

0 思维导图

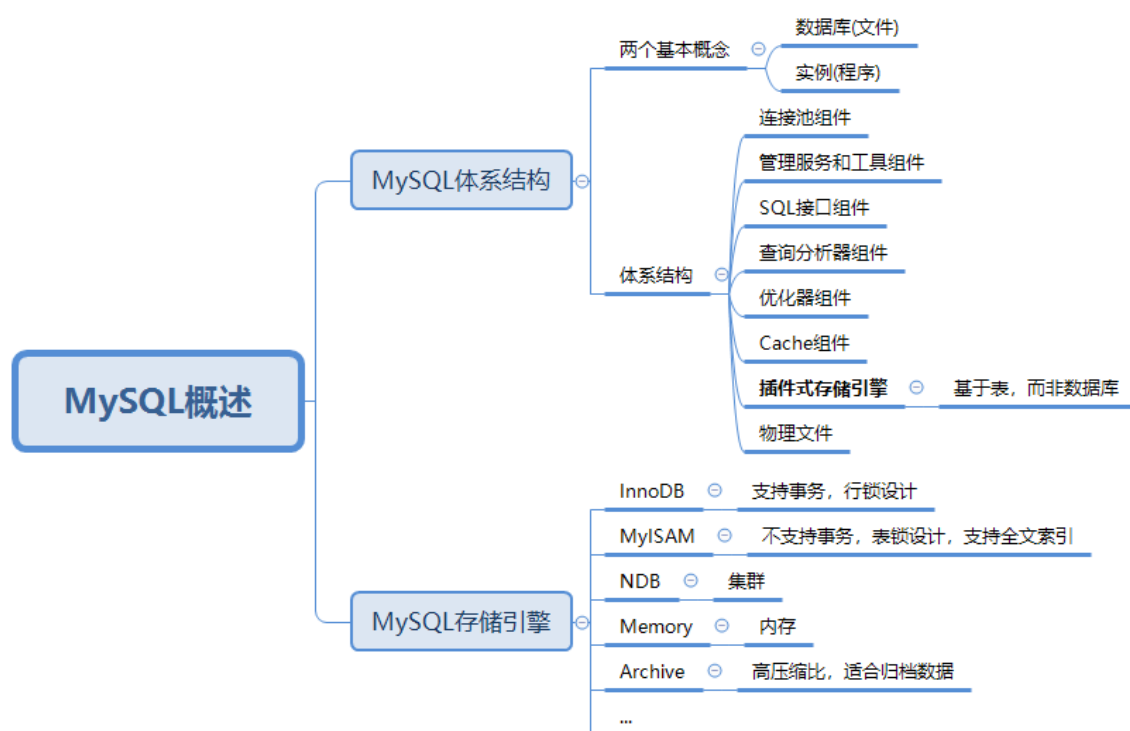
1 MySQL 体系结构概述

1.1 MySQL 体系结构图

1.2 MySQL 存储引擎

1.3 连接MySQL

0 思维导图



1 MySQL 体系结构概述

搞清楚数据库与实例的术语：

- 数据库：物理操作系统文件或其他形式文件类型的集合。
- 实例：MySQL数据库由后台线程以及一个共享内存区组成。MySQL数据库实例在系统上的表现就是一个进程。

直白点说就是，数据库是文件的集合，是依照某种数据模型组织起来并存放于二级存储器中的数据集合；数据库实例是程序，是位于操作系统之间的一层数据管理软件，用户对数据库数据的任何操作(比如数据库定义、查询、维护、运行控

制)等都是在数据库实例下进行的，应用程序必需通过数据库实例才能和数据库打交道。

1.1 MySQL 体系结构图

MySQL由以下几部分组成：连接池组件、管理服务和工具组件、SQL接口组件、查询分析器、优化器组件、Cache组件、插件式存储引擎、物理文件
存储引擎负责底层物理结构的实现，基于表，而非数据库

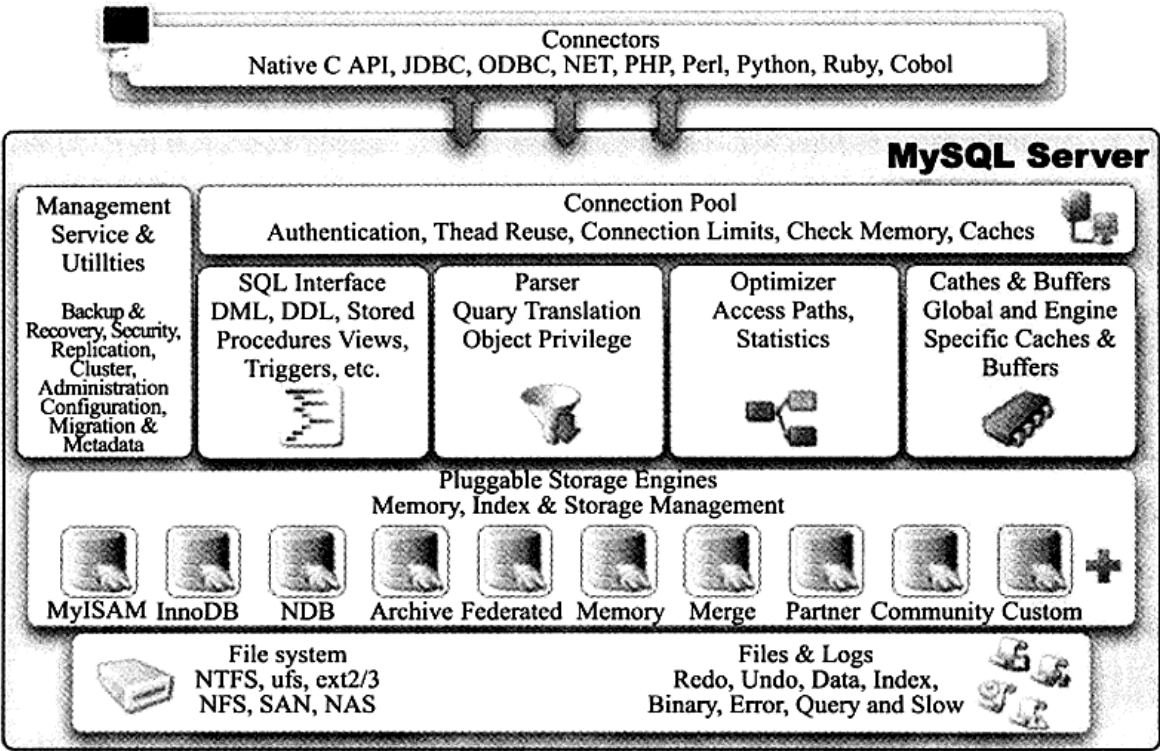


图 1-1 MySQL 体系结构

1.2 MySQL 存储引擎

MySQL是插件式的存储引擎，不同的存储引擎有其各自适合的场景。

SHOW ENGINES 查看数据库支持的引擎

```
mysql> show engines;
```

Engine	Support	Comment	Transactions	XA	Savepoints
FEDERATED	NO	Federated MySQL storage engine	NULL	NULL	NULL
MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
MyISAM	YES	MyISAM storage engine	NO	NO	NO
BLACKHOLE	YES	/dev/null storage engine (anything you write to it disappears)	NO	NO	NO
CSV	YES	CSV storage engine	NO	NO	NO
MEMORY	YES	Hash based, stored in memory, useful for temporary tables	NO	NO	NO
ARCHIVE	YES	Archive storage engine	NO	NO	NO
PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO
InnoDB	DEFAULT	Supports transactions, row-level locking, and foreign keys	YES	YES	YES

9 rows in set (0.07 sec)

1.3 连接MySQL

本质上是一个进程与MySQL进程的通讯。常用的进程通信方式：管道、命名管道、命名套接字、TCP/IP套接字、UNIX域套接字。

- TCP/IP

最常使用的 `mysql -h127.0.0.1 -u root -p` 便是这种方式。

这种方式在TCP/IP连接上建立一个基于网络的连接请求，一般Client和server在不同的机器上，两台机器通过TCP/IP网络连接。Client发送请求后，mysql会去mysql.user表中检查权限。

- UNIX域套接字

这种情况只能Client和Server在同一台linux/unix 机器上才可以。配置文件 `--socket=/tmp/mysql.sock` 指定套接字文件路径。show variables like 'socket' 可以查看配置地址

```
mysql> show variables like 'socket';
+-----+-----+
| Variable_name | Value                               |
+-----+-----+
| socket        | /data/mysql/mysql.sock            |
+-----+-----+
1 row in set (0.05 sec)
```

知道了套接字信息后可以通过，`mysql -uroot -S /tmp/mysql.sock`进行连接。