Holistic Measurement Driven System Assessment

Saurabh Jha, Mike Showerman, Aaron Saxton,
Jeremy Enos, Greg Bauer, Zbigniew Kalbarczyk,
Ravi Iyer, William T. Kramer
UIUC/NCSA

Ann Gentile, Jim Brandt
Sandia National Lab

1 Introduction

The users of high performance computing (HPC) (which includes application developers, system managers, operational staff and vendors) seek the answer to following questions on a daily basis:

- Is application performance variation due to system conditions or code changes ?
- Is system having reliability and performance problems and what action needs to be taken to alleviate the problem ?
- What are the architectural requirements given the site's workload ?
- How can system provide more effective and efficient services ?

To answer these questions effectively, users deploy a suite of monitors to *observe* patterns of failures and performance anomalies to improve operational efficiency, achieve higher application performance and inform the design of future systems. However, the promises and the potential of monitoring data for the answering these questions have largely been not realized due to following challenges: (i) users deploy monitors independently neither leveraging the experience of other users (thus duplicating the effort) nor utilizing the datasets from deployed monitors of other users due to lack of domain knowledge, restriction on user capabilities because of privileged boundaries and vendor locking, (ii) gaps in monitoring and more importantly a user may not be completely aware of all the resources that must be monitored to detect a particular problem, (iii) the know-how of data collection mechanisms and the fusion of these datasets for identification of failure issues, performance drops and design bottlenecks are mostly unknown or impractical for large-scale systems and (iv) development of monitoring to mitigation framework life-cycle is highly manual. To address above challenges, we showcase the design and architecture of a monitoring fabric Holistic Measurement Driven System Assessment (HMDSA) for large-scale HPC facilities, independent of major component vendor, and within budget constraints of money, space, and power. We accomplish this through development and deployment of scalable, platform-independent, open-source tools and techniques for monitoring, coupled with statistical and machine-learning based runtime analysis and feedback, which enables highly efficient HPC system operation and usage and also informs future system improvements. We take a holistic approach through (c.f. 2):

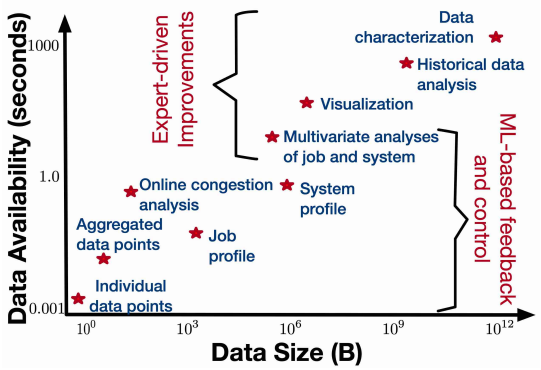

Figure 1. Velocity of data and analytics

- Monitoring of HPC infrastructure at various system granularities (e.g., atmospheric conditions, physical plant, HPC system components, application resource utilization) and fidelity (i.e., support variable collection frequency and feature selection) via existing and newly created monitors. The monitors are created and placed such that the observability in the system is maximized with minimal performance impact,
- Developing scalable storage, retrieval, and analytics platform to provide identification of performance impacting behaviors,
- Developing monitoring data fusion algorithms that are capable of working with both structure and unstructured datasets to detect and localize performance and reliability issues and design bottlenecks,
- Developing feedback and problem (e.g., faults, resource depletion, contention) mitigation strategies (e.g., rate limiting, scaling) and mechanisms targeting applications, system software, hardware, and users.

2 Design and Architecture Features

HMDSA provides unique capabilities for run-time system assessment and for transforming those insights into actionable knowledge. Features of HDMSA which make it unprecedented among operational and monitoring systems are:

Exascale ready. HMDSA's architecture and analysis capabilities are exascale-ready, having been demonstrated in production on current large-scale platforms up to 1/4 size of anticipated exascale platforms. Analysis on full-system, long-term data sets have demonstrated computation times for the most complex full-system analysis (a few minutes) down to

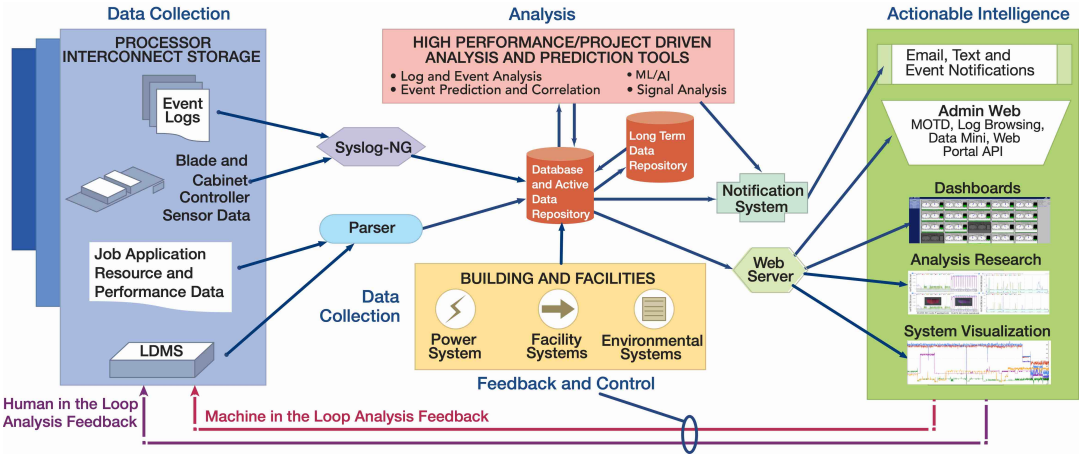

Figure 2. HMDSA Architecture

point-in-time analysis (sub-second) on runtime data (c.f. 1). Storage and retrieval have been demonstrated in support of analysis on several continuous years of full system data. We use Lightweight Distributed Metric Service (LDMS) [1] with host of custom plugins for data collection and transport, and Integrated System Console (ISC) [12] for storing, retrieving, analysis and visualization of datasets.

Unprecedented insights into system and application conditions. HMDSA analyses provide high resolution extraction and classification of phenomena with respect to locality, severity, and temporal extent. HMDSA high-fidelity data enables insights into phenomena occurring on timescales unresolvable by traditional monitoring capabilities [2–6, 8–10, 14]. To achieve actionable intelligence, we use variety of statistical tools and machine learning models (especially probabilistic graphical models) on structured (e.g., network counters) and unstructured datasets (e.g., console logs). These techniques are embedded in openly available tools such as Kaleidoscope [7] and Monet [10] (for structured datasets), and LogDiver [11] and Baler [13] for unstructured datasets.

Respond to system conditions at runtime. Flexible placement of analysis components and multi-directional communications enable low latency feedback of analysis results to system software, applications (machine-in-the-loop), and managers (human-in-the-loop). This allows action in response to system conditions while applications are still running. Examples include application re-balancing based on utilization imbalances, system scheduler decisions based on congestion areas or application profiles, and system administrator decision support.

No adverse application impact. HMDSA's lightweight, low overhead mechanisms enable high fidelity (e.g. sub-second) synchronized, whole-system numeric and event data capture with no adverse application performance impact.

Platform independent. Integration of site-specific resources and capabilties is easy. Operate all platforms in the same way, independent of site and vendor, using HMDSA's platform independent architecture. Site-specific data sources, storage, analysis, and dashboards can be used within or alongside HDMSA capabilities using HDMSA's flexible and scalable APIs and transports.

3 Conclusion and Future Work

In this work, we have shown the initial design and architecture of HMDSA components and described its intended use on HPC systems. Our future work involves: (i) porting HMDSA on existing and future HPC system, (ii) create installation/configuration packages that removes burden from the user, and (iii) work with vendors to ship the tool as a part of the HPC system software ecosystem. We are concurrently working on improving and adding more features to HMDSA on all aspects to support monitoring to mitigation via "human-in-the-loop" or "machine-in-the-loop" control loops.

References

[1] Agelastos, A., et al. Lightweight Distributed Metric Service: A Scalable Infrastructure for Continuous Monitoring of Large Scale Computing Systems and Applications. In *SC14: International Conference for High Performance Computing, Networking, Storage and Analysis* (2014), pp. 154–165.

[2] Brandt, J., Froese, E., Gentile, A., Kaplan, L., Allan, B., and Walsh, E. Network Performance Counter Monitoring and Analysis on the Cray XC Platform. In *Proc. Cray User's Group* (2016).

[3] Brandt, J., Gentile, A., Martin, C., Repik, J., and Taerat, N. New Systems, New Behaviors, New Patterns: Monitoring Insights from System Standup. In *Wrk. on Monitoring and Analysis for High Performance Computing Systems Plus Applications (HPCMASPA) Proc. IEEE Int'l Conf. on Cluster Computing (CLUSTER)* (2015).

[4] Di Martino, C., Baccanico, F., Fullop, J., Kramer, W., Kalbarczyk, Z., and Iyer, R. Lessons learned from the analysis of system failures at petascale: The case of blue waters. In *Proc. of 44th Annual IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN)* (2014).

[5] Di Martino, C., Kramer, W., Kalbarczyk, Z., and Iyer, R. Measuring and understanding extreme-scale application resilience: A field study

of 5,000,000 hpc application runs. In *Dependable Systems and Networks (DSN), 2015 45th Annual IEEE/IFIP International Conference on* (2015), IEEE, pp. 25–36.

[6] Jha, S., Brandt, J., Gentile, A., Kalbarczyk, Z., and Iyer, R. Characterizing supercomputer traffic networks through link-level analysis. In *2018 IEEE International Conference on Cluster Computing (CLUSTER)* (2018), IEEE, pp. 562–570.

[7] Jha, S., Cui, S., Xu, T., Enos, J., Showerman, M., Dalton, M., Kalbarczyk, Z. T., Kramer, W. T., and Iyer, R. K. Live forensics for distributed storage systems. *arXiv e-prints* (Jul 2019), arXiv:1907.10203.

[8] Jha, S., Formicola, V., Di Martino, C., Dalton, M., Kramer, W. T., Kalbarczyk, Z., and Iyer, R. K. Resiliency of HPC Interconnects: A Case Study of Interconnect Failures and Recovery in Blue Waters. *IEEE Transactions on Dependable and Secure Computing* (2017).

[9] Jha, S., Formicola, V., Kalbarczyk, Z., Di Martino, C., Kramer, W. T., and Iyer, R. K. Analysis of gemini interconnect recovery mechanisms: Methods and observations. In *CUG 2016 Conference* (2016), Cray User Group, pp. 8–12.

[10] Jha, S., Patke, A., Lim, B., Brandt, J., Gentile, A., Bauer, G., Showerman, M., Kaplan, L., Kalbarczyk, Z., Kramer, W. T., and Iyer, R. Measuring congestion in high-performance datacenter interconnects. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)* (Feb 2020).

[11] Martino, C. D., Jha, S., Kramer, W., Kalbarczyk, Z., and Iyer, R. K. Logdiver: a tool for measuring resilience of extreme-scale systems and applications. In *Proceedings of the 5th Workshop on Fault Tolerance for HPC at eXtreme Scale* (2015), ACM, pp. 11–18.

[12] Semeraro, B. D., Sisneros, R., Fullop, J., and Bauer, G. H. It takes a village: Monitoring the blue waters supercomputer. In *2014 IEEE International Conference on Cluster Computing (CLUSTER)* (Sep. 2014), pp. 392–399.

[13] Taerat, N., Brandt, J., Gentile, A., Wong, M., and Leangsuksun, C. Baler: deterministic, lossless log message clustering tool. *Computer Science - Research and Development 26*, 3-4 (2011), 285–295.

[14] Tuncer, O., Ates, E., Zhang, Y., Turk, A., Brandt, J., Leung, V., Egele, M., and Coskun, A. Diagnosing Performance Variations in HPC Applications Using Machine Learning. In *Proc. ISC High Performance 2017 (ISC)* (2017).