

Oracle8i基于规则的优化机制对表达式的处理 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/253/2021_2022_Oracle8i_E5_9F_c102_253895.htm ORACLE优化器在任何可能的时候都会对表达式进行评估，并且把特定的语法结构转换成等价的结构，这么做的原因是：要么结果表达式能够比源表达式具有更快的速度 要么源表达式只是结果表达式的一个等价语义结构不同的SQL结构有时具有同样的操作（例如： $= ANY$ (subquery) and IN (subquery)），ORACLE会把他们映射到一个单一的语义结构。下面将讨论优化器如何评估优化如下的情况和表达式：常量 LIKE 操作符 IN 操作符 ANY和SOME 操作符 ALL 操作符 BETWEEN 操作符 NOT 操作符 传递

（Transitivity）确定性（DETERMINISTIC）函数 常量 常量的计算是在语句被优化时一次性完成，而不是在每次执行时。下面是检索月薪大于2000的的表达式： $sal > 24000/12$ $sal > 2000$ $sal*12 > 24000$ 如果SQL语句包括第一种情况，优化器会简单地把它转变成第二种。注意：优化器不会简化跨越比较符的表达式，例如第三条语句，鉴于此，应用程序开发者应该尽量写用常量跟字段比较检索的表达式，而不要将字段置于表达式当中。LIKE 操作符 优化器把使用LIKE操作符和一个没有通配符的表达式组成的检索表达式转换为一个“=”操作符表达式。例如：优化器会把表达式 $ename LIKE SMITH$ 转换为 $ename = SMITH$ 优化器只能转换涉及到可变长数据类型的表达式，前一个例子中，如果ENAME字段的类型是CHAR(10)，那么优化器将不做任何转换。IN 操作符 优化器把使用IN比较符的检索表达式替换为等价的使用“=”和

“ OR ”操作符的检索表达式。例如，优化器会把表达式 `ename IN (SMITH,KING,JONES)` 替换为 `ename = SMITH OR ename = KING OR ename = JONES` ANY和SOME 操作符 优化器将跟随（following）值列表的ANY和SOME检索条件用等价的同等操作符和“ OR ”组成的表达式替换。例如，优化器将如下所示的第一条语句用第二条语句替换：
`sal > ANY (:first_sal, :second_sal)` `sal > :first_sal OR sal > :second_sal` 优化器将跟随子查询的ANY和SOME检索条件转换成由“ EXISTS ”和一个相应的子查询组成的检索表达式。例如，优化器将如下所示的第一条语句用第二条语句替换：
`x > ANY (SELECT sal FROM emp WHERE job = ANALYST)` `EXISTS (SELECT sal FROM emp WHERE job = ANALYST AND x > sal)` ALL 操作符 优化器将跟随值列表的ALL操作符用等价的“ = ”和“ AND ”组成的表达式替换。例如，`sal > ALL (:first_sal, :second_sal)` 表达式会被替换为：`sal > :first_sal AND sal > :second_sal` 对于跟随子查询的ALL表达式，优化器用ANY和另外一个合适的比较符组成的表达式替换。例如，优化器会把表达式 `x > ALL (SELECT sal FROM emp WHERE deptno = 10)` 替换为：`NOT (x` 接下来优化器会把第二个表达式适用ANY表达式的转换规则转换为下面的表达式：`NOT EXISTS (SELECT sal FROM emp WHERE deptno = 10 AND x BETWEEN` 操作符 优化器总是用“ >= ”和“ 例如：优化器会把表达式 `sal BETWEEN 2000 AND 3000` 用 `sal >= 2000 AND sal NOT` 操作符 优化器总是试图简化检索条件以消除“ NOT ”逻辑操作符的影响，这将涉及到“ NOT ”操作符的消除以及代以相应的比较运算符。例如，优化器将下面的第一条语句用第二条语句代替：`NOT deptno`

= (SELECT deptno FROM emp WHERE ename = TAYLOR)
deptno (SELECT deptno FROM emp WHERE ename = TAYLOR)
通常情况下一个含有NOT操作符的语句有很多不同的写法，
优化器的转换原则是使“NOT”操作符后边的子句尽可能的
简单，即使可能会使结果表达式包含了更多的“NOT”操作
符。例如，优化器将如下所示的第一条语句用第二条语句代
替：NOT (sal NOT sal = 1000 AND comm IS NOT NULL 传递
(Transitivity) 如果“WHERE”子句的两个检索条件涉及了
一个共同的字段，优化器有时会根据传递原理推断出第三个
检索条件，随后可以根据这个推断出的条件对语句进行优化
，推断出的条件可能会激活一个原来的检索条件没有激活的
潜在的接口路径（access path）。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