

NEXTSTEP

Title: Using EOF with Oracle SQLNet*V1and SQLNet*V2

Entry Number: 1897

Last Updated: 29 August 1996

Keywords: EOModeler, Oracle, SQLNet

Questions

Q: I want to connect to my Oracle7 server using SQLNet*V2. How can I make it work with EOModeler?

Answers

A: EOModeler should work when the Oracle server and the NEXTSTEP client(s) are properly configured. Here are some suggestions to troubleshoot possible configuration problems:

On the Oracle server side (i.e., HP-UX, Solaris, etc.) :

-When installing Oracle, make sure to install the version of SQLNet (V1 or V2) that you intend to use. You may install both.

Please note that EOF Release 1.0 supports only SQLNet*V1. Release 1.1 support both SQLNet*V1 and V2.

- Check for the following entries in `/etc/services`

```
orasrv 1525/tcp # Oracle SQL+Net V1 port
tnslsnr 1521/tcp # Oracle SQL+Net V2 port
```

Note: If you put these two service entries in the root level domain, it will enable all the machines in the network.

- `/etc/hosts` should contain entries for the NEXTSTEP client machines (i.e., 129.11.33.22 cubs)

On the NEXSTEP client side:

- If you are using SQL*Net V1, you'll need to add the orasrv service entry

(In NetInfoMgr: Edit-> Duplicate services/ftp, and set name=orasrv, port=1525, protocol=tcp)

- If you are using SQL*Net V 2, you'll need to add a file `tnsnames.ora` to the `/etc` directory. Note that both server and client must have the same entry in the `tnsnames.ora` file. You also need to have an entry for `tnslsnr 1521/tcp` in the `/etc/services` on your NEXTSTEP client (similar to your Oracle server) for SQL*Net V2.

- Enter the following information into the Oracle Login panel of EOModeler

For SQL*Net V1 style connection:

Server ID: `your ORACLE_SID`

Host Machine: `your oracle server host name`

User Name: `some_username`

Password: `some_password`

For SQL*Net V2 style connection:

Server ID: `< < This is the aliasname in /etc/tnsnames.ora for SQL*Net V2.1; it could be the ORACLE_SID for SQL*Net V2.0> >`

Host Machine: `< < MUST BE BLANK> >`

User Name: `some_username`

Password: `some_password`

You can also enter the full SQL*Net V2 connection alias into the Server ID field and leave all the other fields blank.

Here is an example of using tnsnames.ora for a SQL*Net V2.1 style connection:

```
Server ID: anv1.world
Host Machine:
User Name: system
Password: ???
```

and the corresponding /etc/tnsnames.ora file:

```
#
# File: tnsnames.ora
# Created ad hoc by trieger 3/21/95 based
# on technical support fax from Oracle
#
# Was just after address, tech support had me remove it.
# (COMMUNITY = tcp.world)
# 3/23/95 - after reading techsupport fax #2, appended
# .world to alias name below, network connections
# locally seem to work now.
ANVL.world =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS =
(PROTOCOL = TCP)
(Host = eddy)
(Port = 1521)
)
)
```

```
(CONNECT_DATA =  
(SID = ANVL)  
)  
)
```

BYPASSING THE TNSNAMES.ORA FILE

If you are unsure of the proper location of your tnsnames.ora file, you can actually enter the full expansion of the oracle TNS connection in the serverId field of the connection dictionary. This approach also allows you to have EOF clients operate without a tnsnames.ora anywhere on their machine.

For example, to generate a new EOModel with connection information, you could create a text file with the following information (and the .eomodel extension in the name):

```
{  
    EOModelVersion = 1;  
    adaptorName = Oracle7;  
    connectionDictionary = {  
        hostMachine = "";  
        password = the_password;  
        serverId = "(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=eddy)  
(PORT=1521))(CONNECT_DATA=(SID=ANVL)))";  
        userName = the_username;  
    };  
}
```

Opening this file in EOModeler will show an empty model, but you'll be able to use the Table Panel to load the required tables. The login panel should not come up.

OTHER POSSIBILITIES WHEN TROUBLE SHOOTING...

In some cases when EOF is unable to open a connection to an Oracle 7.3 (or later) database server an additional key may be necessary in the Connection Dictionary. The NLS_LANG key may be added directly to the .eomodel file, or programmatically:

```
connectionDictionary = {
    NLS_LANG = "";
    hostMachine = "";
    password = the_password;
    serverId = "(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=eddy)
(PORT=1521))(CONNECT_DATA=(SID=ANVL)))";
    userName = the_username;
};
```

Valid for: EOF 1.1, NEXTSTEP 3.2 Developer, NEXTSTEP 3.3 Developer

See Also: