Volume 5, Issue 7, July 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Dynamic Storage Provisioning Based on Application Workload Requirements

Swathi S
Department of Information Science,
BMS College of Engineering, Karnataka, India

Abstract— Businesses today, produce enormous amount of data which is to be stored, managed and protected. Data has become a very important asset to organizations and there is a need to adopt novel and modern approaches to efficiently store and access data. One such approach is using clustered storage solutions that provide high availability and reliability to business critical data. Provisioning a storage cluster is sheerly based on the experience of storage administrators. Lack of experience of storage administrators often leads to an underutilized or an overprovisioned cluster setup that leads to poor performance. This paper proposes a solution that helps in provisioning storage clusters optimally by taking into consideration the different hardware parameters and the application requirements.

Keywords— Clustered Storage, Storage Provisioning, Cluster data objects, application workload, optimizer engine

I. INTRODUCTION

Data is an integral and a critical part of every organization today. Efficient ways to store and retrieve data is an important requirement and a major concern. One of the common infrastructures used by enterprises today are the clustered storage solutions that tie up multiple storage servers to provide high availability, reliability, better performance, fault tolerance and large capacity. With multiple servers however, comes the difficulty of configuring each of the servers individually and also as a group.

A storage server is a stand-alone entity that is specialized for data access. A storage server comes with specialized interfaces and utility functions to manage storage peripherals. Configuration of a storage server involves configuring the associated disks, volumes, LUNs, RAID groups, network interfaces, etc. Storage servers are presently configured manually by storage administrators. Manual configuration is guided by sheer experience and some thumb rules used in the industry. The process of manual configuration of multiple storage servers is tedious, error prone and might also lead to inconsistent configurations across clusters. Also, manual configuration leads to underutilized or over provisioned clusters which impact performance.

In this paper, we present a dynamic approach to provisioning the storage clusters. This approach takes into account the underlying hardware infrastructure as well as the application workload requirements that the intended cluster will be used for.

The approach taken in this paper has two major benefits-One is , it caters to the needs of the application and provides efficient ways to store application data and second is, it provides value for money, in that optimal placement of cluster objects gives a better utilization of the cluster hardware resources.

The paper is organized into following sections: Section II discusses related work done in the field of dynamic cluster configuration, Section III discusses the architecture of the proposed system, section IV discusses the methodology followed and finally we conclude in section V.

II. RELATED WORK

Storage Cluster provisioning are generally done manually by storage administrators who follow their instinct and known best practices in the industry. System is just scaled out to handle increased load in such cases. However, this becomes inefficient as application load grows. Madhyastha.*et.al* describes a storage configuration compiler in [1] that automates cluster configuration decisions and deploys a cost efficient cluster. The decision of what cluster configuration is to be used is based on application behaviour and hardware specifications that are specified formally.

MINERVA, a suite of tools for designing storage systems automatically was discussed in [2]. MINERVA also considers descriptive specification of application behaviour and hardware details and formulates them as a constraint based optimization problem. In order to solve the optimization problem industry benchmark data has been utilized and the actual system performed exactly as what was predicted by MINERVA.

Agarwala.*et.al* discusses the challenges associated with provisioning storage resources in cluster environment in [3].List of best and novel optimization strategies were also proposed to get more resilient clusters. They come up with a tool called Casper the automates the entire process of cluster configuration.

Impact of different configuration parameters on system throughput was studied and analysed in [4].The system considered for the analysis was the ALICE (A Large Ion Collider Experiment), one of the four experiments at the CERN

LHC (Large Hadron Collider) dedicated to the study of nucleus-nucleus interactions. Its data acquisition system has to record the steady stream of very large events resulting from central collisions, with an ability to select and record rare cross section processes. The main aim of the study was to decide what storage configuration produced better performance.

III. ARCHITECTURE

A Storage cluster contains multiple storage servers tied up using a physical backplane to function as a single entity. Individual storage servers are generally called as nodes. Each node has its own set of processors, RAM, disk shelves, associates ports and interfaces. Also, the cluster as a whole has resources such as network interfaces connecting the nodes, associated routers and switches, etc. Storage cluster provisioning involves planning, designing and deciding on how these hardware resources needs to be allocated and managed at both node level and cluster level, in order for effective usage by an application that is intended to be run on the cluster. Storage administrators manage these resources by manually modifying each of the cluster object's properties. In addition to the physical cluster objects, we also have logical objects on each cluster such as RAID groups, volumes, LUNs, interface groups that further help in managing user data more efficiently. Cluster provisioning also involves planning and designing these logical cluster objects.

In this paper, we are taking a dynamic approach to provision clustered storage resources which involves building a decision making agent. The decision making agent automates the process of cluster provisioning by deciding on how and what resource of the cluster must be provisioned on which node. The decision making agent takes both the hardware specification and application workload requirements for resource allocation. Figure 1 shows the high level architecture of the system.

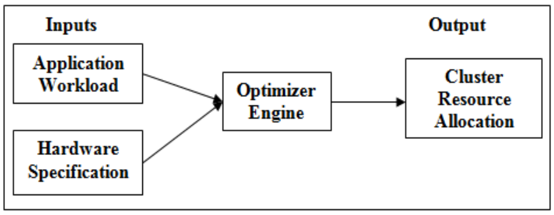

Fig 1 High Level Architecture of the Cluster Storage Provisioning System

As can be seen from Figure 1, the optimizer engine is the heart of the entire system. The optimizer engine takes the application workload and underlying hardware specifications as input and produces an optimal resource allocation for the cluster as output. The optimizer engine employs an algorithm to calculate the placement of various cluster objects, governed by various parameters and constraints. The next section briefs about the algorithm that is required to optimize the configuration.

IV. METHODOLOGY

The optimizer engine must employ an algorithm that produces cluster provisioning required for various cluster objects. For a type of application workload, the algorithm needs to evaluate various parameters such as CPU utilization, network throughput, disk capacity, etc. in order to decide on where and how each of the cluster objects needs to be placed and configured. The algorithm must also account for the hardware resources at hand.

The cluster provisioning system discussed in this paper takes application and underlying hardware specifications as input as shown in Figure 1.When we say application workload, common examples include database applications, mail servers, file servers, OLTP workload and/or combination of multiple such application. The user of the cluster must be allowed to specify multiple workloads in a standard format.

The inputs to the system must be formalised in a standard way such as XML, with the parameters required for processing clearly defined. For instance, application workload requirements specified by a user may include parameters such as disk IOPS, throughput, latency, application read/write pattern, cost/GB, etc. Hardware specifications for a storage server generally speak about number of disks, types of disks, disk capacity and its average IOPS, CPU, RAM, ports & other network infrastructure. Also information about cluster resources such as switches, routers, cluster physical backplane, etc. must be formalised.

A basic outline of the algorithm is shown in figure 2.

1. Consider one evaluation parameter (such as CPU Utilization, Network Throughput, Disk capacity, etc.) at a time to evaluate the impact of application workload on the cluster performance
2. Generate test data for the application workload to evaluate the selected parameter on a particular hardware type
3. Apply the test data on the hardware specified and measure the selected evaluation parameter.
4. Analyse the test results and come up with an equation that best fits the application and the corresponding hardware type
5. Incrementally add new evaluation parameters to make the algorithm more efficient

Fig 2 Cluster Provisioning Algorithm Outline

As stated in the algorithm, for a given hardware and an application type, two main factors that determine the cluster resource allocation are the evaluation parameter considered and the best fit equation derived. For investigation purposes we selected CPU utilization as the evaluation parameter and multiple linear regressions to derive the best fit equation. The test data we used was that of NFS protocol workload with both sequential and random read/write. We assumed that the CPU utilization is just a linear sum of various input parameter such as throughput, latency, average protocol IOPS, write penalty, etc.

The best fit equation is then applied on each node of the cluster and optimal location of various cluster objects is produced as output. For instance, an output from the algorithm gives out details on what properties of the cluster resources needs to be set to what values and how logical objects such as volumes, LUNs, interfaces and interface groups needs to be placed on various nodes of the cluster.

V. CONCLUSION

The process of manual provisioning of storage clusters is not only tedious, but also ineffective. In this paper, we presented an idea to dynamically provision a clustered storage device that best suits the application type and the user ecosystem. Application workloads are typically characterized by a specific read/write pattern and following the approach taken in this paper, we will able to cater best to the application requirements. This also proves to be cost effective as every piece of hardware is optimally used. Further work in this area can be done to include just not the initial application workload pattern but also predict futuristic requirements and have a room for such requests in the allocation.

REFERENCES

[1] Harsha V. Madhyastha , John C. McCullough , George Porter , Rishi Kapoor , Stefan Savage , Alex C. Snoeren , Amin Vahdat “scc: cluster storage provisioning informed by application characteristics and SLAs” Proceedings of the 10th USENIX conference on File and Storage Technologies, p.23-23, February 14-17, 2012, San Jose, CA.

[2] Guillermo A. Alvarez, Elizabeth Borowsky, Susie Go, Theodore H. Romer, Ralph Becker-Szendy, Richard Golding,Arif Merchant,Mirjana Spasojevic,Alistair Veitcand John Wilkes “Minerva: An automated resource provisioning tool for large-scale storage systems” ACM Transactions on Computer Systems, 2001.

[3] Sandip Agarwala, Ramani Routray “Cluster Aware Storage Resource Provisioning in a data centre” NOMS 2010: 12th IEEEIFIP Network Operations and Management Symposium

[4] A. Telesca, P. Vande Vyvre, R. Divi`a, V. Altini, F. Carena, W. Carena, S. Chapeland, V. Chibante Barroso, F. Costa, M. Frauman, I. Makhlyueva, O. Rademakers-Di Rosa, D. Rodriguez Navarro, F. Roukoutakis, K. Schossmaier, C. So`os, B. von Haller “The ALICE DAQ Storage System: an analysis of the impact on the performance of the configuration parameters and of the load of concurrent streams” 2009 16th IEEE-NPSS Real Time Conference.