N E W S
OF THE NATIONAL ACADEMY OF SCIENCES OF THE REPUBLIC OF KAZAKHSTAN
PHYSICO-MATHEMATICAL SERIES
ISSN 1991-346X
Volume 2, Number 318 (2018), 46 – 53

A.Ongarbayeva[1], L.Smagulova[1], S. Nurmukhanbetov[1], G.Issayeva[2]

[1]Zhetysu state university named after I. Zhansugurov, Taldykorgan;
[2]Caspin university, Almaty
aliaod_80@mail.ru guka_issaeva@mail.ru

MANAGING THE MYSQL DATABASE AND THE STAGES OF DEVELOPMENT OF CLIENT SERVER INFORMATION SYSTEM USING MYSQL

Abstract. The paper covers various aspects and tasks of MySQL database administration, main duties and methodological instructions, principal questions and responsibilities. MySQL management software can be used by any user. The article also presents the methods of modification and usage. It also presents the internal features and abilities, work platforms and database copying, the ability to backup deleted data, the stages of data processing using the client-server technology. The database provides excellent technical features. MySQL is a client-server system, including a multithreaded SQL server, supporting a big number of various computers, programs and libraries, administration tools and offering an advanced API. Data types and methods of table creation are given. MySQL provides an easy-to-use installation procedure. MySQL makes it easy to share data between users.

Key words: MySQL,database administrator, development of IS on client server technology, databases, database types.

The main components of MySQL DBMS are covered. As is well known, MySQL is a relational database management system. In a relational database system, the data is stored in multiple separate tables. That is why, this paper discusses methods of working with tables that can facilitate work in MySQL. The questions concerning SQL queries are considered as well.

MySQL. MySQL DBMS consists of several main components. Knowledge of their essence and designation allows to better understand the nature of the managed system and principles of the work of its various tools. It is strongly recommended to spend a little time to better understand the material given further in the text. It will largely ease the work. In particular it is important to work over the following aspects of MySQL.

- MySQL server. The mysqld server performs all operations on databases and tables. To start the server, monitor its operation and restart in case of failure, the program safe_mysqld (daemon) is used.
- MySQL client programs and utilities. To interact with the server and perform a number of administrative tasks, various MySQL programs are used, the most important of which are the following:
- mysql. An interactive program that allows you to send SQL queries to the server and view the results of their execution.
- mysqladmin. An administrative tool that allows you to perform operations such as shutting down the server, creating and deleting databases. The same program can be used to check the status of the server, if something there is something wrong with its work.
  - isamchk or myisamchk. Utilities designed to analyze and optimize tables, and restore them in case of damage.
  - mysqldump. A tool for backing up databases or copying them to another server.
- SQL is a server-side language. You can perform some administrative tasks only with the mysqladmin command-line utility. Sometimes it is much more effective to complete a task if an administrator can "communicate" with the server in its native language. Suppose that you need to check why user privileges do not work as expected. Unfortunately, it is impossible to directly "talk" to the server in human language. But you can use the mysql client program and then send an SQL query to analyze the permission tables.

- MySQL, developed mostly in C / C ++, has been tested on many platforms, including Windows, Linux, FreeBSD, Mac OS X, OS / 2, Solaris, and others.
- MySQL provides the API (Application Programming Interface) for C, C ++, Eiffel, Java, Perl, PHP, Python, Ruby and Tcl. MySQL can be successfully used both for building Web pages using Perl, PHP and Java, and for creating a desktop application program using Delphi, Builder C ++ or the .NET platform.

The development of IS using client-server technology consists of several stages:

1. the server part of the DBMS is installed on the server in a computer network (for example, Microsoft SQL Server, MySQL, Oracle. If a web-interface is implemented, then a web server program is installed on the server too (for example, Apache);
2. If client applications are implemented, then the client part is installed on all client parts of the network (this step is not necessary and is performed only if users of the information system have the ability to manage the server)[7];
3. The server and the client parts of the DBMS and the web server are configured;
4. The data structure (links between tables and field types) is defined, primary and secondary tables are also defined in queries;
5. Tables and queries running on the server side are created. Before creating queries, the tables are filled with initial data. Stored procedures, user functions, diagrams, and triggers are also created;
6. In the case of using a client application, connectivity objects are created using the programming language, they are connected to tables, queries and stored procedures. Also, queries and stored procedures to be executed on the server side are created;
7. Forms are created;
8. Reports are created;
9. The system is filled up with real data.

The MySQL software is open source software.

Open source software means that anyone can use and modify it. Such software can be obtained over the Internet and used for free. In this case, each user can study the source code and change it according to his needs.

Technical capabilities of MySQL DBMS

MySQL is a client-server system that contains a multi-threaded SQL server that supports various database computers, as well as several different client programs and libraries, administrative tools, and a wide range of APIs.

Security

The security system is based on privileges and passwords with the ability to verify them from a remote computer, thereby providing flexibility and security. Passwords are crypted when they are sent over the network in the process of getting connected to the server. Clients can connect to MySQL using TCP / IP sockets, Unix sockets or named pipes (named pipes, under NT)

MySQL is very fast, reliable and easy to use. If you need namely these features, try working with this server. MySQL also has a number of convenient features, developed in close contact with users. Initially, the MySQL server was designed to manage large databases in order to provide faster performance compared to existing analogues at that time. And since then this server has been successfully used by enterprises with high requirements. Despite the fact that MySQL is constantly improving, it already provides a wide range of useful functions. Due to its availability, speed and security, MySQL is very suitable for accessing databases on the Internet.

It's only rarely that you need to view the whole output of the query at once (that is, all the records that satisfy the query expression). For example, we may need only to calculate how many records satisfy a particular condition, or to select only the first 10 records from the data. The mechanism of using sockets implies using client-server technology, which means that a special program must be running in the system-a MySQL server that accepts and processes requests from client programs. Since all work is actually done on the same machine, the overhead of working with network resources is insignificant (installation and maintenance of connection with MySQL server is quite cheap). MySQL has a three-level architecture: databases - tables - records. The MySQL databases and tables are physically represented by files with the frm, MYD, MYI MySQL is very fast, reliable and easy to use. If you need namely these

features, try working with this server. MySQL also has a number of convenient features, developed in close contact with users. Initially, the MySQL server was designed to manage large databases in order to provide faster performance compared to existing analogues at that time. And since then this server has been successfully used by enterprises with high requirements. Despite the fact that MySQL is constantly improving, it already provides a wide range of useful functions. Due to its availability, speed and security, MySQL is very suitable for accessing databases on the Internet.

It's only rarely that you need to view the whole output of the query at once (that is, all the records that satisfy the query expression). For example, we may need only to calculate how many records satisfy a particular condition, or to select only the first 10 records from the data. The mechanism of using sockets implies using client-server technology, which means that a special program must be running in the system-a MySQL server that accepts and processes requests from client programs. Since all work is actually done on the same machine, the overhead of working with network resources is insignificant (installation and maintenance of connection with MySQL server is quite cheap). MySQL has a three-level architecture: databases - tables - records. The MySQL databases and tables are physically represented by files with the frm, MYD, MYI extensions. Logically, a table is a collection of records. A record is a collection of fields of different types. The name assigned to a MySQL database is unique within the system, and the table names are unique within the database and the field names are unique within the table. A single MySQL server can support multiple databases, access to which can be differentiated by login and password. Knowing this login and password, you can work with a specific database. For example, you can create or delete a table in it, add entries, etc. Usually, the name-and password are assigned by the hosting providers, which provide MySQL support for their users..

Data types

Processing numeric data

MySQL provides five integrak types of data, each of which can be signed (by default) or unsigned (by adding the word UNSIGNED after the type name).

Type name	*bits*	*signed range*	*unsigned range*
TINYINT	8	-128..127	0..255
SMALLINT	16	-32768..32767	0..65535
MEDIUMINT	24	-8388608..8388607	0..16777215
INTEGER	32	-147483648..2147483647	0..4294967295
BIGINT	64	$-(2^{63})..(2^{63}-1)$	$0..(2^{64})$

The INT type is used as an alias for INTEGER. Other aliases are: INT1 = TINYINT, INT2 = SMALLINT, INT3 = MEDIUMINT, INT4 = INT, INT8 = BIGINT, and MIDDLEINT = MEDIUMINT.

Processing String Data

MySQL supports the following string types (M denotes the maximum displayed size or PRECISION).

CHAR (M) - a string of fixed length always complemented by spaces on the right. M can be in the range from 1 to 255 characters.

VARCHAR (M) is a string of variable length. Closing whitespace is deleted by the database when writing a value. M can be in the range from 1 to 255 characters.

ENUM ('value1', 'value2', ...) is an enumeration. A string object that can have only one value selected from the specified list (or NULL). The list can have up to 65,535 different values.

SET ('value1', 'value2', ..) is a set. A string object that can have multiple values (or none), each of which must be selected from the specified list. SET can have a maximum of 64 elements.

The length of the CHAR and VARCHAR types is limited to 255 bytes. All string types support binary characters, including NULL (for literal strings, use the $ dbh-> quote () method).

The following aliases are also supported:

BINARY (num) CHAR (num) BINARY

CHAR VARYING VARCHAR

LONG VARBINARY BLOB
LONG VARCHAR TEXT
VARBINARY (num) VARCHAR (num) BINARY

Basic Commands

You can view the list commands supported by mysql program by running it with the --help option:

shell>mysql --help

The list of commands displayed by this command is well documented and therefore is not given here.You can start the MySQL database as follows:

mysql [OPTIONS] database

Description of MySQL database

The client part of MySQL is called mysql. It provides a command-line interface for MySQL and a non-interactive batch processing capability.The options supported by mysql are shown in the table below. You can use either a "short" single character or a more detailed version.

Mysql options

option	**Description**
-\?, --help	Reference
option	**Description**
-d, --debug=[options]	*Output the debug information to the protocol In the following general form 'd: t: o, filename'*
-d, --debug-info	Output debugging information on exiting the program
-e, --exec	*Execute the command and exit, the implicit form of the --batch option*
-f, --force	Continue even if we encounter a SQL error
-h, --hostname=[hostname]	Specifies the name of the server you want to connect to
-P, --port=[port]	Port for connecting to MySQL server
-p, --password=[password]	The user's password for connecting to the MySQL server. Note that there should not be a space between -p and password
-q, --quick	Fast (non-bufferized) output, can slow down the server if the output is suspended
-s, --silent	Silent output
-u, --user=[user]	The user name for connecting to the MySQL server. It is optional, if the user name is the same as your login. By default, your login name is used as a user name, which simplifies configuration
Окончание табл. 1	
Опция	**Описание**
-v, --verbose	Detailed output. -v option can be doubled or tripled for more detailed output.
-w, --wait	If the connection fails, then wait and try again.
-B, --batch	Run in batch mode. No requests and errors shown in STDOUT. It is automatically set when reading from a record to a pipe (pipe). The results will be displayed in a tab-separated format. One line of the result corresponds to one line of output
-I, --help	Equivalent to ?
-V, --version	Print version information

In interactive mode, mysql will print results in a table similar to the example below. If no password or user name are provided, mysql will try to log in to the database server using your login and NULL (BLANK) password. If your mysql login is different from your login in unix, or if you have a password, the connection to MySQL will not be performed.For example:

bsd# mysql -u coobic
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 4 to server version: 4.0.12-max
Type 'help;' or '\h' for help.Type '\c' to clear the buffer.

Implementation of SQL in MySQL DBMS

Database creationTo see a listof databases currently existing on the server, you can use the SHOW command:

mysql> **SHOW DATABASES;**

```
+----------+
| Database |
+----------+
| mysql    |
| test     |
| trf      |
+----------+
```

3 rows in set (0.00 sec)

The mysql database is essential as it stores the user access rights. The test database is often used for experiments.However, you can not see all the databases, if you do not have the permission to run SHOW DATABASES.If the test database exists, try accessing it:

mysql>**USEtest**

Databasechanged

The USE command, like QUIT, does not need a semicolon (of course, you can also end the command with a semicolon). The USE command differs from others also by something else: it should be specified in one line.If the administrator creates a database for you and grants all permissions, you can start working with it right away. In the absence of a database, you will have to create it yourself:

mysql> **CREATE DATABASE perpetuum;**

Query OK, 1 row affected (0.00 sec)

In Unix, the case of database names (unlike in SQL keywords) matters, so in this OS you will always have to remember that perpetuum, and Perpetuum or something else are different names. The same rule applies to table names (in Windows this restriction does not work.However when accessing databases and tables within one query, you should follow only one particular case).When creating a database, it is not selected automatically; you must select it separately. Make perpetuum the current database using the following command:

mysql> **USE perpetuum**

Database changed

You need to create a database only once, but you must select it in each mysql session. You can do this using the USE command above. And also you can choose the database from the command line when running mysql. To do this, just enter its name after the connection parameters . For example:

shell>**mysql -h host -u user -p perpetuum**

Enter password: ********

Note: in the above command, perpetuum is not a password. Enter the password at the command line after the -p option without a space (for example, -pmypassword, not -p mypassword). However, it is still better not to specify the password in the command lined for security reasons.

Creating a table

Using the CREATE TABLE command, we define the structure of the new table:

mysql> **CREATE TABLE TABLE1 (**

-> **NUMBER INTEGER NOT NULL AUTO_INCREMENT,**

-> **FAMALY VARCHAR(35),**

-> **NAME VARCHAR(30),**

-> **OTCHESTVO VARCHAR(35),**

-> **DESCRIPTION BLOB,**
-> **PHOTOFILE BLOB,**
-> **EMAIL VARCHAR(40),**
-> **ZVANIE VARCHAR(34),**
->**DOLGNOST VARCHAR(40),**
-> **PRIMARY KEY (NUMBER)**
-> **);**
Query OK, 0 rows affected (0.06 sec)

Now that the table is created, the SHOW TABLES command should output the following:

mysql>**SHOW TABLES;**

```
+--------------------+
| Tables_in_perpetuum |
+--------------------+
| TABLE1              |
+--------------------+
1 row in set (0.00 sec)
```

1 row in set (0.00 sec)

To check if the table was created correctly according to the plan, you can use the DESCRIBE command:

mysql>**DESCRIBE pet;**

```
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| name    | varchar(20) | YES  |     | NULL    |       |
| owner   | varchar(20) | YES  |     | NULL    |       |
| species | varchar(20) | YES  |     | NULL    |       |
| sex     | char(1)     | YES  |     | NULL    |       |
| birth   | date        | YES  |     | NULL    |       |
| death   | date        | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
```

You can use the DESCRIBE command at any time, for example, if you forget the column names or the types to which they belong.

mysql>**describe TABLE1;**

```
+-------------+-------------+------+-----+---------+----------+
| Field       | Type        | Null | Key | Default | Extra    |
+-------------+-------------+------+-----+---------+----------+
| NUMBER      | int(11)     |      | PRI | NULL |auto_increment|
| FAMALY      | varchar(35) | YES  |     | NULL    |          |
| NAME        | varchar(30) | YES  |     | NULL    |          |
| OTCHESTVO   | varchar(35) | YES  |     | NULL    |          |
| DESCRIPTION | blob        | YES  |     | NULL    |          |
| PHOTOFILE   | blob        | YES  |     | NULL    |          |
| EMAIL       | varchar(40) | YES  |     | NULL    |          |
| ZVANIE      | varchar(34) | YES  |     | NULL    |          |
| DOLGNOST    | varchar(40) | YES  |     | NULL    |          |
+-------------+-------------+------+-----+---------+----------+
```

9 rowsinset (0.02 sec)

Loading data into a table

Having created a table, you need to fill it with data. To do it you can use the LOAD DATA and INSERT commands .

mysql>**INSERT INTO TABLE1 VALUES (NULL,**
->'Nurbekov', 'Dias', 'Muratovich',
->'coobic', NULL, 'nur_bek@mail.ru',
->NULL, NULL);
Query OK, 1 row affected (0.03 sec)

For the system to work properly you need the following pre-installed software: Apache Web Server, PHP 4.3.x, MySql, on the FreeBSD or Linux platform. You can also install the system on the Windows platform. To work with the client part of the system, you need a computer connected via TCP / IP to the network in which the MySQL server is located

REFERENCES

[1] Administrirovaniye MySQL. M.: NOI Intuit. **2016**.
[1] Dokumentatsiya k serveru MySQL. Elektr. dan. – Rezhim dostupa: http://www.mysql.org
[2] Izuchayem SQL. Alan Byuli. Sankt-Peterburg. **2007**.
[3] MySQL. Rukovodstvo administratora. Viliams. **2005**.
[4] Proyektirovaniye informatsionnykh sistem. Khabarovsk Izdatelstvo KhGTU **2005**. G. K. Konopelko
[5] MySQL. Rukovodstvo po izucheniyu yazyka. L.Ulman. Piter. **2004**.
[6] Burkov A.V. Proyektirovaniye informatsionnykh sistem po tekhnologii kliyent – server v «Microsoft SQL Server 2008» i «Microsoft Visual Studio 2008»Izdatelstvo: Mariyskiy gosudarstvennyy universitet. **2009**.

А.Д. Онгарбаева[1], Л.А. Смагулова[1],
С.М. Нурмуханбетов [1], Г.Б. Исаева[2]

[1]Жетысуский государственный университет им.И. Жансугурова, Талдыкорган, [2]Капийский общественный университет, Алматы

УПРАВЛЕНИЕ БАЗАМИ ДАННЫХ MYSQLI ЭТАПЫ РАЗРАБОТКИ КЛИЕНТ-СЕРВЕРНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ MYSQL

Аннотация: В работе рассматривается различные аспекты и обязанности администрирования на базе данных MySQL основные обязанности и методические инструкции, основные вопросы и обязанности управления программное обеспечение MySQL могут использоваться любым пользователем. Поэтому в статье приводится методы службы модификации и утилизацию. Также приводятся внутренняя характеристика и способности, рабочие платформы и копирование базы данных, возможность резервирования удаляющихся данных, этапы обработки данных технологией клиент-сервер, обработанные в связи с пользователем MySQL предлагают большими возможностями по технической характеристике MySQL-это система клиент-сервер, он включает многопоточный SQL-сервер, а также поддерживает базы данных вычислительных машин, программы и библиотеки, инструменты администрирования, предлагает расширенный программный интерфейс. Приведены типы данных и методы создания таблиц. Развитие систем базы данных облегчает процедуру инсталляции и применение MySQL. MySQL обеспечивает простоту работы между пользователями.

Рассмотрены основные компоненты СУБД MySQL. Как известно, MySQL является реляционной системой управления базами данных. В реляционной системе базы данных не все данные сохраняются, но многие данные сохраняются в отдельных таблицах. Поэтому в данной работе рассматриваются методы

работы с таблицами, которые могут облегчить работу в MySQL. Рассматриваются вопросы MySQL как структурированный язык запросов на SQL.

Ключевые слова: MySQL, администратор баз данных, разработка ИС по технологии клиент сервер, базы данных, типы базы данных MySQL

УДК 007

Д. Онгарбаева[1], Л.А. Смагулова[1], С.М. Нұрмұханбетов[1], Г.Б. Исаева [2]

І. Жансүгіров атындағы ЖМУ, Талдықорған қаласы, Каспий қоғамдық университеті

MYSQLДЕРЕКТЕР ҚОРЫН БАСҚАРУ МЕН ОНЫ ҚОЛДАНЫП КЛИЕНТ-СЕРВЕРЛІК АҚПАРАТТЫҚ ЖҮЙЕНІ ӨҢДЕУ ЭТАПТАРЫ

Аннотация. Мақалада MySQL администрирлеудің түрлі аспектілері мен міндеттерді сәтті орындауға меңгеруге қажет негізгі сұрақтар қарастырылады. Администратордың негізгі міндеттері мен оны орындау нұсқаулықтары, MySQL басқарудың негізгі сұрақтары мен басқарушының міндеттері мен MySQL бағдарламалық қамсыздандыруы- кез-келген адам оны қолдануына және модифицирлеуіне болатын ашық кодты бағдарламалық қамсыздандыру жайлы айтылады. MySQL клиенттік программалары мен утилиталарының қызметтеріне шолу жасалады. Ішкі сипаттамасы мен қабілеттілігі, жұмыс жасау платформалары мен деректер қорын көшіру және резервтеудегі ақпараттың жойылуы. Сонымен қоса ақпараттық жүйені клиент-сервер технологиясы бойынша өңдеу этаптары. Қолданушымен тығыз қатынасқа өңделген MySQL бірқатар ыңғайлы мүмкіндіктері, MySQL қолданушымен тығыз қатынаудың бірқатар мүмкіндіктеріне ие. Техникалық сипаттамалары бойынша MySQL клиент-серверлік жүйе, көплегті SQL-серверді ендіреді, түрлі есептеуіш машиналар деректер қорларын қолдайды, сонымен қатар клиентті бағдарламалар мен кітапханаларды, администрирлеу құралдары мен бағдарламалық интерфейстің кеңейтілген спектрін ұсынады. Деректер типтері мен оларды кесте құруда қолдану жазылған. Деректер қоры жүйелерінің дамуы инсталляциялау процедурасы мен MySQL қолдану жеңілдей түседі. MySQL жұмыстың қарапайымдылығы қолданушылар арасындағы кеңінен қолданудың себебі болғандығы жайлы қарастырылады.

Түйін сөздер: MySQL, деректер қоры администраторы, клиент-сервер технологиясы бойынша АЖ өңдеу, деректер қоры, MySQL деректер қорының типтері.