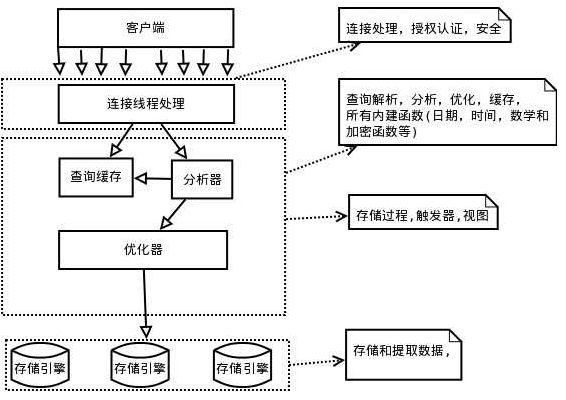
第一章 MySQL架构与历史

存储引擎架构：

MySQL最重要、最与众不同的特性，它可以使的查询处理与其它的系统任务和数据的存储/提取相分离。



1. 连接管理与安全性

* 每个客户端与MySQL服务器的连接都会有一个线程。
* 服务器会负责缓存线程，因此不需要每次连接都创建或销毁线程。

2. 优化与执行

查询过程：

（1） 首先检查查询缓存中是否能找到对应查询。

（2） 如果可以找到则直接执行。

（3） 如果找不到则，MySQL会解析查询，并创建内部数据结构（解析树）。

（4） 然后对其进行各种优化，包括重写查询、决定表的读取顺序、选择合适的索引等。

（5） 执行查询语句。

注：

JDBC中PreparedStatement与Statement？

3. 锁

（1）读写锁

**·** 共享锁（读锁）：加读锁时，其他线程可以读同一资源，但是不可以写。

**·** 排他锁（写锁）：加写锁时，其它线程不能读也不能写。

（2）锁粒度

**·** 表级锁：

优点：开销小。缺点：因为锁住整张表所以并发较差。

**·** 行级锁：

优点：可以最大程度的支持并发处理。

缺点：带来了最大的锁开销。

4. 事务

（1） ACID

**·** 原子性

要么全部执行成功，要么全部执行失败。

**·** 一致性

事务执行前后，数据保持一致的状态。例如，转账不管成功与否，总价钱不变。

**·** 隔离性

通常来说，一个事务所做的修改在最终提交以前，对其它事务是不可见的。

**·** 持久性

一旦事务提交，则其所做的修改就会永久保存到数据库中。

（2）隔离级别

· READ UNCOMMITTED（未提交读）

事务可以读取未提交的数据，也就是脏读。

也就是：

我修改数据的时候你也能读。

· READ COMMITTED（已提交读）

其它事务不能看到某个事务未提交的修改。

但是可能会导致不可重复读，两次同样的查询结果不一致。

也就是：

我在读的时候你可以修改数据，但是修改数据不能读。

· REPEATABLE READ（可重复读）

也就是：

我在读的时候你不能修改数据，写的时候也不能改。但是对数据的新增与删除不影响，所以会出现幻读。

· SERIALIZABLE（序列化）

强制事务串行执行。

完全锁住：不能做任何事。



5. 死锁

死锁是指两个事务持有对方需要的锁并且互相等待对方释放锁的恶性循环。

6. MySQL中的事务

支持事务的引擎：InnoDB存储引擎，NDB Cluster；第三方引擎包括XtraDB、PBXT

自动提交：

MySQL默认采用自动提交，也就是说，如果不是显示地开始一个事务，那么默认每个查询都被当做一个事务执行提交操作。

在事务中使用混合存储引擎：

回滚事务地时候会出现问题，非事务型地表上的变更无法撤销。

7. 显示锁与隐式锁

InnoDB采用的是二阶段锁定协议。在事务执行中可以随时锁定，即为显示锁。

锁只有在执行commit或者rollback的时候才会被释放。

* select … lock in share mode
* select … for update

8. 多版本并发控制

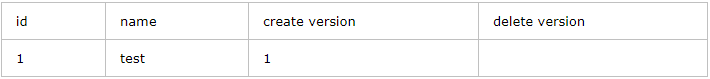
在每一行记录的后面增加两个隐藏列，记录创建版本号和删除版本号，

而每一个事务在启动的时候，都有一个唯一的递增的版本号。

（1）插入

在插入操作时 ： 记录的创建版本号就是事务版本号。

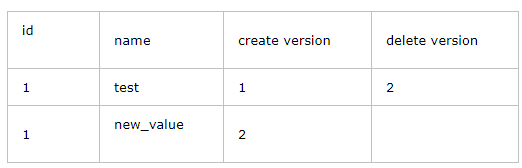
比如我插入一条记录, 事务id 假设是1 ，那么记录如下：也就是说，创建版本号就是事务版本号。



（2）更新

在更新操作的时候，采用的是先标记旧的那行记录为已删除，并且删除版本号是事务版本号，然后插入一行新的记录的方式。

比如，针对上面那行记录，事务Id为2 要把name字段更新



（3）删除

删除操作的时候，就把事务版本号作为删除版本号。比如

delete from table where id=1;



（4）查询

从上面的描述可以看到，在查询时要符合以下两个条件的记录才能被事务查询出来：

1) 删除版本号 大于 当前事务版本号，就是说删除操作是在当前事务启动之后做的。

2) 创建版本号 小于或者等于 当前事务版本号 ，就是说记录创建是在事务中（等于的情况）或者事务启动之前。

这样就保证了各个事务互不影响。从这里也可以体会到一种提高系统性能的思路，就是：

通过版本号来减少锁的争用。

另外，只有read-committed和 repeatable-read 两种事务隔离级别才能使用mvcc

read-uncommited由于是读到未提交的，所以不存在版本的问题

而serializable 则会对所有读取的行加锁。

9. MySQL的存储引擎

（1）InnoDB（默认存储引擎）（如无意外，尽量用InnoDB）

索引类型：聚簇索引

事务：支持事务

锁：行级锁、表级锁

（2）MyISAM（旧版默认）

事务：不支持

锁：表级锁

（3）Memory

特性：数据保存在内存中（类似于redis）