

# Mysql 技术内幕：InnoDB 学习（3，4）

jask

2024-08-11

## Mysql 技术内幕

### 文件

日志文件记录了影响 MySQL 数据库的各种类型活动。MySQL 数据库中常见的日志文件有：

错误日志 (error log)

二进制日志 (binlog)

慢查询日志 (slow query log)

查询日志 (log)

### 二进制日志

恢复 (recovery)：某些数据的恢复需要二进制日志，例如，在一个数据库全备文件恢复后，用户可以通过二进制日志进行 point-in-time 的恢复。

复制 (replication)：其原理与恢复类似，通过复制和执行二进制日志使一台远程的 MySQL 数据库（一般称为 slave 或 standby）与一台 MySQL 数据库（一般称为 master 或 primary）进行实时同步。

审计 (audit)：用户可以通过二进制日志中的信息来进行审计，判断是否有对数据库进行注入的攻击。

### InnoDB 存储引擎文件

InnoDB 采用将存储的数据按表空间 (tablespace) 进行存放的设计。

每个 InnoDB 存储引擎至少有 1 个重做日志文件组 (group)，每个文件组下至少有 2 个重做日志文件，如默认的 ib\_logfile0 和 ib\_logfile1。

### 表

在 InnoDB 存储引擎中，表都是根据主键顺序组织存放的，这种存储方式的表称为索引组织表 (index organized table)。在 InnoDB 存储引擎表中，每张表都有个主键 (Primary

Key)，如果在创建表时没有显式地定义主键，则 InnoDB 存储引擎会按如下方式选择或创建主键：

首先判断表中是否有非空的唯一索引 (Unique NOT NULL)，如果有，则该列即为主键。

如果不符合上述条件，InnoDB 存储引擎自动创建一个 6 字节大小的指针。

### 逻辑存储结构

从 InnoDB 存储引擎的逻辑存储结构看，所有数据都被逻辑地存放在一个空间中，称之为表空间 (tablespace)。表空间又由段 (segment)、区 (extent)、页 (page) 组成。

**表空间** 表空间可以看做是 InnoDB 存储引擎逻辑结构的最高层，所有的数据都存放在表空间中。第 3 章中已经介绍了在默认情况下 InnoDB 存储引擎有一个共享表空间 `ibdata1`，即所有数据都存放在这个表空间内。如果用户启用了参数 `innodb_file_per_table`，则每张表内的数据可以单独放到一个表空间内。

### Compact 行记录

Compact 行记录格式的首部是一个非 NULL 变长字段长度列表，并且其是按照列的顺序逆序放置的，其长度为：

若列的长度小于 255 字节，用 1 字节表示；

若大于 255 个字节，用 2 字节表示。

### InnoDB 数据页结构

InnoDB 数据页由以下 7 个部分组成：File Header, Page Header, Infimum 和 Supremum Records, User Records, Free Space, Page Directories, File Trailer

### 约束

在 InnoDB 存储引擎表中，域完整性可以通过以下几种途径来保证：

选择合适的数据类型确保一个数据值满足特定条件。

外键 (Foreign Key) 约束。

编写触发器。

还可以考虑用 DEFAULT 约束作为强制域完整性的一个方面。

**ENUM 和 SET 约束** MySQL 数据库不支持传统的 CHECK 约束，但是通过 ENUM 和 SET 类型可以解决部分这样的约束需求。例如表上有一个性别类型，规定域的范围只能是 male 或 female，在这种情况下用户可以通过 ENUM 类型来进行约束。

## 分区

对于 OLAP(在线分析业务) 的应用, 分区的确是可以很好地提高查询的性能, 因为 OLAP 应用大多数查询需要频繁地扫描一张很大的表。假设有一张 1 亿行的表, 其中有一个时间戳属性列。用户的查询需要从这张表中获取一年的数据。如果按时间戳进行分区, 则只需要扫描相应的分区即可。这就是前面介绍的 Partition Pruning 技术。

**在表和分区之间交换数据** MySQL 5.6 开始支持 ALTER TABLE...EXCHANGE PARTITION 语法。该语句允许分区或子分区中的数据与另一个非分区的表中的数据进行交换。如果非分区表中的数据为空, 那么相当于将分区中的数据移动到非分区表中。若分区表中的数据为空, 则相当于将外部表中的数据导入到分区中。

要使用 ALTER TABLE...EXCHANGE PARTITION 语句, 必须满足下面的条件:

要交换的表需和分区表有着相同的表结构, 但是表不能含有分区

在非分区表中的数据必须在交换的分区定义内

被交换的表中不能含有外键, 或者其他的表含有对该表的外键引用

用户除了需要 ALTER、INSERT 和 CREATE 权限外, 还需要 DROP 的权限

此外, 有两个小的细节需要注意:

使用该语句时, 不会触发交换表和被交换表上的触发器

AUTO\_INCREMENT 列将被重置

MySQL 数据库支持 RANGE、LIST、HASH、KEY、COLUMNS 分区, 并且可以使用 HASH 或 KEY 来进行子分区。需要注意的是, 分区并不总是适合于 OLTP 应用。