

EUS Migration from OID to OUD

This page illustrates how to migrate from Oracle Internet Directory (OID) to Oracle Unified Directory (OUD) as an LDAP server for Database Authentication and Enterprise User Security (EUS).

We had an OID 11.1.1.5.0 [installation](#) which provided the basis for Oracle database authentication. This installation served us well, as long as the underlying database was available. But now, the Linux installation where OID ran on was quite out of date, so we tried to install a fresh Linux VM and install another OID instance there. This gave us quite a bit of trouble, since the old OID version did not fit on any modern linux any more, and more current OID 11.1.1.9.0 required a database repository upgrade which would affect the productive OID also. Even when we succeeded with that upgrade, we had a number of problems with the installation and we gained the overall impression that OID was not currently developed and supported in a very eager fashion any more. After we had been sitting on an open SR for half a year, we started looking in alternatives, the most promising was to replace the whole OID installation with the more modern OUD.

Unlike OpenLDAP, OUD is a supported Oracle product tested für database authentication and EUS, and we already had quite good experiences with new OUD installations. The EUS setup in OUD is a [well documented process](#). And unlike OID, OUD is lightweight and doesn't depend on a running database schema, while providing full replication possibilities.

- [OUD and Weblogic installation](#)
 - [Linux system installations](#)
 - [Java installations](#)
 - [Oracle MiddleWare installation](#)
 - [Oracle Unified Directory \(OUD\)](#)
 - [WebLogic Server 11g \(WLS\)](#)
 - [Application Developerment Framework\(ADF\)](#)
- [Configuration](#)
 - [Deploy ODSM](#)
 - [OUD Setup](#)
 - [Schema preparation in OUD](#)
- [Export of OID data](#)
- [Adjustments in LDIF data](#)
- [Import to OUD](#)
 - [Import Groups](#)
 - [Import Users](#)
- [Configuration of a database client](#)
- [Final migration](#)
 - [Re-Export OID data](#)
 - [Re-Transform data](#)
 - [Check numbers](#)
 - [Delete OUD contents](#)
 - [Re-Import](#)
 - [De- and Re-Register productive database](#)
 - [Re-create EUS mappings](#)
 - [Re-test](#)
- [Caveats](#)

OUD and Weblogic installation

The installation of OUD is quite straightforward as described in [Configure EUS with OUD, AD and DB12c](#).

We used Red Hat Enterprise Linux Server release 7.4 (Maipo) for this installation.

Linux system installations

```
yum install compat-libcap1
yum install compat-libstdc++-33.x86_64 compat-libstdc++-33.i686
yum install gcc gcc-c++
yum install libaio-devel
yum install libstdc++.i686
yum install ksh
yum install sysstat
yum install numactl numactl-devel
yum install motif motif-devel
yum install libX11-common
yum install libX11-devel
yum install libxtst6
```

```
yum install libxtst
yum install glibc-devel.i686
```

Java installations

We need a real JDK, not just an JRE.

```
rpm -i jdk-7u131-linux-x64.rpm
```

There is a criticality between the Oracle Database and OUD: [the database requires SSLv3 authentication, which OUD cannot provide if it is disabled in Java security settings](#). You have to manually edit the file `jre/lib/security/java.security` to re-enable SSLv3.

In case a Linux JDK installation is used, JDK may be updated bei the Linux system update mechanism, which results in the database not being able to authenticate any more with error "ORA-28030". So, the recommendation is to install a separate JDK just for the Oracle MiddleWare.

Oracle MiddleWare installation

OUD is installed using these Middleware components:

- wls1036_generic.jar
- ofm_appdev_generic_11.1.1.79.0_disk1_1of1.zip
- ofm_oud_generic_11.1.2.3.0_disk1_1of1.zip

We also installed ApacheDirectoryStudio-2.0.0.v20161101-M12-linux.gtk.x86_64.tar.gz to have a nice LDAP editor on-site, which comes handy during migration.

Oracle Unified Directory (OUD)

- Unpack software archive
- Run installer:
 - `./runInstaller -jreLoc /usr/java/jdk1.8.0_65/jre`
 - OUD Base Location: `/u01/Middleware`
 - everything else: default

WebLogic Server 11g (WLS)

WLS 10.3.6 (11g) is used for ODSM (Oracle LDAP GUI in Weblogic). You could also use an Open Source product like Apache Directory Studio.

- Check available disk space in `/tmp` before installation
- `/usr/java/jdk1.8.0_65/bin/java -Djava.io.tmpdir=/var/tmp -jar wls1036_generic.jar`
- Create a new Middleware Home in `/u01/Middleware`
- Ignore a warning message regarding the non-empty middleware directory
- Custom Installation, without Coherence
- Do not run QuickStart at the end of installation

Application Developerment Framework(ADF)

- Unpack software archive
- `./runInstaller -jreLoc /usr/java/jdk1.8.0_65/jre`

Caution, use at least ADF version 11.1.1.9 to prevent problems with ODSM throwing "OracleJSP error: [java.io.FileNotFoundException](#)" after login (see Doc ID 2066834.1).

Configuration

Deploy ODSM

ODSM is a WebLogic application that can be used to edit the LDAP schema. Stand-alone java application [Apache Directory Studio](#) is far more capable, tough.

```
[Middleware]$ oracle_common/common/bin/config.sh
```

- Create new WebLogic Domain "odsm"
- Select component "Oracle Directory Services Manager" (Oracle JRF is automatically selected)
- Username (weblogic) / password for WebLogic Domain
- Setup-type is "Production"

Start Weblogic server

```
$ user_projects/domains/odsm/bin/startWebLogic.sh
```

Provide the web logic password for unattended WebLogic server startup:

```
$ cd /u01/Middleware/user_projects/domains/odsm/servers/AdminServer/
[AdminServer]$ mkdir security
[AdminServer]$ cd security
[security]$ vi boot.properties
username=weblogic password=...
```

At weblogic startup, the password is automatically encrypted in the file boot.properties, so you do not have to provide it any more.

OUD Setup

```
/u01/Middleware/Oracle_OUD1/oud-setup
/u01/Middleware/asinst_1/OUDBin/start-ds
/u01/Middleware/asinst_1/OUDBin/status
```

OUD can create an init.d-style start script:

```
./create-rc-script --outputFile /etc/init.d/oud
[root@oraldap3 ~]# cat /etc/init.d/oud
#!/bin/sh
#
# Copyright (c) 2012, Oracle and/or its affiliates. All rights reserved.
#
#
# chkconfig: 345 90 30
# description: Oracle Unified Directory startup script
#
# Set the path to the Oracle Unified Directory instance to manage
INSTALL_ROOT="/u01/Middleware/asinst_1/OUDBin"
export INSTALL_ROOT
# Determine what action should be performed on the server
case "${1}" in
start)
    /bin/su - root -- "${INSTALL_ROOT}/bin/start-ds" --quiet
    exit $?
    ;;
stop)
    /bin/su - root -- "${INSTALL_ROOT}/bin/stop-ds" --quiet
    exit $?
    ;;
restart)
    /bin/su - root -- "${INSTALL_ROOT}/bin/stop-ds" --restart --quiet
    exit $?
    ;;
*)
    echo "Usage: $0 { start | stop | restart }"
    exit 1
    ;;
esac
```

```

[root@oraldap3 ~]# cat /etc/init.d/weblogic
#!/bin/sh
#
#
# chkconfig: 345 90 30
# description: WebLogic 10.3.6 startup script
#
BASE_DOMAIN="/u01/Middleware/user_projects/domains/odsm/"
export BASE_DOMAIN
# Determine what action should be performed on the server
case "${1}" in
start)
    /bin/su - oracle -c "nohup ${BASE_DOMAIN}/bin/startWebLogic.sh &"
    exit ${?}
    ;;
stop)
    /bin/su - oracle -c "${BASE_DOMAIN}/bin/stopWebLogic.sh"
    exit ${?}
    ;;
restart)
    /bin/su - oracle -c "${BASE_DOMAIN}/bin/stopWebLogic.sh"
    /bin/su - oracle -c "nohup ${BASE_DOMAIN}/bin/startWebLogic.sh &"
    exit ${?}
    ;;
*)
    echo "Usage: $0 { start | stop | restart }"
    exit 1
    ;;
esac

```

```

chkconfig --add oud
chkconfig --add weblogic
chkconfig --list
oud            0:off 1:off 2:off 3:on 4:on 5:on 6:off
weblogic       0:off 1:off 2:off 3:on 4:on 5:on 6:off

```

```

[root@oraldap3 oracle]# /u01/Middleware/asinst_1/OUDBin/status
>>>> Specify Oracle Unified Directory LDAP connection parameters
Administrator user bind DN[cn=Directory Manager]:
Password for user 'cn=Directory Manager':

    - Server Status -

Server Run Status:      Started
Open Connections:      1

```

- Server Details -

Host Name: oraldap3.cust.de
Administrative Users: cn=Directory Manager
Installation Path: /u01/Middleware/Oracle_OUD1
Instance Path: /u01/Middleware/asinst_1/OU
Version: Oracle Unified Directory 11.1.2.3.0
Java Version: 1.8.0_141
Administration Connector: Port 4444 (LDAPS)

Schema preparation in OUD

An LDIF like this has to be added to create the OracleContext in the LDAP schema:

```
[oracle@oraldap3 ~]$ cat modifyRealm.ldif
# Please uncomment the following one line if you are importing this
# LDIF file via OVD Manager or OVD Server's ldapmodify tool.
#version: 1
dn: cn=Common,cn=Products,cn=OracleContext
changetype: modify
replace: orclSubscriberSearchBase
orclSubscriberSearchBase: dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext
changetype: modify
replace: orclSubscriberNickNameAttribute
orclSubscriberNickNameAttribute: dc
dn: cn=Common,cn=Products,cn=OracleContext
changetype: modify
replace: orclDefaultSubscriber
orclDefaultSubscriber: dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext,dc=cust,dc=de
changetype: modify
replace: orclCommonUserSearchBase
orclCommonUserSearchBase: ou=people,dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext,dc=cust,dc=de
changetype: modify
replace: orclCommonUserCreateBase
orclCommonUserCreateBase: ou=people,dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext,dc=cust,dc=de
changetype: modify
replace: orclCommonDefaultUserCreateBase
orclCommonDefaultUserCreateBase: ou=people,dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext,dc=cust,dc=de
changetype: modify
replace: orclCommonGroupCreateBase
orclCommonGroupCreateBase: ou=groups,dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext,dc=cust,dc=de
```

```
changetype: modify
replace: orclCommonDefaultGroupCreateBase
orclCommonDefaultGroupCreateBase: ou=groups,dc=cust,dc=de
dn: cn=Common,cn=Products,cn=OracleContext,dc=cust,dc=de
changetype: modify
replace: orclCommonGroupSearchBase
orclCommonGroupSearchBase: ou=groups,dc=cust,dc=de
```

Next, we need a basic structure for users and groups in the LDAP subtree:

```
[oracle@oraldap3 ~]$ /u01/Middleware/Oracle_OUD1/bin/ldapmodify -D "cn=Directory Manager" -j /tmp/p -a -f
add_cust_structure.ldif
```

```
Processing ADD request for cn=Users,dc=cust,dc=de
ADD operation successful for DN cn=Users,dc=cust,dc=de
Processing ADD request for cn=Groups,dc=cust,dc=de
ADD operation successful for DN cn=Groups,dc=cust,dc=de
```

```
[oracle@oraldap3 ~]$ cat add_cust_structure.ldif
```

```
#
# LDAPv3
# base <dc=cust,dc=de> with scope oneLevel
# filter: (objectclass=*)
# requesting: *
#
# Users, cust.de
dn: cn=Users, dc=cust,dc=de
objectclass: top
objectclass: orclContainer
cn: users
# Groups, cust.de
dn: cn=Groups, dc=cust,dc=de
objectclass: top
objectclass: orclContainer
cn: Groups
# search result
# numResponses: 5
# numEntries: 4
```

Finally, we create a test entry in OUD (Directory Manager Password is put into the text file /tmp/p first):

```
[oracle@oraldap3 Loopback]$ cat testuser.ldif
```

```
dn: cn=test_oud,cn=Users, dc=cust,dc=de
cn: test_oud
mail: test.test\_oud@mycompany.com
objectclass: top
objectclass: inetOrgPerson
objectclass: orcluser
```

```

objectclass: organizationalPerson
objectclass: person
orclglobalid: 000
orclpkcs12hint: test_oud
sn: test_oud1
uid: test_oud1
userpassword: XXX

[oracle@oraldap3 ~]$ /u01/Middleware/Oracle_OUD1/bin/ldapmodify -D "cn=Directory Manager" -j /tmp/p -a -f
testuser.ldif

Processing ADD request for cn=test_oud,cn=Users,dc=cust,dc=de
ADD operation successful for DN cn=test_oud,cn=Users,dc=cust,dc=de

```

Export of OID data

The export can be done on the OUD host with the Linux ldapsearch binary against the still-running OID server, if reachable:

```

/usr/bin/ldapsearch/ldapsearch -p 3060 -h oraldap1.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=Groups,
dc=cust,dc=de" "(objectclass=*)" "*" > oid-groups.ldif

```

```

/usr/bin/ldapsearch/ldapsearch -p 3060 -h oraldap1.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=users,
dc=cust,dc=de" "(objectclass=*)" "*" > oid-users.ldif

```

Nested groups have to be fetched separately:

```

/usr/bin/ldapsearch -p 3060 -h oraldap1.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=CERA_GROUPS,cn=Groups,
dc=cust,dc=de" "(objectclass=*)" "*" > oid-groups-NESTED.ldif

```

```

/usr/bin/ldapsearch -p 3060 -h oraldap1.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=ADMIN_GROUPS,
cn=Groups,dc=cust,dc=de" "(objectclass=*)" "*" >> oid-groups-NESTED.ldif

```

Adjustments in LDIF data

Usually, you do not need to provide new hashes for the user passwords in the LDAP directory.

OID password hashes are located in the LDAP attribute authpassword;oid.

OID hashes can be provided in orclcommonrpdattribute with an {SASL-MD5} prefix.

But this does not mean you have to populate the orclcommonrpdattribute with data from the authpassword attribute, as it can be misunderstood from <http://blogs.oracle.com/sduloutr/migration-from-oid-to-oud:-adapting-eus-metadata> or https://docs.oracle.com/cd/B14099_19/idmanage.1012/b14082/pwdstore.htm

The hashes can just be imported from the LDAP userpassword attribute which is already provided by OID. We just have to delete any "orclpassword" and "authpassword" attributes since they will cause import errors in the OUD schema.

We just strip the "orclpassword" attribute from the export:

```

cat oid-users.ldif | grep -v "authpassword" | grep -v "orclpassword" > oid-users-20170901-FILTERED.ldif

```

The filtered LDI file now looks like this:

```

# alfonso_xxx, Users, cust.de
dn: cn=alfonso_xxx,cn=Users, dc=cust,dc=de
cn: alfonso_xxx
mail: alf.xxx@libero.it
objectclass: top
objectclass: organizationalPerson
objectclass: inetOrgPerson
objectclass: orcluser
objectclass: person
orclglobalid: 000

```

```
orclpkcs12hint: alfonso_xxx
sn: alfonso_xxx
uid: alfonso_xxx
userpassword:: e01ENHlwTlhHZRTEbjJnXXXOcnM0K3HUJJIBPT0=
```

We also didn't find it necessary to provide the SSHA prefix to the password attribute, as one could think after reading Doc ID 2242156.1 or Doc ID 2056712.1.

If you try this, you run into this error:

```
[oracle@oraldap3 Loopback]$ cat t1.ldif
dn: cn=test_oud2,cn=Users,dc=cust,dc=de
cn: test_oud2
mail: test.test_oud2@mycompany.com
objectclass: top
objectclass: inetOrgPerson
objectclass: orcluser
objectclass: organizationalPerson
objectclass: person
orclglobalid: 000
orclpkcs12hint: test_oud2
sn: test_oud2
uid: test_oud2
userPassword: {SSHA}/BfyptGYvxo+nJc18vCbm/i56vHuKIKIr7DSUg==
```

```
[oracle@oraldap3 Loopback]$ ldapmodify -D "cn=Directory Manager" -j /tmp/p -a -f t1.ldif
Processing ADD request for cn=test_oud2,cn=Users,dc=cust,dc=de
ADD operation failed
Result Code: 53 (Unwilling to Perform)
Additional Information: Pre-encoded passwords are not allowed for the password attribute userPassword
```

Import to OUD

Import Groups

We can import the groups LDIF file we have created earlier from OID:

```
[oracle@oraldap3 Loopback]$ /u01/Middleware/Oracle_OUD1/bin/ldapmodify -D "cn=Directory Manager" -j /tmp/p -a -f
oid-groups.ldif
Processing ADD request for cn=OCS_PORTAL_USERS,cn=groups,dc=cust,dc=de
ADD operation successful for DN cn=OCS_PORTAL_USERS,cn=groups,dc=cust,dc=de
Processing ADD request for cn=PORTAL_GROUPS,cn=Groups,dc=cust,dc=de
ADD operation successful for DN cn=PORTAL_GROUPS,cn=Groups,dc=cust,dc=de
Processing ADD request for cn=my_dummy,cn=groups,dc=cust,dc=de
ADD operation successful for DN cn=my_dummy,cn=groups,dc=cust,dc=de
Processing ADD request for cn=CERA_GROUPS,cn=Groups,dc=cust,dc=de
ADD operation successful for DN cn=CERA_GROUPS,cn=Groups,dc=cust,dc=de
Processing ADD request for cn=ADMIN_GROUPS,cn=Groups,dc=cust,dc=de
```


ADD operation successful for DN cn=ADMIN_GROUPS,cn=Groups,dc=cust,dc=de

And the additional nested groups:

```
ldapmodify -D "cn=Directory Manager" -j /tmp/p -a -f oid-groups-NESTED.ldif
```

Import Users

```
ldapmodify -D "cn=Directory Manager" -j /tmp/p -a -f oid-users-20170901-FILTERED.ldif
```

Configuration of a database client

We have to configure the new LDAP server in the database ldap.ora file:

```
[oracle@odal:testdb1 ]$ cat /u01/app/oracle/product/11.2.0.4/dbhome_1/network/admin/ldap.ora
```

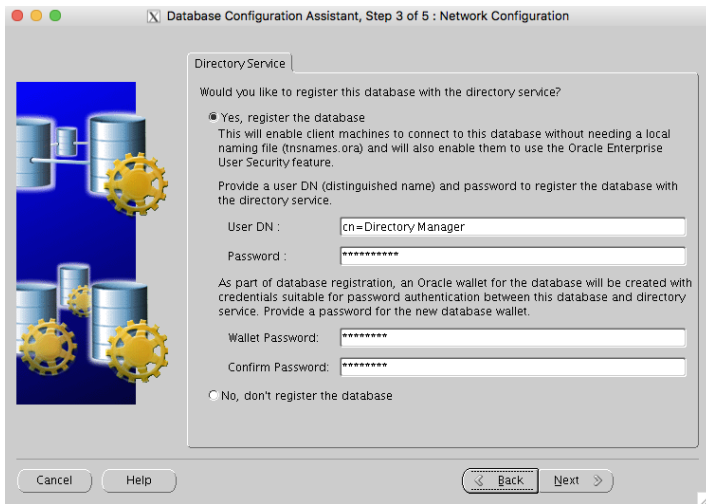
```
DIRECTORY_SERVERS= (oraldap3.cust.de:389:636)
```

```
DEFAULT_ADMIN_CONTEXT = "dc=cust,dc=de"
```

This can also be done by NetCA:



Next, we run the **Database Configuration Assistant (dbca)** to re-register the database with the new OUD LDAP server and its OracleContext:



Let's check the EUS configuration:

```
[oracle@odal:testdb1 ]$ eusm listMappings realm_dn="dc=cust,dc=de" ldap_port=389 ldap_host=oraldap3.cust.de ldap_user_dn="cn=Directory Manager" ldap_user_password=XXX database_name=testdb
```

LIST OF DATABASE SCHEMA MAPPINGS::

```
-----
Mapping Name:  MAPPING0
Mapping Type:  SUBTREE
Mapping DN:    cn=Users,dc=cust,dc=de
Mapping schema:GLOBAL_IDENT
Mapping Level :DATABASE
```

And test database login:

```
[oracle@odal:testdb1 ]$ sqlplus test_oud/dummy
SQL*Plus: Release 11.2.0.4.0 Production on Fri Aug 25 14:41:59 2017
Copyright (c) 1982, 2013, Oracle. All rights reserved.
Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,
Data Mining and Real Application Testing options
SQL> show user;
USER is "GLOBAL_IDENT"
SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,
Data Mining and Real Application Testing options
```

Final migration

After testing, we want to redo the migration to new OUD directory server and connect the productive database. For this, it is the easiest way to delete the previously imported data in OUD and re-import the current data from old OUD.

Re-Export OID data

```

/usr/bin/ldapsearch -p 3060 -h oraldapl.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=users,dc=cust,dc=de" "(objectclass=*)" "*" > oid-users-20171124.ldif

/usr/bin/ldapsearch -p 3060 -h oraldapl.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=Groups,dc=cust,dc=de" "(objectclass=*)" "*" > oid-groups-20171124.ldif

/usr/bin/ldapsearch -p 3060 -h oraldapl.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=CERA_GROUPS,cn=Groups,dc=cust,dc=de" "(objectclass=*)" "*" > oid-groups-20171124-NESTED.ldif
/usr/bin/ldapsearch -p 3060 -h oraldapl.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=ADMIN_GROUPS,cn=Groups,dc=cust,dc=de" "(objectclass=*)" "*" >> oid-groups-20171124-NESTED.ldif

/usr/bin/ldapsearch -p 3060 -h oraldapl.cust.de -D "cn=orcladmin" -w "XXX" -L -s one -b "cn=OracleDefaultDomain,cn=OracleDBSecurity,cn=Products,cn=OracleContext,dc=cust,dc=de" -o ldif-wrap=no "(objectclass=*)" "*" > eus-mappings-20171124.ldif

```

Re-Transform data

```
cat oid-users-20171124.ldif | grep -v "authpassword" | grep -v "orclpassword" > oid-users-20171124-FILTERED.ldif
```

Check numbers

```

[oracle@oraldap3 Loopback]$ cat oid-users-20171124-FILTERED.ldif | grep "^dn: " | wc -l

8498

[oracle@oraldap3 Loopback]$ ldapsearch -h oraldap3 -p 389 -D "cn=Directory Manager" -j ~/.xxx -s one -b "cn=Users,dc=cust,dc=de" -o ldif-wrap=no "(objectclass=*)" "*" > OUD_Users-20171124.ldif
[oracle@oraldap3 Loopback]$ cat OUD_Users-20171124.ldif | grep "^dn: " | wc -l

6314

[oracle@oraldap3 Loopback]$ ldapsearch -h oraldap3 -p 389 -D "cn=Directory Manager" -j ~/.xxx -s one -b "cn=Groups,dc=cust,dc=de" -o ldif-wrap=no "(objectclass=*)" "*" > OUD_Groups-20171124.ldif

```

Delete OUD contents

```

[oracle@oraldap3 Loopback]$ cat OUD_Users-20171124.ldif | grep "^dn: " | sort | uniq > OUDusers.list

[oracle@oraldap3 Loopback]$ ldapdelete -D "cn=Directory Manager" -j ~/.xxx -f a.txt
Processing DELETE request for dn: cn=test,cn=users,dc=cust,dc=de
DELETE operation failed
Result Code: 34 (Invalid DN Syntax)
Additional Information: The provided value "dn: cn=test,cn=users,dc=cust,dc=de" could not be parsed as a valid distinguished name because character ':' at position 2 is not allowed in an attribute name

[oracle@oraldap3 Loopback]$ cat OUDusers.list | awk '{print $2}' > OUDusers-SYNTAXed.list
[oracle@oraldap3 Loopback]$ ldapdelete -D "cn=Directory Manager" -j ~/.xxx -f OUDusers-SYNTAXed.list -c

[oracle@oraldap3 Loopback]$ cat OUD_Groups-20171124.ldif | grep "^dn: " | sort | uniq > OUDgroups.list
[oracle@oraldap3 Loopback]$ cat OUD_Groups-20171124-NESTED.ldif | grep "^dn: " | sort | uniq >> OUDgroups.list

[oracle@oraldap3 Loopback]$ cat eus-mappings-20171124.ldif | grep "^dn: " | sort | uniq | awk '{print $2}' > OUDeusmaps.list
[oracle@oraldap3 Loopback]$ ldapdelete -D "cn=Directory Manager" -j ~/.bindpw -f OUDeusmaps2delete.list -c

```

Re-Import

```
[oracle@oraldap3 Loopback]$ ldapmodify -D "cn=Directory Manager" -j ~/.xxx -a -f oid-users-20171124-FILTERED.
ldif -c > OUD_Users_Import-20171124.log

[oracle@oraldap3 Loopback]$ ldapmodify -D "cn=Directory Manager" -j ~/.xxx -a -f oid-groups-20171124-NESTED.
ldif -c >> OUD_Groups_Import-20171124.log
```

De- and Re-Register productive database

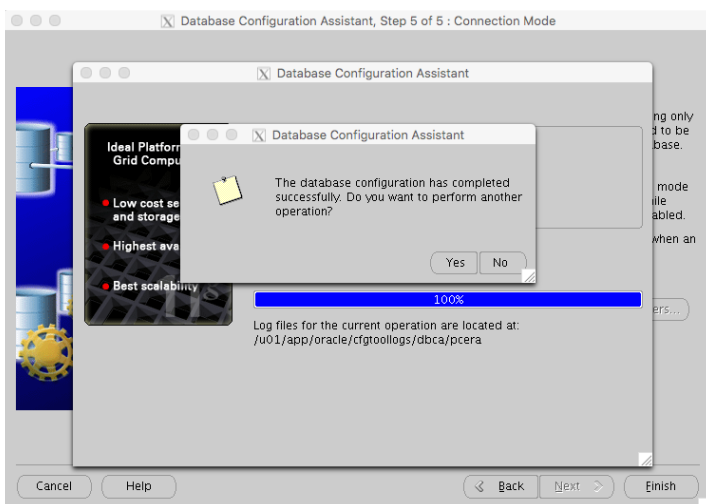
- See whether your old database wallet is still readable / the password is still available. You may re-create your wallet with DBCA later in case the password is lost:

```
[oracle@odal:pcera1 ]$ orapki wallet display -wallet /u01/app/oracle/admin/pcera/wallet/ewallet.p12
Oracle PKI Tool : Version 11.2.0.4.0 - Production
Copyright (c) 2004, 2013, Oracle and/or its affiliates. All rights reserved.
Enter wallet password:
Requested Certificates:
User Certificates:
Oracle Secret Store entries:
ORACLE.SECURITY.DN
ORACLE.SECURITY.PASSWORD
Trusted Certificates:
Subject:      OU=Class 1 Public Primary Certification Authority,O=VeriSign\, Inc.,C=US
Subject:      OU=Class 3 Public Primary Certification Authority,O=VeriSign\, Inc.,C=US
Subject:      OU=Class 2 Public Primary Certification Authority,O=VeriSign\, Inc.,C=US
Subject:      OU=Secure Server Certification Authority,O=RSA Data Security\, Inc.,C=US
Subject:      CN=GTE CyberTrust Global Root,OU=GTE CyberTrust Solutions\, Inc.,O=GTE Corporation,C=US
```

- Run DBCA and de-register the database from the directory
- Edit ldap-ora:

```
[oracle@odal:pcera1 ]$ vi /u01/app/oracle/product/11.2.0.4/dbhome_x/network/admin/ldap.ora
DIRECTORY_SERVERS= (oraldap3.cust.de:389:636)
DEFAULT_ADMIN_CONTEXT = "dc=cust,dc=de"
DIRECTORY_SERVER_TYPE = OID
```

- Run DBCA to register the database with the new directory:



Re-create EUS mappings

```
[oracle@odal:pceral ]$ eusm createMapping realm_dn="dc=cust,dc=de" ldap_port=389 ldap_host=oraldap3.cust.de
ldap_user_dn="cn=Directory Manager" ldap_user_password=XXX database_name="pcera" map_type="SUBTREE" map_dn="
cn=Users,dc=cust,dc=de" schema=PUBLIC_DATASET
```

```
[oracle@odal:pceral ]$ eusm listMappings realm_dn="dc=cust,dc=de" ldap_port=389 ldap_host=oraldap3.cust.de
ldap_user_dn="cn=Directory Manager" ldap_user_password=XXX database_name=pcera
```

```
LIST OF DATABASE SCHEMA MAPPINGS::
```

```
-----
Mapping Name:  MAPPING0
Mapping Type:  SUBTREE
Mapping DN:    cn=Users,dc=cust,dc=de
Mapping schema:PUBLIC_DATASET
Mapping Level :DATABASE
```

Make sure the mappings really point to the public schema used in the production database, in case this is different from how your test database was configured.

Re-test

```
[oracle@odal:pceral ]$ sqlplus akira/XXX
SQL*Plus: Release 11.2.0.4.0 Production on Fri Nov 24 16:04:44 2017
Copyright (c) 1982, 2013, Oracle. All rights reserved.
Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,
Data Mining and Real Application Testing options

SQL>
```

Caveats

Even if the database login works fine in most tools, very old Oracle 9 and JDBC-driven clients may encounter the error:

ORA-28274: No ORACLE password attribute corresponding to user nickname exists

at login after the migration.

The OUD logs and even a trace at database level doesn't log any error in this case.

This is caused by the clients using a very old Oracle database login mechanism called O3LOGON. This expects the LDAP attribute "orclpassword", which may have been present in OID, but we didn't transfer it to OUD, since all this is deprecated and insecure anyway. See also: Doc ID 1073893.1:

Unable To Authenticate Enterprise Users (EUS) To OVD Using 9i RDBMS Clients (Or 10g JDBC Thin Clients): ORA-28274: No ORACLE password attribute corresponding to user nickname exists. (Doc ID 1073893.1)

CAUSE

This is the expected behavior.

Even when using current JDBC drivers, you may encounter this error if JDBC was configured to use O3LOGIN previously by a code line like:

```
// System.setProperty("oracle.jdbc.thinLogonCapability", "o3");
```