ISSN (online): 2349-784X

A Holistic Approach for Performance Analysis of Embedded Storage Array

Naveenkumar Jayakumar
Research Scholar
Department of Computer Engineering
Bharati Vidyapeeth deemed University,
College of Engineering, Pune - 43

Tanya Bhardwaj
UG Student
Department of Computer Engineering
Bharati Vidyapeeth deemed University,
College of Engineering, Pune - 43

Kamakshi Pant
UG Student
Department of Computer Engineering
Bharati Vidyapeeth deemed University,
College of Engineering, Pune - 43

Dr. S.D. Joshi
Professor
Department of Computer Engineering
Bharati Vidyapeeth deemed University,
College of Engineering, Pune - 43

Suhas. H. Patil
Professor
Department of Computer Engineering
Bharati Vidyapeeth deemed University, College of Engineering, Pune - 43

Abstract

There is an increase in data intensive applications now a days. With this popularity of data intensive applications, the data requests gets flooded in the network and creates a bottleneck near the storage disks. The existing architecture contains different set of storage nodes with huge amount of disk capacity and data scattered among these storage nodes. The Application request is broken down into multiple I/O requests and routed to various storage nodes to get serviced. When considering the performance modelling of such storage nodes in network, researchers tend to propose the performance modelling of the system from the perspective of application and factors hindering the functioning of applications. None of the papers have ever discussed in detail how and what approach was adhered to realise the proposed performance modelling. This paper proposes a holistic approach for validating the proposed performance model of an embedded storage array. The holistic approach will mostly cover how the configuration and component architecture of storage array and tools are to be used for validation of mathematical model along with application workloads.
Keywords: Embedded System, Storage Array, Performance Approach, Distributed System, Parallel System, Active Storage

I. INTRODUCTION

The storage array is a system with huge amount of disk capacity and high end processors acting as the disk controllers for acknowledging the I/O intensive requests and responding to the application host node[1]. The storage array has deep I/O stack layers[2], [3] through which the request needs to be flown and at the same time the request needs to be broken down into multiple I/O blocks which increases the input output operations per second. These I/O blocks will vary according to the application executing at the server end. The applications have different workloads. These workload variations will vary the I/O request throughout the I/O stack within the storage array.

The storage array stores data in the raid disks. Since the data are striped along the various disks within the array and these disks may not contain the whole data. The data may be partially stored on each of the LUN i.e. logical unit number, which the volume evolved from one or multiple stripe on disks. The storage array was embedded with a storage operating system which has the data manager and some specific storage functions which are automatically managed by the Operating system once the data is stored in the disks. The storage and retrieval process is efficiently managed by this storage operating system.

Since there is burst in data processing and data volume, the disk storage array are becoming more and more bottleneck for the operations read and write. The storage disks gets saturated after some I/O requests which increases the latency time and reduces the performance of application. The waiting time increases as the disk speed is saturated.

Now days to provide more disk oriented resources virtualization are being used. The virtualization is a container with device drivers for various hardware platform and hypervisor which supports multiple operating system to work on same metal hardware as per the device driver support. These virtualization[4], [5] in turn add one more layer to the I/O stack in the storage array. They increase the latency time and as well distribute and share the same physical resources to multiple application requests directed to multiple logical resources.

In this environment, performance becomes the buzz word. How well the virtualized storage array will perform? What kind of analysis is required in order to sustain the workloads in the given environment? The performance of the storage array will vary from application to application and time to time. They are never constant. But what can be the same in performance analysis is the process adhered to analyse the performance of the system.

A storage array will be having set of layers through which the application request needs to be moved to get processed within the storage array. The active storage concept is about utilizing the disk controllers to its maximum extent. The disk controllers remain unused for longer period since they are only involved in storage functions. The disk controllers of the array can be utilized by virtualizing the storage array and embedding the user end application in the array. The embedding can be done along with the container getting migrated from the application server to the storage node or only the data intensive part of the application.

II. HOLISTIC APPROACH

Considering the performance of such system, it is necessary to view the configuration and architecture of the storage array. When simulating the similar array architecture performance measurement is done through some tools like iometer. The various factors considered while analysing the performance of storage systems are shown in figure 1.

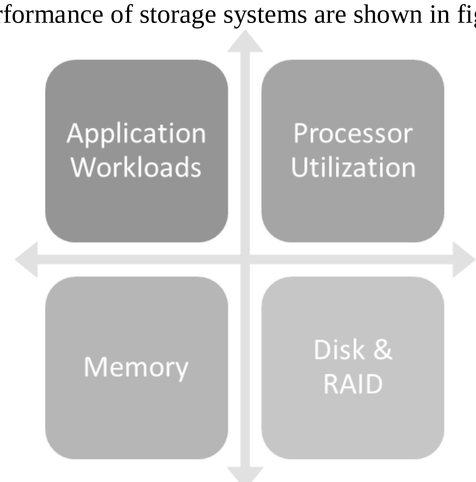

Fig. 1: Factors Influencing Performance of Storage Array

Many researched have proposed various math formulae related to performance of the processor, memory, application and disks. But no paper have discussed on approach to evaluate and compare the theoretical formulae on performance. The current architecture of the storage system with embedded compute is shown in the figure 2.

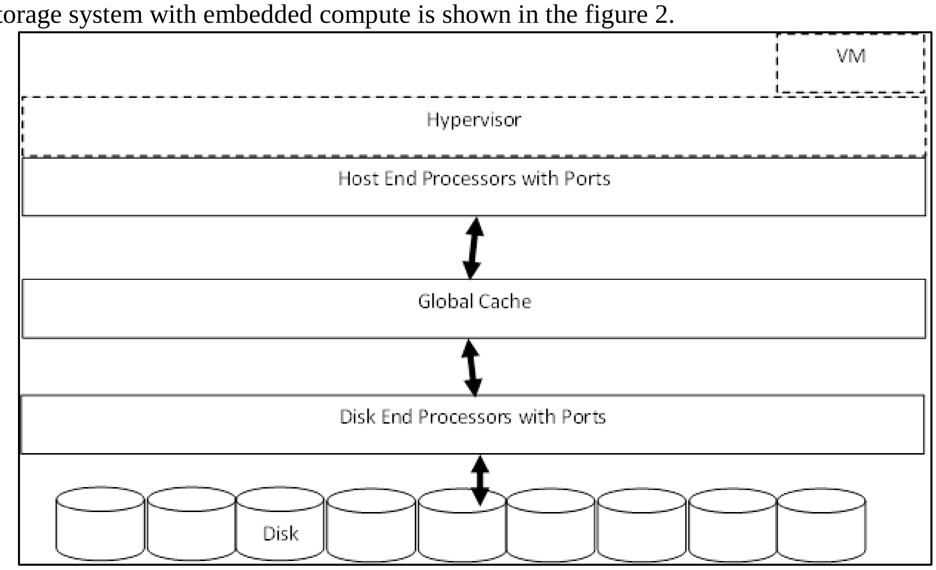

Fig. 2: Storage Array Generic Components [1]

In figure 2 the dotted line is the logical component and the strong line is hardware components of the storage array. Any I/O request will be received at the Host end ports then flow through guest operating system, acknowledged by processor one at a time, some getting served at cache and some gets split into multiple I/O requests which are further received by the file systems and then server at the disk through disk controllers. [6]–[8]

The proposed approach starts with understanding the workload to be applied to array system. The workloads vary according to the application and its characteristics. The approach is as follows:

1) Step 1 Configure the storage system. In order to understand the Memory impact on performance, while creating the virtual machine set the memory or ram amount of 1GB to the virtual machine. Keep the other components like number of processors, number of cores, % read and % write, I/O size constant.
2) Step 2 Execute the workloads on the current setup for a specific time interval.
3) Step 3 Monitor and note down the metrics like % Hit ratio, %Request served, time taken and I/O size.
4) Step 4 Repeat step 1 by increasing the memory capacity and keeping the other values constant.
5) Step 5 repeat step 2 and 3.

Repeat the above steps until maximum memory is reached. From the monitored values of metrics, it is possible to correlate the actual measured metrics with the theoretical equations and propose the viable solution to improve the performance via efficient understanding of locality of reference.

The above steps are also executed for the processors and disks.

1) Step 1 Configure the storage system. In order to understand the processor/core impact on performance, while creating the virtual machine set the processor and cores quantity to one for virtual machine. Keep the other components like memory, % read and % write, I/O size constant.
2) Step 2 Execute the workloads on the current setup for a specific time interval.
3) Step 3 Monitor and note down the metrics like % hit ratio, %Request served, time taken and I/O size, MB/sec, IOPS.
4) Step 4 Repeat step 1 by increasing the quantity of cores first followed by the quantity of processor. Each increment should have the complete set of execution and monitoring the metrics.
5) Step 5 repeat step 2 and 3.

These steps will be repeated until the maximum processors and cores are reached. For each increment of the component quantity, the monitoring will yield with different set of metric values. These values can be further analysed in order to understand the utilization pattern of cores and processors of the storage system.

The workload generator can be a tool installed at the guest operating system embedded into the storage array. The tool like Iometer.[1] Workload is a logical classification of work that can be performed on a computer system. Workload is basically is an independent service or a collection of code that can be executed. Another way to look at it is the amount of work that needs to be accomplished by computer resources in a certain period of time.

Some industry experts include the application, operating system, and middleware in their definition of a workload. Of course, different workloads have different characteristics, and the best platform for a particular workload to run on depends on the nature of the specific workload.

Because computing requirements are varied, so are the workloads. This list explains some of the kinds of workloads you might find in a hybrid cloud environment, and the table that follows compares them to each other:

1) Batch workloads
2) Transactional workloads
3) Analytic workloads
4) High-performance workloads
5) Database workloads

Iometer is generally used to measure the I/O performance but it can also generate the workload that is used to test the system .This part is also called Dynamo. The dynamo component must be installed on each system for which the performance results are required. It can be configured to emulate the disk or network I/O load of any program or benchmark or can be used to generate entirely synthetic I/O load. It can generate and measure loads on single and multiple system. At iometer command, dynamo performs I/O operations and records performance information, then returns the data to the iometer. [2][3]

To calculate the workload universally we need to associate unit of work with it. This is a measurable quantity of work done. The final step is to form a service level agreement between the organization providing the services and the customer for which the services are intended. Workloads often help in developing these agreements as they measure the system performance. Customer requirements are the main factor which are considered as a major factor while forming the service level agreement.

III. CONCLUSION

This paper focuses on proposing an approach to evaluate the testbed for verifying the storage array performance analytical model. In general, the performance model is of storage system contains three main metrics, IOPS, bandwidth and Latency. These three measurements will yield you the performance and applicability, reliability of the storage system. All the research papers till now have explained the theoretical performance analytical model of the storage and its component's but none of the research paper have discussed in detail the approach of evaluating the proposed model through simulating or executing the testbed configuration on storage array. This paper relates the testbed configuration of components of storage array and how to apply the various workloads and monitor the change to these components. The variation is made only to one component under the hood at single time of execution. This will yield in understanding in depth impact analysis of the workloads on various components of the storage node.

REFERENCES

[1] D. T. Jayakumar, Naveenkumar, Raj, SDjoshi, "International Journal of Advanced Research in Computer Science and Software Engineering," Int. J., vol. 2, no. 9, pp. 62 – 70, 2012.
[2] O. F. Computer, "© i a e m e," pp. 46–51, 2014.
[3] N. Jayakumar, "International Journal of Advanced Research in Computer Science and Software Engineering," vol. 3, no. 5, pp. 991–996, 2013.
[4] "Storage Systems Performance Testing."
[5] L. Cherkasova and R. Gardner, "Measuring CPU overhead for I/O processing in the Xen virtual machine monitor," Proc. USENIX Annu. …, pp. 387–390, 2005.
[6] S. W. Son, S. Lang, P. Carns, R. Ross, R. Thakur, B. Ozisikyilmaz, P. Kumar, W. K. Liao, and A. Choudhary, "Enabling active storage on parallel I/O software stacks," in 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, MSST2010, 2010.
[7] S. J. Kim, S. W. Son, W. K. Liao, M. Kandemir, R. Thakur, and A. Choudhary, "IOPin: Runtime profiling of parallel I/O in HPC systems," Proc. - 2012 SC Companion High Perform. Comput. Netw. Storage Anal. SCC 2012, pp. 18–23, 2012.
[8] O. Delivers and C. Performance, "Benefits of Offloading I / O Processing to the Adapter," pp. 10–13.
[9] "Iometer User ' s Guide," 2004.
[10] http://vmpete.com/2013/08/19/iometer-as-good-as-you-want-to-make-it/
[11] www.iometer.org/