Asia Pacific Journal of Research
A peer reviewed international Journal
Print -ISSN -2320 - 5504
Online-E-ISSN - 2347 - 4793

ENERGY EFFICIENCY USING DATA REPLICATION AND POWER CONSUMPTION BASED ON SCHEDULING.

K.Sangeetha [#1] ***M.Meenakshi*** [*2]

[#] *Assistant Professor,Dept. of Computer Science and Engineering*
Apollo Engineering college, Chennai, Tamil Nadu
[*] *PG Scholar , Department of Computer Science and Engineering*
Apollo Engineering college, Chennai, Tamil Nadu

ABSTRACT

Cloud Computing is an emerging concept combining many fields of computing. It provides services, software and processing capacity over the internet. The energy efficiency of the data centers storing this data is one of the biggest issues in data intensive computing. In order to address such issues, we are designing and analyzing a series of energy efficient data aware strategies involving data replication and CPU scheduling. In this paper, we present a new strategy for data replication, called Pseudo Least-Recently-Used (PLRU), and study its performance to determine if it is an energy efficient strategy. We measure the performance of PLRU and existing replica strategies on a small green cluster to study the running time and power consumption of the strategies with and without data backfilling. Results from this study have demonstrated that energy efficient data management can reduce energy consumption without negatively impacting response time.

Index Terms- *Data backfilling, Data cluster, Data grid, Data intensive computing, Data replication, Energy Power consumption, QLFU, PLRU.*

ISSN: 2320-5504, E-ISSN-2347-4793

1. INTRODUCTION

This project provides efficient power utilization of large data storage environment and also can provide the ability to solve large-scale applications which require the processing of large amounts of data, they have been recognized as extremely energy inefficient. Computing elements can be located far away from the data storage elements. A common solution to improve availability and file access time in such environments is to replicate the data, resulting in the creation of copies of data files at many different sites. The energy efficiency of the data centers storing this data is one of the biggest issues in data intensive computing. Since power is needed to transmit, store and cool the data, we propose to minimize the amount of data transmitted and stored by utilizing smart replication strategies that are data aware. Computing elements can be located far away from the data storage elements. In this paper, we present a new strategy for data replication, called Pseudo Least-Recently-Used (PLRU), and study its performance to determine if it is an energy efficient strategy. We measure the performance of PLRU and existing replica strategies on a small green cluster to study the running time and power consumption of the strategies with and without data backfilling. Results from this study have demonstrated that energy efficient data management can reduce energy consumption without negatively impacting response time.

2. RELATED WORKS

The need to consider the energy efficiency of computing systems is becoming increasingly popular. With a number of white papers, a group of information technology professionals called the Green Grid, have been proposing energy efficient strategies for large scale data centers. Their goal was to design successful data metrics that would analyze and scale the efficiency and productivity of the data transfers. In their work they suggested two major metrics:

Power usage of efficiency (PuE)

$$= \frac{\text{Total data center facility power}}{\text{IT equipment power}}$$

Data center infrastructure efficiency (DCiE) $= {}^{1}/_{PuE}$

They also consider the Telecommunications Equipment Energy Efficiency Rating (TEEER) developed by Verizon. In the authors propose a blueprint for an Energy Aware Grid that incorporates energy efficiency and thermal management for data centers. Their results demonstrate the economic feasibility and energy savings of their approach. As another example of the increasing awareness of the need for energy efficiency, IEEE Computer magazine began with an issue dedicated to green issues and dedicated a column to Green IT. Insight into the CPU utilization of servers is provided in, indicating servers are almost never totally idle or operating at their maximum utilization. Peak energy efficiency (processed requests per Watt) of servers occurs at peak utilization. However, most servers spend most of their time at the 20%–30% utilization range, at which point the energy efficiency is less than half of the energy efficiency that can be found at peak performance. Server processors can provide a wide dynamic power range, while the range of other components is much smaller, e.g. 25% for disk drives and 15% for net working switches. Disk devices have a latency and energy penalty for transition from an inactive to an active mode and networking equipment typically does not have low-power modes. As a result, alternative strategies are needed to reduce the data transmitted and the amount of disk storage utilized in a data grid. Benchmarks for power supplied to a single unit server, such as SPEC, have been applied by some organizations that analyze performance.

In their paper, the authors propose a power consumption model that was based on a benchmark set by a prior published report by the Transaction Processing Performance Council (TPC-C).They apply various power usage trends to the model to analyze the consumption trends and to identify the components that are power-intensive. They report advancements in software and hardware systems that could make them energy efficient. This acts as a standard to compensate between the performance and price effectiveness of computing systems. Other work by a group of researchers surveys power and energy management techniques that are applied to modern day server systems. The authors link these results to the workloads related to each of the servers and identify techniques that could provide significant savings in energy together with minimal compensation in the systems performance. Their survey concentrates on energy saving at the individual component level, as well as cluster-wide power savings in a data grid. They compare current and past work done, and suggest a future course of action in the area of power savings in a data grid. Data replication is one way to reduce the amount of data transmitted, but smart replication is needed to reduce storage. Early work on data replication in data grids has placed the most attention on decreasing the data access latency and the network bandwidth assumption. In [21], six replica strategies are presented and their results indicate that the best strategy has significant savings in latency and bandwidth consumption if the access patterns contain a moderate amount of geographical locality. The HotZone algorithm is presented to place replicas in a wide area network, so that the client-to-replica latency is minimized. A dynamic replica replication strategy HBR is proposed that benefits from ‘‘network-level locality’’, utilizing the broadest bandwidth to the site of the job execution. Later, researchers began to focus on the availability of the data. The authors propose providing replicated fragments of files to increase data access performance and load balancing. A dynamic model-driven replication approach in which peers create replicas automatically in a decentralized fashion is presented. The authors propose to meet the data availability goal based on the assumption that the total system replica storage is large enough to hold all the data replica copies. The authors propose a data replication strategy for star-topology data grids that replicates files based on their popularity as calculated periodically. We note that none of this work on data replication considers power consumption. There has been much research in the area of Dynamic Voltage Scaling (DVS) , which builds on the tradeoff between performance and energy savings. It has been recognized that low-performance, low-power processing is adequate in many situations. Low performance results from lowering the operating frequency of the processor. By lowering the frequency, the processor can operate at a lower voltage, and hence, requires less energy. This work demonstrates the potential for energy savings if the performance scales better than linearly as CPU frequency is decreased. Recent work considers powering down disk storage during non-peak hours. In a subset of the disks is powered down, but all data items are still available in the system. The goal is to have little impact on the performance of the system during normal operation. Virtual nodes and migration are utilized in, so that a small number of disks can be used without overloading them. The authors also introduce an extension of their strategy in which some of the data is replicated. We note that while we also consider replication in our work, we do not power down the disk storage. The work in the Petashare project proposes several approaches to data aware scheduling for CPU-oriented batch schedulers and proposes a new data aware paradigm. Experiments indicate increasing parallelism and concurrency do not always result in an increase in the data transfer rate. In addition, their results indicate that data placement by a scheduler can improve the performance and reliability of a system. The authors also consider such factors as remote I/O versus staging. The authors do not consider the energy efficiency of their approaches.

3. Architecture and Algorithm

When the data are distributed among different sites in a data intensive environment, there is a high probability the system will access data which is not in the local site. Remote data file access can be a very expensive operation, considering the sizes of the files and limited networking bandwidth or network congestion. To reduce the amount of time needed to access the data, data replication can be utilized to avoid remote file access. However, in order to decrease the amount of energy needed to store the data, it is imperative that a limit be placed on the size of the storage. We propose to minimize the energy consumed through the use of smart data replication to reduce the cost of accessing and storing the data.

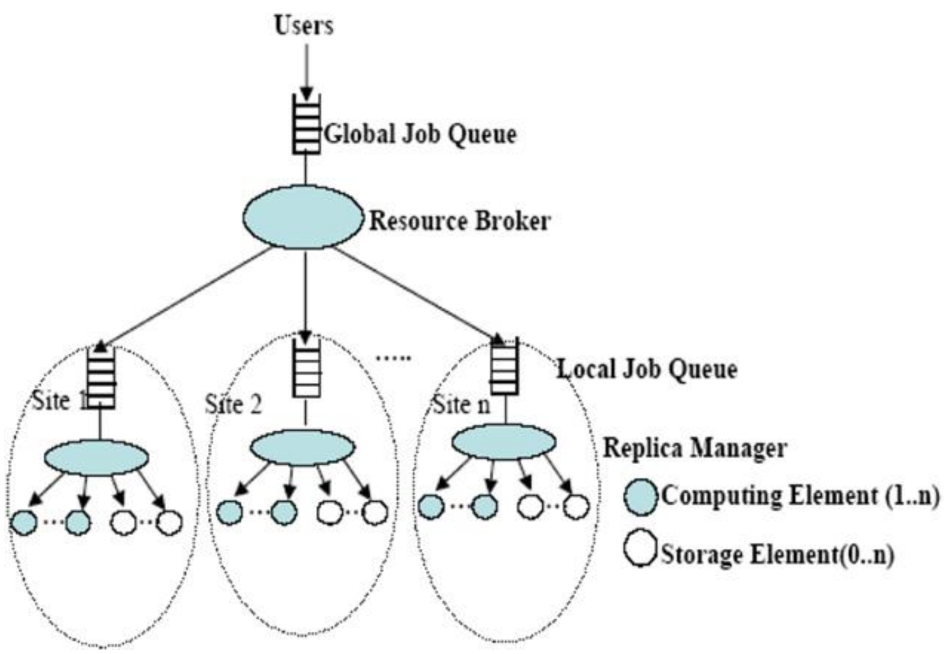

Fig. 1. Data grid architecture.

We consider single-tier grids in our work. We assume that in an organization a mixture of applications exists, whereby, some of the grids will be single-tier in structure [30], while other applications will use a more complicated structure, such as a multi-tier grid. We expect the strategies developed for single-tier grids here can be used within the multi-tier structure and will investigate as such in the future. Fig. 1 illustrates single-tier grid architecture. A data grid typically consists of job execution sites containing the components: the Computing Element CE, the Storage Element SE and a Replica Manger containing a replica optimizer. Given a set of jobs, J = (J1, J2, J3, . . . , JN), they are submitted to the Resource Broker agent through a user interface. The resource broker agent then schedules them to the appropriate computing sites, where the jobs may be queued in the site's job queue. Meanwhile, each job will request more than one file to complete its own task from the file set, F = (f1, f2, . . . , fk), where each file fi is of size Si. It is common for a job in a data grid to list all the files needed to complete its task. Hence, we utilize this aspect in designing a data replication scheme.

3.1 QLFU ALGORITHM

This is an existing algorithm which has new replication and replacement scheme QLFU, which is called the Queued LFU (least-frequently-used) replica schema. QLFU considers the number of times a file is accessed in the future and how far the replica optimizer will look ahead into the job queue. Traditional LFU considers the references to the file in the past and replaces the file that has been used the least frequently in the past. LFU assumes a file not frequently used in the past is not likely to be used in the future. This differs from QLFU which considers only future requests when making decisions for the immediate present. QLFU replaces files that are the least likely to be used in the future. This means every time a pending request for a file is queued, a count associated with that file is incremented, which represents the number of times the file will be requested in the future. The file with the smallest count that is resident in the storage element is the one to be replaced.

4 PROPOSED SCHEME

4.1 PLRU Algorithm

Pseudo Least Recently used algorithm is a cache algorithm used to improve the Least Recently Used Algorithm. PLRU usually refers to two cache algorithm. They are tree-PLRU and bit-PLRU. Comparing PLRU with the existing algorithm QLFU this PLRU also can able to decrease the power consumption of the system and also can able to increase the average run time of the system. The goal of the PLRU algorithm is to replace the page that has not been referenced for the longest time.

4.1.1 Example for the PLRU Algorithm

In this proposed system the VM ware Player is used to mount Linux Operating System in order to access the Open stack private cloud.. The computing resources are provisioned by using the resource provisioning model and the provision resources are network, storage, CPU processing power**.**

Time		0	1	2	3	4	5	6	7	8	9	10
Requests			*c*	*a*	*d*	*b*	*e*	*b*	*a*	*b*	*c*	*d*
Page Frames	0	*a*										
	1	*b*										
	2	*c*										
	3	*d*										
Faults												
Time page last used												

Fig 4.1 Total number of files in the system

Time		0	1	2	3	4	5	6	7	8	9	10
Requests			*c*	*a*	*d*	*b*	*e*	*b*	*a*	*b*	*c*	*d*
Page Frames	0	*a*	*a*	*a*	*a*	*a*	*a*	*a*	*a*	*a*	*a*	*a*
	1	*b*	*b*	*b*	*b*	*b*	*b*	*b*	*b*	*b*	*b*	*b*
	2	*c*	*c*	*c*	*c*	*c*	***e***	*e*	*e*	*e*	*e*	***d***
	3	*d*	*d*	*d*	*d*	*d*	*d*	*d*	*d*	*d*	***c***	*c*
Faults							•				•	•
Time page last used						$a=2$ $b=4$ $c=1$ $d=3$				$a=7$ $b=8$ $e=5$ $d=3$	$a=7$ $b=8$ $e=5$ $c=9$	

Fig 4.2 Checking the number of unused files

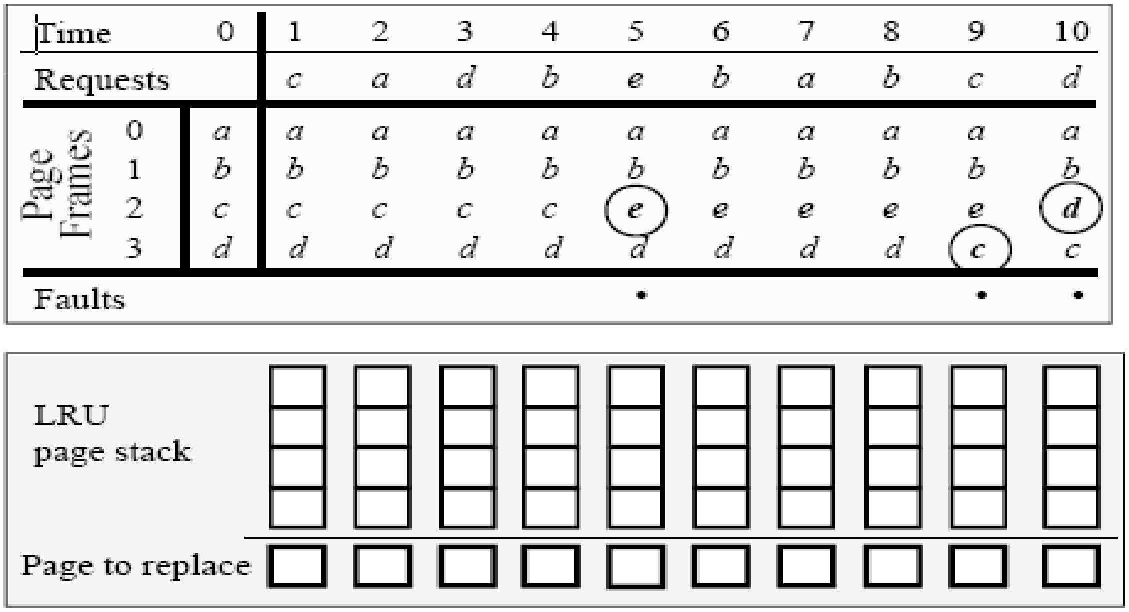

Fig 4.3 Maintaining the stack of number of used pages.

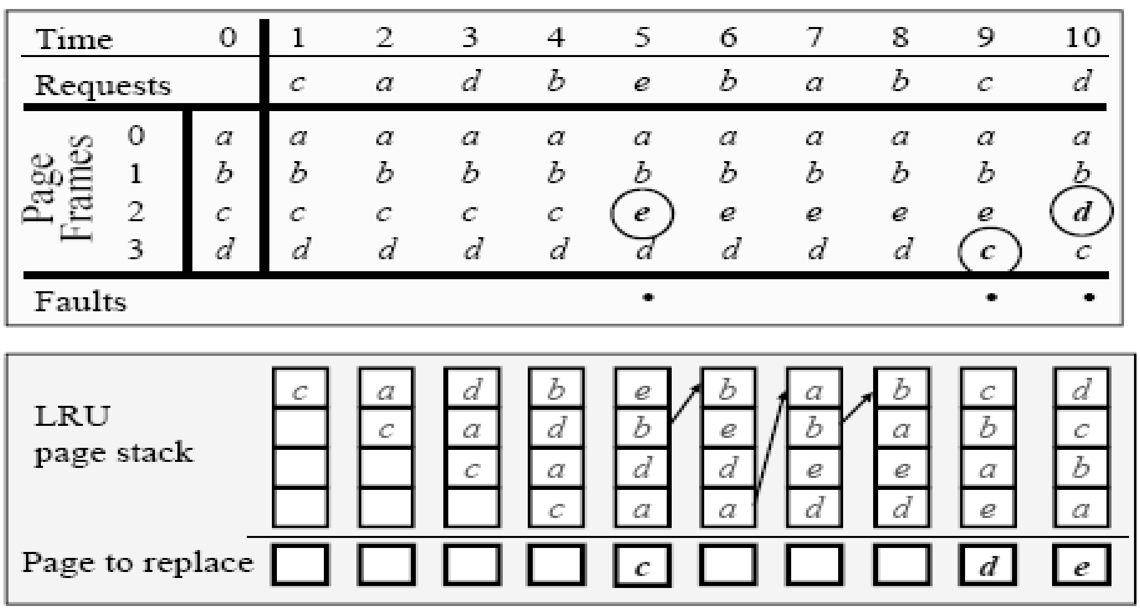

Fig 4.4 Replacing the unused files.

ISSN: 2320-5504, E-ISSN-2347-4793

5. RESULTS AND DISCUSSIONS

The performance of the proposed PLRU algorithm is implemented using NS2 at LINUX operating system. The average run time and the average energy consumed is calculated and verifying using PLRU algorithm and finally the result has been shown. In Fig 5. It displays the transferring of files in a cluster form using PLRU algorithm.

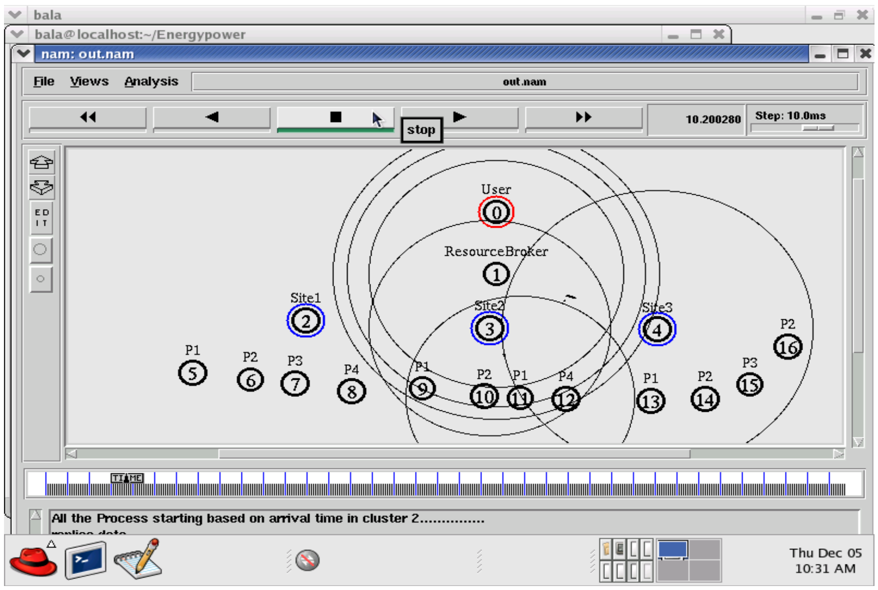

Fig.5 File Transfer in a cluster form.

6. CONCLUSIONS

The insufficiency of the traditional distributed computing systems and existing CPU-oriented batch schedulers in dealing with the complex data handling problem has yielded a new emerging era: the data-aware schedulers. One of the first examples of such schedulers is the data placement scheduler. As a result, we present a replica strategy, called PLRU that utilizes pending file requests to make current replication decisions. PLRU is designed to minimize the amount of data transmitted and storage needed. The PLRU strategy was implemented on our small Sage cluster, which was built considering both energy sufficiency and cost. Performance results on our cluster indicate our PLRU strategy performs better than existing strategies, such as FIFO, LFU LRU, MRU, SWIN and QLFU in terms of both average running time and energy consumed per job. PLRU is beneficial in terms of energy consumed per job regardless of the number of nodes or the sizes of the files. We also plan to use data backfilling and to schedule the jobs so that we can utilize forced periods of suspension time for power saving. We are studying powering down nodes depending on the load of the cluster. Data gathered with these tests would reveal the worth, or cost, of powering down nodes dynamically to decrease power consumption at the expense of a slower response to sudden dramatic increases in job requests. We anticipate that the strategies presented in this paper could also be useful for saving energy in service-oriented grids.

7. REFERENCES

[1] C. Patel, R. Sharma, C. Bash, S. Grauper, Energy aware grid: global workload placement based on energy efficiency, Hewlett-Packard (2002).

[2] R. Fraser, T. Rankine, R. Woodcock, Service oriented grid architecture for geosciences community, in: Proceedings of the 5th Australasian Symposium on Grid Computing and e-Research, AusGrid, 2007.

[3] D. Abadi, Data management in the cloud: limitations and opportunities, in: Bulletin of the IEEE Technical Committee on Data Engineering, 2009.

[4] T. Kosar, M. Balman, A new paradigm: data-aware scheduling in grid computing, Future Generation Computer Systems 25 (2009) 406–413.

[5] Green grid, Green grid metrics data center power efficiency metrics: PUE and DCiE, Technical Committee White Paper, February 2007.

[6] US Environmental Protection Agency, Report to Congress on Server and data center energy efficiency Public law 109–431, in: Energy Star Program, 2007.

[7] P. Korcan, P.-J. Bochard, Storage efficiency with IBM N series, Information Infrastructure Forum (2009).

[8] J. Baliga, R. Ayre, K. Hinton, R. Tucker, Green cloud computing: balancing energy in processing, storage, and transport, Proceedings of the IEEE 99 (1) (2011) 149–167.

[9] J. Koomey, Growth in Data Center Electricity Use 2005 to 2010, Analytics Press, Oakland, CA, 2011.

[10] K. Cameron, Green introspection, IEEE Computer 42 (1) (2009) 101–103.

[11]http://gcn.com/articles/2010/10/04/datacentermanagement.aspx

[12]DellComputersThegreendataproject:http://www.greendataproject.org/index.php?option=com_content&task=view&id=44&Itemid=60/.

[13] D.J. Brown, C. Reams, Toward energy-efficient computing, Communications of the ACM 53 (3) (2010) 50–58.

[14] D.G. Cameron, R. Carvajal-Schiaffino, A.P. Millar, C. Nicholson, K. Stockinger, F. Zini, Evaluating scheduling and replica optimisation strategies in OptorSim, in: 4th International Workshop on Grid Computing, Grid2003, IEEE Computer Society Press, Phoenix, Arizona, 2003, November 17.

[15] M. Lei, S. Vrbsky, R. Horton, A remote data access element for data grids, grid computing and applications, in: WorldComp'09 2009 International Conference on Grid Computing and Applications, GCA'09, Las Vegas, NV, July 2009, pp. 188–194.