文章编号: 1008-1542(2005) 03-0245-04

Oracle 9i 数据库中自动 Undo 表空间管理

刘俊英[1], 陈军霞[2]

(1. 石家庄邮电职业技术学院计算机系, 河北石家庄　050021; 2. 河北科技大学经济管理学院, 河北石家庄　050018)

摘　要: 在 Oracle 9i 数据库中, 用自动 Undo 管理(Automatic Undo Management) 代替了以前版本中回退段的手工管理, 即使用 Undo 表空间, 并自动管理该表空间, 使回退段的管理简单化, 主要探讨了 Oracle 9i 数据库 Undo 表空间的概念、创建、管理和维护等问题。

关键词: Oracle 数据库; Undo 表空间; 回退段; 实例

中图分类号: TP311. 132　　文献标识码: A

About Oracle 9i database's automatic Undo tablespace management

LIU Jun-ying[1], CHEN Jun-xia[2]

(1. Department of Computer, Shijiazhuang Vocational Technical College of Post and Telecommunication, Shijiazhuang Hebei 050021, China; 2. College of Economics and Management, Hebei University of Science and Technology, Shijiazhuang Hebei 050018, China)

Abstract: In Oracle 9i database, automatic Undo management which automatically manage Undo tablespace has replaced rollback segment manual management in previous version. This simplifies rollback segment management. This paper discusses Undo tablespace's concept, creation, management and maintennance.

Key words: Oracle database; Undo tablespace; rollback segment; instance

Oracle 关系数据库以其卓越的性能获得了广泛的应用, Oracle 9i 是 Oracle 数据库的较新版本之一。在以前的版本中, 对于回退段的手工设置与管理是一个困扰数据库管理员的难题。在 Oracle 9i 数据库中, 用自动 Undo 管理(Automatic Undo Management) 代替了回退段的手工管理, 即使用 Undo 表空间, 并自动管理该表空间, 使回退段的管理简单化。

1　自动 Undo 表空间管理概述

回退段用于存放数据修改之前的值(包括数据修改之前的位置和值) , 见图 1。该技术允许在特定条件下回退数据, 这些信息用于生成读一致性、在数据库恢复时回退未提交的事务。1 个事务只能使用 1 个回退段来存放它的回退信息, 而 1 个回退段可以存放多个事务的回退信息。对于大的事务常常由于回退段不足而被终止, 或在 RBS 表空间的回退段中消耗过大, 或出现回退段的争用问题, 所以在 Oracle 9i 之前的数据库中, 对于回退段的设计、管理与维护非常复杂。在 Oracle 9i 中使用 Undo 表空间, 系统对 Undo 表空间自动管理, 不再需要数据库管理员去计算、调整回退段数量及大小, 回退段的空间利用率、竞争等问题都由系统自动管理, 使得回退段的管理走向简单化[1]。

收稿日期: 2004-09-29; 修回日期: 2004-11-29; 责任编辑: 张　军

作者简介: 刘俊英(1972-) , 女, 河北沧州人, 讲师, 主要从事数据库设计与应用方面的研究。

© 1994-2013 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

Oracle 9i 数据库中可以创建多个 Undo 表空间，1 个实例只对应 1 个 Undo 表空间，即同一时间只允许 1 个被激活，该表空间的组织与管理由 Oracle 内部自动完成，回退段的管理由数据库实例自动进行。当 1 个回退段不足时，1 个事务可以使用多个回退段，Oracle 会自动使用其他回退段，不终止事务的运行。作为 Oracle 9i 的数据库管理员只需了解 Undo 表空间是否有足够的空间，而不必为每一个事务设置回退段。

当然，在 Oralce 9i 中，除使用 Undo 表空间处理用户数据的回退问题外，也仍然支持使用传统的回退段保持数据一致性及事务的回退。

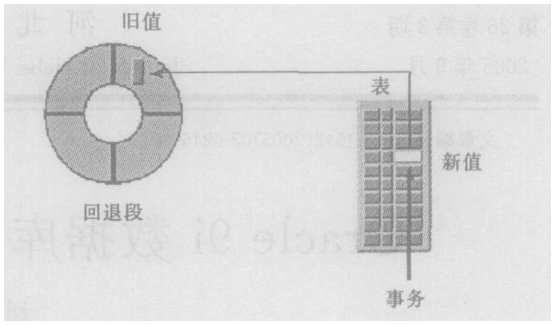

图 1 回退段

Fig. 1 Rollback segment

2 自动 Undo 表空间管理参数设置

在 Oracle 9i 中，系统默认使用 Undo 表空间自动管理。使用 Undo 表空间自动管理，需要设置以下参数。

1) Undo_Management 确定 Undo 表空间的管理方式。设置为 AUTO 表示系统使用自动 Undo 表空间管理，需要 1 个 Undo 表空间；设置为 MANUAL 表示使用手工方式创建、维护回退段，以回退段方式启动数据库，类似 Oracle 9i 以前的版本。

2) Undo_Tablespace 表示当使用 Undo 表空间自动管理时，系统默认使用的 Undo 表空间名称，默认为 Undotbs，也可以指定其他的 Undo 表空间。此参数可以在参数文件中设定，数据库启动后也允许动态地修改：SQL> ALTER SYSTEM SET undo_tablespace = UNDOTBS。

3) Undo_retention 决定 Undo 数据的维持时间，即用户事务结束后，为保证读一致性 Undo 数据保留的时间。这要求 Undo 表空间有大的空间来保存这些 retention 的数据。该参数以秒计，可以在参数文件中设定，以后也可以动态地修改：SQL> ALTER SYSTEM SET UNDO_RETENTION= 900。

3 Undo 表空间的创建

3.1 创建数据库时创建 Undo 表空间

在参数文件中设置如下参数：

```
# 系统管理的撤销和回退段
Undo_management= AUTO
```

这样，数据库依此参数文件启动数据库至 unmount 状态，然后创建数据库，使用 Undo tablespace undotbs datafile 子句指出在创建数据库时，自动创建 Undo 表空间。如：

```
SQL> CREATE DATABASE db01
…
UNDO TABLESPACE undo 1
DATA FILE '/u01/ oradata/ undo 1 db01. dbf' SIZE 200M
AUTOEXTEND ON
…
```

如果在参数文件中设置了自动 Undo 表空间管理，但在 CREATE DATABASE 语句中没有 Undo tablespace 子句，则 Oracle 自动创建名称为 sys_undotbs 的 Undo 表空间。sys_undotbs 表空间对应的数据文件是：'dbu1< oracle_sid> . dbf'，位于$ ORACLE_HOME/dbs 目录下，空间大小区分不同的操作系统，AUTOEXTEND 被设置为 ON。

3.2 命令行方式创建 Undo 表空间

若在创建数据库时没有创建 Undo 表空间，那么可以在数据库创建后手工创建。如：

© 1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

```
SQL> CREATE UNDO TABLESPACE undo 1
DATA FILE '/u01/ oradata/ undo 1 db01. dbf' SIZE 200M
Autoextend on next 100 maxsize unlimited
Extend management local autoallocate
```

在创建 Undo 表空间时, 不得使用数据字典管理该表空间, 它只适用于本地化管理的方式。新创建的 Undo 表空间在没有被激活前是不能被使用的。

4　Undo 表空间的维护

4.1　修改 Undo 表空间

可以为 Undo 表空间进行以下修改:

```
ADD DATAFILE
RENAME
DATAFILE [ONLINE|OFFLINE]
BEGIN BACKUP
END BACKUP
```

如为 Undo 表空间增加一个数据文件:

```
SQL> ALTER TABLESPACE undotbs1
ADD DATAFILE '/u01/ oradata/ undotbs2. dbf'
SIZE 30M
AUTOEXTEND ON
```

4.2　激活 Undo 表空间

对一个数据库而言, 可以有多于 1 个的 Undo 表空间, 但同一时刻只能有 1 个被激活使用。如果想使用另外 1 个 Undo 表空间, 必须用以下命令进行设置:

```
SQL> ALTER SYSTEM SET UNDO_TABLESPACE= UNDOTBS2
```

4.3　删除 Undo 表空间

系统正在使用的 Undo 表空间是不得删除的, 只有未激活的 Undo 表空间才能被删除。如:

```
SQL>  DROP TABLESPACE UNDOTBS2
```

如果删除正在使用的 Undo 表空间, 必须首先把数据库实例指向其他的 Undo 表空间, 然后等待所有正使用该 Undo 表空间的事务结束后方可删除该表空间。使用下面的查询语句看是否在该表空间上仍然有活动的事务:

```
SQL>  SELECT  a. name, b. status
    2  FROM v$ rollname a,  v$ rollstat b
    3  WHERE a. name IN ( SELECT segment_name)
    4  FROM dba_segments
    6   AND a. usn =  b. usn;
NAME                          STATUS
- - - - - - - - - - - - - - - - - - - - - - -
_SYSSMU4$                     PENDING OFFLINE
```

这表明仍然有活动的事务使用一个段。当该查询不再返回任何数据时, 说明所有事务已完成, 这时才可以删除 Undo 表空间。

4.4　监控 Undo 表空间

系统每隔 10 min 采 1 次样, 并在动态性能视图 v$ undostat 中产生 1 行记录, 说明 10 min 内的 Undo 信息总和。数据库管理员定期查询一下此视图, 掌握 Undo 的使用信息。

```
SQL> SELECT end_time, begin_time, undoblks
FROM v$ undostat;
```

© 1994-2013 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

```
END_TIME          BEGIN_TIME          UNDOBLKS
- - - - - - - - - - - - - - - - - - - - - - -
22-JAN-01 13: 44: 18 22-JAN-01 13: 43: 04 19
22-JAN-01 13: 43: 04 22-JAN-01 13: 33: 04 1474
22-JAN-01 13: 33: 04 22-JAN-01 13: 23: 04 1347
22-JAN-01 13: 23: 04 22-JAN-01 13: 13: 04 1628
22-JAN-01 13: 13: 04 22-JAN-01 13: 03: 04 2249
22-JAN-01 13: 03: 04 22-JAN-01 12: 53: 04 1698
22-JAN-01 12: 53: 04 22-JAN-01 12: 43: 04 1433
22-JAN-01 12: 43: 04 22-JAN-01 12: 33: 04 1532
22-JAN-01 12: 33: 04 22-JAN-01 12: 23: 04 1075
...
```

这表示每一个时间段使用的 Undo 块, 如在 13: 23 到 13: 33 时间内共使用了 1 347 个 Undo 块。尽管时间间隔是 10 min, 但最近一行的信息返回的是从 1 个间隔点到目前时间的信息。

5 确定所需的 **Undo** 表空间的大小

确定 Undo 表空间的空间大小需要 3 个参数。

UR: UNDO_RETENTION 大小;

UPS: 每秒钟产生的 Undo 数据块的数目;

DBS: db_block_size。

其中 UR, DBS 可以通过查参数文件得到; 而 UPS 可以通过查询动态性能视图 v$ undostat 得到。

```
SQL> SELECT (SUM(undoblks) / SUM
  2  ((end_time-begin_time) * 86400)
  3  FROM v$ undostat;
```

利用下面的查询语句求出 Undo 表空间大概的所需字节数。

```
SQL> SELECT (UR * (UPS * DBS)) + (DBS * 24) AS "Bytes"
  2  FROM (SELECT value AS UR
  3  FROM v$ parameter
  4  WHERE name = 'undo_retention'),
  5  (SELECT (SUM(undoblks)/SUM
  6  (((end_time-begin_time)* 86400))) AS UPS
  7  FROM v$ undostat),
  8  (SELECT value AS DBS
  9  FROM v$ parameter
 10 WHERE name = 'db_block_size');
Bytes
- - - - - - - - - -
19106213
```

其中, (DBS* 24) 适用于保存系统信息的额外开销, 如分批管理用的位图、事务表等[2]。

如想获得较理想的结果, 应该在 1 d 中系统比较繁忙, 业务量较大的时候计算该值, 以供管理员调整 Undo 表空间的大小, 使空间大小能承载大的事务量。

参考文献:

[1] COUCHMAN J S, MARISETTI S N. Oracle 9i Database Fundamentals Ⅱ[M]. 钟 鸣, 石永平译. 北京: 机械工业出版社, 2003.

[2] 滕永昌. Oracle 9i 数据库管理员使用大全[M]. 北京: 清华大学出版社, 2004.

[3] 北京计算机教育培训中心. Oracle 9i 数据库系统培训教程[M]. 北京: 清华大学出版社, 2003.

© 1994-2013 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net