

【书评:Oracle 查询优化改写】第四章

BLOG 文档结构图

▲ 【书评:Oracle 查询优化改写】第四章
1.1 导读
1.2 实验环境介绍
1.3 前言
1.4 check 的特殊用法
1.5 merge 语句
1.6 总结
1.7 about me

1.1 导读

各位技术爱好者，看完本文后，你可以掌握如下的技能，也可以学到一些其它你所不知道的知识，~O(∩_∩)O~：

- ① check 的特殊用法
- ② sql 优化中使用 merge 语句代替 update 语句（重点）

本文如有错误或不完善的地方请大家多多指正，ITPUB 留言或 QQ 皆可，您的批评指正是我写作的最大动力。

1.2 实验环境介绍

目标库：11.2.0.3 RHEL6.5

1.3 前言

前 3 章的链接参考相关连接：

- 【书评:Oracle 查询优化改写】第一章 <http://blog.itpub.net/26736162/viewspace-1652985/>
- 【书评:Oracle 查询优化改写】第二章 <http://blog.itpub.net/26736162/viewspace-1654252/>
- 【书评:Oracle 查询优化改写】第三章 <http://blog.itpub.net/26736162/viewspace-1660422/>

今天来写写这本书的第四章的内容，第四章主要讲了 UPDATE 语句的正确用法，以及**什么时候 UPDATE 语句应改写为 MERGE**，第四章的内容目录如下：

第 4 章 插入、更新与删除

4.1 插入新记录

4.2 阻止对某几列插入

4.3 复制表的定义及数据

4.4 用 WITH CHECK OPTION 限制数据录入

4.5 多表插入语句

4.6 用其他表中的值更新

4.7 合并记录

4.8 删除违反参照完整性的记录

4.9 删除名称重复的记录

1.4 check 的特殊用法

我们知道 sysdate 不能用于 check 约束，但是有这种需求的时候怎么办呢？如下例子利用视图加 with check option 即可解决。

```
09:39:08 SQL> create table ttt(create_date date check(create_date > sysdate));
create table ttt(create_date date check(create_date > sysdate))
*
ERROR at line 1:
ORA-02436: date or system variable wrongly specified in CHECK constraint

09:41:56 SQL> insert into (select empno,ename,hiredate from scott.emp where hiredate <= sysdate with check option)
09:42:13 2 values ( 9999,'test',sysdate+1);
insert into (select empno,ename,hiredate from scott.emp where hiredate <= sysdate with check option)
*
ERROR at line 1:
ORA-01402: view WITH CHECK OPTION where-clause violation

Elapsed: 00:00:00.12
09:42:14 SQL> insert into (select empno,ename,hiredate from scott.emp where hiredate <= sysdate with check option)
09:42:56 2 values ( 9999,'test',sysdate-1);

1 row created.

Elapsed: 00:00:00.03
09:42:57 SQL>
```

1.5 merge 语句

关于 update 的一个容易出错的地方就是不写 where 子句，这样的话会更新掉全表的数据，**一个技巧就是把 set 中的值复制到 where 子句中即可。**

另外，建议大家在做多表关联更新的时候修改为 merge 语句，因为 merge into 语句只访问了一次表：

```
[oracle@rhel6_lhr ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.3.0 Production on 星期二 5月 19 10:26:55 2015

Copyright (c) 1982, 2011, Oracle. All rights reserved.

连接到:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

10:26:55 SQL> set autot on;
10:28:05 SQL> alter table lhr.emp_bk add dname varchar2(50) default 'noname';

表已更改。

已用时间: 00: 00: 01.23
10:30:04 SQL> update lhr.emp_bk a
10:30:09 2      set a.dname =(select b.dname from lhr.dept_bk b where b.deptno=a.deptno and b.dname in ('ACCOUNTING','RESERCH'))
10:30:09 3 WHERE EXISTS (select 1 from lhr.dept_bk b where b.deptno=a.deptno and b.dname in ('ACCOUNTING','RESERCH' ))
10:30:09 4 ;

已更新 3 行。

已用时间: 00: 00: 00.05

执行计划
-----
Plan hash value: 3525057516

-----
| Id | Operation | Name | Rows | Bytes | Cost (%CPU) | Time |
-----
| 0 | UPDATE STATEMENT | | 4 | 544 | 28 (18) | 00:00:01 |
| 1 | UPDATE | EMP_BK | | | | |
* | 2 | HASH JOIN SEMI | | 4 | 544 | 8 (13) | 00:00:01 |
| 3 | TABLE ACCESS FULL | EMP_BK | 14 | 1596 | 3 (0) | 00:00:01 |
* | 4 | TABLE ACCESS FULL | DEPT_BK | 1 | 22 | 4 (0) | 00:00:01 |
* | 5 | TABLE ACCESS FULL | DEPT_BK | 1 | 22 | 4 (0) | 00:00:01 |
-----

Predicate Information (identified by operation id):
-----

 2 - access("B"."DEPTNO"="A"."DEPTNO")
 4 - filter("B"."DNAME"='ACCOUNTING' OR "B"."DNAME"='RESERCH')
 5 - filter("B"."DEPTNO"=:B1 AND ("B"."DNAME"='ACCOUNTING' OR
      "B"."DNAME"='RESERCH'))

Note
-----
- dynamic sampling used for this statement (level=2)
```

统计信息

69	recursive calls
13	db block gets
121	consistent gets
9	physical reads
3012	redo size
837	bytes sent via SQL*Net to client
997	bytes received via SQL*Net from client
3	SQL*Net roundtrips to/from client
12	sorts (memory)
0	sorts (disk)
3	rows processed

已用时间: 00: 00: 00.00

```
10:33:13 SQL> merge into lhr.emp_bk a
10:33:32 2 using (select b.dname,deptno from lhr.dept_bk b where b.dname in ('ACCOUNTING','RESERCH')) bb
10:33:32 3 on (bb.deptno=a.deptno)
10:33:32 4 when matched then
10:33:32 5 update set a.dname =bb.dname
10:33:32 6 ;
```

3 行已合并。

已用时间: 00: 00: 00.03

执行计划

Plan hash value: 1386289611

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	MERGE STATEMENT		4	492	8 (13)	00:00:01
1	MERGE	EMP_BK				
2	VIEW					
* 3	HASH JOIN		4	592	8 (13)	00:00:01
* 4	TABLE ACCESS FULL	DEPT_BK	1	22	4 (0)	00:00:01
5	TABLE ACCESS FULL	EMP_BK	14	1764	3 (0)	00:00:01

Predicate Information (identified by operation id):

```
3 - access("DEPTNO"="A"."DEPTNO")
4 - filter("B"."DNAME"='ACCOUNTING' OR "B"."DNAME"='RESERCH')
```

Note

- dynamic sampling used for this statement (level=2)

统计信息

20	recursive calls
7	db block gets
38	consistent gets
1	physical reads
1872	redo size
838	bytes sent via SQL*Net to client
942	bytes received via SQL*Net from client
3	SQL*Net roundtrips to/from client
3	sorts (memory)
0	sorts (disk)
3	rows processed

10:33:32 SQL>

另外几篇关于使用 merge 语句来优化的案例：

update 修改为 merge (max+decode) ： <http://blog.itpub.net/26736162/viewspace-1244055/>

采用 merge 语句的非关联形式再次显神能 ： <http://blog.itpub.net/26736162/viewspace-1222423/>

采用 MERGE 语句的非关联形式提升性能 ： <http://blog.itpub.net/26736162/viewspace-1218671/>

采用 MERGE 语句的非关联形式提升性能 ---后传 ： <http://blog.itpub.net/26736162/viewspace-1222417/>

走了索引为啥还像蜗牛一样 ： <http://blog.itpub.net/26736162/viewspace-1208814/>

1.6 总结

到此 SQL 查询优化改写第四章基本 over，重点是对 merge 语句的领悟和掌握，尤其是哥列出的几个案例，希望对做 SQL 优化的童鞋有所帮助。

1.7 about me

.....

本文作者：小麦苗，只专注于数据库的技术，更注重技术的运用

ITPUB BLOG ： <http://blog.itpub.net/26736162>

本文地址： <http://blog.itpub.net/26736162/viewspace-1661906/>

本文pdf版： <http://yunpan.cn/QCwUAI9bn7g7w> 提取码：af2d

QQ：642808185 若加 QQ 请注明你所正在读的文章标题

创作时间地点：2015-05-19 09:00~ 2015-05-19 11:20 于外汇交易中心

<版权所有，文章允许转载，但须以链接方式注明源地址，否则追究法律责任!>

.....

