Query Operation in Trader Service: LDAP Directory versus Database

Farkhana Muchtar* 1, Ajibade Lukuman Saheed 2, Mohd Helmy Abd Wahab 3, Syed Zulkarnain Syed Idrus 4

1, 2 Faculty of Computing, Universiti Teknologi Malaysia, 81310, Skudai, Johor, Malaysia
3 Faculty of Electronic and Electrical Engineering, Universiti Tun Hussein Onn, 86400 Batu Pahat, Johor, Malaysia
4 Faculty of Applied and Human Sciences, Universiti Malaysia Perlis, 01000 Kangar, Perlis, Malaysia

E-mail: farkhana@gmail.com, ajibade.sa@gmail.com, helmy@uthm.edu.my, syzul@unimap.edu.my

Abstract. This study inspects the capability of LDAP in improving the performance of query operation in the trading service. Thus, an experiment is conducted to distinguish the performance result gained from a trading service which uses LDAP with a trading service that uses relational database as its backend storage. From the outcome of the experiment conducted, we verified that a trading service that use LDAP provide a better performance as compared to a trading service that relational database.

1. Introduction
LDAP is not the only storage that supports fast read access. There are also other storage varieties that have a faster read operation compared to LDAP. Hence, in this section, we shall take into consideration another vital requisite of the trading service besides performance, which is the scalability criterion. The topic of scalability in the trading service is also critical, especially for commercial and enterprise requirements [1].

The simple file (flat file with text and/or binary content), simple DBMs, which is a database manager (Berkeley DB, GNU DB, etc.) and other primary storage such as memory are the instances of storage types that have a faster read operation associated to LDAP. Nevertheless, none of these storage categories can satisfy the trader server requirements due to data size limitations and number of records that can be stored at any one time. Moreover, all the other storages stated have a scalability drawback.

Other than LDAP, there are also other directory services that available and are used in many applications. X.500 will certainly not be chosen as it is a directory service with a complex architecture, and it is not designed for the performance factor. Historically, LDAP substitutes X.500 as a general directory service attributable to its simplicity and performance aspect [2-3].

Other directory services such as finger, DNS, and NIS, are special purpose directory services which stores data in the form of a flat structure [4]. The core problem with these directories is that it was by no means intended to oblige as a general-purpose directory as was LDAP. Another drawback
of these directories is that it cannot sustenance the scalability requisite. Therefore, these directory services are not appropriate to be used as the backend storage for a trader server.

All the above proclamations state the motive why we choose LDAP instead of other storage. Additional description on the subject of LDAP and its benefits will be described in more depth in the next section.

2. Related Work

2.1. Related Work on Performance Evaluation of LDAP’s Read Operation

In this section, two series of past study that emphasized the benefits of LDAP particularly in the read-optimized process will be described in depth.

According to author in [5], directory servers (especially LDAP) are designed to do more than read activity than write. Therefore, the length of time for the write to be higher than the read is as expected. This feature is the main reason LDAP was chosen as the solution in this research to improve the performance of trader query operation.

For this research, only the first experiment from author [5] was considered because it involved the LDAP directory service load to determine performance differences in the read and write mechanism. Dummy process is generated as a load to the directory service where dummy process generator and LDAP are placed on a single machine to avoid network effect affecting result. From the experiments performed, both read and write performance improved almost linear and the write operation showed 40-400 percent more expensive compare to read operation.

Study done by Wang et al [6] is another study that has a close resemblance with [5]. The research conducted by him is on the use of tools for LDAP performance benchmarks. Contribution provided includes benchmarking on overall LDAP performance in terms of latency and throughput. The scaling used in the research is directory size, entry size, session re-use and several factors that determine the scalability of LDAP. In addition Wang et al. also investigated the need for modification and usage patterns in improving the performance of LDAP server.

Besides taking into consideration the values of latency and throughput during the search operation, his work also did a comparison experiment on the latency and throughput between the search (read) operation and add (write) operation in order to further recognize the directory server load. The results of his experiment are among the evidence that LDAP storage is read optimized storage.

Based on the criteria contained in LDAP it shows the potential to be used as backend storage for trader servers in trading service. Hierarchical data enables the data search process to be more specific and to speed up the process of data access than data stored in a tabular form. In addition, the communication protocol used by LDAP is simpler than the database and can reduce the communication bottleneck between application and storage. These features also provide LDAP read optimized storage features. At the same time, the optimized read criteria that exist in LDAP have also been proven in the research conducted by author in [5] and [6]. Both findings show that read operation performance is significantly higher than write operation performance. Therefore, read optimized criteria in LDAP have been proven practical and they are not conceptual.

Since LDAP was formerly released until today, the characteristic that is defined by LDAP directory is to provide quick response to high-volume search operations. Hence, there are various works that have suggested LDAP as a query and manipulation language for many applications e.g. [7-113].

Similarly to applications in the works stated above, the characteristic of LDAP which is prime to read operation is also allied with the requisite of the trading service in query operations. Therefore, this study is commenced to make utilize of the LDAP advantage to increase the performance of query responses in the trading service.
2.2. Related Work on LDAP’s Read Optimize Criteria

As compared to the database storage, LDAP is designed as read optimized storage (read to write ratio scale) [5, 14-20]. Equated to other directory services, the use of LDAP is wide and varied. Moreover, there are numerous choices of LDAP products exist either commercially or created by the open source community.

According to Le [21], LDAP has possesses its own uniqueness as compared to the database, but both of them are used for different determinations. The database is more appropriate in an environment that required data integrity (transactional support) and, at the same time, needed the write to read ratio in which the performance is emphasized for the inserting and updating of data operation. Meanwhile, LDAP is more suitable for an environment that performs read operation more often than write operation. For a more detailed argument on LDAP, we denote to [4, 14, 22-25]. Beside these, there are abundant of collected works, internet standards and tutorial that describe LDAP’s read optimize features e.g. [3-4, 14, 25-28].

2.3. OpenLDAP Compared to Other Products

Howard Chu compared OpenLDAP, FedoraDS, ApacheDS and OpenDS in 2007 [29] and determined that OpenLDAP was significantly faster in all features (authentication rate, search rate, latency and database load times). Symas compared OpenLDAP and Fedora DS in 2007 [30] and revealed that OpenLDAP has lower database load time, higher authentication and search rates than Fedora Directory Server. Thornton, Mundy and Chadwick analyzed seven LDAP server products in 2003 [31]. OpenLDAP performed splendidly compared to products by Microsoft, IBM and Novell. Author in [32] compared OpenLDAP to Microsoft and Novell options in 2001 and revealed that OpenLDAP would require more work to compete in performance and features. All of these comparisons are linked to these research study since they recognize that OpenLDAP is the fastest open source LDAP server. They present a baseline of tests required to estimate OpenLDAP performance which is a main part of these research study.

3. Design and Implementation

Two kinds of trading service with diverse backend storage were developed in this article: trading service using OpenLDAP and trading service using a MySQL. It is vital to note that in this article, the trader server that utilizes LDAP as its backend storage is recognized as OtosuL (an acronym for OMG Trading Service using LDAP) while the trader server that uses a database as its backend storage is known as OtosuD (an acronym for OMG Trading Service using Database).

A system was developed to represent the implementation of the proposed solution. This system comprehended the trader server, the trader client (exporter and importer) and the trader console. Generally, the experiment process for this research is presented in Figure 1.
4. Results and Discussion

It is essential to note, every experiment utilised a different independent parameter to denote different situations. As a hint, blue line in the graph shall denote OtosuL, while the green line in the graph shall denote OtosuD. To examine whether the performance of OtosuL was superior to the performance of OtosuD, the mean scores of both data were used for each experiment.

Table 1 portrayed that the mean score for query response time for OtosuL was 0.60680 and the mean score for OtosuD was 0.076833.

**Table 1. Descriptive Statistic Result for the Experiment’s Number of Offers in Storage**

| Description               | Type   | Value (N) | Mean       | Std Deviation | Std Error Mean |
|---------------------------|--------|-----------|------------|---------------|----------------|
| Query response time (s)   | OtosuD | 100       | 0.076833   | 0.0376247     | 0.003762       |
|                           | OtosuL | 100       | 0.060680   | 0.0288986     | 0.002889       |

Table 2 demonstrated that the mean score for query response time for OtosuL was smaller than the mean scores for query response time for OtosuD.

**Table 2 Descriptive Statistic Result for Experiment’s Number of Offers Returned**

| Description               | Type   | Value (N) | Mean       | Std Deviation | Std Error Mean |
|---------------------------|--------|-----------|------------|---------------|----------------|
| Query response time (s)   | OtosuD | 110       | 0.165783   | 0.0241873     | 0.0023062       |
|                           | OtosuL | 110       | 0.107320   | 0.0053088     | 0.0005062       |
| Query response time (s)   | OtosuD | 110       | 0.193878   | 0.0203738     | 0.0019426       |
|                           | OtosuL | 110       | 0.143970   | 0.0042768     | 0.0004078       |
| Query response time (s)   | OtosuD | 110       | 0.230778   | 0.0201488     | 0.0019211       |
|                           | OtosuL | 110       | 0.180039   | 0.0047314     | 0.0004511       |

Table 3 shown that the mean scores of query response time for OtosuL was always lower than the mean scores of query response time for OtosuD.

**Table 3. Descriptive Statistic Result for Experiment’s Number of Properties/Constraints**

| Description               | Type   | Value (N) | Mean       | Std Deviation | Std Error Mean |
|---------------------------|--------|-----------|------------|---------------|----------------|
| Query response time (s)   | OtosuD | 100       | 0.024100   | 0.0016907     | 0.0001691       |
|                           | OtosuL | 100       | 0.021680   | 0.0011360     | 0.0001136       |
| Query response time (s)   | OtosuD | 80        | 0.024812   | 0.0014849     | 0.0001660       |
|                           | OtosuL | 80        | 0.022500   | 0.0015178     | 0.0001697       |
| Query response time (s)   | OtosuD | 60        | 0.025033   | 0.0015510     | 0.0002002       |
|                           | OtosuL | 60        | 0.024200   | 0.0011320     | 0.0001461       |

Table 4 illustrated that the mean score for query response time for OtosuL was 0.172983 and the mean score for OtosuD was 0.270684.
Table 4. Descriptive Statistic Result for Experiment’s Number of Inheritance Nest

| Description               | Type  | Value (N) | Mean     | Std Deviation | Std Error Mean |
|----------------------------|-------|-----------|----------|---------------|----------------|
| Query response time (s)    | OtosuD| 70        | 0.270684 | 0.1121672     | 0.0134065       |
|                            | OtosuL| 70        | 0.172983 | 0.0566437     | 0.0067702       |

It is remarkable to note that the outcomes attained from each experiment (as in Table 1, Table 2, Table 3 and Table 4) revealed that OtosuL had continually given a superior performance as compared to OtosuD in query process of trading service. The magnitude of difference recognised when reading data from an LDAP versus gaining the same data from MySQL for each experiment lies on the variances of the characteristic in data that is designed in LDAP versus MySQL. LDAP is designed and optimized to support simple data, which read often, but written rarely. MySQL on the other hand, is designed and optimized for both read and write on data and is designed to tackle highly complex data, as opposed to LDAP, which is essentially a text-based directory storage system. Additionally, databases have been develop for transaction integrity and consistency. Thus, we can determine here that the light weightiness of OtosuL comes from its main characteristic that contributes precedence on the read performance to retrieve and search information in the distributed environment.

5. Conclusion
This study is an initiative towards increasing the query performance of trader service so that it can offer a fast response when a query is performed. The method used to improve the performance of query operations in the trading service for this study is by making use of the LDAP read optimize advantage. From the outcome of the study done in this research, it was verified that there was a significant improvement in performance of the query operation when LDAP was utilised as the backend storage. This further demonstrated that the use of read optimized storage caused in a expectant performance of the trading service.

References

[1] Steinder M and Zielinski K 1997 Proc. of 13th Annual ACM SIGPLAN Conference on Object-Oriented Program, System and Language Application (OOPSLA’97), Atlanta, USA
[2] Arkills B 2003 LDAP Directories Explained - An Introduction and Analysis (Boston: Manning Publications)
[3] Howes TA 1995 The Lightweight Directory Access Protocol: X.500 Lite Technical Report TR-95-8, University of Michigan, USA
[4] Howes TA, Smith MC and Good GS 2003 Understanding and Deploying LDAP Directory Services, 2nd ed. (USA: Macmillan Technical Publishing)
[5] Geisler R 1999 A Remote Monitor System for Distributed Application using the Soft Real-Time Scheduler Master Thesis, University of Illinois at Urbana-Champaign, Germany
[6] Wang X, Schulzrinne H, Kandlur D and Verma D 2000 SIGMETRICS Performance Evaluation 28 (1) pp 156-165
[7] Zuzana M 2017 PGP (GPG) encryption for small and medium enterprises Master Thesis, Masaryk University, Czechia
[8] Victor EAS 2017 Performance Evaluation of Scalable and Distributed IOT Platforms for Smart Regions Master’s Thesis, Luleå University, Sweden
[9] Nick H 2015 Secure Single Sign-On A comparison of protocols Master Thesis, CCV & Radboud University Nijmegen, Netherlands
[10] Saku-MP 2015 Integration of a heterogeneous authentication environment to Active Directory. Master’s Degree Programme in Information Technology (YAMK) Technology and transport, JAMK University of Applied Sciences, Finland

[11] Kaspar M 2015 Centralized User Account Management Process in Substation Automation. Master of Science, Tallinn University of Technology. Tallinn, Estonia

[12] Johan J 2008 Identity management within an organization. Master Thesis Computer Science (Management & Technology), Radboud University Nijmegen, Netherlands

[13] Ebner H 2006 Collaborilla: An Enhancement to the Conzilla Concept Browser for Enabling Collaboration Master Thesis, Stockholm University, Sweden

[14] Giovanni C 2018 LDAP3 Documentation Release, 2.4.1.

[15] Erik G 2012 Resilient and optimized LDAP database implementation for a large scale. HLR/HSS Master Thesis, Uppsala University, Sweden

[16] Fink J 2003 User Modeling Servers: Requirements, Design and Implementation PhD Thesis University of Essen, Germany

[17] Zimmerli S, Steinemann M and Braun T 2003 SIGCOMM Computing Communication Review 33 (3) pp 145-151

[18] Desprez F, Quinson M and Suter F 2001 Proc. of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA’01) 3 pp 1421-1427 (Las Vegas: CSREA Press)

[19] Steen H and Yang C 2001 Integrating ABB Aspect Directory with Microsoft Active Directory Master Thesis, Mälardalens Högskola University, Sweden

[20] Jaakkola H 2000 Directory Enabled Networks: A Study about Security and Performance Master Thesis, Kungliga Tekniska högskolan University, Stockholm, Sweden

[21] Le N-T 2003 User Management in Distributed Systems Master Thesis, Fachbereich Informatik, Universität Hamburg, Germany

[22] Donley C 2003 LDAP Programming, Management and Integration. Greenwich, (CT: Manning Publications)

[23] Donnelly M 2000 Introduction to LDAP. The Journal of the System Administrators Guild of Australia 6 (2) pp 15-21

[24] Wilcox M 1999 Implementing LDAP (Birmingham: Wrox Press)

[25] Steinemann M.-A 2005 Distributed Architectures for Laboratory-Based E-Learning PhD Thesis, University of Bern, Switzerland

[26] Marshall B 2004 System Administrators Guild of Australia Conference (SAGE’04), Horbart Tasmania, Australia

[27] Loshin P 2000 Big Book of Lightweight Directory Access Protocol (San Francisco, CA, USA: Morgan Kaufmann)

[28] Johner H, Melot M, Stranden H and Widhiasta P 1999 LDAP Implementation Cookbook. Number SG24-6163-00 in IBM Redbooks. IBM Corporation, International Technical Support Organization, USA

[29] Howard C OpenLDAP Key Notes. http://highlandsun.com/hyc/SambaXP.pdf

[30] Symas Corporation Fedora Directory Server versus OpenLDAP. http://www.connexitor.com/blog/pivot/ entry.php?id=130

[31] Thornton EJ, Mnudy D and Chadwick DW 2003 A Comparative Performance Analysis of 7 Lightweight Directory Access Protocol Directories Technical Report, IS Institute, University of Salford.

[32] Norbert K 2001 Directory Services for Linux in comparison with Novell NDS and Microsoft Active Directory Master Thesis, Rheinisch-Westfälische-Technische Hochschule Aachen, Germany