International Journal of Electronics and Communication Engineering.
ISSN 0974-2166 Volume 16, Number 1 (2023), pp. 31-41

http://www.irphouse.com

Integrating Monitoring Tools for Enhanced Storage System Management: A Focus on Primera, Alletra, and Distributed Storage Systems with Odin

Akshay Shankar and Prof. Prapulla S B

Dept of Computer Science and Engineering,
R.V. College of Engineering Bengaluru-5600592

Abstract

Efficient management of storage systems is essential for maintaining optimal performance and availability. This paper explores monitoring tools for Primera, Alletra, and distributed storage systems, along with their integration with Odin, a comprehensive storage management platform. The paper discusses the necessity of monitoring tools and emphasizes specific Python scripts developed for monitoring critical metrics in storage systems. These scripts, which log into the arrays, cover areas such as CPU utilization, CPG utilization, cache monitoring, credit cache memory page, and memory starvation.

The methodology employed in this project involves two main aspects: tool development and integration with the ODIN platform. The first step involves designing and implementing monitoring scripts using Python scripting. These scripts capture essential metrics such as CPU utilization, CPG utilization, cache monitoring, and memory performance. Special emphasis is placed on script robustness, efficiency, and the ability to retrieve data from storage arrays. The second step focuses on exploring the integration capabilities of the ODIN platform.

The outcomes of this project are geared towards improving storage system management and performance. By integrating the monitoring tools with ODIN, administrators gain access to real-time insights into storage performance, enabling them to proactively identify and address performance anomalies and resource utilization issues.

Keywords: Odin, Monitoring tools, cache monitoring, CPU Utilization, CPG Utilization, Alletra, Primera

1. INTRODUCTION

The main storage systems this paper focuses on are Alletra and Primera. Primera is a storage platform offered by Hewlett Packard Enterprise (HPE) that combines the power of intelligence and simplicity. It is designed to deliver high-performance and highly available storage for mission-critical applications. Primera incorporates HPE InfoSight, an AI-driven analytics platform, which provides proactive monitoring, predictive analytics, and comprehensive insights into the health and performance of the storage system. With features such as advanced data reduction, resiliency, and scalable architecture, Primera offers a robust and efficient storage solution for organizations requiring enterprise-class performance and reliability.

Alletra is a storage system by NetApp that delivers industry-leading performance, scalability, and agility. It is designed to support a wide range of workloads, from traditional applications to modern, data-intensive workloads. Alletra leverages NetApp's expertise in data management and provides advanced features like NVMe-based storage, inline data deduplication, and compression to optimize storage efficiency. Additionally, Alletra integrates with NetApp Active IQ, a cloud-based analytics and monitoring platform, which offers predictive analytics, proactive support, and detailed performance monitoring. With Alletra, organizations can achieve high-performance storage with enhanced data management capabilities for their evolving business needs.

2. Literature review

Gupta, A. K., Shinde, S., Bakshi, P. (2021). "Survey of Open Source Tools for Monitoring I/O & Storage Performance of HPC Systems." International Journal of Computer Techniques, 8(1). This paper provides a survey of open-source tools for monitoring I/O and storage performance in High-Performance Computing (HPC) systems. It explores various tools, their features, and their applicability in HPC environments. The study aims to help researchers and system administrators in selecting appropriate monitoring tools for effectively managing storage performance in HPC systems.[1]

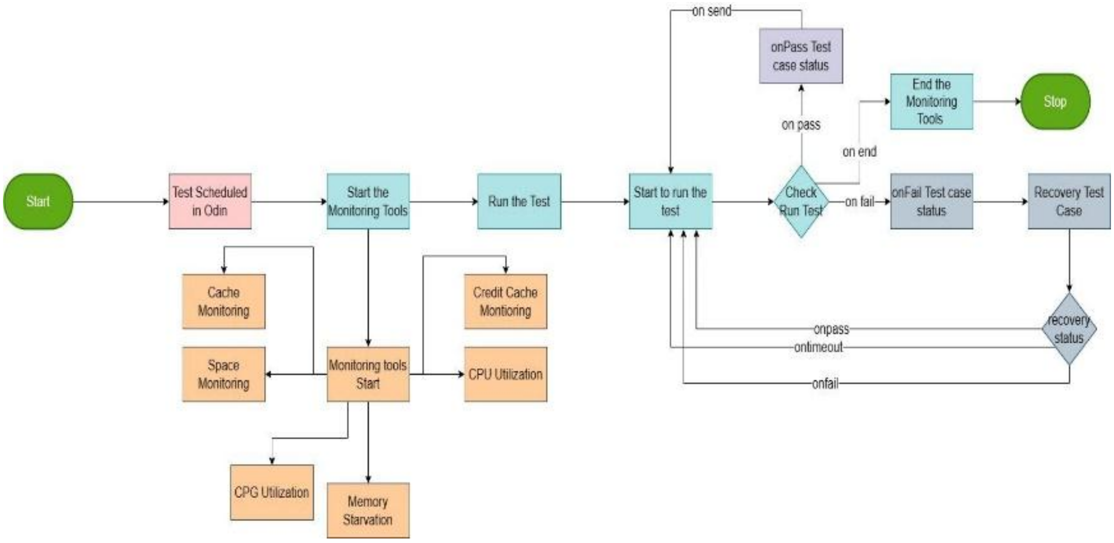
Ganger, G. R., Worthington, B. L., Hou, R. Y., Patt, Y. N. (1994). "Disk arrays: high-performance, high-reliability storage subsystems." IEEE. This IEEE paper introduces the concept of disk arrays as high-performance and high-reliability storage subsystems. It discusses various architectural approaches, fault tolerance mechanisms, and performance optimizations employed in disk arrays. The paper serves as a foundational work in the field of storage subsystems, providing insights into the design and management of storage arrays.[2]

Gu, M., Li, X., Cao, Y. (2014). "Optical storage arrays: a perspective for future big data storage." Light: Science & Applications. In this paper, the authors present a perspective on optical storage arrays as a potential solution for future big data storage. They discuss the advantages of optical storage technology, such as high capacity, long-term data retention, and energy efficiency. The paper explores the challenges and opportunities in the development and implementation of optical storage arrays for managing large-scale data storage requirements.[3]

Cherry, D. (2015). "Storage Area Network Security." Securing SQL Server (Third Edition).
This book chapter by Denny Cherry focuses on the security aspects of Storage Area Networks (SANs). It discusses the potential risks and vulnerabilities associated with SANs and provides insights into securing SAN infrastructure and data. The chapter covers topics like access control, encryption, zoning, and auditing to ensure the security of SAN-based storage systems.[4]
Schulz, M., Tao, J., Jeitner, J., Karl, W. (2002). "A Proposal for a New Hardware Cache Monitoring Architecture." This paper presents a proposal for a new hardware cache monitoring architecture. It discusses the limitations of existing cache monitoring techniques and proposes a novel approach that enables efficient monitoring of cache utilization and performance. The proposed architecture aims to provide accurate and real-time cache monitoring capabilities for optimizing cache utilization in computer systems.[5]
Westman, A. (2022). "Evaluation of cache memory configurations with performance monitoring in embedded real-time automotive systems." This paper focuses on the evaluation of cache memory configurations in embedded real-time automotive systems. It investigates the impact of different cache configurations on system performance and proposes a methodology for performance monitoring and analysis. The study aims to provide insights into cache memory design considerations and optimization techniques for improving the performance of automotive systems.[6]
Chekkilla, A. G., Kalidindi, R. V. (2016). "Monitoring and Analysis of CPU Utilization, Disk Throughput and Latency in servers running Cassandra database." This paper discusses the monitoring and analysis of CPU utilization, disk throughput, and latency in servers running the Cassandra database. It presents a monitoring framework and methodology for capturing and analyzing these performance metrics to identify potential bottlenecks and optimize the performance of Cassandra database deployments.[7]
Wang, J., Yan, Y., Guo, J. (2016). "Research on the Prediction Model of CPU Utilization Based on ARIMA-BP Neural Network." MATEC Web of Conferences. This paper explores the prediction model of CPU utilization based on the ARIMA-BP (Autoregressive Integrated Moving Average-Backpropagation) neural network. It discusses the application of time series analysis and neural networks for predicting CPU utilization patterns. The study aims to develop an accurate and reliable model for CPU utilization prediction, which can aid in resource allocation and capacity planning in computing systems.[8]

3. Proposed System Architecture

Figure 1, depicts the proposed system architecture of the monitoring tools and how it works with Odin. Odin is a comprehensive storage management platform that empowers organizations to efficiently manage and monitor their storage systems. It provides a centralized interface and a suite of powerful tools for administrators to streamline storage operations, optimize performance, and ensure high availability. Odin offers features such as automated provisioning, capacity planning, performance monitoring, and advanced analytics. With its intuitive user interface and robust functionalities, Odin simplifies storage management tasks and enables administrators to gain valuable insights into the health and performance of their storage infrastructure. By integrating monitoring tools, like those developed in this project, with Odin, administrators can leverage its capabilities to obtain a unified view of storage system health, make informed decisions for performance optimization, and efficiently resolve any issues that arise.



Before an Odin test is run, the necessary components need to be added in order to run a test successfully and automatically. The necessary array needs to be added with the required parameters such as the serial ID, the number of nodes and other details. The host virtual machine which is used for the array is added to Odin. An array requires a host to establish a connection and facilitate communication between the storage array and the computing environment. The host serves as the intermediary between the storage array and the applications or operating systems running on the host server. Once these are added and ready, the connections between the host and the array must be created, there are two ways to do that, one create a CFG file for the array so that the port connections are automatically created in the test ring. The second way is to create and add a new component called a switch and include this into the test ring. The presence of a switch between an array and a host in storage is primarily to facilitate connectivity and communication between multiple hosts and the storage array. The addition of switches gives higher scalability, traffic segmentation, fault isolation, etc. Once these are complete, the necessary components are present for Odin to schedule a test. To run any test in Odin, it is essential that there is a workflow present.

When the workflow is created in Odin it needs to be in a consistent state to be saved. If the workflow is saved, then it can be added to the Test Ring. When a test is scheduled all the components such as the test plan, test ring, and virtual machine must be selected and then the test can run. In the proposed system, when a test is scheduled in Odin the system will first start running the monitoring tools, Monitoring needs to happen in the background when a test is running in storage to ensure the continuous health, performance, and availability of the storage infrastructure. The monitoring tools will return complete so Odin can continue with the actual test because Odin can run one test at a particular time and does not support parallel scheduling of tests. Due to Odin being serial, the monitoring tools will show it as though the monitoring has been completed and the test has begun. As the test is running in the front, the background various features of the array are being monitored.

Once the test is complete then the workflow returns to the monitoring tools to stop the monitoring of the array. In this way, the integration of monitoring tools and Odin is successfully done. The need for this integration is:

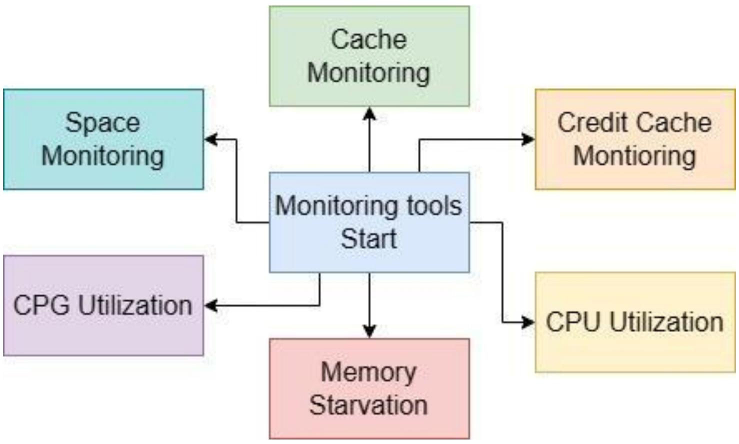
Centralized Management: Integrating monitoring tools with Odin provides a centralized management platform for storage systems. Administrators can access and monitor various aspects of the storage infrastructure, including performance metrics, resource utilization, and system health, all within the Odin interface. This centralized view simplifies management tasks and reduces the need to switch between multiple monitoring tools or interfaces.

Streamlined Workflows: Integration allows for the automation and streamlining of workflows within Odin. Monitoring tools can be seamlessly integrated into the test execution process, triggering monitoring at the start of a test and concluding it upon completion. This integration eliminates the need for manual intervention and ensures that monitoring is consistently performed during critical storage operations, reducing the risk of oversight or missed monitoring opportunities.

Real-time Visibility: The integration of monitoring tools with Odin enables real-time visibility into the health and performance of storage systems. Administrators can monitor key metrics and receive immediate alerts or notifications if any issues or abnormalities are detected. This real-time visibility allows for proactive monitoring and prompt action to resolve issues, minimizing the impact on storage operations and ensuring optimal system performance.

4. Monitoring Tools of Storage Solutions

In this paper, the main focus of monitoring is CPU monitoring, CPG monitoring, Cache monitoring, memory starvation, and space monitoring



4.1 CPU Monitoring

Performance Optimization: The CPU is a critical resource in storage arrays, responsible for processing storage-related operations and managing various tasks. Monitoring CPU utilization helps administrators assess the workload placed on the CPU and identify potential bottlenecks or inefficiencies. By monitoring CPU usage and load trends, administrators can optimize CPU resource allocation, balance workloads, and ensure that the CPU is effectively utilized to maximize system performance.

Capacity Planning: CPU utilization monitoring aids in capacity planning for storage arrays. By analyzing CPU utilization patterns and trends over time, administrators can assess the CPU capacity requirements of the storage environment. This monitoring data helps in determining whether the allocated CPU resources are sufficient to handle the current workloads and whether additional CPU resources are necessary to accommodate future growth or increased performance demands.

4.2 CPG Utilization

Capacity Planning: CPG monitoring helps administrators in capacity planning for storage arrays. A CPG represents a logical grouping of physical disks or RAID groups within the storage array. By monitoring CPG utilization and capacity trends, administrators can assess the storage capacity requirements of different CPGs and make informed decisions regarding storage provisioning and allocation. This monitoring data helps ensure that sufficient storage capacity is available to meet current and future demands.

Resource Allocation: Storage arrays often serve multiple workloads and applications that have varying performance and capacity requirements. CPG monitoring allows administrators to evaluate the allocation of storage resources among different CPGs. By monitoring CPG usage and analyzing performance metrics, administrators can optimize resource allocation, ensure fair distribution of storage capacity and performance, and prevent resource contention or bottlenecks.

4.3 Cache Monitoring

Performance Optimization: The cache plays a crucial role in improving storage system performance by serving as a high-speed buffer between the slower disk-based storage and the faster host applications. Monitoring the cache utilization and performance allows administrators to assess its effectiveness in accelerating data access and optimizing I/O operations. By monitoring cache hit ratios, cache usage, and cache latency, administrators can identify any potential bottlenecks or inefficiencies in the cache subsystem and take appropriate actions to optimize performance.

Capacity Planning: Cache monitoring helps in capacity planning for storage arrays. By monitoring cache usage patterns and trends over time, administrators can analyze the cache capacity requirements and make informed decisions regarding cache allocation and sizing. This monitoring data allows them to determine whether the cache size is adequate for the workload and whether adjustments or upgrades are necessary to meet performance demands.

4.4 Memory Starvation

Optimal System Performance: Memory plays a critical role in storage arrays by facilitating efficient data access, caching, and buffering. Monitoring memory utilization helps administrators ensure that the storage array has sufficient memory resources to handle its operations effectively. Memory starvation monitoring allows them to identify situations where the available memory is insufficient to meet the demands of the storage workload. By detecting memory shortages or high memory utilization, administrators can take appropriate actions to alleviate memory starvation and optimize system performance.

Resource Allocation: Storage arrays typically serve multiple concurrent workloads and applications. Monitoring memory utilization helps administrators assess the allocation of memory resources to different processes or applications within the storage array. By identifying situations of memory starvation, where certain processes or applications are not receiving adequate memory resources, administrators can adjust the memory allocation to ensure fair distribution and optimal performance across all workloads.

4.5 Space Monitoring

Capacity Planning: Space monitoring helps administrators in capacity planning for storage arrays. By monitoring the utilization of storage space, administrators can track the growth rate of data and accurately estimate future storage capacity requirements. This monitoring data enables them to make informed decisions about storage expansion, provisioning, or migration, ensuring that sufficient space is available to accommodate the storage needs of the organization.

Resource Allocation: Storage arrays often support multiple workloads and applications with varying storage requirements. Space monitoring allows administrators to assess the allocation of storage resources among different applications or users. By monitoring space utilization on a per-application or per-user basis, administrators can ensure fair distribution of storage capacity, prevent resource contention, and optimize storage resource allocation based on the specific needs of each workload.

5. Results and Discussion

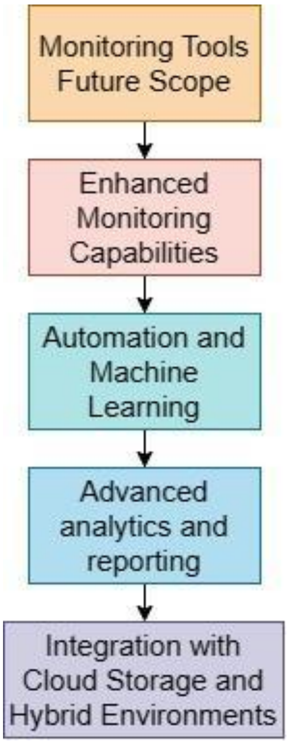
The implementation of monitoring tools integrating with ODIN for Alletra, Primera, and Distributed Storage Systems has yielded promising results in terms of enhancing storage system management and performance. The monitoring tools developed in this project have successfully captured critical metrics such as CPU utilization, CPG utilization, cache monitoring, and memory performance. Through seamless integration with the ODIN platform, administrators have gained real-time insights into the health, performance, and resource utilization of their storage arrays.

The monitoring tools have enabled administrators to proactively identify and address performance anomalies and resource utilization issues. By receiving timely alerts and notifications, administrators can take immediate action to resolve issues, minimizing downtime and maximizing system availability. The resource optimization recommendations provided by the monitoring tools have proven valuable in workload balancing, capacity planning, and resource allocation optimizations. Administrators have been able to optimize resource utilization, ensuring efficient utilization of storage arrays and preventing overutilization or underutilization of resources.

Furthermore, the integration of the monitoring tools with the ODIN platform has streamlined the storage management process. Administrators can leverage the centralized interface and workflows of ODIN to initiate and conclude monitoring activities seamlessly. The unified view of storage system health provided by ODIN simplifies management tasks and facilitates proactive decision-making.

6. Future Scope

The future scope of your project on monitoring tools for Primera, Alletra, and distributed storage systems, integrated with Odin, holds significant potential for further development and expansion. Enhanced Monitoring Capabilities, continuously evolving storage technologies and architectures may introduce new metrics and parameters that need to be monitored. Adding Automation and Machine Learning techniques into the monitoring framework can enhance the efficiency and accuracy of monitoring tasks. This could involve developing algorithms or models that analyze monitoring data to detect patterns, anomalies, or potential issues automatically. By leveraging machine learning, the monitoring system can provide predictive insights, proactive recommendations, or automated actions for optimizing storage performance and resource management. Expanding the capabilities of the monitoring tools to include advanced analytics and reporting features can provide administrators with valuable insights and trends related to storage performance, resource utilization, and system health.



As organizations increasingly adopt cloud storage solutions and hybrid storage architectures, integrating the monitoring tools with these environments becomes crucial. The future scope of your project could involve extending the monitoring capabilities to encompass cloud-based storage systems, enabling administrators to monitor and manage both on-premises and cloud storage resources from a unified interface. With growing concerns around data security and compliance, incorporating monitoring tools that specifically address security-related aspects can be a valuable addition. This could involve monitoring for security events, vulnerabilities, access control violations, or compliance breaches within the storage environment. The integration with Odin can facilitate centralized security monitoring and reporting, enabling administrators to ensure the integrity and confidentiality of stored data.

These future directions expand the possibilities of your project, allowing for continuous innovation and alignment with the evolving needs of storage systems and IT infrastructure management. By exploring these areas, you can contribute to the advancement of storage monitoring and management practices, enabling administrators to effectively optimize storage performance, ensure data integrity, and streamline operations.

7. Conclusion

This project has explored the monitoring tools for Primera, Alletra, and distributed storage systems, and their integration with Odin, a comprehensive storage management platform. The project has highlighted the necessity of monitoring tools in storage environments and emphasized the specific Python scripts developed for monitoring critical metrics such as CPU utilization, CPG utilization, cache monitoring, credit cache memory page, and memory starvation. These monitoring tools, which enable administrators to log into the arrays and gather real-time data, provide invaluable insights into the health, performance, and resource utilization of storage systems.

The integration of these monitoring tools with Odin has been a significant achievement, as it offers a unified platform for storage system management. By incorporating the monitoring tools within Odin workflows, administrators can initiate and conclude monitoring processes seamlessly during storage tests and operations. This integration empowers administrators with a holistic view of storage system health, allowing them to make informed decisions for performance optimization, capacity planning, resource allocation, and issue resolution. The benefits of this integration include improved resource utilization, proactive issue identification, and efficient storage management, ultimately leading to enhanced system performance, stability, and user satisfaction.

By continuing to evolve and adapt the monitoring tools and integration with Odin, administrators can stay ahead of the evolving storage landscape, address emerging challenges, and optimize the management of storage systems for the benefit of organizations relying on efficient and reliable data storage and retrieval.

8. References

[1] Anil Kumar Gupta, Sakshat Shinde, Puneet Bakshi, "Survey of Open Source Tools for Monitoring I/O & Storage Performance of HPC Systems", International Journal of Computer Techniques, Volume 8 Issue 1, January 2021.

[2] G.R. Ganger, B.L. Worthington, R.Y. Hou, Y.N. Patt, "Disk arrays: high-performance, high-reliability storage subsystems", IEEE, March 1994.

[3] Min Gu, Xiangping Li, Yaoyu Cao, "Optical storage arrays: a perspective for future big data storage", Light: Science & Applications, May 2014.

[4] Denny Cherry, Storage Area Network Security, Securing SQL Server (Third Edition), 2015

[5] Martin Schulz, Jie Tao, Jürgen Jeitner, Wolfgang Karl, "A Proposal for a New Hardware Cache Monitoring Architecture", September 2002

[6] Andreas Westman, "Evaluation of cache memory configurations with performance monitoring in embedded real-time automotive systems", June 2022

[7] Avinash Goud Chekkilla, Rajeev Varma Kalidindi, "Monitoring and Analysis of CPU Utilization, Disk Throughput and Latency in servers running Cassandra database", June 2016

[8] Jina Wang, Yongming Yan, Jun Guo, "Research on the Prediction Model of CPU Utilization Based on ARIMA-BP Neural Network", MATEC Web of Conferences, January 2016.

[9] R. Wilhelm and J. Reineke, "Embedded systems: Many cores — many problems," in 7th IEEE International Symposium on Industrial Embedded Systems (SIES'12), 2012. doi: 10.1109/SIES.2012.6356583 pp. 176–180. [Page 1.]

[10] D. Comer, Essentials of Computer Architecture, second edition. Chapman and Hall/CRC, 2017. [Pages 1, 15, and 16.] [3] S. Przybylski, M. Horowitz, and J. Hennessy, "Performance tradeoffs in cache design," in [1988] The 15th Annual International Symposium on Computer Architecture. Conference Proceedings, 1988. doi: 10.1109/ISCA.1988.5239 pp. 290–298. [Page 1.]

[11] Book E: Enhanced PowerPC Architecture Version 1.0, NXP Semiconductors, 2002. [Page 7.]

[12] Reference Manual, Rev. 8, 11/2016, NXP Semiconductors, 2016. [Page 7.]

[13] Core Reference Manual, Rev. 2, NXP Semiconductors, 2015. [Pages 8, 12, 15, 16, 26, 28, and 53.]

[14] S. Basumallick and K. Nilsen, "Cache issues in real-time systems," in ACM SIGPLAN Workshop on Language, Compiler, and Tool Support for Real-Time Systems, vol. 5. Citeseer, 1994. [Page 11.]

[15] F. Sebek, "Instruction cache memory issues in real-time systems," Tech. Rep. ISSN 1404-3041 ISRN MDH-MRTC-60/2002-1-SE, October 2002

[16] S. Altmeyer, R. Douma, W. Lunniss, and R. I. Davis, "Outstanding paper: Evaluation of cache partitioning for hard real-time systems,"

[17] Ministry of Electronics & Information Technology. 2017. High-Performance Computing (HPC). [online]

[18] Shuibing He, Xian-He Sun, Yanlong Yin "BPS: A Performance Metric of I/O System" 2013 IEEE 27th International Symposium.

[19] J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach: Morgan Kaufmann Pub, 4th edition, 2011.
[20] D. Bitton, M. Brown, R. Catell, S. Ceri, T. Chou, D. DeWitt, D. Gawlick, H. Garcia-Molina, B. Good, and J. Gray, "A Measure of Transaction Processing Power," Datamation, vol. 31, pp. 112-118, 1985.
[21] Stanisic L., Reuter K. (2020) MPCDF HPC Performance Monitoring System: Enabling Insight via Job-Specific Analysis. In: Schwardmann U. et al. (eds) Euro-Par 2019: Parallel Processing Workshops. Euro-Par 2019. Lecture Notes in Computer Science, vol 11997. Springer, Cham

Authors

Akshay Shankar is currently pursing his B.E in computer science in R.V. College of Engineering, Bangalore. He was born and raised in Bangalore India. In Engineering he has worked many projects related to Computer Graphics, Artificial Intelligence, Big Data, etc. He also indulged in club activities in his college, particularly he was very activity in coding club. He is currently interning at Hewlett Packard Enterprise. He can be contacted at email: akshayshankar.cs19@rvce.edu.in

Dr. Prapulla S B is an accomplished academician and researcher with a diverse range of achievements in the field of computer science and engineering. With a track record of guiding numerous undergraduate and postgraduate projects, Prapulla has shown a commitment to mentoring and shaping the next generation of researchers. Their extensive publication record includes papers in national conferences, international journals, and presentations at prestigious international conferences. Prapulla's research interests span various areas, including image compression algorithms, mobile agent data security, smart technologies, and cloud architectures. prapullasb@rvce.edu.in