Oracle® Banking APIs

Installation Guide

Release 25.1.0.0.0

G28236-02

April 2025

Oracle Banking APIs Installation Guide, Release 25.1.0.0.0

G28236-02

Contents

Preface

1 Introduction

2 Prerequisites

3 Installation

Preface

Purpose

This guide is designed to help acquaint you with the Oracle Banking application. This guide provides answers to specific features and procedures that the user need to be aware of the module to function successfully.

Audience

This document is intended for the following audience:

- Customers
- Partners

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at Critical Patches, Security Alerts and

Bulletins. All critical patches should be applied in a timely manner to ensure effective security, as strongly recommended by Oracle Software Security Assurance.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Related Resources

For more information on any related features, refer to the following documents:

- Oracle Banking APIs Installation Manuals

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

The list of the acronyms and abbreviations used in this guide are as follows:

Table 1   Acronyms and Abbreviations

Abbreviation	Description
OBAPI	Oracle Banking APIs

1
Introduction

1.1 Purpose of the Document

The purpose of the OBAPI Installation Manual is to provide a step by step overview on the installation process of the solution.

It includes:

- Reference to prerequisites software installation required for OBAPI & OBAPI installer
- Setup of OBAPI with Oracle’s own Core Banking and Origination Products along with Third-party HOST system.
- Running the installation in silent mode
- Advanced Configurations (Post installation)
- Installation Verification
- Multi-Entity Installation and configuration
- Best Practice
- Troubleshoot Overview

2
Prerequisites

OBAPI pre-requisite software should be installed and available before proceeding.

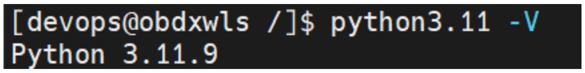
For OBAPI pre-requisite software setup refers document **Oracle Banking Digital Experience Installer Pre-Requisite Setup Manual**.

Installer Pre-requisite Installation

Python 3.11 for Linux Operating System : --

1. Execute below commands to install the python 3.11.9

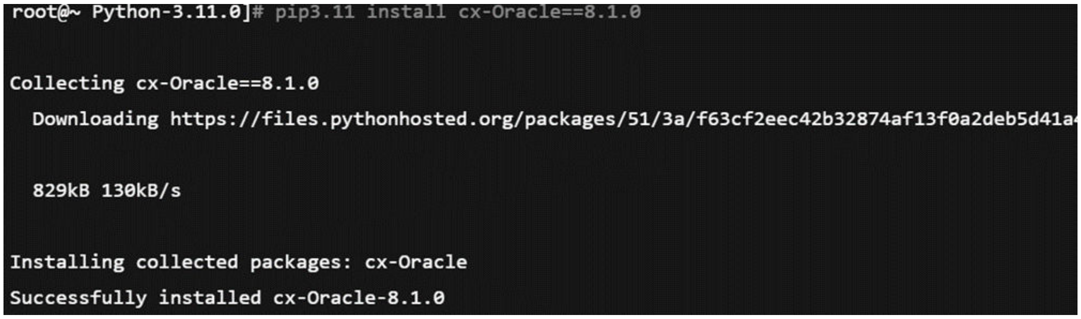
```
dnf
            groupinstall 'development tools'dnf install
            bzip2-devel expat-devel gdbm-devel ncurses-devel openssl-devel
readline-devel wget
            sqlite-devel tk-devel xz-devel zlib-devel
      libffi-develwget
            https://www.python.org/ftp/python/3.11.9/Python-3.11.9.tgz
        tar -xzvf
            Python-3.11.9.tgzcd
            Python-3.11.9./configure
            --enable-optimizationsmake
            altinstallpython3.11
            -version
```



```
[devops@obdxwls /]$ python3.11 -V
Python 3.11.9
```

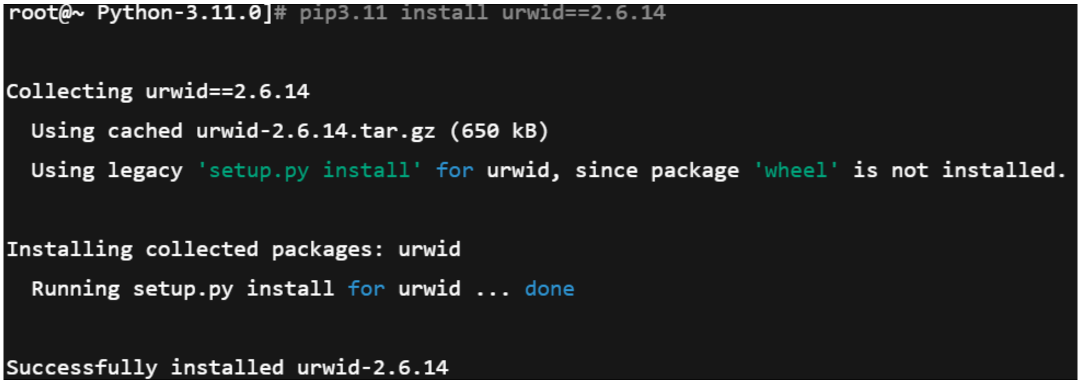
2. Once above steps are executed successfully install the following required modules.

```
pip3.11 install --upgrade pip
pip3.11 install cx-Oracle==8.3
```



```
root@~ Python-3.11.0]# pip3.11 install cx-Oracle==8.1.0

Collecting cx-Oracle==8.1.0
  Downloading https://files.pythonhosted.org/packages/51/3a/f63cf2eec42b32874af13f0a2deb5d41a

  829kB 130kB/s

Installing collected packages: cx-Oracle
Successfully installed cx-Oracle-8.1.0
```

```
pip3.11 install urwid==2.6.14
```



```
root@~ Python-3.11.0]# pip3.11 install urwid==2.6.14

Collecting urwid==2.6.14
  Using cached urwid-2.6.14.tar.gz (650 kB)
  Using legacy 'setup.py install' for urwid, since package 'wheel' is not installed.

Installing collected packages: urwid
  Running setup.py install for urwid ... done

Successfully installed urwid-2.6.14
```

Installer Pre-requisite verification

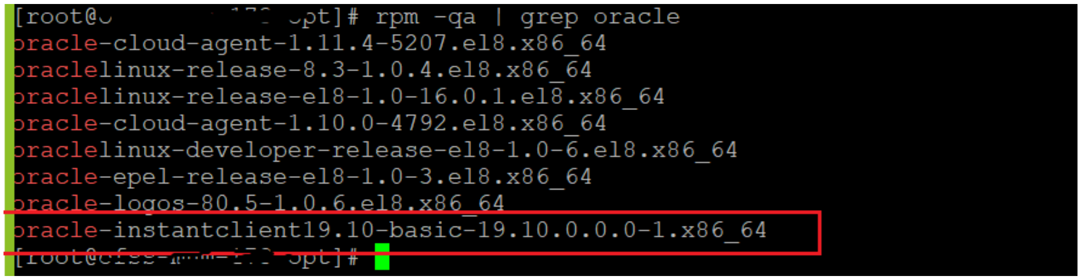
Post installation of OBAPI Installer prerequisite software's, verification can be done using below steps.

> **Note:**
>
> Verification should be performed on Server where Oracle Weblogic is locally installed and by OS user (which is owner for Oracle Weblogic home directory) for non-root steps. The same user will be used to execute installer.

Oracle Instant client

1. Login using root user.
2. Run below command to verify if Oracle Instant client is installed.

```
rpm -qa | grep oracle
```



```
[root@[redacted] opt]# rpm -qa | grep oracle
oracle-cloud-agent-1.11.4-5207.el8.x86_64
oraclelinux-release-8.3-1.0.4.el8.x86_64
oraclelinux-release-el8-1.0-16.0.1.el8.x86_64
oracle-cloud-agent-1.10.0-4792.el8.x86_64
oraclelinux-developer-release-el8-1.0-6.el8.x86_64
oracle-epel-release-el8-1.0-3.el8.x86_64
oracle-logos-80.5-1.0.6.el8.x86_64
oracle-instantclient19.10-basic-19.10.0.0.0-1.x86_64
[root@[redacted] opt]#
```

OBAPI pre-requisite software should be installed and available before proceeding.

For OBAPI pre-requisite software setup refers document **Oracle Banking APIs Installer Pre-Requisite Setup Manual** .

Installer Pre-requisite verification

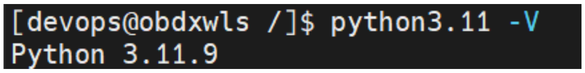
Post installation of OBAPI Installer prerequisite software's, verification can be done using below steps.

> **Note:**
>
> Verification should be performed on Server where Oracle Weblogic is locally installed and by OS user (which is owner for Oracle Weblogic home directory) for non-root steps. The same user will be used to execute installer.

Oracle Instant client

1. Login using root user.
2. Run below command to verify if Oracle Instant client is installed.

```
rpm -qa | grep oracle
```



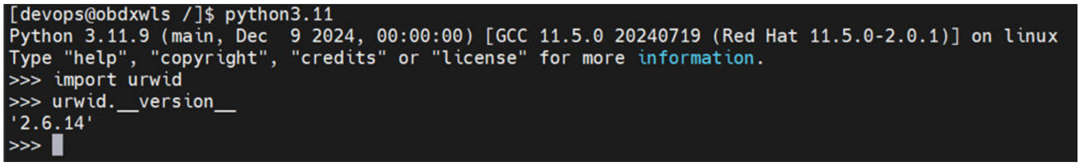
```
[devops@obdxwls /]$ python3.11 -V
Python 3.11.9
```

> **Note:**
>
> Above package verification command is specific to Oracle Linux and RHEL distributions only. For other Linux distributions or OS please refer to OS specific package manager documentation.

Python:

1. Execute python –V command

```
python3.11 -V
```



```
[devops@obdxwls /]$ python3.11
Python 3.11.9 (main, Dec  9 2024, 00:00:00) [GCC 11.5.0 20240719 (Red Hat 11.5.0-2.0.1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import urwid
>>> urwid.__version__
'2.6.14'
>>> 
```

> **Note:**
>
> Ensure Python 3.11.9 supported version is installed. Above command should reflect the same.

cx_Oracle & Urwid:

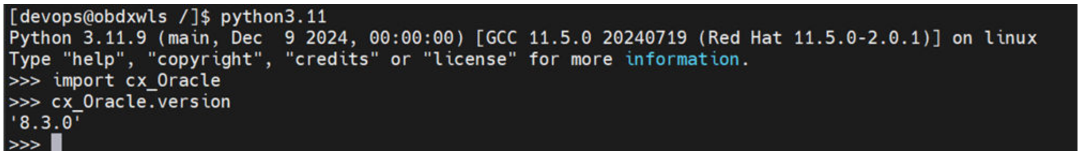
1. Execute python command

   ```
   python
   ```

   > **Note:**
   >
   > Ensure Python 3.11.0 version should be available in PATH variable. Above execution should be done using Python 3.11.0.

2. Import Urwid and check version

   ```
   import urwid (Press Enter)
   urwid.__version__
   ```



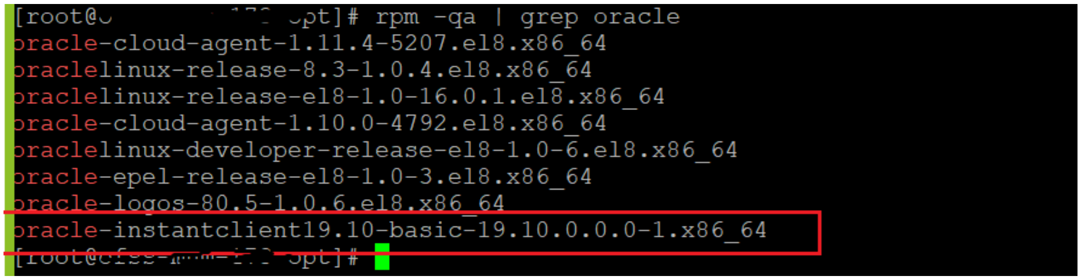
```
[devops@obdxwls /]$ python3.11
Python 3.11.9 (main, Dec  9 2024, 00:00:00) [GCC 11.5.0 20240719 (Red Hat 11.5.0-2.0.1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cx_Oracle
>>> cx_Oracle.version
'8.3.0'
>>> 
```

   If version is displayed, then Urwid is installed and available for use.

   > **Note:**
   >
   > Ensure Urwid 2.6.14 supported version is installed. Above command should reflect the same.

3. Similarly import cx_Oracle and check version

   ```
   import cx_Oracle (Press Enter)
   cx_Oracle.version
   ```



```
[root@[redacted] opt]# rpm -qa | grep oracle
oracle-cloud-agent-1.11.4-5207.el8.x86_64
oraclelinux-release-8.3-1.0.4.el8.x86_64
oraclelinux-release-el8-1.0-16.0.1.el8.x86_64
oracle-cloud-agent-1.10.0-4792.el8.x86_64
oraclelinux-developer-release-el8-1.0-6.el8.x86_64
oracle-epel-release-el8-1.0-3.el8.x86_64
oracle-logos-80.5-1.0.6.el8.x86_64
oracle-instantclient19.10-basic-19.10.0.0.0-1.x86_64
[root@[redacted] opt]#
```

   If version is displayed, then cx_Oracle is installed and available for use.

   > **Note:**
   >
   > Ensure cx_Oracle 8.3.3 supported version is installed. Above command should reflect the same.

3

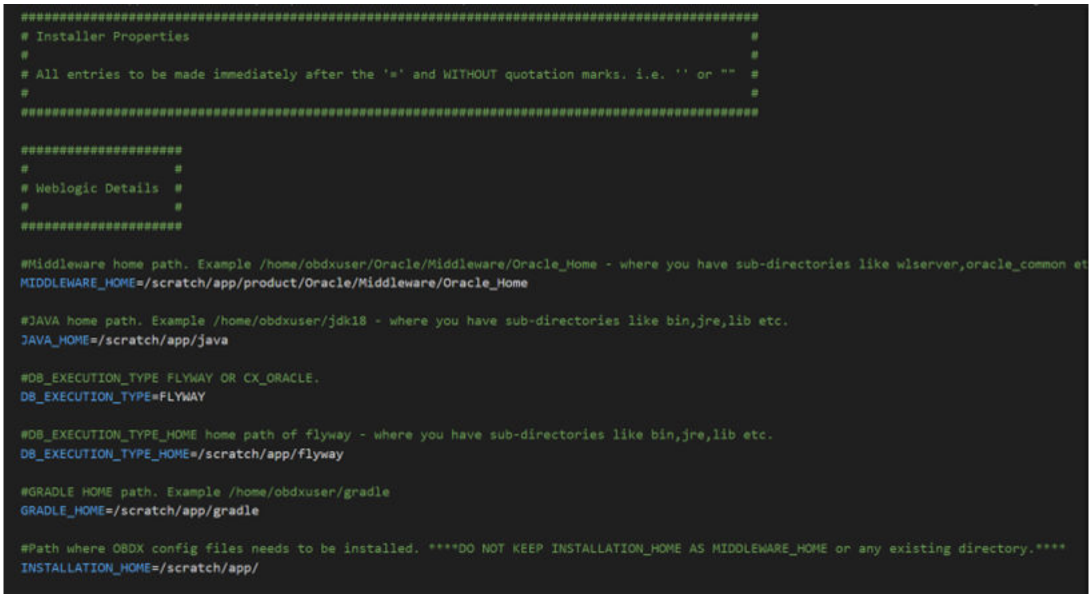
Installation

Pre-Installation

- Install all the prerequisite software and packages mentioned above

Steps of Installation

- Download and extract the installer zip file (Base).
- Navigate to “<OBAPI INSTALLER DIR>/ installables/OBAPI/BASE/25.1.0.0.0/config.
- Open the “installer.properties” file to maintain key configurations for BASE ENTITY (OBAPI_BU)



```
################################################################################################
# Installer Properties                                                                         #
#                                                                                              #
# All entries to be made immediately after the '=' and WITHOUT quotation marks. i.e. '' or ""  #
#                                                                                              #
################################################################################################

#####################
#                   #
# Weblogic Details  #
#                   #
#####################

#Middleware home path. Example /home/obdxuser/Oracle/Middleware/Oracle_Home - where you have sub-directories like wlserver,oracle_common et
MIDDLEWARE_HOME=/scratch/app/product/Oracle/Middleware/Oracle_Home

#JAVA home path. Example /home/obdxuser/jdk18 - where you have sub-directories like bin,jre,lib etc.
JAVA_HOME=/scratch/app/java

#DB_EXECUTION_TYPE FLYWAY OR CX_ORACLE.
DB_EXECUTION_TYPE=FLYWAY

#DB_EXECUTION_TYPE_HOME home path of flyway - where you have sub-directories like bin,jre,lib etc.
DB_EXECUTION_TYPE_HOME=/scratch/app/flyway

#GRADLE HOME path. Example /home/obdxuser/gradle
GRADLE_HOME=/scratch/app/gradle

#Path where OBDX config files needs to be installed. ****DO NOT KEEP INSTALLATION_HOME AS MIDDLEWARE_HOME or any existing directory.****
INSTALLATION_HOME=/scratch/app/
```

IMPORTANT:

- Enter the values right after the **“=”**sign
- DO NOT change anything to the left of the **“=”**
- DO NOT change any of the flag values or pre-filled values (such as **`WLS_JDBC_DIGX_NAME`**, **`WLS_JDBC_DIGX_JNDI`**, Flag values etc) available in **“Factory Shipped”** section.
- Ensure there is no blank space after **“=”** sign, except specific flavor specific configuration.
- Throughout this document consider UBS as UBS core banking with OBPM as payments engine.

Only below parameters should be set in **installer.properties** file

Component	Parameter	Description	Example
DB details (for OBAPI schema)	`OBAPI_DATABASE_HOSTNAME`	Enter the hostname of the database server which would host the database schema for OBAPI	abc.xyc.com
	`OBAPI_DATABASE_PORT`	Enter the port number of the database listener	1521
	`OBAPI_DATABASE_SID`	Enter the Oracle Service Name for database instance	obdxdb.in.oracle.com
	`OBAPI_DATABASE_SYS_USER`	Enter the username with 'sys' privileges	Sys
	`POST_FIX`	For OBAPI schema name like "OBAPI_DEV" POST FIX is 'DEV'. **SHOULD BE IN UPPERCASE ONLY.**	DEV
	`OBAPI_DBA_DIRECTORY_NAME`	Enter the directory name in which you want the OBAPI schema tablespace datafile to be created. Enter Logical name (i.e. `DIRECTORY_NAME` column) from `DBA_DIRECTORIES` table NOT the physical path.	OBAPI_DIR
	`OBAPI_AUDIT_DBA_DIRECTORY_NAME`	Enter the directory name in which you want the OBAPI AUDIT tablespace datafile to be created. Enter Logical name (i.e. `DIRECTORY_NAME` column) from `DBA_DIRECTORIES` table NOT the physical path.	OBAPI_AUDIT_DIR
EHMS DB details (to be configured only in-case of FLAVOR as UBS,FCORE&OBPM)	`EHMS_DATABASE_HOSTNAME`	Enter the hostname for EHMS database server	abc.xyz.com
	`EHMS_DATABASE_PORT`	Enter the port number of EHMS database listener	1521
	`EHMS_SCHEMA_NAME`	Enter the Complete OBAPI-EXT (B1A1) HostInterfaceschema name you want installer to create as new schema. **SHOULD BE IN UPPERCASE ONLY.**	EHMS182SCHEMA

Component	Parameter	Description	Example
	`EHMS_DBA_DIRECTORY_NAME`	Enter the directory name in which you want the OBAPI-EXT (B1A1) schema tablespace datafile to be created. Enter Logical name (i.e.`DIRECTORY_NAME` column) from `DBA_DIRECTORIES` table NOT the physical path.	OPATCH_LOG_DIR
	`EHMS_DATABASE_SYS_USER`	Enter the username with 'sys' privileges	Sys
	`EHMS_DATABASE_SID`	Enter the EHMS database Service Name	obdxehms.in.oracle.com
	`EHMS_HOST_SCHEMA_NAME`	Enter the EXISTING EHMS HOST schema name	OBAPIUBS
	`EHMS_CCY`**(to be configured for UBS and OBPM HOST only)**	Enter the Country code for EHMS HOME Branch	GB
	`EHMS_HB` **(to be configured for UBS and OBPM HOST only)**	Enter the Branch code for code for EHMS HOME Branch	AT3
	`EHMS_FCORE_FCUBS_SCHEMA_NAME` **(to be configured for FCORE HOST only)**	FCORE-FCUBS schema name	FCRUBSHOST
Weblogic server details	`MIDDLEWARE_HOME`	Oracle Weblogic Server home path. Example / home/obapiuser/Oracle/ Middleware/ Oracle_Home - where you have sub-directories like wlserver,oracle_common etc.	/home/obapiuser/Oracle/ Middleware/ Oracle_Home
	`JAVA_HOME`	Path where JAVA (JDK) is installed	/home/obapiuser/ jdk17_0_124
	`DB_EXECUTION_TYPE_HOME`	Path where FLYWAY is installed	/home/obapiuser/ (flyway-8.3)
		For CX_ORACLE, no need to update/can be empty	**[No update required for cx_oracle]**
	`DB_EXECUTION_TYPE`	Database execution type	FLYWAY/CX_ORACLE
	`GRADLE_HOME`	Path where GRADLE is installed	/home/obapiuser/ gradle-7.9

Component	Parameter	Description	Example
	`INSTALLATION_HOME`	Path where OBAPI is to be installed. All configuration files will be copied as a sub-directory “config” under this directory. **DO NOT KEEP INSTALLATION_HOME AS MiddlewareHome.**	/home/obapiuser/obapi
	`WLS_DOMAIN_PATH`	Path where OBAPI Weblogic domain should be created. Users can now enter custom path as per their requirements.	/home/obapiuser/domains
	`WLS_CLUSTER_NAME`	Name of cluster; this cluster would have one single managed server.	obapi_cluster
	`WLS_CLUSTER_NODE_HOSTNAME`	Host name or IP address of managed server participating in the cluster. Currently only single node is supported.	abc.xyz.com
	`WLS_ADMIN_SERVER_PORT`	Weblogic AdminServer port. It is the port to access the administration console of the Weblogic server. Generally port 7001 is used as the AdminServer port. Custom port are supported.	7001
	`WLS_ADMIN_SERVER_SSL_PORT`	AdminServer SSL port. It is the port used to securely access (https) the administration console of the Weblogic server.	7002
	`WLS_NODE_PORT`	Node Manager Port. It is the port used by Node Manager to be configured for OBAPI domain. Generally, 5556 is utilized as Node Manager Port. Custom ports are supported.	5556

Component	Parameter	Description	Example
	`WLS_MS_SERVER_NAME`	Managed server name. This will be the name of the managed server created in the cluster followed by indexes. eg- If this is set as 'clip' managed servers would be clip1.	Clip
	`WLS_MS_SERVER_PORT`	Managed Server Port. Managed server will utilize this port for hosting OBAPI components and associated resources. Custom ports are supported.	9001
	`WLS_DOMAIN_NAME`	Enter Weblogic Domain name.	obapi_domain1
	`WLS_DOMAIN_ADMIN_USER`	Domain user ID. The user id will be used to access the Weblogic Administration console.	weblogic
	WLS_NODE_TYPE	Weblogic Node Manager type	Plain/SSL
	WLS_MACHINE_NAME	Weblogic Node Manager machine name	obapi_machine
	APP_ROOT_DIR	Any empty directory path	/scratch/app/dir
	`WLS_JMS_FILEUPLOAD_PS` **(to be configured for all OBAPI supported HOST)**	Set the paths for the persistent store of the FileUpload JMS modules. **DO NOT KEEP path as INSTALLATION_HOME or as sub directory inside INSTALLATION_HOME.**	/scratch/obapi/FileUpload
	`WLS_JMS_AUDIT_PS` **(to be configured for all OBAPI supported HOST)**	Set the paths for the persistent store of the Audit JMS modules. **DO NOT KEEP path as INSTALLATION_HOME or as sub directory inside INSTALLATION_HOME.**	/scratch/obapi/Audit

Component	Parameter	Description	Example
	`WLS_JMS_REPORT_PS` (to be configured for all OBAPI supported HOST)	Set the paths for the persistent store of the Reports JMS modules. **DO NOT KEEP path as INSTALLATION_HOME or as sub directory inside INSTALLATION_HOME.**	/scratch/obapi/Reports
	`WLS_JMS_JPA_PS` (to be configured for all OBAPI supported HOST)	Set the paths for the persistent store of the JPA JMS modules. **DO NOT KEEP path as INSTALLATION_HOME or as sub directory inside INSTALLATION_HOME.**	/scratch/obapi/JPA
	`WLS_JMS_EXTSYSRECEIVER_PS` (to be configured for all OBAPI supported HOST)	Set the paths for the persistent store of the ExtSystemReceiver JMS modules. **DO NOT KEEP path as INSTALLATION_HOME or as sub directory inside INSTALLATION_HOME.**	/scratch/obapi/Receiver
	`WLS_JMS_EXTSYSSENDER_PS`(to be configured for all OBAPI supported HOST)	Set the paths for the persistent store of the ExtSystemSender JMS modules. **DO NOT KEEP path as INSTALLATION_HOME or as sub directory inside INSTALLATION_HOME.**	/scratch/obapi/Sender

Component	Parameter	Description	Example
OBAPI Application Administrator user details	`OBAPI_ADMIN_USERNAME`	Set username for OBAPI application Admin user. USERNAME IS CASE SENSITIVE. In-case of OUD as provider username should be the User ID mentioned during user creation steps mentioned in pre-requisite document (refer To create User and mapping it to the Group section)	superadmin
	`OBAPI_ADMIN_EMAIL`	Enter the Email ID for OBAPI application admin user.	superadmin@oracle.com
	`OBAPI_ADMIN_CONTACT_NO`	Enter the mobile number for OBAPI application admin user. **COUNTRY CODE IS MUST.**	+911234567890

> **Note:**
>
> Apart from above any other property values should not be modified.

Ensure `ORACLE_HOME, JAVA_HOME` variable are set and their binaries are available in PATH variable before proceeding.

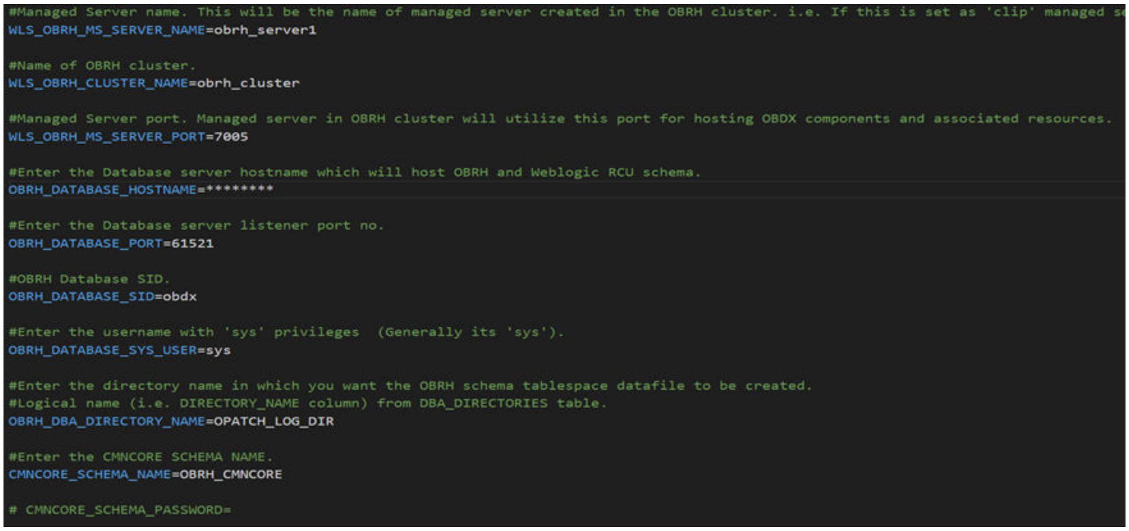
Login with OS user which was used to perform OBAPI pre-requisite software installation (or has ownership on Oracle Weblogic home directory)

Ensure OBAPI Installation home and filestore path's maintained in installer.properties exists and user running the installer has read-write permissions.

> **Note:**
>
> When Proceeding with Product as OBRH or OBAPI+OBRH then only need to update OBRH_config.properties.

Navigate to "<OBAPI INSTALLER DIR>/ installables/OBAPI/BASE/25.1.0.0.0/config"

Open the "OBRH_config.properties" file to maintain key configurations for BASE ENTITY (OBRH).



```
#Managed Server name. This will be the name of managed server created in the OBRH cluster. i.e. If this is set as 'clip' managed se
WLS_OBRH_MS_SERVER_NAME=obrh_server1

#Name of OBRH cluster.
WLS_OBRH_CLUSTER_NAME=obrh_cluster

#Managed Server port. Managed server in OBRH cluster will utilize this port for hosting OBDX components and associated resources.
WLS_OBRH_MS_SERVER_PORT=7005

#Enter the Database server hostname which will host OBRH and Weblogic RCU schema.
OBRH_DATABASE_HOSTNAME=********

#Enter the Database server listener port no.
OBRH_DATABASE_PORT=61521

#OBRH Database SID.
OBRH_DATABASE_SID=obdx

#Enter the username with 'sys' privileges  (Generally its 'sys').
OBRH_DATABASE_SYS_USER=sys

#Enter the directory name in which you want the OBRH schema tablespace datafile to be created.
#Logical name (i.e. DIRECTORY_NAME column) from DBA_DIRECTORIES table.
OBRH_DBA_DIRECTORY_NAME=OPATCH_LOG_DIR

#Enter the CMNCORE SCHEMA NAME.
CMNCORE_SCHEMA_NAME=OBRH_CMNCORE

# CMNCORE_SCHEMA_PASSWORD=
```

IMPORTANT:

- Enter the values right after the “=”sign
- DO NOT change anything to the left of the “=”
- Ensure there is no blank space after “=” sign, except specific flavor specific configuration. Throughout this document consider UBS as UBS core banking with OBPM as payments engine.

Only below parameters should be set in OBRH_config.properties file

Component	Parameter	Description	Example
DB details (for OBAPI schema)	**OBRH_DATABASE_HOSTNAME**	Enter the hostname of the database server which would host the database schema for OBRH	abc.xyc.com
	OBRH_DATABASE_PORT	Enter the port number of the database listener	61521
	OBRH_DATABASE_SID	Enter the Oracle **Service Name** for database instance	obapidb.in.oracle.com
	OBRH_DATABASE_SYS_USER	Enter the username with 'sys' privileges	sys
	PLATO_SCHEMA_NAME	Enter the PLATO SCHEMA name.	OBRH_PLATO
	OBRH_DBA_DIRECTORY_NAME	Enter the directory name in which you want the OBRH schema tablespace datafile to be created. Enter Logical name (i.e. **DIRECTORY_NAME** column) from **DBA_DIRECTORIES** table NOT the physical path.	OPATCH_LOG_DIR

Component	Parameter	Description	Example
Weblogic server details	**WLS_OBRH_MS_SERVER_NAME**	Managed server name. This will be the name of the managed server created in the cluster. eg- **If this is set as “obrh_server” managed servers would be “obrh_server”** ..	obrh_server
	WLS_OBRH_CLUSTER_NAME	Name of cluster; this cluster would have one single managed server.	obrh_cluster
	WLS_OBRH_MS_SERVER_PORT	Managed Server Port. Managed server will utilize this port for hosting OBRH components and associated resources. **Custom ports are supported.**	7005

> **Note:**
>
> : Apart from above any other property values should not be modified

Ensure ORACLE_HOME, JAVA_HOME variable are set and their binaries are available in PATH variable before proceeding.

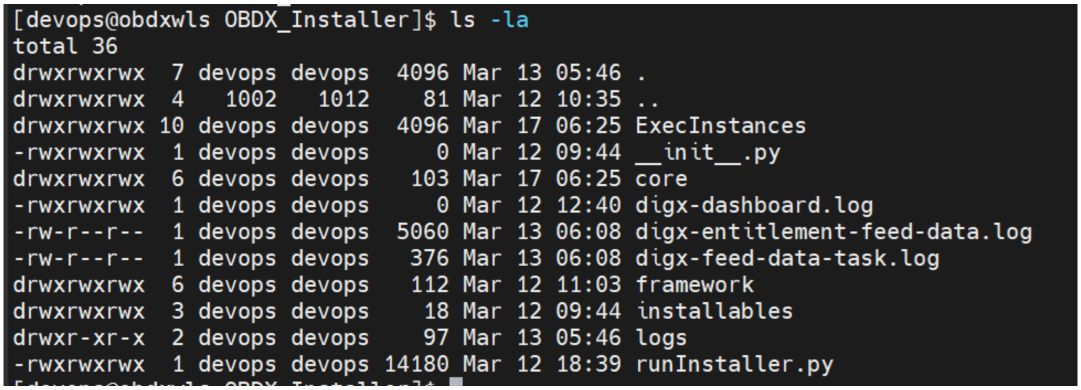
Login with OS user which was used to perform OBRH pre-requisite software installation (or has ownership on Oracle Weblogic home directory)

Ensure OBRH and OBAPI+OBRH Installation home and filestore path’s maintained in installer.properties

And OBRH_config.properties exists and user running the installer has read-write permissions.

Installation Steps:

- From your terminal navigate to <OBAPI INSTALLER DIR>/



```
[devops@obdxwls OBDX_Installer]$ ls -la
total 36
drwxrwxrwx  7 devops devops  4096 Mar 13 05:46 .
drwxrwxrwx  4   1002   1012    81 Mar 12 10:35 ..
drwxrwxrwx 10 devops devops  4096 Mar 17 06:25 ExecInstances
-rwxrwxrwx  1 devops devops     0 Mar 12 09:44 __init__.py
drwxrwxrwx  6 devops devops   103 Mar 17 06:25 core
-rwxrwxrwx  1 devops devops     0 Mar 12 12:40 digx-dashboard.log
-rw-r--r--  1 devops devops  5060 Mar 13 06:08 digx-entitlement-feed-data.log
-rw-r--r--  1 devops devops   376 Mar 13 06:08 digx-feed-data-task.log
drwxrwxrwx  6 devops devops   112 Mar 12 11:03 framework
drwxrwxrwx  3 devops devops    18 Mar 12 09:44 installables
drwxr-xr-x  2 devops devops    97 Mar 13 05:46 logs
-rwxrwxrwx  1 devops devops 14180 Mar 12 18:39 runInstaller.py
```

- Enter the following command

```
python3.11 runInstaller.py
```

- Installation Process

3.1 Installation Process

The installation process involves selecting the product, installation flavor, and other required configurations.

> **Note:**
>
> - The **Back** button is available on every screen, allowing users to navigate one step back.
> - The **Quit** option is present on every page, enabling users to terminate the installation.

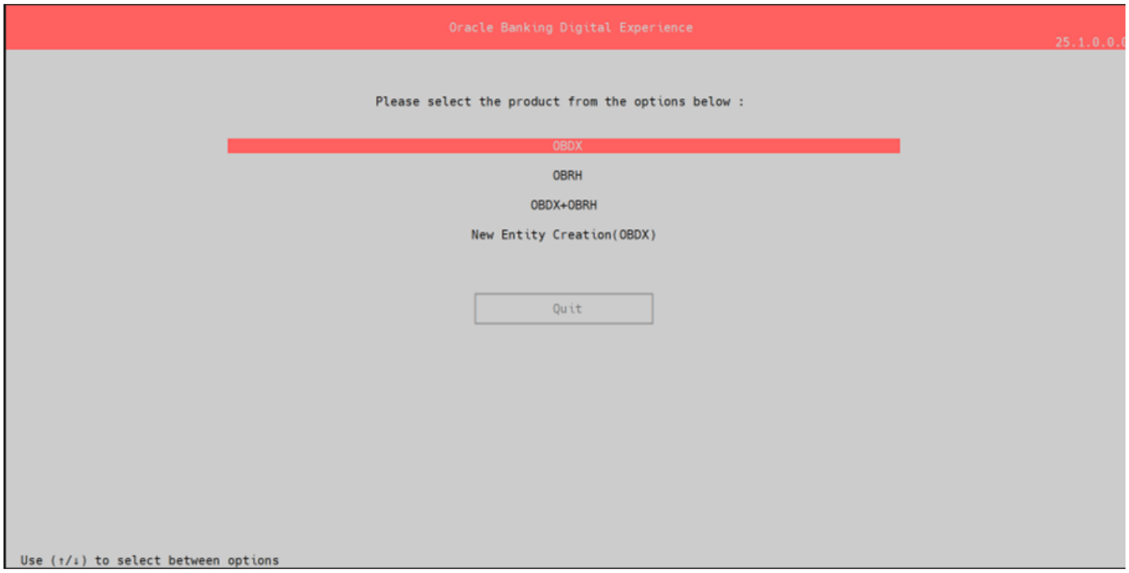
- Step 1: Selecting the Product
- Step 2: Selecting the Installation Flavour
- Step 3: Selecting the Installation Type
- Step 4: Selecting the Installation Version
- Step 5: Selecting the Component
- Step 6: Selecting the Installation Mode
- Step 7: Verification Screen
- Step 8: Entering Required Credentials
- Step9: OBAPI Routing Hub screen appears
- Step10: Finalizing Installation

3.1.1 Step 1: Selecting the Product

You will be prompted to select one of the following products:

> **Note:**
>
> Select **OBDX+OBRH** if you do not have any OBMA product Installed. If you want to Install OBRH then please Install from OBMA products.

1. **OBDX**
2. **OBRH (Not Supported)**
3. **OBDX + OBRH**
4. **New Entity Creation (OBDX)**

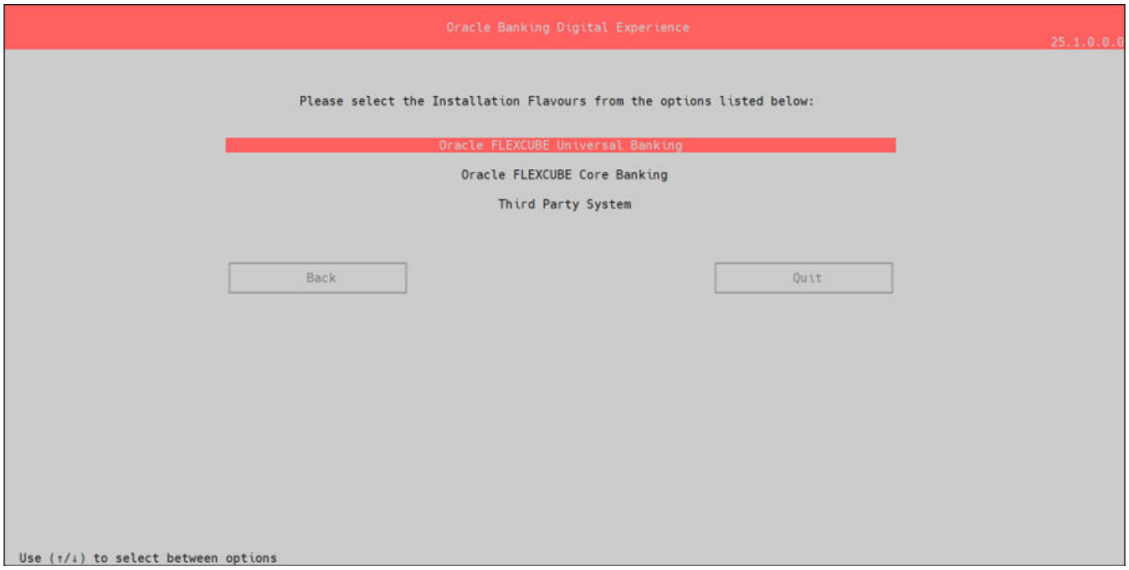


```
Oracle Banking Digital Experience
25.1.0.0.0

Please select the product from the options below :

OBDX
OBRH
OBDX+OBRH
New Entity Creation(OBDX)
```

3.1.2 Step 2: Selecting the Installation Flavour

Next, choose one of the three installation flavours:

1. **Oracle FLEXCUBE Universal Banking (OBPM)**
2. **Oracle FLEXCUBE Core Banking (FCORE)**
3. **Third Party System (THP)**



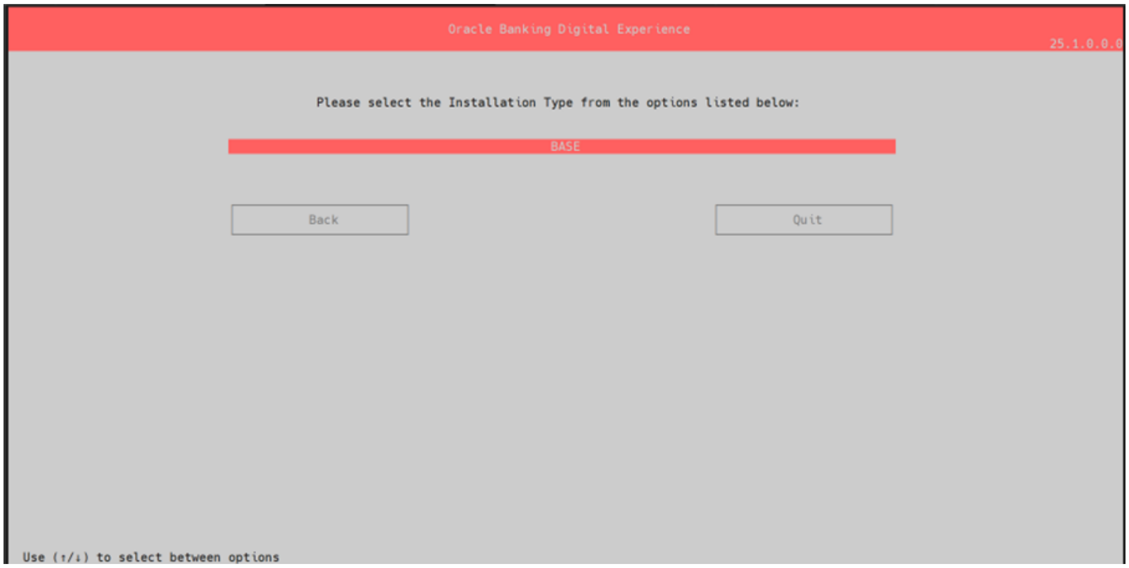
```
Oracle Banking Digital Experience
25.1.0.0.0

Please select the Installation Flavours from the options listed below:

Oracle FLEXCUBE Universal Banking
Oracle FLEXCUBE Core Banking
Third Party System
```

3.1.3 Step 3: Selecting the Installation Type

Select the **Installation Type** as:

- **Base** (for new installations)

If need to go 1 step back, need to select back. Else proceed with selecting Installation Flavour.



Oracle Banking Digital Experience

25.1.0.0.0

Please select the Installation Type from the options listed below:

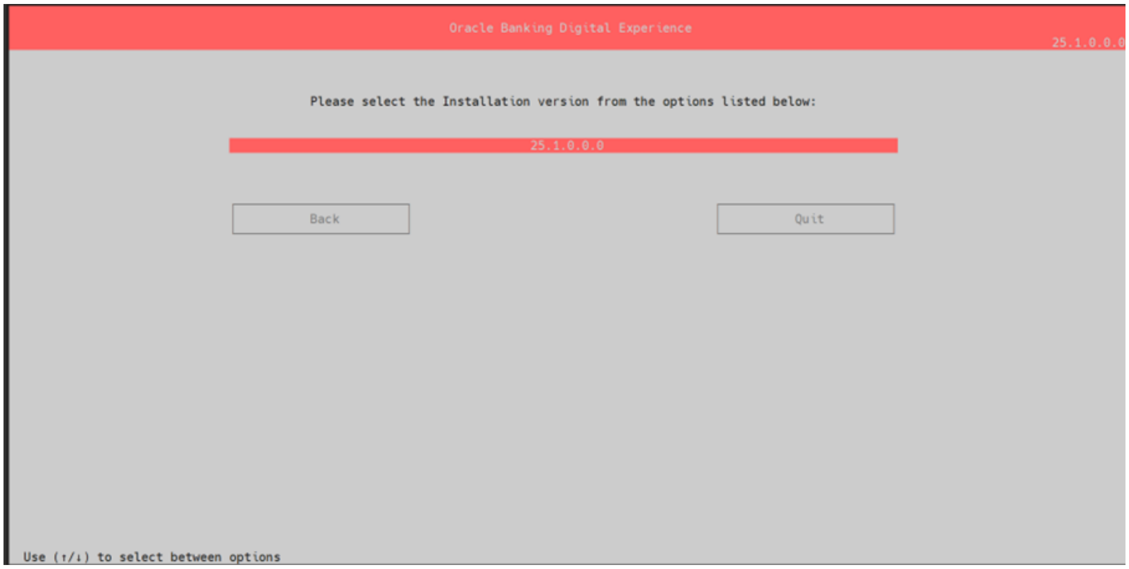
BASE

Use (↑/↓) to select between options

3.1.4 Step 4: Selecting the Installation Version

Choose the **Installation Version** as:

- **25.1.0.0.0** (in our case)



Oracle Banking Digital Experience

25.1.0.0.0

Please select the Installation version from the options listed below:

25.1.0.0.0

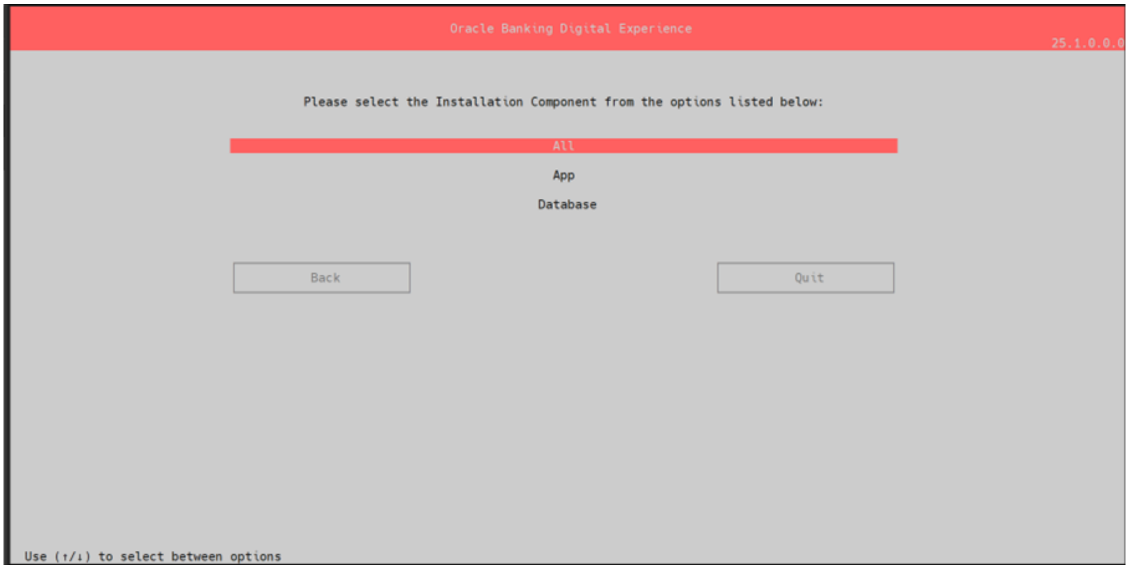
Use (↑/↓) to select between options

3.1.5 Step 5: Selecting the Component

You have three options to choose from:

1. **All** – Installs both **Database** and **Application**
2. **App** – Installs only the **Application**

3. **Database** – Installs only the **Database**



Please select the Installation Component from the options listed below:

Use (↑/↓) to select between options

3.1.6 Step 6: Selecting the Installation Mode

Choose the appropriate installation mode:

1. **New Installation** – Use this option for a fresh installation of **DB** or **App**.
2. **Reinstall** – Use this option to delete the existing component and reinstall it.

In-case of an existing OBDX installation that you want to overwrite OR in case of a previously failed installation user can opt for this option.

Pre-requisites

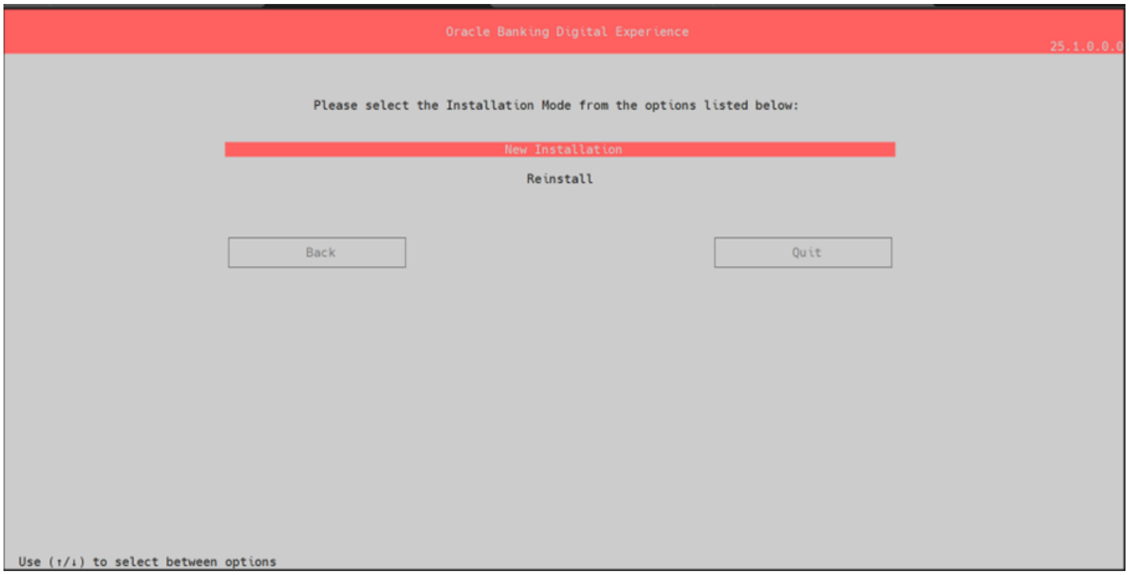
- Weblogic domain processes should be down (i.e. AdminServer, NodeManager, Managed Servers, Derby etc)
- No open session (user should not be logged-in) with OBDX database schema (and OBDX EHMS schema in-case of OBDX UBS;OBPM and FCORE flavor).

Over-write the policies files (Day0Policy.csv; Entitlement.csv; Resources.csv and Task.csv) from OBDX Product zip into <OBDX INSTALLER DIR>/installables/policies directory

Key pointers

- OBDX schema (and OBDX EHMS schema in-case of OBDX UBS flavor) would be dropped and recreated (as per installer.properties). Tablespace would be re-used.
- Weblogic domain (as per installer.properties) would be deleted and created again.
- Installation Home would be cleaned up (all files/ sub-directories would be deleted) and re-created again.

> **Note:**
>
> All input screens are similar to new installation option and as per the host system opted.



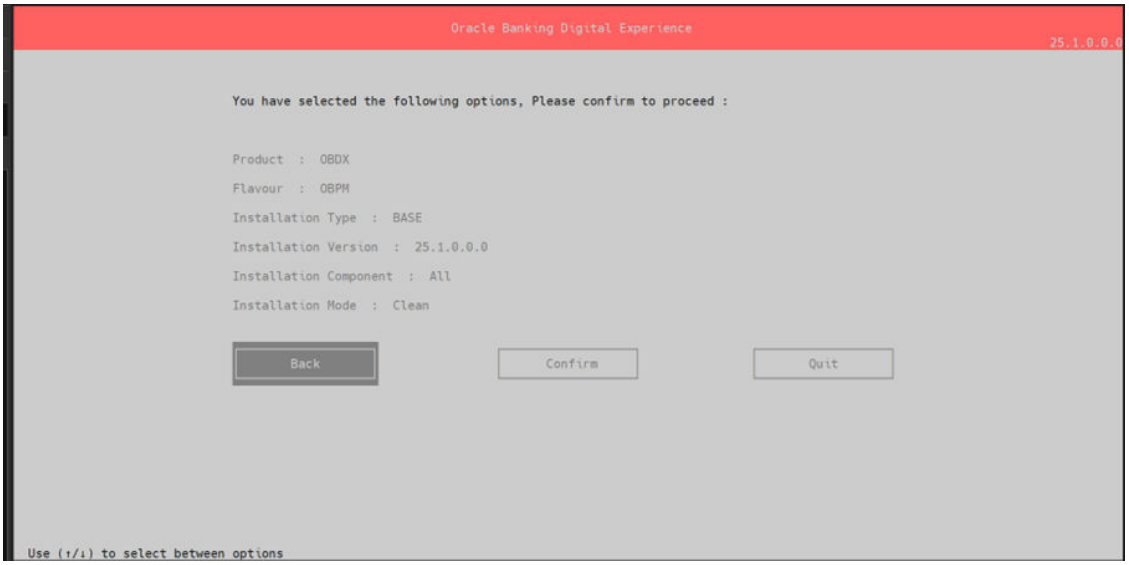
```
Oracle Banking Digital Experience
25.1.0.0.0

Please select the Installation Mode from the options listed below:

New Installation
Reinstall

Use (↑/↓) to select between options
```

3.1.7 Step 7: Verification Screen

At this step, verify the selected configurations:

- **Product**
- **Flavour**
- **Installation Type**
- **Installation Version**
- **Installation Component**
- **Installation Mode**

If everything is correct, proceed by selecting **Confirm**. If there are any mismatches, select **Back** and correct the selections.



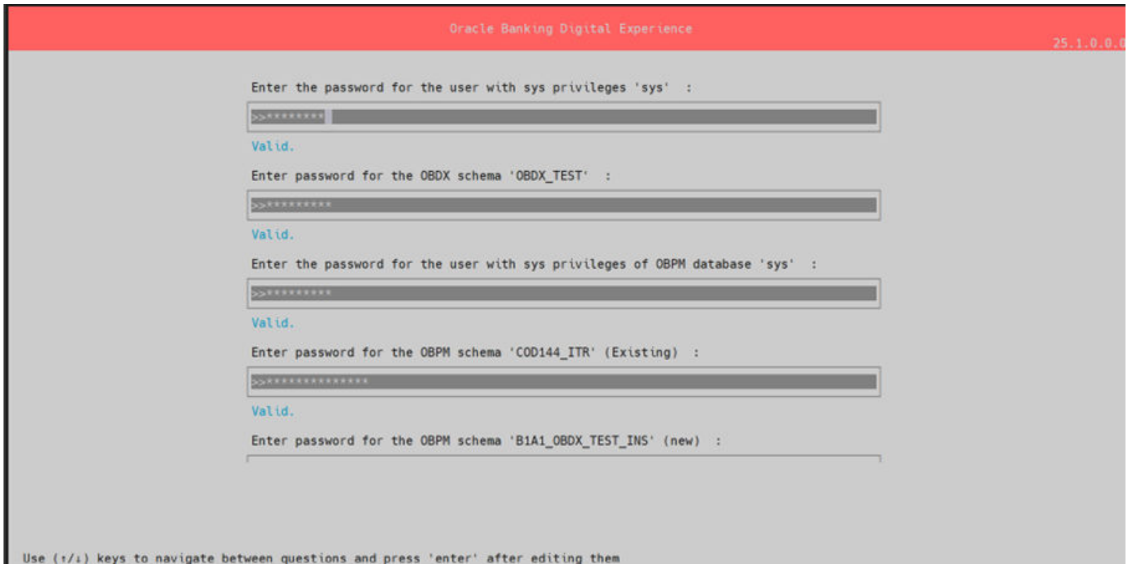
```
Oracle Banking Digital Experience
25.1.0.0.0

You have selected the following options, Please confirm to proceed :

Product  :  OBDX
Flavour  :  OBPM
Installation Type  :  BASE
Installation Version  :  25.1.0.0.0
Installation Component  :  All
Installation Mode  :  Clean

Use (↑/↓) to select between options
```

3.1.8 Step 8: Entering Required Credentials

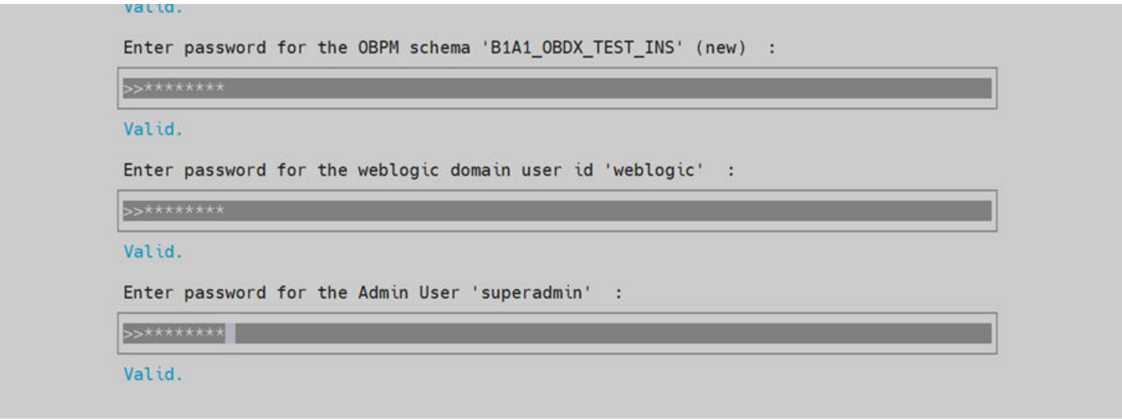
You will be prompted to enter the following passwords during installation:

FOR OBPM:

1. **New SYS Password** for the **Database**
2. **New Schema Password** (For Schema Name mentioned in the property file)
3. **Existing SYS Password** for the **Database**
4. **Existing Database Schema Password**
5. **New Schema Password for B1A1**
6. **WebLogic Password** (for WebLogic administration)
7. **Superadmin User Password**



```
Oracle Banking Digital Experience
                                                         25.1.0.0.0

Enter the password for the user with sys privileges 'sys'  :
>>*********
Valid.
Enter password for the OBDX schema 'OBDX_TEST'  :
>>**********
Valid.
Enter the password for the user with sys privileges of OBPM database 'sys'  :
>>**********
Valid.
Enter password for the OBPM schema 'COD144_ITR' (Existing)  :
>>***************
Valid.
Enter password for the OBPM schema 'B1A1_OBDX_TEST_INS' (new)  :

Use (↑/↓) keys to navigate between questions and press 'enter' after editing them
```

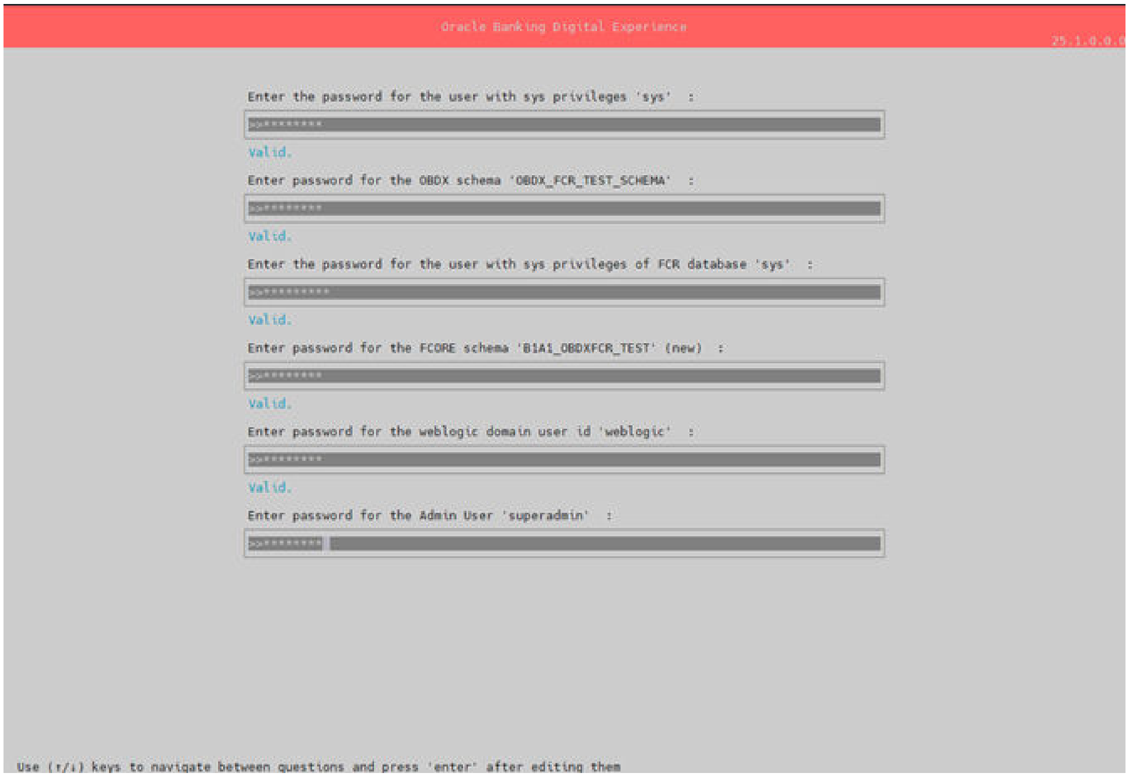


```
Valid.
Enter password for the OBPM schema 'B1A1_OBDX_TEST_INS' (new)  :
>>*********
Valid.
Enter password for the weblogic domain user id 'weblogic'  :
>>*********
Valid.
Enter password for the Admin User 'superadmin'  :
>>*********
Valid.
```

FOR FCORE:

1. **New SYS Password** for the **Database**
2. **New Schema Password** (For Schema Name mentioned in the property file)

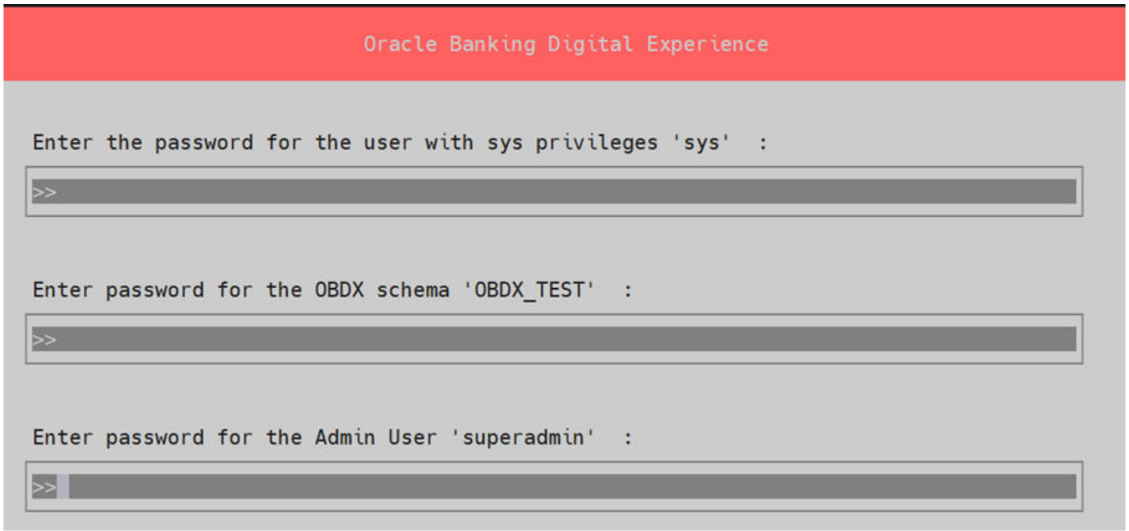
3. **Existing SYS Password** for the **Database**
4. **Existing Database Schema Password**
5. **New Schema Password for B1A1**
6. **WebLogic Password** (for WebLogic administration)
7. **Superadmin User Password**



```
                         Oracle Banking Digital Experience
                                                                              25.1.0.0.0

              Enter the password for the user with sys privileges 'sys'  :
              >>*********
              Valid.
              Enter password for the OBDX schema 'OBDX_FCR_TEST_SCHEMA'  :
              >>*********
              Valid.
              Enter the password for the user with sys privileges of FCR database 'sys'  :
              >>**********
              Valid.
              Enter password for the FCORE schema 'B1A1_OBDXFCR_TEST' (new)  :
              >>*********
              Valid.
              Enter password for the weblogic domain user id 'weblogic'  :
              >>*********
              Valid.
              Enter password for the Admin User 'superadmin'  :
              >>*********

 Use (↑/↓) keys to navigate between questions and press 'enter' after editing them
```

FOR THP:

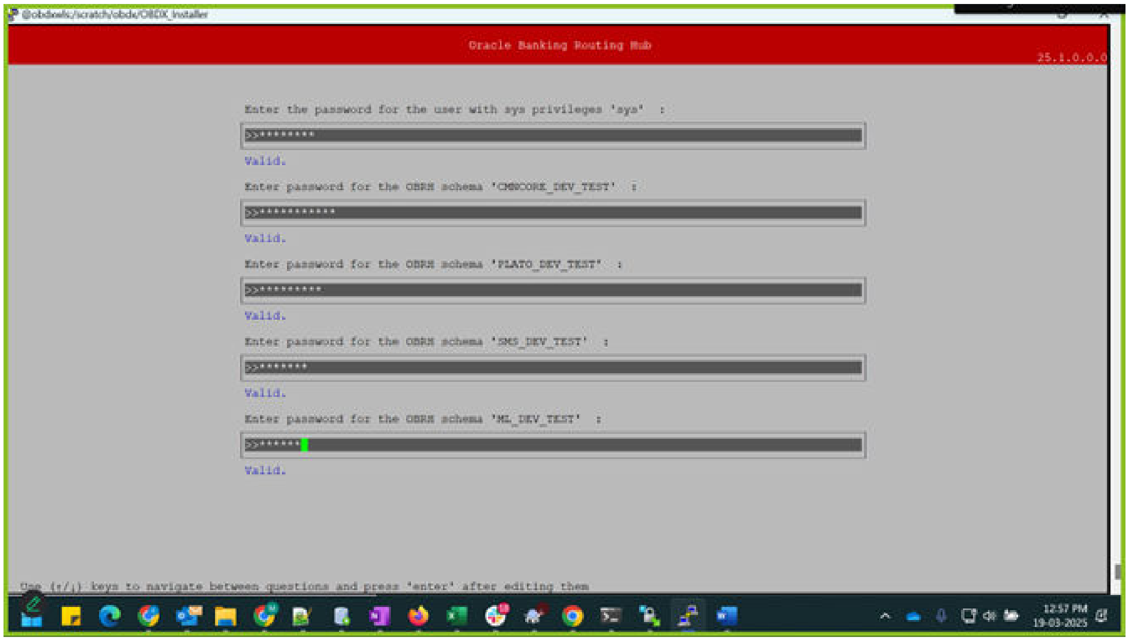
1. **New SYS Password** for the **Database**
2. **New Schema Password** (For Schema Name mentioned in the property file)
3. **Superadmin User Password**



```
Oracle Banking Digital Experience

Enter the password for the user with sys privileges 'sys'  :
>>

Enter password for the OBDX schema 'OBDX_TEST'  :
>>

Enter password for the Admin User 'superadmin'  :
>>
```

3.1.9 Step9: OBAPI Routing Hub screen appears

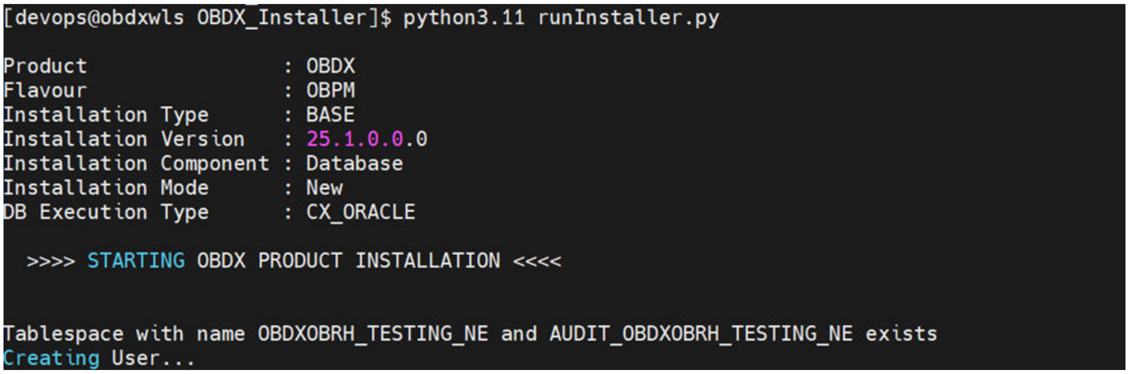
In this screen, need to fill schema passwords and sys password details [THIS SCREEN WILL APPEAR ONLY WHEN SELECTING (OBAPI+OBRH) AS PRODUCT



```
@obdxwls:/scratch/obdx/OBDX_Installer

                    Oracle Banking Routing Hub
                                                          25.1.0.0.0

Enter the password for the user with sys privileges 'sys'  :
>>*********
Valid.
Enter password for the OBRH schema 'CMNCORE_DEV_TEST'  :
>>************
Valid.
Enter password for the OBRH schema 'PLATO_DEV_TEST'  :
>>**********
Valid.
Enter password for the OBRH schema 'SMS_DEV_TEST'  :
>>********
Valid.
Enter password for the OBRH schema 'ML_DEV_TEST'  :
>>*******
Valid.

Use (↑/↓) keys to navigate between questions and press 'enter' after editing them
```

3.1.10 Step10: Finalizing Installation

After entering the required credentials, press **ENTER** to start the final installation process.



```
[devops@obdxwls OBDX_Installer]$ python3.11 runInstaller.py

Product                : OBDX
Flavour                : OBPM
Installation Type      : BASE
Installation Version   : 25.1.0.0.0
Installation Component : Database
Installation Mode      : New
DB Execution Type      : CX_ORACLE

  >>>> STARTING OBDX PRODUCT INSTALLATION <<<<


Tablespace with name OBDXOBRH_TESTING_NE and AUDIT_OBDXOBRH_TESTING_NE exists
Creating User...
```

4

Installation using Silent Mode

This chapter describes how to run the OBAPI installer in silent mode.

What is silent-mode installation?

During installation in silent mode, the installation program reads the details for your configuration parameters (flavor; mode; passwords etc) from the environment variables (same session in which installer is executed) and installer.properties that you set before beginning the installation. The installation program does not display any configuration options during the installation process.

Below values to be exported before running installer in silent mode.

```
export PRODUCT=" "

export FLAVOUR=" "

export INSTALLER_VERSION=""

export Installation_Type=" "

export COMPONENT=""

export DB_SYS_PASSWORD=""

export SCHEMA_PASS=""

export DomainPassword=""

export DBAuthPassword=""

export EHMS_SCHEMA_PASS=""

export EHMS_HOST_SCHEMA_NAME_PASS=" "

export ENTITY_EHMS_DATABASE_HOSTNAME=" "

export ENTITY_EHMS_DATABASE_PORT=""

export ENTITY_EHMS_DATABASE_SID=" "

export ENTITY_EHMS_DATABASE_SYS_USER=" "

export ENTITY_EHMS_DATABASE_SYS_PASS=" "

export ENTITY_SCHEMA_NAME=””

export ENTITY_SCHEMA_PASS=" "

export ENTITY_EHMS_HOST_SCHEMA_NAME=" "

export ENTITY_EHMS_HOST_SCHEMA_NAME_PASS=" "
```

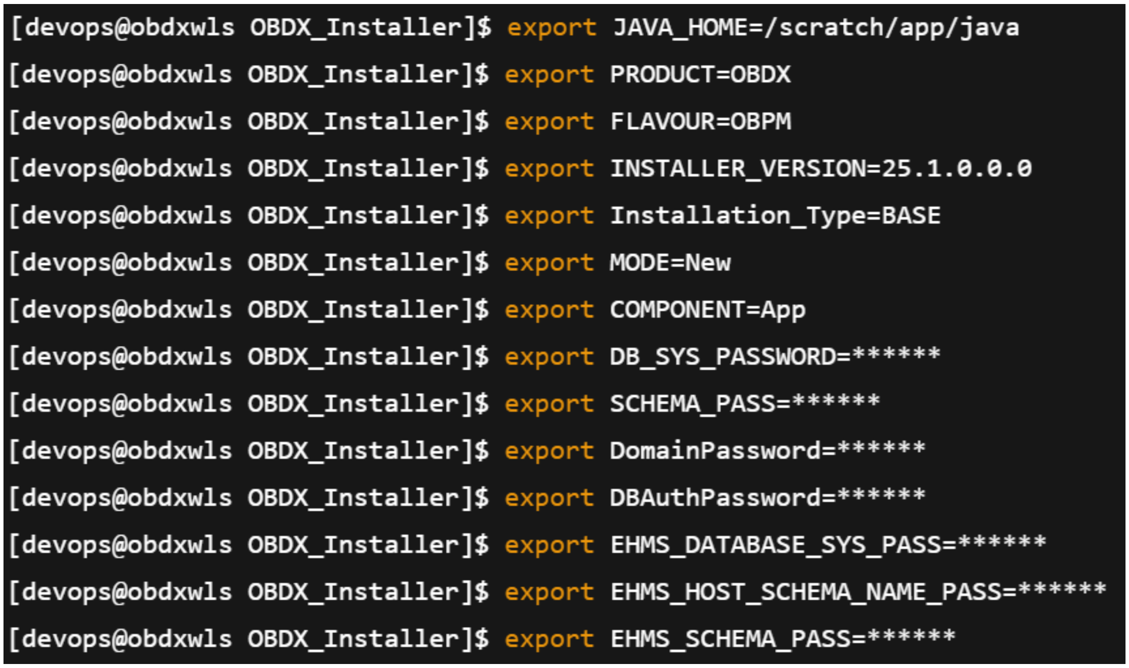
Steps for Silent-Mode Installation

- Download and extract the installer zip file (Base – non localization version).

- – Navigate to **<OBAPI INSTALLER DIR>/installables/OBAPI/BASE/25.1.0.0.0/core/config**
- Open the **installer.properties** file to maintain key configurations for BASE ENTITY (obapi_BU)

**Refer to page 9 to 14 (step 4) for filling up installer.properties.

- Set the environment variables , as shown below



```
[devops@obdxwls OBDX_Installer]$ export JAVA_HOME=/scratch/app/java
[devops@obdxwls OBDX_Installer]$ export PRODUCT=OBDX
[devops@obdxwls OBDX_Installer]$ export FLAVOUR=OBPM
[devops@obdxwls OBDX_Installer]$ export INSTALLER_VERSION=25.1.0.0.0
[devops@obdxwls OBDX_Installer]$ export Installation_Type=BASE
[devops@obdxwls OBDX_Installer]$ export MODE=New
[devops@obdxwls OBDX_Installer]$ export COMPONENT=App
[devops@obdxwls OBDX_Installer]$ export DB_SYS_PASSWORD=******
[devops@obdxwls OBDX_Installer]$ export SCHEMA_PASS=******
[devops@obdxwls OBDX_Installer]$ export DomainPassword=******
[devops@obdxwls OBDX_Installer]$ export DBAuthPassword=******
[devops@obdxwls OBDX_Installer]$ export EHMS_DATABASE_SYS_PASS=******
[devops@obdxwls OBDX_Installer]$ export EHMS_HOST_SCHEMA_NAME_PASS=******
[devops@obdxwls OBDX_Installer]$ export EHMS_SCHEMA_PASS=******
```

Below parameters should be set as environment variables, depending on the Host system the installer should be executed.

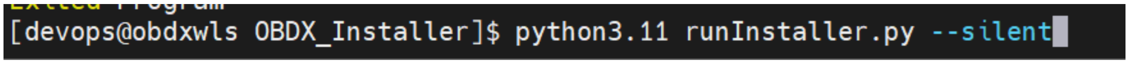
Host	Parameter	Description	Example
Environment variables to set for flavor:UBSFCORE	**FLAVOUR**	Flavour for installation **UBS** for Oracle FLEXCUBE Universal Banking 14.6.0.0.0 (OBAPI with UBS) **FCORE** for Oracle FLEXCUBE Core Banking 11.8.0.0.0 (OBAPI with FCORE)	**export FLAVOUR=OBPM** **or** **export FLAVOUR=FCORE**
	JAVA_HOME	Path for Java	export JAVA_HOME=/scratch/app/java
	PRODUCT	Need to select Product as OBAPI/OBRH/OBAPI+OBRH	export PRODUCT=OBAPI
	INSTALLER_VERSION	Specify Installer Version	**export INSTALLER_VERSION =25.1.0.0.0**

Host	Parameter	Description	Example
	Installation_Type	Specify Type of Installer	**export Installation_Type=BASE**
	MODE	Mode of installation. **New** in-case of a fresh installation of OBAPI for the first run on server **Clean** in-case of an existing OBAPI installation that you want to overwrite OR in case of a previously failed installation or re-installation	**export MODE=New** **or** **export MODE=Clean**
	COMPONENT	Need to specify : App: When only App need to be Installed Database: When only Database needs to be Installed All: When both Database and App needs to be installed	**export COMPONENT=App**
	DB_SYS_PASSWORD	Sys user password of OBAPI database (Existing)	**export DB_SYS_PASSWORD=obapi182sys**
	SCHEMA_PASS	Password for new schema on OBAPI database	**export SCHEMA_PASS=obapi#182**
	DomainPassword	Password for Weblogic Administrator console	**export DomainPassword=wlsadmn**
	EHMS_DATABASE_SYS_PASS	Sys user password of EHMS HOST database (Existing)	**export EHMS_DATABASE_SYS_PASS=obapiehmssys**
	EHMS_HOST_SCHEMA_NAME_PASS ** Only required for UBS & OBPM Host. Ignore this parameter in-case of FCORE Host	Password of existing EHMS HOST schema (Existing)	**export EHMS_HOST_SCHEMA_NAME_PASS =obapiehmshost**
	EHMS_SCHEMA_PASS	**EHMS_SCHEMA_PASS**	Password for new OBAPI EHMS schema on EHMS HOST database

Host	Parameter	Description	Example
	DBAuthPassword	Password for new OBAPI Administrator user of OBAPI application (In-case of OUD as provider, password should similar to one used while user creation in OUD(or User Password field))	**export DBAuthPassword=obapiadmn**
	JAVA_HOME	Path for Java	export JAVA_HOME=/scratch/app/java
	PRODUCT	Need to select Product as OBAPI/OBRH/OBAPI+OBRH	export PRODUCT=OBAPI
Environment variables to set for flavor: **OBAPI (Third-party HOST)**	**FLAVOUR**	Flavour for installation 'OBAPI' for Third Party System 1.0 (OBAPI with THP)	**export FLAVOUR=OBAPI**
	INSTALLER_VERSION	Specify Installer Version	**export INSTALLER_VERSION=25.1.0.0.0**
	Installation_Type	Specify Type of Installer	**export Installation_Type=BASE**
	Mode	Mode of installation. **New** in-case of a fresh installation of OBAPI for the first run on server '**Clean**' in-case of an existing OBAPI installation that you want to overwrite OR in case of a previously failed installation or re-installation	**export MODE=New** **or** **export MODE=Clean**
	COMPONENT	Need to specify : App: When only App need to be Installed Database: When only Database needs to be Installed All: When both Database and App needs to be installed	**export COMPONENT=App**
	DB_SYS_PASSWORD	Sys user password of OBAPI database (Existing)	**export DB_SYS_PASSWORD=obapi182sys**
	SCHEMA_PASS	Password for new schema on OBAPI database	**export SCHEMA_PASS=obapi#182**

Host	Parameter	Description	Example
	DomainPassword	Password for Weblogic Administrator console	**export DomainPassword=wlsadmn**
	wars_to_deploy	Mention the optional wars to deployed	**export wars_to_deploy=digx-cms.war,digx-corporateloan.war,digx-payments.war**
	DBAuthPassword	Password for new OBAPI Administrator user of OBAPI application (In-case of OUD as provider, password should similar to one used while user creation in OUD(or User Password field))	**export DBAuthPassword=obapiadmn**

Run the runInstaller.py file with '--silent ' argument along with '--base' option.

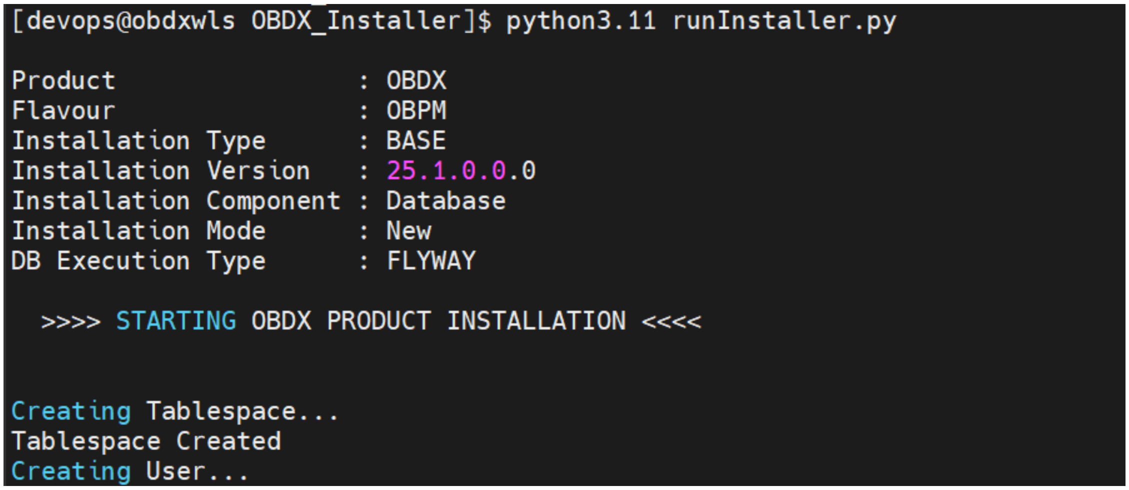


```
Exited Program
[devops@obdxwls OBDX_Installer]$ python3.11 runInstaller.py --silent
```

Installation Status

The status is displayed on the terminal to indicate the progress of the installation.

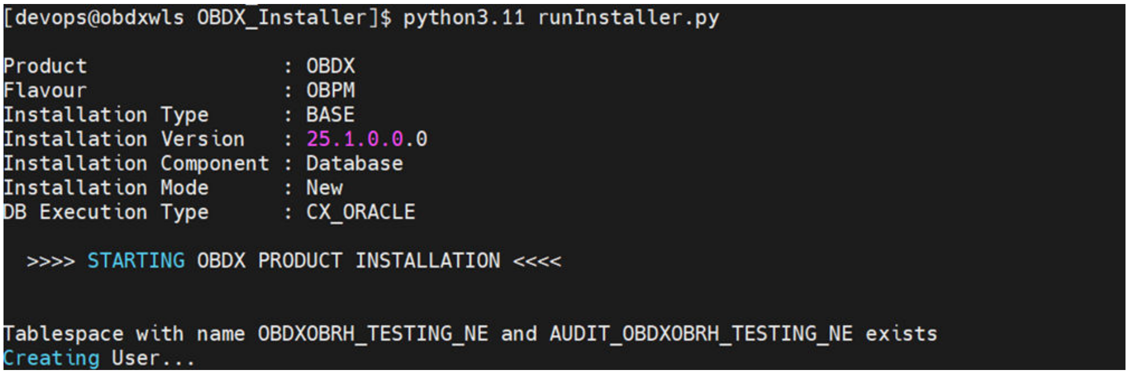
If `DB_EXECUTION_TYPE set to` **`FLYWAY`**



```
[devops@obdxwls OBDX_Installer]$ python3.11 runInstaller.py

Product                  : OBDX
Flavour                  : OBPM
Installation Type        : BASE
Installation Version     : 25.1.0.0.0
Installation Component : Database
Installation Mode        : New
DB Execution Type        : FLYWAY

  >>>> STARTING OBDX PRODUCT INSTALLATION <<<<


Creating Tablespace...
Tablespace Created
Creating User...
```

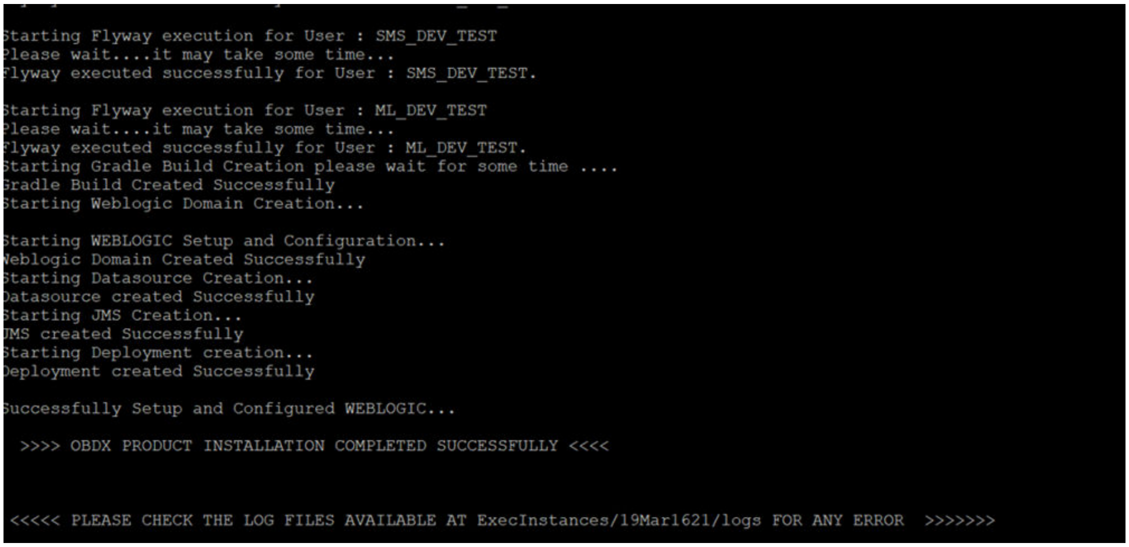
If `DB_EXECUTION_TYPE set to` **`CX_ORACLE`**

When the installation completes, the below message is displayed



```
[devops@obdxwls OBDX_Installer]$ python3.11 runInstaller.py

Product                : OBDX
Flavour                : OBPM
Installation Type      : BASE
Installation Version   : 25.1.0.0.0
Installation Component : Database
Installation Mode      : New
DB Execution Type      : CX_ORACLE

  >>>> STARTING OBDX PRODUCT INSTALLATION <<<<


Tablespace with name OBDXOBRH_TESTING_NE and AUDIT_OBDXOBRH_TESTING_NE exists
Creating User...
```

When the installation completes, the below message is displayed



```
Starting Flyway execution for User : SMS_DEV_TEST
Please wait....it may take some time...
Flyway executed successfully for User : SMS_DEV_TEST.

Starting Flyway execution for User : ML_DEV_TEST
Please wait....it may take some time...
Flyway executed successfully for User : ML_DEV_TEST.
Starting Gradle Build Creation please wait for some time ....
Gradle Build Created Successfully
Starting Weblogic Domain Creation...

Starting WEBLOGIC Setup and Configuration...
Weblogic Domain Created Successfully
Starting Datasource Creation...
Datasource created Successfully
Starting JMS Creation...
JMS created Successfully
Starting Deployment creation...
Deployment created Successfully

Successfully Setup and Configured WEBLOGIC...

 >>>> OBDX PRODUCT INSTALLATION COMPLETED SUCCESSFULLY <<<<


 <<<<< PLEASE CHECK THE LOG FILES AVAILABLE AT ExecInstances/19Mar1621/logs FOR ANY ERROR  >>>>>>>
```

5
Installer Verification

Each execution creates a new directory as <DDMonthHHMM> under <OBAPI INSTALLER DIR>/ExecInstances directory where installer execution logs as described below are stored.

Log Description	PATH
Summarized Installer Activity Log	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/OBAPI_installer.log
Summarized Database Logs	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/DB_installation.log
Detailed OBAPI DB Logs per SQL file	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/OBAPI/OBAPI.log
Detailed EHMS schema Logs per SQL file (specific to EHMS host system only)	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/<EHMSHOST>/<EHMSHOST>.log <EHMSHOST> - values such as; FCORE; OBPM;
Weblogic Configuration Logs	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/app/app_debug.log <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/app/domain.log <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/app/datasource.log <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/app/jms.log <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/app/deployment.log
Detailed OBAPI policy seeding logs	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/Entitlement.log <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/Task.log <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/Dashboard_seed.log **Note:** Check for SEVERE keyword; If found refer to Troubleshot section to re-run the policy
Policy seeding execution Log	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/seedPolicies.log **Note:** Should be empty if no errors during policy execution. In-case non-empty refer to Troubleshot section to re-run the policy

Log Description	PATH
OBRH DB LOGS (When Product is selected as OBRH or OBAPI+OBRH)	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db/OBRH
OBRH APP LOGS (When Product is selected as OBRH or OBAPI+OBRH)	<OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/app

Check all the logs for any errors.

6 Installer Scope

OBAPI Installer currently covers below activities:

Flavor: Third Party system (OBAPI with THP)

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
OBAPI with THP	OBAPI DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
		Policy Seeding	✓	✓
	Weblogic Setup and Configuration	Create and Configure AdminServer, Machine, Managed Server and Cluster	✓	✓
		Configure NodeManager	✓	✓
		Configure JDBC	✓	✓
		JMS servers, Persistent stores and JMS Modules	✓	✓
		Application Deployment	✓	✓
		JTA	✓	✓
		Enable Production Mode	✓	✓
		Start AdminServer and NodeManager	✓	✓
	OBAPI Configuration	Copy config files into OBAPI Installation Home	✓	✓ (Delete old and copy new from installer zip)

Flavor: Oracle FLEXCUBE Core Banking (OBAPI with FCORE)

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
OBAPI with THP	OBAPI DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
		Policy Seeding	✓	✓
	EHMS DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
	Weblogic Setup and Configuration	Create and Configure AdminServer, Machine, Managed Server and Cluster	✓	✓
		Configure NodeManager	✓	✓
		Configure JDBC	✓	✓
		JMS servers, Persistent stores and JMS Modules	✓	✓
		Application Deployment	✓	✓
		JTA	✓	✓
		Enable Production Mode	✓	✓
		Start AdminServer and NodeManager	✓	✓
	OBAPI Configuration	Copy config files into OBAPI Installation Home	✓	✓(Delete old and copy new from installer zip)

Flavor: Oracle FLEXCUBE Universal Banking with Oracle Banking Payments (OBAPI with OBPM)

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
OBAPI with OBPM (14.6.0.0.0 version)	OBAPI DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Execute OBPM HOST specific scripts	✓	✓
		Compile Schema	✓	✓
		Policy Seeding	✓	✓

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
	EHMS DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
	Weblogic Setup and Configuration	Create and Configure AdminServer, Machine, Managed Server and Cluster	✓	✓
		Configure NodeManager	✓	✓
		Configure JDBC	✓	✓
		JMS servers, Persistent stores and JMS Modules	✓	✓
		Application Deployment	✓	✓
		JTA	✓	✓
		Enable Production Mode	✓	✓
		Start AdminServer and NodeManager	✓	✓
	OBAPI Configuration	Copy config files into OBAPI Installation Home	✓	✓(Delete old and copy new from installer zip)

OBAPI+OBRH Installer currently covers below activities:

Flavor: Third Party system (OBAPI+OBRH with THP)

Detailed Activity List

New Installation

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
OBAPI+OBRH with THP	OBAPI+OBRH DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
		Policy Seeding	✓	✓

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
	Weblogic Setup and Configuration	Create and Configure AdminServer, Machine, Managed Server and Cluster	✓	✓
		Configure NodeManager	✓	✓
		Configure JDBC	✓	✓
		JMS servers, Persistent stores and JMS Modules	✓	✓
		Application Deployment	✓	✓
		JTA	✓	✓
		Enable Production Mode	✓	✓
		Start AdminServer and NodeManager	✓	✓
	OBAPI+OBRH Configuration	Copy config files into OBAPI+OBRH Installation Home	✓	✓ (Delete old and copy new from installer zip)

Flavor: Oracle FLEXCUBE Core Banking (OBAPI with FCORE)

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
OBAPI+OBRH with FCORE	OBAPI+OBRH DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
		Policy Seeding	✓	✓
	EHMS DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
	Weblogic Setup and Configuration	Create and Configure AdminServer, Machine, Managed Server and Cluster	✓	✓
		Configure NodeManager	✓	✓
		Configure JDBC	✓	✓

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
		JMS servers, Persistent stores and JMS Modules	✓	✓
		Application Deployment	✓	✓
		JTA	✓	✓
		Enable Production Mode	✓	✓
		Start AdminServer and NodeManager	✓	✓
	OBAPI+OBRH Configuration	Copy config files into OBAPI+OBRH Installation Home	✓	✓ (Delete old and copy new from installer zip)

Flavor: Oracle FLEXCUBE Universal Banking with Oracle Banking Payments (OBAPI+OBRH with OBPM)

Flavor	Activity	Detailed Activity List	New Installation	Reinstall
OBAPI+OBRH with OBPM (14.6.0.0.0 version)	OBAPI+OBRH DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Execute OBPM HOST specific scripts	✓	✓
		Compile Schema	✓	✓
		Policy Seeding	✓	✓
	EHMS DB Setup	Create Tablespace	✓	NA
		Create Schema and Role	✓	✓ (drop and re-create objects)
		Grants	✓	✓
		Load DB object (DDL's and DML's)	✓	✓
		Compile Schema	✓	✓
	Weblogic Setup and Configuration	Create and Configure AdminServer, Machine, Managed Server and Cluster	✓	✓
		Configure NodeManager	✓	✓
		Configure JDBC	✓	✓
		JMS servers, Persistent stores and JMS Modules	✓	✓

Flavor	**Activity**	**Detailed Activity List**	**New Installation**	**Reinstall**
		Application Deployment	✓	✓
		JTA	✓	✓
		Enable Production Mode	✓	✓
		Start AdminServer and NodeManager	✓	✓
	OBAPI+OBRH Configuration	Copy config files into OBAPI+OBRH Installation Home	✓	✓ (Delete old and copy new from installer zip)

7

Post Installation Steps

Credential Store Mapping

The OBAPI system utilizes external integrations to facilitate seamless communication with various services. To establish these connections, credentials are required to authenticate and authorize access. These credentials are not hardcoded but rather initialized post-installation. They are subsequently encrypted and stored within the database, ensuring confidentiality and integrity. Upon application startup, the credentials undergo decryption, enabling secure loading into the system. This subsequent section outlines the procedures and guidelines for configuring and managing these credentials within the OBAPI environment.

To configure and add credentials follow the steps mentioned in the below document

- **Oracle Banking APIs Credential Store Setup Guide**

Fileupload with UBS

Refer below document for File upload configuration with UBS

- **Oracle Banking APIs File Upload Report Configuration**

Origination with OBO

Refer below document (section 5 and 6) for enabling Origination with OBO

- **Oracle Banking APIs OBO Mid-Office and Third Party Setup and Configuration Guide**

Trade Finance (LC and BG) with OBTFPM

Refer below document for enabling **Letter Of Credit** issuance and **Bank Guarantee** issuance with Oracle Banking Trade Finance Management.

Oracle Banking Mid-Office Product Setup and Configuration Guide

OHS

OHS server needs to be configured for all FLAVOR's as a mandatory activity.

To configure OHS server follow steps mentioned in below document before proceeding further.

- **Oracle Banking APIs OHS User Interface Configuration** user manual.

Feedback module:

In order to enable Scale (Rating) icons please refer the section **Creating Procedure** of **Oracle Banking APIs Content Upload Guide** user manual.

Table 7-1 WAR deployments

Sr No	Module	Mandatory (Y/N)
Domainwise deployments		
2	digx-common	Y
4	digx-infra	Y
5	digx-coherence	Y

Table 7-1 (Cont.) WAR deployments

Sr No	Module	Mandatory (Y/N)
8	digx-extxfacesimulator	Y
9	digx-cms	N
10	digx-corporateloan	N
11	digx-creditfacility	N
12	digx-edx	N
14	digx-liquiditymanagement	N
16	digx-payments	N
17	digx-pfm	N
19	digx-processmanagement	N
20	digx-retail	N
21	digx-scf	N
22	digx-scfcm	N
23	digx-tradefinance	N
24	digx-virtual-account	N
25	digx-genai	Y
26	digx-ml-Indb	Y
27	digx-sms	Y
28	digx-ukob	Y
29	digx-webauthn	Y
30	weblogic-remote-cconsole-app	Y

Enabling Kafka in OBAPI

Overview

OBAPI now supports Apache Kafka as a messaging system in addition to JMS. Kafka provides high throughput, scalability, and fault tolerance, making it an excellent choice for event-driven architectures. OBAPI will work with either JMS or Kafka but not both simultaneously. This section explains how to enable Kafka.

1. Enable Kafka
   - JMS is the default messaging system in OBAPI.
   - Enable Kafka only if you want to switch the existing message system to Kafka by executing the following configuration queries in the database. If not enabled, the system will continue using JMS.

Configuration Queries:

UPDATE DIGX_FW_CONFIG_ALL_B SET prop_value = 'KAFKA' WHERE prop_id = 'MESSAGE_BROKER_TYPE';

UPDATE DIGX_FW_CONFIG_ALL_B SET prop_value = '{HOSTIP}:{HOSTPORT}' WHERE prop_id = 'bootstrap.servers';

Replace {HOSTIP}:{HOSTPORT} with the IP address and port of the Kafka broker running the Kafka service.

> **Note:**
>
> Ensure that Kafka is properly installed before making this change.

If you want to migrate the existing messaging system to Kafka, you need to implement Kafka consumer and producer equivalents for all the customized JMS queues and topics. Also, If there are any new customized Kafka topics to be created, the corresponding producer and consumer implementations must be developed. Refer to **Section: Messaging System Integration for OBAPI** in the document **Oracle Banking APIs Extensibility Guide** for the detailed steps and guidelines in implementing Kafka and JMS.

- OBAPI Pre-defined External Kafka Topic Configurations

OBAPI listens to the following external Topics of OBVAM(Oracle Banking Virtual Account Management), OBLM(Oracle Banking Liquidity Management) and OBO(Oracle Banking Origination).

- lm.accountnotification
- vam.virtualaccountnotification
- vam.accountstatusnotification
- vac.accountfacilitynotification
- externalSystemAlertMessage

For these Topics, the following properties need to be updated in the table DIGX_FW_CONFIG_ALL_B. Details should be obtained from the respective HOST system. If the HOST does not support the SSL property, it should be removed.

```
PROP_ID
CATEGORY_ID
PROP_VALUE
        <topic_name>@ssl.truststore.password
KAFKA_CONFIG
<truststore_password>
        <topic_name>@sasl.jaas.config
KAFKA_CONFIG org.apache.kafka.common.security.scram.ScramLogin
Module required
        username="<jaas_username>" password="<jaas_password>";
        <topic_name>@bootstrap.servers
KAFKA_CONFIG
<host_ip>:<host_port>
        <topic_name>@ssl.truststore.location
KAFKA_CONFIG
<truststore_location>
        <topic_name>@security.protocol
KAFKA_CONFIG
<security_protocol>
        <topic_name>@sasl.mechanism KAFKA_CONFIG <sasl_mechanism>
```

8
OBAPI Logging Configuration

Logging Level Configuration with SLF4J & Logback in Weblogic

Logging at package and class levels can be externalized/customized by maintaining a common logback file outside the application for all the wars. This file will be configured as a server start argument.

1. Use the attached sample reference file and copy it to any physical path. (For example, /scratch/obapi/domains/obapi_domain/logbackOverride.xml)
   logbackOverride.xml

   ```
   <configuration scan="true"
           scanPeriod="10 minutes">

           <appender name="STDOUT"
           class="ch.qos.logback.core.ConsoleAppender">
   <!-- encoders are assigned the type
   ch.qos.logback.classic.encoder.PatternLayoutEncoder by
           default -->
   <encoder>
   <pattern>%date{dd MMM yyyy;HH:mm:ss.SSS} [%thread] %X{ecid}
           %-5level %logger{100}[%X{FILE_IDENTIFIER} %X{FILE_REF_ID}] -
           %msg%n</pattern>        </encoder>
           </appender>
       <!--   <logger name="com.ofss.digx.app.sms.service.user.login"
           level="info"/>   <logger
   name="com.ofss.digx.app.sms.service.user.User" level="debug"/>
           -->
           <root level="ERROR">
   <appender-ref ref="STDOUT" />
           </root>
    </configuration>
   ```

2. Configure the same above path in server start arguments as follows.
   `-Dlogback.configurationFile=/scratch/obapi/domains/obapi_domain/logbackOverride.xml`

   Enable package and class level logging :

   If you want to change the logging level of a particular class or a package, you can do so by adding the following snippet in the external logback file and taking managed server restart. (Refer to the sample file)

   a. To configure package logging level:
      `<logger name="com.ofss.digx.app.sms.service.user.login" level="info"/>`

   b. To configure class logging level :
      `<logger name="com.ofss.digx.app.sms.service.user.User" level="debug"/>`

Note:

In order to get the changes reflected without server restart, you can add a "scan" attribute to the <configuration> element in the external logback file. By default, the configuration file will be scanned for changes once every minute. To configure your desired scan period, add the attribute "scanPeriod" with value in milliseconds, seconds, minutes, or hours.

For example,

```
<configuration scan="true" scanPeriod="2 minutes">
```

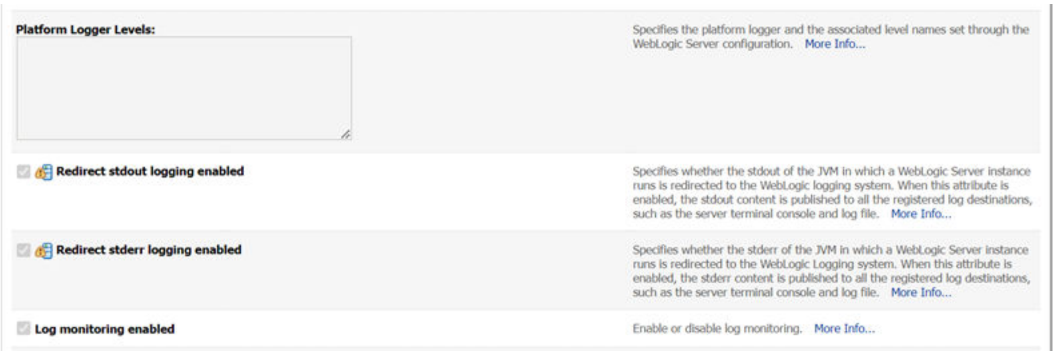
This will scan for the configuration file every 2 minutes for any changes.

Redirecting stdout and stderr logs into a log file :

To redirect standard out and error logs to a log file, please follow the below steps.

Login to Weblogic console → Take Lock & Edit session → Go to Servers inside Environment menu à

Click on the managed server → Go to Logging tab → Advanced → Check the boxes “Redirect stdout logging enabled” and “Redirect stderr logging enabled” as shown below.



Platform Logger Levels:

Specifies the platform logger and the associated level names set through the WebLogic Server configuration. More Info...

☑ **Redirect stdout logging enabled**

Specifies whether the stdout of the JVM in which a WebLogic Server instance runs is redirected to the WebLogic logging system. When this attribute is enabled, the stdout content is published to all the registered log destinations, such as the server terminal console and log file. More Info...

☑ **Redirect stderr logging enabled**

Specifies whether the stderr of the JVM in which a WebLogic Server instance runs is redirected to the WebLogic Logging system. When this attribute is enabled, the stderr content is published to all the registered log destinations, such as the server terminal console and log file. More Info...

☑ **Log monitoring enabled**

Enable or disable log monitoring. More Info...

9 OBAPI Product Verification

Start managed server and verify all deployed applications are in Active state (as shown below).

Weblogic login URL has been changed from “http://<IP Address>/console” to “http://<IP Address>/rconsole”

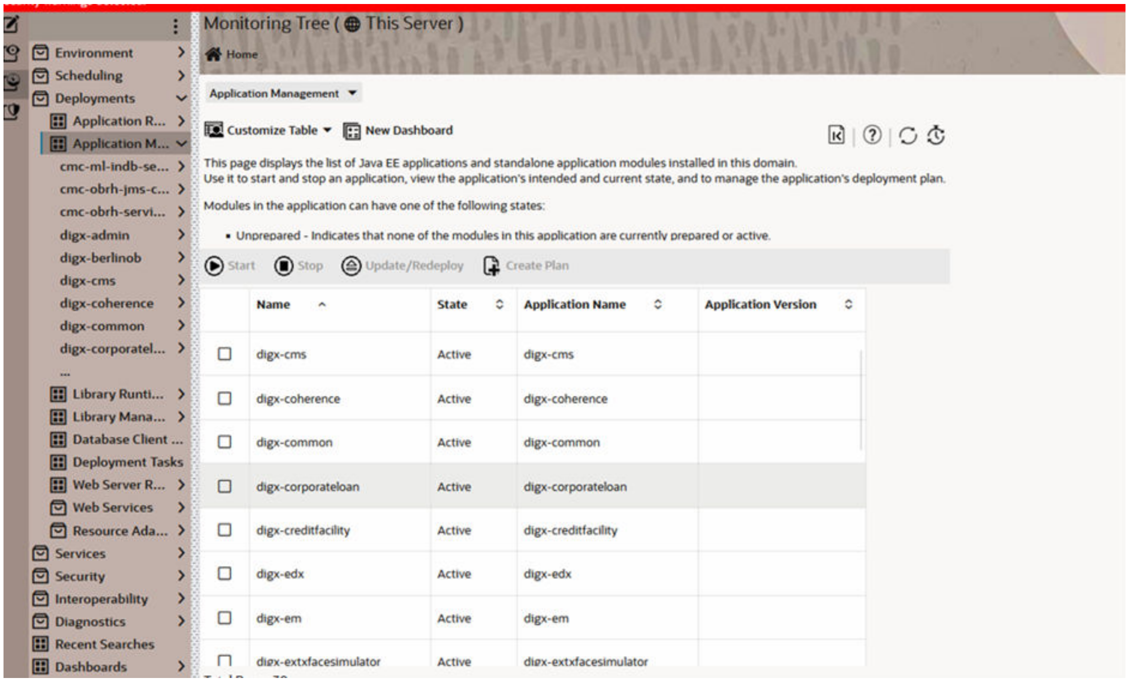
To make these changes,Please follow below Documentation:

`https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-remote-console/administer/set-console.html#GUID-34B825B0-644B[…]3-9F50-7745E20D830B`

Deployment wars status

Start managed server and verify all deployed applications are in Active state (as shown below).

Domainwise deployment wars status

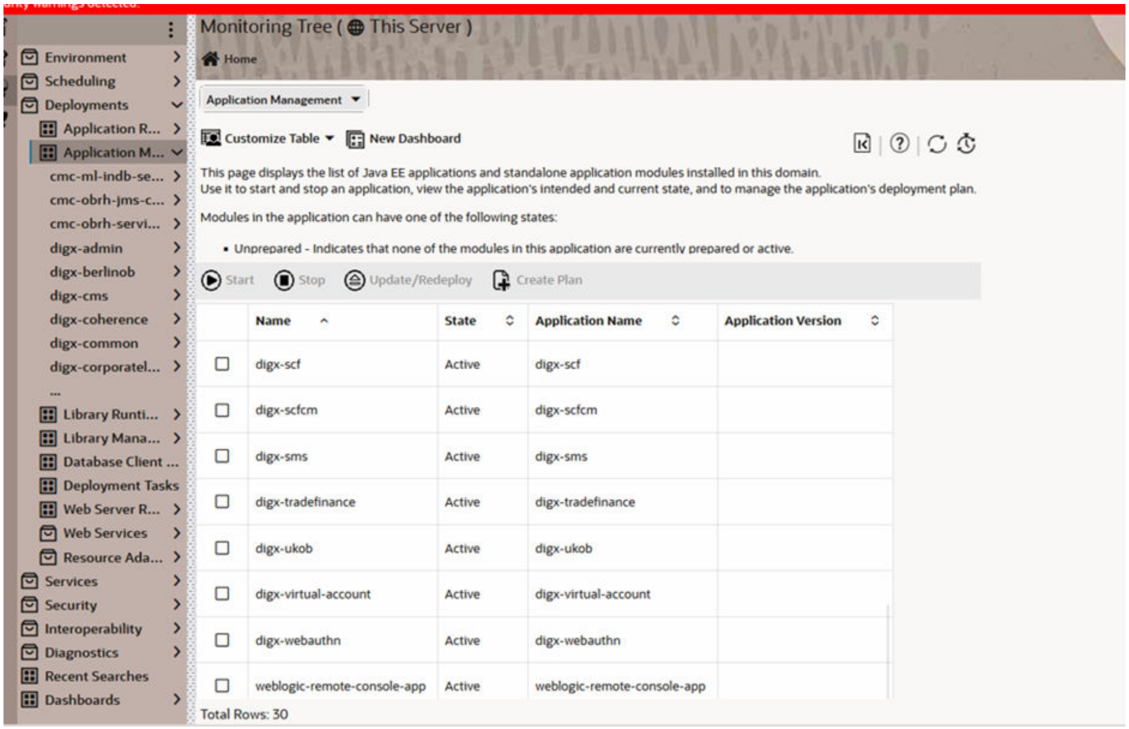


This page displays the list of Java EE applications and standalone application modules installed in this domain.
Use it to start and stop an application, view the application's intended and current state, and to manage the application's deployment plan.

Modules in the application can have one of the following states:

- Unprepared - Indicates that none of the modules in this application are currently prepared or active.

	Name	State	Application Name	Application Version
□	digx-cms	Active	digx-cms	
□	digx-coherence	Active	digx-coherence	
□	digx-common	Active	digx-common	
□	digx-corporateloan	Active	digx-corporateloan	
□	digx-creditfacility	Active	digx-creditfacility	
□	digx-edx	Active	digx-edx	
□	digx-em	Active	digx-em	
□	digx-extxfacesimulator	Active	digx-extxfacesimulator	



Monitoring Tree (This Server)

Home

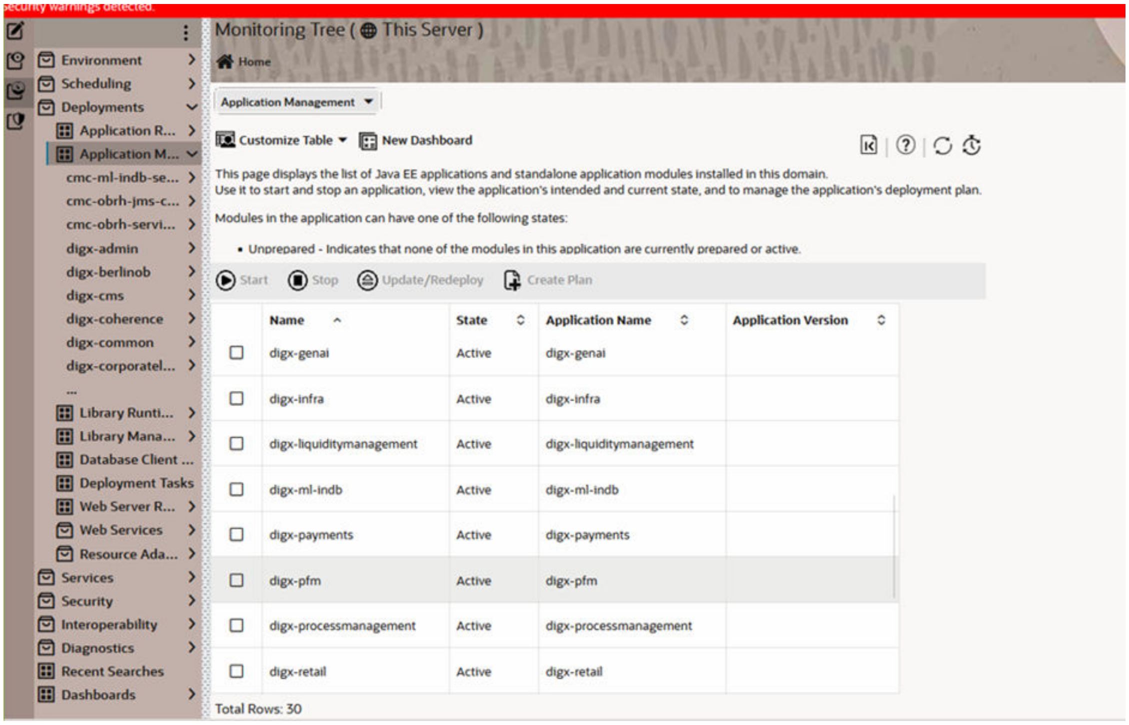
This page displays the list of Java EE applications and standalone application modules installed in this domain.
Use it to start and stop an application, view the application's intended and current state, and to manage the application's deployment plan.

Modules in the application can have one of the following states:

- Unprepared - Indicates that none of the modules in this application are currently prepared or active.

	Name	State	Application Name	Application Version
□	digx-scf	Active	digx-scf	
□	digx-scfcm	Active	digx-scfcm	
□	digx-sms	Active	digx-sms	
□	digx-tradefinance	Active	digx-tradefinance	
□	digx-ukob	Active	digx-ukob	
□	digx-virtual-account	Active	digx-virtual-account	
□	digx-webauthn	Active	digx-webauthn	
□	weblogic-remote-console-app	Active	weblogic-remote-console-app	

Total Rows: 30



Monitoring Tree (This Server)

Home

This page displays the list of Java EE applications and standalone application modules installed in this domain.
Use it to start and stop an application, view the application's intended and current state, and to manage the application's deployment plan.

Modules in the application can have one of the following states:

- Unprepared - Indicates that none of the modules in this application are currently prepared or active.

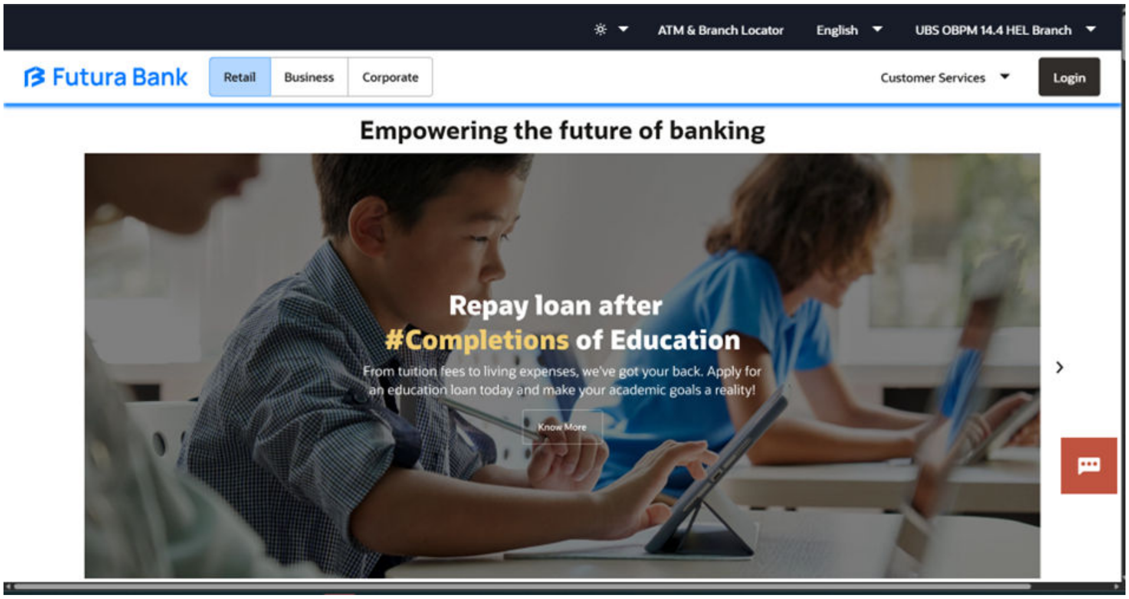
	Name	State	Application Name	Application Version
□	digx-genai	Active	digx-genai	
□	digx-infra	Active	digx-infra	
□	digx-liquiditymanagement	Active	digx-liquiditymanagement	
□	digx-ml-indb	Active	digx-ml-indb	
□	digx-payments	Active	digx-payments	
□	digx-pfm	Active	digx-pfm	
□	digx-processmanagement	Active	digx-processmanagement	
□	digx-retail	Active	digx-retail	

Total Rows: 30

To verify the installation, launch below URL

http://<OHS server ip or hostname>:<OHS port>

Check if the page loads successfully.


ATM & Branch Locator
English
UBS OBPM 14.4 HEL Branch
Futura Bank
Retail
Business
Corporate
Customer Services
Login
Empowering the future of banking
Repay loan after
#Completions of Education
From tuition fees to living expenses, we've got your back. Apply for an education loan today and make your academic goals a reality!
Know More

Day1 Configuration

Universal Banking Solution (OBAPI with UBS)

Refer below document (Section 3. System Configuration) for Day1 configuration required for integration with UBS

Oracle Banking APIs System Configuration

Once day1 is completed, application is available for end-user transactions.

> **Note:**
>
> Post Day1 restart of Managed server is mandatory

Third Party System (OBAPI with THP)

Refer below document (**Section 5. System Configuration – Host System as Third Party**) for Day1 configuration required for integration with Third-party System

Oracle Banking APIs System Configuration

Once day1 is completed, application is available for end-user transactions.

> **Note:**
>
> Post Day1 restart of Managed server is mandatory

Chat Bot Configuration:

Refer below document for Chat Bot configuration.

Oracle Banking APIs Chatbot Configuration

Mobile Application Builder:

Refer below documents for Mobile Applications build and setup.

Oracle Banking Digital Experience Mobile Application Builder-Android

Oracle Banking Digital Experience Mobile Application Builder-iOS

Mid Office Configuration:

Refer below document for Mid Office Configurations i.e. Trade Finance, Corporate Lending.

Oracle Banking Mid-Office Product Setup and Configuration Guide.

Account Uniqueness Configuration:

Some core banking systems support same account number in multiple branches within the entity. OBAPI has support for such core banking systems. However, the configuration is not enabled by default. In case the Bank has core banking system which supports and provides same account numbers across multiple branches, the following scripts should be executed per entity for enabling the support.

```
Insert into  DIGX_FW_CONFIG_ALL_O (PROP_ID, PREFERENCE_NAME, PROP_VALUE,
DETERMINANT_VALUE,
CREATED_BY,CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE)
values
('OBAPI.host.account.uniqueness','ExtSystemsConfig','BRANCH','<ENTITY_ID>','of
ssuser',sysdate,
'ofssuser',sysdate);
Insert into DIGX_FW_CONFIG_ALL_O (PROP_ID, PREFERENCE_NAME, PROP_VALUE,
DETERMINANT_VALUE,
CREATED_BY, CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE)
values
('OBAPI.host.accountbranch.delimiter','ExtSystemsConfig','@~','<ENTITY_ID>','o
fssuser',sysdate,'ofssuser',sysdate);
```

Note:

Ensure that <ENTITY_ID> has been replaced with correct Entity ID for the corresponding entity.

10 Multi Entity

To add entity to existing OBAPI with supported host system follow below steps.

- Add entity through OBAPI Web application, using
  - User Manual Oracle Banking APIs System Configuration
- In case of OBTFPM integration, following document should be referred.
  - Oracle Banking Mid-Office Product Setup and Configuration Guide Running OBAPI installer

Ensure that Managed server should be down and Admin server should be running state.

Ensure `ORACLE_HOME, JAVA_HOME` variable are set and their binaries are available in PATH variable before proceeding.

Login with OS user which was used to perform OBAPI software installation (or has ownership on Oracle Weblogic home directory)

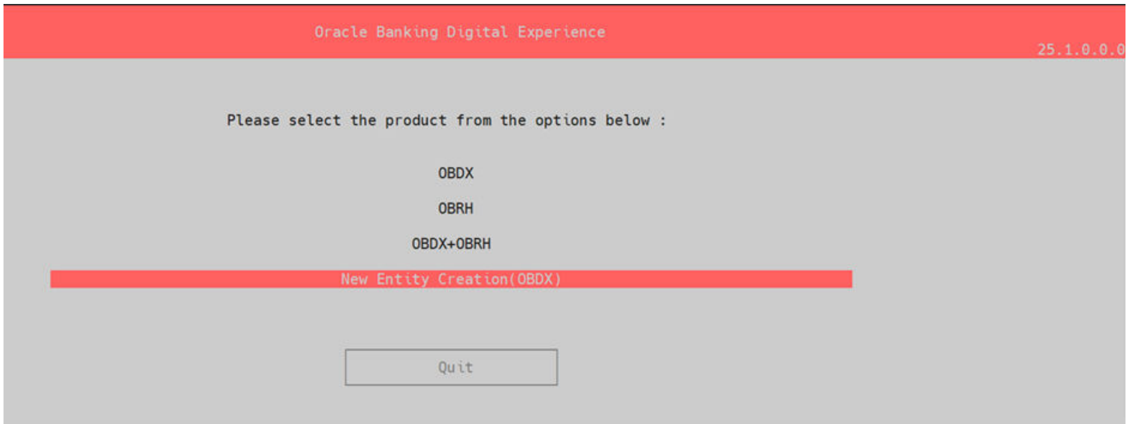
Ensure OBAPI installation details (OBAPI DB; WLS etc) are maintained in installer.properties and user running the installer has read-write permissions.

From your terminal navigate to <OBAPI INSTALLER DIR>

Enter the following command

```
python3 runInstaller.py
```

Select installation type as ‘New Entity Creation’.

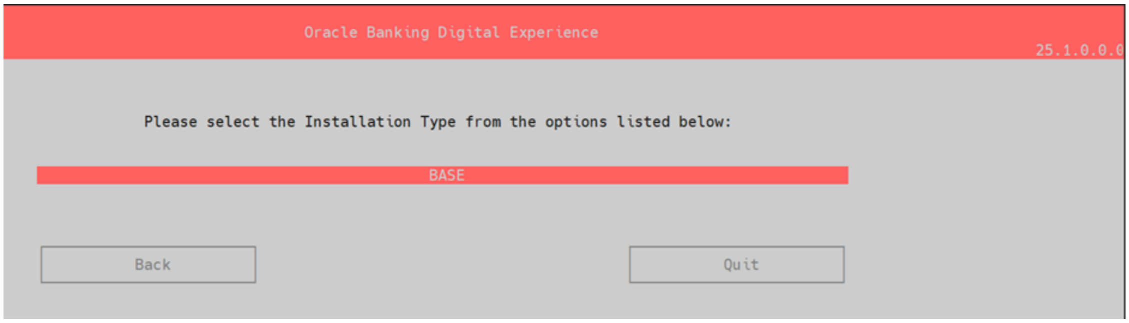


Oracle Banking Digital Experience

25.1.0.0.0

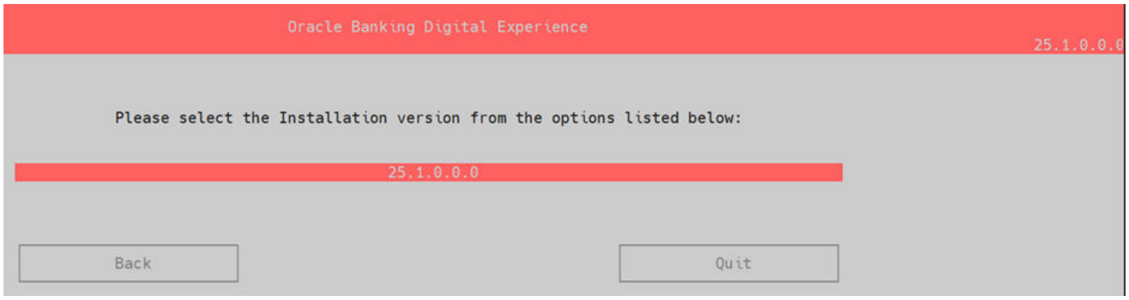
Please select the product from the options below :

Below screen will appear after selecting add entity.



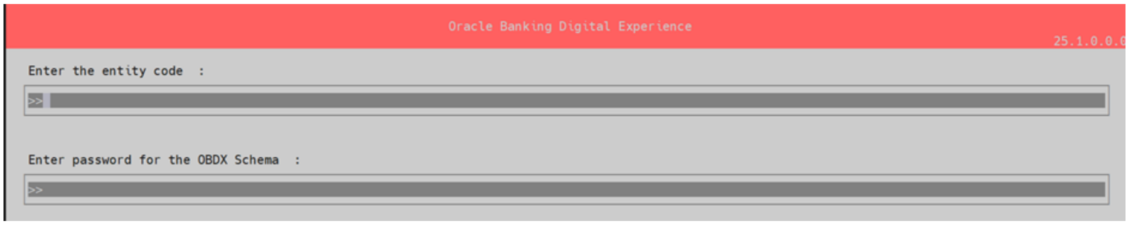
Please select the Installation Type from the options listed below:

BASE



Please select the Installation version from the options listed below:

25.1.0.0.0



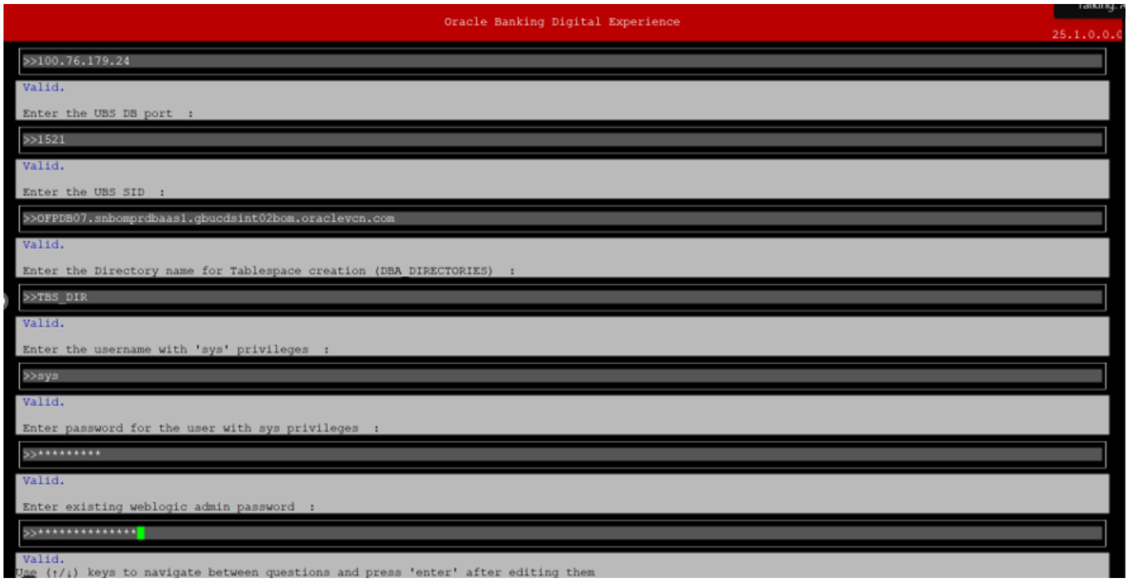
Enter the entity code :

Enter password for the OBDX Schema :

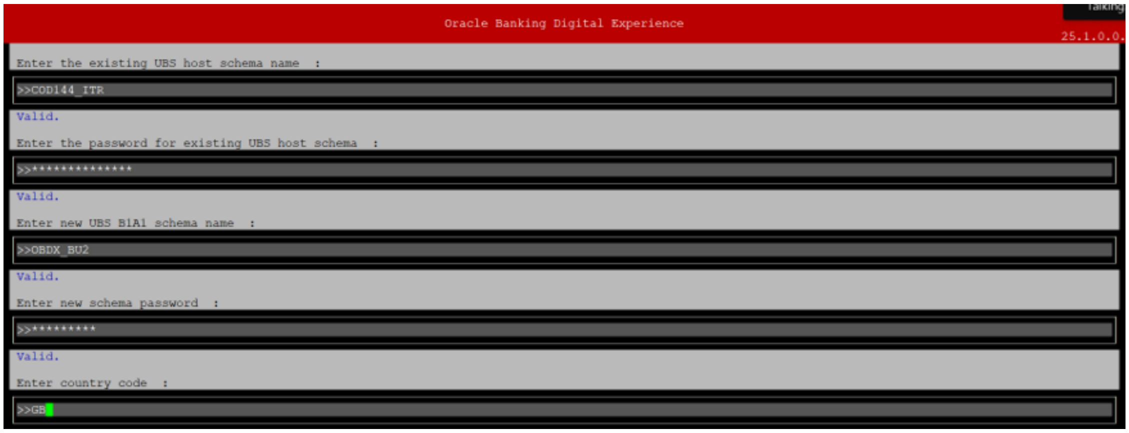
Enter below information:

- Entity code which has been added from screen
- OBAPI schema password

If an entity code belongs to UBS / OBPM host following screen (below screenshot are for OBPM ; for UBS same input are required) will appear:



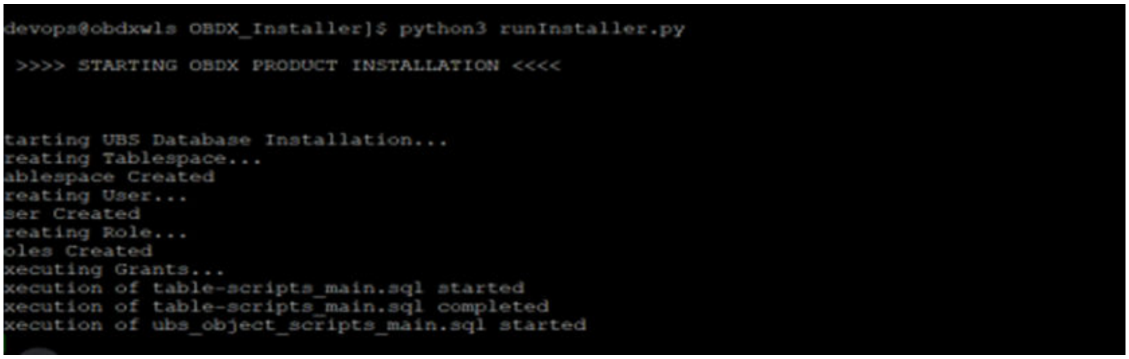
```
>>100.76.179.24
Valid.
Enter the UBS DB port  :
>>1521
Valid.
Enter the UBS SID  :
>>OFPDB07.snbomprdbaas1.gbucdsint02bom.oraclevcn.com
Valid.
Enter the Directory name for Tablespace creation (DBA_DIRECTORIES)  :
>>TBS_DIR
Valid.
Enter the username with 'sys' privileges  :
>>sys
Valid.
Enter password for the user with sys privileges  :
>>*********
Valid.
Enter existing weblogic admin password  :
>>**************
Valid.
Use (↑/↓) keys to navigate between questions and press 'enter' after editing them
```



```
Enter the existing UBS host schema name  :
>>COD144_ITR
Valid.
Enter the password for existing UBS host schema  :
>>**************
Valid.
Enter new UBS B1A1 schema name  :
>>OBDX_BU2
Valid.
Enter new schema password  :
>>*********
Valid.
Enter country code  :
>>GB
```

Enter below details:

- Hostname of the database host server
- Port of the database host server
- Host database Service Name
- Oracle directory name in which you want the database datafile (dbf) to be created. Enter only the name NOT the path.
- Username with 'sys' privileges
- SYS privilege user password where UBS schema would be created
- Weblogic console administrator user password



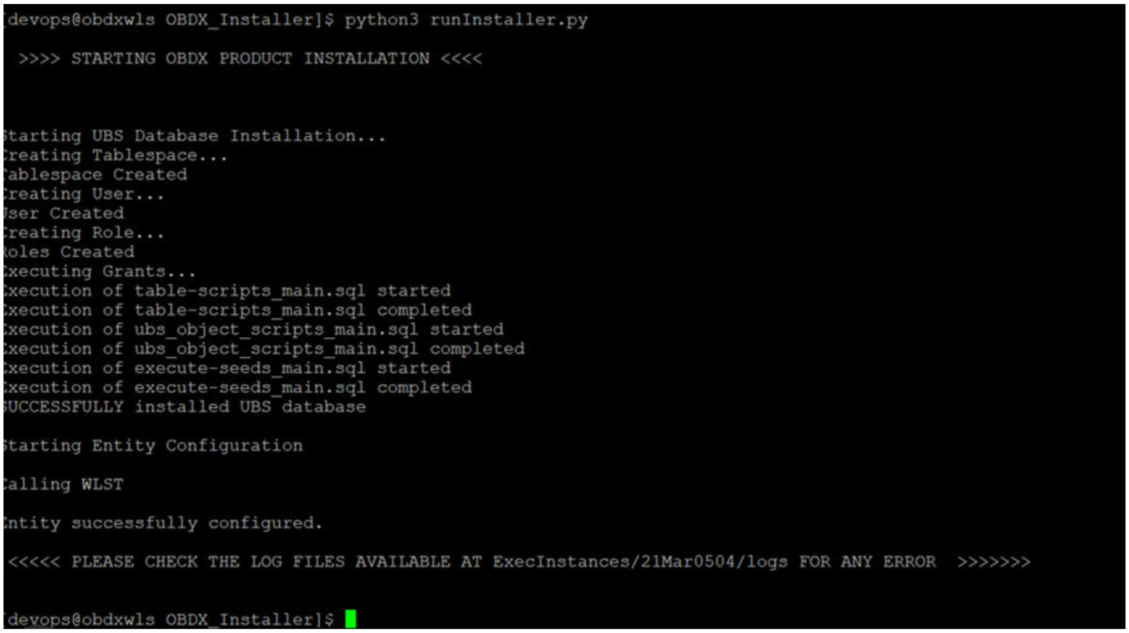
```
devops@obdxwls OBDX_Installer]$ python3 runInstaller.py

 >>>> STARTING OBDX PRODUCT INSTALLATION <<<<


tarting UBS Database Installation...
reating Tablespace...
ablespace Created
reating User...
ser Created
reating Role...
oles Created
xecuting Grants...
xecution of table-scripts_main.sql started
xecution of table-scripts_main.sql completed
xecution of ubs_object_scripts_main.sql started
```

Enter below details:

- EXISTING Host schema name
- Password for EXISTING schema
- Complete EHMS (HostInterface) schema name you want installer to create as new schema
- Password for New schema
- Country Code of entity branch

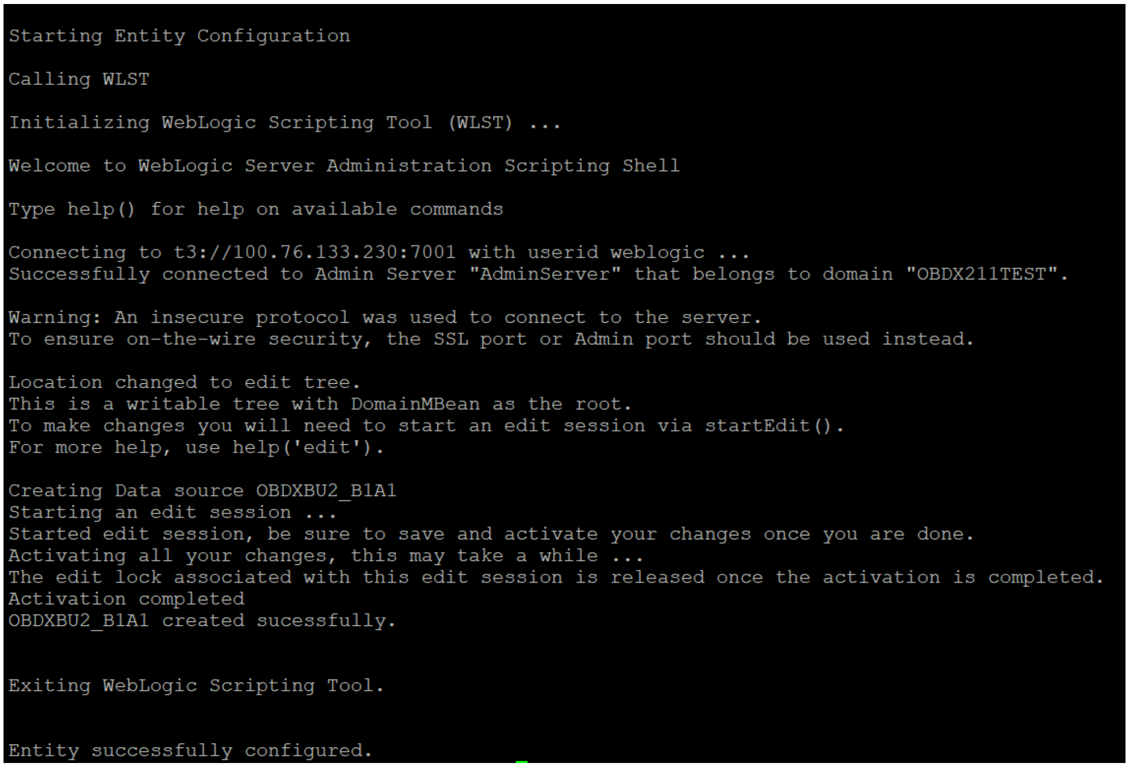
Installation Status in case of UBS / OBPM

After entering all required details, the status is displayed (as shown below) on the terminal to indicate the progress of the installation.



```
[devops@obdxwls OBDX_Installer]$ python3 runInstaller.py

 >>>> STARTING OBDX PRODUCT INSTALLATION <<<<


Starting UBS Database Installation...
Creating Tablespace...
Tablespace Created
Creating User...
User Created
Creating Role...
Roles Created
Executing Grants...
Execution of table-scripts_main.sql started
Execution of table-scripts_main.sql completed
Execution of ubs_object_scripts_main.sql started
Execution of ubs_object_scripts_main.sql completed
Execution of execute-seeds_main.sql started
Execution of execute-seeds_main.sql completed
SUCCESSFULLY installed UBS database

Starting Entity Configuration

Calling WLST

Entity successfully configured.

 <<<<< PLEASE CHECK THE LOG FILES AVAILABLE AT ExecInstances/21Mar0504/logs FOR ANY ERROR  >>>>>>>


[devops@obdxwls OBDX_Installer]$
```

When the installation completes, the below message is displayed



```
Starting Entity Configuration

Calling WLST

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to WebLogic Server Administration Scripting Shell

Type help() for help on available commands

Connecting to t3://100.76.133.230:7001 with userid weblogic ...
Successfully connected to Admin Server "AdminServer" that belongs to domain "OBDX211TEST".

Warning: An insecure protocol was used to connect to the server.
To ensure on-the-wire security, the SSL port or Admin port should be used instead.

Location changed to edit tree.
This is a writable tree with DomainMBean as the root.
To make changes you will need to start an edit session via startEdit().
For more help, use help('edit').

Creating Data source OBDXBU2_B1A1
Starting an edit session ...
Started edit session, be sure to save and activate your changes once you are done.
Activating all your changes, this may take a while ...
The edit lock associated with this edit session is released once the activation is completed.
Activation completed
OBDXBU2_B1A1 created sucessfully.


Exiting WebLogic Scripting Tool.


Entity successfully configured.
```

Post successful installation refer to **Section 8: Post Installation steps** for manual steps to be performed for UBS additional entity (**sub-section : Oracle FLEXCUBE Universal Banking (OBAPI with UBS)**).

No additional steps/ configuration are required.

11

Multi-entity installation using Silent Mode

This chapter describes how to run the OBAPI installer for add entity in silent mode.

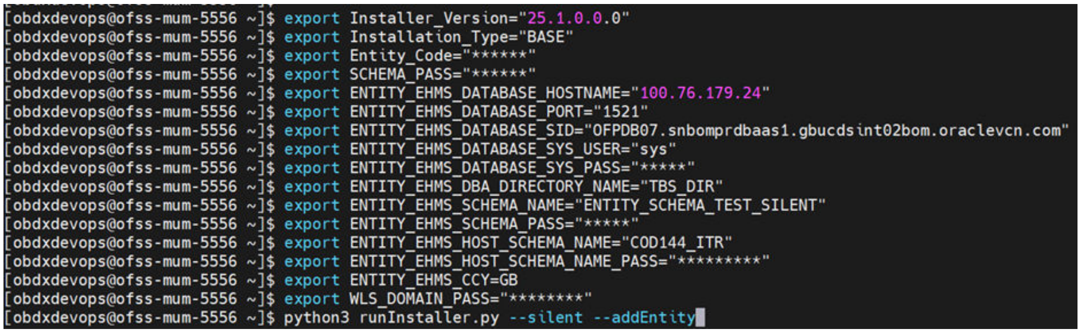
Ensure that Managed server should be down and Admin server should be running.

Ensure ORACLE_HOME, JAVA_HOME variable are set and their binaries are available in PATH variable before proceeding.

Login with OS user which was used to perform OBAPI software installation (or has ownership on Oracle Weblogic home directory)

Steps for Silent-Mode Installation

1. Set the environment variables, as shown below.



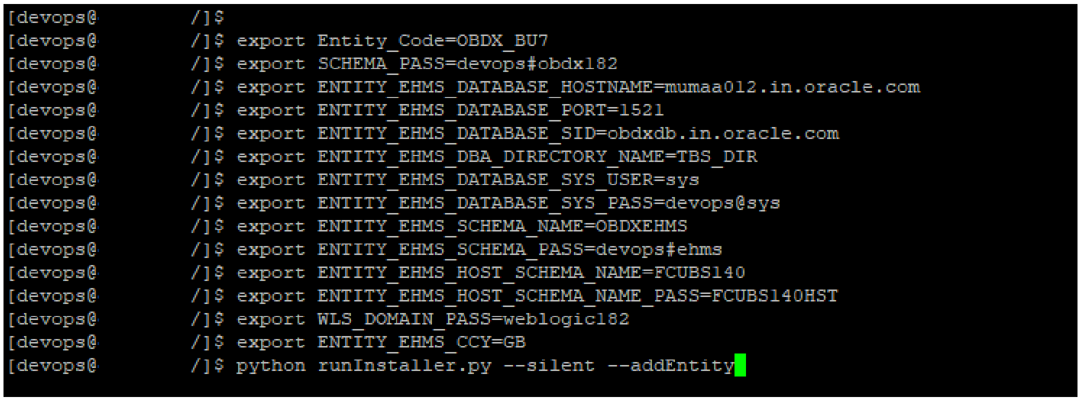
```
[obdxdevops@ofss-mum-5556 ~]$ export Installer_Version="25.1.0.0.0"
[obdxdevops@ofss-mum-5556 ~]$ export Installation_Type="BASE"
[obdxdevops@ofss-mum-5556 ~]$ export Entity_Code="******"
[obdxdevops@ofss-mum-5556 ~]$ export SCHEMA_PASS="******"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_DATABASE_HOSTNAME="100.76.179.24"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_DATABASE_PORT="1521"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_DATABASE_SID="OFPDB07.snbomprdbaas1.gbucdsint02bom.oraclevcn.com"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_DATABASE_SYS_USER="sys"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_DATABASE_SYS_PASS="*****"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_DBA_DIRECTORY_NAME="TBS_DIR"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_SCHEMA_NAME="ENTITY_SCHEMA_TEST_SILENT"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_SCHEMA_PASS="*****"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_HOST_SCHEMA_NAME="COD144_ITR"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_HOST_SCHEMA_NAME_PASS="*********"
[obdxdevops@ofss-mum-5556 ~]$ export ENTITY_EHMS_CCY=GB
[obdxdevops@ofss-mum-5556 ~]$ export WLS_DOMAIN_PASS="********"
[obdxdevops@ofss-mum-5556 ~]$ python3 runInstaller.py --silent --addEntity
```

Below parameters should be set in environment variables

Environment variables	Parameter	Description	Example
Environment variables to set for flavor: **FCORE UBS (14.6.0.0.0 release)** **OBPM (14.6.0.0.0 release)**	`Entity_Code`	Entity code which has been entered from screen	`export Entity_Code=OBDX_BU7`
	`SCHEMA_PASS`	Password for existing OBAPI schema	`export SCHEMA_PASS=devops#obapi182`
	`ENTITY_EHMS_DATABASE_HOSTNAME`	Hostname of the EHMS HOST database host server	`export ENTITY_EHMS_DATABASE_HOSTNAME=xx.xx.xx.xx`
	`ENTITY_EHMS_DATABASE_PORT`	Port of the EHMS HOST database host server	`export ENTITY_EHMS_DATABASE_PORT=1521`
	`ENTITY_EHMS_DATABASE_SID`	EHMS Host database Service Name	export ENTITY_EHMS_DATABASE_SID=obdxdb.in.oracle.com

Environment variables	Parameter	Description	Example
	`ENTITY_EHMS_DBA_DIRECTORY_NAME`	Oracle Directory name in which you want the EHMS (HostInterface) schema datafile (dbf). Enter only the name and NOT the path	`export ENTITY_EHMS_DBA_DIRECTORY_NAME=TBS_DIR`
	`ENTITY_EHMS_DATABASE_SYS_USER`	Username with 'sys' privileges	`export ENTITY_EHMS_DATABASE_SYS_USER=sys`
	`ENTITY_EHMS_DATABASE_SYS_PASS`	Password for EHMS sys user	`export ENTITY_EHMS_DATABASE_SYS_PASS=devops@sys`
	`ENTITY_EHMS_SCHEMA_NAME`	Complete EHMS (HostInterface) schema name you want installer to create as new schema.	`export ENTITY_EHMS_SCHEMA_NAME=OBAPIEHMS`
	`ENTITY_EHMS_SCHEMA_PASS`	Password for new EHMS schema on EHMS HOST database	`export ENTITY_EHMS_SCHEMA_PASS=devops#ehms`
	`ENTITY_EHMS_HOST_SCHEMA_NAME`	EXISTING EHMS Host schema name	`export ENTITY_EHMS_HOST_SCHEMA_NAME=EHMSHOST`
	`ENTITY_EHMS_HOST_SCHEMA_NAME_PASS` **This parameter is only required for UBS & OBPM Host	Password of existing HOST EHMS schema (Existing)	`export ENTITY_EHMS_HOST_SCHEMA_NAME_PASS=ehmshst`
	`WLS_DOMAIN_PASS`	Password for Weblogic admin console	`export WLS_DOMAIN_PASS=weblogic182`
	`ENTITY_EHMS_CCY` **This parameter is only required for UBS & OBPM Host	Country Code for new or additional entity home branch	`export ENTITY_EHMS_CCY=GB`
	`ENTITY_EHMS_FCORE_FCUBS_SCHEMA_NAME` **This parameter is only required for FCORE	FCORE-FCUBS HOST schema name	`export ENTITY_EHMS_FCORE_FCUBS_SCHEMA_NAME=FCRUBSHOST`
Environment variables to set for flavor: **OBAPI (Third-party HOST)**	`Entity_Code`	Entity code which has been entered from screen	`export Entity_Code=OBDX_BU1`
	`SCHEMA_PASS`	Password for existing OBAPI schema	`export SCHEMA_PASS=welcome1`

2. Run the **runInstaller.py** file with '--silent ' argument along with '--addEntity'.

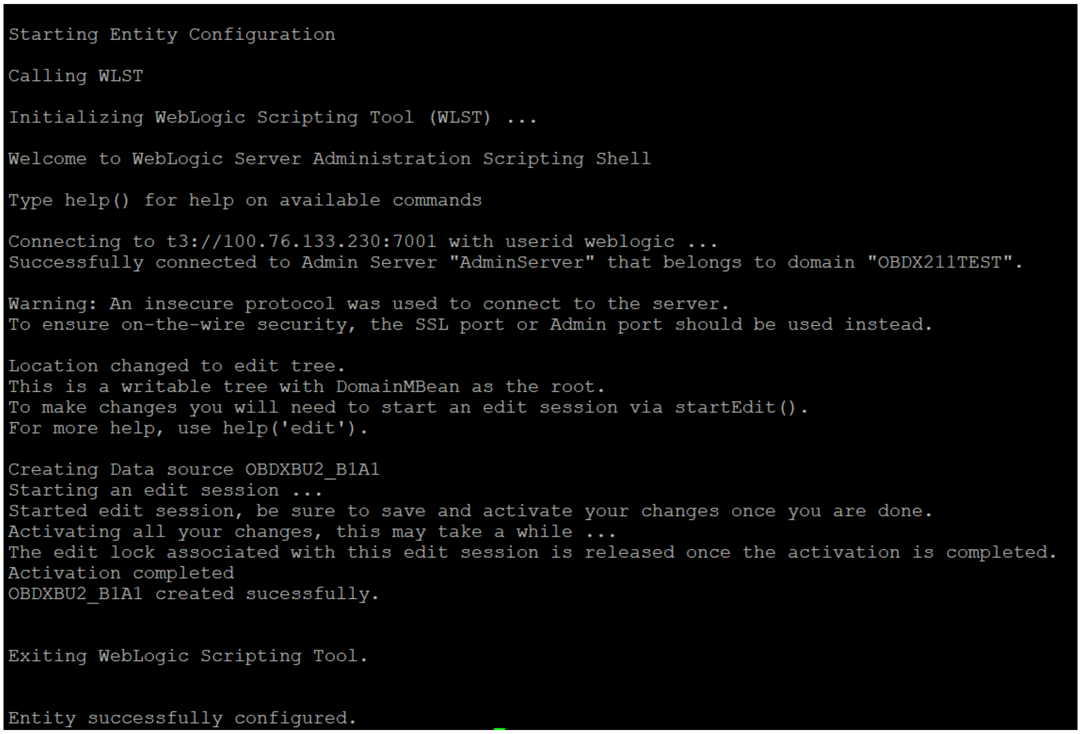


```
[devops@        /]$
[devops@        /]$ export Entity_Code=OBDX_BU7
[devops@        /]$ export SCHEMA_PASS=devops#obdx182
[devops@        /]$ export ENTITY_EHMS_DATABASE_HOSTNAME=mumaa012.in.oracle.com
[devops@        /]$ export ENTITY_EHMS_DATABASE_PORT=1521
[devops@        /]$ export ENTITY_EHMS_DATABASE_SID=obdxdb.in.oracle.com
[devops@        /]$ export ENTITY_EHMS_DBA_DIRECTORY_NAME=TBS_DIR
[devops@        /]$ export ENTITY_EHMS_DATABASE_SYS_USER=sys
[devops@        /]$ export ENTITY_EHMS_DATABASE_SYS_PASS=devops@sys
[devops@        /]$ export ENTITY_EHMS_SCHEMA_NAME=OBDXEHMS
[devops@        /]$ export ENTITY_EHMS_SCHEMA_PASS=devops#ehms
[devops@        /]$ export ENTITY_EHMS_HOST_SCHEMA_NAME=FCUBS140
[devops@        /]$ export ENTITY_EHMS_HOST_SCHEMA_NAME_PASS=FCUBS140HST
[devops@        /]$ export WLS_DOMAIN_PASS=weblogic182
[devops@        /]$ export ENTITY_EHMS_CCY=GB
[devops@        /]$ python runInstaller.py --silent --addEntity
```

Installation Status in case of Oracle FLEXCUBE Core Banking, Oracle FLEXCUBE Universal Banking, Oracle FLEXCUBE Universal Banking with Oracle Banking Payments

After entering all required details, the status is displayed (as shown below) on the terminal to indicate the progress of the installation.

When the installation completes, the below message is displayed



```
Starting Entity Configuration

Calling WLST

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to WebLogic Server Administration Scripting Shell

Type help() for help on available commands

Connecting to t3://100.76.133.230:7001 with userid weblogic ...
Successfully connected to Admin Server "AdminServer" that belongs to domain "OBDX211TEST".

Warning: An insecure protocol was used to connect to the server.
To ensure on-the-wire security, the SSL port or Admin port should be used instead.

Location changed to edit tree.
This is a writable tree with DomainMBean as the root.
To make changes you will need to start an edit session via startEdit().
For more help, use help('edit').

Creating Data source OBDXBU2_B1A1
Starting an edit session ...
Started edit session, be sure to save and activate your changes once you are done.
Activating all your changes, this may take a while ...
The edit lock associated with this edit session is released once the activation is completed.
Activation completed
OBDXBU2_B1A1 created sucessfully.


Exiting WebLogic Scripting Tool.


Entity successfully configured.
```

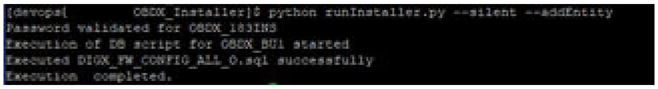
Post successful installation refer to section **Post Installation steps** for manual steps to be performed for

- UBS additional entity (sub-section : Oracle FLEXCUBE Universal Banking Solution (OBAPI with UBS))
- OBPM additional entity (sub-section: Oracle FLEXCUBE Universal Banking with Oracle Banking Payments (OBAPI with OBPM))

3. **Installation Status in case of other hosts as Add Entity**

After entering all required details, the status is displayed (as shown below) on the terminal to indicate the progress of the installation.

- THP(third party as entity)



```
[devops@         OBDX_Installer]$ python runInstaller.py --silent --addEntity
Password validated for OBDX_183INS
Execution of DB script for OBDX_BU1 started
Executed DIGX_FW_CONFIG_ALL_O.sql successfully
Execution  completed.
```

12 Steps to Create Credential Mapping

Credential Store Mapping

The OBDX system utilizes external integrations to facilitate seamless communication with various services. To establish these connections, credentials are required to authenticate and authorize access. These credentials are not hardcoded but rather initialized post-installation. They are subsequently encrypted and stored within the database, ensuring confidentiality and integrity. This subsequent section outlines the procedures and guidelines for configuring and managing these credentials within the OBDX environment.

To utilize the credential mapping functionality, retrieve the **com.ofss.digx.CredentialsStore.jar** file from the designated location:

```
OBDX_Installer/installables/OBDX/BASE/25.1.0.0.0/utils/tools
```

Running the Credential Mapping Application

Execute the application using the following command:

java -jar com.ofss.digx.CredentialsStore.jar <csv_file> <DataBaseCredentials> <DataSeedFlag>

Command Parameters:

1. <csv_file>
   Provide the path to your CSV file containing user credentials by replacing <csv_file> with the actual file location.

   CSV File Format Requirements

   The CSV file must adhere to the following structure:

   - Contain exactly three columns: type, username, and password
   - Include a header row with column names: type,username,password
   - Subsequent rows should contain individual credential entries, with each row representing a distinct set of credentials
   - Ensure that the value in the type column is unique for each credential entry

   Example CSV File

   Table 12-1 WAR deployments

type	username	password
MERCHANT	OBDX	PASSWORD111

2. <DataBaseCredentials>
   Specify the <DataBaseCredentials> parameter as a comma-delimited string comprising the following components:

   - Database username
   - Password

- JDBC URL (in the format jdbc:oracle:thin:@host:port/service_id)

  The expected format for <DataBaseCredentials> is: username,password,jdbc_url.

  Example: User,Password123,jdbc:oracle:thin:@host:port/service_id

  Ensure accurate input of these values to establish a successful connection to the database.

3. <DataSeedFlag>
   To control the seeding of data into the digx_fw_credentials table, set the <DataSeedFlag> parameter to 'Y' to populate the table with the generated credentials. Alternatively, specify 'N' to simply display the credentials without persisting them to the database.

   Example command to run this

   java -jar com.ofss.digx.CredentialsStore.jar data.csv
   DB_USER,DB_PASSWORD,jdbc:oracle:thin:@//HOST:PORT/SERVICE_ID Y

   Upon executing this utility, you will obtain an encrypted password, which can then be utilized in conjunction with other credentials. Subsequently, these credentials will be populated into the database.

Extensibility:

To leverage custom credentials inserted into the system, utilize the following code snippet:

```
ICredentialStore
                store =
CredentialStoreFactory.getCredentials(CredentialStoreKeys.CREDENTIAL_IPMLEMENT
ATION);
Credential credentials = store.getCredentials(<custom_type>);
```

Replace <custom_type> with the desired type associated with the custom credentials.

Import:

Import the jar implementation

```
com.ofss.digx.infra:com.ofss.digx.infra.crypto.impl:$libs_digxVersion
```

into your gradle project

To ensure proper configuration, verify that the entry in the digx_fw_config_all_b table has a prop_id of **credential_impl**, a category_id of **CredentialStore**, and a `PROP_VALUE` of **com.ofss.digx.infra.cred.DatabaseCredentialsStore**. Confirm that these values match exactly to guarantee correct functionality. If discrepancies are found, update the entry accordingly to reflect the specified values.

The AES key is no longer required to be explicitly inserted, as it is dynamically generated by the system when the utility is run and stored within the keystore located at `DIGX_FW_KEYSTORE`.

For any encryption operations that require the use of the AES key, utilize the SymmetricCryptographyProviderFactory class, which is available in the same JAR, instead of relying on the credential. This approach streamlines the encryption process and enhances overall security.

```
SymmetricCryptographyProviderFactory.getInstance().getLatestProvider().encryp
t(data);
```

```
SymmetricCryptographyProviderFactory.getInstance().getLatestProvider().decryp
t(data);
```

13 OBAPI Product Security

Refer below document for OBAPI product security configuration

Oracle Banking APIs Security Guide

14 OBAPI Product – Best Practice

14.1 Tablespace for AUDIT INDEX

The index's used by AUDIT table should be moved into new tablespace from current AUDIT tablespace.

Follow below steps

- Create a new tablespace
- Give quota to OBAPI schema

```
alter user <OBAPI_SCHEMA> quota unlimited on
<OBAPI_AUDIT_INDEX_TABLESPACE>;
```

- Drop and create below index by mapping the newly created tablespace
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_API_AUDIT_LOGGING.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_API_AUDIT_LOG_HIST.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_AUDIT_LOGGING.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_AUDIT_LOGGING_1.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_AUDIT_LOGGING_2.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_AUDIT_LOGGING_3.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_AUDIT_LOGGING_4.sql`
  - `OBAPI_Installer\installables\db\OBAPI\ddl\oracle\audit\IDX_DIGX_AL_AUDIT_LOGGING_DETAILS.sql`

15 JPA and OBAPI multi-cluster

In a multi-cluster environment, below JPA related changes should be implemented

- Go to Weblogic server
- Open config\META-INF\persistence.xml
- Append below configuration for all data-source <property name="eclipselink.cache.coordination.jms.host" value="t3://<WEBLOGIC-HOST-NAME OR IP>:<MANAGED-SERVER-PORT>/" />

  Replace with respective hostname or IP and Port no (this should be the managed server port number which hosts the JPA queues in the cluster)

Key pointers;

- Multi-cluster here refer’s to :
  - Single cluster with multiple nodes (2 or more physical servers hosting the OBAPI product)
  - 2 or more Weblogic cluster’s
- Ensure these (persistence.xml) changes are available to all Managed server by maintaining appropriate classpath

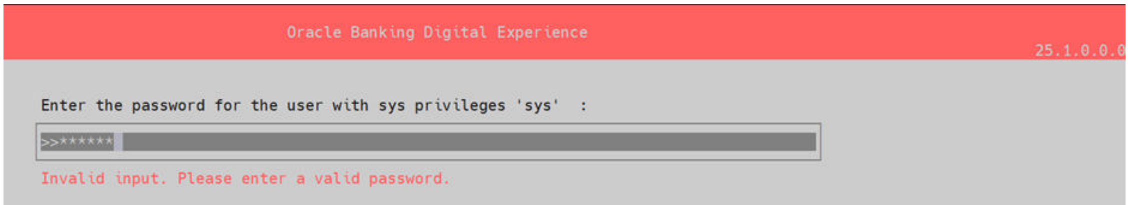
16 Troubleshoot Overview

This section describes how to troubleshoot OBAPI setup.

Invalid database password

This topic contains troubleshooting information if you receive an error when attempting to connect to the database server.

If you get the following error:



```
Enter the password for the user with sys privileges 'sys'  :
>>******
Invalid input. Please enter a valid password.
```

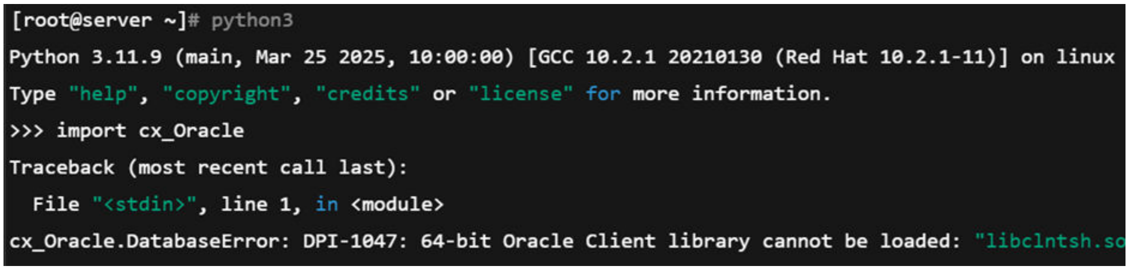
Try one of the following:

- Verify that the database is running.
- Check Network connectivity between Weblogic Server and Database server.
- Check the database configuration in installer.properties file
- Verify that the entered password is correct.

cx_oracle module

This topic contains troubleshooting information about problems with cx_Oracle python module.

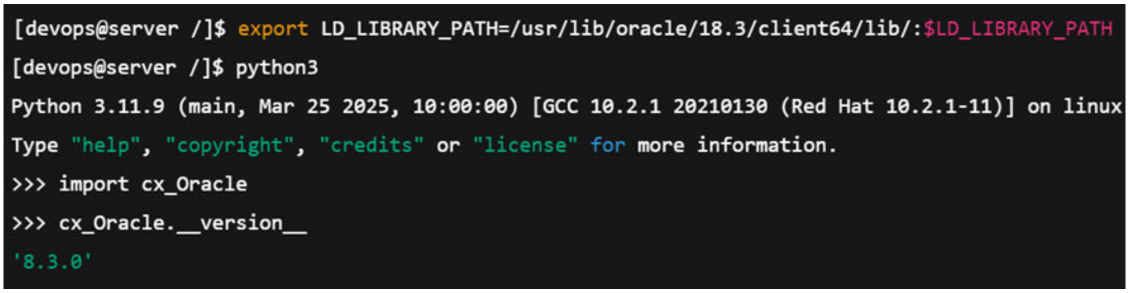
If you get the following error:



```
[root@server ~]# python3
Python 3.11.9 (main, Mar 25 2025, 10:00:00) [GCC 10.2.1 20210130 (Red Hat 10.2.1-11)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cx_Oracle
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
cx_Oracle.DatabaseError: DPI-1047: 64-bit Oracle Client library cannot be loaded: "libclntsh.so
```

Execute the below command:

```
export LD_LIBRARY_PATH=/usr/lib/oracle/19.10/client64/lib:$LD_LIBRARY_PATH
python
    import cx_Orace
    cx_Oracle.__version__
```

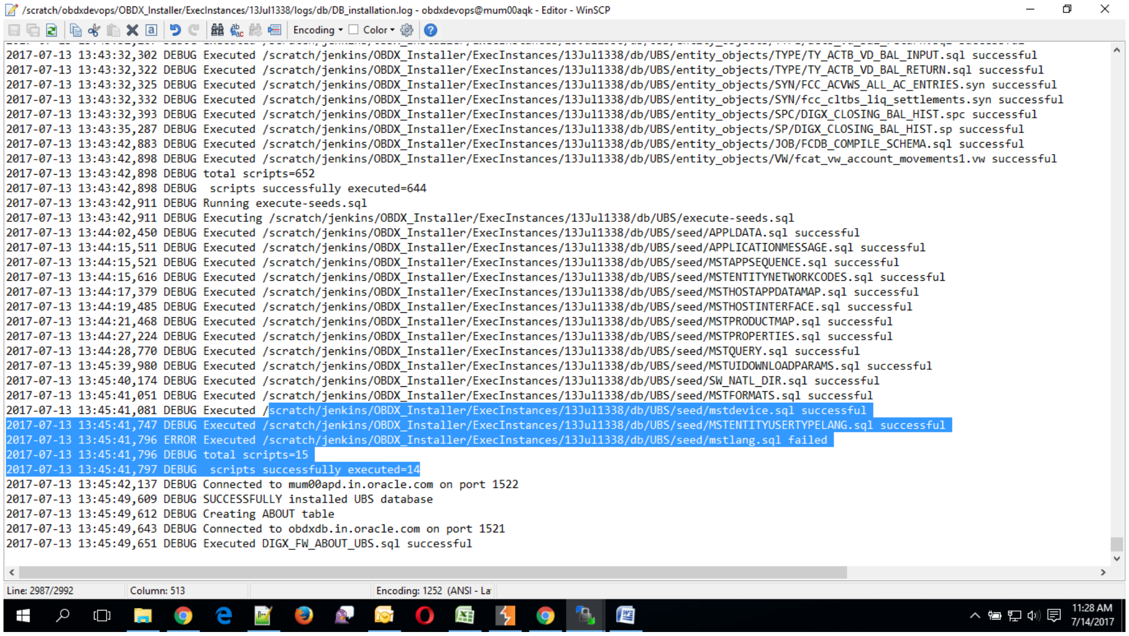


```
[devops@server /]$ export LD_LIBRARY_PATH=/usr/lib/oracle/18.3/client64/lib/:$LD_LIBRARY_PATH
[devops@server /]$ python3
Python 3.11.9 (main, Mar 25 2025, 10:00:00) [GCC 10.2.1 20210130 (Red Hat 10.2.1-11)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cx_Oracle
>>> cx_Oracle.__version__
'8.3.0'
```

Failed Database Scripts

This topic contains troubleshooting information in case of database script failures.

If you get the following error in DB_installation.log:



/scratch/obdxdevops/OBDX_Installer/ExecInstances/13Jul1338/logs/db/DB_installation.log - obdxdevops@mum00aqk - Editor - WinSCP
Encoding
Color
Line: 2987/2992
Column: 513
Encoding: 1252 (ANSI - La
11:28 AM
7/14/2017

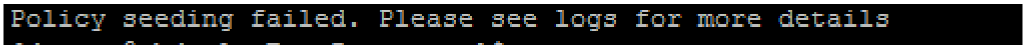
```
2017-07-13 13:43:32,302 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/TYPE/TY_ACTB_VD_BAL_INPUT.sql successful
2017-07-13 13:43:32,322 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/TYPE/TY_ACTB_VD_BAL_RETURN.sql successful
2017-07-13 13:43:32,325 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/SYN/FCC_ACVWS_ALL_AC_ENTRIES.syn successful
2017-07-13 13:43:32,332 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/SYN/fcc_cltbs_liq_settlements.syn successful
2017-07-13 13:43:32,393 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/SPC/DIGX_CLOSING_BAL_HIST.spc successful
2017-07-13 13:43:35,287 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/SP/DIGX_CLOSING_BAL_HIST.sp successful
2017-07-13 13:43:42,883 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/JOB/FCDB_COMPILE_SCHEMA.sql successful
2017-07-13 13:43:42,898 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/entity_objects/VW/fcat_vw_account_movements1.vw successful
2017-07-13 13:43:42,898 DEBUG total scripts=652
2017-07-13 13:43:42,898 DEBUG  scripts successfully executed=644
2017-07-13 13:43:42,911 DEBUG Running execute-seeds.sql
2017-07-13 13:43:42,911 DEBUG Executing /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/execute-seeds.sql
2017-07-13 13:44:02,450 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/APPLDATA.sql successful
2017-07-13 13:44:15,511 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/APPLICATIONMESSAGE.sql successful
2017-07-13 13:44:15,521 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTAPPSEQUENCE.sql successful
2017-07-13 13:44:15,616 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTENTITYNETWORKCODES.sql successful
2017-07-13 13:44:17,379 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTHOSTAPPDATAMAP.sql successful
2017-07-13 13:44:19,485 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTHOSTINTERFACE.sql successful
2017-07-13 13:44:21,468 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTPRODUCTMAP.sql successful
2017-07-13 13:44:27,224 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTPROPERTIES.sql successful
2017-07-13 13:44:28,770 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTQUERY.sql successful
2017-07-13 13:45:39,980 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTUIDOWNLOADPARAMS.sql successful
2017-07-13 13:45:40,174 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/SW_NATL_DIR.sql successful
2017-07-13 13:45:41,051 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTFORMATS.sql successful
2017-07-13 13:45:41,081 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/mstdevice.sql successful
2017-07-13 13:45:41,747 DEBUG Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/MSTENTITYUSERTYPELANG.sql successful
2017-07-13 13:45:41,796 ERROR Executed /scratch/jenkins/OBDX_Installer/ExecInstances/13Jul1338/db/UBS/seed/mstlang.sql failed
2017-07-13 13:45:41,796 DEBUG total scripts=15
2017-07-13 13:45:41,797 DEBUG  scripts successfully executed=14
2017-07-13 13:45:42,137 DEBUG Connected to mum00apd.in.oracle.com on port 1522
2017-07-13 13:45:49,609 DEBUG SUCCESSFULLY installed UBS database
2017-07-13 13:45:49,612 DEBUG Creating ABOUT table
2017-07-13 13:45:49,643 DEBUG Connected to obdxdb.in.oracle.com on port 1521
2017-07-13 13:45:49,651 DEBUG Executed DIGX_FW_ABOUT_UBS.sql successful
```

Check the detailed log of the failed SQL file at <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM> /logs/db folder.

Failure of Policy Seeding

This topic contains troubleshooting information if policy seeding fails during installation.

If you get the following error:



```
Policy seeding failed. Please see logs for more details
```

Try one of the following:

- Check if Entitlement.log is created on following path <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM>/logs/db/ and contains any SEVERE errors for Entitlement policy seeding.

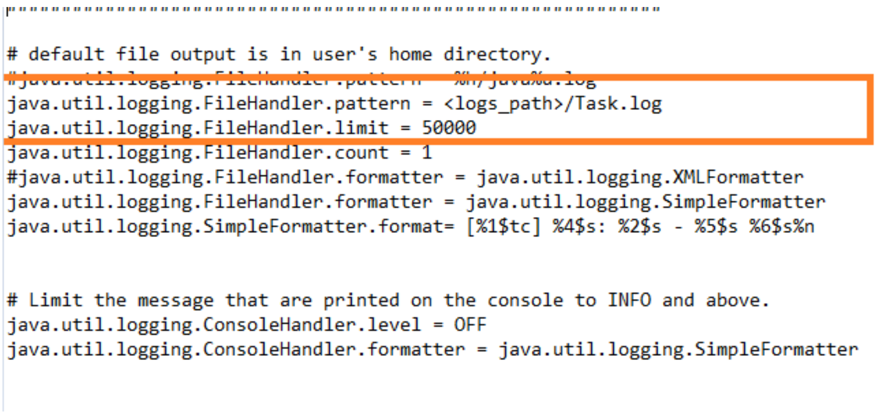
-
  ```
  Check if Task.log is created on following path
  <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM>/logs/db/ and contains
  any SEVERE errors for Task policy seeding.
  ```

-
  ```
  Check if Dashboard_seed.log is created on following path
  <OBAPI INSTALLER DIR>/ExecInstances/<DDMonthHHMM>/logs/db/ and contains
  any SEVERE errors for Dashboard policy seeding.
  ```

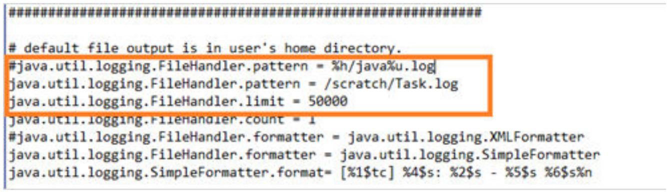
-
  ```
  Check the seedPolicies.log in <OBAPI INSTALLER DIR>/ExecInstances/
  <DDMonthHHMM>/logs/db/ directory
  if it contains any runtime errors generated during execution of the
  policies Seeding in OBAPI schema
  ```

Fix the problem by following below steps:

- Login to OBAPI installer server
- Over-write the policies files (Day0Policy.csv; Entitlement.csv; Resources.csv and Task.csv) from OBAPI Product zip into <OBAPI INSTALLER DIR>/installables/policies directory
- Browse to <OBAPI INSTALLER DIR>\ installables\policies
- Edit Entitlement_log4j.properties , Task_log4j.properties & Dashboard_seed_log4j.properties . Replace <logs_path> with directory where policy seeding logs will be generated
  e.g.



```
#################################################################

# default file output is in user's home directory.
#java.util.logging.FileHandler.pattern = %h/java%u.log
java.util.logging.FileHandler.pattern = <logs_path>/Task.log
java.util.logging.FileHandler.limit = 50000
java.util.logging.FileHandler.count = 1
#java.util.logging.FileHandler.formatter = java.util.logging.XMLFormatter
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
java.util.logging.SimpleFormatter.format= [%1$tc] %4$s: %2$s - %5$s %6$s%n


# Limit the message that are printed on the console to INFO and above.
java.util.logging.ConsoleHandler.level = OFF
java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter
```



```
#################################################################

# default file output is in user's home directory.
#java.util.logging.FileHandler.pattern = %h/java%u.log
java.util.logging.FileHandler.pattern = /scratch/Task.log
java.util.logging.FileHandler.limit = 50000
java.util.logging.FileHandler.count = 1
#java.util.logging.FileHandler.formatter = java.util.logging.XMLFormatter
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
java.util.logging.SimpleFormatter.format= [%1$tc] %4$s: %2$s - %5$s %6$s%n
```

- Run below command manually if “SEVERE” error logs are found in Task.log

  ```
  java -jar -Djava.util.logging.config.file='<logs.properties>’
  com.ofss.digx.utils.feed.data.task.jar "Task.csv"
  "oracle.jdbc.OracleDriver,
  <OBAPI Schema name>,<OBAPI Schema password>,jdbc:oracle:thin:@<OBAPI DB
  ```

```
hostname or
IP>:<OBAPI DB listener port>/<OBAPI Service Name>"
```

e.g.

```
java -jar -Djava.util.logging.config.file='Task_log4j.properties’
com.ofss.digx.utils.feed.data.task.jar ‘Task.csv’
"oracle.jdbc.OracleDriver,OBAPI_THP181,Welcome#1,
jdbc:oracle:thin:@xx.xx.xx.xx:1521/OBAPI"
```

- Run below command manually if “SEVERE” error logs are found in Entitlement.log

```
java -jar -Djava.util.logging.config.file='<logs.properties>’
com.ofss.digx.utils.entitlement.feed.data.jar
‘Resources.csv,Entitlement.csv,Day0Policy.csv’
'KERNEL'  "oracle.jdbc.OracleDriver,<OBAPI Schema name>,
<OBAPI Schema password>,jdbc:oracle:thin:@<OBAPI DB hostname
or IP>:<OBAPI DB listener port>/<OBAPI Service Name>"
```

e.g.

```
java -jar -Djava.util.logging.config.file='Entitlement_log4j.properties’
com.ofss.digx.utils.entitlement.feed.data.jar  ’Resources.csv,Entitlement.c
sv,
Day0Policy.csv’  'KERNEL' "oracle.jdbc.OracleDriver,OBAPI_THP201,Welcome#1,
jdbc:oracle:thin:@xx.xx.xx.xx:1521/OBAPI"
```

> **Note:**
>
> Please remove the space between multiple csv’s if there is any.

- Run below command manually if “SEVERE” error logs are found in Dashboard_seed.log

```
java -jar -Djava.util.logging.config.file='<logs.properties>’
com.ofss.digx.utils.dashboard.jar  ‘<path>/
dashboard_json’  "oracle.jdbc.OracleDriver,<OBAPI Schema name>,
<OBAPI Schema password>,jdbc:oracle:thin:@<OBAPI DB hostname or
IP>:<OBAPI DB listener port>/<OBAPI Service Name>"
```

e.g.

```
java -jar -Djava.util.logging.config.file= Dashboard_seed_log4j.properties’
com.ofss.digx.utils.dashboard.jar
‘/installables/policies/dashboard_json’
"oracle.jdbc.OracleDriver,OBAPI_THP201,
Welcome#1,jdbc:oracle:thin:@xx.xx.xx.xx:1521/OBAPI"
```

- Post successfully execution, restart Managed server.

Index