Oracle® Fusion Middleware

Using Oracle Data Integrator on Oracle Cloud Marketplace

12.2.1.3.0
F20379-03
July 2019

Oracle Fusion Middleware Using Oracle Data Integrator on Oracle Cloud Marketplace, 12.2.1.3.0

F20379-03

Copyright © 2019, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Preface

This book describes how to use Oracle Data Integrator on Oracle Cloud Marketplace.

This preface contains the following topics:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

Audience

This document helps you to use Oracle Data Integrator on Oracle Cloud Marketplace.

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 customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For more information, see the following documents in *Oracle Data Integrator Library*:

- *Release Notes for Oracle Data Integrator*
- *Understanding Oracle Data Integrator*
- *Developing Integration Projects with Oracle Data Integrator*
- *Installing and Configuring Oracle Data Integrator*
- *Upgrading Oracle Data Integrator*
- *Integrating Big Data with Oracle Data Integrator Guide*
- *Application Adapters Guide for Oracle Data Integrator*
- *Developing Knowledge Modules with Oracle Data Integrator*

- *Connectivity and Knowledge Modules Guide for Oracle Data Integrator Developer's Guide*
- *Oracle Data Integrator Tools Reference*
- *Data Services Java API Reference for Oracle Data Integrator*
- *Open Tools Java API Reference for Oracle Data Integrator*
- *Getting Started with SAP ABAP BW Adapter for Oracle Data Integrator*
- *Java API Reference for Oracle Data Integrator*
- *Getting Started with SAP ABAP ERP Adapter for Oracle Data Integrator*
- *Oracle Data Integrator 12c Online Help*, which is available in ODI Studio through the JDeveloper Help Center when you press **F1** or from the main menu by selecting **Help**, and then **Search** or **Table of Contents**.

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.

1

Getting Started with Oracle Cloud Marketplace

Oracle Data Integrator on Oracle Cloud Marketplace is a product offering that enables customers to quickly set up and run Oracle Data Integrator (ODI) on Oracle Cloud and it provides a fully unified solution for building, deploying, and managing complex data warehouses or as part of data- centric architectures in a SOA or business intelligence environment. In addition, it combines all the elements of data integration - data movement, data synchronization, data quality, data management, and data services - to ensure that information is timely, accurate, and consistent across complex systems.

It contains the following sections:

- Launching an Oracle Data Integrator Instance
- OCI Configuration
- Configuring ODI for Autonomous Data Warehouse (Optional)

> **Note:**
>
> Refer to Launching Your First Linux Instance documentation before creating the ODI instance.

Launching an Oracle Data Integrator Instance

Perform the following steps to create an instance of Oracle Data Integrator running on Oracle Cloud Infrastructure (OCI).

1. After finding Oracle Data Integrator on Oracle Cloud Marketplace, click Get App.
2. Select the OCI region in which you wish to install ODI, and click Launch Image.
3. Log in to your OCI account.
4. Click Launch Instance.
5. Select the latest version of the image from the drop-down list.
6. Select the compartment where you wish to install Oracle Data Integrator.
7. Review and accept the Oracle Terms of Use.
8. Click Launch Instance.

OCI Configuration

On the Create Compute Instance page, specify the following:

1. In the Name Your Instance text box, enter your choice of name. For example- `odi1`

2. From the Select an Availability Domain for your Instance drop-down, choose an availability domain.
3. In the Choose an operating system or image source, you can see the name of the OCI Console Marketplace image: `Oracle Data Integrator`.
4. For Choose Instance Type option, select Virtual Machine.
5. For Choose Instance Shape option, you can see a default shape, such as `VM.Standard2.1` shape.

   > **Note:**
   >
   > To use ODI, select a shape with a minimum of 2 OCPUs.

6. For Configure Boot Volume option, accept the default volume size.
7. For Add SSH Key option, choose either of the following options:
   - Select Choose SSH key file option, to specify the file containing your SSH public key that was generated previously.
   - Select Paste SSH keys option and paste the SSH public key content in the text field provided, using the content of the public key that was generated previously.
8. For Configure Networking option:
   - From the Virtual cloud network compartment drop-down list, select the compartment where your VCN has been setup.
   - From the Virtual cloud network drop-down list, choose your VCN.
   - For Subnet compartment option, specify the compartment containing the subnet that you wish to use.
   - For Subnet option, specify the subnet you wish to use (you need not change this normally from the default one, when you select the Subnet compartment).

   > **Note:**
   >
   > If the VM is associated with a public subnet and you wish to assign a public IP address, click the link Show Advanced Options, choose the Networking tab, and then select the Assign public IP address check box.

9. Click Create.

When the instance gets created (provisioned), it appears in the instance list. To view full details about it, including IP addresses, click the instance name in the displayed instance list.

Configuring ODI for Autonomous Data Warehouse (Optional)

If you aim to use an Autonomous Data Warehouse (ADW) or Autonomous Transaction Processing (ATP) in this tenancy as a part of your ODI transformations, you can use the ADW/ATP Dataserver setup UI to quickly setup this connection in ODI. An ODI Oracle Dataserver is created with all the credentials necessary for the connection. If your instance is newly created, wait for few minutes for the server to start functioning.

When the created ODI instance is fully provisioned and running, you can launch the Oracle ADW/ATP Dataserver setup UI by accessing the URL `http://<ipaddress>:8080/odi`.

In the above URL, IP address refers to the DNS hostname or IP address, if connecting to the instance externally. If connecting to the instance within the Virtual Cloud Network (VCN), you can use the Internal FQDN (fully qualified domain name), which is displayed on the OCI Console once the instance has been provisioned. For such internal connections, you can copy and paste this to your browser window and append or setup to get to the ODI ADW/ATP Dataserver setup page.

To list ADW/ATP instances in the ADW/ATP Instance drop-down and configure ADW/ATP Dataserver in ODI repository follow the below steps:

- Create a dynamic group to include matching rules for instances in a specified compartment. For example

  ```
  ALL {instance.compartment.id = 'ocid1.compart-
  ment.oc1..aaaaaaaabgr34tpuanpvq6xfb667nsmy2jz45zj6dexojhxdsv4mjayem3cq'}
  ```

- Navigate to Identity -> Policies -> Create Policy to create policy statements as specified below :
  If you set policy at:
  - ODI compartment level, then all ADW/ATP instances from the compartment where ODI instance is created are listed.
    For example - To List ADW/ATP instances only from the ODI instance compartment, you have to setup the following policy:

    ```
    Create dynamic group like: ALL {instance.compartment.id =
    'ocid1.compartment.oc1..aaaaaaaabgr34tpuanpvq6xfb667nsmy2jz45zj6dex-
    ojhxdsv4mjayem3cq'}
    Allow dynamic-group odi_group to inspect autonomous-database-family
    in compartment odi
    Allow dynamic-group odi_group to read autonomous-database-family in
    compartment odi
    Allow dynamic-group odi_group to inspect compartments in compart-
    ment odi
    ```

  - ODI tenant level, then all ADW/ATP instances from all the compartments of the tenancy are listed.
    For example - To list ADW/ATP instances from all the compartments of tenancy, you have to setup the following policy:

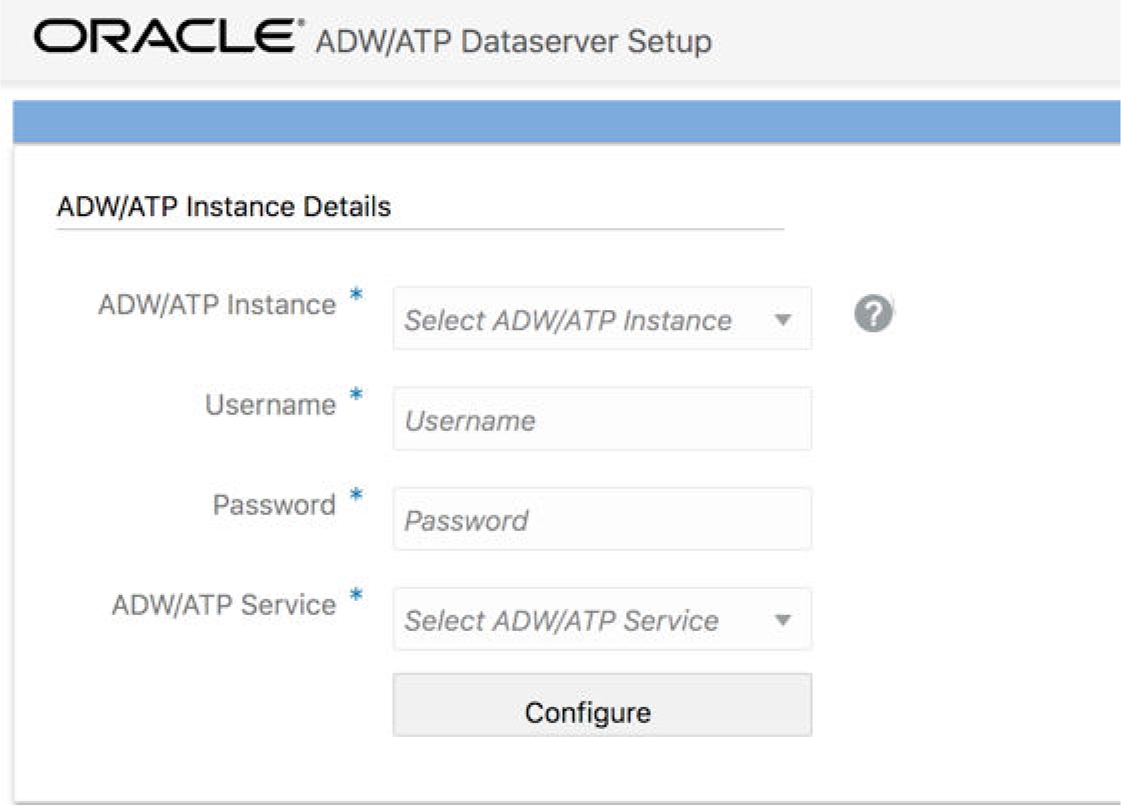
    ```
    Create dynamic group like: ALL {instance.compartment.id =
    'ocid1.compartment.oc1..aaaaaaaabgr34tpuanpvq6xfb667nsmy2jz45zj6dex-
    ojhxdsv4mjayem3cq'}
    Allow dynamic-group odi_group to inspect autonomous-database-family
    in tenancy
    Allow dynamic-group odi_group to read autonomous-database-family in
    tenancy
    Allow dynamic-group odi_group to inspect compartments in tenancy
    ```

- If the policies are not created previously during ODI Instance creation, then create the policies (as mentioned above) and restart the `jettyodi.service` before launching the ADW/ATP Dataserver UI.

Refer to Troubleshooting ODI on OCI, to know how to restart the `jettyodi.service`.

To create an ODI Oracle Dataserver connecting to an ADW/ATP instance,

Provide the following details in the ADW/ATP Instance Details interface:



1. From the ADW/ATP Instance drop-down list, select the required ADW/ATP instance.
   The ADW/ATP instances <`CompartmentName-ADW/ATP InstanceName`> are listed.When you select an instance, in the backend, the ADW/ATP dataserver is created in the format `ADW/ATP Instance Name + "-" + RandomNuber`.
2. In the User Name and Password fields, provide the ADW/ATP user credentials.
3. From the ADW/ATP Service drop-down list, select the connection details.
4. Click Configure.

This creates an Oracle Dataserver for the ADW/ATP instance in the ODI repository.

You can create multiple Dataservers for the same ADW/ATP instance without considering ADW/ATP service profiles (High, Low or Medium).

Note:

Make sure you provide correct ADW user credentials while creating the ADW Dataserver. If you provide wrong user credentials you may not be able to connect to the newly created Dataserver. However, you can update the ADW/ATP user credentials, if needed, from ODI studio once you create the ADW/ATP Dataserver.

2 Working with ODI Instance

This chapter guides you to connect and work with the ODI instance.

It contains the following sections:

Connecting to ODI Instance

You can connect to an ODI instance by using a Secure Shell (SSH) connection. Most Linux distributions include an SSH client by default. For Windows, you can download a free SSH client called PuTTY from http://www.putty.org.

> **Note:**
>
> When you're logged in as the default user - opc, you can use the sudo command to run administrative tasks.

1. To connect to your ODI instance from linux,
   - Log in to your instance using SSH.
   - Use the following command to set the file permissions so that only you can read the file:

     ```
     $ chmod 400 <private_key>
     ```

     where `<private_key>` is the full path and name of the file that contains the private key associated with the instance you want to access.
   - Use the following SSH command to access the instance.

     ```
     $ ssh –i <private_key> <username>@<public-ip-address>
     ```

     where

    - `<private_key>` is the full path and name of the file that contains the private key associated with the instance you want to access.
    - `<username>` is the default name for the instance. The default user name is `opc`.
    - `<public-ip-address>` is your instance IP address that you retrieved from the Console.

2. To connect to your ODI instance from windows,
    - Open putty.exe.
    - In the Category pane, select Window, and then select Translation.
    - In the Remote character set drop-down list, select UTF-8. The default locale setting on Linux-based instances is UTF-8, and this configures PuTTY to use the same locale.
    - In the Category pane, select Session and enter the following:
        a. Host Name (or IP address):`<username>@<public-ip-address>` , where `<username>` is the default name for the instance. For Oracle Linux and CentOS images, the default user name is opc. For the Ubuntu image, the default name is ubuntu and `<public-ip-address>` is your instance public IP address that you retrieved from the console.
        b. Port: 22
        c. Connection type: SSH
    - In the Category pane, expand Connection, expand SSH, and then click Auth.
    - Click Browse, and then select your private key.
    - Click Open to start the session.

    If this is your first time connecting to the instance, you might see a message that the server's host key is not cached in the registry. Click Yes to continue the connection.

Starting ODI Studio

To access ODI studio through VNC, do the following:

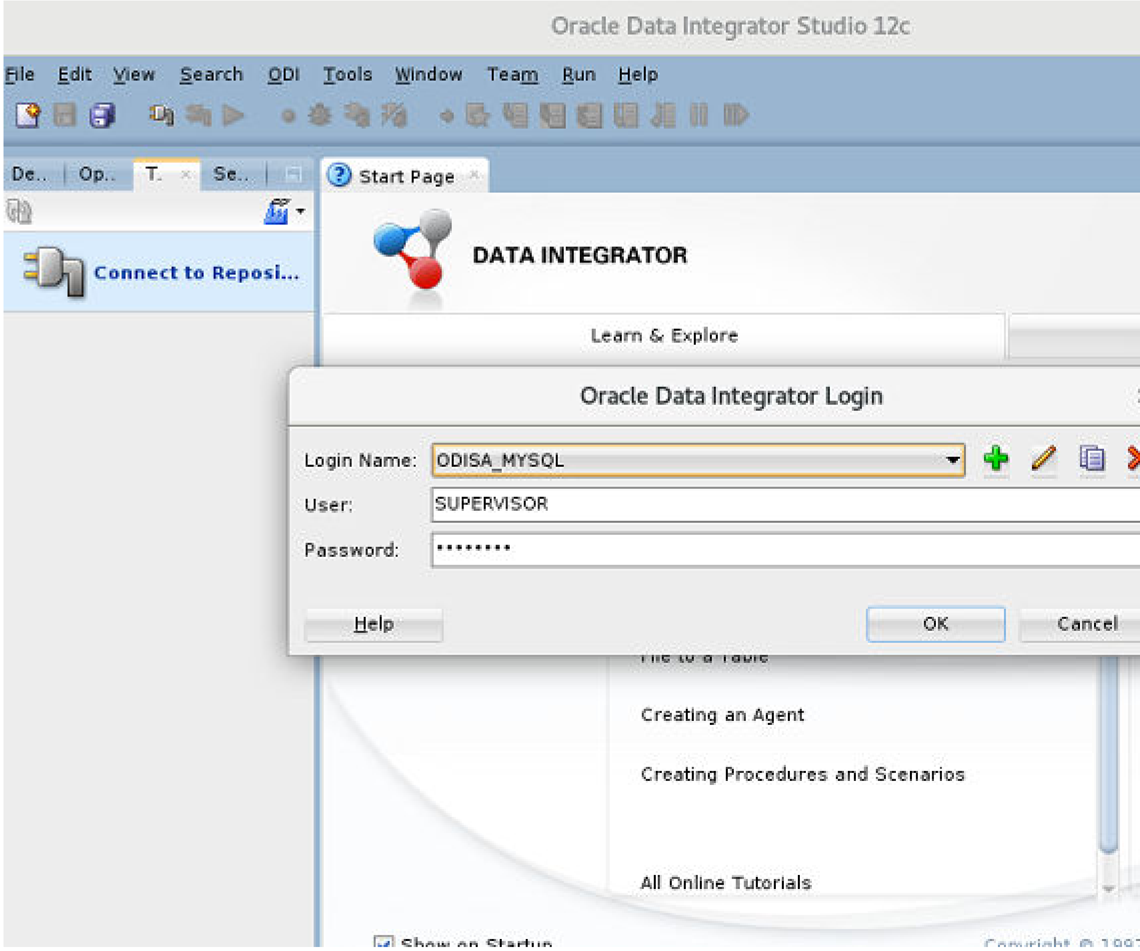
1. Install a VNC viewer on your local computer.
2. Use SSH to connect to the compute instance running the Oracle Data Integrator Image, as described in Connecting to ODI Instance.
3. Configure a VNC password by typing `vncpasswd`.
4. When prompted, enter a new password (for example - welcome1) and verify it.
5. Optionally, enter a view only password.
6. After the `vncpasswd` utility exits, start the VNC server by typing `vncserver`, for example `vncserver -geometry 1920x1000`.
   This will start a VNC server with display number 1 for the opc user, and the VNC server starts automatically if your instance is rebooted.
7. On your local computer, connect to your instance and create a ssh tunnel for port 5901 (for display number 1):`$ ssh -L 5901:localhost:5901 –i id_rsa opc@<IP Address>`.

8. On your local computer, start a VNC viewer and establish a VNC connection to `localhost:1`.
9. Enter the VNC password that you had set earlier.

To launch the ODI instance,

1. From the Applications menu, navigate to Accessories → ODI Studio
   or

   Navigate to the location `/home/opc/oracle/odi/studio/bin/odi` in the VNC.
2. Connect to the repository with the already populated login credentials.



Oracle Data Integrator Login

Login Name: ODISA_MYSQL

User: SUPERVISOR

Password: ••••••••

> **Note:**
>
> While connecting to your repository, if you had closed your MySQL instance by mistake,use the following commands to restart it:
>
> ```
> sudo systemctl stop mysqlodi.service
> sudo systemctl start mysqlodi.service
> ```

Use the following command to get the MySQL user password

```
/home/opc/oracle/odi/common/scripts/getPassword.sh
```

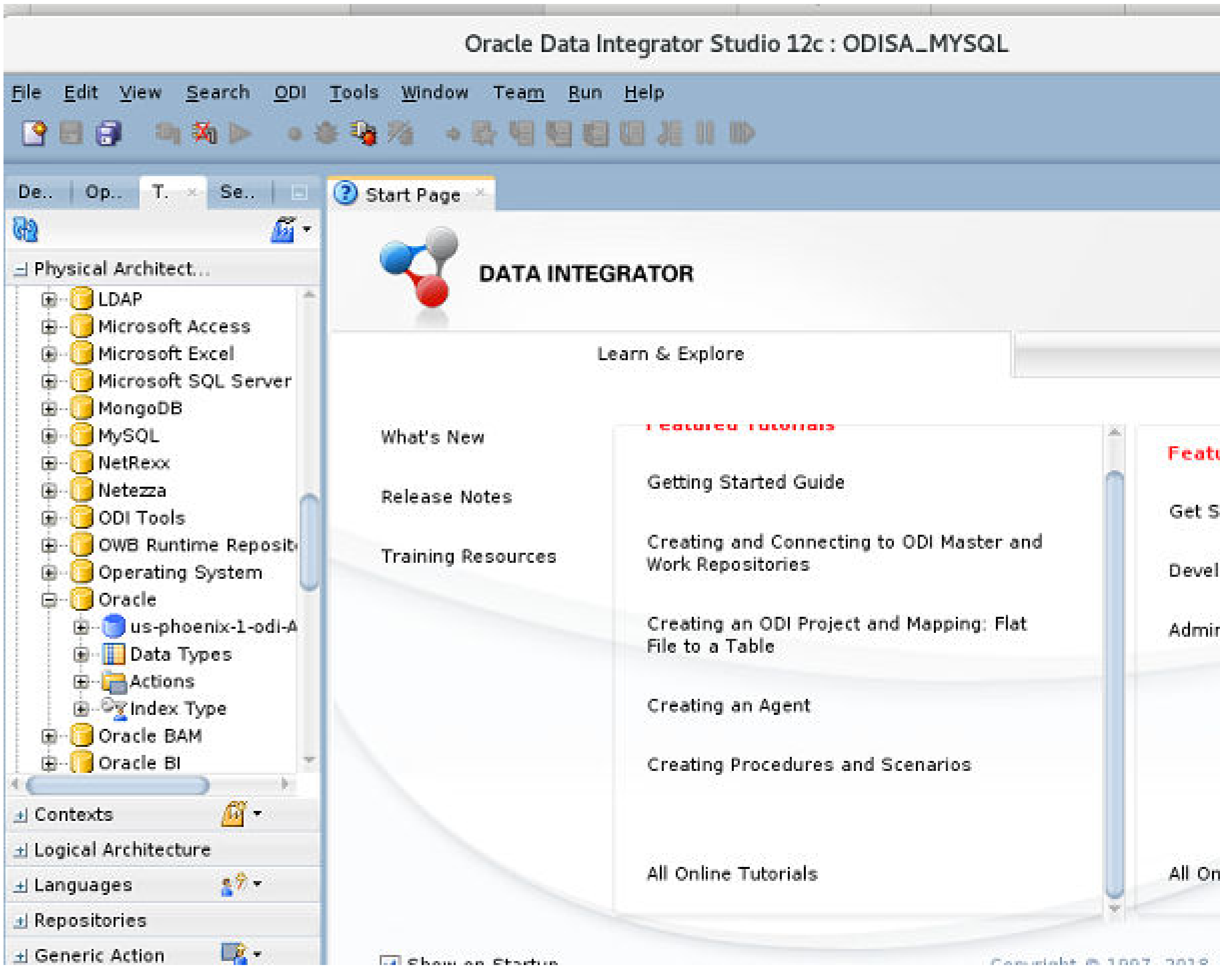
This command helps you to get the default password for all MySQL users (`root/DEV_ODI_REPO`).

> **Note:**
>
> Do not change the default password for DEV_ODI_REPO user.

3. Navigate to the ODI web app and configure the ADWC Dataserver setup. You can access the ODI web app using the URL `http://<IPADDRESS>:8080/odi`. Use the following commands to start and stop the ODI web app:

```
sudo systemctl stop jettyodi.service
sudo systemctl start jettyodi.service
```

4. Post successful configuration, check if the newly created data server is available in the Topology navigator -> Technologies -> Oracle



5. In the ODI studio, check if the Oracle Standalone Agent is working. Use the following commands to stop and start the standalone agent:

   ```
   sudo systemctl stop agentodi.service
   sudo systemctl start agentodi.service
   ```

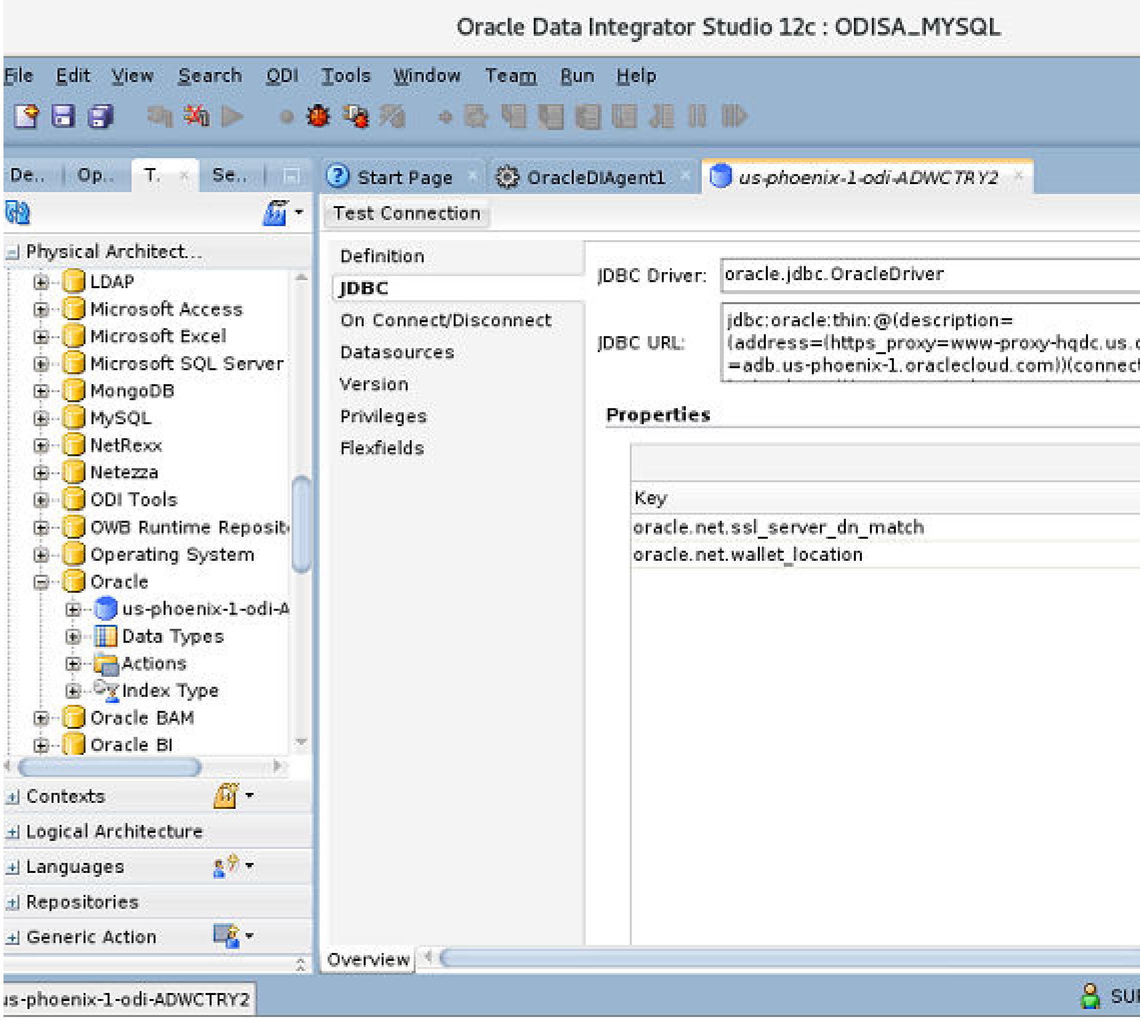
   Use the following command to know the status of the standalone agent:

   ```
   sudo systemctl status agentodi.service
   ```

   Use the following command to view the logs:

   ```
   sudo journalctl -u agentodi.service -f
   ```

6. Depending on your network, you may need to provide proxy details for the database server JDBC connection.



JDBC Driver: oracle.jdbc.OracleDriver

JDBC URL:

```
jdbc:oracle:thin:@(description=
(address=(https_proxy=www-proxy-hqdc.us.
=adb.us-phoenix-1.oraclecloud.com))(connect
```

Key
oracle.net.ssl_server_dn_match
oracle.net.wallet_location

7. Click Test connection, to check if the created ADW Dataserver is working.

8. Depending on your network, you can setup a proxy for ODI. In ODI Studio, navigate to Tools, Preferences, Web Browser and Proxy, to setup a proxy for your net-

work. Proxy may be required for accessing certain hosts, for example - Oracle Object Storage.

For executing mappings and packages with the standalone agent:

- Use the following command to stop the agent:

  ```
  sudo systemctl stop agentodi.service
  ```

- From the ODI OCI instance, navigate to the location `/home/opc/oracle/odi/common/scripts` and edit the file `startAgent.py` and add the following lines after the property *`after -Drepo.props=`*

  ```
  -Dhttp.proxyHost=www-proxy-xxx.com -Dhttp.proxyPort=80
  -Dhttps.proxyHost=www-proxy-xxx.com -Dhttps.proxyPort=80 -cp
  ```

  For example, after adding the above lines, your file should be like this:

  ```
   subprocess.call('nohup java
  -Drepo.props=odi-standalone-studio.properties
  -Dhttp.proxyHost=www-proxy-xxx.com -Dhttp.proxyPort=80
  -Dhttps.proxyHost=www-proxy-xxx.com -Dhttps.proxyPort=80 -cp
  $AGENTCLASSPATH oracle.odi.OdiStandaloneAgentStarter'+' '+oracledia-
  gentPath+"
  &", shell=True)
  ```

- Save the file and use the following command to start the agent:

  ```
  sudo systemctl start agentodi.service
  ```

- Test the standalone agent from ODI studio to see if the agent has started successfully. Then run the mapping using the standalone agent.

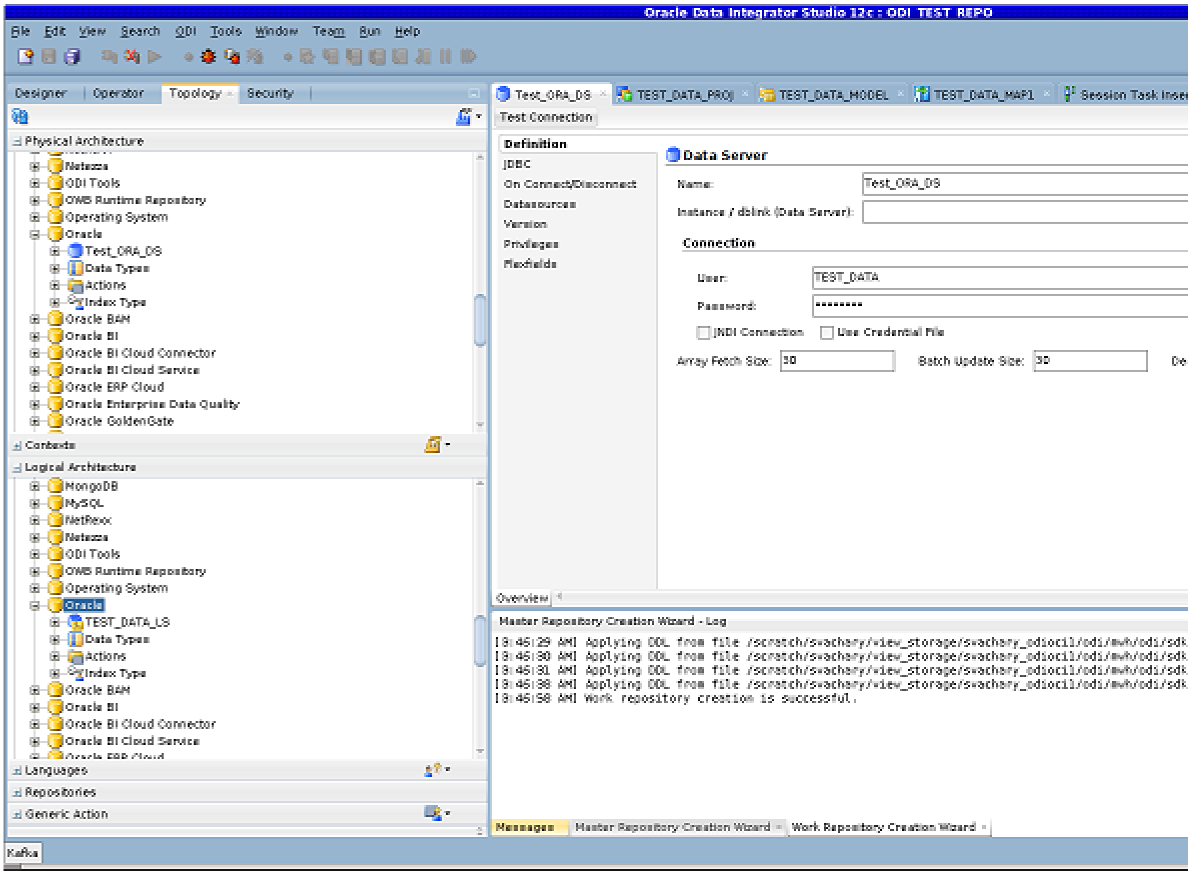
  > **Note:**
  >
  > Ensure you do not add any extra lines or space or tab on this file. Just add -D option within the line content. It is a python script and it requires proper line indentation to work.

> **Note:**
>
> If you are using a BI Cloud Connector Dataserver, you may need to add the BI Cloud Connector host to the Proxy Exclusion field.

Configuring Data Sources and Targets

The physical components that store and expose structured data in Oracle Data Integrator (ODI) are defined as data servers. Each data server is always linked to a single technology. It stores information according to a specific technical logic, which is declared in the physical schemas attached to it.

For example -



Data Server

Name: Test_ORA_DS

Instance / dblink (Data Server):

Connection

User: TEST_DATA

Password: ••••••••

□ JNDI Connection □ Use Credential File

Array Fetch Size: 30 Batch Update Size: 30 De

Master Repository Creation Wizard - Log

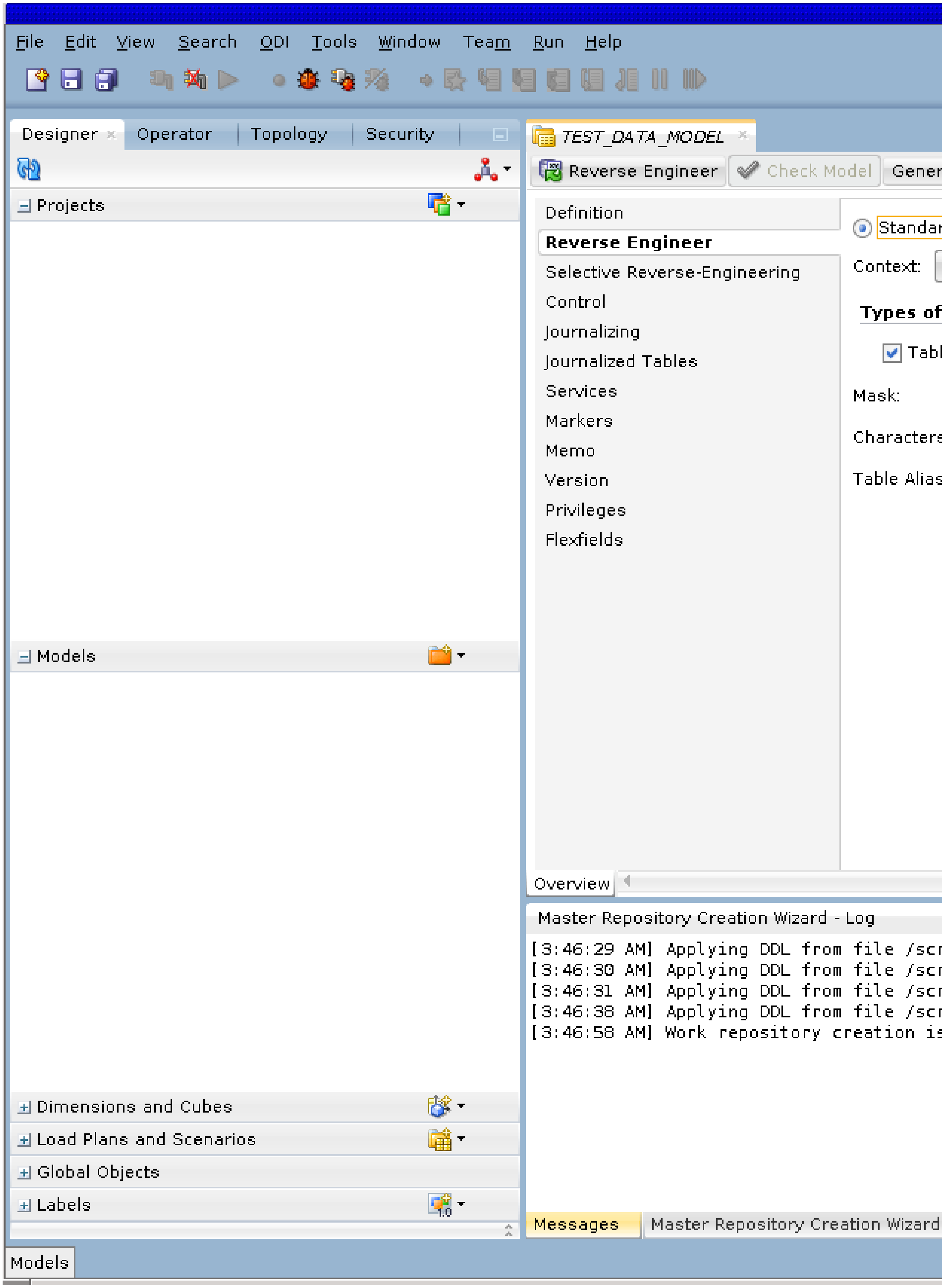
```
[3:46:29 AM] Applying DDL from file /scratch/swachary/view_storage/swachary_odiocil/odi/mwh/odi/sdk
[3:46:30 AM] Applying DDL from file /scratch/swachary/view_storage/swachary_odiocil/odi/mwh/odi/sdk
[3:46:31 AM] Applying DDL from file /scratch/swachary/view_storage/swachary_odiocil/odi/mwh/odi/sdk
[3:46:38 AM] Applying DDL from file /scratch/swachary/view_storage/swachary_odiocil/odi/mwh/odi/sdk
[3:46:58 AM] Work repository creation is successful.
```

For more information, refer to Overview of Oracle Data Integrator Topology chapter in *Developing Integration Projects with Oracle Data Integrator* guide.

Reverse Engineering Data Models

To automatically populate datastores into the model, you reverse-engineer the model. A standard reverse-engineering uses the capacities of the JDBC driver used to connect the data server to retrieve the model metadata. A customized reverse-engineering uses a reverse-engineering Knowledge Module (RKM), to retrieve metadata for a specific type of technology and create the corresponding datastore definition in the data model.

For example -



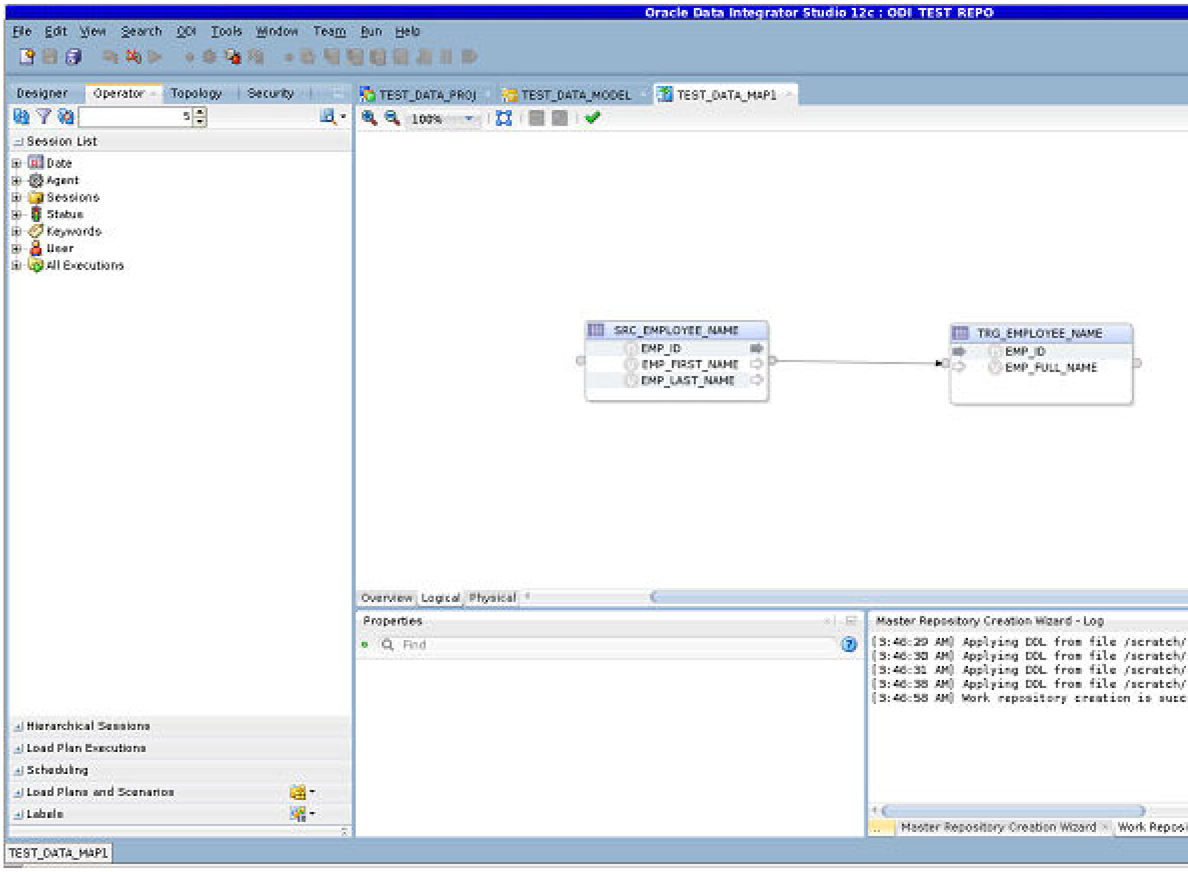
```
[3:46:29 AM] Applying DDL from file /scr
[3:46:30 AM] Applying DDL from file /scr
[3:46:31 AM] Applying DDL from file /scr
[3:46:38 AM] Applying DDL from file /scr
[3:46:58 AM] Work repository creation is
```

For more information, refer to Creating and Reverse-Engineering a Model chapter in *Developing Integration Projects with Oracle Data Integrator* guide.

Creating Mappings

Mappings in Oracle Data Integrator (ODI) are the logical and physical organization of your data sources, targets, and the transformations through which the data flows from source to target. Mappings are made up of several parts, datastores, datasets, re-usable mappings, connectors, knowledge modules, variables, sequences, user functions, and other components. Optionally, you can specify a staging schema. You create and manage mappings using the mapping editor, which opens whenever you open a mapping. Mappings are organized in folders under individual projects, found under Projects in the Designer Navigator.

For example -



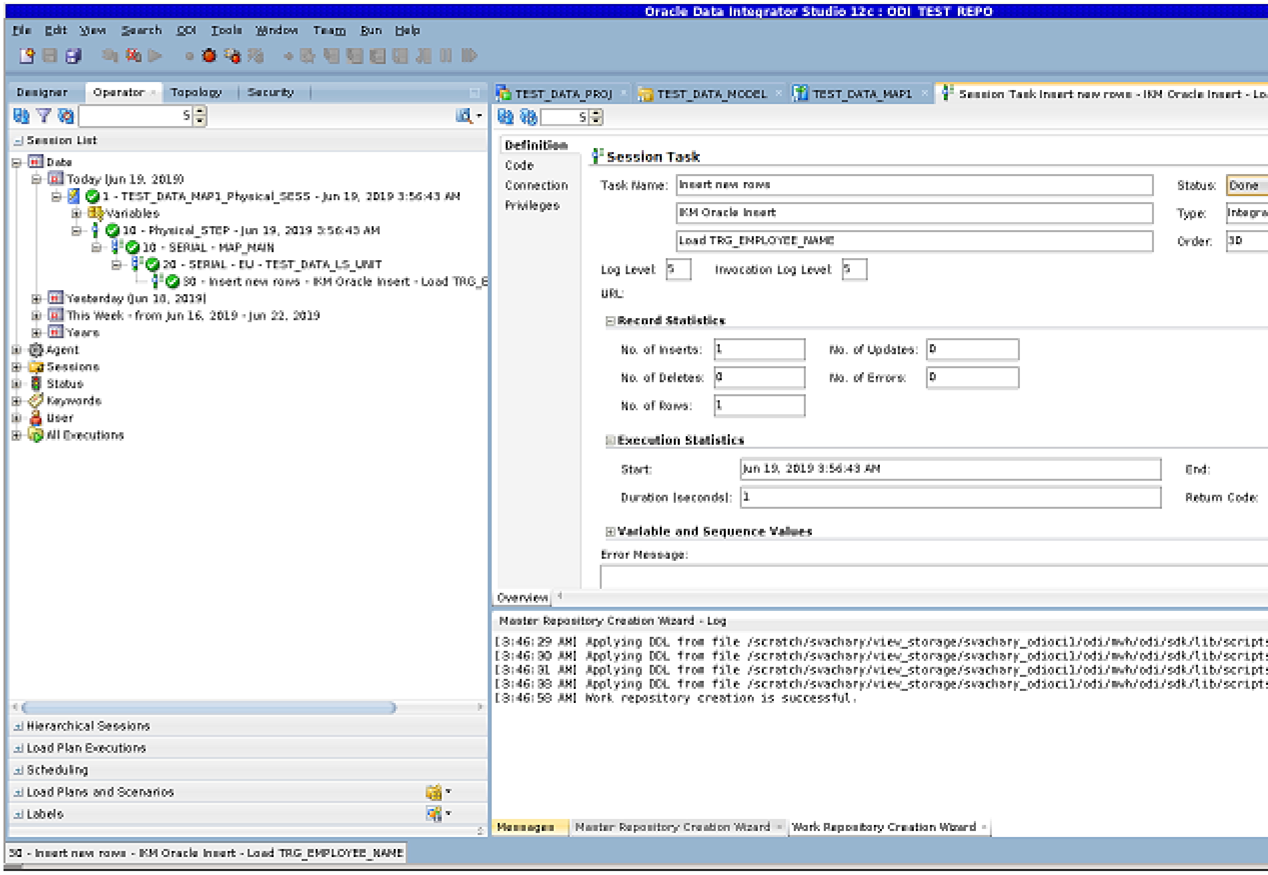
```
[9:46:29 AM] Applying DDL from file /scratch/
[9:46:30 AM] Applying DDL from file /scratch/
[9:46:31 AM] Applying DDL from file /scratch/
[9:46:38 AM] Applying DDL from file /scratch/
[9:46:58 AM] Work repository creation is succ
```

For more information, refer to Creating and Using Mappings in *Developing Integration Projects with Oracle Data Integrator* guide.

Monitoring ODI Executions

Monitoring your development executions consists of viewing the execution results and managing the development executions when the executions are successful or in error. Through Operator Navigator, you can view your execution results and manage your development executions in the sessions, as well as the Scenarios and Load Plans in production. Operator navigator stores this information in a work repository, while using the topology defined in the master repository.

For example -


Oracle Data Integrator Studio 12c : ODI_TEST_REPO
File Edit View Search ODI Tools Window Team Run Help
Designer Operator Topology Security
TEST_DATA_PROJ
TEST_DATA_MODEL
TEST_DATA_MAP1
Session Task Insert new rows - IKM Oracle Insert - Lo
Session List
Hierarchical Sessions
Load Plan Executions
Scheduling
Load Plans and Scenarios
Labels
Definition
Code
Connection
Privileges
Overview
Messages
Master Repository Creation Wizard
Work Repository Creation Wizard

```
Date
  Today (Jun 19, 2019)
    1 - TEST_DATA_MAP1_Physical_SESS - Jun 19, 2019 3:56:43 AM
      Variables
      10 - Physical_STEP - Jun 19, 2019 3:56:43 AM
        10 - SERIAL - MAP_MAIN
          20 - SERIAL - EU - TEST_DATA_LS_UNIT
            30 - Insert new rows - IKM Oracle Insert - Load TRG_E
  Yesterday (Jun 18, 2019)
  This Week - from Jun 16, 2019 - Jun 22, 2019
  Years
Agent
Sessions
Status
Keywords
User
All Executions
```

Session Task

Task Name: Insert new rows

IKM Oracle Insert

Load TRG_EMPLOYEE_NAME

Status: Done

Type: Integra

Order: 30

Log Level: 5 Invocation Log Level: 5

URL:

Record Statistics

No. of Inserts: 1 No. of Updates: 0

No. of Deletes: 0 No. of Errors: 0

No. of Rows: 1

Execution Statistics

Start: Jun 19, 2019 3:56:43 AM End:

Duration (seconds): 1 Return Code:

Variable and Sequence Values

Error Message:

Master Repository Creation Wizard - Log

```
[3:46:29 AM] Applying DDL from file /scratch/svachary/view_storage/svachary_odioci1/odi/mwh/odi/sdk/lib/script
[3:46:30 AM] Applying DDL from file /scratch/svachary/view_storage/svachary_odioci1/odi/mwh/odi/sdk/lib/script
[3:46:31 AM] Applying DDL from file /scratch/svachary/view_storage/svachary_odioci1/odi/mwh/odi/sdk/lib/script
[3:46:38 AM] Applying DDL from file /scratch/svachary/view_storage/svachary_odioci1/odi/mwh/odi/sdk/lib/script
[3:46:58 AM] Work repository creation is successful.
```

30 - Insert new rows - IKM Oracle Insert - Load TRG_EMPLOYEE_NAME

For more information, refer to Monitoring Integration Processes chapter in *Administering Oracle Data Integrator* guide.

Installation Locations

Please note, the following installation locations are used by this image. You may need this information if you want to change any aspects of the installation:

Table 2-1 Installation Locations

Area	Location on Server
Oracle Home (home of the ODI installation for patching).	`/home/opc/oracle`
OPatch Home	`/home/opc/oracle/OPatch`
MySQL Home	`/home/opc/mysql_home`

Note:

To retrieve the password stored in wallet, execute the following command -

```
/home/opc/oracle/odi/common/scripts/getPassword.sh
```

.

Patching

The ODI image on the Oracle Cloud Marketplace contains a standard installation of ODI Marketplace v1.0.

3 Troubleshooting ODI on OCI

This chapter describes about various services associated with ODI on OCI and ways to troubleshoot them when you encounter issues while using them.

> **Note:**
>
> If you are facing issues connecting to ADW dataserver and MySQL repository after long hours of inactivity, try reconnecting to ODI repository to overcome this problem.

Services available in ODI VIM are:

```
jettyodi.service
agentodi.service
mysqlodi.service
```

Essential commands necessary for troubleshooting these services are:

- Use the following command to check the status of embedded jetty service:

  ```
  systemctl status jettyodi.service
  ```

- Use the following command to check the log of embedded jetty service:

  ```
  In a commandline type: journalctl -u jettyodi.service -f
  ```

  `systemd-journald` can be configured to persist your systemd logs on disk, and it also provides controls to manage the total size of your archived logs. These settings are defined in `/etc/systemd/journald.conf`. To start persisting your logs, uncomment the Storage line in`/etc/systemd/journald.conf` and set its value to `persistent`. Your archived logs will be held in `/var/log/journal`. If this directory does not already exist in your file system, `systemd-journald` will create it. After updating your journald.conf, load the change: `sudo systemctl restart systemd-journald`

- Use the following command to restart the embedded jetty service:

  ```
  systemctl daemon-reload
  systemctl restart jettyodi.service
  ```

- Use the following command to restart mysql service:

  ```
  systemctl daemon-reload
  systemctl restart mysqlodi.service
  ```

Note:

Use the following commands to start or stop any of the services:

```
sudo systemctl start <service_name>
sudo systemctl stop <service_name>
```