

## Lab 10.1: Configuring Shared Server Mode

### Objective:

To configure shared server mode in your Oracle database.

### Steps:

#### 1. Open a Terminal Window and Set the Environment Variables for CDBLAB

Open a terminal window and set your environment variables for the CDBLAB database:

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBLAB
The Oracle base remains unchanged with value /u01/app/oracle
```

#### 2. Log in to SQL\*Plus as a User with SYSDBA Privileges

Start SQL\*Plus and log in as a user with SYSDBA privileges:

```
sqlplus / as sysdba
```

#### 3. Determine Whether the Shared Server Architecture is Implemented in Your Database

a. Check the value of the SHARED\_SERVER initialization parameter:

```
SHOW PARAMETER shared_server;
```

##### Expected Output:

NAME	TYPE	VALUE
max_shared_servers	integer	
shared_server_sessions	integer	
shared_servers	integer	1

b. Check the value of the DISPATCHERS initialization parameter:

```
SHOW PARAMETER dispatchers;
```

##### Expected Output:

NAME	TYPE	VALUE
dispatchers	string	(PROTOCOL=TCP) (SERVICE=CDBLABXDB)
max_dispatchers	integer	

### Explanation:

- A shared server and dispatcher are configured for Oracle XML DB (XDB) by default if you created your Oracle database using DBCA. This configuration allows sessions that connect to the XDB service to use shared server.

#### 4. Enable Three Shared Servers in Your Database

```
ALTER SYSTEM SET shared_servers = 3;
```

**Expected Output:**

```
System altered.
```

**5. Determine Whether You Need to Configure Any Additional Dispatchers**

- a. Check the value of the `DISPATCHERS` initialization parameter:

```
SHOW PARAMETER dispatchers;
```

**Expected Output:**

NAME	TYPE	VALUE
dispatchers	string	(PROTOCOL=TCP) (SERVICE=CDBLABXDB)
max_dispatchers	integer	

**Explanation:**

- You need to configure additional dispatchers because a dispatcher with a specified service will only connect to that service.

**6. Change the Dispatcher Service to Connect to Any Service Using TCP/IP**

```
ALTER SYSTEM SET dispatchers = "(PROTOCOL=TCP)";
```

**Expected Output:**

```
System altered.
```

**7. Confirm the Change**

```
SHOW PARAMETER dispatchers;
```

**Expected Output:**

NAME	TYPE	VALUE
dispatchers	string	(PROTOCOL=TCP)
max_dispatchers	integer	

**8. Exit SQL\*Plus**

```
exit
```

**9. Close All Terminals**

```
exit
```

**Part B: Configuring Clients to Use a Shared Server****Overview:**

In this practice, you configure a network service that uses shared server mode.

## Steps:

### 1. Open a Terminal Window and Set the Environment Variables for CDBLAB

Open a terminal window and set your environment variables for the CDBLAB database:

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBLAB
The Oracle base remains unchanged with value /u01/app/oracle
```

### 2. Configure a Network Service That Uses a Shared Server Connection

a. Change the directory to \$ORACLE\_HOME/network/admin and list your current working directory:

```
cd $ORACLE_HOME/network/admin
pwd
```

#### Expected Output:

```
/u01/app/oracle/product/19.3.0/dbhome_1/network/admin
```

b. Make a copy of tnsnames.ora :

```
cp tnsnames.ora tnsnames.ora.4-2
```

c. Edit the tnsnames.ora file to add a new service named test\_ss that uses a dispatcher:

```
vi tnsnames.ora
```

#### Add the following entry:

```
TEST_SS =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL=TCP) (HOST=your_fully_qualified_hostname) (PORT=1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME=CDBLAB)
      (SERVER=shared)
    )
  )
```

### 3. Invoke SQL\*Plus and Connect to the Database Using a Dispatcher

Open a new terminal window and connect to the test\_ss service:

```
sqlplus system@test_ss
```

Enter the password when prompted.

### 4. Open Another Terminal Window and Set the Environment Variables for CDBLAB

Open another terminal window and set your environment variables for the CDBLAB database:

```
. oraenv
ORACLE_SID = [orclcdb] ? CDBLAB
The Oracle base remains unchanged with value /u01/app/oracle
```

#### 5. Log in to SQL\*Plus as a User with SYSDBA Privileges

Start SQL\*Plus and log in as a user with SYSDBA privileges:

```
sqlplus / as sysdba
```

#### 6. View Information About All Shared Server Connections by Querying `V$CIRCUIT`

```
SELECT dispatcher, server, saddr, queue FROM v$circuit;
```

**Expected Output:**

DISPATCHER	SERVER	SADDR	QUEUE
-----	-----	-----	-----
000000009F54D420	00	000000009F967A18	NONE

#### 7. Log out of the SQL\*Plus Session Using the `test_ss` Service

In the first terminal window, log out of the SQL\*Plus session:

```
exit
```

#### 8. View Information About the Shared Server Connection by Querying `V$CIRCUIT` Again

In the second terminal window, query `V$CIRCUIT` again:

```
SELECT dispatcher, server, saddr, queue FROM v$circuit;
```

**Expected Output:**

```
no rows selected
```

#### 9. Log out of SQL\*Plus and Close Both Terminal Windows

In the second terminal window, log out of SQL\*Plus:

```
exit
```

Close both terminal windows:

```
exit
```

## Summary

In this lab, you configured shared server mode in your Oracle database, enabled multiple shared servers, and set up a network service to use a shared server connection. This practice ensures that you understand how to configure and manage shared server mode to optimize database connections and resources.