

# Ejercicio 2 - tema 7

## Almacenamiento en data files

Vicente Romero Andrade

### I. OBJETIVO

EL objetivo es poner en práctica las tareas de administración que permitan el almacenamiento de datos de una tabla en un tablespace y data file específico configurado previamente

### II. DESARROLLO

#### A. sentencias

```

1  whenever sqlerror exit rollback
2  set serveroutput on
3  connect sys/system2 as sysdba
4  --A
5  SELECT
6      FILE_NAME,
7      FILE_ID,
8      TRUNC((BYTES/(1024*1024)),2) SIZE_MB
9  FROM DBA_DATA_FILES WHERE TABLESPACE_NAME = '
10     STORE_TBS_MULTIPLE';
11 --B
12 SELECT
13     TRUNC((SUM(DF.BYTES)-NVL(SUM(S.BYTES),0))
14         /(1024*1024),2) MB_LIBRES,
15     (SUM(DF.BLOCKS)-NVL(SUM(S.BLOCKS),0))
16     BLOQUES_DISPONIBLES
17 FROM DBA_DATA_FILES DF
18 LEFT JOIN DBA_SEGMENTS S
19     ON S.TABLESPACE_NAME = DF.TABLESPACE_NAME
20 WHERE DF.TABLESPACE_NAME='STORE_TBS_MULTIPLE';
21 --C
22 --CREATE USER VRA_TBS_MULTIPLE IDENTIFIED BY
23     VRA_TBS_MULTIPLE
24     --quota unlimited on store_tbs_multiple
25     --default tablespace store_tbs_multiple;
26 --D
27 declare
28     v_count number;
29     v_username varchar2(30) := 'VRA_TBS_MULTIPLE';
30     v_table varchar2(30) := 'VRA_TBS_MULTIPLE';
31 begin
32     --Verificar si la table existe
33     select count(*) into v_count
34     from all_tables
35     where table_name = v_table
36     and owner = v_username;
37
38     if v_count > 0 then
39         execute immediate 'drop table '||v_username||'
40             .'||v_table;
41     end if;
42     execute immediate 'create table '||v_username||'
43         .'||v_table||' (
44             str char(1024 byte)
45         ) segment creation immediate';
46 end;
47 /
48 --E
49 SELECT DF.FILE_NAME,
50     DF.FILE_ID,
51     COUNT(DE.SEGMENT_NAME) NUMERO_EXTENSIONES,
52     SUM(DE.BYTES/(1024*1024)) TOTAL_MB,
53     SUM(DE.BLOCKS) BLOQUES_RESERVADOS
54 FROM DBA_SEGMENTS DS
55 JOIN DBA_DATA_FILES DF
56     ON DS.HEADER_FILE = DF.FILE_ID
57 JOIN DBA_DATA_FILES DF
58     ON DS.HEADER_FILE = DF.FILE_ID
59 JOIN DBA_EXTENTS DE
60     ON DS.SEGMENT_NAME = DE.SEGMENT_NAME
61 WHERE DS.SEGMENT_NAME like '%VRA_TBS_MULTIPLE%'
62 GROUP BY DF.FILE_NAME, DF.FILE_ID;
63 --F
64 declare
65     v_count number := 0;
66 begin
67     while v_count < 512 loop
68         insert into VRA_TBS_MULTIPLE.VRA_TBS_MULTIPLE (
69             str) values('S');
70         v_count := v_count + 1;
71     end loop;
72     commit;
73 end;
74 /
75 --G
76 SELECT DF.FILE_NAME,
77     DF.FILE_ID,
78     COUNT(DE.SEGMENT_NAME) NUMERO_EXTENSIONES,
79     SUM(DE.BYTES/(1024*1024)) TOTAL_MB,
80     SUM(DE.BLOCKS) BLOQUES_RESERVADOS
81 FROM DBA_SEGMENTS DS
82 JOIN DBA_DATA_FILES DF
83     ON DS.HEADER_FILE = DF.FILE_ID
84 JOIN DBA_DATA_FILES DF
85     ON DS.HEADER_FILE = DF.FILE_ID
86 JOIN DBA_EXTENTS DE
87     ON DS.SEGMENT_NAME = DE.SEGMENT_NAME
88 WHERE DS.SEGMENT_NAME like '%VRA_TBS_MULTIPLE%'
89 GROUP BY DF.FILE_NAME, DF.FILE_ID;
90 --H
91 declare
92     v_count number := 0;
93 begin
94     while v_count < 512*5 loop
95         insert into VRA_TBS_MULTIPLE.VRA_TBS_MULTIPLE (
96             str) values('S');
97         v_count := v_count + 1;
98     end loop;
99     commit;
100 end;
101 /
102 SELECT DF.FILE_NAME,
103     DF.FILE_ID,
104     COUNT(DE.SEGMENT_NAME) NUMERO_EXTENSIONES,
105     SUM(DE.BYTES/(1024*1024)) TOTAL_MB,
106     SUM(DE.BLOCKS) BLOQUES_RESERVADOS
107 FROM DBA_SEGMENTS DS
108 JOIN DBA_DATA_FILES DF
109     ON DS.HEADER_FILE = DF.FILE_ID
110 JOIN DBA_DATA_FILES DF
111     ON DS.HEADER_FILE = DF.FILE_ID
112 JOIN DBA_EXTENTS DE
113     ON DS.SEGMENT_NAME = DE.SEGMENT_NAME
114 WHERE DS.SEGMENT_NAME like '%VRA_TBS_MULTIPLE%'
115 GROUP BY DF.FILE_NAME, DF.FILE_ID;
116 --I
117 SELECT
118     TRUNC((SUM(DF.BYTES)-NVL(SUM(S.BYTES),0))

```

```

111 / (1024*1024), 2) MB_LIBRES,
112 (SUM(DF.BLOCKS) - NVL(SUM(S.BLOCKS), 0))
113 BLOQUES_DISPONIBLES
114 FROM DBA_DATA_FILES DF
115 LEFT JOIN DBA_SEGMENTS S
116 ON S.TABLESPACE_NAME = DF.TABLESPACE_NAME
117 WHERE DF.TABLESPACE_NAME = 'STORE_TBS_MULTIPLE';

```

Código 1. s-00-datafile.sql

FILE_NAME	FILE_ID	SIZE_MB
/u01/app/oracle/oradata/VRABDA2/store_tbs_multiple_01.dbf	6	10
/u02/app/oracle/oradata/VRABDA2/store_tbs_multiple_02.dbf	7	10
/u03/app/oracle/oradata/VRABDA2/store_tbs_multiple_03.dbf	8	10

Figure 1. Salida punto A

MB_LIBRES	BLOQUES_DISPONIBLES
165.93	21240

Figure 2. Salida punto B

FILE_NAME	FILE_ID	NUMERO_EXTENSIONES	TOTAL_MB	BLOQUES_RESERVADOS
/u01/app/oracle/oradata/VRABDA2/store_tbs_multiple_01.dbf	7	1	9.0625	

Figure 3. Salida punto E

1) ¿Qué data files se usaron para inicializar su segmento?: Se uso /u03/app/oracle/oradata/VRABDA2/store\_tbs\_multiple\_02.dbf

FILE_NAME	FILE_ID	NUMERO_EXTENSIONES	TOTAL_MB	BLOQUES_RESERVADOS
/u01/app/oracle/oradata/VRABDA2/store_tbs_multiple_01.dbf	6	11	9.0625	16

Figure 4. Salida punto G

2) ¿Qué diferencias se observan?: El numero de bloques reservados y el total de MB usados

FILE_NAME	FILE_ID	NUMERO_EXTENSIONES	TOTAL_MB	BLOQUES_RESERVADOS
/u01/app/oracle/oradata/VRABDA2/store_tbs_multiple_01.dbf	6	19	9	312

Figure 5. Salida punto H

3) ¿Qué diferencias se observan?: en el numero de extensiones, el total de MB y los bloques

4) ¿Cuántas extensiones hasta el momento se han reservado?: 19

5) ¿Cuál es el espacio total reservado hasta el momento?: 4 MB

6) ¿Qué sucede con los demás data files que integran a este tablespace?: Aun no se usan ya que aun hay espacio en el primer datafile

MB_LIBRES	BLOQUES_DISPONIBLES
135.93	17400

Figure 6. Salida punto I

7) ¿Qué diferencias se observan?: El numero de bloques y MB libres disminuyo

### III. CONCLUSIONES

Fue un ejercicio que demostro la forma de poder monitorear el espacio de almacenamiento que se va ocupando en una base de datos, este es ocupado en los datafiles y se distribuye de una forma secuencial.