

OracleSQL性能优化技巧大总结 PDF转换可能丢失图片或格式
， 建议阅读原文

https://www.100test.com/kao_ti2020/207/2021_2022_OracleSQL_E6_c102_207513.htm

(1) 选择最有效率的表名顺序(只在基于规则的优化器中有效)：ORACLE的解析器按照从右到左的顺序处理FROM子句中的表名，FROM子句中写在最后的表(基础表 driving table)将被最先处理，在FROM子句中包含多个表的情况下,你必须选择记录条数最少的表作为基础表。如果有3个以上的表连接查询,那就需要选择交叉表(intersection table)作为基础表,交叉表是指那个被其他表所引用的表。(2) WHERE子句中的连接顺序：ORACLE采用自下而上的顺序解析WHERE子句,根据这个原理,表之间的连接必须写在其他WHERE条件之前,那些可以过滤掉最大数量记录的条件必须写在WHERE子句的末尾。(3) SELECT子句中避免使用‘*’：ORACLE在解析的过程中,会将*依次转换成所有的列名,这个工作是通过查询数据字典完成的,这意味着将耗费更多的时间(4) 减少访问数据库的次数：ORACLE在内部执行了许多工作:解析SQL语句,估算索引的利用率,绑定变量,读数据块等；(5) 在SQL*Plus, SQL*Forms和Pro*C中重新设置ARRAYSIZE参数,可以增加每次数据库访问的检索数据量,建议值为200(6) 使用DECODE函数来减少处理时间：使用DECODE函数可以避免重复扫描相同记录或重复连接相同的表。(7) 整合简单,无关联的数据库访问：如果你有几个简单的数据库查询语句,你可以把它们整合到一个查询中(即使它们之间没有关系)(8) 删除重复记录：最高效的删除重复记录方法(因为使用了ROWID)例子：DELETE FROM EMP E

WHERE E.ROWID > (SELECT MIN(X.ROWID) FROM EMP X WHERE X.EMP_NO = E.EMP_NO). (9) 用TRUNCATE替代DELETE：当删除表中的记录时,在通常情况下,回滚段(rollback segments)用来存放可以被恢复的信息.如果你没有COMMIT事务,ORACLE会将数据恢复到删除之前的状态(准确地说是恢复到执行删除命令之前的状况)而当运用TRUNCATE时,回滚段不再存放任何可被恢复的信息.当命令运行后,数据不能被恢复.因此很少的资源被调用,执行时间也会很短.(译者按: TRUNCATE只在删除全表适用,TRUNCATE是DDL不是DML) (10) 尽量多使用COMMIT：只要有可能,在程序中尽量多使用COMMIT,这样程序的性能得到提高,需求也会因为COMMIT所释放的资源而减少: COMMIT所释放的资源: a. 回滚段上用于恢复数据的信息. b. 被程序语句获得的锁 c. redo log buffer 中的空间 d. ORACLE为管理上述3种资源中的内部花费

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