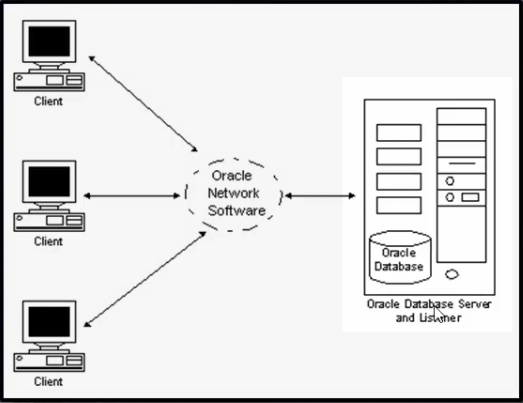
# **Oracle Networks – Concepts**

These networks play a crucial role in establishing connections, transmitting data, and managing database access.  
Here are some key concepts related to Oracle Networks:  
  
**Listener:**  
The Oracle Listener is a process that runs on the database server and listens for incoming connection requests. It acts as a mediator between clients and the database instance. When a client wants to connect to a database, it contacts the Listener, which redirects the connection to the appropriate database service.  
  
**TNS (Transparent Network Substrate) Names:**  
TNS names are a naming method used by Oracle to identify the network location and service names for database connections. These names are typically stored in a tnsnames.ora file, which acts as a local repository for resolving service names to network addresses.  
  
**Connect Descriptor:**  
A connect descriptor is a string containing network information that specifies the location of a database service. It includes details such as the database's hostname or IP address, port number, and service name. Connect descriptors are used by clients to connect to the database.  
  
**Oracle Net Configuration Tools:**  
Oracle provides various tools and utilities for configuring Oracle Net, including the Oracle Net Configuration Assistant (NetCA) and the Oracle Net Manager. These tools simplify the setup and management of network configurations.



* **DATABASE 🡺 LISTENER 🡺 ORACLE CLIENT 🡺 APPLICATION**

**# LISTENER : in database**

**# TNS:application team**

* we can create listener & tns manually.
* Using netca utility also we can configure or delete listener or tns.

**LISTENER:**

* listener should be up & run 24/7. Then application team can connect to the database in any point of time.
* listener is mainly responsible for taking new connections.
* When the user is connected to database and listener is stopped there will be no impact for user connection.
* But if new user try to connect database. that cannot connect.

**LISTENER / TNS default:(location)**

* cd $ORACLE\_HOME/network/admin

**To create listener other than default location:**

* add this in ENV file
* export TNS\_ADMIN=/location /where /we/want
* Execute file = . Hyd.env

**To start listener:mouli**

* lsnrctl start mouli

**To stop listener : mouli**

* lsnrctl stop mouli

**To check listener status: mouli**

* lsnrctl status

**To check tnsping : pytest**

* tnsping PYTEST

**To connect remote database with sys user we need password file.**

* sqlplus sys/sys123@abc as sysdba

**Listener’s are two types:**

1. **static listener**

* irrespective of this database status it up and run 24/7.
* In static listener it contain database name.
* We can connect to remote database with sys user when database is in down state.

1. **Dynamic listener**

* it work only when database is in open state.
* This listener is having default configuration(Like default name(LISTENER)&port(1521).
* It will not contain database name.
* Database should register with listener.

**STATIC LISTENER FOR PYTEST**

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(GLOBAL\_DBNAME = PYTEST)

(ORACLE\_HOME = D:\APP\product\11.2.0\dbhome\_1)

(SID\_NAME = PYTEST)

)

)

LISTENER =

(DESCRIPTION\_LIST =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.200.2.14)(PORT = 1521))

)

(DESCRIPTION =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))

)

)

ADR\_BASE\_LISTENER = D:\APP

**STATIC TNSNAMES FOR PYTEST**

* Ip address & port in tns should be same as listener.

PYTEST =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.200.2.14)(PORT = 1521))

)

(CONNECT\_DATA =

(SERVICE\_NAME = PYTEST)

)

)

**DYNAMIC LISTENER:**

Listener = (DESCRIPTION = ( ADDRESS = (HOST = 192.168.1.56) (PORT = 1522 )(PROTOCOL = TCP)))

• register local\_parameter

• show parameter local

• alter system set local\_listener=‘192.168.1.10:1524/Pune’;

• Alter system register;

* In 11g pmon is responsible for regestring database with listener.
* From 12c new background process LREG is responsible for regestring database with listener.

**DYNAMIC TNSNAMES:**

tns\_hyd= (DESCRIPTION = ( ADDRESS = (HOST = 192.168.1.56) (PORT = 1521 )(PROTOCOL = TCP))(connect\_data=(service\_name=hyd)))