Comparative analysis of efficient methods for storing unstructured data into database with accent on performance

Conference Paper · June 2010

DOI: 10.1109/ICETC.2010.5529222

CITATIONS
4

READS
1,395

2 authors:

Jasmin Azemović
Dzemal Bijedic University of Mostar
17 PUBLICATIONS **74** CITATIONS
SEE PROFILE

Denis Mušić
Dzemal Bijedic University of Mostar
23 PUBLICATIONS **96** CITATIONS
SEE PROFILE

Comparative analysis of efficient methods for storing unstructured data into database with accent on performance

Jasmin Azemović, Mr.Sc.
University „Džemal Bijedić“
Faculty of Information Technologies,
Mostar, Bosnia and Herzegovina
jasmine@fit.ba

Denis Mušić, Mr.Sc.
University „Džemal Bijedić“
Faculty of Information Technologies,
Mostar, Bosnia and Herzegovina
denis@fit.ba

Abstract— **the management of unstructured data is recognized as one of the major unsolved problems in the information technology (IT) industry, the main reason being that the tools and techniques that have proved so successful transforming structured data into business intelligence and actionable information simply don't work when it comes to unstructured data. This unstructured data is often stored outside the database, separate from its structured data. This separation can cause data management complexities. Or, if the data is associated with structured storage, the file streaming capabilities and performance can be limited. User data is produced and generated on big scale where measurement is hundreds of MB or even couple of GB per user per day. Efficient method for storing and accessing those kinds of data is imperative in modern information society.**

In this paper we are presenting results of research on using different data types for storing unstructured data within database and classic file system. Research is inspired with current situation in modern information society.

Keywords-database, unstructured data, filestream, performance

I. INTRODUCTION

It is well known that one result of the Internet's rapid growth has been a huge increase in the amount of information generated and shared by organizations in almost every industry and sector. Problem is not only in generating data. Also what is with storing and accessing issues? Less well known, however, is the degree to which this information explosion has consumed huge amounts of expensive and valuable resources, both human and technical. These demands, in turn, have created an equally huge, but largely unmet, need for tools that can be used to manage what we call unstructured data. Admittedly, the term unstructured data can mean different things in different contexts. For example, in the context of relational database systems, it refers to data that can't be stored in rows and columns. This data must instead be stored in a BLOB (binary large object), a catch-all data type available in most relational database management system (RDBMS) software. Here, unstructured data is understood to include e-mail files, word-processing text documents, presentations, image files, and video files.

On the other hand reason why we have a data explosion is due advances and major transformation in magnetic recording technology. The cost of storage has dropped dramatically from over $4/Megabyte in 1990, to less than $0.001/Megabyte today [1]

According to a study by IDC [3] a leading Information Technology market research and analysis firm, the amount of data that would be captured, stored, and replicated worldwide would grow from 161 Exabyte's in the year 2006, to over 988 Exabyte's in 2010 (1exabyte= 10^{18} bytes). Two key findings of this study are:

- A majority of this data would be in the form of images, captured from a large number of devices, such as digital cameras, camera phones, surveillance cameras, and medical imaging equipment. Most of this data would need to be stored and managed in centralized systems within organizations. The study indicates that, by 2010, although enterprises will create, capture, and replicate only 30% of the *digital universe* [3], they will have to store and manage over 85% of all data in it.
- Over 95 % of the digital universe is unstructured data. According to this study, 80% of all stored data of organizations is unstructured. This growth trend is expected to continue into the future, there by mandating the need for efficient ways of storing searching, structuring, and providing security for unstructured data [3]. Similar trends about the importance of unstructured data processing have also been reported by other market research and advisory firms, such as, the Gartner Group and the Butler Group [4].

It is obviously that unstructured data is our realty and finding efficient method for storing data is key aspect of this research and future findings in this field. We will explain current methods for storing data with accent on good elements and side effects. Practical experiment and comparative analysis will conclude this paper.

II. Related Work

There is a numerous researches based on benchmarking database systems. One paper is particularly interesting and represents a good starting point for our paper "A Benchmark Suite for Unstructured Data Processing" [2].

Their primary goal was assembling workload suite, identify and study a wide range of unstructured data processing applications, distill their key processing and I/O access characteristics, and compose workloads that embody these characteristics. Therefore, in order to design storage systems for this important emerging class of applications, they are create a benchmark suite that can capture their processing and I/O characteristics

Benchmark suite consists of four workloads:

- Edge detection
- Proximity search
- Data scanning
- Data fusion

Conclusion was, that application which use unstructured data for business process are very I/O intensive and place heavy demands on storage system [2].

But there is a big space to explore, where previews researches are leave opportunity to continue those kinds of experiments with unstructured data.

III. Background of Research

In following section will explain background and key elements important for this research: different methods for storing unstructured data (advantages and disadvantages) used data types and experimental environment. Our primary goal is finding an efficient method for storing unstructured data. During that process we will make extensive comparative analysis based on well-defined parameters.

Research is based on following methods for storing unstructured data:

- Unstructured data inside relational database environment;
- Unstructured data outside relational environment;

First we will start with explaining those methods and each one benefit and side effects.

A. Unstructured data inside relational environment

A point of contention usually arises with the topic of relational databases and binary large object (BLOB) data: Is it better to integrate BLOBs within the database or store them in the file system? Each method has its advantages and disadvantages.

Storing unstructured data such as images, audio files, and executable files in the database with typical text and numeric data lets you keep all related information for a given database entity together. And this approach enables easy search and retrieval of the BLOB data; you simply query its related text information. However, storing unstructured data can dramatically increase the size of your databases.

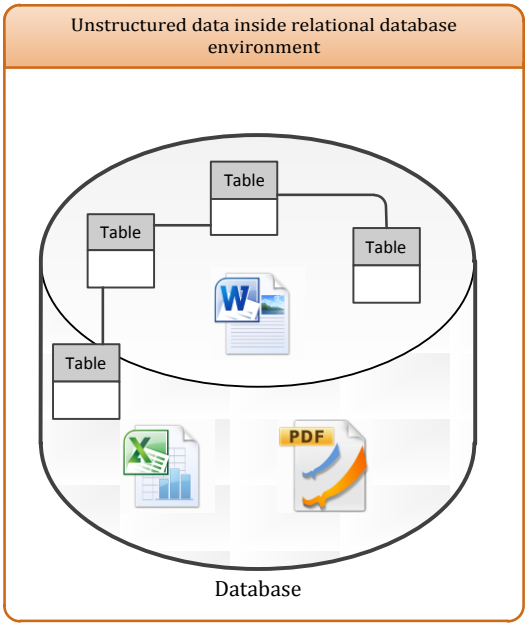

Figure 1. Database enviroment with unstructured data inside

TABLE I. Example of Real Life System

	Database characteristics				
	Database purpose	***Size with unstructured data***	***Number of users***	***Number of BLOB records***	***Backup time***
1.	LMS/CMS	≈ 12 GB	≈ 5000 students	≈ 7000 Files between 100 Kb-15Mb	25 min.

Main advantages of using this method:

- When the data is stored to the database, the backup is consistent. There is no need for separate backup policy and creating incontinency;
- When the data is stored inside the database, it's part of the transaction. So, for example, a rollback includes all traditional database operations along with binary data operations. This usually makes the client solution more robust with less code.

Main disadvantages of using this method

- storing unstructured data can dramatically increase the size of databases;
- backup and restore time can take a long time;
- performance issues with I/O subsystems

B. Unstructured data outside relational environment

The common alternative to this technique is storing binary files outside the database, then including as data in the database a file path or URL to the object.

This separate storage method has a couple of advantages over integrating BLOB data within the database. It's somewhat faster because reading data from the file system involves a bit less overhead than reading data from a database. And without the BLOBs, databases tend to be smaller.

However, we need to manually create and maintain a link between the database and external file system files, which have the potential to get out of sync. In addition, we usually need a unique naming or storage scheme for the OS files to clearly identify the potentially hundreds or even thousands of BLOB files.

Storing BLOB data within the database eliminates these problems by letting you store BLOB data along with its related relational data.

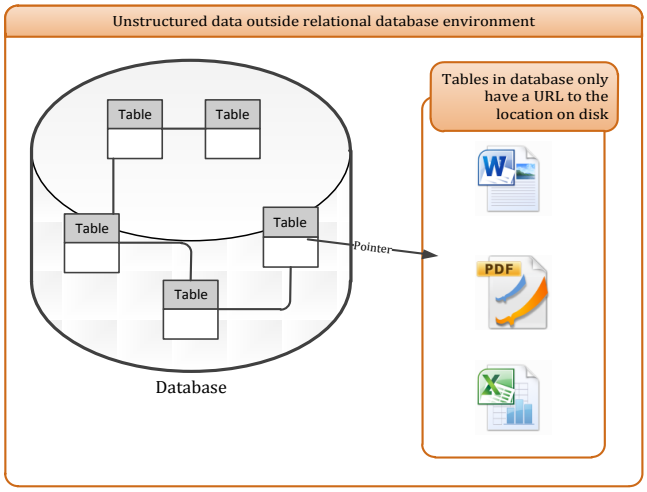

Figure 2. Database enviroment without unstructured data inside

In Table II, we have same database environment like in Table I. Only difference is that we are eliminating couple of tables that hold unstructured data inside physical database file. Difference is more than obviously, database size with same number of users, is half of initial. Let look more closely benefits of this method.

TABLE II. EXAMPLE OF REAL LIFE SYSTEM

	Database characteristics				
	Database purpose	***Size without unstructured data***	***Number of users***	***Number of BLOB records***	***Backup time***
1.	LMS/CMS	≈ 6 GB	≈ 5000 students	0	11 min.

Main disadvantages of using this method:

- When the data is stored outside of the database, the backup is not consistent;
- Unstructured data is not part of the transaction

Main advantages of using this method

- storing unstructured data outside can decries the size of databases and reduce I/O throughput
- backup and restore time can take a less time;

C. Hybrid way of storing unstructured data

Problem with first two methods is that we don't know how actual size of data affects database performance. Results, so far, don't tell us is there any difference in storing BLOBS: 10kb, 1 MB, 100 MB etc.

Major database vendors now support a hybrid way of storing unstructured data. In this case data is "outside" of database environment. But, major advantage is that BLOB's are under database transactional consistency.

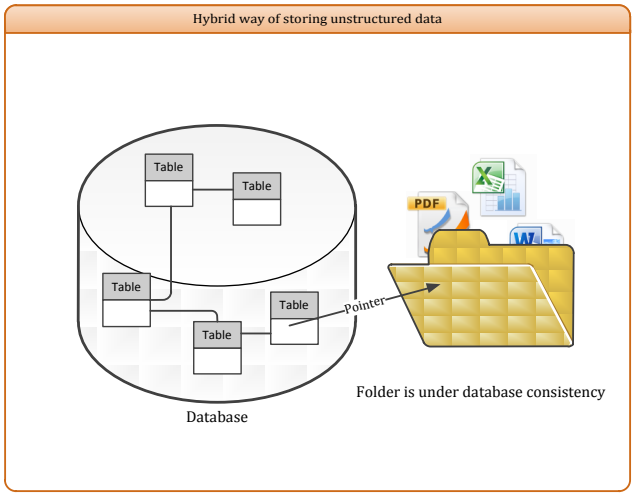

Figure 3. Hybrid way of storing unstructured data

In our case we are decide to use now data types which are offered in new version SQL Server 2008. Using this new technology is actually making hybrid method of storing unstructured data possible. Database engine supports new data type with name filestream

Filestram support combines the benefit of accessing BLOBs directly from the NTFS file system with the

referential integrity and ease of access offered by the traditional relational database engine. In SQL Server, BLOBs can be standard varbinary (max) data that stores the data in tables, or filesteram varbinary (max) objects that store the data in the file system. This research will show that the size and use of the data determines whether you should use database storage or file system storage. Filestream storage is implemented as a varbinary (max) column in which the data is stored as BLOBs in the file system. The sizes of the BLOBs are limited only by the volume size of the file system. The standard varbinary (max) limitation of 2-GB file sizes does not apply to BLOBs that are stored in the file system.

Key aspect of our research is to see is this method of storing data efficient.

Experimental environment

For this purpose we setup testing environment with following components:

- Processor: AMD X2 3.0 Ghz
- 4 GB physical memory
- Database files on C:
- Filestream component on drive E:
- Drives C: and E: on separate physical SATA disk drives
- SQL Server 2008 R2 and custom testing client application are on the same machine

The following charts show average upload times for:

- 100 KB file repeated 3 times for each measurement
- 1 MB file repeated 3 times for each measurement
- 10 MB file repeated 3 times for each measurement

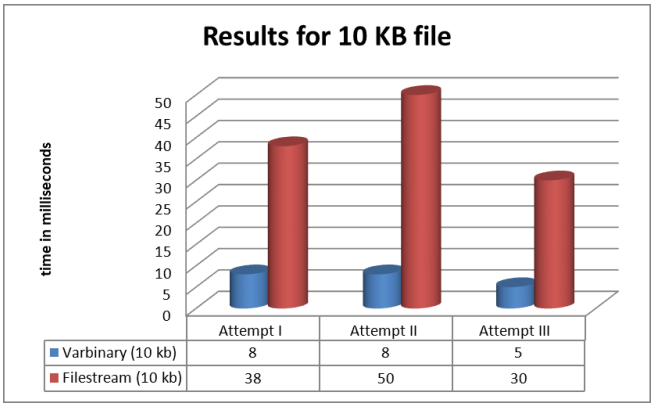

Figure 4. Results of storing 10 KB file inside database and filesteram

In this measurement, you can clearly see the overhead and impact on performance caused by using filestream on "*small*" files. Time, in milliseconds, when storing inside database was every time less than 10 ms. On the other hand time for storing using filestream is 4-5 time longer.

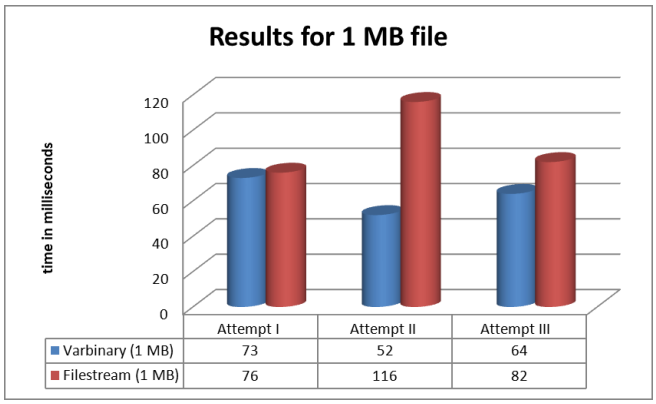

Figure 5. Results of storing 1 MB file inside database and filesteram

With 1 MB of data, the traditional varbinary and the file stream are acting quite similarly and difference between them is max. one time faster.

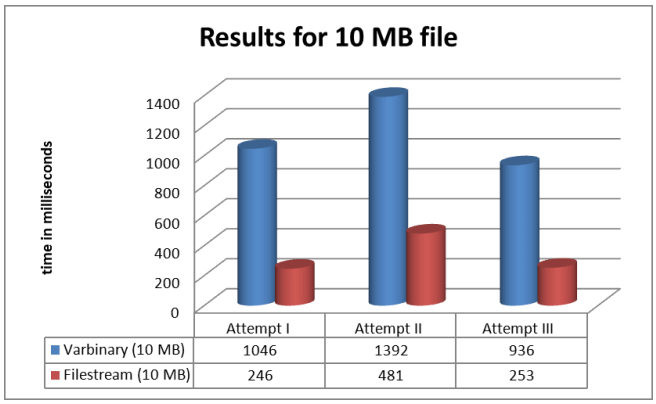

Figure 6. Results of storing 10 MB file inside database and filesteram

When 10 MB files are used, storing the data inside a traditional database file is much slower. Based on these measurements, it's more efficient to use a file stream when the typical file size is about 1 MB or more. If files are small in size (clearly under 1 MB), a traditional storage performs better. When measuring the times, we found that the deletion of the filestream based rows is much, faster than when stored inside the table.

IV. CONCLUSION

We start this research with hypothesis that current method of storing and retrieving unstructured data are not efficient.

New way of using file system under database consistency was good opportunity to test new technology in different areas of storing data. Test examples: 10 KB, 1 MB and 10 MB in real information system environments can be: photos

of users, CV files, small office documents, video and audio files. Based on content our research can be used to model system and clearly notice data storage needs. Benefit is getting maximum performance based on hardware and software infrastructure. Also systems where performance issues already exist, this model of can help to identify bottlenecks and find a way to improve it.

Results of our research produce model for testing and benchmarking system for storing unstructured data. Future steps for improving is implementing this model in real environments where unstructured data are essential (eLearning platforms, social networking, video storage portals etc.). After that this research can become official methodology for analyzing system with unstructured data needs. Current trends [3] show us that www become global storage for all kinds of data, where unstructured is dominated.

V. REFERENCES

[1] Hitachi Global Storage Technologies – HDDT echnology Overviewharts.http://www.hitachigst.com/hdd/technolo/overview/storagetechchart.html.

[2] A Benchmark Suite for Unstructured Data Processing Smullen, C.W.; Tarapore, S.R.; Gurumurthi, S.; Storage Network Architecture and Parallel I/Os, 2007. SNAPI.International Workshop on 24-24 Sept. 2007 Page(s):79 – 83

[3] J.Gantzandetal. The Expanding Digital Universe – A Forecaset of World wide Information Growth Through 2010, March 2007. IDC Whitepaper

[4] C.White. Consolidating, Accessing, and Analyzing Unstructured Data, December2005. Business Intelligence Network article.

[5] R.Chamberlain, M.Franklin, and R.Indeck. Exploiting Reconfigurability for Text Search. In Proceedings of the Workshop on High Performance Embedded Computing(HPEC),September2006.

[6] Improving Data Accessibility with File Area Networks Geer, D.; Computer

[7] Volume 40, Issue 11, Nov. 2007 Page(s):14 - 17 Digital Object Identifier 10.1109/MC.2007.393

[8] A Framework for the Classification of Unstructured Data Ostrowski, D.A.; Semantic Computing, 2009. ICSC '09. IEEE International Conference on 14-16 Sept. 2009 Page(s):373 - 377 Digital Object Identifier 10.1109/ICSC.2009.48

[9] Transforming unstructured data from scattered sources into knowledge Plejic, B.; Vujnovic, B.; Penco, R.; Knowledge Acquisition and Modeling Workshop, 2008. KAM Workshop 2008. IEEE International Symposium on 21-22 Dec. 2008 Page(s):924 - 927 Digital Object Identifier 10.1109/KAMW.2008.4810643

[10] Document warehousing based on a multimedia database system Ishikawa, H.; Kubota, K.; Noguchi, Y.; Kato, K.; Ono, M.; Yoshizawa, N.; Kanemasa, Y.; Data Engineering, 1999. Proceedings., 15th International Conference on 23-26 March 1999 Page(s):168 - 173 Digital Object Identifier 10.1109/ICDE.1999.754921

[11] G.Fountainand S.Drager. Highperformance real - time fusion architecture. In Proceedings of the Fifth International Conference on Information Fusion, pages 1478–1485,2002

[12] R.Narayanan, B.Ozisikyilmaz, J.Zambreno, G.Memik,and A.Choudhary.Mine Bench:A Benchmark Suite for Data Mining Workloads .In IEEE International Symposium on Workload Characterization(IISWC), pages 182–188,October 2006.

[13] E.Riedel, G.Gibson, and C.Faloutsos. Active Storage for Large-Scale Data Mining and Multimedia. In Proceedings of the International Conference on Very Large DataBases (VLDB), pages 62–73, August 1998.

[14] C.White. Consolidating, Accessing, and Analyzing Unstructured Data,December2005. Business IntelligenceNet workarticle.