# 背景

在使用MySQL数据库的过程当中，以下两种场景是大家经常遇到的：

1. 随着业务体量的增大，MySQL数据库可用空间越来越小。解决方法通常有：
2. 扩容，利用更大的磁盘空间扩容，或者使用分布式架构；
3. 清理不用的数据，尽量不用delete操作以减少数据碎片的产生。

第二，优化MySQL时可能发现表中数据只有几千行，即使是全表扫的操作，也与实际打印出来的MySQL的执行时间相差很远。

这两种情况都与MySQL表空间碎片化有关。那么我们如何优化？

# 概述

例如，每当删除一行内容，该段空间就会变为空白，被留空。而在一段时间内的大量删除，会使留空的空间变得比存储列表内容所使用的空间更大。当执行插入撮作时，MySQL会尝试使用空白空间。但如果某个空白空间一直没有被大小合适的数据占用，就形成了碎片。当MySQL对进行扫描时，扫描的对象实际是列表的容量需求上限，碎片越多，就会影响查询的性能。

查询数据库中大于磁盘碎片大于0的表 ：

SELECT TABLE\_SCHEMA DB, TABLE\_NAME,DATA\_FREE,ENGINE FROM information\_schema.TABLES WHERE TABLE\_SCHEMA NOT IN ('INFORMATION\_SCHEMA','MYSQL') AND DATA\_FREE>0;

# 优化

首先查看某个表的碎片大小，并列出所有已经产生碎片的表。对于优化表碎片，不同数据库引擎有不同的方式。例如MylSAM表，通过optimize table表名的方式；lnnoDB表，通过alter table表名engine=lnnoDB的方式进行优化。

引申一下optimize table的操作，在MySQL5.5和5.6中，实际上执行的过程是，首先创建一张新的临时表，把旧表锁住，禁止插入删除只允许读写，接着把数据不断从旧表拷贝到新临时表，拷贝完成后，进行瞬间rename操作，再删除旧表。

因为清除碎片操作会暂时锁表，数据量越大，耗费时间越长。建议：

1. 清除操作首先使用percona的工具进行在线优化。
2. 清除碎片定期选在业务低峰时进行。

## MyISAM

OPTIMIZE TABLE data.emp;

select TABLE\_NAME,ENGINE,data\_length,data\_free FROM TABLES WHERE TABLE\_NAME='emp';

## InnoDB

OPTIMIZE TABLE data.emp;

在共享表空间下，对表做OPTIMIZE TABLE能够使之前因delete浪费的空间回收来重用，但是ibdata并不会收缩。

默认情况下innodb\_file\_per\_table=0 是共享表空间，如果是这种共享表空间物理磁盘释放不了，但是碎片整理后空间可以重复利用，如果不能共享表空间即innodb\_file\_per\_table=1 每个表都有独立的表空间，碎片清理后可以减少物理磁盘的占用。

# 应用