

如何在Oracle数据库字段上建立索引 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\\_\\_E5\\_A6\\_82\\_E4\\_BD\\_95\\_E5\\_9C\\_A8O\\_c102\\_143382.htm](https://www.100test.com/kao_ti2020/143/2021_2022__E5_A6_82_E4_BD_95_E5_9C_A8O_c102_143382.htm) 当where子句对某一列使用函数时，除非利用这个简单的技术强制索引，否则Oracle优化器不能在查询中使用索引。通常情况下，如果在WHERE子句中不使用诸如UPPER、REPLACE或SUBSTRD等函数，就不能对指定列建立特定的条件。但如果使用了这些函数，则会出现一个问题：这些函数会阻碍Oracle优化器对列使用索引，因而与采用索引的情况相比较，查询会花费更多的时间。庆幸的是，如果在使用函数的这些列中包含了字符型数据，可以用这样一种方法修改查询语句，以达到强制性使用索引，更有效地运行查询。这篇文章介绍了涉及的技术，并说明了在两种典型情况下怎样实现。大小写混合情况在讨论由于函数修改了列的内容，如何强制使用索引前，让我们首先看看为什么Oracle优化器在这种情况下不能使用索引。假定我们要搜寻包含了大小写混合的数据，如在表1中ADDRESS表的NAME列。因为数据是用户输入的，我们无法使用已经统一改为大写的的数据。为了找到每一个名为john的地址，我们使用包含了UPPER子句的查询语句。如下所示：  
SQL> 0select address from address where upper(name) like JOHN.在运行这个查询语句前，如果我们运行了命令"set autotrace on"，将会得到下列结果，其中包含了执行过程：  
ADDRESS cleveland 1 row 0selected. Execution Plan SELECT STATEMENT TABLE ACCESS FULL ADDRESS可以看到，在这种情况下，Oracle优化器对ADDRESS表作了一次完整的扫

描，而没有使用NAME 列的索引。这是因为索引是根据列中数据的实际值建立的，而UPPER 函数已经将字符转换成大写，即修改了这些值，因此该查询不能使用这列的索引。优化器不能与索引项比较"JOHN"，没有索引项对应于"JOHN"-只有"john"。值得庆幸的是，如果在这种情况下想要强制使用索引，有一种简便的方法：只要在WHERE 子句中增加一个或多个特定的条件，用于测试索引值，并减少需要扫描的行，但这并没有修改原来SQL 编码中的条件。以下列查询语句为例：SQL> 0select address from address where upper(name) like JO% AND (name like J% or name like j%).使用这种查询语句（已设置AUTOTRACE），可得到下列结果：ADDRESS  
cleveland 1 row 0selected. Execution Plan SELECT STATEMENT  
CONCATENATION TABLE ACCESS BY INDEX ROWID  
ADDRESS INDEX RANGE SCAN ADDRESS\_I TABLE ACCESS  
BY INDEX ROWID ADDRESS INDEX RANGE SCAN  
ADDRESS\_I现在，优化器为WHERE 子句中AND 联结的两个语句中每一个语句确定的范围进行扫描----第二个语句没有引用函数，因而使用了索引。在两个范围扫描后，将运行结果合并。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)