ANALYSIS OF THE FEATURES OF THE OPTIMAL LOGICAL STRUCTURE OF DISTRIBUTED DATABASES

E.V. Nurmatova [1,a], V.V. Gusev [2], V.V. Kotliar [2]

[1] *«MIREA — Russian Technological University», Moscow*

[2] *NRC "Kurchatov Institute" — IHEP*

E-mail: [a] nurmatova@mirea.ru

The questions of constructing optimal logical structure of a distributed database (DDB) are considered. Solving these issues will make it possible to increase the speed of processing requests in DDB in comparison with a traditional database. Optimal logical structure of DDB will ensure the efficiency of the information system on computational resources. The problem of constructing an optimal logical structure of DDB is reduced to the problem of quadratic integer programming. As a result of its solution, the local network of the DDB is decomposed into a number of clusters that have minimal information connectivity with each other. In particular, such tasks arise for the organization of systems for processing huge amounts of information from the Large Hadron Collider. In these systems various DDBs are used to store information about:1) the system of triggers of data collection from physical experimental installations (ATLAS, CMS, LHCb, Alice), 2) the geometry and the operating conditions of the detector while collecting experimental data.

Keywords: optimal logical structure, query implementation scheme, distributed databases, grid architecture.

1. Introduction

Optimal logical structure of DDB will ensure the efficiency of the information system on computational resources.Such tasks arise in various areas of information and communication technologies, where distributed systems, systems with high loads, or systems of increased reliability are already used. In particular, such tasks arise for the organization of systems for processing huge amounts of information from the Large Hadron Collider (LHC). In these systems various DDBs are used to store information about:

- the system of triggers for collecting data from physical experimental installations (ATLAS, CMS, LHCb, Alice),
- the geometry of the detector while collecting experimental data,
- description of the metadata of the events on the detectors (for example, maintenance work on the installation),
- the operating conditions of the detector while collecting experimental data [1].

The last DDB, the Conditions database, is the most complex in its organization. For access to it a special program interface CORAL [2] was written. The use of DDB directly in the process of data processing of the LHC in the distributed computing environment Grid should also be noted. To optimize the transfer of the results of calculations of individual tasks over computer networks, a multi-level hierarchical system is used. This system keeps track of where data is stored and where it is necessary to transfer them [3].

2. Synthesis of the optimal logical structure of a distributed database

DDB construction technologies can be used for storing data in monitoring systems of complex computing infrastructures, which are a combination of engineering, network, software environments[4].

The constant search for methods to reduce the costs of data storage and their processing served as an objective need to deepen research on these issues and determined the relevance of the topic of this work.

Synthesis of the optimal logical structure of a distributed database is the process of finding the optimal mapping of the canonical structure of the DDB to a logical one, which provides the optimal value of a given criterion for the performance of information systems and satisfies the basic system, network, and structural constraints [5]. When mapping a canonical structure into a logical one the groups of data of the canonical structure of the DDB are combined into types of logical records with their simultaneous distribution among the nodes of the computing system.

The logical structure of DDB will be understood as an ordered set of logical records and connections (relations) between them, distributed over the nodes of the computing system. These logical records and connections reflect the semantic and functional properties and features of a given subject area of the information system.

Let the logical structure of the database be given by the graph $G(N, L)$. To implement the q-th query on the graph $G(N, L)$, it is necessary to select the search tree $G_q(N_q, L_q)$, where $N_q \subset N$ is a subset of vertices, and $L_q \subset L$ is the subset of connections of the graph G chosen by the search tree G_q. It should be noted that one of the vertices N_q^0 is the entry point to the logical structure of the database.

Figure 1 shows an example of a search tree G_q and the direction of its bypass.

Reducing the response time to a query can be achieved by eliminating the vertices to which it is necessary to return for passing to the next branch. The graph G_u thus obtained will contain only those vertices (records) whose elements form the output structures of the query q. Thus, the task of minimizing the response time to a query of the q-th type is reduced to the problem of choosing the optimal connection between the records of the graph G_u.

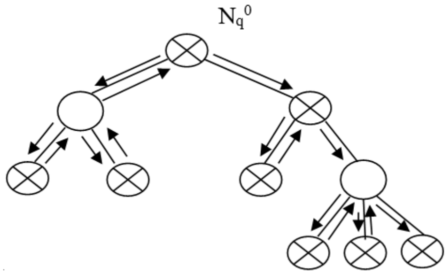

Figure 1. The scheme of the graph $G_q(N_q, L_q)$ and of direction of its bypass

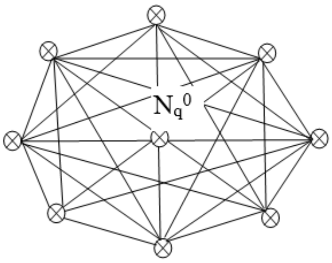

Figure 2. The scheme of a complete graph G_u and connections between all vertices of a subset N_q^u

The scheme of a complete graph (see Figure 2) can be used to pose the problem of choosing the optimal structure of connections between records used to form the output structures of the q-th query with restrictions on the possibility of using a certain access path.

The created DDBs can have a large dimension and therefore they are loaded and introduced in parts. To this end, the LS of DDB should be divided into a number of substructures or clusters that have minimal mutual connectivity under the following restrictions:

- on the dimension of clusters,
- on types of storage devices used,
- on the degree of semantic proximity of logical records included in clusters, etc.

The source for the formulation and solution of this problem is the information obtained as a result of the synthesis of the LS of DDB (see Table 1).

Table 1. The notation for source variables.

$N = \{n_j / j = \overline{1, J}\}$	the multiplicity of the record types of the LS of DB
$W = \left\| w_{jj'} \right\|$	the matrix of connections of records structure
$F = \left\| f_{jj'}^K \right\|$	the matrix of access paths for implementing queries of users
$R = \{r_1, \ldots r_j, \ldots, r_J\}$	the vector of a set of logical records
$\rho = \{\rho_1, \ldots \rho_j, \ldots, \rho_J\}$	the vector of characteristics of records length in bytes

To formalize the task, we introduce the variables $X_{je} = 1$, if the j-th record is included in the e-th cluster; $X_{je} = 0$ - otherwise.

The task of decomposition of the logical structure of the RDB into the multiplicity of clusters that provide minimal connectivity between them is formulated as follows:

$$\min_{\{x_{j,\varepsilon}\}} \sum_{\varepsilon,\varepsilon^{'}=1}^{\varepsilon_0} \sum_{j,j^{'}=1}^{J} \lambda_{jj^{'}} x_{j\varepsilon} x_{j^{'}\varepsilon^{'}},$$

with restrictions on the one-time inclusion of a logical record in the cluster,

$$\sum_{\varepsilon=1}^{\varepsilon_0} x_{j\varepsilon} = 1, j = \overline{1, J},$$

and with restrictions on the total number of logical records in a cluster

$$\sum_{j=1}^{j_0} x_{j\varepsilon \le M} = 1, \varepsilon = \overline{1, \varepsilon_0},$$

where M is an allowable number of records in the cluster.

This task is a problem of quadratic integer programming.

As a result of solving the posed problem, the LS of DDB is decomposed into a number of clusters, which have minimal information connectivity with each other.

The development of a database of a separate cluster can be further carried out taking into account the importance of the logical records included in it in terms of user requirements, development complexity, and other factors.

The solution of such a task is of great practical importance for computer-aided design of the LS of DDB and for generating specifications for queries and corrections of DDB.

This is especially important when organizing a DDB in the “client-server” architecture, in which the query language SQL is used as an interface. The query specifications for it include two main parts: query objects and search conditions. In the request objects the information elements and the logical records required by the user are listed. It should be noted that the manual construction of queries takes a long time, because the user requires a detailed knowledge of the composition and structure of logical records, the interconnections between them, the characteristics of the organization, etc.

3. The logic circuit implementation of the query

Queries in the DDB are characterized by the composition of the requested data, the frequency of their use, and the performance characteristics of the average values of the number of analyzed and selected when searching for instances records. The logic scheme of the implementation of a query of DDB user includes the following sequence of operations:

- Designing and initiating a request by the user in the node of the computing system to which it is attached, in the query language of the selected control system of DDB.
- Transferring the information to the database server via communication channels for the implementation of the query.
- Processing the query using the appropriate methods and by means of the database and solving the following main tasks: selecting from the database the data required in the query; decomposition of a query for subqueries (tasks), the number of which is determined by the number of required database servers; selecting optimal access ways to the required database servers; establishing logical connections with the nodes of the computing system on which the required database servers are located.
- Transferring subqueries to the required database servers via communication channels.
- Servicing subqueries with database servers.
- Transferring the blocks of selected data from the database servers to the server of the node that initiated the request via communication channels.
- Assembling blocks of the data into an array required in the query.

Evaluation criteria, as a tool of designing a DDB, are necessary for choosing a rational database structure among several alternative possibilities. It can be said that most of the problems and failures in developing a database model arise from a vague idea of what is meant by optimal database design. At present, as well as in the nearest future, uncertainty in the choice of these criteria will remain the weakest link in the development of a database model.

4. Time, cost and volume characteristics of DB functioning

Difficulties in determining the criteria for choosing alternative solutions are mainly caused by two factors [5]:

1. The first problem is that almost infinite number of different database structures that satisfy the same set of system requirements can be built. Selection criteria should allow differentiation of all currently available alternatives.
2. The second problem is that alternatives are difficult to evaluate, since the criteria have the property of sensitivity and the duration of the various criteria is different.

When solving the problem of the synthesis of the LS of DDB, the following basic time, cost and volume characteristics of the functioning of the DDB are used.

The main time characteristics are: the duration of the implementation of a given set of queries T^q and the duration of the implementation of a given set of transactions T^k.

In sum, these two indicators give the total duration of the “workload” of the DDB:

$$T = \sum_{p=1}^{p_0} T_p^q + \sum_{s=1}^{s_0} T_s^k,$$

where T_p^q - the duration of the implementation of the p-th user query; T_s^k - the duration of the implementation of the s-th transaction.

The components of the times T_p^q and T_s^k depend on many factors: the network and telecommunications equipment used, the technical parameters of the servers, the network and telecommunications system-wide software, the characteristics of the relational database management system (RDBMS), the bandwidth of the communication channels, the running times of programs at various levels of network protocols, etc.

The main cost characteristics of a DDB are:

— the cost of storing information in the DDB E_{xp};

— the query and transaction costs at a given time interval E_{ex}^q;

— the cost of transmitting information via communication channels E_{ex}^k.

The sum of these components determines the total cost of the functioning of the DDB:

$$E = E_{xp} + E_{ex}^q + E_{ex}^k.$$

The cost of storing information in the DDB is determined by the physical amount of information V_{xp} and the cost of storing a unit of information (one logical record) on the server k_{xp}. If we accept that the cost of storing information in all nodes of the computing system is a constant value, then

$$E_{xp} = V_{xp} \cdot k_{xp}.$$

In practice, k_{xp} approximately equals to 1.2 - 1.5.

And finally, the volumetric characteristics of the functioning of the DDB, the sets of user queries and transactions are calculated from the following assumption: the amount of memory occupied by one instance of the record equals to the sum of the amount of useful information and the product of length of one address reference and the number of pointers in the record.

5. Conclusion

The problem of constructing an optimal logical structure of DDB is reduced to the problem of quadratic integer programming. As a result of its solution, the local network of the DDB is decomposed into a number of clusters that have minimal information connectivity with each other.

Further research will be related to the design of an algorithm for the synthesis of the optimal logical structure of a distributed database. The algorithm will consist of two interrelated stages. At the first stage, it is supposed to solve the problem of distribution of clusters of a distributed database between a server and clients, then the solution of the problem of optimal distribution of data groups of each node to logical record types follows. At the second stage of the algorithm, the problem of localizing the database by computer network nodes is solved, and in addition to the results of the first stage, the characteristics of the distributed database are taken into account.

References

[1] A. V. Vaniachine LHC Databases on the Grid: Achievements and Open Issues — The Proceedings of the IV International Conference on "Distributed computing and Gridtechnologies in science and education" (Grid2010), JINR, Dubna, Russia, 28 June - 3 July, 2010.

[2] Radovan Chytracek, Dirk Düllmann, Giacomo Govi, Alexander Kalkhof, Zsolt Molnár, Andrea Valassi "Distributed Database Access in the LHC Computing Grid with CORAL" IEEE Nuclear Science Symposium conference record. Nuclear Science Symposium, 2008.

[3] D. Ciangottini, J. Balcas, M. Mascheroni, E. A. Rupeika, E. Vaandering, H. Riahi, J. M. D. Silva, J. M. Hernandez, S. Belforte, T. T. Ivanov "A comparison of different database technologies for the CMS AsyncStageOut transfer database" J. Phys.: Conf. Ser., 898, 042048, 2017.

[4] V. Kotliar, V. Anshukov, V. Ezhovac, V. Gusev, A. Kotliar, G. Latyshev, A. Shishov "Development of the active monitoring system for the computer center at IHEP", CEUR-WS, ISSN 1613-0073, 2017.

[5] N. A. Kuznetsov Methods of analysis and synthesis of modular information management systems. - M .: Fizmatlit, 800 p., 2012.