

项目一 数据库概述

任务 1-1 数据库简介

一、数据库概念

什么是数据库？

在日常生活中，我们几乎每天都在使用数据库，比如我们手机里的通讯录，存储的联系人，我们输入姓名、电话号码等信息，打电话的时候直接查找出来就可以进行拨打电话，其实这个通讯录就是一个数据库。

数据库(Database)就是存储数据的仓库。这个仓库按照一定的规律，对数据进行组织和存储，我们可以通过数据库管理系统对这些数据进行管理。

那什么是数据库管理系统？数据库管理系统(Database Management System)用于创建、应用和维护数据库，管理员可以对数据库进行统一的管理和维护，用户可以通过数据库访问数据库中的数据。

二、数据库分类

目前常用的数据库模型主要分为关系型数据库和非关系型数据库。

关系数据库是以关系模型为基础进行创建的数据库，关系型数据库模型用二维表的形式表示复杂的数据结构，将数据的操作几乎都建立在一个或多个关系表格上，通过这些二维表对数据进行管理。我们现实世界中的联系都是用关系模型来表示的。

关系数据库的代表有：MySQL、Oracle、Microsoft SQL Server、Access 及 PostgreSQL 等。

例：

学生表		
学号	姓名	分数
2017001	张三	89
2017002	李四	94
2017003	王五	73

表 1-1 数据表例子

非关系型数据库是对不同于传统的关系数据库的数据库管理系统的统称。与关系数据库最大的不同点是不使用 SQL 作为查询语言。

非关系数据库的代表有：BigTable(Google)、Cassandra、MongoDB、CouchDB。

三、MySQL 简介

MySQL 软件开发者为瑞典 MySQL AB 公司，被 Sun 公司收购后，又被 Oracle 公司收购。目前被广泛地应用在 Internet 上的大中小型网站，如网易、新浪、Facebook 等。

MySQL 的特点：

1、MySQL 是一款开源的软件。开源意味着任何人都可以修改它的缺陷，大家可以得到它的源代码，从而进行学习和修改甚至重新发放

2、MySQL 可以跨平台。MySQL 可以在 Windows 操作系统上运行，也可以在 Linux、UNIX、Mac OS 等操作系统上运行，而 SQL Server 数据库只能在 Windows 操作系统上运行。

3、MySQL 具有很大的价格优势。MySQL 分为社区版和企业版。社区版可以在官网自由免费下载使用，但没有官方的技术支持，企业版则需要付费下载和使用，有官方完善的技术支持，收费也很便宜。

4、MySQL 功能强大且使用方便。

MySQL 是多用户、多线程的数据库服务器。它运行速度快，数据处理上安全有效。MySQL 的操作非常简单易用，占用的空间也相对比较小。

MySQL 也有一些不足，对于大型的项目来说，MySQL 在容量和安全性上就相对没有 Oracle 数据库有优势。

任务 1-2 MySQL 的工作流程

1、客户端向服务器发送连接请求，连接管理器接受到请求，线程管理器将一个用户请求对应一个线程，用户模块对用户连接请求进行权限验证，通过验证就与 MySQL 数据库连接，如果没有指令则这个管理连接就处于空闲状态，太长时间不使用就会自动断开连接。

2、连接上数据库，就可以发送 sql 语句，不同的语句会有不同的解析器负责处理，如果是查询语句，查询缓存会检查之前是否执行过这个操作，如果有，那么就直接把结果返回给用户，当然，如过表发生了变化，那么与之相关的缓存就会失效。一般查询缓存默认情况下是不开启的。

3、如果缓存中没有检查到存在这个操作命令，就会把命令发过去给解析器，解析器判断命令属于什么语句，如果是 select 语句就由优化器进行优化；如果是 update、insert、delete 语句就交给表定义模块；如果是 repair 的相关语句就交给表维护模块；如果是复制功能，主 mysql 服务器上的数据同步到从 mysql 服务器上，就交给复制模块。无论哪个模块，用户必须具有权限才能完成操作。

4、结果返回给客户端，如果查询缓存开启了，就会备份一份到查询缓存中。

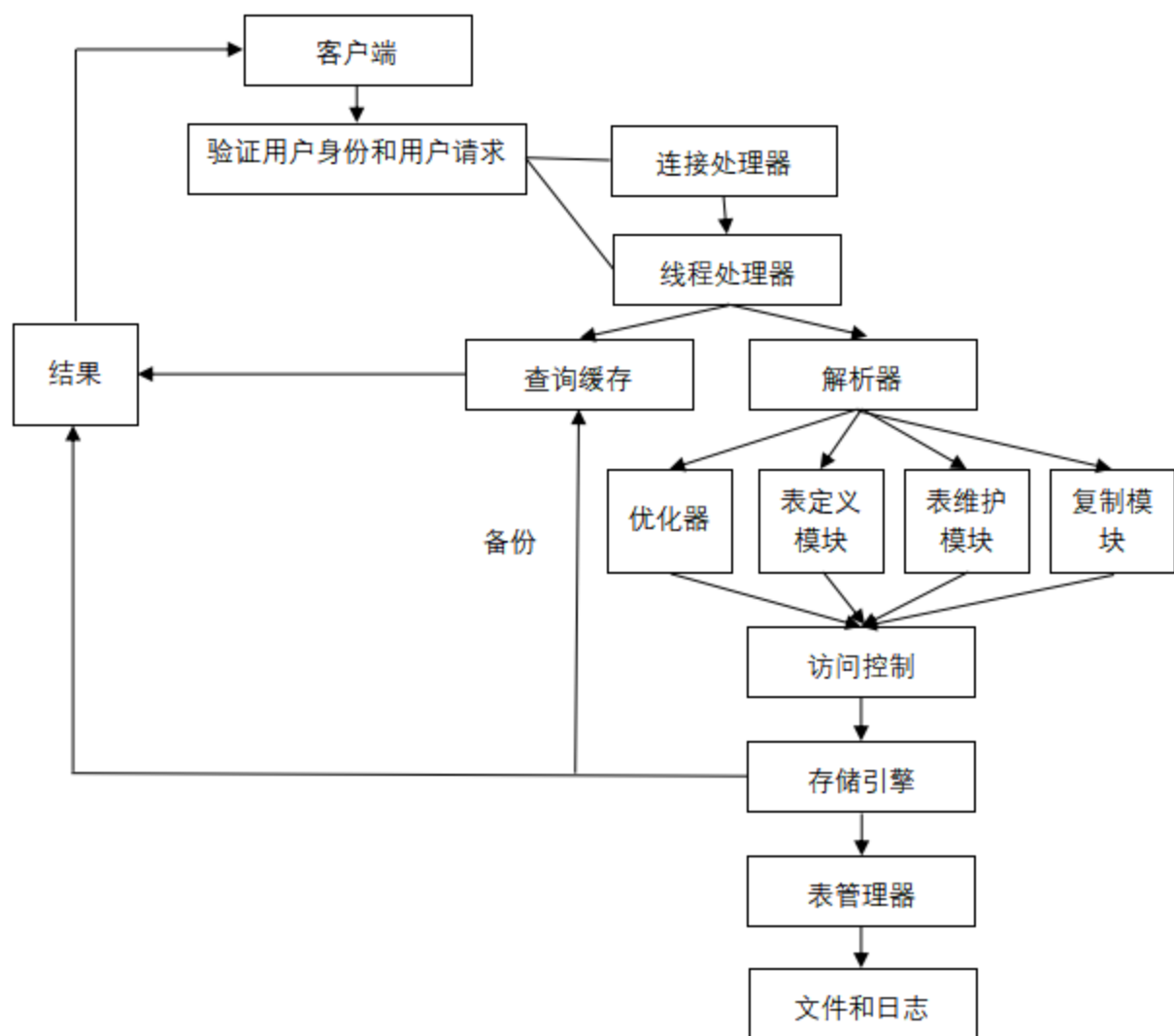


图 1-1 MySQL 工作流程

任务 1-3 MySQL 数据库技术构成

数据库技术的出现是为了更加有效的管理和存取大量的数据资源。简单的来讲，数据库技术主要包括数据库系统、SQL 语言、数据库访问技术。

一、数据库系统

数据库系统比数据库的范围大很多，由数据库、数据库管理系统以及应用开发工具等构成，也可以包括数据库管理员和用户。

数据库用来存储数据，数据库管理系统是用来定义、管理、维护数据的软件，它是数据库系统的重要组成部分。应用开发工具就是用来开发应用系统的。应用系统是要使用数据库的软件。比如成绩管理系统就是一个应用系统，这个应用系统需要数据库管理它的数据。

二、SQL 语言

SQL 全称 Structured Query Language，即结构化查询语言。SQL 语言指令

只能被数据库识别。其它编程语言可以组织 SQL 语句发送给数据库，数据库根据接收到的 SQL 语句，再执行相应的操作。

根据功能划分，SQL 语言主要由以下几部分组成：

1. DML (Data Manipulation Language, 数据操作语言), 用于插入、修改、删除数据库, 如 INSERT、UPDATE、DELETE。

2. DDL (Data Definition Language, 数据定义语言), 用于建立数据库、数据库对象、定义数据表结构等, 如 CREAT TABLE、CREAT VIEW、DROP TABLE。

3. DQL (Data Query Language, 数据查询语言), 用于对数据库中的数据进行查询, 如 SELECT。

4. DCL (Data Control Language, 数据控制语言), 用于控制数据库组件的存取许可、存取权限等。

三、数据库访问技术

在不同的应用系统, 会有各自不同的程序语言使用不同的数据库访问技术访问数据库, 程序语言通过这些技术, 执行 SQL 语句, 进行数据库的管理。

1. ODBC

Open Database Connectivity (ODBC, 开放数据库互连), 提供了一种标准的 API (应用程序编程接口) 方法来访问数据库管理系统 (DBMS)。这些 API 利用 SQL 来完成其大部分任务。ODBC 本身也提供了对 SQL 语言的支持, 用户可以直接将 SQL 语句送给 ODBC。ODBC 的设计者们努力使它具有最大的独立性和开放性: 与具体的编程语言无关, 与具体的数据库系统无关, 与具体的操作系统无关。

2. ADO

微软公司的 ActiveX Data Objects (ADO) 是一个用于访问数据源的 COM 组件, 作为高层的编程界面层。ADO 是在 OLE DB 之上, 包含了很多层次化的 COM 对象与集合 (Collections, 也是一类对象, 在其里面包含了其他层级对象)。允许开发人员编写访问数据的代码而不用关心数据源是如何实现与访问驱动的, 而只用关心到数据库的连接。访问数据库的时候, 关于 SQL 的知识不是必要的, 但是特定数据库支持的 SQL 命令仍可以通过 ADO 中的命令对象 (Command) 来执行。

3. MDAC

Microsoft Data Access Components (MDAC) 是微软专门为数据访问功能而发展的应用程序开发接口, 做为微软的统一化数据访问 (Universal Data Access; UDA) 解决方案的核心组成, 最初的版本在 1996 年 8 月发表。目前其组成组件有 ODBC, OLE DB 以及 ADO, 其中 ADO 是在 Visual Basic 上唯一的数据访问管道,

而 OLE DB 则是基于 COM 之上，供 C/C++ 访问与提供数据的接口，ODBC 则是统一化的数据访问 API。

4. JDBC

Java Database Connectivity (JDBC, Java 数据库连接) 是 Java 语言中用来规范客户端程序如何来访问数据库的应用程序接口，提供了诸如查询和更新数据库中数据的方法。JDBC 也是 Sun Microsystems 的商标。JDBC 是面向关系型数据库的。

任务 1-4 MySQL 服务器和端口号

我们前面已经介绍了，数据库系统由数据库管理系统和数据库(数据)组成。而数据库服务器是由在局域网中运行的计算机与数据库管理系统软件共同构成的，数据库服务器可以根据客户端的请求提供相关数据服务。

数据库服务器在数据库系统基础上建立，既有数据库系统的特点，又有它自己的特点。数据库服务器一般有以下功能：

- 1、管理数据库，包括管理配置系统、存取与更新数据管、数据完整性和数据安全性的管理。
- 2、的查询和操作数据库，如检索数据库和修改数据库。
- 3、维护数据库，包括导入/导出数据，维护数据库结构、恢复数据和监测性能。

由于数据库一般来说在同一时间需要允许多人同时访问，因此数据库服务器必须支持并行运行机制，能够同时处理多个同时发生的事件。

MySQL 服务器就是一个非常优秀的数据库服务器，它快速，支持多用户多线程。MySQL 服务器不但支持使用关键任务、重负载生产系统，还可以将它嵌入到一个大配置(mass- deployed)的软件中去。

可以通过命令“show global variables like 'port';”查看端口号，mysql 的默认端口是 3306。

```
mysql> show global variables like 'port';
```

可以通过编辑/etc/my.cnf 文件对端口号进行修改。

任务 1-5 腾讯云 MySQL 环境的安装和配置

腾讯云 MySQL 环境的安装与配置

1、下载 mysql

在官网上下载安装包 `mysql-5.7.26-e17-x86_64.tar.gz`

下载地址: <https://downloads.mysql.com/archives/community/>

MySQL Product Archives
MySQL Community Server (Archived Versions)

Please note that these are old versions. New releases will have recent bug fixes and features!
To download the latest release of MySQL Community Server, please visit MySQL Downloads.

Product Version: **5.7.26** ← 选择要下载版本
Operating System: **Red Hat Enterprise Linux / Oracle Linux** ← 选择对应的系统
OS Version: **Red Hat Enterprise Linux 7 / Oracle Linux 7 (x86_64-bit)** ← 选择系统版本

Package Name	Release Date	Size	Download
RPM Bundle <small>(mysql-5.7.26-1.el7.x86_64.rpm-bundle.tar)</small>	Apr 15, 2019	506.3M	Download
RPM Package, MySQL Server <small>(mysql-community-server-5.7.26-1.el7.x86_64.rpm)</small>	Apr 15, 2019	165.5M	Download
RPM Package, Client Utilities <small>(mysql-community-client-5.7.26-1.el7.x86_64.rpm)</small>	Apr 15, 2019	24.2M	Download
RPM Package, Embedded <small>(mysql-community-embedded-5.7.26-1.el7.x86_64.rpm)</small>	Apr 15, 2019	44.9M	Download
Compressed TAR Archive <small>(mysql-5.7.26-e17-x86_64.tar.gz)</small>	Apr 13, 2019	681.0M	Download
Compressed TAR Archive, Test Suite <small>(mysql-test-5.7.26-e17-x86_64.tar.gz)</small>	Apr 13, 2019	30.5M	Download
TAR <small>(mysql-5.7.26-e17-x86_64.tar)</small>	Apr 13, 2019	711.4M	Download

点击下载.tar.gz安装包

将下载好的安装上传到 Linux 服务器/home 目录下, 或者在 Linux 系统中使用 `wget` 下载

```
[root@localhost ~]# wget
```

```
https://cdn.mysql.com/archives/mysql-5.7/mysql-5.7.26-e17-x86\_64.tar.
```

[gz](https://cdn.mysql.com/archives/mysql-5.7/mysql-5.7.26-e17-x86_64.tar.gz)

2、MySQL 安装与配置

```
[root@localhost home]# tar -zxvf mysql-5.7.26-e17-x86_64.tar.gz -C /usr/local/ #将mysql解压到/usr/local目录中
```

```
[root@localhost home]# cd /usr/local/ #切换到/usr/local目录
```

```
[root@localhost local]# mv mysql-5.7.26-e17-x86_64 mysql #将刚解压目录改名为mysql
```

```
[root@localhost local]# mkdir mysql/data #在mysql目录创建data目录用于存放数据库
```

```
[root@localhost local]# useradd mysql -M -s /sbin/nologin #创建mysql用户 -M不创建用户家目录, -s指定shell为/sbin/nologin, 使用该用户
```

无法登录系统

```
[root@localhost mysql]# chown mysql:mysql /usr/local/mysql #设置
/usr/local/mysql 目录所有者、所属组都为 mysql
```

```
[root@localhost ~]# vim /etc/my.cnf #使用
```

vim打开/etc/my.cnf, 并将mysql配置文件修改如下

```
[mysqld]
datadir=/usr/local/mysql/data
port=3306
# Disabling symbolic-links is recommended to prevent assorted security
risks
symbolic-links=0
# Settings user and group are ignored when systemd is used.
# If you need to run mysqld under a different user or group,
# customize your systemd unit file for mariadb according to the
# instructions in http://fedoraproject.org/wiki/Systemd
[mysqld_safe]
log-error=/var/log/mariadb/mariadb.log
pid-file=/var/run/mariadb/mariadb.pid
#
# include all files from the config directory
#
!includedir /etc/my.cnf.d
```

```
[root@localhost ~]# mkdir /var/log/mariadb
```

```
[root@localhost ~]# touch /var/log/mariadb/mariadb.log [root@localhost
```

```
~]# chown -R mysql:mysql /var/log/mariadb/
```

```
[root@localhost bin]# ./mysqld --initialize --user=mysql
--basedir=/usr/local/mysql --datadir=/usr/local/mysql/data #初始化
mysql
```

```
[root@localhost bin]# ./mysqld --initialize --user=mysql --basedir=/usr/local/mysql --datadir=/usr/local/mysql/data
2023-07-20T17:25:44.287631Z # [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for more details).
2023-07-20T17:25:44.368652Z # [Warning] InnoDB: New log files created, LSN=45790
2023-07-20T17:25:44.487982Z # [Warning] InnoDB: Creating foreign key constraint system tables.
2023-07-20T17:25:44.617722Z # [Warning] No existing UUID has been found, so we assume that this is the first time that this server has been started. Generating a new UUID: 789f966-722-13ab-6600-4804480c7cc.
2023-07-20T17:25:44.679652Z # [Warning] SQL table is not ready to be used. Table 'mysql.plugin' cannot be opened.
2023-07-20T17:25:44.688572Z # [Note] A temporary password is generated for root@localhost: 10371716gn ← 初始化后会生成临时密码, 需要把它记下来
```

```
[root@localhost ~]# /etc/init.d/mysql.server start
```

3、设置PATH变量

```
[root@localhost ~]# mysqld
bash: mysqld: 未找到命令...
[root@localhost ~]# echo "export PATH=$PATH:/usr/local/mysql/bin" >>
/etc/profile          #写入环境变量
[root@localhost ~]# source /etc/profile          #使当成生效
[root@localhost ~]# mysql -uroot -p           #进行mysql,
没有修改密码无使用指令
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.26 MySQL Community Server (GPL)

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights
reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.
mysql> show databases
```

4、修改用户密码

我们使用数据库的时候,会遇到 MySQL 需要修改密码的情况,比如密码太简单需要修改等。以下是修改用户密码的方法。

```
mysql> alter user root@localhost identified by '12345678';
```

5、设置 mysql 开机随系统启动

我们要使用 mysql 数据库就要先开启它的服务,否则就会出错。我们可以通过设置让 mysql 服务开机自启动的方式,减少这样的出错。方法如下:

```
[root@localhost ~]#cp -a /usr/local/mysql/support-files/mysql.server
/etc/init.d/mysqld          #拷贝服务文件到、/etc/init.d 目录下,并将文件
名更改为 mysqld
[root@localhost ~]#chmod a+x /etc/init.d/mysqld
[root@localhost ~]#chkconfig --add /etc/init.d/mysqld
[root@localhost init.d]# chkconfig mysqld on
```


6、远程连接 mysql

在日常实际使用 MySQL 数据库的过程中，我们的 MySQL 数据库存储在自己的电脑上，这个时候就需要装有 mysql 服务的电脑就必须开启远程连接，才能让自己的点奥连接 MySQL 服务。方法如下：

添加 mysql 到防火墙允许列表中。

```
[root@localhost ~]# firewall-cmd --permanent --zone=public
--add-port=3306/tcp
[root@localhost ~]# firewall-cmd --reload
mysql> GRANT ALL ON . TO root@'%' IDENTIFIED BY '12345678' WITH GRANT
OPTION;          #开启 mysql 远程访问权限
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

项目二 数据库和数据表操作

任务 2-1 创建和删除数据库

一. 创建数据库

创建数据库语句 create database

语法格式:

```
create database [IF NOT EXISTS] 数据库名
```

```
[[DEFAULT] CHARACTER SET 字符集
```

```
[[DEFAULT] COLLATE 校对规则;
```

说明:

IF NOT EXISTS: 在创建数据库时会进行判断数据库是否存在, 只有在该数据库不存的情况才会创建;

CHARACTER SET :指定数据库采用的字符集;

COLLATE:指定数据库字符集的校对规则。

在创建数据库时默认字符集为: UTF8, 校对规则为: UTF8_general_ci(不区分大小写)

为了避免在数据库中存储的数据出现乱码的情况一般会在创建数据库时指定字符集, 如果不指定字符集, 那么就使用系统的默认字符集。

MySQL 的数据存储区是以目录方式来表示 MySQL 数据库, 因此数据库名称要符合操作系统的文件夹命名规则, 不能以数字开头, MySQL 数据库是不允许在同一系统下创建两个相同名称的数据库的。

二. 删除数据库

删除数据库语句 DROP DATABASE

语法格式:

```
DROP DATABASE 数据库名
```

注意: 执行此语句时一定要谨慎, 避免误删。

information_schema 和 mysql 这两个数据库是 MySQL 安装后, 系统自动创建的, 存放一些和数据库相关的信息, 如果删除了这两个数据库, MySQL 就不能正常工作了。

三. 范例:

1. 创建 test01 数据库。

```
mysql> CREATE DATABASE test01;
```