

ORACLE优化SQL语句提高效率(2)Oracle认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022_ORACLE_E4_BC_98_E5_c102_645094.htm "tb42" class="mar10"> 索引是表的一个概念部分，用来提高检索数据的效率，Oracle使用了一个复杂的自平衡B-tree结构。通常,通过索引查询数据比全表扫描要快。当 Oracle找出执行查询和Update语句的最好路径时, Oracle优化器将使用索引。同样在联结多个表时使用索引也能够提高效率。另一个使用索引的好处是,他提供了主键(primary key)的唯一性验证。那些LONG或LONG RAW数据类型,您能够索引几乎任何的列。通常,在大型表中使用索引特别有效.当然,您也会发现,在扫描小表时,使用索引同样能提高效率。虽然使用索引能得到查询效率的提高,但是我们也必须注意到他的代价。索引需要空间来存储,也需要定期维护,每当有记录在表中增减或索引列被修改时,索引本身也会被修改。这意味着每条记录的INSERT, DELETE, UPDATE将为此多付出4、5次的磁盘I/O。因为索引需要额外的存储空间和处理,那些不必要的索引反而会使查询反应时间变慢。定期的重构索引是有必要的: ALTER INDEX gt. REBUILD gt.

(18) 用EXISTS替换DISTINCT: 当提交一个包含一对多表信息(比如部门表和雇员表)的查询时,避免在SELECT子句中使用DISTINCT。一般能够考虑用EXIST替换, EXISTS使查询更为迅速,因为RDBMS核心模块将在子查询的条件一旦满足后,立即返回结果。例子: (低效): SELECT DISTINCT DEPT_NO,DEPT_NAME FROM DEPT D, EMP E WHERE D.DEPT_NO = E.DEPT_NO (高效): SELECT

DEPT_NO,DEPT_NAME FROM DEPT D WHERE EXISTS (SELECT ' X FROM EMP E WHERE E.DEPT_NO = D.DEPT_NO). (19) SQL语句用大写的; 因为Oracle总是先解析SQL语句, 把小写的字母转换成大写的再执行。 (20) 在Java代码中尽量少用连接符“+”连接字符串。 (21) 避免在索引列上使用NOT通常, 我们要避免在索引列上使用NOT, NOT会产生在和在索引列上使用函数相同的影响。当Oracle“碰到”NOT,他就会停止使用索引转而执行全表扫描。 (22) 避免在索引列上使用计算。WHERE子句中, 假如索引列是函数的一部分。优化器将不使用索引而使用全表扫描。 举例: 低效: SELECT ... FROM DEPT WHERE SAL * 12 gt. 25000/12. (23) 用gt.: 高效: SELECT * FROM EMP WHERE DEPTNO gt.3 两者的区别在于, 前者DBMS将直接跳到第一个DEPT等于4的记录而后者将首先定位到DEPTNO=3的记录并且向前扫描到第一个DEPT大于3的记录。 (24) 用UNION替换OR (适用于索引列): 通常情况下, 用UNION替换WHERE子句中的OR将会起到较好的效果。对索引列使用OR将造成全表扫描。注意, 以上规则只针对多个索引列有效。假如有column没有被索引, 查询效率可能会因为您没有选择OR而降低。在下面的例子中, LOC_ID 和REGION上都建有索引。 高效: SELECT LOC_ID, LOC_DESC, REGION FROM LOCATION WHERE LOC_ID = 10 UNION SELECT LOC_ID, LOC_DESC, REGION FROM LOCATION WHERE REGION = “MELBOURNE” 低效: SELECT LOC_ID, LOC_DESC, REGION FROM LOCATION WHERE LOC_ID = 10 OR REGION =

“ MELBOURNE ” 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