International Journal Of Computer Science And Applications Vol. 1, No. 1, June 2008
ISSN 0974-1003

Adaptive Compression Techniques and Efficient Query Evaluation for XML Databases-An overview

V. S. Gulhane
Sipna's College of Engineering & Technology, Badnera road ,
Amravati-444607
91-721-2522343
v_gulhane@rediffmail.com

Dr. M. S. Ali
Prof. & Head, Dept.of Computer Science & Engineering
Prof. Ram Meghe Institute of Technology & Research,
Badnera-Amravati - 444701
softalis@hotmail.com

ABSTRACT
Extensible Markup Language (XML) is proposed as a standardized data format designed for specifying and exchanging data on the Web. With the proliferation of mobile devices, such as palmtop computers, as a means of communication in recent years, it is reasonable to expect that in the foreseeable future, a massive amount of XML data will be generated and exchanged between applications in order to perform dynamic computations over the Web. However, XML is by nature verbose, since terseness in XML markup is not considered a pressing issue from the design perspective. In practice, XML documents are usually large in size as they often contain much redundant data. The *size problem* hinders the adoption of XML, since it substantially increases the costs of data processing, data storage, and data exchanges over the Web. As the common generic text compressors, such as Gzip, Bzip2, WinZip, PKZIP, or MPEG-7 (BiM), are not able to produce usable XML compressed data, many XML specific compression technologies have been recently proposed. The essential idea of these technologies is that, by utilizing the exposed structure information in the input XML document during the compression process, they pursue two important goals at the same time. First, they aim at achieving a good compression ratio and time compared to the generic text compressors. Second, they aim at generating a compressed XML document that is able to support efficient evaluation of queries over the data. This paper discuses survey of some of the Adaptive Compression Techniques for XML namely Xmill ,Xpress ,Xgrind.

Categories and Subject Descriptors
Compression Techniques,

General Terms
Compression. Decompression, query

Keywords
XML, Adaptive, Compression

1 INTRODUCTION:
XML: An Overview

XML or extensible markup language is a markup language for documents containing structured information. Structured information could be content and some indication of what role that content plays. XML specifies the structure and content of a document. A markup language is a mechanism to identify structures in a document. XML has a simple, flexible text format. It is a simplified subset of SGML. Here is an XML snippet:

```
<Student SID = "S1234">
<LastName>Smith</LastName>
<FirstName>Joe</FirstName>
<MInitial>M</MInitial>
<Address>620 12th Ave, Coralville</Address>
</Student>
```

Fig 1 – XML example

2 NEED FOR COMPRESSION
As evident from the snippet in Fig 1, XML representations are very large and can be up to ten times as large as equivalent binary representations . Consider the following:

1. There is lot of "redundant" data in XML documents, including white space, and element and attribute names.

2. Its self-describing and document size is larger than other formats. This will affect query processing.

3. As a self-describing format, XML brings flexibility, but compromises efficiency.

4. Most XML documents are stored in file systems, so we need an efficient way to store file-based XML.

5. XML is the lingua franca of web services, thus necessitating large volumes of XML data to be sent over networks. Reducing data size helps conserve network bandwidth.

3 DESIRABLE FEATURES OF XML COMPRESSION

Following are some desirable quality features for XML compression technology.

I. *Effective Compression.*

ii. *Expressive Query Language and Efficient Querying Engine.*

iii. *Minimal User Intervention and Auxiliary Structures*

4 COMPRESSION TECHNIQUES

XMill

XMill is a user-configurable XML compressor. The lossless compressor and decompressor are named XMill and Xdemill respectively. XMill achieves about twice the compression rate of general-purpose compressors like *gzip* at about the same speed. It does not need a document type definition (DTD) for compression and preserves the input XML file. It also allows the users to combine existing compressors in order to compress heterogeneous XML data. It is extensible with user-defined compressors for complex data types such as DNA sequences or images. XMill claims to reduce network bandwidth considerably.

Benefits

XMill achieves better compression rate compared to gzip (by a factor of 2, for data-like XML documents) without sacrificing speed. This owes to the fact than it separates structure from content. This makes it a clear winner for applications like data archiving since these applications require lesser disk space. At the same time, it reduces network bandwidth. XMill is moderately faster than gzip in XML publishing. However, relative advantage of XMill depends on the application it is used.

Disadvantages

The main disadvantages of XMill are as follows:

1. Compressed output of XMill is not queryable. To be queried, the document has to be decompressed.

2. If the size of the input document is less than 20KB, XMill will not exhibit any significant advantage over gzip.

3 . Here the compression is targeted for applications like data exchanging, data archiving etc, but not for deriving a meaningful view of the input document as is the case of compressing images or video sequences .

4. To apply specialized compressors to containers, human intervention is required to specify the required container. Path processor is configured by user commands to map values. This is inconvenient.

5. XMill precludes incremental processing of compressed documents; it actually hinders compressors other than gzip, and requires user assistance to achieve the best compression [8].

XGrind

XGrind is a queryable XML compressor, i.e., we can execute queries on XML documents compressed using XGrind. The major reason for this is that the compressed document retains the structure of the original XML document. This is very important for resource limited computing devices like palmtops. Even though resources are available, querying on compressed documents reduces query response time. Here the disk seek times are highly reduced and at the same time disk bandwidth is increased. XGrind does compression by separating data from structure, but at the same time maintains the document structure of the input document.

5 PERFORMANCE RESULTS

XGRIND has a lower compression ratio than XMILL. Results indicate that the compression ratio for XGrind improves with the increase in the number of enumerated attributes.

Benefits

1. Considerable improvements in query response time
2. Disk bandwidth is effectively increased as increased information density
3. Memory hit buffer ratio increases
4. Compresses at the granularity of element/attribute value using context-free compression scheme
5. Range and Partial match queries have on the fly
6. decompression of only those elements that feature in the query
7. predicates

Disadvantages

1. XGrind does not support several operations like non-equality selections. In addition, it cannot perform any join, aggregation, and nested queries or construct operations.

2. Compression of documents is a one-time operation; at the same time querying, could be a repeated occurrence.

3. The statistics could change if there are many updates made to the compressed XML document. Lastly, XGrind uses a fixed root-to-leaf navigation strategy, which is insufficient to provide alternative evaluation strategies.

XGrind vs. Xmill

* Compression Time: Factor of 2 of that of XMill
* Uses element/attribute granularity than document granularity
* Simple character-based Huffman coding scheme rather than a

pattern based approach

* Makes 2 passes over the original document to provide context-free compression

Xpress

Xpress also allows queries on compressed data. XPress works only on XML trees. It cannot handle ID/IDREF tags. It creates bisimilar partitions of elements in the XML document. Then, it encodes partitions by disjoint intervals and allows query evaluation by operations on these intervals. It uses an encoding method known as the reverse arithmetic encoding. It is a combination of differential and binary encoding methods. This is an efficient path encoding method, which yields fewer overheads of partial decompression and quicker path evaluation. XPRESS provides high compression ratio.

Features

- Xpress query time is 2.83 times better than that of XGrind [5].
- Xpress compression results are 80% better than XMill [6] and 3.14 times zip [7]. XML-Xpress was also faster, running 3% faster than Zip and 55% faster than XMill [7].

5. Research issues related to XML compression:

i. To exploit data semantics in XML databases and query workload statistics in order to make an XML compressor adaptive to an application domain.

ii. To design algorithm(s) for updating operations over compressed XML databases.

iii. To investigate a more effective auxiliary structure, such as an indexing scheme, to aid querying compressed XML databases.

6 CONCLUSION:

We recognize that the *size problem* already hinders the adoption of XML, since in practice, it subsequently increases the cost of data processing, data storage and data exchange over the web. In this paper, three XML specific compressors were overviewed. It can be concluded that specific application types determine the choice of the compressor. There is also a possible argument that since bandwidth is very cheap today, there is really no need for compression. However, with the advent of wireless devices such as phones, palmtops, PDAs etc, bandwidth is at a premium. Therefore, compressing data to transmit to such devices becomes essential.

7 REFERENCESS

[1] G. Antoshenkov. Dictionary-Based Order-Preserving String Compression. VLDB Journal 6, page 26-39, (1997).

[2] A. Arion, A. Bonifati, G. Costa, S. D'Aguanno, I. Manolescu, and A.Pugliese. Efficient Query Evaluation over Compressed XML Data. Proceedings of EDBT (2004).

[3] A. Arion, A. Bonifati, G. Costa, S. D'Aguanno, I. Manolescu, and A. Pugliese. XQueC: Pushing Queries to Compressed XML Data. Proceedings of the 29th International Conference on Very Large Data Bases (VLDB'03), (2003).

[4] S. Boag et al. XQuery 1.0: An XML Query Language, Nov. (2002).http://www.w3.org/TR/xquery.

[5] http://www.cse.ogi.edu/class/cse582/Lectures/Lecture15/XML compression discussant.ppt

[6] http://www.cs.uu.nl/~johani/publications/gp4xml.pdf

[7] http://www.ictcompress.com/PDF/XML WhitePaper.pdf

[8] Cheney. Compressing XML with Multiplexed Hierarchical PPM Models

[9] J. Cheng and W. Ng. XQzip: Querying Compressed XML Using Structural Indexing. Proceedings of EDBT (2004).

[10] J. Clark. XML Path Language (XPath), (1999). http://www.w3.org/TR/xpath.

[11] [11] Extensible Markup Language (XML) 1.0 (Second Edition) W3C Recommendation, October (2000). http://www.w3.org/TR/REC-xml/.

[12] J. Gailly and M. Adler. gzip 1.2.4. http://www.gzip.org/.

[13] W. Y. Lam, W. Ng, P. T. Wood, and M. Levene. XCQ: XML Compression and Querying System. Poster Proceedings, 12th International World-Wide Web Conference (WWW2003), May (2003).

[14] H. Liefke and D. Suciu. XMill: An Efficient Compressor for XML Data. Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 153-164 (2000). M.Martinez.MPEG-7Overview(version9).

[15] http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm.

[16] J. K. Min, M. J. Park, and C. W. Chung. XPRESS: A Queriable Compression for XML Data. Proceedings of the ACM SIGMOD International Conference on Management of Data (2003).

[17] pkzip. http://www.pkware.com/.

[18] T. M. Cover and J. A. Thomas. Elements of Information Theory. Wiley-Interscience, John Wiley &S ons, Inc., New York, (1991).

[19] The bzip2 and libbzip2 official home page. http://sources.redhat.com/bzip2/.

[20] P. M. Tolani and J. R. Haritsa. XGRIND: A Query-friendly XML Compressor. IEEE Proceedings of the 18th International Conference on Data Engineering (2002).

[21] Winzip. http://www.winzip.com/.

[22] XML compression techniques: A survey-by Smitha Nair-Dept.of Comp Sci. University of Iowa, USA.

[23] Comparative Analysis of XML Compression technologies: Wilfred Ng. ,James Cheng, Lam yeung, TheHong Kong university of Science & technology, Hong Kong