# Mysql数据库

## 概述

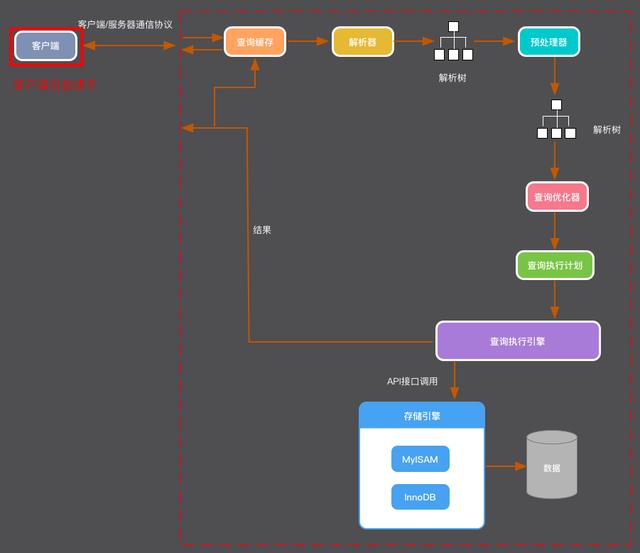
MySQL是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

## 特点

* Mysql是开源的，所以你不需要支付额外的费用。
* Mysql支持大型的数据库。可以处理拥有上千万条记录的大型数据库。
* MySQL使用标准的SQL数据语言形式。
* Mysql可以允许于多个系统上，并且支持多种语言。这些编程语言包括C、C++、Python、Java、Perl、PHP、Eiffel、Ruby和Tcl等。
* Mysql对PHP有很好的支持，PHP是目前最流行的Web开发语言。
* MySQL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。
* Mysql是可以定制的，采用了GPL协议，你可以修改源码来开发自己的Mysql系统。

# ****MySQL的执行流程****

## 执行流程图



## 十个执行步骤

1.当我们请求mysql服务器的时候,MySQL前端会有一个监听,请求到了之后,服务器得到相关的SQL语句,执行之前(虚线部分为执行),还会做权限的判断

2.通过权限之后,SQL就到MySQL内部,他会在查询缓存中,看该SQL有没有执行过,如果有查询过,则把缓存结果返回,说明在MySQL内部,也有一个查询缓存.但是这个查询缓存,默认是不开启的,这个查询缓存,和我们的Hibernate，Mybatis的查询缓存是一样的,因为查询缓存要求SQL和参数都要一样,****所以这个命中率是非常低的（没什么卵用的意思）。****

3.如果我们没有开启查询缓存,或者缓存中没有找到对应的结果,那么就到了解析器,解析器主要对SQL语法进行解析

4.解析结束后就变成一颗解析树,这个解析树其实在Hibernate里面也是有的,大家回忆一下,在以前做过Hibernate项目的时候,是不是有个一个antlr.jar。这个就是专门做语法解析的工具.因为在Hibernate里面有HQL，它就是通过这个工具转换成SQL的,我们编程语言之所以有很多规范、语法,其实就是为了便于这个解析器解析,这个学过编译原理的应该知道.

5.得到解析树之后,不能马上执行,这还需要对这棵树进行预处理,也就是说,这棵树,我没有经过任何优化的树,预处理器会这这棵树进行一些预处理,比如常量放在什么地方,如果有计算的东西,把计算的结果算出来等等...

6.预处理完毕之后,此时得到一棵比较规范的树,这棵树就是要拿去马上做执行的树,比起之前的那棵树,这棵得到了一些优化

7.****查询优化器，是MySQL里面最关键的东西****,我们写任何一条SQL,比如SELECT \* FROM USER WHERE USERNAME = toby AND PASSWORD = 1,它会怎么去执行?它是先执行username = toby还是password = 1?每一条SQL的执行顺序查询优化器就是根据MySQL对数据统计表的一些信息,比如索引,比如表一共有多少数据,MySQL都是有缓存起来的,在真正执行SQL之前,他会根据自己的这些数据,进行一个综合的判定,判断这一次在多种执行方式里面,到底选哪一种执行方式,可能运行的最快.这一步是MySQL性能中,最关键的核心点,也是我们的优化原则.****我们平时所讲的优化SQL,其实说白了,就是想让查询优化器,按照我们的想法,帮我们选择最优的执行方案,因为我们比MySQL更懂我们的数据.MySQL看数据,仅仅只是自己收集到的信息,这些信息可能是不准确的,MySQL根据这些信息选了一个它自认为最优的方案,但是这个方案可能和我们想象的不一样.****

8.这里的查询执行计划,也就是MySQL查询中的执行计划,比如要先执行username = toby还是password = 1

9.这个执行计划会传给查询执行引擎,执行引擎选择存储引擎来执行这一份传过来的计划,到磁盘中的文件中去查询,这个时候重点来了,影响这个查询性能最根本的原因是什么?就是硬盘的机械运动,也就是我们平时熟悉的IO,所以一条查询语句是快还是慢,就是根据这个时间的IO来确定的.那怎么执行IO又是什么来确定的?就是传过来的这一份执行计划.

10.如果开了查询缓存,则返回结果给客户端,并且查询缓存也放一份。