

有关OracleIndex的三个问题（一）PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/143/2021\\_2022\\_\\_E6\\_9C\\_89\\_E5\\_85\\_B3Orac\\_c102\\_143358.htm](https://www.100test.com/kao_ti2020/143/2021_2022__E6_9C_89_E5_85_B3Orac_c102_143358.htm) 索引 ( Index ) 是常见的数据库对象，它的设置好坏、使用是否得当，极大地影响数据库应用程序和Database 的性能。虽然有许多资料讲索引的用法，DBA 和Developer们也经常与它打交道，但笔者发现，还是有不少的人对它存在误解，因此针对使用中的常见问题，讲三个问题。此文所有示例所用的数据库是Oracle 8.1.7 OPS on HP N series，示例全部是真实数据，读者不需要注意具体的数据大小，而应注意在使用不同的方法后，数据的比较。本文所讲基本都是陈词滥调，但是笔者试图通过实际的例子，来真正让您明白事情的关键。第一讲、索引并非总是最佳选择。如果发现 Oracle 在有索引的情况下，没有使用索引，这并不是Oracle 的优化器出错。在有些情况下，Oracle 确实会选择全表扫描 ( Full Table Scan )，而非索引扫描 ( Index Scan )。这些情况通常有：1. 表未做 statistics, 或者statistics 陈旧，导致Oracle 判断失误。2. 根据该表拥有的记录数和数据块数，实际上全表扫描要比索引扫描更快。对第1种情况，最常见的例子，是以下这句sql 语句：`0select count(*) from mytable.`在未作statistics 之前，它使用全表扫描，需要读取 6000 多个数据块（一个数据块是8k），做了statistics 之后，使用的是INDEX (FAST FULL SCAN)，只需要读取450 个数据块。但是，statistics 做得不好，也会导致Oracle 不使用索引。第2种情况就要复杂得多。一般概念上都认为索引比表快，比较难以理解什么情况下全表扫描要比索引扫描快。为了讲清楚这个

问题，这里先介绍一下Oracle 在评估使用索引的代价（cost）时两个重要的数据：CF(Clustering factor) 和 FF(Filtering factor). CF: 所谓CF, 通俗地讲，就是每读入一个索引块，要对应读入多少个数据块。FF: 所谓FF, 就是该sql 语句所选择的结果集，占总的数据量的百分比。大约的计算公式是： $FF * (CF \text{ 索引块个数})$ ，由此估计出，一个查询，如果使用某个索引，会需要读入的数据块数。需要读入的数据块越多，则cost 越大，Oracle 也就越可能不选择使用 index.（全表扫描需要读入的数据块数等于该表的实际数据块数）其核心就是，CF 可能会比实际的数据块数量大。CF 受到索引中数据的排列方式影响，通常在索引刚建立时，索引中的记录与表中的记录有良好的对应关系，CF 都很小；在表经过大量的插入、修改后，这种对应关系越来越乱，CF 也越来越大。此时需要DBA 重新建立或者组织该索引。如果某个sql 语句以前一直使用某索引，较长时间后不再使用，一种可能就是CF 已经变得太大，需要重新整理该索引了。100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)