

::天意随缘，该到之时必将至

[主页](#)
[博客](#)
[相册](#)
[个人档案](#)
[好友](#)

[资源藏品](#)
[查看文章](#)

[转]Oracle中INITRANS和MAXTRANS参数

2009年04月15日 星期三 下午 12:57

每个块都有一个块首部。这个块首部中有一个事务表。事务表中会建立一些条目来描述哪些事务将块上的哪些行/元素锁定。这个事务表的初始大小由对象的INITRANS 设置指定。对于表，这个值默认为2（索引的INITRANS 也默认为2）。事务表会根据需要动态扩展，最大达到MAXTRANS 个条目（假设块上有足够的自由空间）。所分配的每个事务条目需要占用块首部中的23～24 字节的存储空间。注意，对于Oracle 10g，MAXTRANS 则会忽略，所有段的MAXTRANS 都是255。

也就是说，如果某个事物锁定了这个块的数据，则会在这个地方记录事务的标识，当然那个事务要先看一下这个地方是不是已经有人占用了，如果有，则去看看那个事务是否为活动状态。如果不活动，比如已经提交或者回滚，则可以覆盖这个地方。如果活动，则需要等待(门的作用)

所以，如果有大量的并发访问使用的这个块，则参数不能太小，否则资源竞争将导致系统并发性能下降。

测试了一下ORACLE 并发事务的时候的块分配和ITL 管理，略去大部分的测试过程,大概的结果小结如下:

1. INITRANS =1 时 并发多个INSERT 事务(本次测试最多5个)的时候并不会由于ITL的争用而等待阻塞,ORACLE 采取的策略是每个INSERT事物分配不同的一些块来使用,这样各个会话之间就不会产生冲突，除非段没有多余的块(次种情况与本次的主题无关).

2.INITRANS =1 时 并发多个UPDATE事务(本次测试最多7个)的时候也不会由于ITL的争用而导致等待产生,此时ORACLE除了使用默认的ITL之外，另外动态扩展所需要的ITL，紧紧在非常极端的情况下才会出现等待，(当然应用层面的死锁或等待与本主题无关)。

- 1) 该BLOCK没有FREE空间了，注意FREE 参数的设置不能太小。
- 2) 该块使用的ITL总数，超过该块允许的ITL的最大值min(round(block_size*0.5/24) - 2 ,255) 。

要达到这样的极端情况实际的生产情况是很难的，应该比业务SQL的死锁出现的概率更小。

小结：创建表的时候除非已经清楚，大部分的情况下没有必要调整INITRANS参数，通常1-4以下足够用了，INITRANS 设置非常大的时候ORACLE 有出现坏块的BUG，另外FREE 参数倒是要注意不能随意改小,除非你已经很清楚更改的后果。

参考部分INSERT 测试代码：(分3个SESSION insert的，x字段1开头的是SESSION 1 插入的，其他2个也类似)

SQL>Plus: Release 10.1.0.4.0 - Production on Wed Apr 16 10:40:17 2008
 Copyright (c) 1982, 2005, Oracle. All rights reserved.
 Connected to:
 Oracle Database 10g Enterprise Edition Release 10.1.0.4.0 - 64bit Production
 With the Partitioning, Real Application Clusters and Data Mining options

```

SQL> create table xx (x number) storage(initial 64k next 64k) initrans 2
Table created.
SQL> truncate table xx;
Table truncated.
SQL> select dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx;
no rows selected
SQL> INSERT INTO XX SELECT 11 FROM DUAL;
1 row created.
SQL>
SQL> set linesize 200
SQL> column SEGMENT_NAME format a16
select SEGMENT_NAME,EXTENT_ID,BLOCKS,BYTES from user_extents where segment_name ='XX';SQL>

SEGMENT_NAME      EXTENT_ID    BLOCKS    BYTES
-----
XX                0           128    1048576

SQL> select x ,dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx;

      X    FILE#    BLOCK#
-----
      11         51       4590

```

SQL> INSERT INTO XX SELECT 12 FROM DUAL;

```
1 row created.

SQL> select x ,dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx;

      X  FILE#  BLOCK#
-----
      12      51    4534
      11      51    4590

SQL>
SQL> INSERT INTO XX SELECT 13 from dual;

1 row created.

SQL> select x ,dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx;

      X  FILE#  BLOCK#
-----
      12      51    4534
      11      51    4590
      13      51    4658

SQL> INSERT INTO XX SELECT 14 from dual;

1 row created.

SQL> select x ,dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx;

      X  FILE#  BLOCK#
-----
      12      51    4534
      11      51    4590
      13      51    4658
      14      51    4658

SQL> select x ,dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx;

      X  FILE#  BLOCK#
-----
      12      51    4534
      11      51    4590
      13      51    4658
      14      51    4658

SQL> commit;

Commit complete.

SQL> INSERT INTO XX SELECT 15 from dual;

1 row created.

SQL> insert into xx select 16 from dual;

1 row created.

SQL> commit;

Commit complete.

SQL> select x ,dbms_rowid.rowid_relative_fno(rowid) file#, dbms_rowid.rowid_block_number(rowid) block# from xx order by x;

      X  FILE#  BLOCK#
-----
      11      51    4590
      12      51    4534
      13      51    4658
      14      51    4658
      15      51    4658
      16      51    4658

      21      51    4537
      22      51    4537
      23      51    4537
      24      51    4537
      25      51    4537
      26      51    4537

      31      51    4680
      32      51    4614
      33      51    4614
      34      51    4614
      35      51    4614
      36      51    4614

18 rows selected.
```

相关文章：

- [Oracle学习笔记:INTRANS和MAXTR...](#)

最近读者：



[登录](#)后，您就出现在这里。



[ahul206](#)



[helicopter1](#)



[杀无血刃](#)



[zhuwenfen](#)



[sevenqxy](#)



[yonered](#)

[g518](#)

网友评论：

发表评论：

姓 名：[注册](#) | [登录](#)

网址或邮箱： (选填)

内 容： 插入表情

验证码： 请点击后输入四位验证码，字母不区分大小写