INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Network Performance of Proxy-Enabled Server Using Three Configurations

Jon Laurence B Wenceslao[1], **Rhonaida B Wenceslao**[1]*

[1] Faculty, Institute of Information and Computer Studies Northern, Iloilo Polytechnic State College Main Campus, Estancia, Iloilo, Philippines

OPEN ACCESS

Received: 20-08-2021

Accepted: 20-12-2021

Published: 19.05.2022

Citation: Wenceslao JLB, Wenceslao RB (2022) Network Performance of Proxy-Enabled Server Using Three Configurations. Indian Journal of Science and Technology 15(18): 850-856. https://doi.org/10.17485/IJST/v15i18.1551

* **Corresponding author**.

jonlaurencewenceslao@nipsc.edu.ph

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Background: This study aimed to test the mean network performance of the proxy server from the Out-Of-The-Box (OOTB), caching, filtering, and bandwidth allocation configurations when categorized according to the time of access and type of websites. More so, determine significant differences in the network performance of the proxy server with the OOTB and compared it with caching, blocking, and bandwidth allocation configuration when categorized according to the time of access and type of websites. **Methods:** Experimental design was employed to determine the difference in the network performance of proxy servers using three configurations namely, caching, filtering, and bandwidth allocation as compared to the network performance of proxy-server with the OOTB configuration. These three configurations were made as independent variables that determined the network performance of a proxy-enabled server. The researchers used the Squid Proxy version 3.4. The data were obtained by setting up a computer system running Ubuntu Linux version 19.10. Four sets of experiments were conducted to test the performance of the OOTB, caching, filtering and bandwidth allocation configurations simultaneously. The time of access was defined as A.M. session and P.M. session while the type of websites was defined as static webpages and dynamic webpages. The mean and t-Test were the statistical tools used in the study where t-Test was set at 0.05 alpha level. **Findings:** It was found out that the caching configuration, in both time of access and type of websites, was the most efficient among the three configurations. The proxy-enabled server with blocking configuration was found to have the least performance efficiency for both times of access and types of websites among the three configurations. **Novelty :** With the results of this study, it is highly recommended to network administrators to employ caching configuration on proxy servers to improve the performance efficiency of computer networks.

Keywords: ProxyServer; OutOfTheBox; Caching; Filtering; Bandwidth Allocation

1 Introduction

Bandwidth is generally referred to as the volume of information per unit of time that a transmission medium (like an internet connection) can handle(1). In modern business-oriented offices where the internet is found as the better half of daily transactions, the demand for faster task accomplishments rapidly increases. This implies that the bandwidth utilization over the business' network must be efficiently used. To be efficient in using bandwidth over the network, however, the network traffic must be free from congestion and bottleneck. But since this typical problem is unavoidable, experts in networking created different strategic plans to ease the issue. Different solutions were raised to somehow resolve the problem in network traffic. Obviously, subscribing to additional bandwidth is the most logical step(2). However, it also requires additional expenses to the organization. An alternative option is to employ a proxy server to decongest the network bottlenecks by improving network performance(3).

A proxy server is a dedicated computer or a software system running on a computer that acts as an intermediary between an endpoint device, such as a computer, and another server from which a user or client is requesting a service(4,5). When a client or end-user accessed the web page through the proxy server, it first checks whether the requested page has already been previously accessed. If so, it will serve the requested webpage from the proxy server; otherwise, it will forward the request to the source's location. This can result to better performance by way of caching. Caching run at a fraction of the cost of bandwidth increases and allows large organizations to access and retrieve webpages quickly and easily(6). It can reduce web user delays in addition to reducing network traffic and the load on web servers(7). Many techniques have been introduced to improved network performance through caching. For instance, Benadit and Francis(8) used a Very Fast Decision Tree Classifier to improve the performance of a proxy cache. Their experiments revealed that their proposed approach indeed improved the overall performance of the cache replacement of algorithms. In another work, Tiwari, Kumar, and Khan(9) developed an algorithm for Distributed Web Cache, which incorporates cooperation among proxy servers of one cluster. Test results showed that congestion and scalability problems were dealt with by the clustering concept. It yielded in higher hit ratio of caches, with lesser latency delay for requested pages.

Aside from caching, proxy servers can also be configured to perform filtering and bandwidth management. In filtering, all web traffic is checked against a defined database, and then permits or denies access to a site based on whether it is found in the database(10,11). Filters can be implemented using a software program on a personal computer and via network infrastructure such as proxy servers that provide Internet access(12). The goal is to limit inappropriate web surfing while preserving network bandwidth and worker productivity(13). The proxy servers with filtering configuration usually contain a black-list which identifies the URLs considered as inappropriate and therefore must be blocked. On the other hand, it also contains a white-list that has permissible sites(14). However, some network administrators would argue that because the individual web pages being requested are being checked as they passed the proxy servers against a pre-existing "black-list", it becomes a choke point resulting in a slower response.

On the other hand, bandwidth management is a set of mechanisms that control data rate allocation, delay variability, timely delivery, and delivery reliability(15). The primary goal of bandwidth management is to improve network performance by monitoring and setting a limit as to the allocated resources for each user or group of users. Network administrators will be shaping the traffic by allotting a certain amount of bandwidth to an individual client connecting over the network and prioritizing those clients who critically need it. However, Sharma, Kumar, and Thakiu(16) believed that due to the increasing number of network users, no amount of bandwidth can be said to be enough to satisfy the ever-growing demands of the user community. Thus, a well-planned and implemented bandwidth management can be a key in improving the performance of the existing Internet Connection. Renuka and Prafulla Shashikiran(17) attempted to improve the proxy server performance by introducing a model utilizing load balancing between the live servers and proxy servers. They concluded that their technique provided no additional overhead and subsequently improved the proxy server's performance. Network administrators and industry practitioners argued that setting the proxy server using different configurations optimized the network performance of the Internet service. However, one configuration may provide a better network performance improvement as compared to the other configurations.

The goal of this study was to investigate the network performance of a proxy server using different configurations. In particular, the researchers wanted to determine the mean network performance of the proxy server from the OOTB, caching, filtering, and bandwidth allocation configurations when categorized according to the time of access and type of websites. They also wanted to test whether significant differences in network performance of the proxy server with the OOTB and compared it with the caching, filtering, and bandwidth allocation and categorized according to the time of access and type of websites.

2 Research Methodology

2.1 Research Design Used

An experimental research design was used in this study. In[18], he defined experimental research design as a blueprint of the procedure that enables the researcher to test his hypothesis by reaching valid conclusions about relationships between independent and dependent variables. It refers to the conceptual framework within which the experiment is conducted. In doing such, the researchers manipulated over a certain variable and observed the effect on other variables. Experiments were conducted to determine the network performance of the proxy server in four (4) configurations namely OOTB, which was the basic configuration, caching, filtering, and bandwidth allocation, which were considered customized configurations. The data collected were used to compare the network performances of the OOTB versus caching, OOTB versus filtering, and OOTB versus bandwidth allocation as to times of access and types of websites.

2.2 Source of Data

In this study, the researchers first determined the offices that were available to participate in the experiments. Each office must have a computer unit that served as a workstation. These workstations have already been connected to the school's computer network. The researchers then asked permission from the officials of the identified offices to change some settings in the workstations so that those could pass through the proxy server.

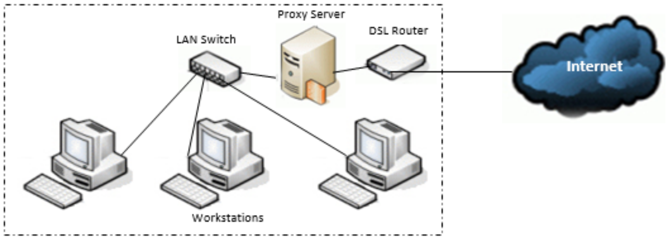

Fig 1. The Network Topology Used in the Study

Meanwhile, the researchers identified one (1) computer unit and configured it to become a proxy server by installing Squid version 3.4 inside Ubuntu Linux 19.10. The proxy server was then placed between the network switch and the DSL router to act as an intermediary agent between the clients or workstations and the Internet[15]. Figure 1 shows the network topology as implemented in the study.

To determined and record the browsing time of each identified website, the researchers used Google Timer Interface. This is a built-in tool in the Google Chrome web browser that is used to monitor the actual content downloading and can be accessed by clicking the Developer Tool at the web browser settings. The researchers used this tool to determine the browser's loading time of the web pages during the conduct of the actual experiments.

2.3 Data Collection Procedures

In this study, the researchers compared the network performance of proxy servers using different configurations according to time of access and type of websites. The time of access refers to the time of the day where clients simultaneously accessing the Internet defined as A.M. and P.M. Sessions. On the other hand, the type of website refers to the type of content being browsed by the client defined as static and dynamic websites. These variables were important to determine the network performance of proxy servers using different configurations.

During the experiments, the researchers first gathered the browsing time of the proxy-enabled network with OOTB configuration that served as the baseline performance. All of the identified offices were visited by the researchers and performed the simulations in the office's workstation. While there were eight computer units connected to the network, three of these were allowed to access Facebook or YouTube websites while the data gathering in the remaining five computer units was on-going. This was to ensure that there was a fair utilization of bandwidth at any given time mimicking the usual day-to-day operations of the offices.

After three days of data gathering and monitoring for the proxy-enabled network with OOTB configuration, the researchers reconfigured the proxy server to activate the caching configuration. The same numbers of computer units were used with the same data gathering procedures. It was followed by reconfiguring the proxy server with filtering and bandwidth allocation configurations, deactivating the previous settings with activating the new setting, as the case may be. For filtering configuration, the YouTube and Facebook websites were blocked for access while in the bandwidth allocation; the researchers allocated 175Kbps of each computer unit. All in all, the experiments lasted for twelve days.

To be able to determine whether there was an improvement in the network performance yielded by a particular configuration, its average score must be lower than the OOTB configuration. It meant that the web pages were loaded faster into the browser. On the other hand, a higher average score of a particular configuration than the OOTB meant that the web pages were loaded slowly into the browser.

3 Results and Discussions

3.1. Average Network Performance of Proxy-Enabled Network with OOTB, Caching, Blocking and Bandwidth Allocation When Categorized According to the Time of Access

As shown in Figure 2, the average performance of the proxy-enabled network during the A.M. sessions revealed that with OOTB was 81.7505s; caching was 59.1614s; filtering was 71.9893s and bandwidth allocation was 71.5694s. The results revealed that there were positive improvements in the network performance when the proxy-server was configured to each setting as compared to the out-of-the-box proxy-server configurations. It simply meant that the three configurations yielded better performance than the OOTB with the caching configuration having the best improvement.

In the P.M. sessions, the average network performance showed that with OOTB was 77.5774s; caching was 72.8317s; filtering was 136.6944s and bandwidth allocation was 71.4066s. From these results, it was noted that the caching and bandwidth allocations resulted in positive improvements since the average performance from these configurations were all lower as compared to that of the OOTB configuration. However, in the filtering configuration, there was a decrease of 56.72% in the performance with reference to the OOTB result.

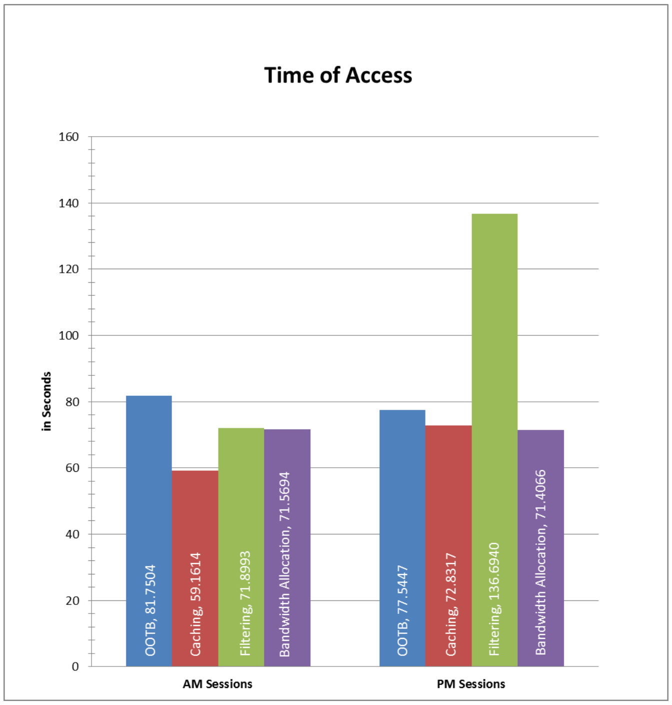

Fig 2. Average Network Performance of Proxy-Enabled Network with OOTB, Caching, Blocking and Bandwidth Allocation when categorized According to the Time of Access

3.2. Average Network Performance of Proxy-Enabled Network with OOTB, Caching, Blocking and Bandwidth Allocation When Categorized According to the Type of Websites

When categorized according to the type of websites, the experiments revealed that when the type of website is static, the OOTB, as the benchmark, was able to finish content loading the content at 7.2972s. When the proxy server was configured by caching, its average score was 4.2972s; filtering was 4.8648s while bandwidth allocation was at 6.9231s.

It indicated positive improvements in all configurations are compared to the OOTB. Therefore, it is sufficient to say that the three configurations indeed generated better performance than the OOTB.

For dynamic webpages, the network performance of the OOTB configuration was 151.792s. When the proxy server was configured to caching, the network performance was 127.2343s; for filtering, it was computed at 202.875s while bandwidth allocation, the network performance was produced 136.0553s. These results showed that there were positive improvements in the network performance when the configurations were caching and bandwidth allocation. On the other hand, when the configuration was filtering, the network performance became slow. It is naturally so since in the filtering configuration, the requests of the end-users still need to be checked against the blacklist and thus can also take time to process. Figure 3 shows the results.

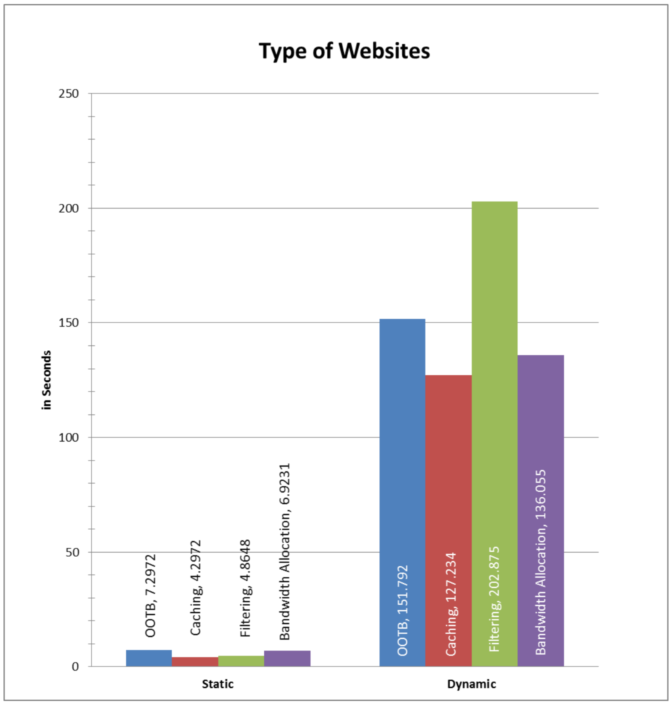

Fig 3. Average Network Performance of Proxy-Enabled Network with OOTB, Caching, Blocking and Bandwidth Allocation when categorized According to the Type of Web

3.3. Difference in the Network Performance of Proxy-enabled Server with OOTB against Caching, Filtering and Bandwidth Allocation When Categorized According to the Time of Access

Table 1 showed the t-Test result on the network performance of proxy-enabled server with OOTB against the proxy server configured with caching, filtering, and bandwidth allocation when categorized according to the time of access. Upon comparing the network performance of the proxy server with OOTB versus the proxy server with caching configuration, the obtained Sig. (2-tailed) value was .037. This was lower than the 0.05 alpha level which meant that there was a significant difference between the performance of the two configurations.

When comparing the proxy server with OOTB versus filtering configuration, the Sig. (2-tailed) result was .254 while comparing OOTB versus bandwidth allocation the Sig. (2-tailed) result was .982. These results were all higher than the 0.05 level of significance. Therefore, no significant differences were found between OOTB versus filtering and OOTB versus bandwidth allocation configurations respectively.

Table 1. Difference in the Network performance of Proxy-Enabled Server When Configured with Caching, Filtering and Bandwidth Allocation When Categorized According to the Time of Access.

Configurations	t-Test	df	Sig. (2-tailed)
OOTB versus Caching	-2.087	958	.037*
OOTB versus Filtering	-1.141	958	.254
OOTB versus Bandwidth Allocation	.023	958	.982

*Significant at 0.05 Alpha level.

3.4. Difference in the Network Performance of Proxy Server with OOTB against Caching, Filtering and Bandwidth Allocation When Categorized According to the Type of Websites

When categorized according to the type of websites, the t-Test statistic showed that the network performance when comparing the proxy server with OOTB against proxy server being configured with caching, the proxy server with OOTB versus the proxy server with filtering configuration, and the proxy server with OOTB versus the proxy server with bandwidth allocation, the obtained Sig. (2-tailed) values were .000 respectively. Thus, the results showed that were significant differences in the network performance when the proxy server with OOTB was compared against the proxy server being configured with caching, filtering, and bandwidth allocation when categorized according to the type of websites. Table 2 shows the result.

Table 2. Difference in the Network performance of Proxy-Enabled Server When Configured with Caching, Blocking and Bandwidth Allocation When Categorized According to the Type of Websites

Configurations	t-Test	df	Sig. (2-tailed)
OOTB versus Caching	-23.436	958	.000*
OOTB versus Filtering	-3.512	958	.000*
OOTB versus Bandwidth Allocation	-22.229	958	.000*

*Significant at 0.05 Alpha level

These findings supported the observation made by Hao et al.[19] who argued that caching configuration done in a proxy server is an effective method to improve the response time and ultimately network traffic performance. Sofi & Garg[11] argued that by storing popular documents, caching proxies can save network traffic and reduce web latency. Moreover, Lin et al.[20] observed significant improvement in the performance when they employed an early detection filtering algorithm that would decide whether to block or pass the requested URL through the proxy server. Also, Chitanana[21] concluded that managing bandwidth to provide quality of services for university mission-critical applications is important since it is not practical to meet the increased demand for bandwidth by simply buying more.

As network administrators plan and implement proxy server configurations to improve the performance of computer networks, the findings of this study could help them decide as to the specific configuration they can implement. When the consideration is the time of access, it would be more efficient to use the caching configuration. On the other hand, when configuring the proxy server with reference to the types of websites, any of the three options can be implemented.

4 Conclusion

The network performance of the proxy server when configured to caching, filtering and bandwidth allocation performed better as compared to the proxy server with OOTB when the time of access was in the A.M. sessions. The caching and bandwidth allocation configurations were also better in performance while the filtering configuration was slower than the OOTB in the P.M. sessions. Similarly, the proxy servers configured to either caching, filtering and bandwidth allocation were also better in performance when the type of websites was static. For dynamic websites, the proxy server with caching and bandwidth allocation outperformed the OOTB while the proxy server with the filtering configuration showed slower performance than the OOTB.

There was a significant difference between the proxy server with OOTB and caching configurations based on the time of access. On the other hand, the filtering and bandwidth allocation configuration showed no significant differences in the network performance of the proxy server with OOTB configuration in both A.M. and P.M. sessions.

In addition, it was found out that there were significant differences in the network performance between the proxy server with OOTB configuration and the proxy server with caching, blocking, and bandwidth allocation when categorized according to the types of website.

Finally, it was found out that the caching configuration in both times of access and types of websites, proved to be more efficient in network performance among the three configurations. The proxy server with filtering configuration was seen to have the least results for both times of access and types of websites among the three configurations.

5 Recommendations

For future researchers, similar studies may be conducted employing other configuration settings such as web caching versus proxy caching, transparent proxy, packet shaping and other methods.

References

1) What is Bandwidth? . . Available from: https://www.lifewire.com/what-is-bandwidth-2625809.Dateaccessed:04/11/2021.
2) Abiona O, Oluwaranti A, Oluwatope A, Bello S, Onime C, Sanni M, et al. Proxy Server Experiment and Network Security with Changing Nature of the Web. *International Journal of Communications, Network and System Sciences*. 2014;07(12):519–528. Available from: https://dx.doi.org/10.4236/ijcns.2014.712052.
3) Ali AHM, Mustafa DA, ABDALLA DAAS. Proxy Impact on Network Performance Review Study. *International Journal of Innovative Research in Science, Engineering and Technology*. 2014;03(11):17107–17113. Available from: https://dx.doi.org/10.15680/ijirset.2014.0311002.
4) Proxy Server. . Available from: http://whatis.techtarget.com/definition/proxy-server.
5) Chhabra Y. A Study of Recent Research Trends of Proxy Server. *International Journal of Advanced Technology in Engineering and Science*. 2015;3(01):159–164. Available from: http://ijates.com/images/short_pdf/226a.pdf.
6) Caching Content Can Cut Down on Internet Bandwidth Usage. . Available from: http://www.its-networks.com/caching-content-can-cut-down-on-internet-bandwidth-usage/.Dateaccessed:11/7/2021.
7) Kumar C, Marston S. Accelerating the Internet in the presence of Big Data: Reducing user delays by leveraging historical user request patterns for web caching. *Journal of Information Systems and Technology Management*. 2019;16(1):1–10. Available from: https://dx.doi.org/10.4301/s1807-1775201916006.
8) Benadit PJ, Francis FS. Improving the Performance of a Proxy Cache Using Very Fast Decision Tree Classifier. *Procedia Computer Science*. 2015;48:304–312. Available from: https://dx.doi.org/10.1016/j.procs.2015.04.186.
9) Tiwari R, Kumar K, Khan G, Patel RB, Singh BP. Load Balancing in Distributed Web Caching: A Novel Clustering Approach. *AIP Conference Proceedings*. 2010;p. 341–345. doi:10.1063/1.3526228.
10) URL Filtering vs Content Filtering. . Available from: https://networkinterview.com/url-filtering-vs-content-filtering/.Date.
11) Sofi A, Ahmad, Garg A. Analysis of various techniques for improving Web performance. *International Journal of Advance Research in Computer Science and Management*. 2015;3(3):271–279.
12) Karthikeyan VKT. Web Content Filtering Techniques: A Survey. *International Journal of Computer Science & Engineering Technology*. 2014;5(3):203–208. Available from: http://www.ijcset.com/docs/IJCSET14-05-03-038.pdf.
13) How To Speed Up Web Performance While Using URL Filtering. . Available from: https://www.computerworld.com/article/2568997/how-to-speed-up-web-performance-while-using-url-filtering.html.Date.
14) Polpinij J, Sibunruang C, Paungpronpitag S, Chamchong R, Chotthanom A. A web pornography patrol system by content-based analysis: In particular text and image. *2008 IEEE International Conference on Systems, Man and Cybernetics*. 2008;p. 500–505. doi:10.1109/ICSMC.2008.4811326.
15) Bandwidth Control. . Available from: https://mikrotik.com/testdocs/ros/2.9/root/queue.php.Date.
16) Sharma, Vikas, Kumar, Viram, Thakiu B, Singh. Need of Bandwidth Management and Formulation of Policy Framework for Effective Utilisation of Internet Services within a University Campus. *International Journal of Computer Science and Communication*. 2011;2(1):173–178. Available from: http://csjournals.com/IJCSC/PDF2-1/Article_31.pdf.
17) Renuka BS, Shashikiran GTP. Model of Load Distribution Between Web Proxy Servers Using Network Traffic Analysis. *SN Computer Science*. 2020;1(2):103–103. Available from: https://dx.doi.org/10.1007/s42979-020-0108-7.
18) . . Available from: https://www.yumpu.com/en/document/view/10797014/module-r13-experimental-research-and-design.Dateaccessed.
19) Hao W, Fu J, He J, Yen IL, Bastani F, Chen IR. Extending Proxy Caching Capability: Issues and Performance. *World Wide Web*. 2006;9(3):253–275. Available from: https://dx.doi.org/10.1007/s11280-006-8556-0.
20) Lin PC, Liu MD, Lin YD, Lai YC. An Early Decision Algorithm to Accelerate Web Content Filtering. *Information Networking Advances in Data Communications and Wireless Networks*. 2006;p. 833–841.
21) Chitanana L, Starlin M, Jones M. A Comparative Study of Cognitive and Affective Outputs of Students with Computer Prior-Knowledge (CPK) and Students without Computer Prior-Knowledge in Their Computer Studies. *Journal of Educational and Developmental Psychology*. 2011;1(1):62–76. Available from: https://dx.doi.org/10.5539/jedp.v1n1p95.