2018年8月17日

11:54

**监控和管理Oracle UNDO表空间的使用**

原创 作者：[尛样儿](http://blog.itpub.net/23135684/) 时间：2013-12-30 13:41:52 42794 2

    对Oracle数据库UNDO表空间的监控和管理是我们日常最重要的工作之一，UNDO表空间通常都是Oracle自动化管理（通过undo\_management初始化参数确定）；UNDO表空间是用于存储DML操作的前镜像数据，它是实例恢复，数据回滚，一致性查询功能的重要组件；我们常常会忽略对它的监控，这会导致UNDO表空间可能出现以下问题：

1).空间使用率100%，导致DML操作无法进行。

2).告警日志中出现大量的ORA-01555告警错误。

3).实例恢复失败，数据库无法正常打开。

**一.对Oracle自动化管理UNDO进行干预。**

   由于UNDO是自动化管理，可干预的地方非常的少，更多的是监控，通过以下几个地方可对UNDO表空间实施一定的干预：

1).初始化参数

undo\_management=AUTO     表示实例自动化管理UNDO表空间，从Oracle 9i开始，Oracle引进了AUM（Automatic Undo Management）。

undo\_retention=900              事务提交后，相应的UNDO数据保留的时间，单位：秒。

undo\_tablespace=UNDOTBS1 活动的UNDO表空间。

\_smu\_debug\_mode=33554432

\_undo\_autotune=TRUE

2).Automatic UNDO Retention

    Automatic UNDO Retention是10g的新特性，在10g和之后的版本的数据库，这个特性是默认启用的。

    在Oracle Database 10g中当自动undo管理被启用，总是存在一个当前的undo retention，Oracle Database尝试至少保留旧的undo信息到该时间。数据库收集使用情况统计信息，基于这些统计信息和UNDO表空间大小来调整undo retention的时间。

    Oracle Database基于undo表空间大小和系统活动自动调整undo retention，通过设置UNDO\_RETENTION初始化参数指定undo retention的最小值。

    查看Oracle自动调整UNDO RETENTION的值可以通过以下查询获得：

SELECT TO\_CHAR(BEGIN\_TIME, 'MM/DD/YYYY HH24:MI:SS') BEGIN\_TIME,

    TUNED\_UNDORETENTION FROM V$UNDOSTAT;

    针对自动扩展的UNDO表空间，系统至少保留UNDO到参数指定的时间，自动调整UNDO RETENTION以满足查询对UNDO的要求，这可能导致UNDO急剧扩张，可以考虑不设置UNDO RETENTION值。

    针对固定的UNDO表空间，系统根据最大可能的undo retention进行自动调整，参考基于UNDO表空间大小和使用历史进行调整，这将忽略UNDO\_RETENTION,除非表空间启用了RETENTION GUARANTEE。

自动调整undo retention不支持LOB，因为不能在undo表空间中存储任何有关LOBs事务的UNDO信息。

可以通过设置\_undo\_autotune=FALSE显示的关闭Automatic UNDO Retention功能。

3).TUNED\_UNDORETENTION计算的值很大导致UNDO表空间增长很快？

    当使用的UNDO表空间非自动增长，tuned\_undoretention是基于UNDO表空间大小的使用率计算出来的，在一些情况下，特别是较大的UNDO表空间时，这将计算出较大的值。

为了解决此行为，设置以下的实例参数：

\_smu\_debug\_mode=33554432

设置该参数，TUNED\_UNDORETENTION就不基于undo表空间大小的使用率计算，代替的是设置(MAXQUERYLEN +300)和UNDO\_RETENTION的最大值。

4).UNDO表空间数据文件自动扩展

    如果UNDO表空间是一个自动扩展的表空间，那么很有可能UNDO表空间状态为EXPIRED的EXTENT不会被使用（这是为了减少报ORA-01555错误的几率），这将导致UNDO表空间变得很大；如果将UNDO表空间设置为非自动扩展，那么状态为EXPIRED的EXTENT就能被利用，这样可以一定程度控制UNDO表空间的大小，但这样会增加ORA-01555报错和UNDO空间不足报错的风险。合理的非自动扩展的UNDO表空间大小，以及合理的UNDO\_RETENTION设置可以确保稳定的UNDO空间使用。

5).UNDO表空间guarantee属性

    如果UNDO表空间是noguarantee状态，Oracle不确保提交后的事务对应的UNDO表空间中的数据会保留UNDO\_RETENTION指定的时长，如果UNDO表空间不足，其他事务将可能偷盗相应的未过期的空间；将UNDO表空间设置为guarantee能够确保提交后的事务对应UNDO表空间中的数据在任何情况下都将保留UNDO\_RETENTION指定的时长。

SQL> SELECT tablespace\_name, retention FROM dba\_tablespaces where tablespace\_name='UNDOTBS1';

TABLESPACE\_NAME                                              RETENTION

------------------------------------------------------------ ----------------------

UNDOTBS1                                                     NOGUARANTEE

SQL> alter tablespace undotbs1 retention guarantee;

表空间已更改。

SQL> SELECT tablespace\_name, retention FROM dba\_tablespaces where tablespace\_name='UNDOTBS1';

TABLESPACE\_NAME                                              RETENTION

------------------------------------------------------------ ----------------------

UNDOTBS1                                                     GUARANTEE

6).UNDO表空间大小

   针对不同类型的业务系统，需要有充足的UNDO表空间，确保系统能够正常的运行。UNDO空间的大小跟业务系统有关系，也跟UNDO\_RETENTION和UNDO表空间的GUARANTEE属性有关系，通常我们可以通过V$UNDOSTAT的统计信息估算出需要的UNDO表空间大小。

**二.监控UNDO表空间使用情况。**

   作为管理员来说，针对UNDO表空间更重要的是日常的监控工作，监控常用到以下的视图：

a).DBA\_ROLLBACK\_SEGS

DBA\_ROLLBACK\_SEGS describes rollback segments.

b).V$ROLLSTAT

V$ROLLSTAT contains rollback segment statistics.

c).V$TRANSACTION

V$TRANSACTION lists the active transactions in the system.

d).V$UNDOSTAT

V$UNDOSTAT displays a histogram of statistical data to show how well the system is working. The available statistics include undo space consumption, transaction concurrency, and length of queries executed in the instance. You can use this view to estimate the amount of undo space required for the current workload. Oracle uses this view to tune undo usage in the system. The view returns NULL values if the system is in manual undo management mode.

Each row in the view keeps statistics collected in the instance for a 10-minute interval. The rows are in descending order by the BEGIN\_TIME column value. Each row belongs to the time interval marked by (BEGIN\_TIME, END\_TIME). Each column represents the data collected for the particular statistic in that time interval. The first row of the view contains statistics for the (partial) current time period. The view contains a total of 576 rows, spanning a 4 day cycle.

e).DBA\_UNDO\_EXTENTS

DBA\_UNDO\_EXTENTS describes the extents comprising the segments in all undo tablespaces in the database.  This view shows the status and size of each extent in the undo tablespace.

DBA\_UNDO\_EXTENTS.STATUS有三个值：

ACTIVE      表示未提交事务还在使用的UNDO EXTENT，该值对应的UNDO SEGMENT的DBA\_ROLL\_SEGMENTS.STATUS一定是ONLINE或PENDING OFFLINE状态，一旦没有活动的事务在使用UNDO SEGMENT，那么对应的UNDO SEGMENT就变成OFFLINE状态。

EXPIRED     表示已经提交且超过了UNDO\_RETENTION指定时间的UNDO EXTENT。

UNEXPIRED 表示已经提交但是还没有超过UNDO\_RETENTION指定时间的UNDO EXTENT。

   Oracle重复使用UNDO EXTENT的原则如下：

1).ACTIVE状态的EXTENT在任何情况下都不会被占用。

2).如果是自动扩展的UNDO表空间，Oracle会保证EXTENT至少保留UNDO\_RETENTION指定的时间。

3).如果自动扩展空间不足或者UNDO表空间是非自动扩展，Oracle会尝试重复使用同一个段下面EXPIRED状态的EXTENT，如果本段中没有这样的EXTENT，就会去偷别的段下面EXPIRED状态的EXTENT，如果依然没有这样的EXTENT，就会使用本段UNEXPIRED的EXTENT，如果还是没有，那么会去偷别的段的UNEXPIRED的EXTENT，这个都没有，就会报错。

1.UNDO表空间空间使用情况。

1).UNDO表空间总大小。

   UNDO表空间下也以段的形式存储数据，每个事务对应一个段，这种类型的段通常被称为回滚段，或者UNDO段。默认情况下，数据库实例会初始化10个UNDO段，这主要是为了避免新生成的事务对UNDO段的争用。

UNDO表空间的总大小就是UNDO表空间下的所有数据文件大小的总和：

SQL> select tablespace\_name,contents from dba\_tablespaces where tablespace\_name='UNDOTBS1';

TABLESPACE\_NAME                                              CONTENTS

------------------------------------------------------------ ------------------

UNDOTBS1                                                     UNDO

SQL> select tablespace\_name,sum(bytes)/1024/1024 mb from dba\_data\_files where tablespace\_name='UNDOTBS1'  group by tablespace\_name;

TABLESPACE\_NAME                                                      MB

------------------------------------------------------------ ----------

UNDOTBS1                                                             90

2).查看UNDO表空间的使用情况。

    该使用情况可以通过两个视图来查看：

SQL> select owner,segment\_name,bytes/1024/1024 mb from dba\_segments where tablespace\_name='UNDOTBS1';

OWNER      SEGMENT\_NAME                           MB

---------- ------------------------------ ----------

SYS        \_SYSSMU12\_2867006942$                .125

SYS        \_SYSSMU11\_3120896088$                .125

SYS        \_SYSSMU10\_1735367849$               2.125

SYS        \_SYSSMU9\_3051513041$                2.125

SYS        \_SYSSMU8\_2280151962$                2.125

SYS        \_SYSSMU7\_825858386$                 .9375

SYS        \_SYSSMU6\_2597279618$                3.125

SYS        \_SYSSMU5\_247215464$                 3.125

SYS        \_SYSSMU4\_437228663$                 2.125

SYS        \_SYSSMU3\_3104504842$                5.125

SYS        \_SYSSMU2\_2464850095$                2.125

SYS        \_SYSSMU1\_2523538120$                3.125

已选择12行。

SQL>  select segment\_name, v.rssize/1024/1024 mb

  2    From dba\_rollback\_segs r, v$rollstat v

  3    Where r.segment\_id = v.usn(+)

  4    order by segment\_name ;

SEGMENT\_NAME                           MB

------------------------------ ----------

SYSTEM                           .3671875

\_SYSSMU10\_1735367849$           2.1171875

\_SYSSMU11\_3120896088$

\_SYSSMU12\_2867006942$

\_SYSSMU1\_2523538120$            3.1171875

\_SYSSMU2\_2464850095$            2.1171875

\_SYSSMU3\_3104504842$            5.1171875

\_SYSSMU4\_437228663$             2.1171875

\_SYSSMU5\_247215464$             3.1171875

\_SYSSMU6\_2597279618$            3.1171875

\_SYSSMU7\_825858386$              .9296875

\_SYSSMU8\_2280151962$            2.1171875

\_SYSSMU9\_3051513041$            2.1171875

已选择13行。

    通过上面的两个查询可以看出，两个视图查询的值几乎一致，通常在巡检的时候，我们习惯查询dba\_segments视图来确定UNDO表空间的使用情况，但查询V$ROLLSTAT数据更加准确。

3).查询事务使用的UNDO段及大小。

    很多客户想知道，我的UNDO表空间超过了90%，是哪些会话的事务占用了这些空间：

SQL>  select s.sid,s.serial#,s.sql\_id,v.usn,segment\_name,r.status, v.rssize/1024/1024 mb

  2    From dba\_rollback\_segs r, v$rollstat v,v$transaction t,v$session s

  3    Where r.segment\_id = v.usn and v.usn=t.xidusn and t.addr=s.taddr

  4    order by segment\_name ;

       SID    SERIAL# SQL\_ID        USN SEGMENT\_NAME                                         STATUS                                   MB

---------- ---------- -------------------------- ---------- ------------------------------------------------------------ -------------------------------- ----------

         8        163                                     5 \_SYSSMU5\_247215464$                                  ONLINE                            3.1171875

    通过这个SQL语句可以查询到会话对应的活动事务使用的UNDO段名称，以及该段占用的UNDO空间大小，对于非活动事务占用了UNDO空间是由Oracle实例根据参数配置自动化管理的。

2.根据Oracle对UNDO表空间的统计信息调整UNDO参数及大小。

    最后我们要谈谈V$UNDOSTAT视图，该视图的作用是用于指导管理员调整UNDO表空间的参数及表空间大小，每行表示的是10分钟的数据，最多可保留576行，4天一个周期，如果该视图没有数据，那么UNDO可能是手动管理方式。下面对该视图字段的含义进行说明：

|  |  |  |
| --- | --- | --- |
| BEGIN\_TIME | DATE | Identifies the beginning of the time interval  时间间隔开始时间。 |
| END\_TIME | DATE | Identifies the end of the time interval    时间间隔结束时间。 |
| UNDOTSN | NUMBER | Represents the last active undo tablespace in the duration of time. The tablespace ID of the active undo tablespace is returned in this column. If more than one undo tablespace was active in that period, the active undo tablespace that was active at the end of the period is reported.  时间间隔活动的UNDO表空间个数，返回的是活动UNDO表空间的ID号，如果大于1个活动的UNDO表空间，将报告在时间间隔最后被激活的UNDO表空间ID号。 |
| UNDOBLKS | NUMBER | Represents the total number of undo blocks consumed. You can use this column to obtain the consumption rate of undo blocks, and thereby estimate the size of the undo tablespace needed to handle the workload on your system.  表示总共消费的UNDO块数，可以使用这个字段获得undo块的消费比率，由此来估算处理系统负载需要的UNDO表空间大小。 |
| TXNCOUNT | NUMBER | Identifies the total number of transactions executed within the period  在这个时期内总共执行的事务数。 |
| MAXQUERYLEN | NUMBER | Identifies the length of the longest query (in seconds) executed in the instance during the period. You can use this statistic to estimate the proper setting of theUNDO\_RETENTION initialization parameter. The length of a query is measured from the cursor open time to the last fetch/execute time of the cursor. Only the length of those cursors that have been fetched/executed during the period are reflected in the view.  在这个时期该实例执行的最长查询时间（单位：秒），可以使用这个统计信息估算UNDO\_RETENTION初始化参数的大概值。查询的时间精确到从游标打开到最后提取/执行时间。只有当这些游标的查询时间在这个时期被提取/执行才能被反映到该视图。 |
| MAXQUERYID | VARCHAR2(13) | SQL identifier of the longest running SQL statement in the period  在这个时期运行最长时间的SQL语句标识符。 |
| MAXCONCURRENCY | NUMBER | Identifies the highest number of transactions executed concurrently within the period  在这个时期并行执行的最大事务数。 |
| UNXPSTEALCNT | NUMBER | Number of attempts to obtain undo space by stealing unexpired extents from other transactions  尝试从其他事务通过偷盗的方式获得的未过期的undo空间区间数。 |
| UNXPBLKRELCNT | NUMBER | Number of unexpired blocks removed from certain undo segments so they can be used by other transactions  从某些UNDO段移除未过期的块数，他们被用于其它事务。 |
| UNXPBLKREUCNT | NUMBER | Number of unexpired undo blocks reused by transactions  事务重新使用未过期的undo块数。 |
| EXPSTEALCNT | NUMBER | Number of attempts to steal expired undo blocks from other undo segments  尝试从其他UNDO段偷盗过期的UNDO块数。 |
| EXPBLKRELCNT | NUMBER | Number of expired undo blocks stolen from other undo segments  从其他UNDO段偷盗的过期的UNDO块数。 |
| EXPBLKREUCNT | NUMBER | Number of expired undo blocks reused within the same undo segments  在相同UNDO段重新使用的过期的UNDO块数。 |
| SSOLDERRCNT | NUMBER | Identifies the number of times the error ORA-01555occurred. You can use this statistic to decide whether or not the UNDO\_RETENTION initialization parameter is set properly given the size of the undo tablespace. Increasing the value of UNDO\_RETENTION can reduce the occurrence of this error.  标识ORA-01555错误发生的次数，可以使用这个统计信息决定针对给定的UNDO表空间是否设置UNDO\_RETENTION初始化参数。增加UNDO\_RETENTION的值可以减少这个错误的发生。 |
| NOSPACEERRCNT | NUMBER | Identifies the number of times space was requested in the undo tablespace and there was no free space available. That is, all of the space in the undo tablespace was in use by active transactions. The corrective action is to add more space to the undo tablespace.  在UNDO表空间没有自由空间活动的情况下，空间请求的次数，所有UNDO表空间的空间被活动的事务使用，这需要添加更多的空间到UNDO表空间。 |
| ACTIVEBLKS | NUMBER | Total number of blocks in the active extents of the undo tablespace for the instance at the sampled time in the period  在时间间隔，针对该实例，UNDO表空间活动区间的块个数。 |
| UNEXPIREDBLKS | NUMBER | Total number of blocks in the unexpired extents of the undo tablespace for the instance at the sampled time in the period  在时间间隔，针对该实例，UNDO表空间未过期的块个数。 |
| EXPIREDBLKS | NUMBER | Total number of blocks in the expired extents of the undo tablespace for the instance at the sampled time in the period  在时间间隔，针对该实例，UNDO表空间过期区间的块个数。 |
| TUNED\_UNDORETENTION | NUMBER | Amount of time (in seconds) for which undo will not be recycled from the time it was committed. At any point in time, the latest value of TUNED\_UNDORETENTION is used to determine whether data committed at a particular time in the past can be recycled.  提交之后UNDO不能被回收的总时间（单位：秒）。 |

下面是查询V$UNDOSTAT的例子：

SELECT TO\_CHAR(BEGIN\_TIME, 'MM/DD/YYYY HH24:MI:SS') BEGIN\_TIME,

  TO\_CHAR(END\_TIME, 'MM/DD/YYYY HH24:MI:SS') END\_TIME,

  UNDOTSN, UNDOBLKS, TXNCOUNT, MAXCONCURRENCY AS "MAXCON",

  MAXQUERYLEN, TUNED\_UNDORETENTION

  FROM v$UNDOSTAT;

通常当字段UNXPSTEALCNT和EXPBLKREUCNT是非零值，表示有空间压力。

如果字段SSOLDERRCNT是非零值，表示UNDO\_RETENTION设置不合理。

如果字段NOSPACEERRCNT是非零值，表示有一系列空间问题。

在10g DBA\_HIST\_UNDOSTAT视图包括了V$UNDOSTAT快照统计信息。

注意：如果参数\_undo\_autotune=FALSE,X$KTUSMST2将没有数据生成，该表是DBA\_HIST\_UNDOSTATS视图的源表。

**三.释放UNDO表空间。**

    UNDO表空间被撑得过大，有些时候我们需要释放这些空间，通常的做法是新建一个UNDO，然后设置使用新建的UNDO表空间，最后DROP原有UNDO表空间。下面通过一个例子来演示这个过程：

SQL> col segment\_name format a30

SQL> col tablespace\_name format a30

SQL>

SQL> select segment\_name, tablespace\_name, r.status,

  2    (initial\_extent/1024) InitialExtent,(next\_extent/1024) NextExtent,

  3    max\_extents, v.curext CurExtent

  4    From dba\_rollback\_segs r, v$rollstat v

  5    Where r.segment\_id = v.usn(+)

  6    order by segment\_name ;

SEGMENT\_NAME                   TABLESPACE\_NAME                STATUS                           INITIALEXTENT NEXTEXTENT MAX\_EXTENTS  CUREXTENT

------------------------------ ------------------------------ -------------------------------- ------------- ---------- ----------- ----------

SYSTEM                         SYSTEM                         ONLINE                                  112            56       32765          4

\_SYSSMU10\_1735367849$          UNDOTBS1                       ONLINE                                  128            64       32765          2

\_SYSSMU11\_3120896088$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU12\_2867006942$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU1\_2523538120$           UNDOTBS1                       ONLINE                                  128            64       32765          2

\_SYSSMU2\_2464850095$           UNDOTBS1                       ONLINE                                  128            64       32765          2

\_SYSSMU3\_3104504842$           UNDOTBS1                       ONLINE                                  128            64       32765          2

\_SYSSMU4\_437228663$            UNDOTBS1                       ONLINE                                  128            64       32765          2

\_SYSSMU5\_247215464$            UNDOTBS1                       ONLINE                                  128            64       32765          3

\_SYSSMU6\_2597279618$           UNDOTBS1                       ONLINE                                  128            64       32765          3

\_SYSSMU7\_825858386$            UNDOTBS1                       ONLINE                                  128            64       32765          9

\_SYSSMU8\_2280151962$           UNDOTBS1                       ONLINE                                  128            64       32765          3

\_SYSSMU9\_3051513041$           UNDOTBS1                       ONLINE                                  128            64       32765          2

已选择13行。

当前所有的回滚段在属于UNDOTBS1表空间。

SQL> create undo tablespace undotbs2 datafile 'E:\APP\ORADATA\ORCL3\undotbs02.dbf' size 20m autoextend on next 100m;

表空间已创建。

SQL> show parameter undo

NAME                                 TYPE                   VALUE

------------------------------------ ---------------------- ------------------------------

undo\_management                      string                 AUTO

undo\_retention                       integer                900

undo\_tablespace                      string                 UNDOTBS1

SQL> alter system set undo\_tablespace='UNDOTBS2';

系统已更改。

SQL> show parameter undo

NAME                                 TYPE                   VALUE

------------------------------------ ---------------------- ------------------------------

undo\_management                      string                 AUTO

undo\_retention                       integer                900

undo\_tablespace                      string                 UNDOTBS2

SQL> select segment\_name, tablespace\_name, r.status,

  2    (initial\_extent/1024) InitialExtent,(next\_extent/1024) NextExtent,

  3    max\_extents, v.curext CurExtent

  4    From dba\_rollback\_segs r, v$rollstat v

  5    Where r.segment\_id = v.usn(+)

  6    order by segment\_name ;

SEGMENT\_NAME                   TABLESPACE\_NAME                STATUS                           INITIALEXTENT NEXTEXTENT MAX\_EXTENTS  CUREXTENT

------------------------------ ------------------------------ -------------------------------- ------------- ---------- ----------- ----------

SYSTEM                         SYSTEM                         ONLINE                                  112            56       32765          5

\_SYSSMU10\_1735367849$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU11\_3120896088$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU12\_2867006942$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU13\_3398750080$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU14\_3208386744$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU15\_2082453576$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU16\_2746861185$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU17\_3752120760$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU18\_3475721077$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU19\_1407063349$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU1\_2523538120$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU20\_910603223$           UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU21\_1261247597$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU22\_1117177365$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU2\_2464850095$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU3\_3104504842$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU4\_437228663$            UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU5\_247215464$            UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU6\_2597279618$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU7\_825858386$            UNDOTBS1                       ONLINE                                  128            64       32765          9

\_SYSSMU8\_2280151962$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU9\_3051513041$           UNDOTBS1                       OFFLINE                                 128            64       32765

已选择23行。

    虽然将数据库实例使用的UNDO表空间指向了新表空间，但是依然有过去的事务在使用UNDOTBS1表空间下面的段，这个时候不能直接DROP UNDOTBS1（执行DROP命令也会报错），必须等待UNDOTBS1表空间下的所有段状态变成OFFLINE才能DROP。

SQL> r

  1  select segment\_name, tablespace\_name, r.status,

  2    (initial\_extent/1024) InitialExtent,(next\_extent/1024) NextExtent,

  3    max\_extents, v.curext CurExtent

  4    From dba\_rollback\_segs r, v$rollstat v

  5    Where r.segment\_id = v.usn(+)

  6\*   order by segment\_name

SEGMENT\_NAME                   TABLESPACE\_NAME                STATUS                           INITIALEXTENT NEXTEXTENT MAX\_EXTENTS  CUREXTENT

------------------------------ ------------------------------ -------------------------------- ------------- ---------- ----------- ----------

SYSTEM                         SYSTEM                         ONLINE                                  112            56       32765          5

\_SYSSMU10\_1735367849$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU11\_3120896088$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU12\_2867006942$          UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU13\_3398750080$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU14\_3208386744$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU15\_2082453576$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU16\_2746861185$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU17\_3752120760$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU18\_3475721077$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU19\_1407063349$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU1\_2523538120$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU20\_910603223$           UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU21\_1261247597$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU22\_1117177365$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU2\_2464850095$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU3\_3104504842$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU4\_437228663$            UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU5\_247215464$            UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU6\_2597279618$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU7\_825858386$            UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU8\_2280151962$           UNDOTBS1                       OFFLINE                                 128            64       32765

\_SYSSMU9\_3051513041$           UNDOTBS1                       OFFLINE                                 128            64       32765

已选择23行。

    UNDOTBS1表空间下的所有段状态都变成了OFFLINE，这个时候可以DROP UNDOTBS1来释放空间。

SQL> drop tablespace undotbs1 including contents and datafiles;

表空间已删除。

    虽然能DROP，只是说明没有事务在使用旧的UNDO表空间，这并不表示所有的UNDO EXTENT已经过期（DBA\_UNDO\_EXTENTS.STATUS），如果有某些查询需要用到这些存储在旧UNDO表空间上过期或未过期的EXTENT时，将收到ORA-01555的报错。

SQL> select segment\_name, tablespace\_name, r.status,

  2    (initial\_extent/1024) InitialExtent,(next\_extent/1024) NextExtent,

  3    max\_extents, v.curext CurExtent

  4    From dba\_rollback\_segs r, v$rollstat v

  5    Where r.segment\_id = v.usn(+)

  6    order by segment\_name ;

SEGMENT\_NAME                   TABLESPACE\_NAME                STATUS                           INITIALEXTENT NEXTEXTENT MAX\_EXTENTS  CUREXTENT

------------------------------ ------------------------------ -------------------------------- ------------- ---------- ----------- ----------

SYSTEM                         SYSTEM                         ONLINE                                  112            56       32765          5

\_SYSSMU13\_3398750080$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU14\_3208386744$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU15\_2082453576$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU16\_2746861185$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU17\_3752120760$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU18\_3475721077$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU19\_1407063349$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU20\_910603223$           UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU21\_1261247597$          UNDOTBS2                       ONLINE                                  128            64       32765          0

\_SYSSMU22\_1117177365$          UNDOTBS2                       ONLINE                                  128            64       32765          0

已选择11行。

   有益的补充《Oracle 11gR2 Database UNDO表空间使用率居高不下》：<http://blog.itpub.net/23135684/viewspace-1406011/>

    有关AUM更多详细的信息，请参考文章：

   《FAQ – Automatic Undo Management (AUM) / System Managed Undo (SMU) (文档 ID 461480.1)》

   《AUM 常用分析/诊断脚本 (文档 ID 1526122.1)》

--end--

源文档 <<http://blog.itpub.net/23135684/viewspace-1065601/>>