

## 什么是数据库

- MySQL 是最流行的关系型数据库管理系统，在WEB应用方面 MySQL 是最好的 RDBMS(Relational Database Management System: 关系数据库管理系统)应用软件之一。
- 数据库 (Database) 是按照数据结构来组织、存储和管理数据的仓库，每个数据库都有一个或多个不同的API用于创建，访问，管理，搜索和复制所保存的数据。
- 我们也可以将数据存储于文件中，但是在文件中读写数据速度相对较慢。所以，现在我们使用关系型数据库管理系统 (RDBMS) 来存储和管理的大数据量
- 所谓的关系型数据库，是建立在关系模型基础上的数据库，借助于集合代数等数学概念和方法来处理数据库中的数据。

## 关系型数据库的特点

- 数据以表格的形式出现
- 每行为各种记录名称
- 每列为记录名称所对应的数据域
- 许多的行和列组成一张表单
- 若干的表单组成database

## 关系型数据库的一些术语

- 数据库: 数据库是一些关联表的集合。
- 数据表: 表是数据的矩阵。在一个数据库中的表看起来像一个简单的电子表格。
- 列: 一列(数据元素) 包含了相同的数据, 例如邮政编码的数据。
- 行: 一行 (=元组, 或记录) 是一组相关的数据, 例如一条用户订阅的数据。
- 冗余: 存储两倍数据, ++冗余降低了性能++, 但提高了数据的安全性。
- 主键: 主键是唯一的。一个数据表中只能包含一个主键。你可以使用主键来查询数据。
- 外键: 外键用于关联两个表。
- 复合键: 复合键 (组合键) 将多个列作为一个索引键, 一般用于复合索引。
- 索引: 使用索引可快速访问数据库表中的特定信息。索引是对数据库表中一列或多列的值进行排序的一种结构。类似于书籍的目录。
- 参照完整性: 参照的完整性要求关系中不允许引用不存在的实体。与实体完整性是关系模型必须满足的完整性约束条件, 目的是保证数据的一致性。

## MySQL数据库

- 是一个关系型数据库管理系统，由瑞典MySQLAB公司开发，目前属于Oracle公司，它是一个关联型数据库，将数据保存在不同的表中，通过主外键或者复合键将各个数据表进行关联，而不是将所有数据放到一个大仓库中，增加了速度以及提高了灵活性。
- 开源，免费
- 可处理上千万记录的大学数据库
- 使用标准的SQL语句
- 支持多系统，多语言 (c、c++、python、java、Perl、PHP、eiffel等等)
- 对PHP有很好的支持，PHP是目前最流行的web开发语言
- 支持大型数据库，支持5000万条记录的数据仓库，32位系统表文件最大可支持4GB，64位系统支持最大的表文件为8TB。
- 可以定制的，采用了GPL协议，你可以修改源码来开发自己的 MySQL 系统。

## 其他知识

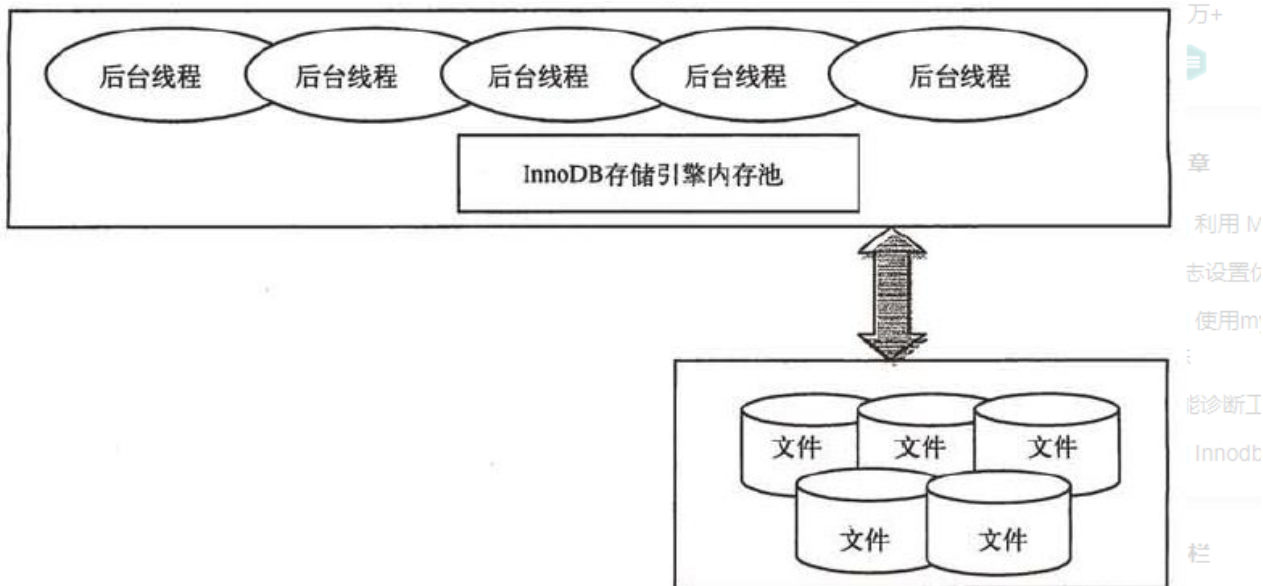
- 前端：只是展示内容，用户交互窗口，展示数据给用户，html,css,jquery都无法访问数据库，真正的 内容在数据库 php java c#可以访问数据库（DB）；
- 后台：接收前端的请求，访问数据库，获取数据模型，返回给前端
- 无论什么系统管理数据的方式 其底层都是 增 删 改 查。
- 数据库：（database）DB:是一个存储数据的仓库，实际上是安装在计算机上的一种服务，将数据按照特定的规律存储在硬盘上。存储东西
  - 保存数据
  - 管理数据（新增 删除 更新 查询数据）
  - 数据库管理系统：DBMS(database management system)
  - 数据库存储引擎：如何存储数据，如何为存储的数据建立索引，如何更新数据，查询数据的实现方式。MYSQL数据库支持的存储引擎：
    - show engines;
    - Engine :存储引擎名称
    - Support:MySQL数据库是否真支持该存储引擎
    - Comment:存储引擎的描述
    - Transactions:表示是否支持事务
    - XA： 表示是否支持分布式事务
    - Savepoints:是否支持保存点，以便事务回滚到保存点。
- MySQL 默认使用了**InnoDB存储引擎**，优势是提供了良好的事务管理能力，崩溃修复能力以及高并发。

### 关于MyISAM存储引擎。

MyISAM是默认存储引擎，它基于更老的ISAM代码，但有很多有用的扩展。（注意MySQL 5.1不支持ISAM）。每个MyISAM在磁盘上存储成三个文件。每一个文件的名字均以表的名字开始，扩展名指出文件类型。.frm文件存储表定义。数据文件的扩展名为MYD (MYData)。

### 关于InnoDB存储引擎

简单来说，InnoDB是由一系列后台线程和一大块内存组成。InnoDB是事务安全的存储引擎，设计上借鉴了很多Oracle的架构思想，一般而言，在OLTP应用中，InnoDB应该作为核心应用表的首先存储引擎。InnoDB是由第三方的Innobase Oy公司开发，现已被Oracle收购，创始人是Heikki Tuuri，芬兰赫尔辛基人，和著名的Linux创始人Linus是校友。

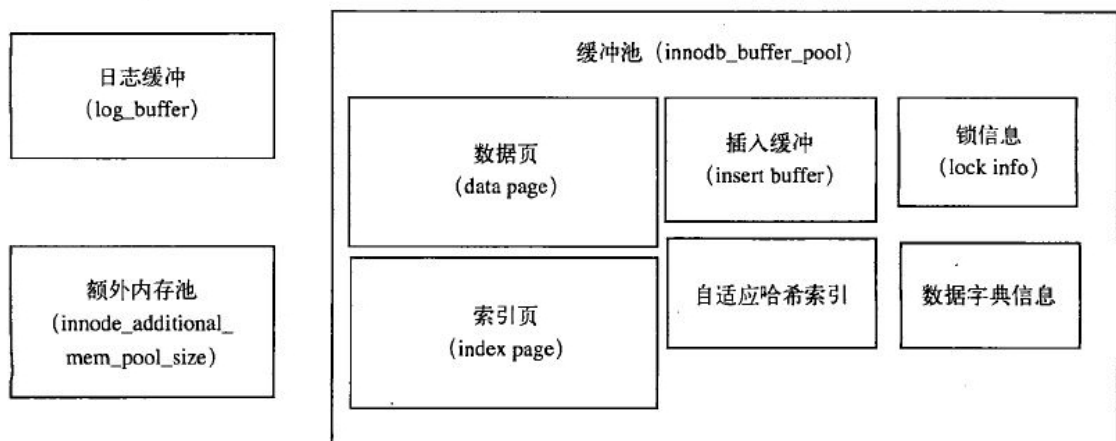


## 后台线程

- 默认情况下，InnoDB的后台线程有7个 —— 4个IO thread, 1个master thread, 1个lock monitor thread, 一个error monitor thread

## 内存

- InnoDB的内存主要有以下几个部分组成：缓冲池 (buffer pool)、重做日志缓冲池(redo log buffer) 以及额外的内存池 (additional memory pool)
- 其中缓冲池占最大块内存，用来缓存各自数据，++数据文件按页（每页16K）读取到缓冲池，按最近最少使用算法（LRU）保留缓存数据++。
- 缓冲池缓冲的数据类型有：数据页、索引页、插入缓冲、自适应哈希索引、锁信息、数据字典信息等，++其中数据页和索引页占了绝大部分内存++。



## Master 后台线程

InnoDB的主要工作都是在一个单独的Master线程里完成的。Master线程的优先级最高，它主要分为以下几个循环：主循环（loop）、后台循环（background loop）、刷新循环（flush loop）、暂停循环（suspend loop）。

- 其中每秒一次的操作包括：
  - 刷新日志缓冲区（总是）
  - 合并插入缓冲（可能）

- 至多刷新100个脏数据页（可能）
- 如果没有当前用户活动，切换至background loop（可能）

和Oracle类似，即使事务未提交，也会每秒刷新重做日志缓冲区。

其中每10秒一次的操作包括：

- 合并至多5个插入缓冲（总是）
- 刷新日志缓冲（总是）
- 刷新100个或10个脏页到磁盘（总是）
- 产生一个检查点（总是）
- 删除无用Undo 页（总是）

接着来看后台循环，若当前没有用户活动或数据库关闭时，会切换至该循环执行以下操作：

- 删除无用的undo页（总是）
- 合并20个插入缓冲（总是）
- 跳回到主循环（总是）
- 不断刷新100个页，直到符合条件跳转到flush loop（可能）

如果flush loop中也没有什么事情可做，就切换到suspend loop，将master线程挂起。