

MySQL逻辑架构

MySQL逻辑架构图

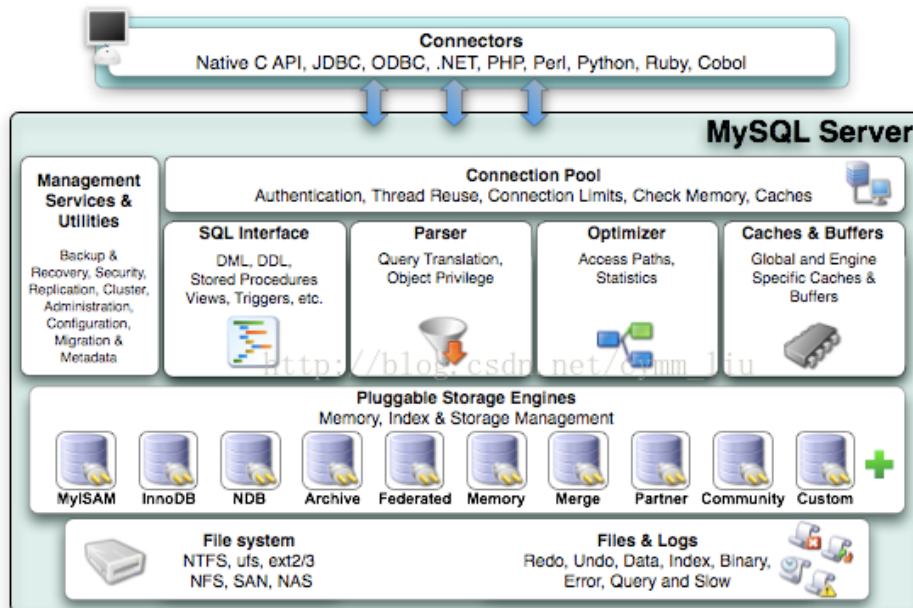


图2

MySQL逻辑架构分四层

1.连接层: 主要完成一些类似连接处理, 授权认证及相关的安全方案。

2.服务层: 在MySQL数据库系统处理底层数据之前的所有工作都是在这一层完成的, 包括权限判断, SQL接口, SQL解析, SQL分析优化, 缓存查询的处理以及部分内置函数执行(如日期,时间,数学运算,加密)等等。各个存储引擎提供的功能都集中在这一层, 如存储过程, 触发器, 视图等。

3.引擎层: 是底层数据存取操作实现部分, 由多种存储引擎共同组成。真正负责MySQL中数据的存储和提取。就像Linux众多的文件系统一样。每个存储引擎都有自己的优点和缺陷。服务器是通过存储引擎API来与它们交互的。这个接口隐藏了各个存储引擎不同的地方。对于查询层尽可能的透明。这个API包含了很多底层的操作。如开始一个事物, 或者取出有特定主键的行。存储引擎不能解析SQL, 互相之间也不能通信。仅仅是简单的响应服务器的请求。

4.存储层: 将数据存储于裸设备的文件系统之上, 完成与存储引擎的交互。

1.连接层

最上层是一些客户端和连接服务, 包含本地sock通信和大多数基于客户端/服务端工具实现的类似于tcp/ip的通信。主要完成一些类似于连接、授权认证、及相关的安全方案。在该层上引入了线程池的概念, 为通过认证安全接入的客户端提供线程。同样在该层上可以实现基于SSL的安全链接。服务器也会为安全接入的每个客户端验证它所具有的操作权限。

2.服务层

第二层架构主要完成大多数的核心服务功能, 如SQL接口, 并完成缓存的查询, SQL的分析和优化及部分内置函数的执行。所有跨存储引擎的功能也在这一层实现, 如过程、函数等。在该层, 服务器会解析查询并创建相应的内部解析树, 并对其完成相应的优化如确定查询表的顺序是否利用索引等, 最后生成相应的执行操作。如果是select语句, 服务器还会查询内部的缓存。如果缓存空间足够大, 这样在解决大量读环境中能够很好的提升系统的性能。

3.引擎层

存储引擎层, 存储引擎真正的负责了MySQL中数据的存储和提取, 服务器通过API与存储引擎进行通信。不同的存储引擎具有的功能不同, 我们可以根据自己的实际需要进行选取。后面介绍MyISAM和InnoDB

4.存储层

数据存储层, 主要是将数据存储运行于裸设备的文件系统之上, 并完成与存储引擎的交互。

<https://www.cnblogs.com/ChangAn223/p/10686639.html>
<https://www.bilibili.com/video/BV12b411k7Zu?p=186>