A Hardware/Software Performance/Resilience/Power Co-Design Tool for Extreme-scale Computing

Christian Engelmann* and Thomas Naughton, Oak Ridge National Laboratory
**engelmannc@ornl.gov / (865)-574-3132*

The path to exascale poses several challenges related to power, performance, resilience, productivity, programmability, data movement, and data management. Resilience, i.e., providing efficiency and correctness in the presence of faults, is one of the most important challenges as systems scale up in component count (up to 1,000,000 nodes, each with up to 10,000 cores) and component reliability decreases (7 nm technology at near-threshold voltage) [22, 47, 8, 13, 21, 32, 45]. A number of high-performance computing (HPC) resilience technologies have been or are being developed, such as checkpoint/restart, message logging, algorithm-based fault tolerance, containment domains, and redundancy [12, 15, 18, 19, 24, 25, 35, 33, 38, 49, 50, 51, 52, 9]. However, there are no tools, methods, and metrics to compare them and to identify the cost/benefit trade-off between the key system design factors: performance, resilience, and power consumption. There are also no tools to evaluate their efficiency and correctness in the presence of faults.

The Extreme-scale Simulator (xSim) [14, 3, 16, 30] is a performance investigation toolkit that permits running native HPC applications or proxy applications [43] in a controlled environment with millions of concurrent execution threads, while observing application performance in a simulated extreme-scale system for hardware/software co-design. Using a lightweight parallel discrete event simulation (PDES), xSim executes an MPI application on a much smaller system in a highly oversubscribed fashion with a virtual wall clock time, such that performance data can be extracted based on a processor and a network model with an appropriate simulation scalability/accuracy trade-off. xSim is designed like a traditional performance tool, as an interposition library that sits between the MPI application and the MPI layer, using the MPI profiling interface. It has been run up to 134,217,728 (2^{27}) communicating MPI ranks using a 960-core Linux cluster.

xSim's ability to simulate the architectural properties of extreme-scale HPC systems at *reasonable* accuracy and to run real applications or proxies makes it the prime candidate for a HPC resilience hardware/software co-design toolkit for investigating the performance/resilience/power trade-off of future HPC architecture choices with different resilience properties (vulnerabilities) and strategies (mitigation techniques). Our overall effort focuses on adding features to (1) permit the injection of different faults, errors, and failures into the simulation, (2) model various propagation, isolation, and detection properties of the simulated system, (3) support a variety of avoidance, masking, and recovery strategies, (4) model the power consumption of the entire simulated system, and (5) study the performance, resilience, and power consumption impact with different parameter sets for (1), (2), (3), and (4) using standardized metrics.

Initial efforts focused on adding new features that permit the injection of MPI process failures, their propagation/detection/notification within the simulation, and their handling using application-level checkpoint/restart [17]. A process failure is injected by scheduling it at the targeted process. It is activated when the simulated process clock reaches the scheduled time of failure. A separate simulated process failure detection simulation offers a per-process detection latency using simulated communication timeouts. Job failures are simulated upon detection if the error handler on the communicator is MPI_ERRORS_ARE_FATAL. A simulated job restart feature offers continuous simulation timing after an abort and following a restart. These new capabilities enable the observation of application behavior and performance under failure within a simulated future-generation HPC system using the most common fault handling technique.

Further work aimed at a user-level failure mitigation (ULFM) [2] capability for algorithm-based fault tolerance (ABFT) using the discussed fault tolerance extension to the MPI standard. ULFM enables application-level process failure handling using communicator error handlers, such as MPI_ERRORS_RETURN. Additional MPI calls offer notification, agreement, and reconfiguration. An MPI error due to a failed process returns the error code MPI_ERR_PROC_FAILED (or MPI_ERR_PENDING for MPI_ANY_SOURCE receives). MPI_Comm_revoke() can be used to notify other processes about a tainted communicator, while MPI_Comm_shrink() allows to create a new one that excludes failed ranks. MPI_Comm_agree() facilitates

an agreement on a single value to enable containment domains and failure-tolerant collectives [27]. The ULFM capability is fully implemented in xSim with realistic communication characteristics based on the network model. Ongoing work focuses on evaluating the performance of resilient applications under failure.

Future work in xSim will focus on ABFT simulation for silent data corruption, probabilistic fault injection/prediction simulation, prediction-based resilience mechanisms (e.g. MPI process migration and dynamic adaptive redundancy), full/partial redundancy simulation, as well as, processor, network, and storage reliability and power models. Future work in hardware/software performance/resilience/power co-design will target studying trade-offs with different applications, resilience mechanisms, and system architectures. The proposed approach may also be adopted to study other execution models.

Related Work

A number of simulation tools exist, ranging from cycle-accurate processor and memory simulators, such as gem5 [1] and DRAMsim [53, 42], to communication-accurate trace-driven solutions, such as BigSim [54] and DIMEMAS [23] (which processes MPIDTrace traces to generate output for PARAVER [40] and Vampir [31]). The Structural Simulation Toolkit (SST) [41] offers simulation of novel compute-node architectures using a modular PDES framework. SST/macro is focused on coarser grained simulation to investigate the architectural effects on MPI application performance with reduced accuracy similar to xSim [28]. Also, SimGrid [10] and OMNeT++ [37] are multi-purpose simulation toolkits that have been used to investigate the runtime capabilities of future system architectures. There are a also variety of network simulators, such as ns3 (http://www.nsnam.org) and NetSim (http://tetcos.com/software.html), that are able to provide network performance metrics at various abstraction levels, such as network, sub-network, and packet traces. Fault injection has only recently been used in HPC [39, 20, 11, 44, 34, 4], mostly in the context of soft errors. A number of non-HPC fault injection tools exist [29, 46, 26], including static and dynamic instrumentation tools, like Pin [6, 36], KernInst [48], DynInst [5], and DTrace [7]

Assessment

Challenges addressed: This approach addresses the exascale modeling and simulation challenges for developing co-design tools to understand the failure and response behavior of systems and applications, as well as, to evaluate system and application efficiency and correctness in the presence of faults.

Maturity: Our initial results offer the capability to investigate application behavior and performance under failure using the most common fault handling technique, application-level checkpoint restart. The targeted research extends this capability to other failure classes and resilience techniques.

Uniqueness: The scale and close coupling at which exascale systems are expected to operate makes this approach particularly unique, especially considering the design trade-off between power, performance, and resilience. These constraints are unique to exascale systems.

Novelty: This work is novel in several ways. The proposed xSim is (a) the first fault injection toolkit for extreme-scale systems, (b) the only fault injection toolkit with parameterized resilience properties and strategies, and (c) the first HPC co-design toolkit that considers architectural performance and resilience parameters to optimize parallel application performance within a given power consumption budget.

Applicability: At smaller-scale, the design trade-off between power, performance, and resilience in parallel and distributed systems is also important to data centers, cloud computing, and sensor networks. The proposed approach could produce valuable results applicable to these areas.

Effort: Our initial accomplishments show that there are low-hanging fruits that provide an early pay-off. We are targeting an iterative approach based on realistic assumptions, such as by focusing on the most likely failures and the most prominent failure handling techniques.

This research was sponsored in part by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory (ORNL) and by the Office of Advanced Scientific Computing Research, U.S. Department of Energy (DOE). This manuscript has been authored by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725 with the DOE.

References

[1] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood. The gem5 simulator. *SIGARCH Comput. Archit. News*, 39(2):1–7, Aug. 2011. ISSN 0163-5964. doi: 10.1145/2024716.2024718. URL http://doi.acm.org/10.1145/2024716.2024718.

[2] W. Bland, A. Bouteiller, T. Herault, J. Hursey, G. Bosilca, and J. J. Dongarra. An evaluation of user-level failure mitigation support in mpi. In *Proceedings of the 19th European conference on Recent Advances in the Message Passing Interface*, EuroMPI'12, pages 193–203, Berlin, Heidelberg, 2012. Springer-Verlag. ISBN 978-3-642-33517-4. doi: 10.1007/978-3-642-33518-1_24. URL http://dx.doi.org/10.1007/978-3-642-33518-1_24.

[3] S. Böhm and C. Engelmann. xSim: The extreme-scale simulator. In *Proceedings of the International Conference on High Performance Computing and Simulation (HPCS) 2011*, pages 280–286, Istanbul, Turkey, July 4-8, 2011. IEEE Computer Society, Los Alamitos, CA, USA. ISBN 978-1-61284-383-4. doi: http://dx.doi.org/10.1109/HPCSim.2011.5999835. URL http://www.christian-engelmann.info/publications/boehm11xsim.pdf.

[4] G. Bronevetsky and B. de Supinski. Soft error vulnerability of iterative linear algebra methods. In *Proceedings of the 22nd annual international conference on Supercomputing*, ICS '08, pages 155–164, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-158-3. doi: 10.1145/1375527.1375552. URL http://doi.acm.org/10.1145/1375527.1375552.

[5] B. Buck and J. K. Hollingsworth. An api for runtime code patching. *Int. J. High Perform. Comput. Appl.*, 14(4):317–329, 2000. ISSN 1094-3420. doi: http://dx.doi.org/10.1177/109434200001400404.

[6] P. P. Bungale and C.-K. Luk. PinOS: A programmable framework for whole-system dynamic instrumentation. In *Proceedings of the 3rd International Conference on Virtual Execution Environments (VEE'07)*, pages 137–147, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-630-1. doi: http://doi.acm.org/10.1145/1254810.1254830.

[7] B. M. Cantrill, M. W. Shapiro, and A. H. Leventhal. Dynamic instrumentation of production systems. In *Proceedings of the USENIX Annual Technical Conference (USENIX'04)*, pages 15–28. USENIX, June 27 – July 2, 2004. URL http://www.usenix.org/events/usenix04/tech/general/full_papers/cantrill/cantrill.pdf.

[8] F. Cappello, A. Geist, B. Gropp, L. V. Kale, W. Kramer, and M. Snir. Toward exascale resilience. Technical Report TR-JLPC-09-01, University of Illinois at Urbana-Champaign (UIUC) - Institut National de Recherche en Informatique et en Automatique (INRIA) Joint Laboratory on PetaScale Computing, June 2009. URL http://institutes.lanl.gov/resilience/docs/Toward%20Exascale%20Resilience.pdf.

[9] J. Chung, I. Lee, M. Sullivan, J. H. Ryoo, D. W. Kim, D. H. Yoon, L. Kaplan, and M. Erez. Containment domains: A scalable, efficient, and flexible resilience scheme for exascale systems. In *the Proceedings of SC12*, November 2012.

[10] P.-N. Clauss, M. Stillwell, S. Genaud, F. Suter, H. Casanova, and M. Quinson. Single Node On-Line Simulation of MPI Applications with SMPI. In *International Parallel & Distributed Processing Symposium*, Anchorange (AK), États-Unis, May 2011. IEEE. URL http://hal.inria.fr/inria-00527150. RR-7426 RR-7426.

[11] C. da Lu and D. A. Reed. Assessing fault sensitivity in mpi applications. In *SC '04: Proceedings of the 2004 ACM/IEEE conference on Supercomputing*, page 37, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2153-3. doi: http://dx.doi.org/10.1109/SC.2004.12.

[12] J. Elliott, K. Kharbas, D. Fiala, F. Mueller, K. Ferreira, and C. Engelmann. Combining partial redundancy and checkpointing for HPC. In *Proceedings of the* 32nd *International Conference on Distributed Computing Systems (ICDCS) 2012*, Macau, China, June 18-21, 2012. IEEE Computer Society, Los Alamitos, CA, USA. URL http://www.christian-engelmann.info/publications/elliott12combining.pdf.

[13] M. Elnozahy, R. Bianchini, T. El-Ghazawi, A. Fox, F. Godfrey, A. Hoisie, K. McKinley, R. Melhem, J. Plank, P. Ranganathan, and J. Simons. System resilience at extreme scale. Technical report, Defense Advanced Research Project Agency (DARPA), 2008. URL http://institutes.lanl.gov/resilience/docs/Toward%20Exascale%20Resilience.pdf.

[14] C. Engelmann. Scaling to a million cores and beyond: Using light-weight simulation to understand the challenges ahead on the road to exascale. *Future Generation Computer Systems (FGCS)*, 2013. To appear.

[15] C. Engelmann and S. Böhm. Redundant execution of HPC applications with MR-MPI. In *Proceedings of the* 10^{th} *IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN) 2011*, pages 31–38, Innsbruck, Austria, Feb. 15-17, 2011. ACTA Press, Calgary, AB, Canada. ISBN 978-0-88986-864-9. doi: http://dx.doi.org/10.2316/P.2011.719-031. URL http://www.christian-engelmann.info/publications/engelmann11redundant.pdf.

[16] C. Engelmann and F. Lauer. "Facilitating co-design for extreme-scale systems through lightweight simulation". In *Proceedings of the* 12^{th} *IEEE International Conference on Cluster Computing (Cluster) 2010:* 1^{st} *Workshop on Application/Architecture Co-design for Extreme-scale Computing (AACEC)*, pages 1–8, Hersonissos, Crete, Greece, Sept. 20-24, 2010. IEEE Computer Society. ISBN 978-1-4244-8395-2. doi: http://dx.doi.org/10.1109/CLUSTERWKSP.2010.5613113. URL http://www.csm.ornl.gov/~engelman/publications/engelmann10facilitating.pdf.

[17] C. Engelmann and T. Naughton. Toward a performance/resilience tool for hardware/software co-design of high-performance computing systems. In *Proceedings of the* 42^{nd} *International Conference on Parallel Processing (ICPP) 2013:* 4^{th} *International Workshop on Parallel Software Tools and Tool Infrastructures (PSTI)*, Lyon, France, Oct. 2, 2013. IEEE Computer Society, Los Alamitos, CA, USA. To appear.

[18] C. Engelmann, G. Vallée, T. Naughton, and S. L. Scott. Proactive fault tolerance using preemptive migration. In *Proceedings of the* 17^{th} *Euromicro International Conference on Parallel, Distributed, and network-based Processing (PDP) 2009*, pages 252–257, Weimar, Germany, Feb. 18-20, 2009. IEEE Computer Society. ISBN 978-0-7695-3544-9. URL http://doi.ieeecomputersociety.org/10.1109/PDP.2009.31.

[19] G. Fagg, E. Gabriel, G. Bosilca, T. Angskun, Z. Chen, J. Pjesivac-grbovic, K. London, and J. Dongarra. Extending the mpi specification for process fault tolerance on high performance computing systems. In *In Proceeding of International Supercomputer Conference (ICS)*, 2003.

[20] D. Fiala, F. Mueller, C. Engelmann, K. Ferreira, R. Brightwell, and R. Riesen. Detection and correction of silent data corruption for large-scale high-performance computing. In *Proceedings of the* 25^{th} *IEEE/ACM International Conference on High Performance Computing, Networking, Storage and Analysis (SC) 2012*, pages 78:1–78:12, Salt Lake City, UT, USA, Nov. 10-16, 2012. ACM Press, New York, NY, USA. ISBN 978-1-4673-0804-5. URL http://www.christian-engelmann.info/publications/fiala12detection2.pdf.

[21] A. Geist and R. F. Lucas. Major computer science challenges at exascale. Technical report, International Exascale Software Project, Feb. 2009. URL http://www.exascale.org/mediawiki/images/8/87/ExascaleSWChallenges-Geist_Lucas.pdf. Whitepaper.

[22] A. Geist, B. Lucas, M. Snir, S. Borkar, E. Roman, M. Elnozahy, B. Still, A. Chien, R. Clay, J. Wu, C. Engelmann, N. DeBardeleben, R. Ross, L. Kaplan, M. Schulz, M. Heroux, S. Krishnamoorthy, L. Nowell, A. Vishnu, and L.-A. Talley. U.s. department of energy fault management workshop. Workshop report submitted to the U.S. Department of Energy, Aug. 2012. URL http://www.christian-engelmann.info/publications/geist12department.pdf.

[23] S. Girona, J. Labarta, and R. M. Badia. "Validation of dimemas communication model for MPI collective operations". In *Lecture Notes in Computer Science: Proceedings of the* 7^{th} *European PVM/MPI Users' Group Meeting (EuroPVM/MPI) 2000*, volume 1908, pages 39–46, Balatonfüred,

Hungary, Sept. 10-13 2000. Springer Verlag, Berlin, Germany. ISBN 978-3-540-41010-2. URL http://dx.doi.org/10.1007/3-540-45255-9_9.

[24] N. Gottumukkala, B. Leangsuksun, N. Taerat, R. Nassar, and S. L. Scott. Reliability-aware resource allocation in hpc systems. In *Proceedings of the 2007 IEEE International Conference on Cluster Computing*, CLUSTER '07, pages 312–321, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 978-1-4244-1387-4. doi: 10.1109/CLUSTR.2007.4629245. URL http://dx.doi.org/10.1109/CLUSTR.2007.4629245.

[25] P. H. Hargrove and J. C. Duell. Berkeley Lab Checkpoint/Restart (BLCR) for Linux clusters. In *Journal of Physics: Proceedings of the Scientific Discovery through Advanced Computing Program (SciDAC) Conference 2006*, volume 46, pages 494–499, Denver, CO, USA, June 25-29, 2006. Institute of Physics Publishing, Bristol, UK. URL http://www.iop.org/EJ/article/1742-6596/46/1/067/jpconf6_46_067.pdf.

[26] S. K. S. Hari, S. V. Adve, H. Naeimi, and P. Ramachandran. Relyzer: exploiting application-level fault equivalence to analyze application resiliency to transient faults. In *Proceedings of the seventeenth international conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS XVII, pages 123–134, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-0759-8. doi: 10.1145/2150976.2150990. URL http://doi.acm.org/10.1145/2150976.2150990.

[27] J. Hursey and R. Graham. Preserving collective performance across process failure for a fault tolerant MPI. In *16th International Workshop on High-Level Parallel Programming Models and Supportive Environments (HIPS) held in conjunction with the 25th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Anchorage, Alaska, May 2011.

[28] C. L. Janssen, H. Adalsteinsson, S. Cranford, J. P. Kenny, A. Pinar, D. A. Evensky, and J. Mayo. A simulator for large-scale parallel computer architectures. *International Journal of Parallel and Distributed System Technology*, 1(2):57–73, Apr. 2010. ISSN 1947-3532. doi: 10.4018/jdst.2010040104. URL http://dx.doi.org/10.4018/jdst.2010040104.

[29] João Carreira and Henrique Madeira and João Gabriel Silva. Xception: A Technique for the Experimental Evaluation of Dependability in Modern Computers. *IEEE Transactions on Software Engineering*, 24(2), Feb. 1998. URL http://www.xception.org/files/IEEETSE98.pdf.

[30] I. S. Jones and C. Engelmann. Simulation of large-scale HPC architectures. In *Proceedings of the* 40^{th} *International Conference on Parallel Processing (ICPP) 2011:* 2^{nd} *International Workshop on Parallel Software Tools and Tool Infrastructures (PSTI)*, pages 447–456, Taipei, Taiwan, Sept. 13-19, 2011. IEEE Computer Society, Los Alamitos, CA, USA. ISBN 978-0-7695-4511-0. doi: http://dx.doi.org/10.1109/ICPPW.2011.44. URL http://www.christian-engelmann.info/publications/jones11simulation.pdf.

[31] A. Knüpfer, H. Brunst, J. Doleschal, M. Jurenz, M. Lieber, H. Mickler, M. S. Müller, and W. E. Nagel. The vampir performance analysis tool-set. In M. Resch, R. Keller, V. Himmler, B. Krammer, and A. Schulz, editors, *Tools for High Performance Computing*, pages 139–155. Springer Berlin Heidelberg, 2008. ISBN 978-3-540-68564-7.

[32] P. Kogge et al. ExaScale computing study: Technology challenges in achieving exascale systems. Technical report, Defense Advanced Research Project Agency (DARPA) Information Processing Techniques Office (IPTO), 2008. http://users.ece.gatech.edu/~mrichard/ExascaleComputingStudyReports/exascale_final_report_100208.pdf.

[33] P. Lemarinier, A. Bouteiller, T. Herault, and G. Krawezik. Improved message logging versus improved coordinated checkpointing for fault tolerant mpi. In *in IEEE International Conference on Cluster Computing (Cluster 2004). IEEE CS*. Press, 2004.

[34] D. Li, J. S. Vetter, and W. Yu. Classifying soft error vulnerabilities in extreme-scale scientific applications using a binary instrumentation tool. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, SC '12, pages 57:1–57:11, Los Alamitos, CA, USA, 2012. IEEE Computer Society Press. ISBN 978-1-4673-0804-5. URL

http://dl.acm.org/citation.cfm?id=2388996.2389074.

[35] M. Li, S. Vazhkudai, A. Butt, F. Meng, X. Ma, Y. Kim, C. Engelmann, and G. Shipman. Functional partitioning to optimize end-to-end performance on many-core architectures. In *Proceedings of the* 23^{rd} *IEEE/ACM International Conference on High Performance Computing, Networking, Storage and Analysis (SC) 2010*, pages 1–12, New Orleans, LA, USA, Nov. 13-19, 2010. ACM Press, New York, NY, USA. ISBN 978-1-4244-7559-9. doi: http://dx.doi.org/10.1109/SC.2010.28. URL http://www.christian-engelmann.info/publications/li10functional.pdf.

[36] C.-K. Luk, R. Cohn, R. Muth, H. Patil, A. Klauser, G. Lowney, S. Wallace, V. J. Reddi, and K. Hazelwood. Pin: building customized program analysis tools with dynamic instrumentation. In *PLDI '05: Proceedings of the 2005 ACM SIGPLAN conference on Programming language design and implementation*, pages 190–200, New York, NY, USA, 2005. ACM. ISBN 1-59593-056-6. doi: http://doi.acm.org/10.1145/1065010.1065034.

[37] C. Minkenberg and G. R. Herrera. Trace-driven co-simulation of high-performance computing systems using omnet++. In *OMNeT++ 2009: Proceedings of the 2nd International Workshop on OMNeT++ (hosted by SIMUTools 2009)*, ICST, Brussels, Belgium, Belgium, 2009. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[38] N. Naksinehaboon, N. Taerat, C. Leangsuksun, C. F. Chandler, and S. L. Scott. Benefits of software rejuvenation on hpc systems. In *Proceedings of the International Symposium on Parallel and Distributed Processing with Applications*, ISPA '10, pages 499–506, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-0-7695-4190-7. doi: 10.1109/ISPA.2010.82. URL http://dx.doi.org/10.1109/ISPA.2010.82.

[39] T. Naughton, W. Bland, G. Vallée, C. Engelmann, and S. L. Scott. Fault injection framework for system resilience evaluation – Fake faults for finding future failures. In *Proceedings of the* 18^{th} *International Symposium on High Performance Distributed Computing (HPDC) 2009:* 2^{nd} *Workshop on Resiliency in High Performance Computing (Resilience) 2009*, pages 23–28, Munich, Germany, June 9, 2009. ACM Press, New York, NY, USA. ISBN 978-1-60558-587-1. URL http://doi.acm.org/10.1145/1552526.1552530.

[40] V. Pillet, J. Labarta, T. Cortes, and S. Girona. PARAVER: A Tool to Visualize and Analyze Parallel Code. In *Proceedings of WoTUG-18: Transputer and occam Developments*, pages 17–31, mar 1995.

[41] A. F. Rodrigues, K. S. Hemmert, B. W. Barrett, C. Kersey, R. Oldfield, M. Weston, R. Risen, J. Cook, P. Rosenfeld, E. CooperBalls, and B. Jacob. The structural simulation toolkit. *SIGMETRICS Perform. Eval. Rev.*, 38(4):37–42, Mar. 2011. ISSN 0163-5999. doi: 10.1145/1964218.1964225. URL http://doi.acm.org/10.1145/1964218.1964225.

[42] P. Rosenfeld, E. Cooper-Balis, and B. Jacob. Dramsim2: A cycle accurate memory system simulator. *Computer Architecture Letters*, 10(1):16–19, 2011. ISSN 1556-6056. doi: 10.1109/L-CA.2011.4.

[43] Sandia National Laboratories, Albuquerque, MN, USA. "Mantevo Project", 2012. https://software.sandia.gov/mantevo/.

[44] R. Sass, R. R. Sharma, and N. DeBardeleben. Towards a hardware fault-injection testbed to support reproducible resiliency experiments. In *Proceedings of the* 18^{th} *International Symposium on High Performance Distributed Computing (HPDC) 2009:* 2^{nd} *Workshop on Resiliency in High Performance Computing (Resilience) 2009*, pages 15–22, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-593-2. doi: http://doi.acm.org/10.1145/1552526.1552529.

[45] B. Schroeder and G. A. Gibson. Understanding failures in petascale computers. In *Journal of Physics: Proceedings of the Scientific Discovery through Advanced Computing Program (SciDAC) Conference 2007*, volume 78, pages 2022–2032, Boston, MA, USA, June 24-28, 2007. Institute of Physics Publishing, Bristol, UK. URL http://www.iop.org/EJ/abstract/1742-6596/78/1/012022.

[46] J. G. Silva, J. Carreira, H. Madeira, D. Costa, and F. Moreira. Experimental assessment of parallel systems. In *Proceedings of the 26th Annual International Symposium on Fault-Tolerant Computing*

(FTCS'96), pages 415–424, June 25-27, 1996.

[47] M. Snir, R. W. Wisniewski, J. A. Abraham, S. V. Adve, S. Bagchi, P. Balaji, B. Carlson, A. A. Chien, P. Diniz, C. Engelmann, R. Gupta, F. Johnson, J. Belak, P. Bose, F. Cappello, P. Coteus, N. A. Debardeleben, M. Erez, S. Fazzari, A. Geist, S. Krishnamoorthy, S. Leyffer, D. Liberty, S. Mitra, T. Munson, R. Schreiber, J. Stearley, and E. V. Hensbergen. Addressing failures in exascale computing. Workshop report, Apr. 2013. URL http://www.christian-engelmann.info/publications/snir13addressing.pdf.

[48] A. Tamches and B. P. Miller. Fine-Grained Dynamic Instrumentation of Commodity Operating System Kernels. In *Proceedings of 3rd Symposium on Operating Systems Design and Implementation (OSDI'99)*, Feb. 1999.

[49] C. Wang, F. Mueller, C. Engelmann, and S. L. Scott. A job pause service under LAM/MPI+BLCR for transparent fault tolerance. In *Proceedings of the* 21^{st} *IEEE International Parallel and Distributed Processing Symposium (IPDPS) 2007*, Long Beach, CA, USA, Mar. 26-30, 2007. ACM Press, New York, NY, USA. ISBN 978-1-59593-768-1. URL http://www.csm.ornl.gov/~engelman/publications/wang07job.pdf.

[50] C. Wang, F. Mueller, C. Engelmann, and S. L. Scott. Proactive process-level live migration in HPC environments. In *Proceedings of the IEEE/ACM International Conference on High Performance Computing, Networking, Storage and Analysis (SC) 2008*, Austin, TX, USA, Nov. 15-21, 2008. ACM Press, New York, NY, USA. ISBN 978-1-4244-2835-9. doi: http://doi.acm.org/10.1145/1413370.1413414. URL http://www.csm.ornl.gov/~engelman/publications/wang08proactive.pdf.

[51] C. Wang, F. Mueller, C. Engelmann, and S. L. Scott. Hybrid checkpointing for MPI jobs in HPC environments. In *Proceedings of the* 16^{th} *IEEE International Conference on Parallel and Distributed Systems (ICPADS) 2010*, pages 524–533, Shanghai, China, Dec. 8-10, 2010. IEEE Computer Society, Los Alamitos, CA, USA. ISBN 978-0-7695-4307-9. doi: http://dx.doi.org/10.1109/ICPADS.2010.48. URL http://www.christian-engelmann.info/publications/wang10hybrid2.pdf.

[52] C. Wang, F. Mueller, C. Engelmann, and S. L. Scott. Proactive process-level live migration and back migration in HPC environments. *Journal of Parallel and Distributed Computing (JPDC)*, 72(2):254–267, Feb. 2012. ISSN 0743-7315. doi: http://dx.doi.org/10.1016/j.jpdc.2011.10.009. URL http://www.christian-engelmann.info/publications/wang12proactive.pdf.

[53] D. Wang, B. Ganesh, N. Tuaycharoen, K. Baynes, A. Jaleel, and B. Jacob. Dramsim: a memory system simulator. *SIGARCH Comput. Archit. News*, 33(4):100–107, Nov. 2005. ISSN 0163-5964. doi: 10.1145/1105734.1105748. URL http://doi.acm.org/10.1145/1105734.1105748.

[54] G. Zheng, G. Kakulapati, and L. V. Kale. "BigSim: A parallel simulator for performance prediction of extremely large parallel machines". In *Proceedings of the* 18^{th} *IEEE International Parallel and Distributed Processing Symposium (IPDPS) 2004*, Santa Fe, New Mexico, Apr. 26-30, 2004. IEEE Computer Society. ISBN 0-7695-2132-0. URL http://doi.ieeecomputersociety.org/10.1109/IPDPS.2004.1303013.