AUTOMATIC OPTIMIZATION OF SQL-QUERIES ON BASIS OF GENETIC ALGORITHMS

Yurij Shabatura[1], Igor Shtelmakh[2], Maksim Shabatura[3]

Vinnytsya national technical university, Khmel'nitske highway, 95, Vinnytsya, 21021, Ukraine,
[1]ph.: (0432) 59-85-71, E-mail: shabatura@vstu.vinnica.ua,
[2]ph.: (0432) 50-58-55, E-mail:info@contentum.com.ua, E-mail: smartmax@i.ua

Annotation

This article is devoted to the problem of automatic optimization of SQL-queries in the computer systems integrated to computer networks. The new way is offered near realization of optimization on the basis of queries structure modification using genetic algorithms.

The task of optimization of SQL- queries is in-process formalized, the flow diagram of the system of optimization is developed. The objective model of query, which is pilchard parser on the basis of its text and allows to modify a query programmatic, is developed. The algorithm of genetic optimization is described and the results of its experimental research are resulted, which confirm a practical value for offered approach.

Entry

The key component of any modern computing system (CS) is a database. The computer systems of the mass use are computer-integrated in computer networks (CSICN) carry out thousands of queries to the databases. Speed of implementation of these queries and consumption by them the limited calculable resources often become principal reason of worsening of dynamics CSICN, or even refuses in maintenance of query as a result of exceeding of possible time of waiting.

A syntax of SQL of queries in modern DBMS is extraordinarily flexible, and getting the same necessary set of queries is possible using a few ten or even hundreds of combinations of statements of query. In same queue the method of construction of SQL of queries can a substantial rank influence on their productivity [1]. For example, at implementation of query of SELECT on massive of information with the use of statement of limitation WHERE, where contained number a few fields, the productivity will depend on order on which these the number written in. On the first place the number of column, which returns the minimum set of records, must be placed expression of WHERE, on the second is a number of column, which returns the next minimum set of records etc [2].

Thus the actual task of automatic optimization of structure of SQL of query appears by the choice of such combination of statements, which provides its most productivity. Such work can be hand-drafted, however needs for this purpose the detailed knowledge of DBMS, all its features, principles of work and secrets, and also lead through of the detailed analysis of array of data, in tables which are used in queries.

Formalization of task

New approach which will allow to do optimization of the productivity of SQL-queries on the basis of genetic algorithms is in-process offered. In quality the object of research DBMS MYSQL, which is most widespread in modern CSICN, is chosen. On the initial stage of optimization a query is entered in the system Q_0 what returns the set of data D_0 for a time $t = t_0$. The system carries out the selection of optimum combination of statements in given query, that the got query Q_{opt} will return set of data $D_{opt} = D_0$ for the interval of time $t = t_{\min}$.

Structure of the system for SQL-queries optimization

The flow diagram of the system of optimization of queries of SQL is showed on fig. 1. The system of optimization consists of two basic constituents: parser of SQL and procedures of genetic optimization. On the entrance of the system an initial request is lodged Q_0. Parser SQL processes an entrance query Q_0 and creates an objective model on the basis of it, and also allows to carry out reverse transformation of the modified objective model in SQL for subsequent implementation and testing of the productivity by procedure of calculation of function of accordance.

Procedure of genetic optimization encodes the objective model of query as a chromosome, and carries out the search of optimum chromosome which provides the minimum value for function of accordance.

The function of accordance settles accounts the proper procedure, and means time of implementation of query of SQL on the server of DBMS. An optimum query which has a minimum time of implementation on DBMS and returns the necessary retrieval of data ensues.

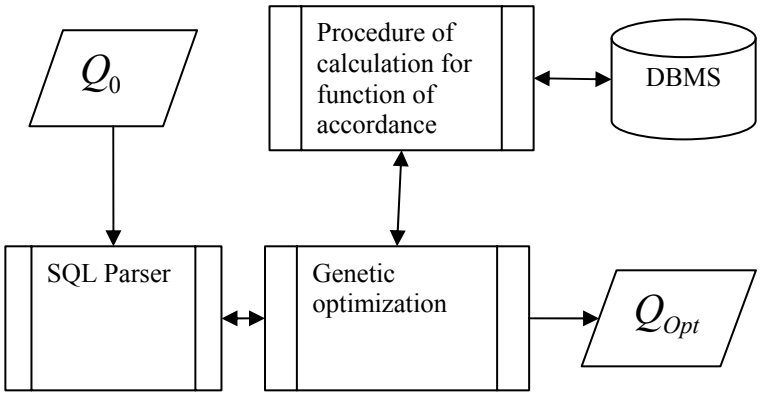

Fig.1. Flow diagram of the system of optimization of queries of SQL

Development of objective model for SQL-query

A syntax of SQL language is difficult for automatic programmatic treatment, as SQL-query is an urgent variable, with the certain set of statements and names of tables, fields, and others like that For this reason for presentation of SQL-query the parser languages of SQL are developed in a comfortable kind, which allows to get the objective model of this query. The diagram of objective model is showed on fig. 2.

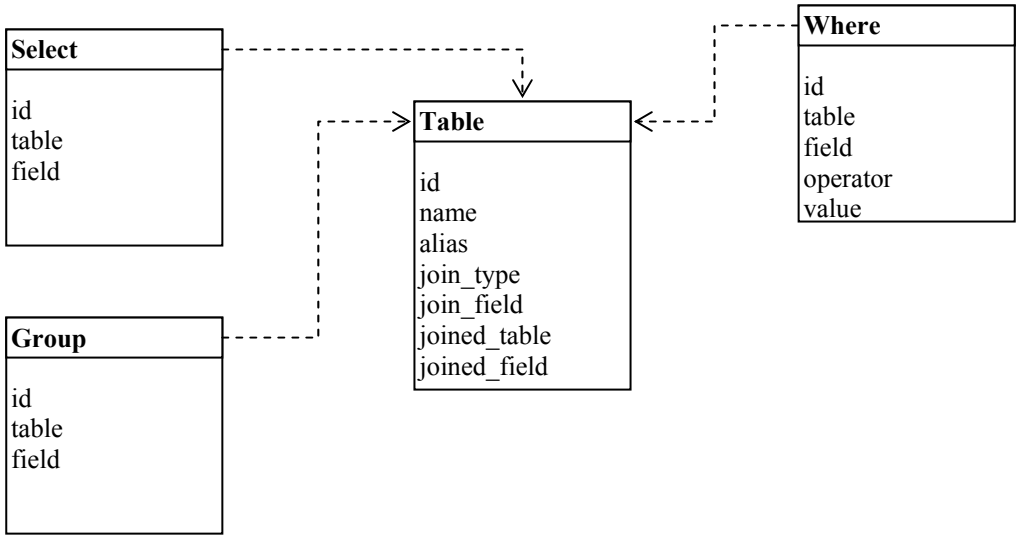

Fig.2. Diagram of classes of objective model of query of SQL

As evidently from resulted figure, a query has such basic classes of objects: Select, Table, Where, Group. The class Select describes the fields which the selection of data will consist of, contains attributes the number of the field (id), code of table, which this field (table) and names of the field belongs to (field). The class Table describes tables which a selection is carried out from, contains attributes the number of table (id), its name (name), code (alias), as connection with other table (join_type), names of the field on which a connection is established with other table (join_field), with which a connection (joined_table) is established the code of table and will name its fields (joined_field). The class of Where is described by limitation which are imposed on data set. Contains the attributes of number of limitation (id), code of table (table) and field (field), on which imposed limitation, statement of limitation (operator) and its value (value). The class of Group describes the terms of grouping of results of selection, contains the attributes of number of condition (id), koda of table (table) and name of the field (field), which grouping is carried out on.

Developed an objective model allows fully to describe SQL-query, it can be easily modified a necessary rank and again to be regenerate in the urgent variable of SQL-query for subsequent implementation. However, the resulted model is not complete, as a syntax of language of SQL is difficult enough, that is why in the process of practical introduction this model will be extended and by other objects for that it could describe SQL-query more detailed.

Genetic optimization of SQL-query

Genetic optimization of SQL-query consists of the next basic stages:

1. Converting of entrance query is into an objective model $Q_0 \to \overline{Q}_{0obj}$.
2. A code of chart of transformation of objective model is in binary a term (chromosome) $\overline{Q}_{0obj} \to \overline{v}_0$.
3. Generation of initial population of chromosomes $\overline{V}_{start}$.
4. Calculation of function of accordance of every chromosome $eval(\boldsymbol{v}_k)$.
5. Selection of the most suitable chromosomes.
6. Crossing and mutation.

The resulted algorithm is cyclic, stages 4-6 executed the set amount of iterations, with the purpose of search of chromosome of SQL-query which will be executed more faster.

Transformation of entrance query Q_0 into object model $\overline{Q}_{0obj}$ carried out by developed parser of SQL.

Essence of optimization of SQL-queries in this work consists in the search of such transformed objective model which will provide the least duration of implementation of query. For realization transformation the charts of optimization of queries are used found in official documentation on DBMS MYSQL [1] and opened on the basis of own supervisions. Will point two basic charts transformations of queries, used in this work:

1. Limitation of WHERE gives the best results, if the first in it is place a condition which chops off the greater set of records as possible.
2. Connecting of auxiliary tables can considerably increase duration of implementation of SQL-queries, as at treatment of considerable volumes of data DBMS forced to create temporal tables for their storage. Replacement of such auxiliary tables by separate data-dictionaries directly in the program allows to accelerate implementation of query and disburden the server of bases given.

For the code of the first chart the plural of numbers is entered $W = \{w_1, w_2, ..., w_{nw}\}$, what mark the sequence number of every expression of condition, where nw - amount of expressions of condition. Numbers which mark a sequence number encoded in a binary kind, dimension of which ws depends on the amount of expressions of condition nw.

For the code of the second chart the plural of binary numbers is used $D = \{d_1, d_2, ..., d_{nt}\}$, where d_n - variable that determines replacement of table with a sequence number n by the programmatic dictionary of data, nt - amount of tables which are used in washed down.

Thus a chromosome will be an aggregate of plurals of binary numbers, which encode the charts of transformation. For the resulted two charts of transformation a chromosome will purchase a kind

$$v = W \cup D = \{w_1, w_2, ..., w_{nw}, d_1, d_2, ..., d_{nt}\}. \quad (1)$$

For the generation of initial population of chromosomes the random generator of numbers, which generates population, is used ($popsize$) chromosomes (binary numbers) by a dimension

$$vs = nw * ws + nt. \quad (2)$$

For the solvable task of optimization the function of accordance is equivalent an objective function (to duration of implementation of SQL of query)

$$eval(\boldsymbol{v}_k) = f(\boldsymbol{x}^k),\ k = 1,2,..., popsize, \quad (3)$$

where $f(\boldsymbol{x}^k)$ turns out on the basis of implementation of SQL-query coded a chromosome $\boldsymbol{x}^k$ by procedure of calculation of function of accordance.

By the function of accordance the estimation of chromosome is executed for the degrees of their adjusted to implementation of criterion of optimization [3].

For the selection of the most suitable chromosomes the method «wheel of roulette» is used, resulted in [3]. A selection is carried out on the basis of function of distributing, which is built the proportionally calculated functions of accordance of generated variants of chromosomes. The thumb-nail sketch of algorithm is resulted below:

1. Calculate the value of function of accordance $eval(v_k)$ for every chromosome concordantly (1).

2. Calculate the public function of accordance of population:

$$F = \sum_{k=1}^{popsize} \left(eval(v_k) - \min_{j=1,popsize} \{eval(v_j)\} \right) \quad (4)$$

3. Calculate probability of selection p_k for every chromosome v_k:

$$p_k = \frac{eval(v_k) - \min\limits_{j=1,popsize}\{eval(v_j)\}}{F}, \quad k = 1,2,...,popsize . \quad (5)$$

4. Calculate the combined probability q_k for every chromosome v_k:

$$q_k = \sum_{j=1}^{k} p_j , \qquad k = 1,2,...,popsize . \quad (6)$$

The process of selection is begun with the rotation of wheel one times; thus for every cycle one chromosome is elected:

1. Generate a random number r from an interval [0, 1].

```
begin
  k :=0 ;
  while ( k ≤ popsize ) do
     r_k := random [0,1] ;
     if ( r_k < sp ) then
       to choose a chromosome v_k for crossing;
     end;
     k := k + 1 ;
  end;
end.
```

Fig.3. Algorithm for procedure of crossing

2. If $r \le q_1$, choose the first chromosome v_1; choose otherwise k-th chromosome v_k ($2 \le k \le popsize$) such that $q_{k-1} < r \le q_k$.

For crossing of chromosomes a method is used from одною exact an exchange, resulted in [3]. In accordance with this method, by chance one point of exchange, which parts of chromosomes-parents switch places in relation to, is elected. The algorithm of procedure of crossing is resulted on rice. 3, where is probability of crossing.

A mutation consists in a change one or more genes with probability even the coefficient of mutation . During work with binary chromosomes a mutation consists in the inversion of the proper bit. Elected a casual rank $mn = Round(mp * [vs * popsize])$ integers are from an interval $[1, vs * popsize]$ but form a plural $M = \{m_1, m_2, ..., m_{mn}\}$. This plural of numbers determines the sequence numbers of bits from the general aggregate of bits of chromosomes, which are subject a mutation. Procedure of mutation is carried out after an algorithm, resulted on fig. 4.

Experimental research and analysis of the got results

For the leadthrough of experimental research it is chosen difficult SQL-query which is used in the real system of analysis of data sale and carries out a selection from four tables, processes over 300 000 queries. The model of data request for the language SQL is showed on fig. 5. Duration of implementation made $t = t_0 = 28{,}832$ s.

```
begin
  k :=0 ;
  while ( k ≤ popsize ) do
     i :=0;
     while ( i ≤ vs ) do
          if ((k * vs + i ) in M) then v_k[i] = not v_k[i] ;
          end;
          i := i + 1 ;
     end;
     k := k + 1 ;
  end;
end.
```

Fig.4. Algorithm of procedure of mutation

select ap.customer_id, af.fabric, ap.value, ap.type, cc.kurs, ap.month from ArtikelsPrognozes ap join tbl_customers c ON c.Auftr_geb_ = ap.customer_id join currency cc ON cc.cur_name = c.Wahrg left join tbl_article_fabrics af ON af.customer=ap.customer_id AND af.article=ap.article where ap.year='2008' and ap.value>0 group by ap.customer_id

Fig.5. Input SQL-query

By parser of SQL transformation is carried out $Q_0 \to \overline{Q}_{0obj}$ to objective model, showed on fig. 6.

Fields:

Table 1

Id	1	2	3	4	5	6
Table	ap	af	ap	ap	cc	ap
Field	customer_id	fabric	value	type	kurs	month

Table 2

Id	Name	Alias	JoinType	JoinField	JoinedTable	JoinedField
1	ArtikelsPrognozes	ap				
2	tbl_customers	c	JOIN	Auftr_geb_	ap	customer_id
3	currency	cc	JOIN	cur_name	c	Wahrg
4	tbl_article_fabrics	af	LEFT JOIN	article	ap	article

Where rules:

Id	Table	Field	Operator	Value
1	ap	year	=	'2008'
2	ap	value	>	0

Fig.6. Objective model of entrance query

On the basis of objective model the amount of expressions of condition and amount of tables turns out which are used in washed down $nt = 4$. A chromosome looks like $v =$ [xx xx xxxx], where the first two pair of binary numbers are marked by the sequence numbers of expressions of condition, four last numbers mark the necessity of replacement of tables the programmatic dictionaries of information.

After the generation of initial population of chromosomes and leadthrough of genetic optimization, on a 258-th iteration, the optimized query was got Q_{opt} transformed a chromosome $v = [10010001]$, the model of this request for the language of SQL is resulted on fig. 7. The chart of process of optimization is resulted on fig.8.

```
select ap.customer_id, ap.value, ap.type, cc.kurs, ap.month from ArtikelsPrognozes ap join tbl_customers c ON
c.Auftr_geb_ = ap.customer_id join currency cc ON cc.cur_name = c.Wahrg where ap.value>0 and ap.year='2008'
group by ap.customer_id
```

Fig. 7. SQL query is optimized

As evidently from to fig. 7, one of tables of query was transferable the programmatic dictionary of information, and also the order of expressions of condition was transferable, that allowed to get the necessary retrieval of data $D_{opt} = D_0$ for the interval of time $t = t_{\min}$ = 1,034 s, that was 3% from duration of implementation of initial query. On fig. 8 the chart of dependence of time of implementation of optimized query is showed from to the number of iteration of genetic algorithm.

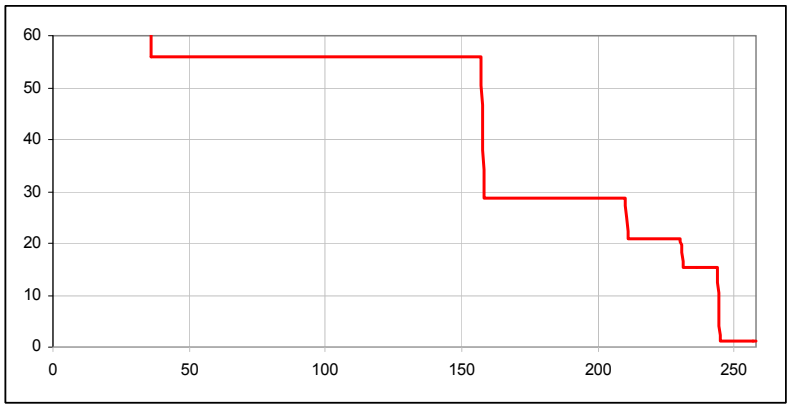

Fig.8. Chart for process of optimization

Conclusions

1. The new going is developed near optimization of SQL-query on the basis of modification of their structure with the use of genetic algorithms.
2. An objective model is developed which allows automatically to analyse and modify the structure of SQL-query.
3. The SQL-parser and block of genetic optimization are realized which will allow to apply the developed approach in practice.
4. Experimental researches s the capacity of the developed approach and it considerable efficiency during optimization of difficult queries. Duration of implementation of entrance test query succeeded to be shortened on 97%, that testifies to the considerable prospects of application of the developed approach for the increase of efficiency of functioning of CSICN.

References:

[1] Optimizing SELECT and Other Statements. Reference Manual for the MYSQL Database System, v. 5.1
[2] Akhmed Abualsemid. Optimization of the productivity of databases for Web. A library is the Internet of Industry of I2R.ru. URL: www.i2r.ru/static/480/out_11207.shtml
[3] A. P. Rotshteyn. Intellectual technologies in authentication. Vinnitsya: Universum. - 1999. - P. 300