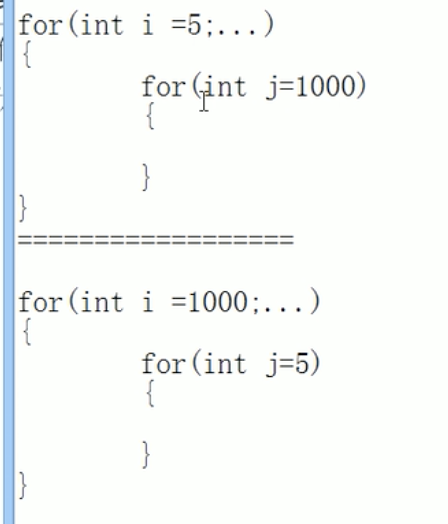
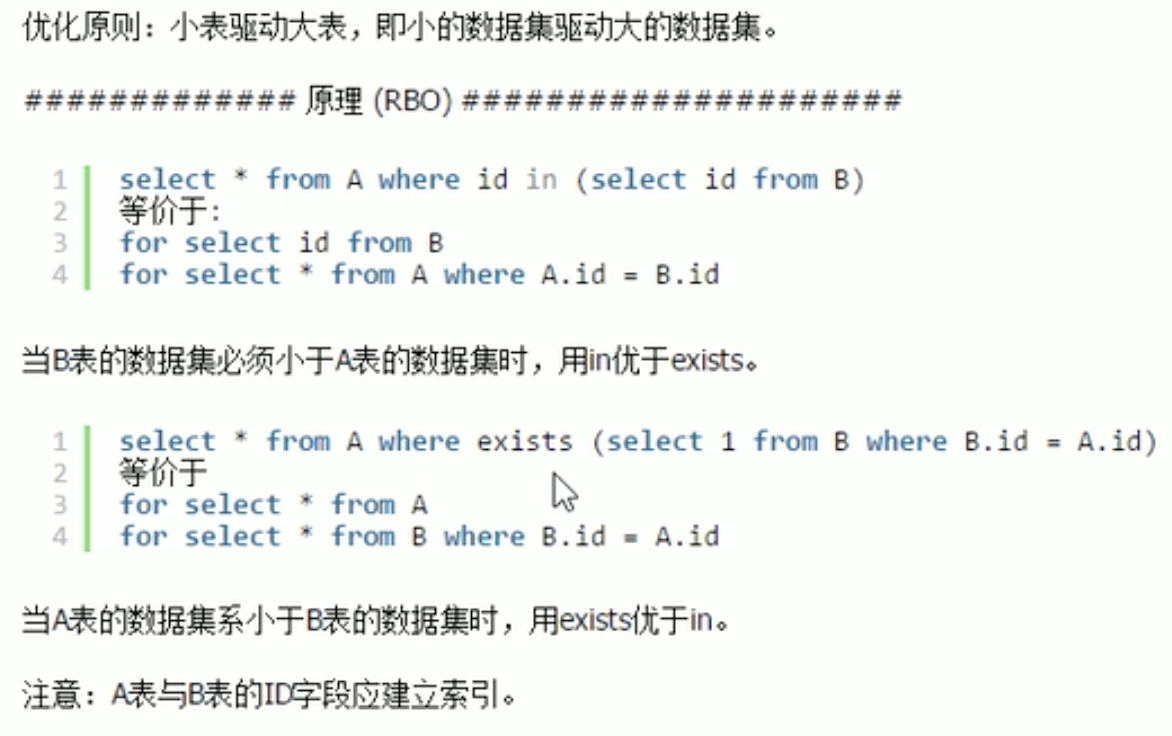
1. 慢查询的开启并捕获
2. explain + 慢SQL分析
3. show profile查询SQL在MySQL服务器里面的执行细节和声明周期情况
4. SQL数据服务器的参数调优

查询优化：

1. 永远小表驱动大表：小的数据集驱动大的数据集



类似上图，虽然两种写法，最后都是循环了5000次。但是对数据库来说，每一次查询需要有一次连接，如果是外层是小表，可以大大减少连接的次数。



exists

select … from table where exists (subquery)

1. 该语法可以理解为：将主查询的数据，放到子查询中做条件验证，根据验证结果（true、false）来决定主查询的数据结果是否保留。
2. exists subquery只会返回true和false，因此子查询中的select\* 或者其他没啥区别。
3. exists子查询不是万能的，是否有效要具体问题具体分析。
4. exists实际执行过程可能经过了优化，而不是我们理解的逐条对比。

Exists是in写法的变种：

Select \* from employee e where e.deptid in (select id from dept d);

Select \* from employee e where exists (select 1 from dept d where d.id=e.deptid);

以上两个SQL返回的结果是相同的，写法不同。怎么选择要看两张表的数据量大小，看哪种写法性能更好。

当然，第二个SQL中子查询的select 1 也可以修改为select \* 或者select ‘X’常量。

Order by优化

尽量使用index方式排序，避免filesort。

Index可以直接用索引本身完成排序，filesort方式效率低

Order by满足两种情况，会使用index:

1. order by 语句使用索引最左前列
2. 使用where子句与order by子句条件列组合满足索引最左前列

如果使用filesort，有两种算法：

双路排序：

MySQL4.1之前。先读取指针和order by列，对他们排序，然后扫描已排好序的列表，按顺序读出结果。

从磁盘读取排序字段，在buffer中进行排序，再从磁盘读取其他字段。

两次扫描磁盘。很耗IO。

单路排序：

MySQL4.1之后。读一次磁盘，会把查询需要的所有列放入buffer中，效率更快，但是会很费空间，因为把每一行保存在了内存中

优化策略：

1. 增大sort\_buffer\_size参数设置，缓冲区大小。

不管用哪种算法，这个参数设置都能提高性能。但是要根据自身系统去判断设置，因为这个参数是针对每个进程的

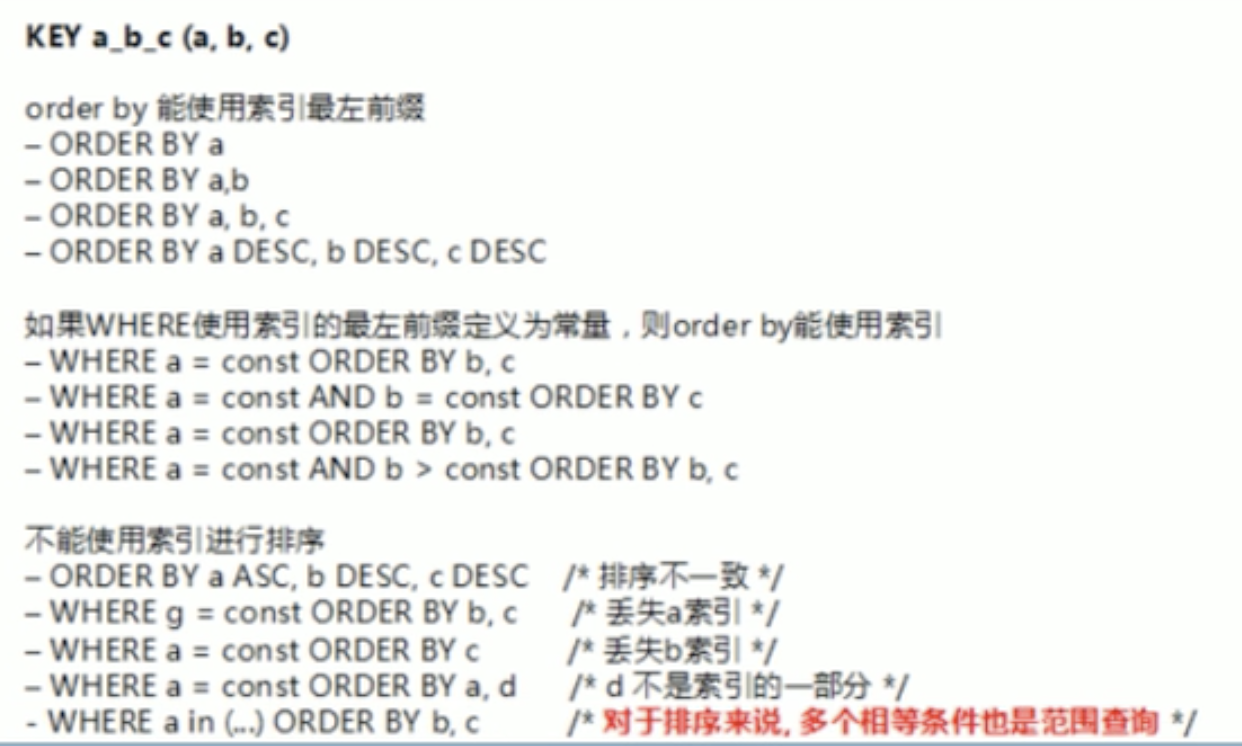
1. 增大max\_length\_for\_sort\_data参数

提高这个参数，会增加改进算法的概率，但是如果设的太高，数据总容量超出sort\_buffer\_size的概率增大，明显症状是高IO低CPU

1. order by时select \*很不好，因为：
   1. 当查询的字段大小总和小于max\_length\_for\_sort\_data而且排序字段不是text/blob类型时，会用单路排序；否则会用老算法排序
   2. 两种算法的数据都有可能会超出sort\_buffer的容量，超出之后会创建tmp临时文件进行合并排序，导致多次IO，用单路排序算法的风险更大一些，所以要提高sort\_buffer\_size

总结：

1. MySQL两种排序方式：文件排序using filesort或扫描有序索引排序using index
2. MySQL能为排序和查询使用相同的索引
3. 几个排序的栗子：

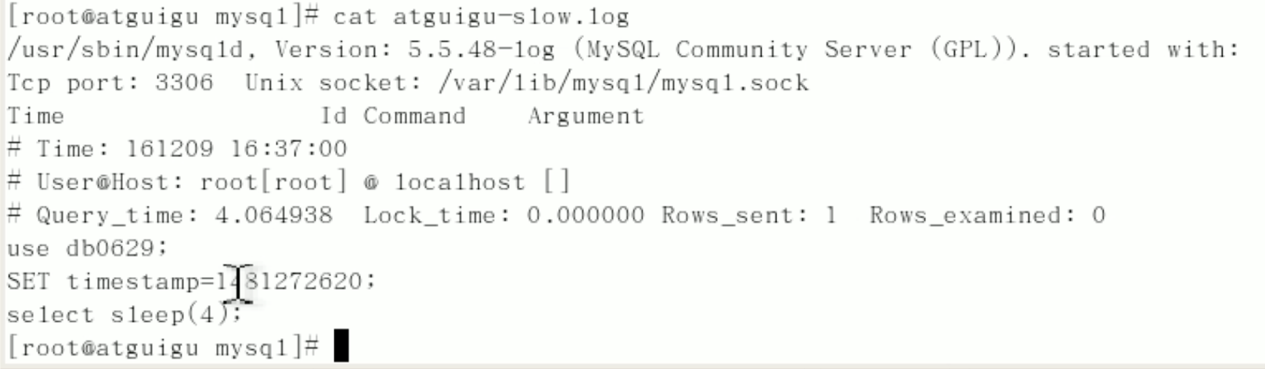


group by：几乎与order by一致

1. 实质是先排序后分组，遵循索引的最佳左前缀
2. 当无法使用索引列，增大max\_length\_for\_sort\_data和sort\_buffer\_size参数设置
3. where高于having，能写在where限定的条件就不要去having限定了

慢查询日志：

1. 查询是否开启慢查询：show variables like ‘%slow\_query\_log%’;
2. 开启后，运行时长大于long\_query\_time的SQL会被记录下来：show variables like ‘%long\_query\_time%’; 单位：秒 MySQL是判断大于long\_query\_time，不包含等于
3. 设置了long\_query\_time后再查不生效，需要新开一个窗口
4. select sleep(4) : 睡眠4秒
5. 如果有慢SQL，可以在日志文件中看到内容：



1. 查询当前系统中有多少条慢sql：show global status like ‘%Slow\_queries%;

想要批量向数据库中插数据，可以通过写函数或者存储过程。

Show profile:

Mysql提供的可以分析当前会话语句执行的资源消耗情况，可以用于SQL的调优测量。

默认关闭，并保存最近15次的运行结果。