Performance Impact of Disk I/O Configuration on High-Throughput Servers

*Amisu Salam-Alada and Abdul Waheed
E-mail: {alada, awaheed} @ccse.kfupm.edu.sa
Phone: +966 3 860 1423
Computer Engineering Department
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia

Abstract

The gap between processor and I/O performance continues to grow. Performance of several high-throughput network application servers is limited by disk I/O subsystem overhead despite continued exponential increases in processor performance. This paper considers three I/O subsystem configurations for high throughput servers: single disk, disk array, and Redundant Array of Inexpensive Disks (RAID). It looked at the impact of these disk I/O configurations on the performance of high throughput servers based on I/O access patterns. The I/O accesses pattern was based on low-level device access logs for high throughput servers.

Compared to all other components in a system, a disk is the only device that uses mechanical components. Consequently, the performance of a disk (in terms of access latency) lags behind memory performance by several orders of magnitude. Therefore, this paper focuses on the performance impact of the different disk I/O subsystem on the high throughput servers in a network infrastructure.

Keywords: high-throughput servers, disk I/O access patterns, and disk I/O configurations.

1. INTRODUCTION

In the majority of institutions with high volume web sites, Input/Output (I/O) operations have been recognized as critical system performance inhibitors. Performance of I/O is particularly important for those who want to serve large volumes of user requests simultaneously. This is due to the fact that whatever the users want to obtain are residing on disk storage. As a web site grows in popularity and the traffic increases, the performance requirements for such disk I/O on high throughput can be a challenge to meet. Hence the I/O handling program in such a high throughput sever should be able to accommodate thousands of requests per minute in order to give better throughput.

In the last decades, innovations in technology have resulted in extraordinary advances in computer processing speed. These advances have led many computer system performance analysts to focus their attention on measuring processor performance to the near exclusion of all other metrics; some have even equated a computer system’s performance with how well its CPU performs. This viewpoint, which makes system-wide performance synonymous with CPU speed, is becoming increasingly impractical due to growing disparity with I/O speeds.

If CPU performance continues to improve at its current fast pace and disk I/O at a moderate pace, eventually the performance of all applications such as the web server that perform excessive I/O operations will be limited by these I/O operations. Thus further CPU performance improvements will have no effect on performance [7]. An ever-widening mismatch between disk I/O and processor performance is rendering disk I/O performance evaluation increasingly important.

Several studies have analyzed disk I/O subsystem using simulation modeling and I/O workload characterization [2][4]. Some have modeled each of the disk system components. In this work, we are looking into the performance requirements of disk I/O behavior when it is used in a high throughput network infrastructure device such as web server or web proxy server. Such servers serve thousands of users simultaneously and the users expect a timely response. Therefore, in order for such high throughput servers to perform, each component of the server, i.e., the hardware components like CPU, memory, disk I/O and network I/O and software components like the HTTP demon, the TCP/IP implementation and the file system must also perform well. We have noticed that one area where bottlenecks occur frequently is the disk I/O component.
Considering the high utilization of the web these days, server performance requirements in terms of high throughput continue to increase. Some of the high throughput servers that are often used on the Internet include: web servers, proxy servers, streaming media servers, ftp servers, etc.

In this paper, we use measurement-based evaluation to look at the performance impact of different disk configurations on high throughput server. We compare three I/O subsystem configurations of our interest: single disk, disk array, and Redundant Array of Inexpensive Disks (RAID) and evaluate a particular class of access patterns that was obtained from the high throughput server considered.

The paper is organized as follows: In section 2 we describe the background for our work. In section 3, we describe our experimental design and results obtained. Finally, our conclusions are presented in section 4.

2. BACKGROUND

Two types of high-throughput servers are considered, these are: proxy servers, and web servers.

Proxy Servers

A web proxy server is a software system that runs on a dedicated server platform and typically acts as a proxy for a client and forwards web traffic from servers to client and vice versa. The primary purpose is to save network bandwidth and to reduce user perceived network latency by filtering and caching web traffic [1]. Using the caching proxy may result in reduction of external network traffic and reducing the response time for request [8]. The explosive growth of web traffic in recent years, the increasing user

Top

demand for low latency service and high cost of bandwidth make the use of proxy servers a very attractive option for saving resources.

The proxy behaves similar to a web server; it listens for and responds to clients' requests. The methods by which a proxy server meets the client request depends on the type of proxy server that is being use. If it does not support caching the user request will be forwarded to the remote server that is specified by the URL in the client request. Once it receives the request it passes to the client. But if it uses cache, the cache will be searched for the requested document before passing request to the remote server.

High throughput servers often make use of disks for caching. A few research efforts report work on reducing the overhead of disk I/O in this area. Carlos Maltzahn and K. J. Richardson [5] noted that the most common bottleneck for large caching proxy server is disk I/O. In their work, they evaluated several ways to reduce the amount of required disk I/O for a proxy server. This was done by comparing the file system interaction of two existing web proxy servers and showing how design adjustment to the cache architecture could reduce disk I/O and improve performance. They discovered that preserving locality of HTTP reference stream while translating them into cache references and use of virtual memory instead of file system for objects smaller than the system page size are effective strategies that also reduce disk I/O overhead on the server.

With continued growth in demand for large caches and higher performance for web proxy servers, the common bottleneck facing proxy servers is disk I/O. Since not all information being requested can stay in the memory, therefore a web proxy server will have to make use of disk I/O, thereby causing I/O overheads and reducing its performance.

3. Experimental Design and Results

Before presenting any measurement-based evaluation, we describe our experimental setup in this section. Experimental setup consists of hardware platform and software tools. We present testbed and relevant software tools in the following subsection.

Evaluation of Widely Distributed Strides Access Pattern (WDSA) Workload

Disk I/O accesses from a typical web proxy server can result in widely distributed strides access pattern. We use Microsoft ISA [9] Proxy server on Windows 2000 Server platform and load it with Web Polygraph [6] benchmark to generate WDSA workload. Web Polygraph uses a standard workload, called Polymix-3 for proxy server evaluation at a target load rate expressed in terms of HTTP transactions per second.

In order to conduct these experiments, we vary two main factors: target HTTP transaction rate and disk configuration. Standard workload from Web Polygraph benchmark (Polymix-3) [6] was used to test ISA proxy servers under identical conditions. For this proxy server, we gradually increase the target transaction rate from 120 to 800 transactions/sec with an increment of 400 in each test to determine the peak transaction rate.

Top

The Microsoft ISA proxy was tested in stand-alone mode with single server. Finally, we evaluated the performance impact of downloading large files onto the disk subsystem through ISA proxy server.

To carry out our experiment, we considered two basic factors, in order to evaluate their impact on the high throughput server. These factors are: (1) three different disk configurations, that is, single, multiple and RAID system; and (2) the target load that server can handle for each test. The Hit rate of the proxy server is fixed by Polymix-3 to more than 50% and the proxy servers are expected to meet that hit rate when load is within their normal operating range.

We use two sets of performance metrics: the first focuses on proxy server performance and the second set of metrics focuses on the system performance (of proxy server). Metrics to evaluate the proxy server performance include: HTTP request rate (or connection rate), throughput (in Mbps), hit ratio, and average response time (per transaction), while metric related to the system performance include: CPU utilization of the proxy server host, disk utilization, disk transaction rate, and disk queue length.

Benchmarking Tool

In order to benchmark proxy server, the most extensive public-domain benchmark is Web Polygraph [6] This generates robots PCs that can make connection to the server through the proxy.

Web Polygraph [6] is a public-domain benchmark and is extensively used by Proxy Caching industry to evaluate the performance of various products under realistic workload scenarios. This benchmarking tool simulates web clients, servers, and a large number of client/server IP addresses to stress a proxy server. It provides standard workloads (e.g., Polymix-3) that describe the content type, client and server behaviors, document popularity distribution, target proxy server load, target proxy server hit rate, and a number of other configurable parameters to make the test realistic. It simulates a user as a "robot" that can open multiple connections (client agents) simultaneously that access origin servers through proxy. Polymix-3 workload consists of two phases of heavy load (top1 and top2) and an initial (fill) phase to warm-up the cache.

Server Performance under WDSA

In order to evaluate the disk I/O performance under WDSA workload, we present the results of Web Polygraph tests on Microsoft Internet Server Accelerator (ISA) proxy server. The experiment used an advanced Windows2000 Server platform that runs ISA proxy server. Hardware platform is a PIII 677MHZ processor IBM Netfinity server with three SCSI hard disks (one for OS and two for caching), 1 GB RAM and 1000 Mbps NIC and connected to (simulated) clients and servers through a Gigabit Ethernet switch (Cisco3550).

We modified several parameters in the Windows2000 Server registry to enable it to become a highly available server with possibility of handle thousands of simultaneous connections.

Top

Figure 4 presents server performance in terms of transaction rate, data throughput (in Mbps), target document hit ratio, and average transaction latency with respect to offered load (in terms of transactions/sec). In this figure, server performance is presented in terms of actual transaction rate, bit throughput, achieved document hit ratio, and average transaction latency with respect to varying offered HTTP transactions load.

We notice that the throughput increases linearly from target load of 120 transactions/sec to 600 transactions/sec and thereafter starts to decrease. This is because of the server limitation, which can hardly support any transactions beyond 600 transactions/sec. We observe similar trend in all the three configurations; however, the RAID configuration provides higher transaction rate and throughput with heavy offered load (in the range of 800 transactions/sec).

We notice a decrease in the measured hit ratio delivered by the server as the transaction rate increases. With higher transaction load, the rate at which the server responds to client requests drops. Therefore, the document-hit ratio is expected to reduce under overload.

Three configurations have similar behavior of transaction latency except at large offered load. Latency increases significantly when offered load reaches 600 transactions/sec. A single disk configuration appears to cope well with high load compared to other two configurations. In disk array and RAID configurations, requests are made to different disks at the same time; hence, we expect the latency to respond to increase under heavy load.

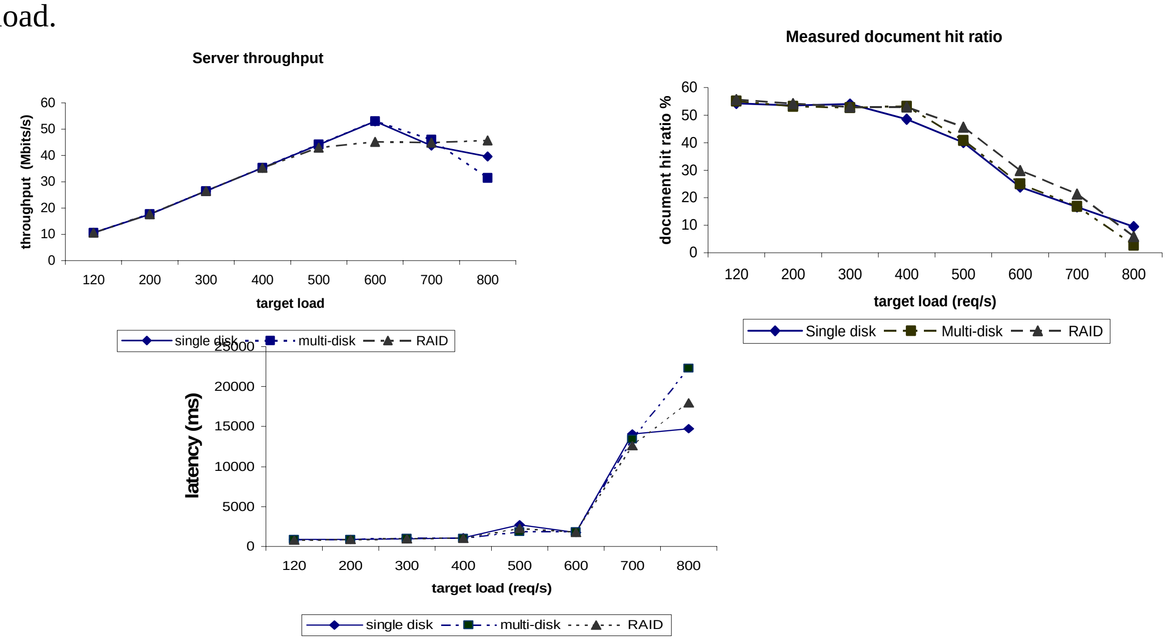


Figure 4(a-c): Server performance measurement under WDSA workload with varying target load and three configurations: single disk, disk array, and RAID5.

Top

System Performance under WDSA

Figure 5 presents the average CPU utilization during the maximum load phase (top2) of each polygraph test under three configurations: single disk, disk array, and RAID. Although three configurations are similar in terms of CPU utilization, RAID configuration shows higher CPU utilization between 400 to 600 transaction rate. This is largely because the RAID configuration adds additional fault tolerance to the system.

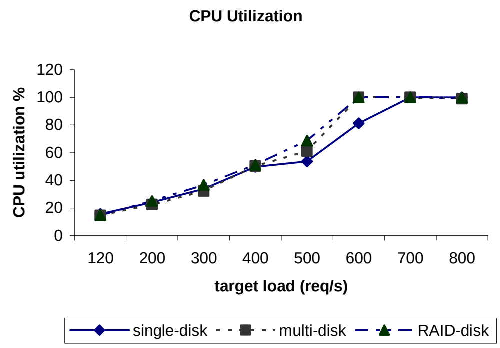
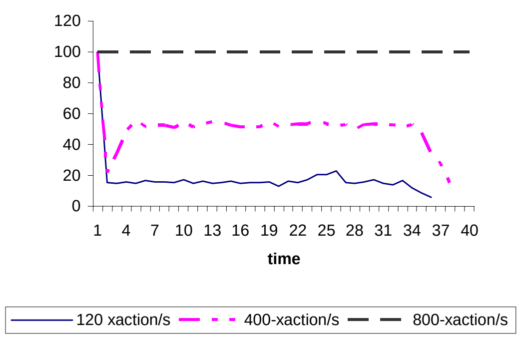
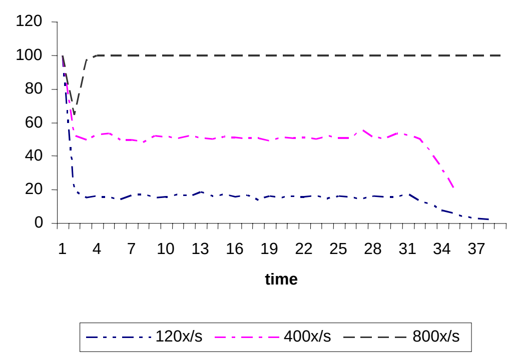


Figure 5: System performance in terms of average CPU utilization during the maximum loading phases of Web Polygraph benchmark runs that generates WDSA workload.

In addition to considering average CPU utilization, we also present the instantaneous samples of CPU utilization during top loading phase for all configurations. These measurements are plotted over observed phase in Figure 6. CPU utilization is obtained under three levels of loads: 120, 400, and 800 transactions/sec for each of the three configurations: single disk, disk array, and RAID5. Web proxy server CPU saturates whenever the target load increases to 800 transactions/sec. While the three graphs represent the three configuration tested, the graphs have similar nature, indicating that disk system has very little to do when increasing the target load.



Single disk



Disk arrav

Top

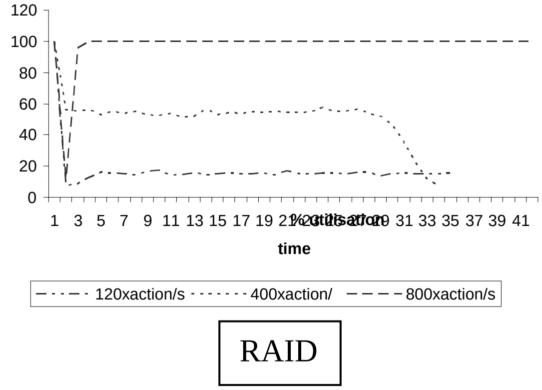


Figure 6 (a-c): System performance in terms of CPU utilization during the maximum loading phases of Web Polygraph benchmark runs that generates WDSA workload.

I/O Performance under WDSA

Figures 7 presents the disk access rate in terms of read and write operations to a single disk under WDSA workload. As expected, frequency of read transactions is significantly greater than the frequency of write transactions during the proxy server peak period referred to as top2 phase of Web Polygraph experiments. This is especially true when the proxy server operates close to its peak throughput level around 400-600 transactions/sec range and reduces when load increases beyond this limit. These are typical symptoms of a performance bottleneck due to disk I/O throughput limitations. A maximum of about 140 disk transactions/sec throughput is achieved, which is a limit for the SCSI disk used for this experiment.

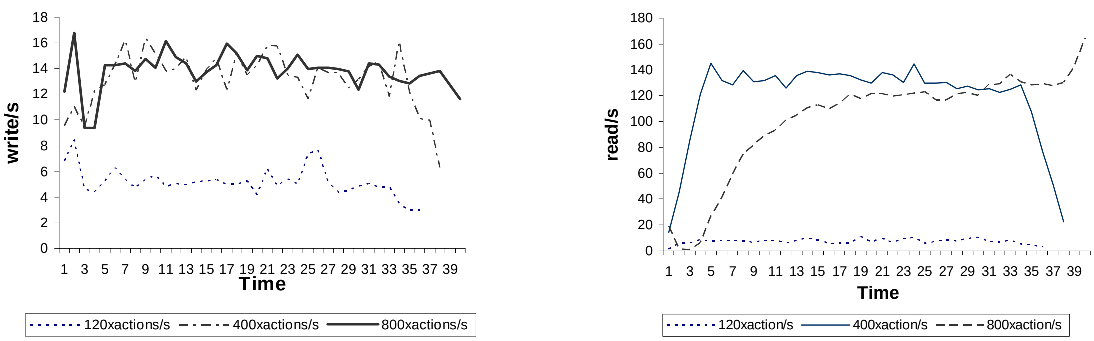


Figure 7 (a-b): Frequency of read and write transactions at three different levels of load to the proxy server: 120, 400, and 800 transactions/sec.

Disk throughput under different disk configurations is shown in Figure 8. At high target transactions load, the throughput observed in single and multiple disks system is close to 10 transfer/s, however, the RAID 5 configuration shows a relatively higher (and unbalanced among three disks) throughput in the range of 40 transactions/sec.

Top

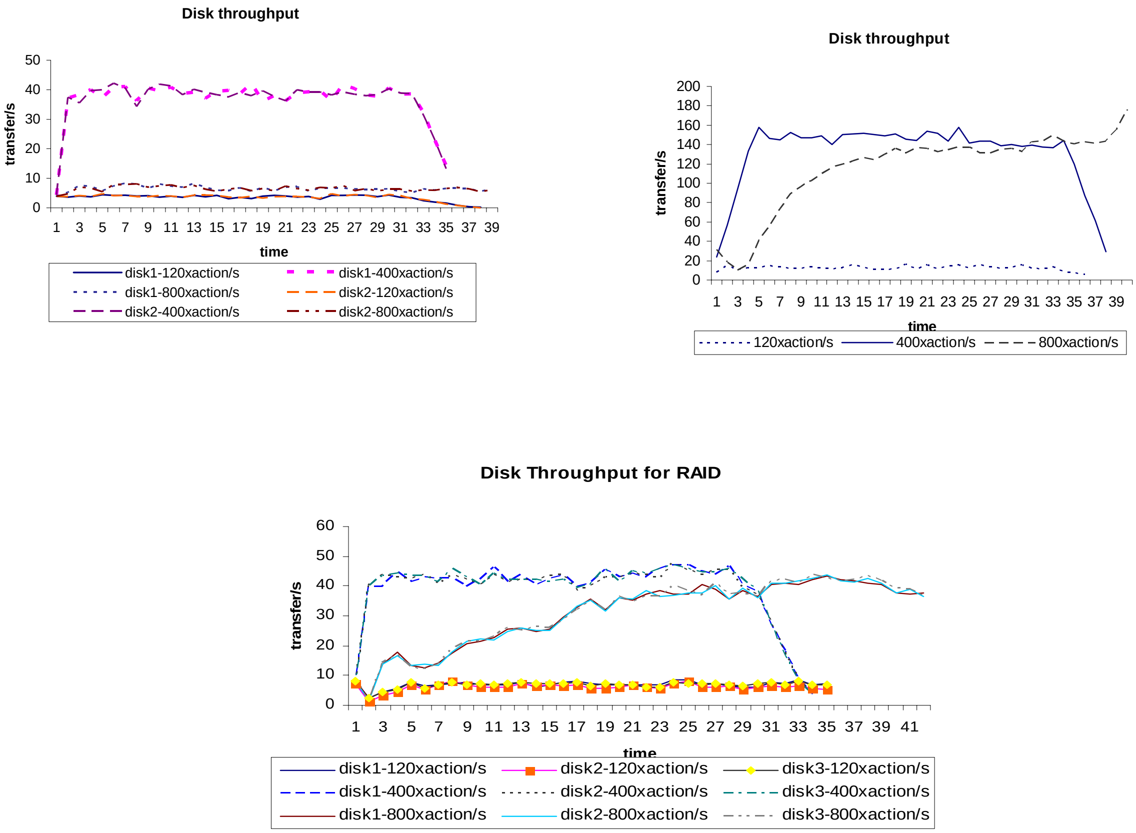


Figure 8(a-c): I/O performance in terms of disk throughput under WDSA workload for three configurations: single disk (b) disk array and (c) RAID5.

Using WDSA workload for measurement-based evaluation, we have validated the simulation results related to corresponding case for three configurations. Measurements validate that using either disk array or RAID can improve throughout and reduce latency at high server load.

4. Conclusion

In this paper, we have presented the results of our measurement-based evaluation of the impact of disk I/O subsystem on the server performance. Performance of the server system under different disk configurations was evaluated. We report that disk array and RAID 5 configurations give better performance than single disk system under heavy loads resulting in higher server throughput. With WDSA, the results indicates having a disk array or RAID configurations, the server was able to deliver higher throughput especially when the server is stressed to its maximum peak. However, CPU utilization is higher compare to when single disk is used.

Top

Reference

[1] Ari Luotonen. “Web Proxy Servers.” Prentice Hall, Upper Saddle River, NJ, 1998.

[2] B. Worthington, G. Ganger and Y. Patt. “Scheduling Algorithms for Modern Disk Drives.” Proc. ACM Sigmetrics Conf. pp. 241-251, May 1994.

[3] Cabrera, L. F. and Long, D. E. “Exploiting Multiple I/O Streams to provide High Data rates.” Technical Report UCSC-CRL-91-08, UC San Diego / CSE, April 1991.

[4] Cabrera, L. F. and Long, D. E. Swift. “Using Distributed Disk Striping to provide High I/O data rates.” Technical Report UCSC-CRL-91-46, UC San Diego / CSE, December 1991.

[5] Carlos Maltzahn, Kathy J. Richardson, and D. Grunwald. “Reducing the Disk I/O of Web Proxy Server Caches.” In Proceeding of USENIX 1999 Annual Technical Conference California USA June 1999.

[6] Web polygraph version 2.7.6 available at http://www.web-polygraph.org/

[7] G. M. Amdahl. ‘‘Validity of the single processor approach to achieving large scale computing capabilities’’, Proceedings AFIPS 1967 Spring Joint Computer Conference 30 (April 1967), 483-485.

[8] S. Chen and D. Towsley. “A performance evaluation of RAID architectures.” IEEE Transactions on Computers, 45(10): 1116–30, October 1996

[9] Microsoft Internet Security and Acceleration Server, available at http://www.microsoft.com/isaserver/