Study the Time of Access to Data in Heterogeneous Databases.

Ognyan Nakov, Daniela Gotseva, Ivan Stankov
Technical University, Sofia, Bulgaria, istankov@tu-sofia.bg

***Abstract**: This article assesses the performance of two of the most popular databases - ORACLE and MS SQL, with and without indexing data. After a brief overview of the opportunities, which afforded us to compare the two databases and to create a heterogeneous database. The main objective is to examine the execution time of queries when searching for information in various databases and to decide how the times can be improved.*

***Keywords:** MS SQL, Oracle, performance, comparison, improvement.*

1. INTRODUCTION

With computerization in almost all spheres of life, information that is necessary to keep acquiring huge amounts of data bases, where they are stored , becomes enormously large. People face problems, related to:1. the access time, i.e. how to optimize time for access to the data; 2. the huge volume of data ,stored in the more commonly used heterogeneous databases.
Working with database provides savings in using personal computers, reduces errors by centralizing data and increased responsibilities to managerial needs. Data can be divided into parts, based on regional product or other principle, when they are mainly used on site. Another way is duplication of data in any location. Thus, running the risk of inaccuracy and inconsistency of information on individual locations, is refreshed. For the first method and the constant need to monitor data or to maintain a centralized index.
Separating parts of data in remote locations requires appropriate methods to locate relevant data and to guarantee continuous remote access. To not waste time, searching for data and location of storage, the database needs to maintain a centralized index ,that maintains location information data. In this case, search request is sent to a central server location of the necessary data [1].
In this article, we will examine the requests to two databases: Oracle Database 10g and MS Sql 2005. The results from the research will be analyzed on the basis of the results and suggested improvements wil be proposed.
Microsoft ® SQL Server ™ 2005 is the next generation platform for managing and analyzing data, which offers enhanced security, opportunity for expansion and accessibility to data and applications for analysis of the right place at the desired time. In addition, SQL Server 2005 makes it easy to create, deploy and manage applications. SQL Server 2005 is developed on the sound basis of SQL Server 2000, which offers an integrated solution for managing and analyzing data. It helps the following features:
• Build and deploy applications that are more secure, more reliable and also greater opportunities for scalability;
• Maximize application performance by reducing complexity in the construction, installation and management of databases;
• Assists developers with a rich, flexible and modern environment for developing more secure applications;
• Transmits data between multiple platforms, applications and devices;
• Provides powerful, integrated information resources to help the developer in making informed business decisions and increases productivity throughout the organization;
• Controls the costs without sacrificing performance, reliability and scalability [2].

In Oracle Procedure is contained the language PL / SQL which is used most often for the realization of logic modules for different applications. PL / SQL can be used in the development of stored procedures and triggers control elements for loops (looping controls), conditional statements and processing errors. One can compile and store PL / SQL-procedures in the database and also run the PL / SQL-Blocks in SQL * Plus, an interactive tool which controls if the store procedure is present in all versions of Oracle. Optimized Oracle Database 1Og contains a kernel of PL / SQL, and like Ogas1e9 / allows the creation and storage of precompiled PL / SQL-program fragments [3].

2. FILE SAVING

Initially, we will gradually fill in the created database with different range of records to make certain orders and to take account of the relevant performance. The results are presented in graphic form through diagrams.A research will be carried out for different numbers of records, such as a number of rabbits. It will be submitted respectively to 10 000, 20 000, 30 000, 40 000, 50 000, 60 000 , 80 000 and 100 000 records.The diagrams, created during the corresponding time, depict execution time of applications to both MS SQL, and ORACLE databases. The databases consist of three different record types: images, video and text. With the completion of these studies , the first part of the study will be completed. The second part is aimed at improving productivity and performance. The databases will be indexed and will be tested again with the same productivity applications. Finally, we will analyze the results and propose conclusion.[4].

2.1. Test computer configuration

Notebook Asus X51r
Processor: Genuine Intel (R) CPU T2130@1.86 GHz 1mb cash memory, RAM: 2GB RAM DDR2 @ 677 MHz, OS: Windows 7 Ultimate, type of operating system: 32 bit.

2.2. Proportion of records in the database

Records:text/image/video.
10000-7000/2500/1000
20000-15000/5000/2000
30000-23000/7000/3000
40000-30000/10000/3500
50000-37000/12000/4400
60000-45000/15000/5500
80000-65000/19000/7000
100 000 -80000/23000/9800

2.3. Requests

Request 1
This query should return a result containing all the names of students from the student table, whose names are associated with a video or image file.
Request 2
This query will display all the names of students who have an image , associated with ID in the range from 870115000 to 890237000.
Request 3
Request 3 will need to bring all students who have an associated only a video or video and image.

2.4. Results from the report

Testing of MS SQL

The results are shown in graphs, time is converted into seconds. Before the introduction of specific request , it has to contain the following features specific to MS SQL studio-it: SET STATISTICS TIME ON, this function returns the exact execution time of the application in Milliseconds, for objective reporting milliseconds are converted into seconds.

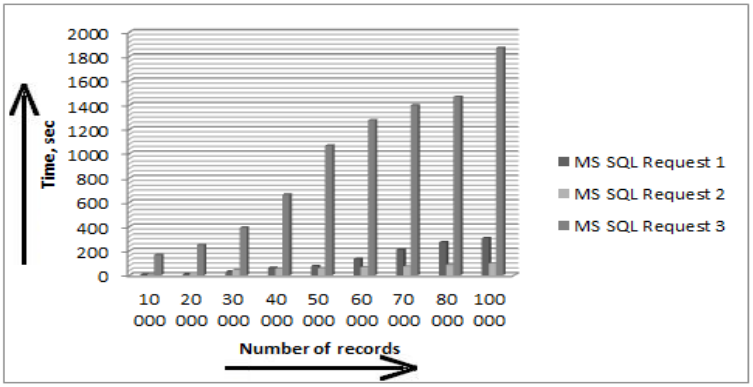

Fig. 1: Comparison of three testing times , without SQL query data to be indexed

Tests of Oracle

The results are shown in graphs and times are in seconds.

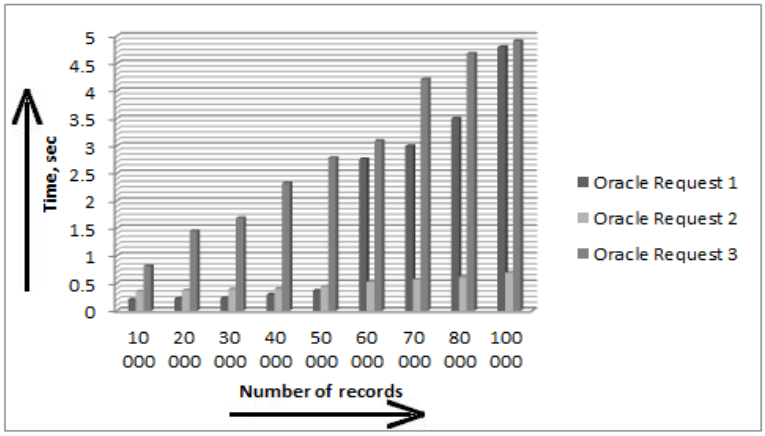

Fig. 2: Time Access to Oracle database without indexing

After tests and reported results, the records are indexed and we explore new performance applications.
Tests with indexing :

There are two main issues in indexes: Why is it necessary to create indexes?
The main reasons to create indexes are the following:

•Accelerated time to access data;
•An unique, only one row
Why should not create indexes?
•More memory is spent;
•Possible tolerated losses [4]
Usage of the Index.

Indexing databases and gradually filled in up to 10 000 records, after that it is filled in again to 20 000 records and after each new cycle, the perious indexes are updated.

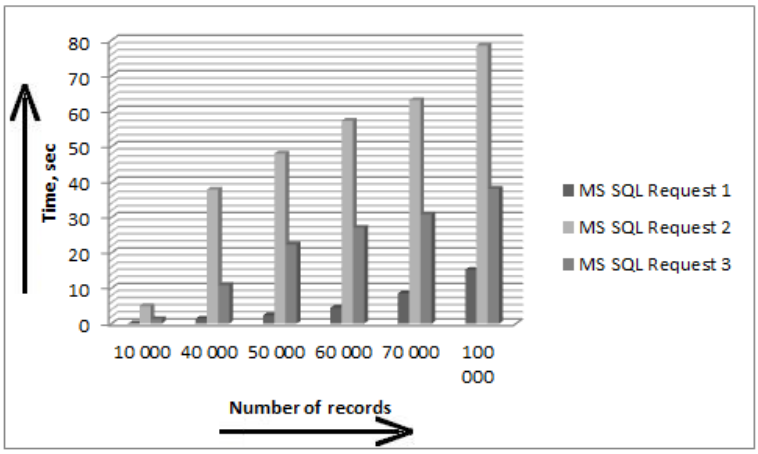

Fig. 3: Time access to MS SQL database with index

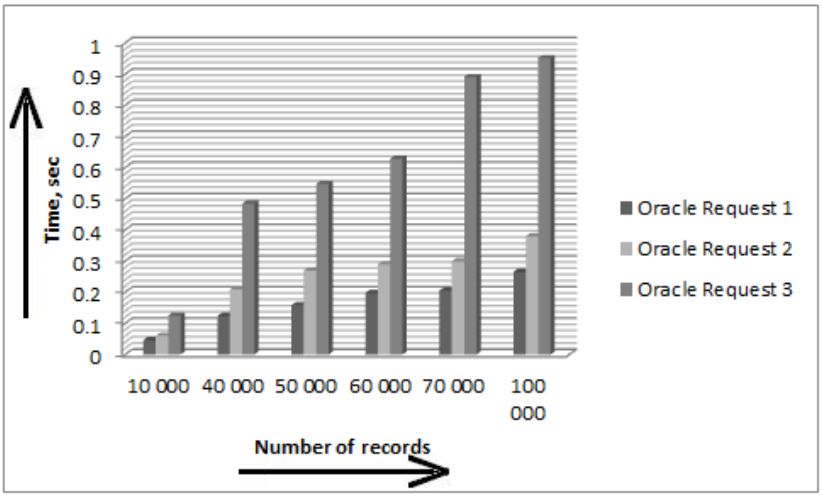

Fig. 4 Time access to Oracle database with index

3. ANALYSIS OF RESULTS

The research very clearly shows that the usual search time for application in the Select MS SQL database and the execution time when Oracle is used, are extremely large (MS SQL fig. 1 and Oracle fig. 2).
In MS SQL database request 1 shows the names of all students who have video or pictures associated. Request 2 will bring all students , whose image is associated with ID in the range 870,115,000 to 890,237,000, and request 3 is for students with associated video file or video file and image. On request 3 is observed relatively constant behaviour of the curve. In the first 2 queries the curve behavoiur is different: first, to 30 000th record the time for query 1 is less than the time for query 2, then the time of application 1 increases rapidly, and after recording 50 000th there is a sharp increase.
In Oracle, the behaviour of query 2 is a very slight gradual increase. The same behaviour is observed in query 2 to 50,000th record , followed by sharp increases of around the same times. On request 3 , a significant increase was observed in 80 000th record.
After entering the index, studies were made in the number of records where without index more interesting results have been observed (in 10000, 40000, 50000, 60000, 70000 and 100 000 records)

For MS SQL there is a reduction in multiple times, as they continue to run on the same exponent as if no index is applied(requests 1 and 3). In request 2, minimum observed reduction in time is registered, and indexing values before and after the index are very close.
In Oracle, the access time also experienced a decrease, but not in such large amounts as in MS SQL. The interesting fact is that after indexing in query time 3, there was no such sharp increase in time at 60 000th record, and after that the time continues to grow slowly. At 70 000 and 100 000 records, there is not very big change in the timeand we can speak about a kind of saturation.

4. CONCLUSION

The issues with the productivity of the database requests have not deepen yet, but in the future, this will be one of the main requirements of developers and IT professionals. After the attempted performance of the applications, we can say that Oracle is doing much better than MS SQL within the trials we have done.The performance of applications is improved by using an indexed database. This improvement has been studied and registered in both databases, but is particularly valid for MS SQL.

5. REFERENCES

[1] Hernandez, M. H., *Design of databases*, SoftPres.
[2] MSDN. *Microsoft SQL Server 2005.* Microsoft library: http://msdn.microsoft.com/en-us/sqlserver/aa336270
[3] Greenwald, R., Fundamentals of Oracle "Oracle Database 10g".
[4] JAN PANUŠ J. &, PIRKL. J. (2006) Testing of Oracle database utilization (vol 3).