

提纲挈领的告诉你，SQL语句的执行计划和性能优化有什么关系？

之前我们已经彻底搞清楚了MySQL的索引结构，也知道了索引平时要怎么样写SQL才能用上，而且也是用一个案例给大家讲解了，平时我们做一个系统，写好代码之后，要如何去设计表的索引，让每个查询都可以用上索引，所以这里纠正了大家平时的一个观念，可能有些人认为，平时设计好表就必须同时设计好索引，其实完全不是这么回事。

一般开发一个系统，都是先设计表结构，表结构必须满足业务需求，然后写代码，代码都写好之后，再根据你的代码如何查询表的，来设计里面的索引，考虑设计几个索引，是不是联合索引，选择哪些字段，字段顺序如何排列，才能让查询语句都用上索引。

那么接着我们就要进入MySQL学习中极为重要的一个环节了，那就是MySQL的查询语句的执行计划分析以及SQL优化，这可以说是MySQL实践中对于开发人员最常见最需要掌握的一个技能了，但是很多人对MySQL内核级的原理的理解较为肤浅，同时对于索引结构和查询时使用索引的原理也不甚了解，更谈不上说能看懂MySQL的执行计划了。

如果是这样的话，你还怎么去说自己可以进行MySQL的SQL优化呢？

可能有人反问了，你不是都告诉我们索引结构和使用原理了么，使用规则我们也知道了，那SQL优化无非就是开发的时候让自己写的SQL都用上索引不就ok了？

这个话也对，也不对。应该这么说，根据查询语句设计良好的索引，让所有查询都尽可能用上索引，这本身就是一种SQL优化的技巧，但是他仅仅只是其一罢了，并不能说掌握这个，就掌握了所有的SQL优化技巧

反过来说，SQL优化技巧中包含了我们之前讲的设计索引以及让SQL用上索引，但是SQL优化还有很多其他的東西。

实际上有时候往往你会发现自己的数据库里有很多表，每个表的数据量也不小，然后写出来的SQL也比较复杂，各种关联和嵌套子查询，搞的人看的都眼晕，然后表面上看起来这个SQL部分用上了索引，结果性能还是差，差，差，这是为什么呢？

所以说，基础的以及日常的SQL优化就是设计好索引，让一般不太复杂的普通查询都用上索引，但是针对复杂表结构和大数据量的上百行复杂SQL的优化，必须得建立在你先懂这个复杂SQL是怎么执行的

你有那么多的数据表，每个表都有一个聚簇索引，聚簇索引的叶子就是那个表的真实数据，同时每个表还设计了一些二级索引，那么上百行的复杂SQL跑起来的时候到底是如何使用各个索引，如何读取数据的？

这个SQL语句（不管是简单还是复杂），在实际的MySQL底层，针对磁盘上的大量数据表、聚簇索引和二级索引，如何检索查询，如何筛选过滤，如何使用函数，如何进行排序，如何进行分组，到底怎么能把你想要的东西查出来，这个过程就是一个很重要的东西：**执行计划**！

也就是说，每次你提交一个SQL给MySQL，他内核里的查询优化器，都会针对这个SQL语句的语义去生成一个执行计划，这个执行计划就代表了，他会怎么查各个表，用哪些索引，如何做排序和分组，看懂这个执行计划，你就学会了真正的SQL优化的一半了！

当你看懂执行计划之后，还能根据他的实际情况去想各种办法改写你的SQL语句，改良你的索引设计，进而优化SQL语句的执行计划，最终让SQL语句的性能得到提升，这个就是所谓的SQL调优

好了，今天先提纲挈领的给大家讲一下执行计划和SQL优化的关系，下一次开始，我们就正式先学习如何读懂MySQL的SQL执行计划！

End