Towards a Comprehensive XML Benchamrk

Fatima Maktoum Al-Sedairi, Mohammed Al-Badawi and Abdallah Al-Hamdani
Department of Computer Science, Sultan Qaboos University, Muscat, Oman

Keywords: XML, XML Benchmark, XQuery Performance, XML/RDBMS Mapping.

Abstract: XML benchmarks are tools used for measuring and evaluating the performance of new XML developments such as XML/RDBMS/OO mapping techniques and XML storages. With XML benchmarks, the evaluation process is done by executing predefined query-set over the benchmark's dataset members; where the performance of the new development is compared against the performance of some existing techniques. Yet, none of the existing XML benchmarks seems directly investigated the effect of sought data location on the query performance. This research is an attempt to investigate the rationale of adding the Data Dimension (DD) to the 3D~XBench features. To achieve this, a new set of queries was added to the query-set of the 3D~XBench to test the effect of changing the location of the sought records. The preliminarily experimental results have shown that the query execution time is also driven by the location of the sought data in the underlying XML database; and therefore, the Data Dimension can be added to the existing features of the 3D XML Benchmark.

1 INTRODUCTION

XML databases these days offer more effective way to represent the dramatic increase in the Web data (Roy and Ramanujan, 2000). As a result, the need for developing efficient XML technologies for representing this data is crucial, and has recently become the intension of many researchers. Thus, founding efficient and comprehensive evaluation tools for new developments has become increasingly important requirement in the field of XML database management.

Specifically, XML benchmarks are devised to mimic and test capabilities of particular types of XML database management systems based on a certain real-world scenario to get the performance result that is very useful and essential for improving DBMS technologies. Each of these benchmarks has a role in addressing a number of distinct issues related to the performance testing and evaluation of the new XML developments (i.e., to test either the application's overall performance or to evaluate individual XML functionalities of a specific XML implementation). Therefore, each benchmark must be representative of just some applications of XML technology. However, most of the benchmarks are focusing on testing data management systems and query engines (Nicola et al., 2007; Mlynkova, 2008).

In terms of functionalities, the XML benchmarks offer a set of queries each of which is designed to test a particular primitive of the query processor or storage engine (Nicola et al., 2007; Mlynkova, 2008; Sakr, 2010). To this end, the researchers have intended to use a comprehensive set of queries, which cover the major aspects of query processing to get reliable results. Yet, none of the existing XML benchmarks seem directly and explicitly concerned about testing the effectiveness of the location of sought data on the query performance. This research is based on extending the 3D XML Benchmark (3D~XBench (Al-Badawi et al., 2010)) to cover this aspect.

The rest of this paper is organized as following. Section 2 presents a brief literature about some of the existing XML benchmarks, while Section 3 describes the 3DXBench in more details. Section 4 formulates the 3DXBench extension and Section 5 provides some experimental results to test the effectiveness of the new extension. The paper is concluded in Section 6.

2 RELATED WORK

The raise of XML importance and the dramatic increase in the number of the XML new techniques

Al-Sedairi F., Al-Badawi M. and Al-Hamdani A..
Towards a Comprehensive XML Benchamrk.
DOI: 10.5220/0005474101120117
In *Proceedings of the 11th International Conference on Web Information Systems and Technologies* (WEBIST-2015), pages 112-117
ISBN: 978-989-758-106-9

have risen the need for tools to evaluate those new developments. XML benchmarks are designed to simulate and test capabilities of particular types of XML data management systems based on a certain real-word scenario in order to get the performance result that is very useful and essential for improving DBMS technologies.

Several XML benchmarks have been proposed in the literature to help both researchers and users to compare XML databases independently. Each of these benchmarks is addressing a number of distinct issues; to test either the application's overall performance or to evaluate individual XML functionalities of a specific XML implementation. As a result, each benchmark targets some applications of XML technology. However, most of them focus on testing data management systems and query engines. To this end, the benchmarks offer a set of queries, and each of which is intended to challenge a particular primitive of the query processor or storage engine. Thus, the researchers intended to use a comprehensive set of queries, which covers the major aspects of query processing. Moreover, the overall workload consists of scalable XML databases with specific aspects regarding the testing parameters of each benchmark.

Generally, XML benchmarks should be simple, portable, scaled to different workload sizes, and they allow objective comparisons of competing systems and tools (Gray, 1993). Although their applications are diverse and complex, developing meaningful and realistic benchmarks for XML is a truly a big challenge for all XML researchers. In addition, XML processing tools fall into many categories, from simple storage services to sophisticated query processors, and this adding to the complexity of developing relevant and realistic XML benchmarks.

Considering the tradition research, researches in XML benchmarks tend to compare the newly proposed XML benchmarks with the existing ones, as well as analysing the behaviour of a particular one with various types of data. Also the literature compares some of the existing XML benchmarks in (Nicola et al., 2007; Mlynkova, 2008; Al-Badawi et al., 2010; Sakr, 2010). Both dataset and query set are the main criteria which were considered by all the comparisons such as in (Al-Badawi et al., 2010; Nicola et al., 2007). Furthermore, they investigated the benchmarks from different aspects such as benchmark type (micro, Application level), the number of users, applications, schema and the key parameters of testing data.

Finally, in terms of existing XML benchmarks, the set includes XMark (Schmidt et al., 2002), XOO7 (Bressan et al., 2003), XBench (Yao et al., 2004), XMach~1 (Böhme and Rahm, 2001), MBench (Runapongsa et al., 2006), XPathMark (Franceschet, 2005), MemBeR (Afanasiev et al., 2005) and TPoX (Nicola et al., 2007). Recently some new benchmarks were added to the list including the 3D~XBench (Al-Badawi et al., 2010), EXRT (Carey et al., 2011) and Renda-RX (Zhang et al., 2011).

3 THE 3D XML BENCHMARK

The 3D~XBench was proposed to test the effect of three XML document aspects on the XML query performance. These are the depth, breadth and the size of the underlying XML database and their reflection on the XQuery syntax; hence is called the 3D~XBench. The depth defines the number of levels in XML tree while the breadth represents the average fanouts of XML nodes. The size is measured by the number of nodes in the XML tree which is mainly used in the scalability testing.

Like other benchmarks, the 3D~XBench framework is based on executing a set of pre-defined XML queries over a number of XML databases which are selected carefully to reflect the three XML aspects mentioned above. The following two subsections explain more about the 3D~XBench's dataset and query-set respectively.

3.1 Dataset

Three different databases have been used in the benchmark from different sources which are either real or synthetic. The DBLP (DBLP, 2014) and the TreeBank (PennProj, 2014) are real databases, while the XMark (Schmidt et al., 2002) dataset is a synthetic (code generated dataset). Dataset base members (the original XML databases) are versioned two more times at 50% and 25% of the base database to vary the database size dimension. The other two dimensions are varied naturally due the nature of the used databases which were intentionally and carefully selected to reflect the depth and breadth dimensions.

Figure 1 and Table1 depict the variation aspect of the three dimensions over the benchmark's dataset.

Table 1: Characterises of the 3D~XBench's Dataset.

	Size (Base DB)	Max. Depth	Avg. Breadth
DBLP	2,439,294	6	11
XMark	2,437,669	11	6
TreeBank	2,437,667	36	3

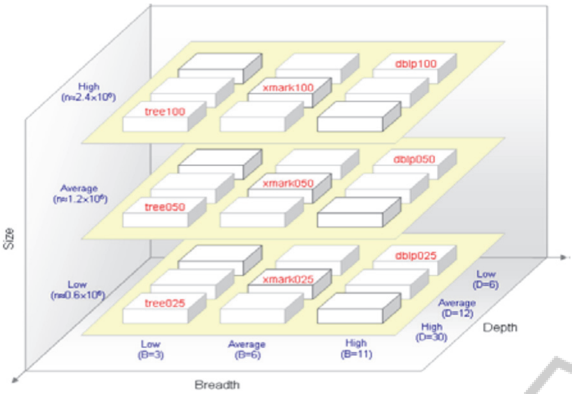

Figure 1: The 3D~XBench Architecture Design (Adopted from al-Badawi M., et al. 2010).

3.2 Query Set Design

3D~XBench adopts the query-set used by the XMark benchmark (Schmidt et al., 2002). The XMark's 20 queries are grouped into 14 categories each of which targets specific database querying functionality. Out of XMark's 20 queries, 3D~XBench adopts 10 queries only which descend from the following categories:

- Exact Matching
- Order Access
- Path Traversal
- Sorting
- Aggregation
- Reg. path Exp.
- Missing Element

4 THE EXTENSION

None of the listed benchmarks, including the 3D~XBench, has taken care of investigating the effectiveness of the sought data location on the query performance. This research is the first step in that direction.

The 3D~XBench extension is done by adding a new set of queries to the query set of the 3D~XBench to test the effect of changing the location of the sought records. The Data-Dimension, whenever is linked with other features of the 3D-XBench, is expected to strengthen the benchmark's testing capabilities and make it a comprehensive testing model than ever found in the literature. Figure 2 illustrates the new extension graphically.

The main idea behind the extension (Data Dimension) is to divide the base dataset into three pre-set zones to test the effect of query performance among three different locations. These zones are determined by specific range of nodes from the root node. The first zone of each database is restricted to be within the first 30% of the database, while the second zone starts from 45% to 75% and the third zone comes after the 90%. Each query category is executed among the three different zones (ranges) to test the effect of the Data Dimension.

Adding the Data Dimension leads to almost the same testing requirements (framework) as for the original dimensions. For example, the dataset remained the same, while the query set was altered to include the new dimension (i.e. the Data Dimension). As a result, the query-set selection considered only those queries of which the location of the sought data matters. In addition, some queries were modified to adopt the new specifications.

Figure 2: Visualization of the New Extension of the 3D XML Benchmark.

5 EXPERIMENTAL RESULTS

In order to evaluate the effectiveness of the new extension on benchmark's testing environment, the extended 3D~XBench has been used to compare two representatives mapping techniques from the literature and observe whether the benchmark's new extension is going to produce a consistence performance over different versions of XML queries.

5.1 Mapping Techniques Selection

The research followed the same evaluation process that was used by (Al-Badawi et al., 2010) to evaluate the 3D~XBench when introduced earlier. In that, the evaluation process is based on using the XML/RDBMS mapping environment and selects a set of mapping techniques, which represent the existing once in the literature. The selected set includes the Edge (Florescu and Kossmann, 1999) and XParent (Jiang et al., 2002) mapping techniques to represent the single-relation and multiple-relations mapping techniques. These two techniques were used by (Al-Badawi et al., 2010) too. The relational schema of each mapping technique was implemented in FoxPro database engine and all dataset members (total 9 databases; 3 versions from each: DBLP, XMark, and TreeBank) were mapped to both schemas.

5.2 Query-set Selection

The previous evaluation process of the base 3D~XBecnch used 10 queries (expressed in XQuery syntax as imported from (Schmidt et al., 2002)), which were divided into 7 categories. The new evaluation process adopted a subset of those categories which, their workload can be affected by the location of the sought data. So, the experiment considered 2 categories with 2 different queries from each. These are: the exact-matching category (Shallow and Deep), and the Join-on-Value category (join and join with filter).

The 4 queries in the query-set were translated over the 9 dataset members using the 2 selected mapping techniques. Three more versions, that each targets a specific range in the underlying database, were produced from each query in the query-set members. The total number of queries in the query-set becomes 216.

5.3 Execution Conditions

The experiments was conducted in a stand-alone PC (Intel® Xeon® CPU 2.93 GHz, 6GB RAM), running Windows7 (64 bit). Further, all XML databases and XQuery queries were translated to the FoxPro relational environment, and each query was executed 20 times against the concerned database; every time the execution time is taken in mill-seconds. For the validity, the experiment considered the average of the middle 18 readings.

5.4 Preliminary Results

Due the space restrictions, this section presents only a subset of the results obtained from the above experiment. These results are illustrated by the diagrams given at the end of this paper, along with a short discussion as following.

Figure 3 shows that the Shallow Exact Matching query was slower in the first range over the shallow (DBLP) and average (XMark) databases in the single mapping technique. It went slightly faster (about one millisecond) in both databases over the other two ranges. However, the Shallow Exact Matching query was faster in the first range over the deep databases than that of the other two ranges. In general, it seems that the increasing Data-Dimension has an opposite impact on the Shallow-exact matching query as far as the single-relation mapping technique is considered.

Like in the single-relation mapping technique, DD had the same effect over the deep and shallow databases in the multiple-relations mapping techniques (Figure 7). However, DD had an inconsistent change over the average database. Generally, in all cases the DD seems not much affecting the Shallow Exact Matching query over all the databases. More generally, it seems that DD has less effect on the single-relation mapping technique than the multiple-relation mapping technique as illustrated in Figure 3 and 7.

When concerning the Deep Exact Matching query, Figure 4 clearly shows that the deep exact matching query gets slower when the search-range is increased over the wide and deep XML databases, but it gets faster over the average width/breadth database. However, the difference in query performance was much clear over the deep XML database (i.e. 39-35=4, 45-39=6) while the difference was very narrow over the wide and average-width database (i.e. the difference was only one unit).

On the other hand, Figure 8 shows the DD effect on the Deep Exact Matching query over the base databases (DBLP, XMark, TreeBank) within multiple mapping techniques (XPerent). The figure shows that there is an inconsistent change over the average and shallow databases. The elapsed time has a little decrease over the deep databases. In brief, DD has a minor effect on this query type over all database categories for the both mapping techniques.

In terms of Join on Value queries, Figure 5 presents the effect of DD over the base databases (DBLP, XMark, TreeBank) within single mapping technique (Edge). It presents that DD caused a consistent increasing elapsed time over the shallow and average-width databases; while there was an inconsistent change in the elapsed time over the deep databases as shown in the Figure.

Similarly, Figure 9 shows the effect of DD on the Join on Value query over the base databases (DBLP, XMark, TreeBank) using multiple mapping technique (XPerent). The effect of the DD within this query was a minority over the deep databases, while it produced an inconsistent elapsed time average-width over database as illustrated in the Figure. The query time was proportional few units when increasing search range over the wide XML database.

Finally, Figures 6 and 10 show the elapsed time of the “Join on Value with Range Filter” query. The elapsed time of this query type was increasing over the shallow databases in both mapping techniques. This was also valid over the average-depth database but only for multiple-relations mapping technique. However, it seems that the query performance was not that much affected over the deep database for this query type when executed over both mapping techniques as seen Figures 6 and 10.

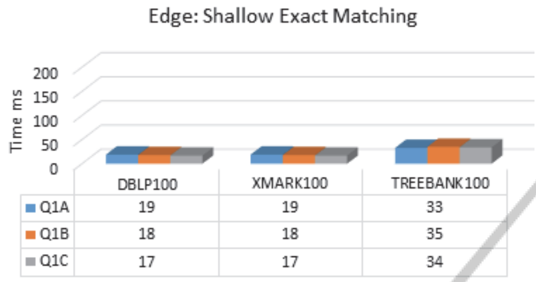

Figure 3: Edge, Base DB, Shallow Exact Matching.

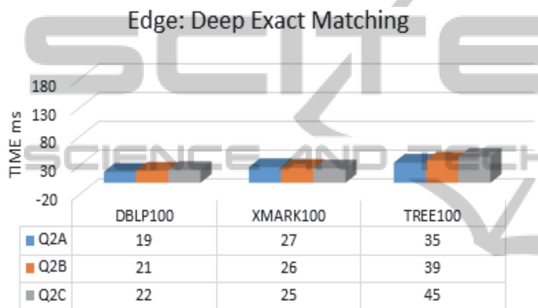

Figure 4: Edge, Base DB, Deep Exact Matching.

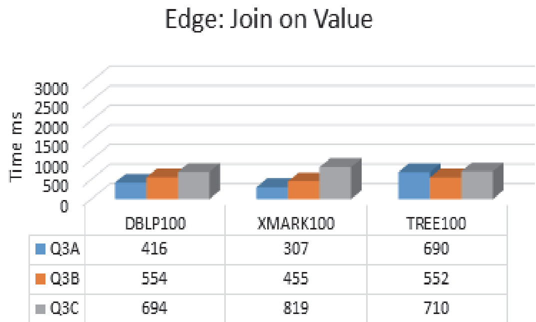

Figure 5: Edge, Base DB, Join on Value.

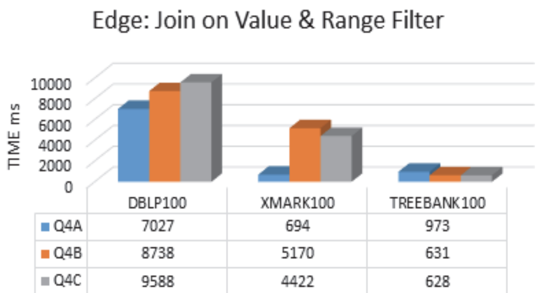

Figure 6: Edge, Base DB, Join on Value with Filter.

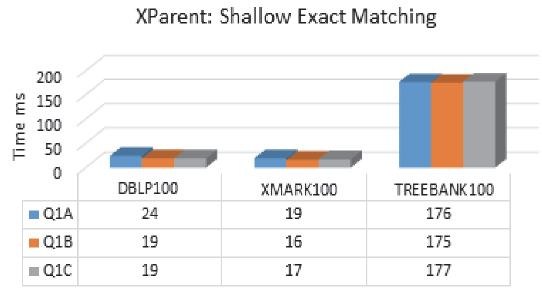

Figure 7: XParent, Base DB, Shallow Exact Matching.

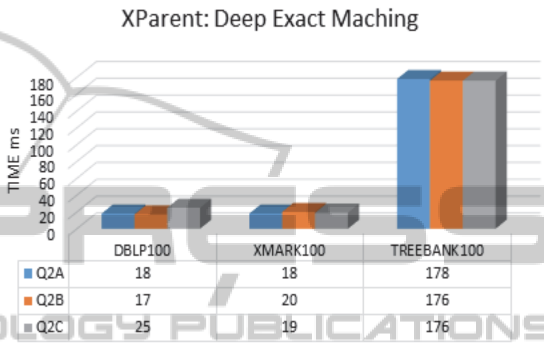

Figure 8: XParent, Base DB, Deep Exact Matching.

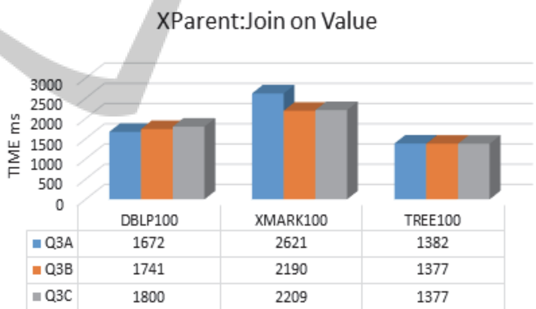

Figure 9: XParent, Base DB, Join on Value.

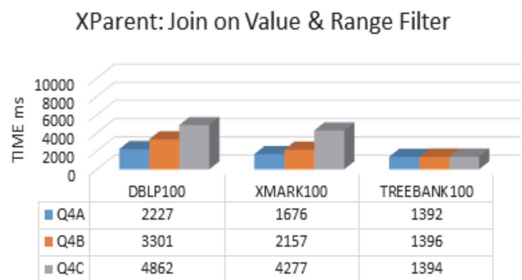

Figure 10: XParent, Base DB, Join on Value with Filter.

6 CONCLUSIONS

This paper discussed the rationale of extending the functionalities of the 3D XML Benchmark (Al-

Badawi et al., 2010) by adding a new feature, which concerns about testing the effectiveness of the sought data location on the query performance.

To evaluate the extension, a new experiment was conducted using the same datasets as in the original 3DXBench, but with an expanded query set that includes queries to test the effect of the DD. The experiment used two representative mapping techniques (one single-relation and one multiple-relation mapping techniques).

The experimental results show that the Data Dimension (DD) has a significant influence on the query elapsed time with respect of database structure (depth, breadth, size) and query categories. The performance of different mapping approaches (single vs. multiple) is also affected by DD. Thus, DD can be included as the 4th dimension in the 3DX~Bench Benchmark.

A further research can be carried out into different directions. First, one can expand the valuation process to test the effect of the DD on other query types introduced in (Schmidt et al., 2002) such as the path traversal, order access, sorting, aggregation, missing elements and others. Also, a further evaluation for the new extension can consider measuring other experimental variables such as CPU usage, memory consumption and I/O operations. Moreover, the experiment can be conducted over different mapping technique like PACD and/or native XML databases.

ACKNOWLEDGEMENTS

This research is supported by The Research Council (Oman) under the Grant RC/SCI/COMP/12/01.

REFERENCES

Afanasiev, L., Manolescu, I., and Michiels, P., 2005. MemBeR: A Micro-Benchmark Repository for XQuery. In Proceedings of 3rd International XML Database Symposium, LNCS. Springer-Verlag.

Al-Badawi, M., North, S., and Eaglestone, B., 2010. The 3D XML Benchmark. In the proceedings of the WebIST'10, pp. 13-20, Valencia, Spain.

Böhme, T. and Rahm, E., 2001. XMach-1: a Benchmark for XML Data Management. In Datenbank systeme in BuroTechnik und Wissenschaft, 9. GI-Fachtagung, pp.264–273, Springer-Verlag, London, UK.

Bressan, S., Lee, M-L., Li, Y.G., Lacroix, Z. and Nambiar, U., 2003. The XOO7 benchmark. In Proceedings of VLDB 2002 Workshop EEXTT and CAiSE 2002 Workshop, pp.146–147, London, UK.

Carey, M. J., Ling, L., Nicola, M., and Shao, L., 2011. Exrt: Towards a simple benchmark for xml readiness testing. In Performance Evaluation, Measurement and Characterization of Complex Systems, pp. 93-109. Springer Berlin Heidelberg.

DBLP, 2014. The DBLP Website, Available at http://dblp.uni-trier.de/, [Accessed on 13/116/2014].

Franceschet., M., 2005. XPathMark: an XPath benchmark for XMark generated data. *International XML Database Symposium* (XSYM).

Florescu, D., and Kossmann, D., 1999. A Performance Evaluation of alternative Mapping Schemas for Storing XML Data in a Relational Database. INRIA Rocquencourt, France

Gray J., 1993. The Benchmark Handbook for Database and Transaction Systems. Morgan Kaufmann, San Francisco, CA, USA, ISBN 1-55860-292-5.

Jiang, H., Lu, H., Wang, W., and Yu, J., 2002. XParent: An Efficient RDBMS-Based XML Database System. International conference on Data Engineering, pages 1-2, CA, USA.

Mlynkova, I,. 2008. XML Benchmarking. IADIS International Conference Informatics, pages 59- 66.

Nicola, M., Kogan, I., and Schiefer, B., 2007. An XML Transaction Processing Benchmark. In Proceedings of ACM SIGMOD international conference on Management of data (pp. 937-948).

PennProj, 2014. The Penn Treebank Project. Available online at http://www.cis.upenn.edu/~treebank/, [Accessed on: 13/11/2014].

Roy, J., and Ramanujan, A., 2000. XML: data's universal language. IT Professional, 2(3), 32-36.

Runapongsa, K., Patel, M., Jagadish, H., Chen, Y., and Al-Khalifa, S., 2006. The Michigan Benchmark: Towards XML Query Performance Diagnostics. International Journal of Information systems, 31(2), pages 73-97.

Sakr, S., 2010. Towards a comprehensive assessment for selectivity estimation approaches of XML queries. International Journal of Web Engineering and Technology, 6(1), 58-82.

Schmidt, A., Waas, F., Kersten, M., Carey, D., Manolescu, I., and Busse. R., 2002. XMark: A Benchmark for XML Data Management. International conference on VLDB, Hong Kong, pages 1-12.

Yao, B. B., Ozsu, M. T., and Khandelwal, N., 2004. XBench benchmark and performance testing of XML DBMSs. In IEEE Proceedings, 20th International Conference on Data Engineering (pp. 621-632).

Zhang, X., Liu, K., Zou, L., Du, X., and Wang, S., 2011. Renda-RX: A Benchmark for Evaluating XML-Relational Database System. LNCS, Volume 6897, 2011, pp 578-589.