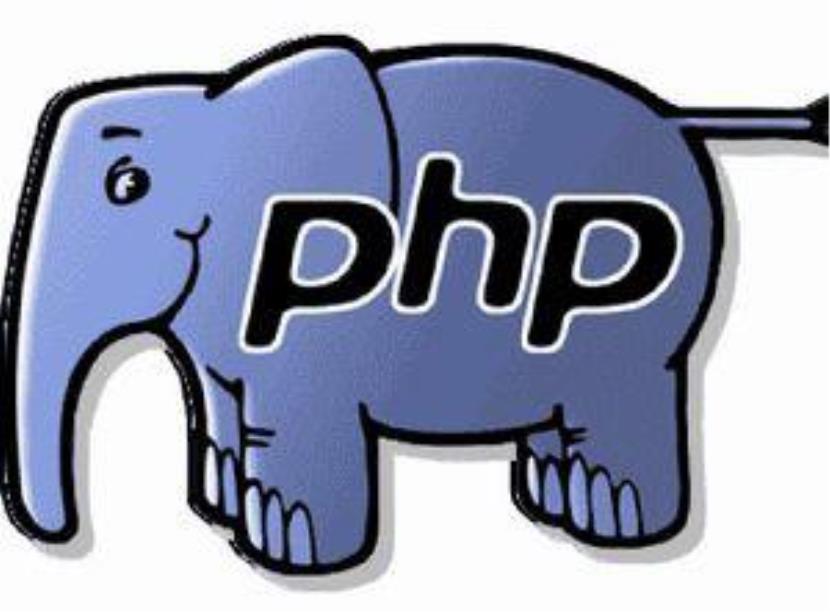
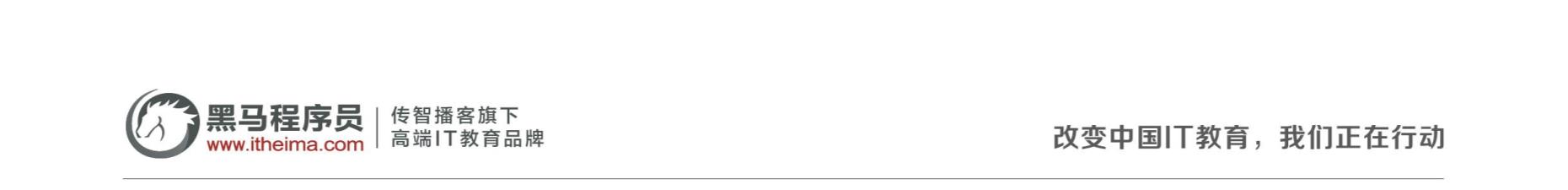
MySQL 优化专题拓展

上海校区就业部 PHP 组——薛杰

2017 年 7 月

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



一、SQL 优化

1、分析和定位策略

1、通过 show status 了解各种 SQL 的执行频率

2、定位执行效率低的 SQL 语句：①通过慢日志定位；②使用 show processlist 命令查看当前

在进行的线程

3、通过 explain 分析低效 SQL

4、通过 show profile 分析 SQL

5、通过 trace 分析优化器的选择

2、优化

1、大批量插入数据，使用如下方式能快速导入大量数据（在 myisam 引擎下速度更为显著）：

alter table 表名 disable keys;

Loading the data

Alter table 表名 enable keys;

2、优化 insert 语句：

①对同一客户插入很多行，尽量使用多个值表的 insert 语句，能大大缩减客户端与数据

库之间的连接。例如 insert into 表名 values(1,2),(1,3),(1,4)····

②对不同客户插入很多行，可以使用 insert delayed 语句更高速

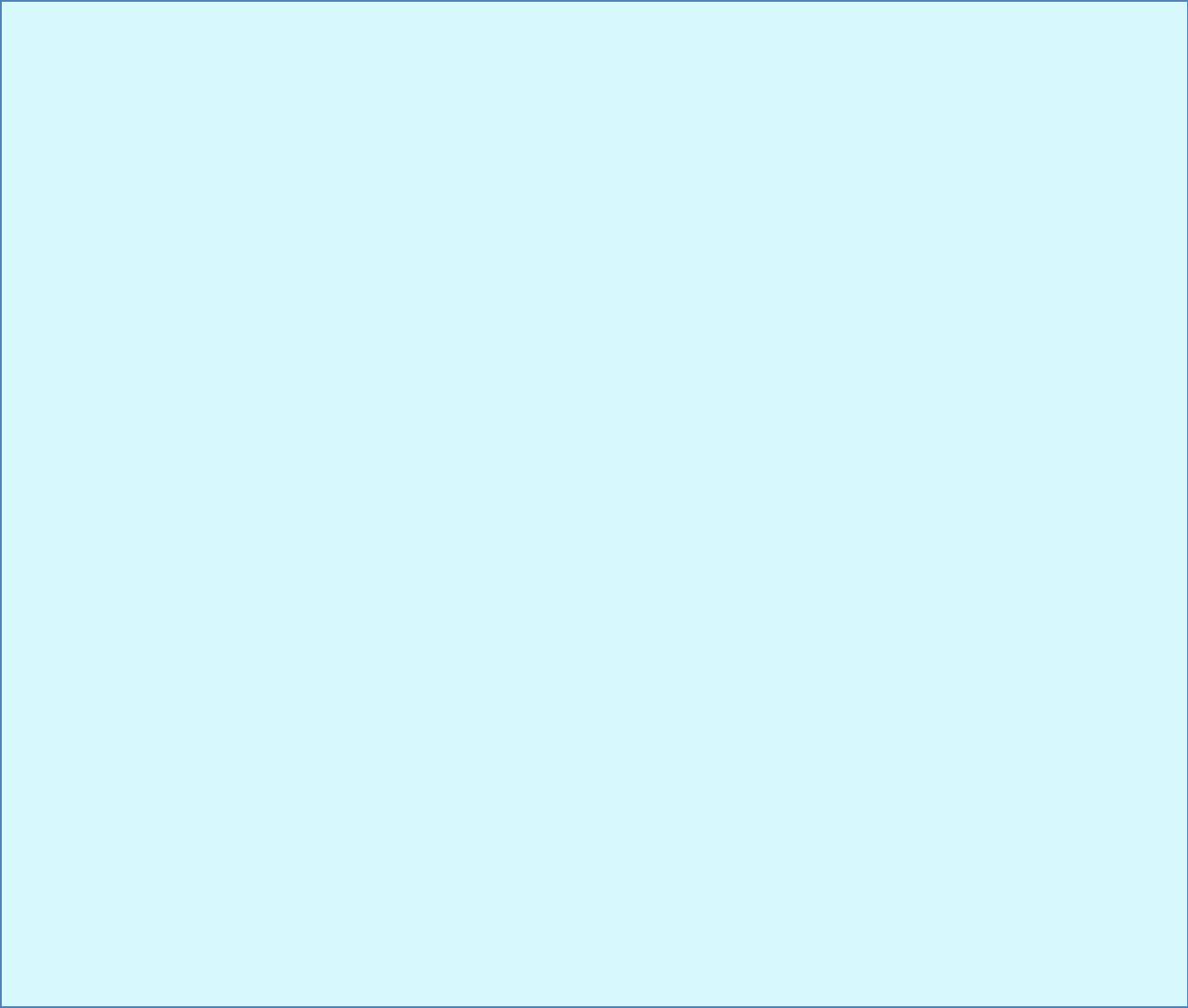
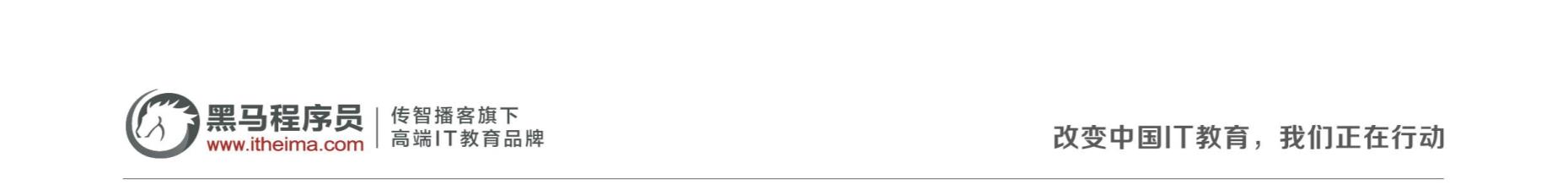
③将索引文件和数据文件分在不同的磁盘上存放

④从一个文本文件装载一个表时，使用 load data infile，速度会比 insert 快 20 倍左右

3、优化 order by 语句：

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



①mysql 中的两种排序方式：第一种是有序索引顺序扫描直接返回有序数据；第二种是

通过对返回数据进行排序（Filesort 排序）

②Filesort 优化：通过两次扫描算法和一次扫描算法

4、优化 group by 语句：使用 group by null 可以避免用户排序结果的消耗

5、优化嵌套查询：子查询可以一次性完成多个步骤查询，同时可以避免事务或者是表死锁。

但是在有些情况下，连接查询可以代替子查询

6、优化 or 条件：正确使用 or 条件查询，当时在对有独立索引的列查询时 or 操作能够快速

查找到结果；而在对有复合索引的列上做 or 操作时，却不能用到索引

7、优化分页查询：

①思路一：在索引上完成排序分页的操作，最后根据主键关联回原表查询所需要的其他

列内容，这种方式让 mysql 扫描尽可能少的页面来提高分页效率

②思路二：把 limit 查询转换成某个位置查询。这样把 limit m,n 转换成 limit n 只适合在

排序字段不会出现重复值的环境下

8、使用 SQL 提示：

①use index：在查询语句中表名的后面，添加 use index 来提供 mysql 去参考的索引列

表，让 mysql 不再去考虑其他索引

②ignore index：使用 ignore index 可以让 mysql 忽略一个或者多个索引

③force index：强制 mysql 使用一个特定的索引，一定情况下可以避免全秒扫描

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



PS：索引问题

1、索引的存储分类

①B-Tree 索引：最常见的索引，大部分引擎都支持

②Hash 索引：只有 memory 引擎支持

③R-Tree 索引（空间索引）：使用很少，只做了解

④Full-text（全文索引）：主要用于全文索引，innodb 从 mysql5.6 开始提供支持

2、Mysql 中使用索引的场景

①匹配全值：对索引中所有列都指定具体值，即对索引中的所有列都有等值匹配的条件

例：select \* from rental where rental\_date=’2017-06-27 17:40:59’ and customer\_id=343;

②匹配值的范围查询：对索引的值能够进行范围查找

例：select \* from rental where customer\_id>=373 and customer\_id<400;

③匹配最左前缀：仅仅使用索引中的最左边列进行查找。

例：在 A+B+C 字段上的联合索引能够被 A、A+B、A+B+C 的等值查询利用，但是不能被

B、B+C 的等值查询利用到。

④仅仅对索引进行查询，档查询的列都在索引的字段中时，查询的效率更高

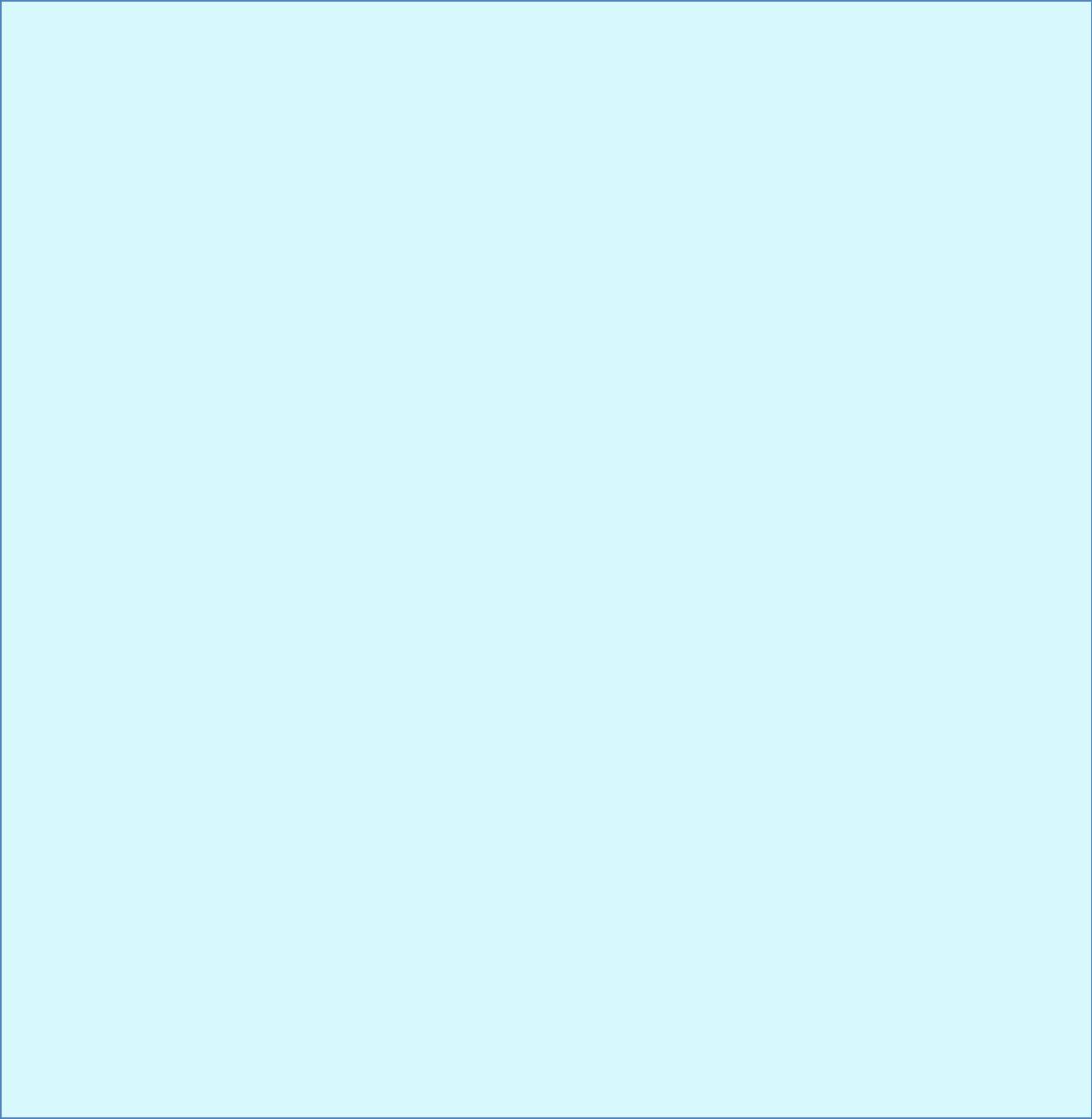
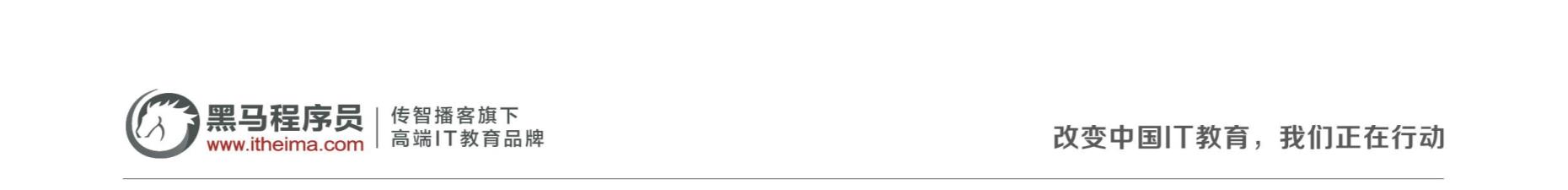
⑤匹配列前缀：仅仅使用索引中的第一列，并且只包含索引第一列的开头一部分进行查

找

⑥索引匹配部分精确而其他部分进行范围匹配：

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



例：select a\_id from rental where rental\_date=’2006-02-14’ and customer\_id >=300 and

customer\_id < 400

⑦如果列名是索引，那么使用 column\_name is null 就会使用索引

3、存在索引但不能使用索引的场景

①以%开头的 Like 查询不能够利用索引

②数据类型出现隐式转换的时候也不会使用索引，特别是当列类型是字符串，一定要在

where 条件中把字符串常量值用引号引起来，否则即便是有索引也是无效的

③复合索引的情况下，假如查询条件不满足最左原则，是不会使用索引的

④如果 mysql 估计使用索引比全表扫描更慢，则不使用索引

⑤用 or 分割开的条件，如果 or 前的条件中的列有索引，而后面的列中没有索引，那么

涉及的索引都不会被用到

二、优化数据库对象

1、优化表的数据类型：表需要使用何种数据类型是需要根据应用来判断的。可以使用函数

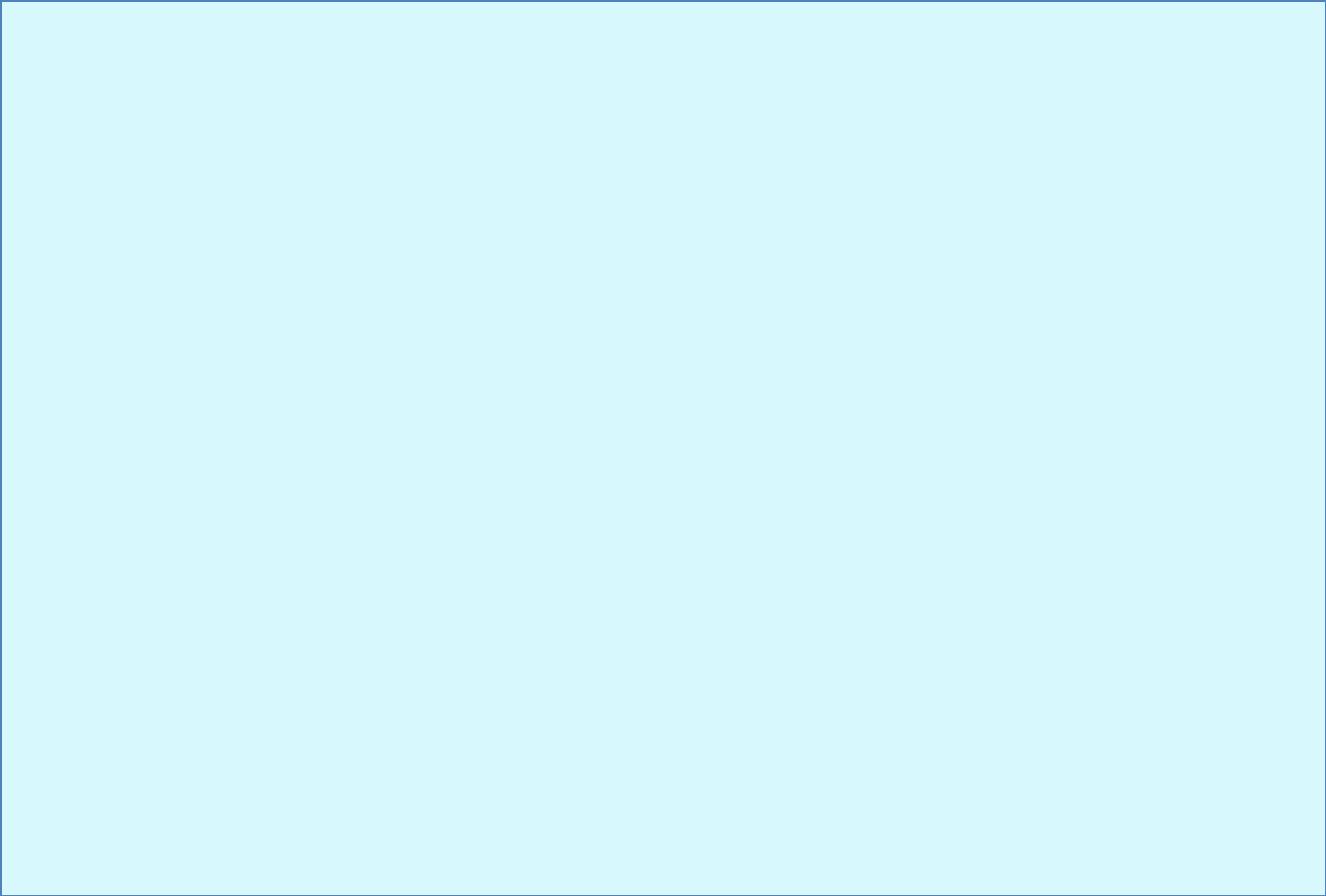
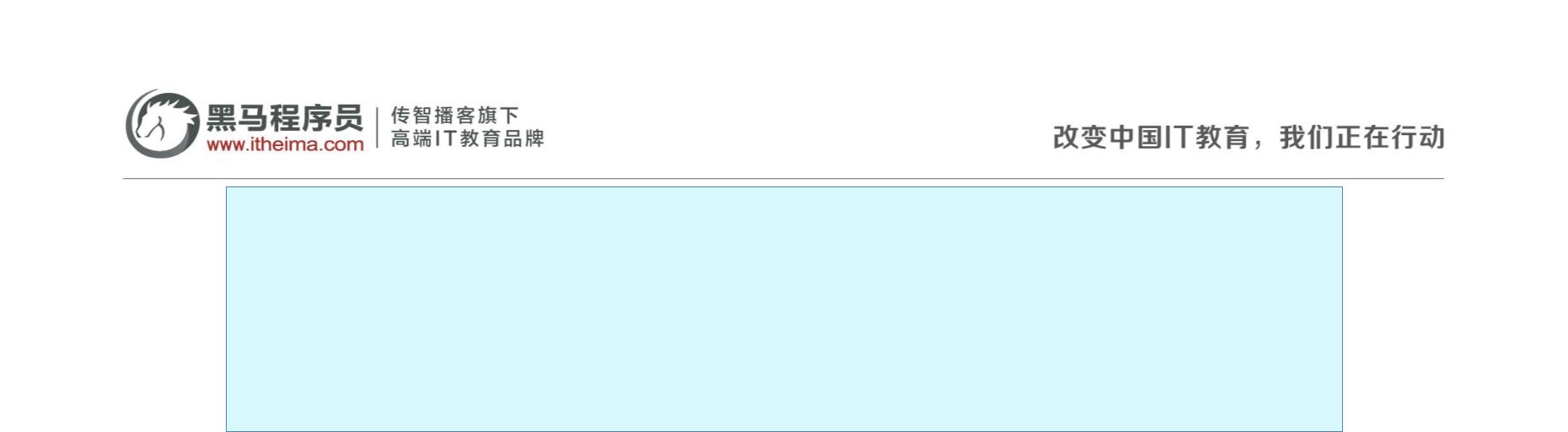
procedure analyse()对当前应用的表进行分析，可以根据当前表提出优化建议

2、通过拆分提高表的访问效率：

①垂直拆分：把主码和一些列放到一个表，然后把主码和另外的列放到另一个表中

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



②水平拆分：根据一列或多列的值把数据行放到两个独立的表中

3、使用中间表提高统计查询速度：

①中间表复制源表部分数据，并且与源表相“隔离”，在中间表上做统计查询不会对在

线用户产生负面影响

②中间表上可以灵活地添加索引或者增加临时用的新字段，从而达到提高统计查询效率

和辅助统计查询作用

三、锁问题

1、MySQL 的 3 种锁

①表（级）锁：开销小，加锁快；不会出现死锁；锁定粒度大，发生锁冲突概率最高，

并发度最低（MyISAM，锁住整个表，可同时读，不可写）

②行（级）锁：开销大，加锁慢；会出现死锁；锁定粒度小，发生锁冲突概率最低，并

发度也最高（InnoDB，单独的一行记录加锁）

③页面锁：开销和加锁时间介于表锁和行锁之间；会出现死锁；锁定粒度介于表锁和行

锁之间，并发度一般

行锁

表锁

√

页面锁

√

MyISAM

BDB

√

InnoDB

√

√

2、MyISAM 的表级锁

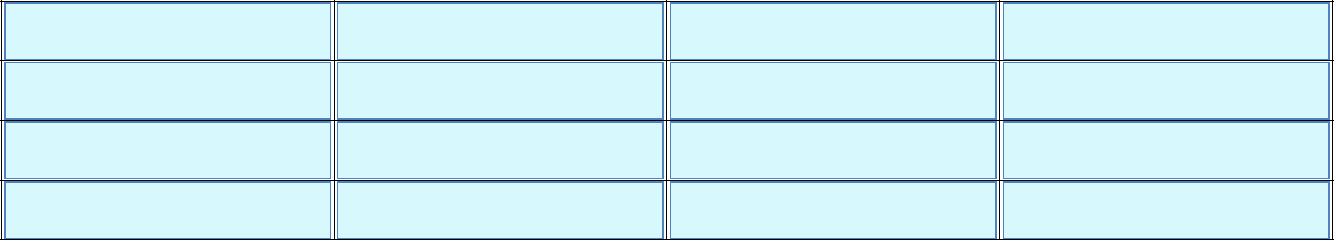
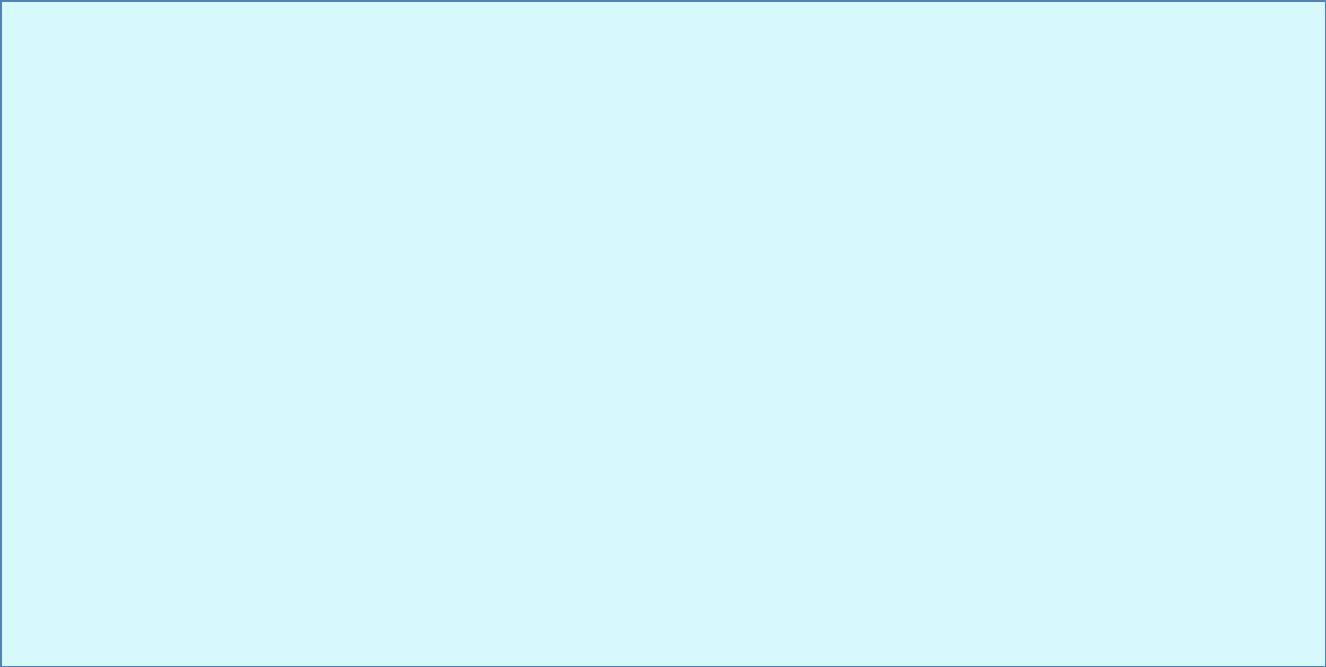
①两种模式：

表共享读锁（所有用户可以同时去读同一个加了该表锁的表，但是不能去对其进行

其他操作）

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



表独占写锁（一个用户在对加了该表锁的表进行写操作的时候，其他用户不能对其

进行操作）。MyISAM 表的读操作与写操作之间，以及写操作之间是串行的，只允许上述其

中一种操作进行

②加锁：MyISAM 在执行查询语句（SELECT）前，会自动给涉及的所有表加读锁，在执

行更新操作（UPDATE、DELETE、INSERT 等）前，会自动给涉及的表加写锁，这个过程并不

需要用户干预，都是由 MyISAM 引擎自动完成加锁

③MyISAM 锁调度：MyISAM 存储引擎的读锁和写锁是互斥的，读写操作是串行的。当

一个进程请求某个 MyISAM 表的读锁，同时另一个进程也请求同一表的写锁，此时 MySQL

会让写进程先获得锁。不仅如此，即使读请求先到锁等待队列，写请求后到，写锁也会插到

读锁请求之前

3、InnoDB 的锁

①几种锁模式：

共享锁（S 锁）：允许一个事务去读一行，阻止其他事务获得相同数据集的排他锁

排它锁（X 锁）：允许获得排他锁的事务更新数据，阻止其他事务取得相同数据集

的共享读锁和排他写锁。另外，为了允许行锁和表锁共存，实现多粒度锁机制

意向共享锁（IS 锁）：事务打算给数据行加行共享锁，事务在给一个数据行加共享

锁前必须先取得该表的 IS 锁。

意向排它锁（IX 锁）：事务打算给数据行加行排他锁，事务在给一个数据行加排他

锁前必须先取得该表的 IX 锁。

②加锁方式：意向锁是 InnoDB 自动加的，不需用户干预。对于 UPDATE、DELETE 和 INSERT

语句，InnoDB 会自动给涉及数据集加排他锁（X)；对于普通 SELECT 语句，InnoDB 不会加任

何锁

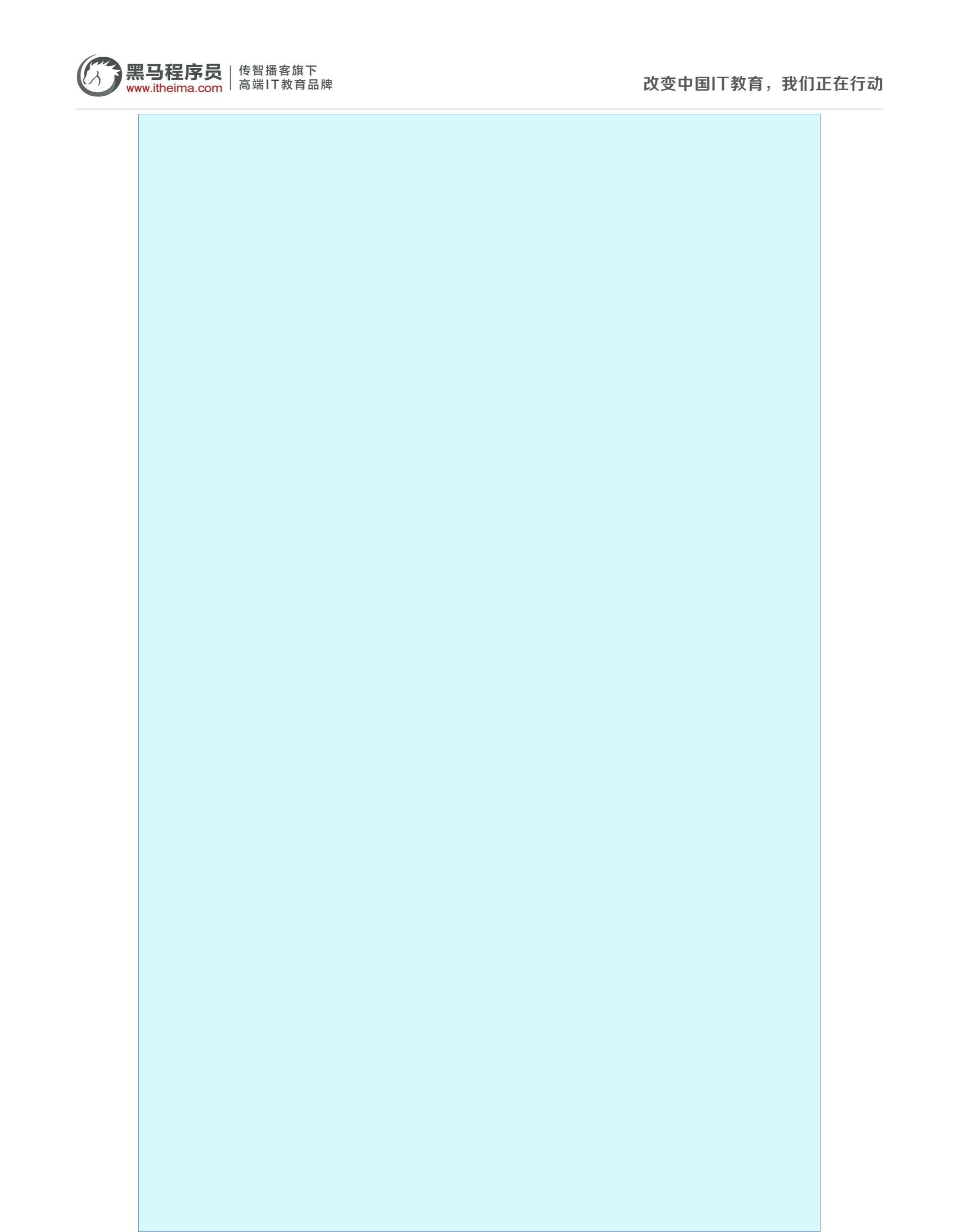
③在不通过索引条件查询的时候，InnoDB 确实使用的是表锁，而不是行锁

④由于 MySQL 的行锁是针对索引加的锁，不是针对记录加的锁，所以虽然是访问不同

行的记录，但是如果是使用相同的索引键，是会出现锁冲突

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



⑤当表有多个索引的时候，不同的事务可以使用不同的索引锁定不同的行，另外，不论

是使用主键索引、唯一索引或普通索引，InnoDB 都会使用行锁来对数据加锁。

4、锁使用总结：

①表锁更适合与以查询为主，只有少量按索引条件更新数据的应用，如 Web 应用；

②行锁更适合于有大量按索引条件并发更新少量不同数据，同时又有并发查询的应用，

入一些在线事务处理；

四、应用优化

1、使用连接池

连接池：需要访问数据库的的地方，都已经预先创建好，可以直接获取链接，分配给应用使

用，大大减少了创建新连接所耗费的资源。在访问结束后，链接将重新交还给连接池，以供

新访问使用

2、减少对 mysql 的访问

①避免对同一数据做重复检索

②使用查询缓存（query cache）

③增加 cache 层

PS：第③个和第②个作用差不多，只是 cache 层比 query cache 要更大，层次要深点，cache

层可以看做是 mysql 的二级数据库，query cache 相当于 mysql 内部的缓存

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



3、负载均衡：是重用的一种优化方式，采用某种均衡算法，将固定的负载量分布到不同的

服务器上，以此来减轻单台服务器的负载均衡，达到优化的目的

①利用 mysql 复制分流查询操作：一个主服务器承担更多操作，而多台从服务器承担产查询

操作，主从之间通过复制实现数据的同步。多台从服务器一方面用来确保可用性，一方面可

以创建不同的索引以满足不同的查询的需要。

②采用分布式数据库架构：分布式的数据库架构适合大数据量、负载高的情况，它具有良好

的扩展性和高可用性。（该情况只支持 innodb 存储引擎）

4、其他优化措施

①对于没有删除行操作的 myisam 表，插入操作和查询操作可以并行进行，因为没有删除操

作的表查询期间不会阻塞插入操作

②充分利用列有默认值的事实，只有当插入的值不同于默认值时，才明确地插入值。这会减

少 mysql 需要做的语法分析从而提升插入速度

③表的字段尽量不使用自增长变量，在高并发的情况下该字段的自增长可能对效率有比较大

的影响

概念补充

①更新丢失（Lost Update）：当两个或多个事务选择同一行，然后基于最初选定的值更新该

行时，由于每个事务都不知道其他事务的存在，就会发生丢失更新问题－－最后的更新覆盖

了由其他事务所做的更新。

②脏读（Dirty Reads）：一个事务正在对一条记录做修改，在这个事务完成并提交前，这条

记录的数据就处于不一致状态；这时，另一个事务也来读取同一条记录，如果不加控制，第

二个事务读取了这些“脏”数据，并据此做进一步的处理，就会产生未提交的数据依赖关系。

这种现象被形象地叫做"脏读"。

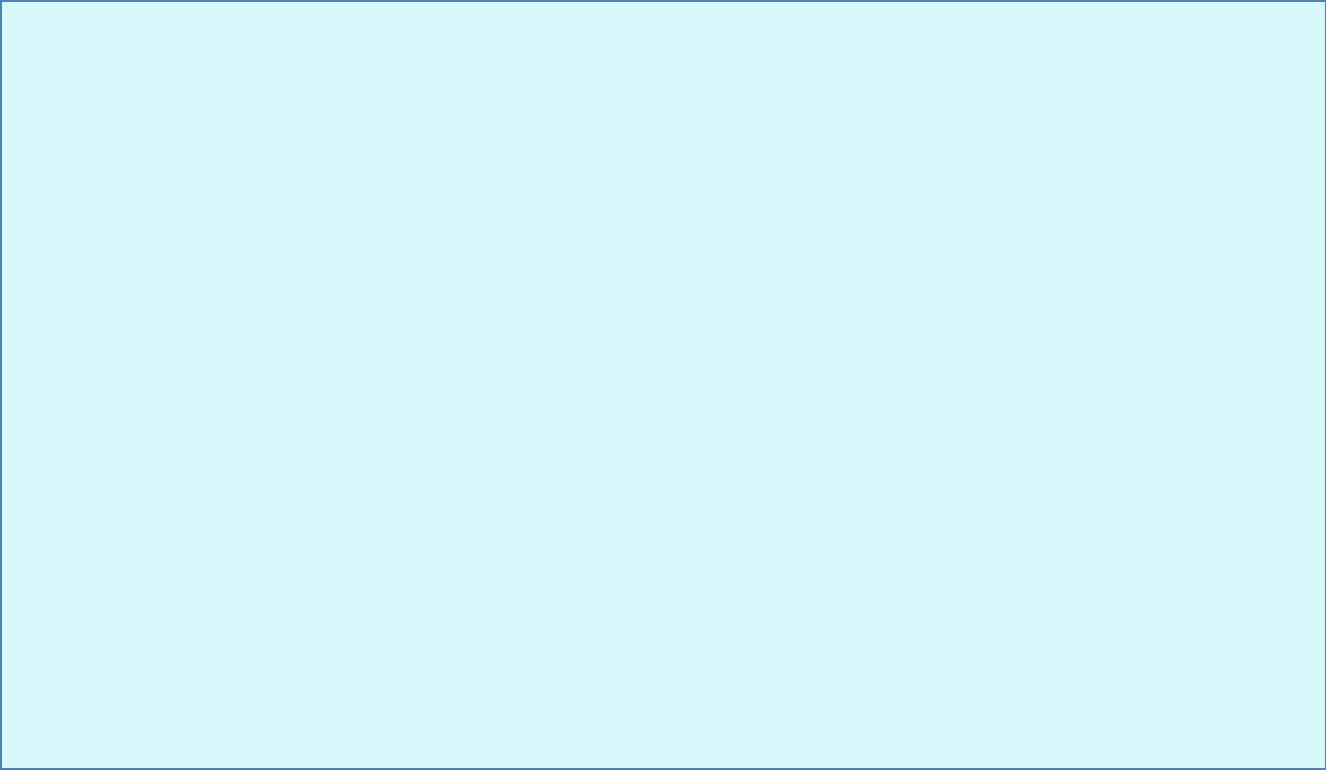
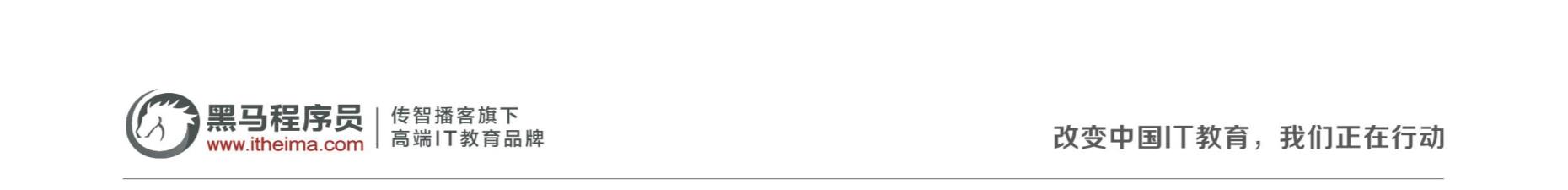
③不可重复读（Non-Repeatable Reads）：一个事务在读取某些数据后的某个时间，再次读

取以前读过的数据，却发现其读出的数据已经发生了改变、或某些记录已经被删除了！这种

现象就叫做“不可重复读”。

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090



④幻读（Phantom Reads）：一个事务按相同的查询条件重新读取以前检索过的数据，却发

现其他事务插入了满足其查询条件的新数据，这种现象就称为“幻读”。

⑤死锁（Deadlock）：是指两个或两个以上的进程在执行过程中，因争夺资源而造成的一种

互相等待的现象，这些永远在互相等待的进程称为死锁进程。表级锁不会产生死锁，所以解

决死锁主要还是真对于最常用的 InnoDB。

PS：在读取数据前，对其加锁，阻止其他事务对数据进行修改可避免脏读、不可重复读、幻

读

⑥悲观锁（Pessimistic Lock）：悲观锁的特点是先获取锁，再进行业务操作，即“悲观”的

认为获取锁是非常有可能失败的，因此要先确保获取锁成功再进行业务操作

⑦乐观锁（Optimistic Lock）：乐观锁的特点先进行业务操作，不到万不得已不去拿锁。即

“乐观”的认为拿锁多半是会成功的，因此在进行完业务操作需要实际更新数据的最后一步

再去拿一下锁就好

⑧热备份：热备份是在数据库运行的情况下，备份数据库的方法。即热备份是系统处于正常

运转状态下的备份

⑨冷备份：冷备份发生在数据库已经正常关闭的情况下，当正常关闭时会提供一个完整的数

据库。冷备份时将关键性文件拷贝到另外的位置的一种说法。冷备份是最快和最安全的方法。

北京市昌平区建材城西路金燕龙办公楼一层

电话：400-618-9090

