

Oracle索引原理方法学习 PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/285/2021_2022_Oracle_E7_B4_A2_E5_c102_285323.htm Oracle提供了大量索引选项。知道在给定条件下使用哪个选项对于一个应用程序的性能来说非常重要。一个错误的选择可能会引发死锁，并导致数据库性能急剧下降或进程终止。而如果做出正确的选择，则可以合理使用资源，使那些已经运行了几个小时甚至几天的进程在几分钟得以完成，这样会使您立刻成为一位英雄。这篇文章就将简单的讨论每个索引选项。主要有以下内容：[1] 基本的索引概念 查询DBA_INDEXES视图可得到表中所有索引的列表，注意只能通过USER_INDEXES的方法来检索模式(schema)的索引。访问USER_IND_COLUMNS视图可得到一个给定表中被索引的特定列。[2] 组合索引 当某个索引包含有多个已索引的列时，称这个索引为组合（concatented）索引。在Oracle9i引入跳跃式扫描的索引访问方法之前，查询只能在有限条件下使用该索引。比如：表emp有一个组合索引键，该索引包含了empno、ename和deptno。在Oracle9i之前除非在where之句中对第一列（empno）指定一个值，否则就不能使用这个索引键进行一次范围扫描。特别注意：在Oracle9i之前，只有在使用到索引的前导索引时才可以使用组合索引！[3] ORACLE ROWID 通过每个行的ROWID，索引Oracle提供了访问单行数据的能力。ROWID其实就是直接指向单独行的线路图。如果想检查重复值或是其他对ROWID本身的引用，可以在任何表中使用和指定rowid列。[4] 限制索引 限制索引是一些没有经验的开发人员经常犯的错误之一。在SQL中有

很多陷阱会使一些索引无法使用。下面讨论一些常见的问题：

4.1 使用不等于操作符 (、!=) 下面的查询即使在cust_rating列有一个索引，查询语句仍然执行一次全表扫描。

```
0select cust_Id,cust_name from customers where cust_rating ' aa ' .
```

把上面的语句改成如下的查询语句，这样，在采用基于规则的优化器而不是基于代价的优化器（更智能）时，将会使用索引。

```
0select cust_Id,cust_name from customers where cust_rating ' aa ' .
```

特别注意：通过把不等于操作符改成OR条件，就可以使用索引，以避免全表扫描。

4.2 使用IS NULL或IS NOT NULL 使用IS NULL 或IS NOT NULL同样会限制索引的使用。因为NULL值并没有被定义。在SQL语句中使用NULL会有很多的麻烦。因此建议开发人员在建表时，把需要索引的列设成NOT NULL。如果被索引的列在某些行中存在NULL值，就不会使用这个索引（除非索引是一个位图索引，关于位图索引在稍后在详细讨论）。

4.3 使用函数 如果不使用基于函数的索引，那么在SQL语句的WHERE子句中对存在索引的列使用函数时，会使优化器忽略掉这些索引。下面的查询不会使用索引（只要它不是基于函数的索引）

```
0select empno,ename,deptno from emp where trunc(hiredate)= ' 01-MAY-81 ' .
```

把上面的语句改成下面的语句，这样就可以通过索引进行查找。

```
0select empno,ename,deptno from emp where hiredate
```