

G. Observar los valores de la columna undotsn de la consulta del inciso E. Confirmar que a partir del cambio del tablespace undo, el valor de esta columna cambia. Generar una sentencia SQL que permita mostrar los nombres de los tablespaces asociados a los valores de esta columna. Tip: auxiliarse de v\$tablespace. En la consulta incluir únicamente el periodo de la muestra, id del tablespace y su nombre

```
1 SELECT BEGIN_TIME,  
2         END_TIME,  
3         UNDOTSN,  
4         T.NAME  
5 FROM V$UNDOTSTAT U JOIN V$TABLESPACE T ON U.UNDOTSN=T  
        .TS# WHERE ROWNUM<=20 ORDER BY BEGIN_TIME DESC;
```

Código 5. sentencia consulta tablespace id-nombre

	开始 BEGIN_TIME	结束 END_TIME	窗口 UNDOTSN	窗口 NAME
1	2021-07-21 20:38:13	2021-07-21 20:48:10		8 UNDOTBS2
2	2021-07-21 20:28:13	2021-07-21 20:38:13		8 UNDOTBS2
3	2021-07-21 20:18:13	2021-07-21 20:28:13		8 UNDOTBS2
4	2021-07-21 20:08:13	2021-07-21 20:18:13		8 UNDOTBS2
5	2021-07-21 19:58:13	2021-07-21 20:08:13		2 UNDOTBS1
6	2021-07-21 19:48:13	2021-07-21 19:58:13		2 UNDOTBS1
7	2021-07-21 19:38:13	2021-07-21 19:48:13		2 UNDOTBS1
8	2021-07-21 19:28:13	2021-07-21 19:38:13		2 UNDOTBS1
9	2021-07-21 19:18:13	2021-07-21 19:28:13		2 UNDOTBS1
10	2021-07-21 19:08:13	2021-07-21 19:18:13		2 UNDOTBS1

Figure 4. Salida punto G

H. Generar una consulta que muestre las siguientes columnas: Nombre del tablespace undo que fue creado en pasos anteriores, total de bloques que contiene, total de bloques libres, porcentaje de bloques libres. Confirmar que el porcentaje de bloques libres es bajo, menor al 20%. Esta condición pone en riesgo la operación correcta que requieren hacer uso de los datos Undo

```

1 SELECT T.TABLESPACE_NAME,
2        DF.BLOCKS TOTAL_BLOQUES,
3        (DF.BLOCKS-SUM(E.BLOCKS)) BLOQUES_LIBRES,
4        TRUNC(((DF.BLOCKS-SUM(E.BLOCKS))/DF.BLOCKS *100)
5              ,2) PORCENTAJE_LIBRE FROM DBA_TABLESPACES T
6 JOIN DBA_DATA_FILES DF
7      ON T.TABLESPACE_NAME=DF.TABLESPACE_NAME
8 JOIN DBA_UNDO_EXTENTS E
9      ON T.TABLESPACE_NAME=E.TABLESPACE_NAME
10 WHERE T.TABLESPACE_NAME='UNDOTBS2'
11 GROUP BY T.TABLESPACE_NAME,DF.BYTES,DF.BLOCKS;

```

Código 6. sentencia consulta tablespace bloques libres

	TABLESPACE_NAME	TOTAL_BLOQUES	BLOQUES_LIBRES	PORCENTAJE LIBRE
1	UNDOTBS2	3840	3080	80.2

Figure 5. Salida punto H

I. Empleando el usuario creado en temas anteriores, generar una tabla con la siguiente estructura

```
1 DECLARE
2     v_count number;
3     v_count_s number;
4     v_username varchar2(30) := 'VRA_TBS_MULTIPLE';
5     v_table varchar2(30) := 'VRA_CADENA_2';
6     v_secuencia varchar2(30) := 'SEC_VRA_CADENA_2';
7 BEGIN
8     --Verificar si la table existe
9     select count(*) into v_count
10    from all tables
```

```

where table_name = v_table
and owner = v_username;

--Verificar si la secuencia existe
select count(*) into v_count_s
  from all_sequences
where sequence_name = v_secuencia
  and sequence_owner = v_username;
--Si existe la tabla, entonces se borra
if v_count > 0 then
  execute immediate 'drop table '||v_username||'.'
    ||v_table;
end if;
if v_count_s > 0 then
  execute immediate 'drop sequence '||v_username||'
    '||v_secuencia;
end if;
execute immediate 'create table '||v_username||'.'
  ||v_table||' (
    id number constraint vra_cadena_2_pk primary key
  ,
    cadena varchar2(1024)
  ) nologging';
execute immediate 'create sequence '||v_username||'
  '||v_secuencia;
nd;

- Inserts
eclare
  v_rows number;
  v_stmt varchar2(200);
  v_username varchar2(30) := 'VRA_TBS_MULTIPLE';
  v_table varchar2(30) := 'VRA_CADENA_2';
  v_secuencia varchar2(30) := 'SEC_VRA_CADENA_2';
egin
  _rows := 50000;
  _stmt := 'insert into '||v_username||'.'||v_table||'
    '
    (
      id, cadena
    ) values (:1, :2)';
or v_index in 1 .. v_rows loop
  execute immediate v_stmt using VRA_TBS_MULTIPLE.
    SEC_VRA_CADENA_2.nextval, dbms_random.string('P'
    ,1024);
nd loop;
nd;

ommit;

```

Código 7. sentencia crear tabla

J. Replicación del error al ejecutar sentencias DML

[illegible]

Figure 6. Salida consulta punto E

Registros eliminados	# bloques utilizados	# transacciones ejecutadas	#bloques activos	Retención en segundos calculado
1 - 5000	835	138	160	9835
5001 - 10000	1669	144	160	1896
10001 - 15000	835	12	160	1741

Table I
TABLA REGISTROS ELIMINADOS

1) Revisar los datos de la tabla y generar una conclusión al respecto: En este caso el unico cambio significativo es el tiempo de retención el cual fue bajando

K. ¿Qué acciones se pueden realizar para corregir el error adicional a aumentar el tamaño del tablespace undo?

A parte de aumentar el tamaño de datafile que contiene el tablespace undo otra acción sería configurar la cláusula *retention noguarantee* sacrificando datos los datos unexpired.

L. Replicación del error snapshot too old

1) ¿Cuántas instrucciones delete se tuvieron que ejecutar para provocar el error snapshot too old?: 4 instrucciones

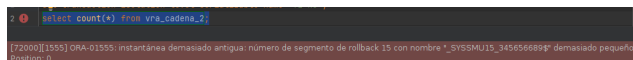


Figure 7. Salida error snapshot too old

M. Provocar ahora el comportamiento inverso: Hacer que la base de datos le de preferencia a consultas que requieran hacer uso de datos undo en lugar de dar preferencia a las sentencias SQL

```
1 -- Se configura el retention guarantee
2 alter tablespace undotbs2 retention guarantee;
```

Código 8. Configuración priorizar datos undo

```
1 declare
2   v_rows number;
3   v_stmt varchar2(200);
4   v_username varchar2(30) := 'VRA_TBS_MULTIPLE';
5   v_table varchar2(30) := 'VRA_CADENA_2';
6 begin
7   v_rows := 10;
8   v_stmt := 'delete from ' || v_username || '.' || v_table
9             || '
10              where id in (SELECT id from ' || v_username || '.' ||
11                          v_table || ' WHERE ROWNUM<=5000)';
12   execute immediate v_stmt;
13 end;
14 /
15 commit;
```

Código 9. script que se ejecuta constantemente para eliminar los datos y llenar undo

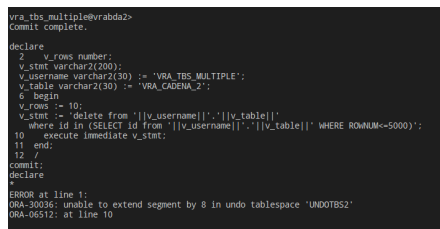


Figure 8. Salida error comportamiento inverso

III. CONCLUSIONES

En este ejercicio se logro comprender mejor la administración de datos undo así como lo que implican, se puede elegir entre configurar para que estos tengan prioridad o para que las sentencias DML siempre puedan ser ejecutadas sin problema.