

1. 试述查询优化在关系数据库系统中的重要性和可能性

重要性：关系系统的查询优化既是 RDBMS 实现的关键技术又是关键系统的优点所在。它减轻了用户选择存取路径的负担。用户只要提出干什么，不必指出怎么干。

查询优化的优点不仅在于用户不必考虑如何最好的表达查询以获得更好的效率，而且在于系统可以比较用户程序的优化做得更好。

可能性：

(1) 优化器可以从数据字典中获取许多统计信息，优化器可以根据这些信息选择有效的执行计划，而用户程序则难以获得这些信息。

(2) 如果数据库的物理统计信息改变了，系统可以自动对查询进行重新优化以选择相适应的执行计划。在非关系系统中必须重写程序，而重写程序在实际应用中往往是不太可能的。

(3) 优化器可以考虑数十甚至数百种不同的执行计划，从中选出较优的一个，而程序员一般只能考虑有限的几种可能性。

(4) 优化器中包括了很多复杂的优化技术，这些优化技术往往只有最好的程序员才能掌握，系统的自动优化相当于是的所有人都拥有这些优化技术。

2. 假设关系 $R(A, B)$ 和 $S(B, C, D)$ 情况如下： R 有 20 000 个元组， S 有 1 200 个元组，一个块能装 40 个 R 的元组，能装 30 个 S 的元组，估算下列操作需要多少次磁盘块读写。

(1) R 上没有索引， $\text{select } * \text{ from } R;$

(2) R 中 A 为主码， A 有 3 层 $B+$ 树索引， $\text{select } * \text{ from } R \text{ where } A = 10;$

(3) 嵌套循环连接 $R \bowtie S;$

(4) 排序合并连接 $R \bowtie S$ ，区分 R 与 S 在 B 属性上已经有序和无序两种情况。

(1) R 有 20000 个元组，一个块能装 40 个 R 的元组，所以 $20000/40=500$ 块

(2) 高度每下降一级，即读取一次磁盘，加上该元组所在的一块， $3+1=4$ 次

(3) 嵌套循环连接 $R \bowtie S$ $R \bowtie S$ 没有给出连接操作使用的内存缓冲区块数 K ，所以无法确定设在内存中存放 X 块 R 元组和 Y 块 S 元组读 R 表块数+读 C 表遍数每遍块数

$W1=20000/40+[20000/(40X)][1200/(30Y)]$ 连接后的元组数为 1200。设每块能装 Z 个元组，则写出 $W2=1200/Z$ 块。 $W1+W2$

(4) 排序合并连接 $R \bowtie S$ $R \bowtie S$ ，区分 R 与 S 在 B 属性上已经有序和无序两种情况。
 R 和 S 只读取一遍， $20000/40+1200/30=540$ ，
未排序相对于排好序的要加上排序代价