

# Ejercicio 2 - tema 6

## Estructuras lógicas de almacenamiento - Segmentos

Vicente Romero Andrade

### I. OBJETIVO

EL objetivo es comprender el mecanismo de creación de los distintos tipos de segmentos a partir de la creación de una tabla. Explorar las vistas del diccionario de datos para verificar la creación de segmentos asociados a los objetos de un usuario.

### II. DESARROLLO

#### A. C1. s-01-config-compartido.sql y respuesta del inciso C

```
1  whenever sqlerror exit rollback
2  set serveroutput on
3  connect sys/system2 as sysdba
4  -- A
5  alter system set dispatchers='(dispatchers=2) (
6    PROTOCOL=tcp)' scope=memory;
7  alter system set shared_servers=4 scope=memory;
8  show parameter;
9  --B
10 alter system register;
11 --C
12 select program,pid,pname
13    from v$process
14    where pname like 'S0%' or pname like 'D0%'
15    order by program;
16 whenever sqlerror continue
```

Código 1. s-01-config-compartido.sql

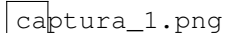


Figure 1. respuesta inciso C

#### B. C2. s-02-conexiones.sql y tnsnames.ora

```
1  whenever sqlerror exit rollback
2  set serveroutput on
3  connect sys@vrabda2_dedicated/system2 as sysdba
4
5  declare
6    v_count number;
7    v_username varchar2(30) := 'VRA0501';
8    v_table varchar2(30) := 'T01_SESSION_DATA';
9  begin
10     --Verificar si la table existe
11     select count(*) into v_count
12       from all_tables
13      where table_name = v_table
14      and owner = v_username;
15     --Si existe la tabla, entonces se borra
16     if v_count > 0 then
17       execute immediate 'drop table ' || v_username
18         || '.' || v_table;
19     end if;
20     execute immediate 'create table ' || v_username
21       || '.' || v_table || ' (
22         id number,
```

```
21     sid number,
22     logon_time date,
23     username varchar2(20),
24     status varchar2(8),
25     server varchar2(20),
26     osuser varchar2(30),
27     process varchar2(12),
28     port number
29   );
30 end;
31 /
32 -- A
33 insert into vra0501.t01_session_data (
34   id,sid,logon_time,username,status,server,
35   osuser,process,port)
36 select 1,sid,logon_time,username,status,server,
37        osuser,process,port from v$session where
38        username = 'SYS';
39 commit;
40 --B
41 connect sys@vrabda2_shared/system2 as sysdba
42
43 insert into vra0501.t01_session_data (
44   id,sid,logon_time,username,status,server,
45   osuser, process,port)
46 select 2,sid,logon_time,username,status,server,
47        osuser, process,port from v$session where
48        username = 'SYS';
49 commit;
50 --C
51 connect VRA0501@vrabda2_dedicated/VRA0501
52
53 insert into vra0501.t01_session_data (
54   id,sid,logon_time,username,status,server,
55   osuser, process,port)
56 select 3,sid,logon_time,username,status,server,
57        osuser, process,port from v$session where
58        username = 'VRA0501';
59 commit;
60 --D
61 connect VRA0501@vrabda2_shared/VRA0501
62
63 insert into vra0501.t01_session_data (
64   id,sid,logon_time,username,status,server,
65   osuser, process,port)
66 select 4,sid,logon_time,username,status,server,
67        osuser, process,port from v$session where
68        username = 'VRA0501';
69 commit;
70 whenever sqlerror continue
```

Código 2. s-02-conexiones.sql

```
1  # tnsnames.ora Network Configuration File: /u01/
2  app/oracle/product/19.0.0/dbhome_1/network/admin
3  /tnsnames.ora
4  # Generated by Oracle configuration tools.
5
6  VRABDA1 =
7    (DESCRIPTION =
8      (ADDRESS = (PROTOCOL = TCP) (HOST = pc-vra.fi.
9        unam) (PORT = 1521))
```

```

7      (CONNECT_DATA =
8        (SERVER = DEDICATED)
9        (SERVICE_NAME = vrabda1.fi.unam)
10     )
11  )
12
13  LISTENER_VRABDA1 =
14    (ADDRESS = (PROTOCOL = TCP) (HOST = pc-vra.fi.
15      unam) (PORT = 1521))
16
17  VRABDA2 =
18    (DESCRIPTION =
19      (ADDRESS = (PROTOCOL = TCP) (HOST = pc-vra.fi.
20        unam) (PORT = 1521))
21      (CONNECT_DATA =
22        (SERVER = DEDICATED)
23        (SERVICE_NAME = vrabda2)
24      )
25    )
26
27  VRABDA2_DEDICATED =
28    (DESCRIPTION =
29      (ADDRESS_LIST =
30        (ADDRESS = (PROTOCOL = TCP
31          ) (HOST = pc-vra.fi.unam) (PORT = 1521))
32      )
33      (CONNECT_DATA =
34        (SERVICE_NAME = vrabda2)
35        (SERVER=DEDICATED)
36      )
37    )
38
39  VRABDA2_SHARED =
40    (DESCRIPTION =
41      (ADDRESS_LIST =
42        (ADDRESS = (PROTOCOL = TCP
43          ) (HOST = pc-vra.fi.unam) (PORT = 1521))
44      )
45      (CONNECT_DATA =
46        (SERVICE_NAME = vrabda2)
47        (SERVER=SHARED)
48      )
49    )

```

Código 3. tnsname.ora

### C. C3. s-03-consultas.sql

```

1 whenever sqlerror exit rollback
2 set serveroutput on
3 connect sys/system2 as sysdba
4
5 declare
6   v_count number;
7   v_username varchar2(30) := 'VRA0501';
8   v_table1 varchar2(30) := 'T02_DISPATCHER_CONFIG';
9   v_table2 varchar2(30) := 'T03_DISPATCHER';
10  v_table3 varchar2(30) := 'T04_SHARED_SERVER';
11  v_table4 varchar2(30) := 'T05_QUEUE';
12  v_table5 varchar2(30) := 'T06_VIRTUAL_CIRCUIT';
13 begin
14   --Verificar si la table existe
15   select count(*) into v_count
16   from all_tables
17   where table_name = v_table1
18   and owner = v_username;
19   --Si existe la tabla, entonces se borra
20   if v_count > 0 then
21     execute immediate 'drop table ' || v_username || '
22     .' || v_table1;
23   end if;
24
25   --Verificar si la table existe
26   select count(*) into v_count
27   from all_tables
28   where table_name = v_table2

```

```

28   and owner = v_username;
29   --Si existe la tabla, entonces se borra
30   if v_count > 0 then
31     execute immediate 'drop table ' || v_username || '
32     .' || v_table2;
33   end if;
34
35   --Verificar si la table existe
36   select count(*) into v_count
37   from all_tables
38   where table_name = v_table3
39   and owner = v_username;
40   --Si existe la tabla, entonces se borra
41   if v_count > 0 then
42     execute immediate 'drop table ' || v_username || '
43     .' || v_table3;
44   end if;
45
46   --Verificar si la table existe
47   select count(*) into v_count
48   from all_tables
49   where table_name = v_table4
50   and owner = v_username;
51   --Si existe la tabla, entonces se borra
52   if v_count > 0 then
53     execute immediate 'drop table ' || v_username || '
54     .' || v_table4;
55   end if;
56
57   --Verificar si la table existe
58   select count(*) into v_count
59   from all_tables
60   where table_name = v_table5
61   and owner = v_username;
62   --Si existe la tabla, entonces se borra
63   if v_count > 0 then
64     execute immediate 'drop table ' || v_username || '
65     .' || v_table5;
66   end if;
67
68 /
69
70 create table vra0501.t02_dispatcher_config as(
71   select 1 as id,dispatchers,connections,sessions,
72   service from v$dispatcher_config
73 );
74
75
76
77
78
79 create table vra0501.t03_dispatcher as(
80   select 1 as id,name,network,status,messages,
81   trunc(bytes/(1024*1024),2) messages_mb,
82   (select count(*) from v$circuit) circuits_created,
83   trunc(idle/(60*60),2) idle_min
84   from v$dispatcher
85 );
86
87
88 create table vra0501.t04_shared_server as(
89   select 1 as id,name,status,messages,
90   trunc(bytes/(1024*1024),2) messages_mb,
91   requests,trunc(idle/(60*60),2) idle_min,
92   trunc(busy/(60*60),2) busy_min
93   from v$shared_server
94 );
95
96
97 create table vra0501.t05_queue as(
98   select 1 as id,queued,wait,totalq from v$queue
99 );
100
101 create table vra0501.t06_virtual_circuit as(
102   select 1 as id,c.circuit,dp.name,c.server,c.status
103   ,c.queue
104   from v$dispatcher dp join v$circuit c on(
105     dp.paddr=c.dispatcher)
106 );
107
108 whenever sqlerror continue

```

Código 4. s-03-consultas.sql

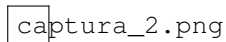


Figure 2. T02\_DISPATCHER\_CONFIG

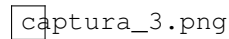


Figure 3. T03\_DISPATCHER

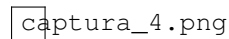


Figure 4. T04\_SHARED\_SERVER

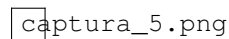


Figure 5. T05\_QUEUE

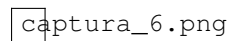


Figure 6. T06\_VIRTUAL\_CIRCUIT

#### D. C4. Respuesta inciso A y s-04-procesos.sql

```

1 whenever sqlerror exit rollback
2 set serveroutput on
3 connect sys/system2 as sysdba
4
5 declare
6   v_count number;
7   v_username varchar2(30) := 'VRA0501';
8   v_table1 varchar2(30) := 'T07_SESSION_INFO_CONTEXT';
9   v_table2 varchar2(30) := 'T08_SESSION_INFO_VIEW';
10  v_table3 varchar2(30) := 'T09_PROCESS_INFO';
11  v_table4 varchar2(30) := 'T10_BACKGROUND_PROCESS';
12  v_table5 varchar2(30) := 'T11_FOREGROUND_PROCESS';
13 begin
14   --Verificar si la table existe
15   select count(*) into v_count
16   from all_tables
17   where table_name = v_table1
18   and owner = v_username;
19   --Si existe la tabla, entonces se borra
20   if v_count > 0 then
21     execute immediate 'drop table '|| v_username ||'
22     .'||v_table1;
23   end if;
24
25   --Verificar si la table existe
26   select count(*) into v_count
27   from all_tables
28   where table_name = v_table2
29   and owner = v_username;
30   --Si existe la tabla, entonces se borra
31   if v_count > 0 then
32     execute immediate 'drop table '|| v_username ||'
33     .'||v_table2;
34   end if;
35
36   --Verificar si la table existe
37   select count(*) into v_count
38   from all_tables
39   where table_name = v_table3
40   and owner = v_username;
41   --Si existe la tabla, entonces se borra
42   if v_count > 0 then
43     execute immediate 'drop table '|| v_username ||'
44     .'||v_table3;
45   end if;
46
47   --Verificar si la table existe
48   select count(*) into v_count
49   from all_tables
50   where table_name = v_table4
51   and owner = v_username;
52   --Si existe la tabla, entonces se borra
53   if v_count > 0 then
54     execute immediate 'drop table '|| v_username ||'
55     .'||v_table4;
56   end if;
57
58   --Verificar si la table existe
59   select count(*) into v_count
60   from all_tables
61   where table_name = v_table5
62   and owner = v_username;
63   --Si existe la tabla, entonces se borra
64   if v_count > 0 then
65     execute immediate 'drop table '|| v_username ||'
66     .'||v_table5;
67   end if;
68 end;
69
70 create table vra0501.t07_session_info_context (
71   host varchar2(20),
72   os_user varchar2(30),
73   user_id number,
74   session_id number
75 );
76
77 insert into vra0501.t07_session_info_context (host,
78   os_user, user_id, session_id)
79 select sys_context('USERENV','HOST') as host,
80   sys_context('USERENV','OS_USER') as os_user,
81   sys_context('USERENV','SESSION_USERID') as
82   user_id,
83   sys_context('USERENV','SID') as session_id
84 from dual;
85
86 create table vra0501.t08_session_info_view as (
87   select s.sid as session_id, s.paddr as
88   process_address,
89   s.username bd_username, s.status as session_status,
90   s.port as client_port, s.process as
91   os_client_process_id,
92   s.program as client_program from v$sqlprocess p
93   join gv$sqlsession s on p.addr=s.paddr
94   join (select sys_context('USERENV','SID') as
95   session_id
96   from dual) c on c.session_id=s.sid where s.
97   username = 'SYS'
98 );
99
100 create table vra0501.t09_process_info as (
101   select sosid, pname, background, tracefile from
102   v$sqlprocess
103   join gv$sqlsession s on addr=s.paddr join (
104   select sys_context('USERENV','SID') as
105   session_id
106   from dual) c on s.sid = c.session_id
107 );
108
109 create table vra0501.t10_background_process as (
110   select addr, sosid, pname, username as
111   os_username, background
112   from v$sqlprocess where background='1'
113 );
114
115 create table vra0501.t11_foreground_process as (
116   select p.addr, p.sosid, p.pname, p.username as
117   bd_username, s.osuser as os_username, p.
118   background
119   from v$sqlprocess p left outer join v$sqlsession s

```

```
104      on (p.addr=s.paddr) where p.background is null
105    );
106
107 whenever sqlerror continue
```

Código 5. s-04-procesos.sql

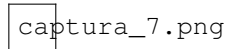
captura\_7.png

Figure 7. T07\_SESSION\_INFO\_CONTEXT

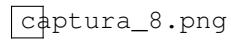
captura\_8.png

Figure 8. T08\_SESSION\_INFO\_VIEW

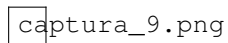
captura\_9.png

Figure 9. T09\_PROCESS\_INFO

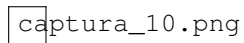
captura\_10.png

Figure 10. T10\_BACKGROUND\_PROCESS

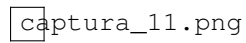
captura\_11.png

Figure 11. T11\_FOREGROUND\_PROCESS

```
1 ps -ef | grep oracle
```

Código 6. instrucciones SO

### III. CONCLUSIONES