УДК 004.65

A. I. Brakovich, A. Hassan
Belarussian State Technological University

THE CLASSIFICATION AND BRIEF ANALYSIS OF EXISTING DEVELOPMENTS FOR THE SEARCH OPTIMIZATION IN DATABASES

The article is devoted to the classification and brief analysis of existing developments, which can be used to optimize the search database in the cloud. The growing demand for service providers offering a broad range of cloud computing services for large numbers of users all over the world, leading to increase in the number of applications, the purpose of which is to process large data sets. The operation of the database in the cloud leads to the need to find new search instruments. The present level of development of information technology makes real-time information resources available. In the article the description of query optimization in the cloud SQL type database, NoSQL type and architecture-specific solutions are given. For all methods advantages and disadvantages are presented. Common to all methods is the lack of synthetic nature of the results, that is, that the introduction of statistics obtained in artificial systems created just for testing approach. However, relatively to large number of studies for a fairly young field suggests that the problem is urgent and the optimization of such developments in the near future will be in demand.

Key words: database, information technologies, optimization, cloud computing, information resources, forms and methods.

Introduction. The present level of development of information technology makes available real-time information resources of different volume and content. To facilitate the handling of large volumes of information are developed a variety of forms and methods, of its presentation, as well as search techniques, which is expressed, for example, in the creation of proprietary standards and systems, individually configurable by the user.

Widely used the notion of "information system" has virtually no single conceptual definition. Most often this concept is treated as a "complex of information collection and procedures: management, updates, information retrieval and post-processing – which allows to accumulate, store, update and provide information".

Such user-utilitarian definition of information systems is associated with a well-established and already familiar, but, nevertheless, a special form of purposeful human activity – processing of information as the information about something material presented in the form of traditional paper documents or computer readable media. That is, the "system" reflects the essence of the functional relationship: the composition and structure of the information system is determined based on the requirements for the efficiency of the service the information needs of end-users, especially in terms of being in accumulated arrays of those documents, which are suspected to contain necessary information.

However, the crucial factor in determining the direction of developing modern information systems, is that the user interaction with information resources takes place in the informational self-service, when a user is substantially no longer divides his activities on information and basic one. It is especially important in the processes of information support of scientific research, where the search object can not be clearly defined in advance, and when the originally defined search target may change in the course of the search. For example, when reading the documents found, and the fact of changing the target may not be comprehended researcher clearly, that in the end can lead to an incomplete search results.

In addition, using information resources, along with original author's presentation of the material in the majority are characterized by high systematic-dosing as well as almost obligatory presence of background information. Also it should be noted that the search tools and technologies used for the implementation of information requirements, depend on the type and condition of the problem to be solved by the user operations: the relation of his knowledge and ignorance about the object.

The functioning of modern information retrieval system is based on two assumptions:

– documents needed by a user, are united by the presence of some feature or combination of features;

– a user is able to specify this feature.

Both of these assumptions are not fulfilled in practice, and we can only talk about the likelihood of their implementation. Therefore, the information search process is typically a sequence of steps leading to the system by means of some results, and its completeness. In this case user's behavior as organizing principle of management search process, motivated by the need of not only information, but also of a variety of strategies, technologies and tools provided by the system. Concepts such as strategy and search technology, tools, and methods, models and algorithms are enough to consume, but different authors use these terms in different contexts [1].

Main part. For solving the problem of query optimization in the cloud storage system should be taken into account network topology.

Query optimization in the cloud SQL type database. Request handling is reduced to transform the high-level inquiry into the equivalent of a low-level form, and the main difficulty in this case is to ensure the efficiency of the conversion-specific cloud storage. Standard SQL statements use the connection, a selection, projections, groups (group-by). The key principles of this architecture are as follows:

– all files are stored in the local file system (e. g., file system Windows, Linux, etc.);

– cloud database is designed to store and manage huge amounts of index files and metadata. It should be noted that the cloud database and all its contents is deployed on top of a distributed file system;

– enter the query and get the results performed by the web user interface;

– upon receipt of a user request is executed the current semantic search query plan in the global scheme (as a subset).

The results of the pilot implementation of the architecture show four times performance increase, indicating the efficiency of the algorithms.

Query optimization in cloud databases NoSQL type. Programming model map-reduce (MR) is a popular platform for cloud computing, which allows analysis of large amounts of data in the cloud. MR facilitates parallel execution of special, long-term problems of the analysis of large data volumes in a cluster with a shared-nothing architecture. The basic idea of MR model is simple. Each task is represented as a map and reduce tasks. Target map indicating how to be processed key or value pairs to generate a set of intermediate pairs, whereas reference reduce determines how to combine all intermediate values associated with one intermediate. MR Kernel for storage and data replication uses a distributed file system.

At the core of this approach lies the use of algebra queries and application of some higher order operators that are implemented in the existing map-reduce systems (for example, Hadoop). It should be noted that the proposed approach is primarily targeted for use with MRQL language. Unlike other map-reduce existing languages such as HiveQL and PigLatin, which allow to create scripts using non-declarative languages, MRQL expressive enough and allows a user to write own scripts for a large range of tasks in declarative form, and at the same time lends itself to optimization.

As is the case with relational databases, to optimize MRQL query is to find an optimal execution plan. Performance evaluation plan algorithm MRQL requests consists of the following steps:

– simplification of the request;

– building a query graph;

– the query graph representation in algebraic form;

– formation of algebraic form card for assessment and improvement plan using the algebraic method optimization;

– create a function of MR combination basing on MR reduce function.

The advantage of this approach is that the developed algorithms are implemented as a framework, the source code that is freely available. This project is currently being actively developed [2].

Another example is genetic algorithm. It is a search procedure inspired by principles from natural selection and genetics. It is often used as an optimization method to solve problems where little is known about the objective function. The operation of the genetic algorithm is quite simple. It starts with a population of random individuals, each corresponding to a particular candidate solution to the problem to be solved. Then, the best individuals survive, mate, and create offspring, originating a new population of individuals. This process is repeated a number of times, and typically leads to better and better individuals.

Generic algorithm theory is centered around the notion of a building block. The study will be talking about deception, population sizing studies, the role of parameters and operators, building block mixing, and linkage learning. These studies are motivated by the desire of building better Generic algorithms. Algorithms that can solve difficult problems quickly, accurately, and reliably. It is therefore a theory that is guided by practical matters.

Genetic algorithm operation describes the operation of a simple genetic algorithm. The exposition uses a step-oriented style and is written from an application perspective. The steps of applying a genetic algorithm are:

– to choose an encoding;

– choose a fitness function;

– choose operators;

– choose parameters;

– choose initialization method and stopping criteria.

Large amount of information serves as a huge information repository for organizations. However, it also makes finding relevant information from it extremely difficult. How to help users find their required information is the central task of any information retrieval system or search engine. However, precision and recall, the two most commonly used performance measures, of commonly used search engines are usually very low.

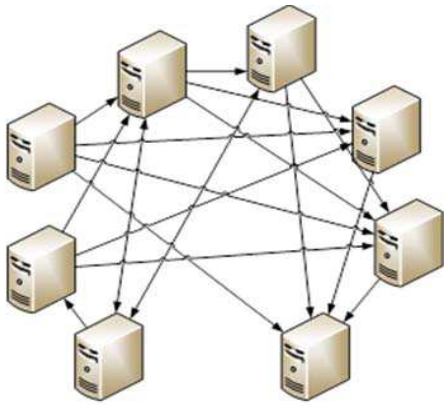
Retrieval performance of an information retrieval system can be affected by many factors: the ambiguity of query terms, unfamiliarity with system features, as well as factors relating to document representation. Many approaches have

been proposed to address these issues. For example, query expansion techniques based on a user's relevance feedback have been used to discover a user's real information need. Similarly, document descriptions have been modified. Another very important factor is the ranking/matching function. It is this ranking function that to focus most of the discussion on. A ranking function is used to order documents in terms of their predicted relevance to a particular query. It is very difficult to design such a ranking function that can be successful for every query, user, or document collection (which we will call contexts). In this search, it is argue in favor of a method that systematically adapts a ranking function and tailors it to different users' needs (i. e. in different contexts). In particular, an inductive learning technique, for the adaptation purpose and compare our results against two well-known retrieval systems.

Fuzzy theory, as a framework describing formally the concepts of vagueness, imprecision, uncertainty and inconsistency provide interesting extensions to the area of information retrieval. Imprecision and vagueness are present in natural language and take part in real-world human communication. User friendly and flexible advanced information retrieval system should be able to offer user interface for non experienced users allowing natural deployment of these concepts in user system interaction for more effective information retrieval. Information retrieval models exploiting fuzzy techniques can overcome some of the limitations pointed out in first part of this article. They support different grades of document-query relevance, cut inaccuracies and oversimplifications happening during document indexing and introduce the concepts of vagueness and imprecision in query language.

Architecture-specific solutions. Decentralized management of migrating virtual machines in large-scale cloud-based processing centers. The main interest of this approach lies in the fact that its purpose is to balance the load on equipment through the migration of virtual machines in the cloud environment, which indirectly leads to an increase in the quality of the search (it is obvious that the speed with which the cloud responds to user requests, is one of the search quality criteria).

It often happens so, that for resource management in large-scale data centers are developed and implemented a centralized solution, but in this case the occurrence of a failure at the control node, resulting in malfunction of the whole system. As shown in Figure, each active node in the process of functioning selectively at a predetermined interval sends its own index of congestion in some nodes of the system, at the same time to get the index of congestion random active nodes.



Decentralized exchange of index congestion

This change target nodes at each iteration. Information on the utilization of other components added to the vector of current workload. Thus, the average length of the vector unit load is equal to the number of iterations of sending the index. A load information will be stored in a decentralized, in order to avoid trouble in the event of a failure of the node, another positive aspect is that the network traffic is distributed across all active nodes (as opposed to the scheme with centralized management, where all packets should go through common node).

Unit load index is a tuple of the form:

$$LI = <src, dest, util>,$$

where *src* – node identifier, when it has received index; *dest* – contains a node ID that will receive the index, *util* – CPU usage source node (src).

Conclusion. As virtual machines are host to deploy a variety of applications with different workloads on the CPU, then eventually the physical CPU utilization can vary. The decision on the virtual machine migration can be taken in two cases:

– when the CPU usage exceeds a certain level (the upper limit). The purpose of establishing an upper threshold is to keep the additional computing power in case of situations with a sharp (unplanned) increasing load;

– when the CPU usage is below a certain level (the lower limit) – unit underutilized. The purpose of establishing a lower threshold is to possibly increasing the number of physical units that transferred to the "sleep" mode, thus reducing power consumption.

After the decision to migrate a virtual machine, start searching the destination node. To do this a user has to crawl load vector of the current node to detect the node with lowest amount of CPU provided in contact with predetermined intervals. If a node can not detect this unit searches for the index which load when transferring to it the selected virtual machine does not exceed the lower utilization of the border. If in this case no search results, one

of the nodes are in “sleep” mode is transferred to the active state and migrating.

In the article is given the description of query optimization in the cloud SQL type database, NoSQL type and architecture-specific solutions. Each of mentioned methods has both advantages and disadvantages. Common to all methods is the lack of synthetic nature of the results, that is, that the introduction of statistics obtained in artificial systems created just for testing approach. However, the relatively large number of studies for a fairly young field suggests that the problem is urgent and optimization such developments in the near future will be in demand.

References

1. Vakkari P., Hakala N. An Appropriate Boolean Query Reformulation Interface for Information Retrieval Based on Adaptive Generalization. *In WIRI*, 2015, pp. 145–150.

2. Ullman Jeffrey D., Hector Garcia–Molina, Widom Jennifer. Database Systems: the complete book. 2nd ed. USA, Pearson, 2013. P. 218–256.

Information about the authors

Brakovich Andrei Igorevich – PhD (Engineering), Associate Professor, Assistant Professor, the Department of Information Systems and Technologies. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: brakovich@yandex.ru

Hassan Ali – PhD student. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: teacher.ali.h@hotmail.com

Received 26.04.2017