation method, the correlated CPU and memory counters on June 05 and 22, July 06, 11 and 24, and August 22 and 30 would not be identified. However, if Spearman-Rank correlation is used as the only correlation method, the correlated CPU and memory counters on June 16, July 23 and August 04 and 11 would not be identified. Our results show that:

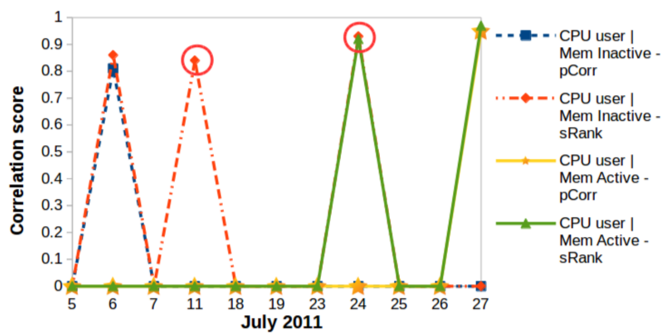
- There is a strong relationship between CPU and memory activities on 13 dates.

[1]ECC is an acronym for Error Correcting Code.

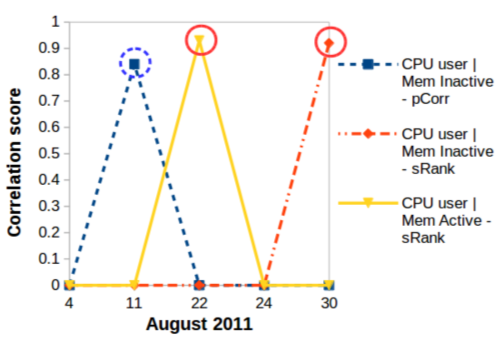
- Pearson correlation and Spearman-Rank correlation are suitable methods. Pearson correlation identified the CPU & memory resource usage that follow a linear pattern and Spearman-Rank correlation identified the CPU & memory resource usage that follow a monotonically increasing function.



(a) June 2011.



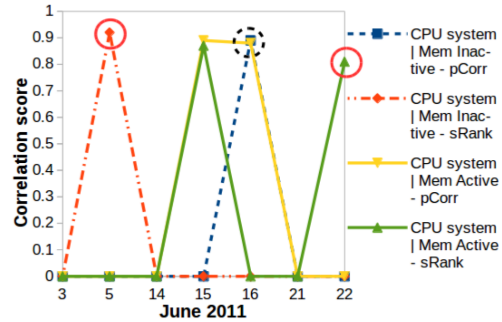
(b) July 2011.



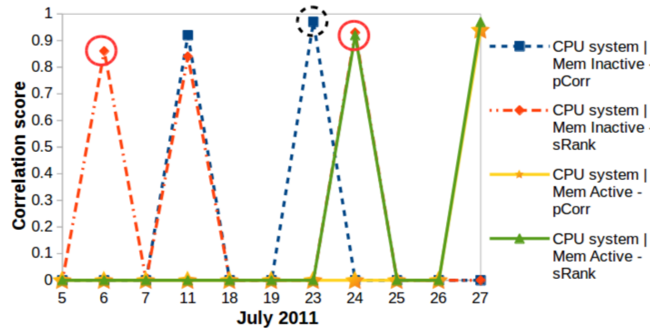
(c) August 2011.

Fig. 5. Correlation of "CPU user" and "MEM Inactive" counters, and correlation of "CPU user" and "MEM Active" counters. The full-circled counters were identified by Spearman-Rank correlation only. The dot-circled counters were identified by Pearson correlation only.

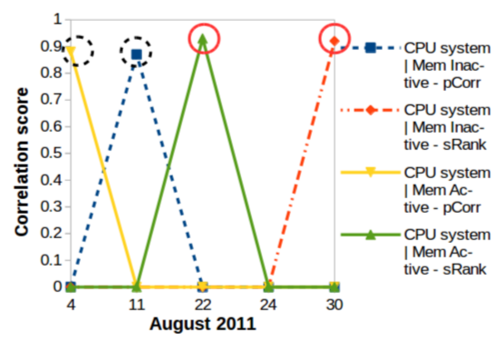
2) Phase 2: Correlated Chipset & ECC Errors: The northbridge is a chip in the core logic chipset architecture on a computer motherboard. The northbridge is connected directly to the CPU and it typically handles communication among the CPU, memory and graphics controller. ECC memory is used in computers where internal data corruption can not be tolerated under any circumstances. When the CPU attempts to access corrupted data stored in ECC memory, the northbridge reports a Northbrigde error (`Northbridge error`), the CPU core (`core`) that attempted to access the data, and recovery from an ECC error (`ECC error`). From Fig. 7, we observed that `Northbridge error` events are strongly correlated to `ECC error` events with scores that range between 0.99 and 1 on 26 dates, and `Northbridge error` events are strongly correlated to `core` events with scores that range between 0.99 and 1 on 26 dates. We observed that both Pearson and Spearman-Rank correlations [1] identified the correlated northbridge, CPU and ECC error events on all 26 dates. Our results show that internal data corruption have occurred on a daily basis on Ranger. Further, we observed that correlations



(a) June 2011.



(b) July 2011.



(c) August 2011.

Fig. 6. Correlation of "CPU system" and "MEM Inactive" counters, and correlation of "CPU system" and "MEM Active" counters. The full-circled counters were identified by Spearman-Rank correlation only. The dot-circled counters were identified by Pearson correlation only.

of CPU and memory resource use counters occurred on all the dates when northbridge, core and ECC errors are correlated.

Correlations with failures: Next, we manually scanned the list of correlated events to determine the strength of the correlation between `Northbridge error` and soft lockup events, `core` and soft lockup events, and `ECC error` and soft lockups events. We found that `Northbridge error`, `core` and `ECC error` events are *weakly* correlated to soft lockup events on all 26 dates. This represents a recovery rate of 100%.

Detailed diagnosis: When correlations of CPU & memory resource use counters and correlations of chipset & ECC errors occur on the same date, it shows that CPU memory usage activities are associated with the generation of chipset and memory errors. When the CPU accessed corrupted data stored in ECC memory, this triggered an ECC error which was subsequently corrected. Therefore, the correlated CPU & memory resource use counters and correlated chipset & memory errors can be used to monitor recovery from internal data corruption.

3) Phase 3: Earliest times of change: From Fig. 8, we observed that the earliest times of change in the correlated CPU & memory resource use counters and correlated chipset & ECC errors on each date are different. The times of change: (i) occurred first in the correlated CPU and memory resource use counters on five dates, (ii) occurred first in the correlated chipset and ECC errors on seven dates, and (iii) occurred in both the correlated counters and correlated errors at the same time on one date. If the correlated errors were used as the only source, the earliest times of change on five dates would not be identified. Having said that, if the correlated resource use counters were used as the only source, the earliest times of change on seven dates would not be identified. Our results show that both the correlated resource use counters and correlated errors are required to identify the earliest times of change in the system behaviour on all dates. Further, we observed there are different time-windows between the times of change identified on all dates. The time-windows range from one to 19-hours.

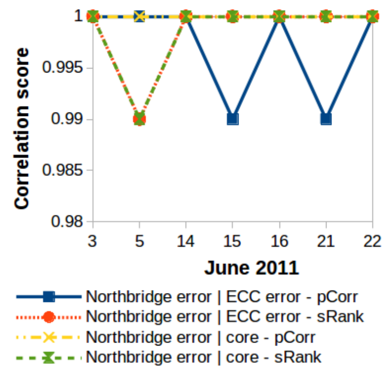
4) Validation: Next, we test the significance of: (i) the correlation coefficient of the strongly *positive* correlated resource use counter groups, and (ii) the correlation coefficient of the strongly *positive* correlated error groups. We test all the correlation coefficients against the null hypothesis and obtained the z-scores for all the correlation coefficients and a summary is given in Table III. From Table III, we observed

TABLE III
SUMMARY OF z-SCORES. n CONTAINS THE NUMBER OF HOURLY TIME-BINS IN ONE DAY OF LOGS.

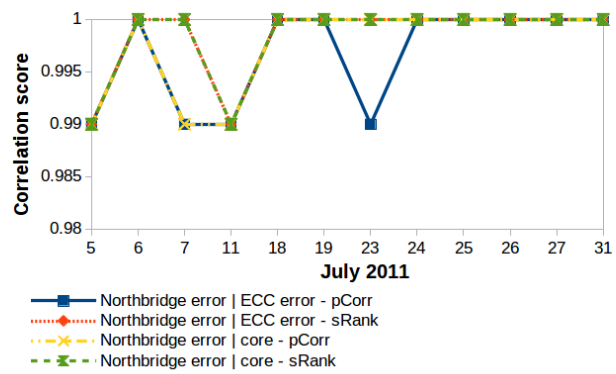
Correlated groups	June 2011	July 2011	Aug 2011
CPU & Memory counters ($n = 24$)	$3.74 \leq z_r \leq 5.86$	$3.74 \leq z_r \leq 8.16$	$4.17 \leq z_r \leq 6.18$
Chipset & ECC errors ($23 \leq n \leq 24$)	$z_e = 10.68$	$z_e = 10.68$	$z_e = 10.68$

that the z-scores for all the correlation coefficients range from 3.74 to 10.68. At the 99% confidence level, under the null hypothesis $z_{0r} = 2.64$ and $z_{0e} = 2.64$. Hence, we reject the null hypothesis in favour of the alternate hypothesis.

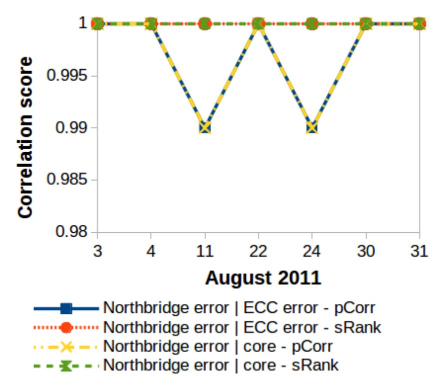
Next, we determine the probability of rejecting the null hypothesis when it is true. We apply a one-sided test and use the significance level, $\alpha = 0.01$ for all given hypothesis tests to obtain a P-value. From Table III, we observed that the lowest z-score is 3.74. Since this is a one-sided test, the P-value is equal to the probability of observing a value greater than 3.74 in the standard normal distribution, or $P(Z > 3.74) = 1 - P(Z \leq 3.74) = 1 - 0.99992 = 0.00008$. Using the Bonferroni correction to counteract the problem of inflation in false positive due to multiple hypothesis tests [2],



(a) June 2011.

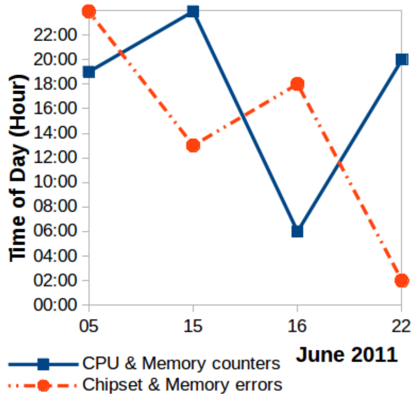


(b) July 2011.

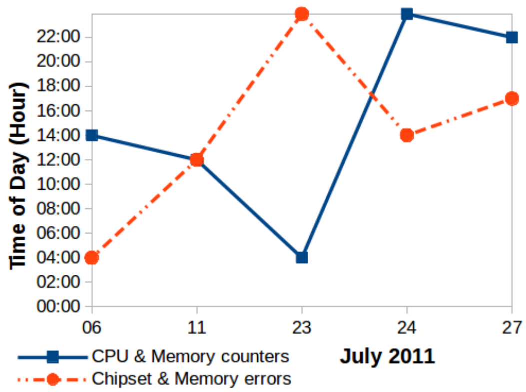


(c) August 2011.

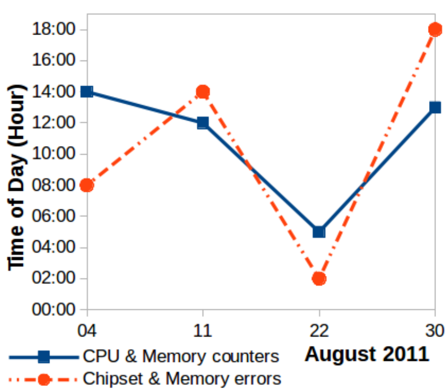
Fig. 7. Correlation of "Northbridge error" and "ECC error" events, and correlation of "Northbridge error" and "core" events.



(a) June 2011.



(b) July 2011.



(c) August 2011.

Fig. 8. Times of change in the correlated CPU & memory resource use counters and correlated chipset & ECC errors.

we obtained the adjusted P-value $0.00008 \times 26 = 0.00208$ where 26 is the number of dates. The P-value is less than 0.01, indicating it is highly unlikely this result would be observed under the null hypothesis. All the z-scores in Table III are greater than or equal to 3.74 and all the adjusted P-values are less than 0.01, indicating it is highly unlikely these results would be observed under the null hypothesis.

ACKNOWLEDGEMENTS

We would like to thank the Texas Advanced Computing Center (TACC) for providing the Ranger cluster log data and granting access to their systems administrators. We also thank Karl Solchenbach (Intel Corporation, Europe) for granting access to his research scientists. This research is supported by The Alan Turing Institute under the EPSRC grant EP/N510129/1, The Alan Turing Institute-Intel partnership and the National Science Foundation under OCI awards #0622780 and #1203604 to TACC at the University of Texas at Austin.

REFERENCES

[1] R. E. Walpole, R. H. Myers, and S. L. Myers, *Probability and Statistics for Engineers and Scientists*. Prentice Hall International, 1998.

[2] J. J. Goeman and A. Solari, "Multiple hypothesis testing in genomics," *Statistics in Medicine*, vol. 33, no. 11, pp. 1946–1978, 2014. [Online]. Available: http://dx.doi.org/10.1002/sim.6082