



Autonomous Data Warehouse Cloud

Updated: May 23, 2018

Lab 9: Using Oracle Data Sync

Oracle Data Sync is a client tool that uploads data from files, relational tables, and OTBI (Oracle Transactional Business Intelligence) to a variety of Oracle Cloud Services including ADWC.

Oracle Data Sync can connect to over 20 different Databases and other sources, and load data into any Oracle Cloud Service.

Use Data Sync when you want to:

- Load data from a variety of data sources, such as CSV, XLSX, Oracle databases and non-Oracle sources.
- Perform incremental data loads into ADWC.
- Merge data from multiple sources during loading.
- Transform data while loading.
- Schedule the data loads.
- Perform insert-only or append strategies.

In this lab you will use **Data Sync** to load data from an Oracle Database to ADWC.

Objectives

- Configure connection to ADWC using **Data Sync**
- Load a table residing in an Oracle Database to ADWC
- Perform incremental data loads into ADWC

Required Artifacts

- The Lab requires an Oracle Autonomous Data Warehouse Cloud subscription.
 - Locate your **Cloud Account Name, Username, and Password**.
- Access to the Lab VM.
 - Locate the Lab VM's **IP Address, User Name and Password**
- Connection to a source Oracle Database
 - Locate the Oracle Database **User ID and Password** provided to you by the instructor for the Data Sync Lab.

Lab Steps

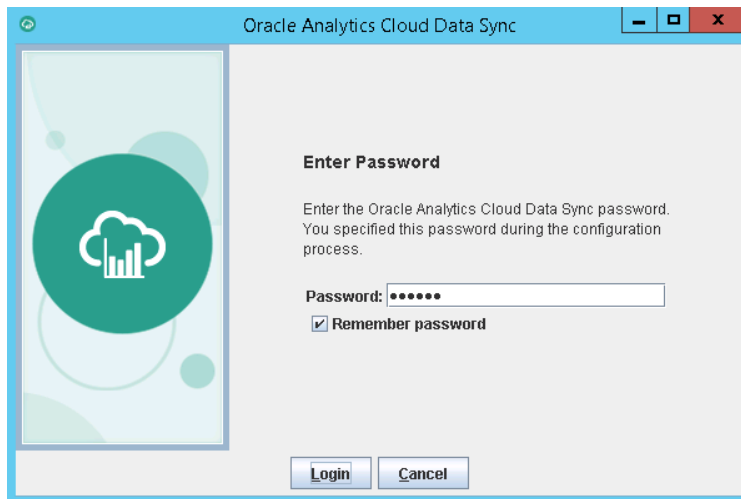
Step 1: Start Data Sync

- Sign in to the Lab VM using the credentials provided to you by the instructor.
- Start **Data Sync** by clicking on **DataSync.bat** on the desktop.

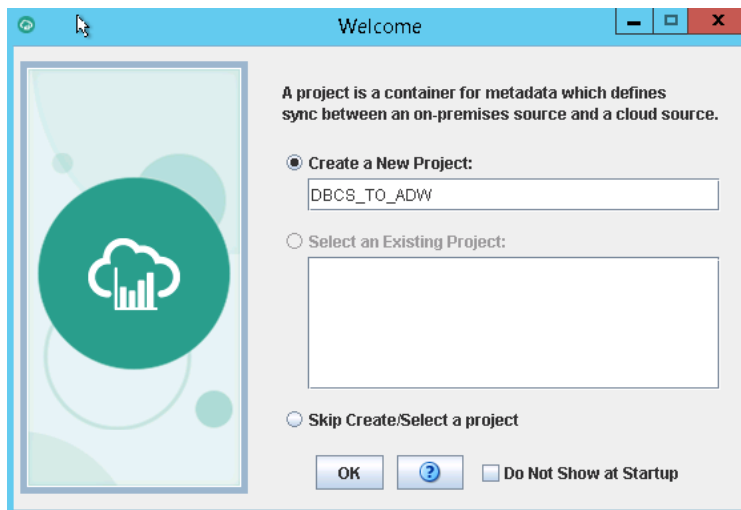


If starting Data Sync for the first time, it will create a **Repository** and will prompt for a Repository Name and a Password. In the Lab environment the Repository is pre-created and the password is saved.

- On the **Enter Password** pop-up, note that the password has been already entered and saved. Click **Login**.



- Select **Create a New Project**, enter **DBCS_TO_ADW** as the project name and Click **OK**.



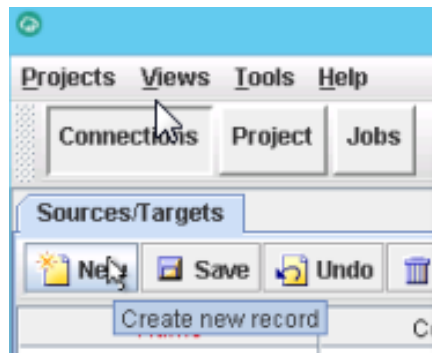
Step 2: Create Connection to ADWC (Target)

In Data Sync, use the **Sources/Targets** dialog in the Connections view to specify connections details for your target database and your source databases. Data Sync loads data from these sources to the target location.

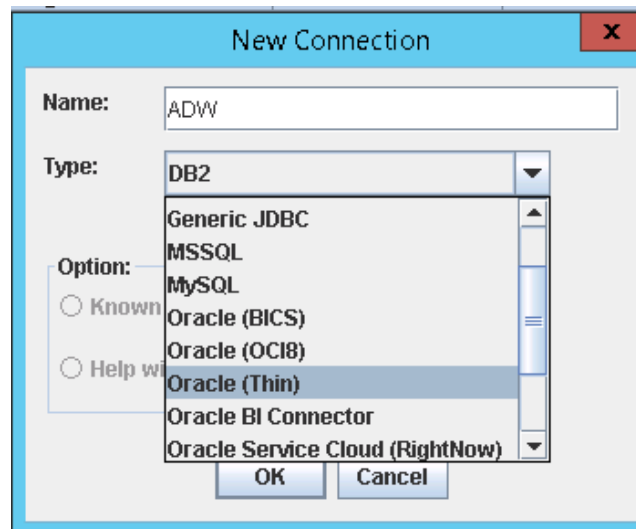
Note: If you're loading data only from data files, for example XLSX or CSV format, then you don't need a connection in Data Sync.

You need to create two connections for this Lab, one for the **Source** Oracle Database, and one for the **Target** Autonomous Data Warehouse Cloud Service Database.

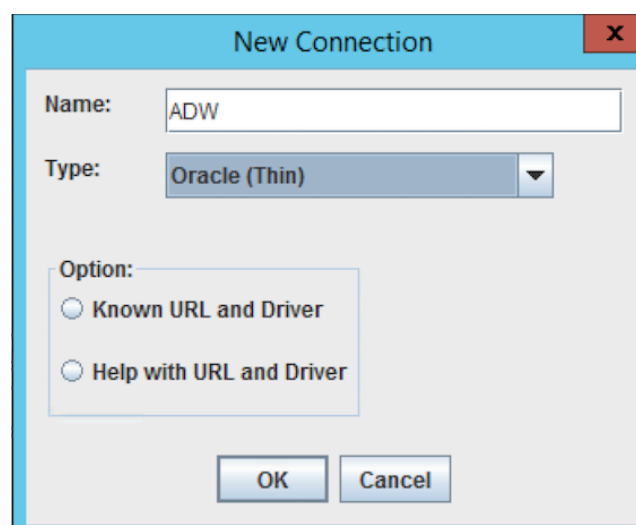
- Create the connection to ADWC which will be the **Target**.
- Click **Connections** and then click **New**.



- In the **New Connection** window enter the following:
 - Name: **ADW**
 - Type: **Oracle (Thin)**



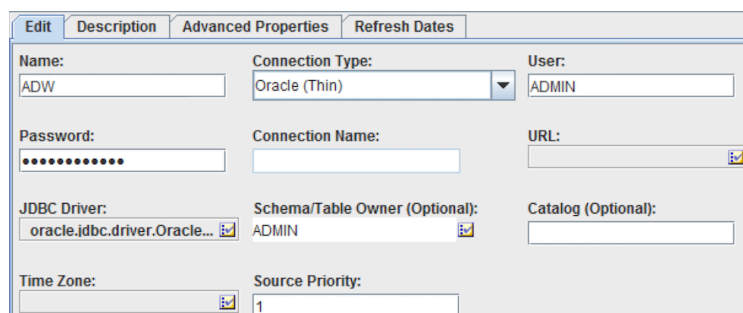
- Do not select any of the **Options** presented. Click **OK**. This will add the ADW Connection.



- Ensuring that **ADW** is selected in the upper half, enter the following in **Edit** tab in the lower half:
 - User: **ADMIN**
 - Password: Enter the **Password** for ADMIN specified earlier
 - Schema/Table Owner (Optional): **ADMIN** (Uppercase)
 - Click on **JDBC Driver** and enter the following in the pop-up and click **OK**:

```
oracle.jdbc.driver.OracleDriver
```

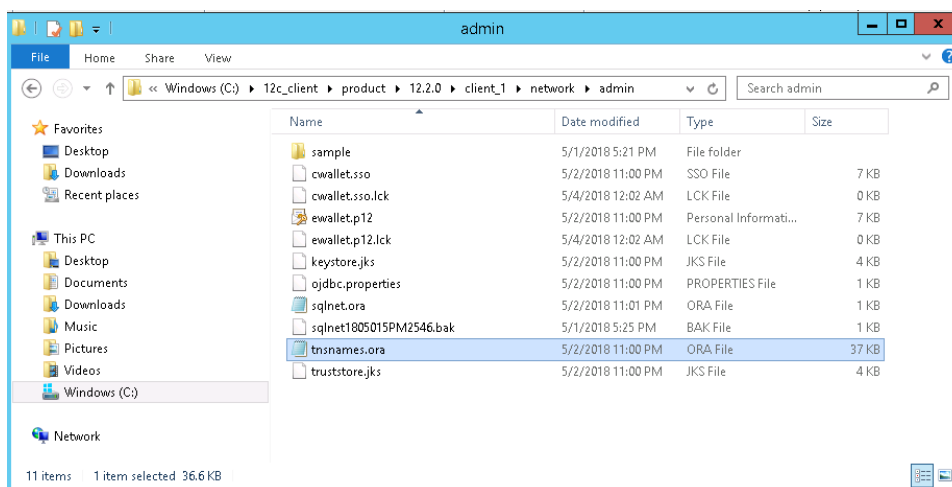
- This is how your **Edit** tab should look so far:



- Next, browse to the following folder in ORACLE_HOME in the Lab VM:

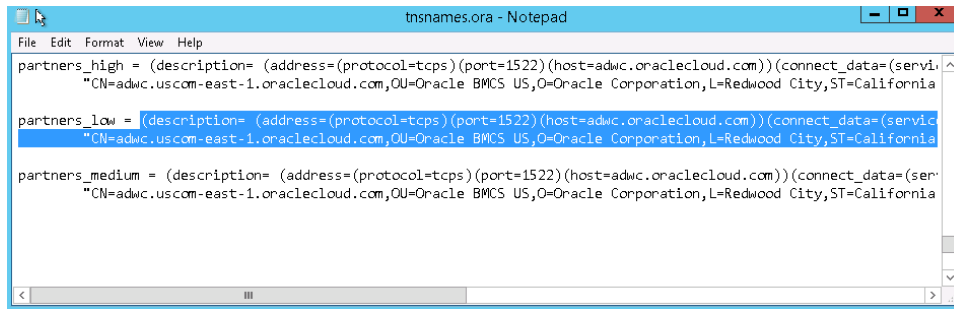
```
C:\12c_client\product\12.2.0\client_1\network\admin
```

- View the contents of the **tnsnames.ora** file.



- You may see many connections defined in this file. For each individual ADWC database end point, there should be 3 entries, with each entry corresponding to the database service name for ADWC.

- Locate the service name entries for your ADWC Database, and more specifically the service name that ends with **Low**. Copy the text starting from "(DESCRIPTION)" to the end of connect string.



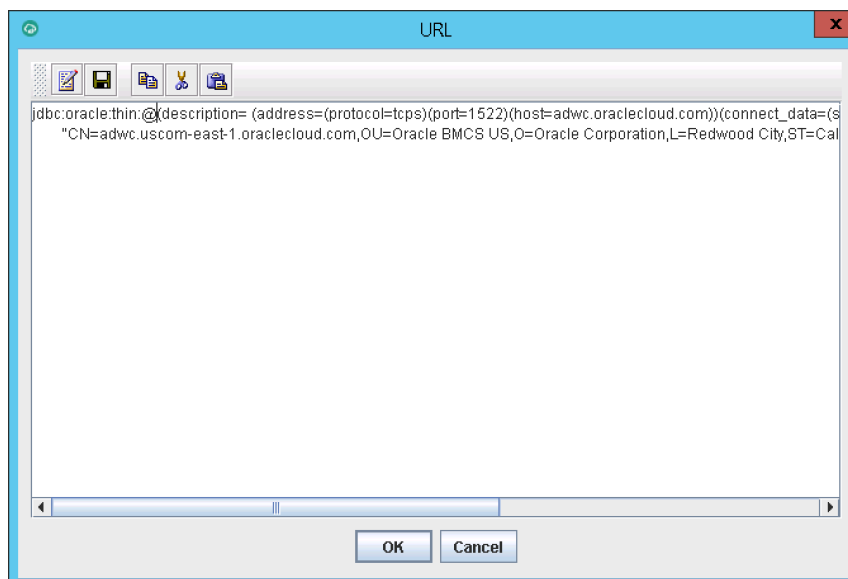
- We need to add the connect string you just copied to the URL text box in the **Edit** tab. Click on **URL** and enter the following in the pop-up:

```
jdbc:oracle:thin:@
```

- Next to the "@" sign, paste the contents that you copied earlier. Your constructed connect string should look like below.

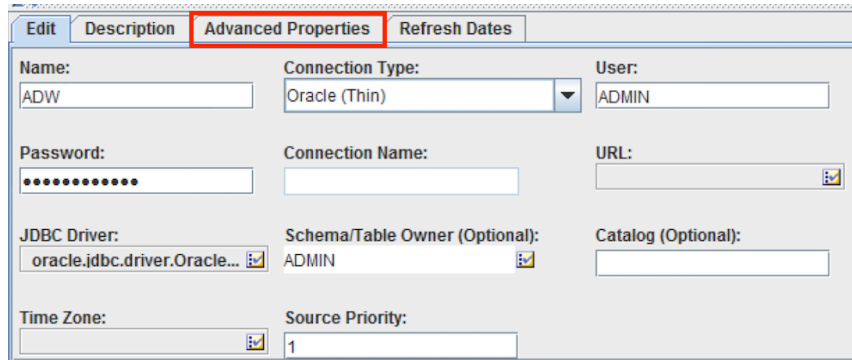
```
jdbc:oracle:thin:@(description= (address=(protocol=tcps)(port=1522)(host=adwc.oraclecloud.com))(connect_data=(service_name=partners_low.adwc.oraclecloud.com))(security=(ssl_server_cert_dn=
    "CN=adwc.uscom-east-1.oraclecloud.com,OU=Oracle BMCS US,O=Oracle Corporation,L=Redwood City,ST=California,C=US"))))
```

- Click **OK** to close the URL dialog.



- Since the ADWC service requires the use of Wallets, we need to add the location of Wallet Credentials Zip

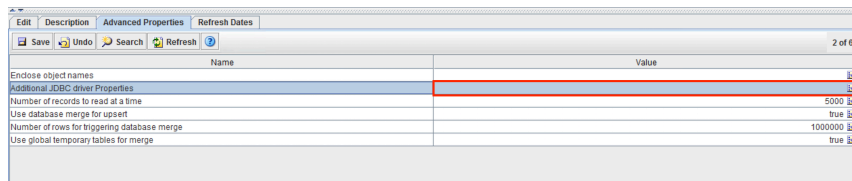
file to the connection. Click the **Advanced Properties** tab.



The screenshot shows a configuration window with the following fields:

- Name: ADW
- Connection Type: Oracle (Thin)
- User: ADMIN
- Password: [masked]
- Connection Name: [empty]
- URL: [empty]
- JDBC Driver: oracle.jdbc.driver.Oracle...
- Schema/Table Owner (Optional): ADMIN
- Catalog (Optional): [empty]
- Time Zone: [empty]
- Source Priority: 1

- Click on **Advanced JDBC driver Properties** as below:

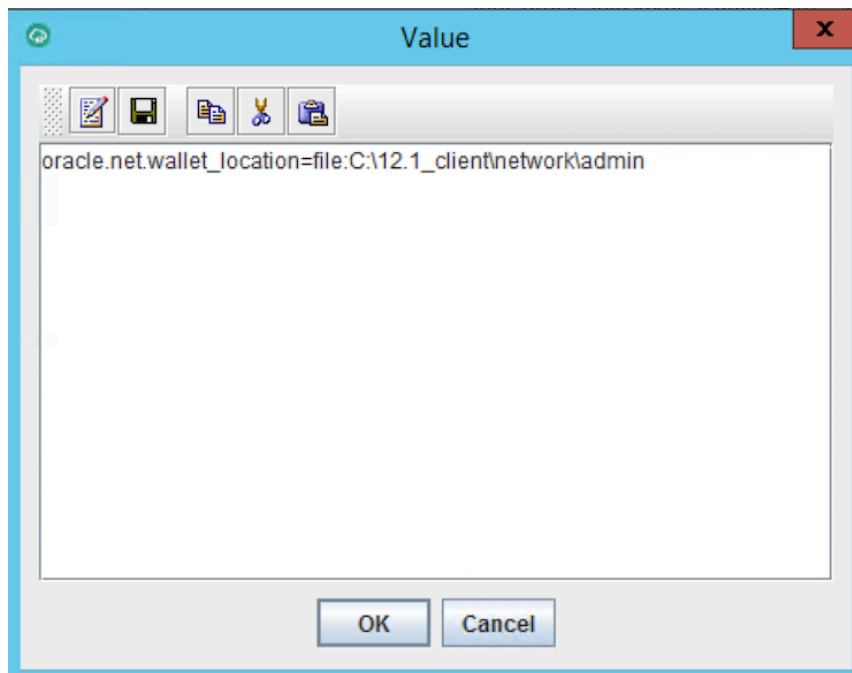


Name	Value
Enclose object names	
Additional JDBC driver Properties	
Number of records to read at a time	5000
Use database merge for update	true
Number of rows for triggering database merge	1000000
Use global temporary tables for merge	true

- In the pop-up window, enter the location of **cwallet.sso**, which in our case is %ORACLE_HOME%/network/admin (from Lab 2). Enter the following in the pop up box:

```
oracle.net.wallet_location=file:C:\12.1_client\network\admin
```

- Enter **OK**.

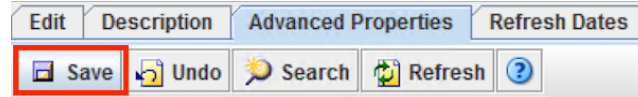


The dialog box titled 'Value' contains the following text:

```
oracle.net.wallet_location=file:C:\12.1_client\network\admin
```

Buttons: OK, Cancel

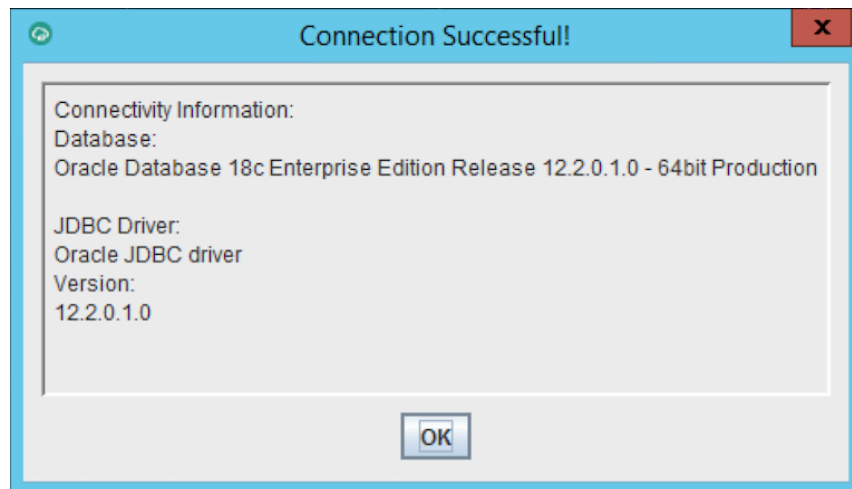
- Click **Save** to save the changes.



- Test the connection. Click on the **Edit** tab and click **Test Connection**.

A screenshot of a software interface showing a tabbed menu with 'Edit', 'Description', 'Advanced Properties', and 'Refresh Dates'. The 'Advanced Properties' tab is active. The form contains several fields: 'Name' (ADW), 'Connection Type' (Oracle (Thin)), 'User' (ADMIN), 'Password' (masked with dots), 'Connection Name' (empty), 'URL' (jdbc:oracle:thin:@(descri...)), 'JDBC Driver' (oracle.jdbc.driver.Oracle...), 'Schema/Table Owner (Optional)' (ADMIN), 'Catalog (Optional)' (empty), 'Time Zone' (empty), and 'Source Priority' (1). At the bottom, there are three buttons: 'Save', 'Test Connection', and 'Undo'.

- You should get a **Connection Successful!** message. Click **OK**.



Step 3: Create Connection to the Oracle Database (Source)

The **Source** Database used in this lab resides on an preconfigured **Oracle Database Cloud Service (DBCS)** instance in the Oracle Cloud. The schema used as the source for this lab has been prepopulated with sample data. The connection to the DBCS source has been preconfigured as **DBCS**.

- From the **Connections** tab, click on **DBCS** in the **Sources/Targets** section.

Name	Connection Type	User	Connection Name	URL
ADW	Oracle (Thin)	ADMIN		jdbc:oracle:thin:@(description= (a...
DBCS	Oracle (Thin)	USER_XX		jdbc:oracle:thin:@(DESCRIPTION...
File source	File source			
TARGET	Oracle (BICS)			

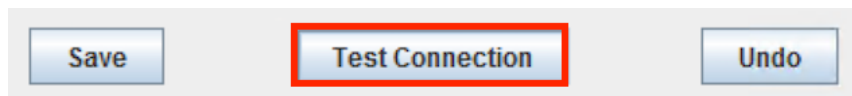
- Click the **Edit** tab towards the bottom half of the window and enter the following:
 - User: **User ID** (Supplied to you by the instructor)
 - Schema/Table Owner (Optional): **User ID** (same as above)
 - Do not change the password as the password has been saved already
- In the screenshot below, observe that the **User ID** used is *USER_02* (Replace this with your own).

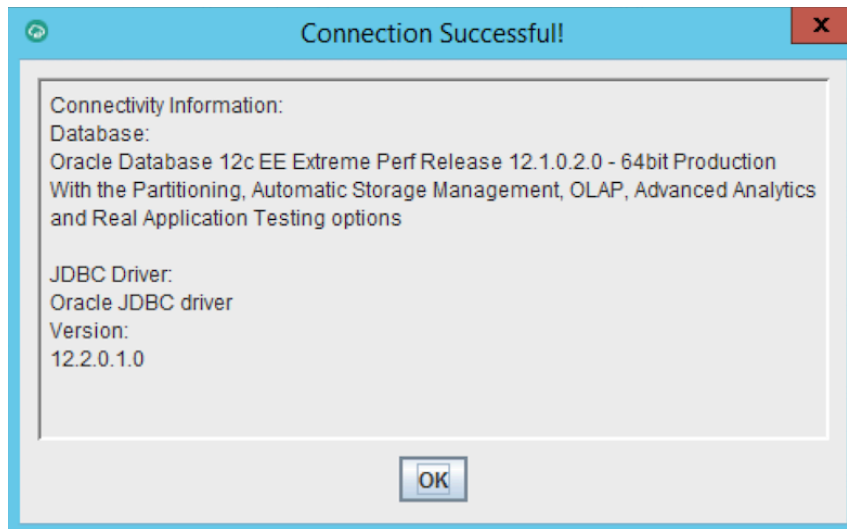
The screenshot shows the 'Edit' tab of the 'Sources/Targets' window. The 'Name' field contains 'DBCS'. The 'Connection Type' is 'Oracle (Thin)'. The 'User' field is highlighted with a red box and contains 'USER_02'. The 'Password' field is masked with dots. The 'Connection Name' and 'URL' fields are empty. The 'JDBC Driver' is 'oracle.jdbc.driver.Oracle...'. The 'Schema/Table Owner (Optional)' field is highlighted with a red box and contains 'USER_02'. The 'Catalog (Optional)' field is empty. The 'Time Zone' and 'Source Priority' fields are also empty. At the bottom, there are 'Save', 'Test Connection', and 'Undo' buttons.

- Click **Save**.



- Click **Test Connection** to validate your connection.

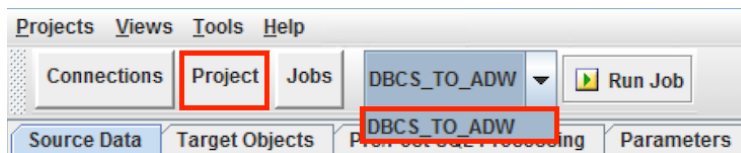




- You have now successfully configured the connections to both the Source DBCS and the Target ADWC.

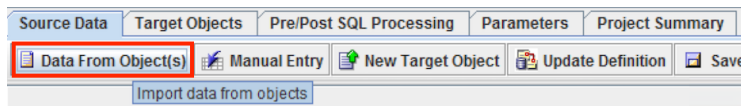
Step 4: Configure Data Load Options on the Source Tables

- First we need to add tables from the Source database. Select the **Project** tab and click on **DBCS_TO_ADW** project that was created in Step 1.

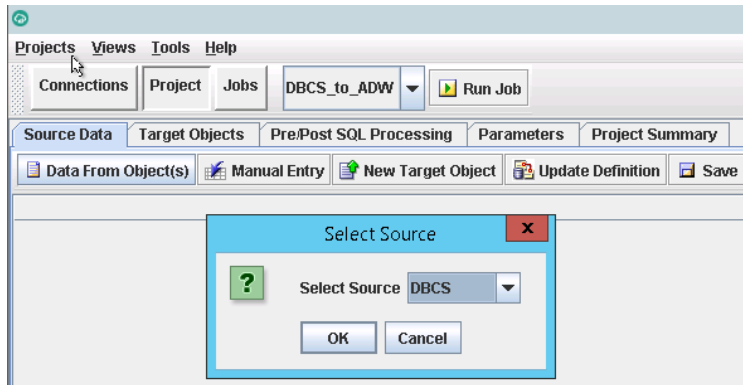


If you do not see the **DBCS_TO_ADW** Project, you need to create it by clicking on **Projects -> New**. Enter the details specified in **Step 1** of this Lab.

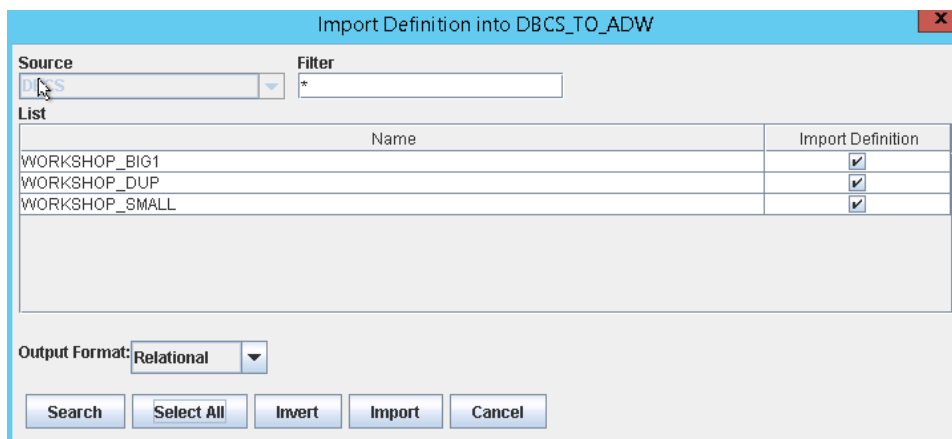
- Under **Source Data** click on **Data from Object(s)**.



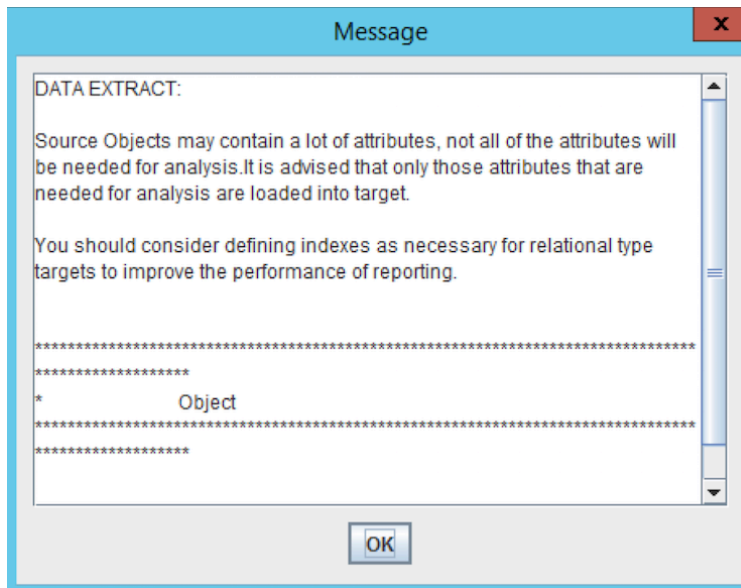
- In the pop-up box choose **DBCS** as the source. Click **OK**.



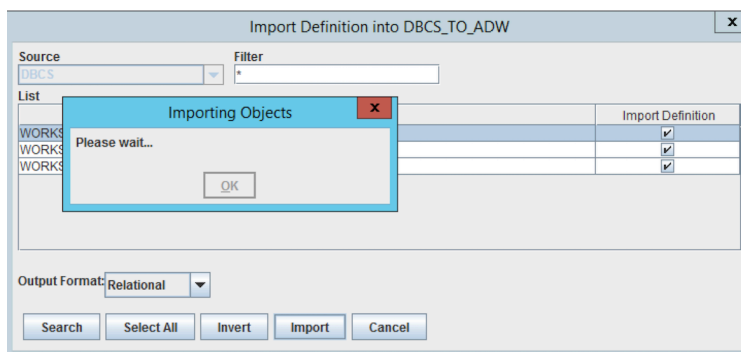
- Select **Discover objects by listing**. Click **OK**.
- In the **Import Definition** pop-up, click **Search** to get a listing of all tables in the Schema.
- All tables in the schema will be listed. Click **Select All** and then **Import**. This will import the table definitions to the Data Sync repository.



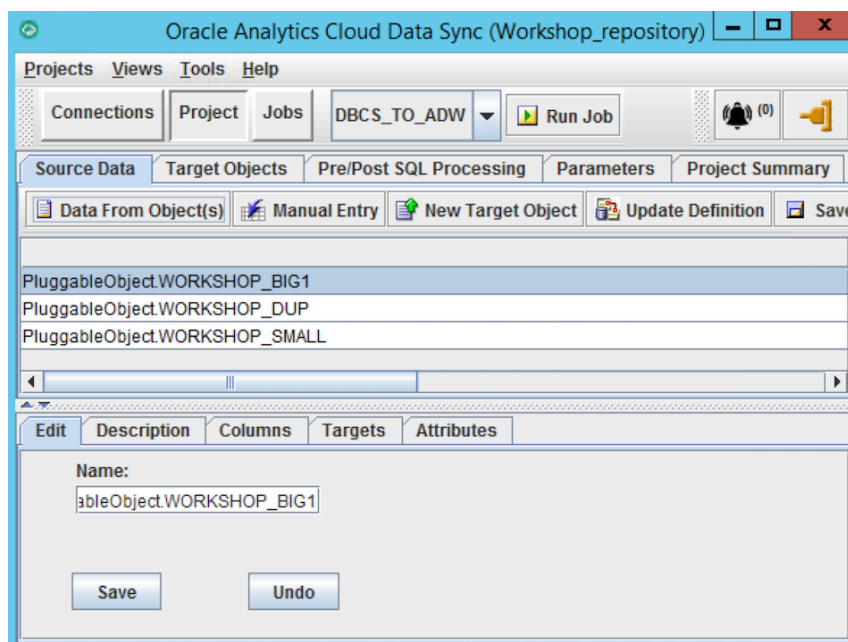
- Click **OK** when prompted with the below message:



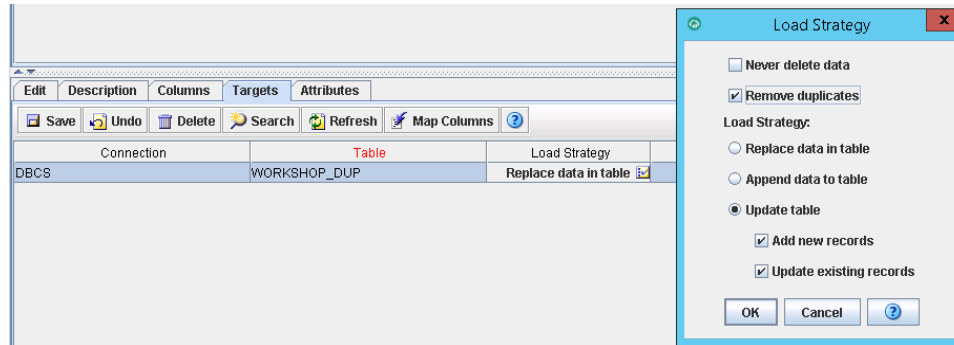
- Wait for the tables to be imported.



- All the 3 tables will appear in **Data From Object(s)**.



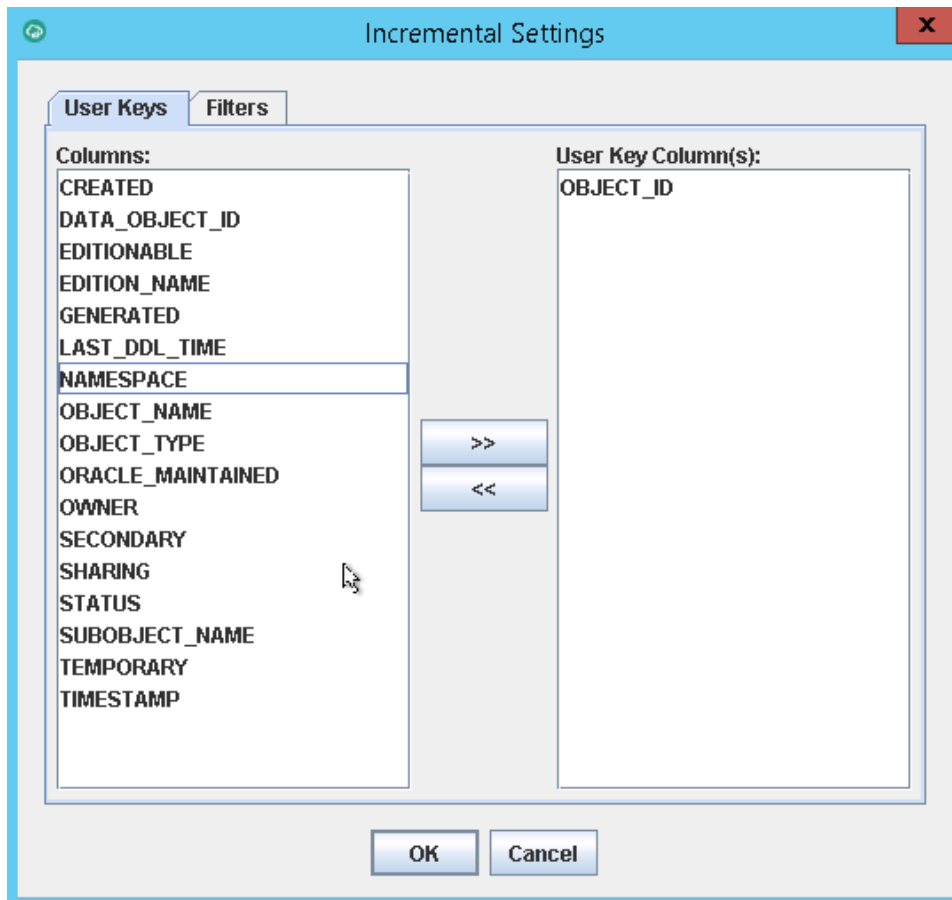
- Next, set the strategy for loading duplicate rows. Let the strategy be that we eliminate duplicate rows during the load and insert only unique rows based on **User Keys** column **OBJECT_ID**.
- Click on **WORKSHOP_DUP** table and then the **Target** tab in the lower half of the window.
- Double click on **Load Strategy**.
- Check **Remove duplicates**, **Update table**, **add new records** and **Update existing records** as show below:



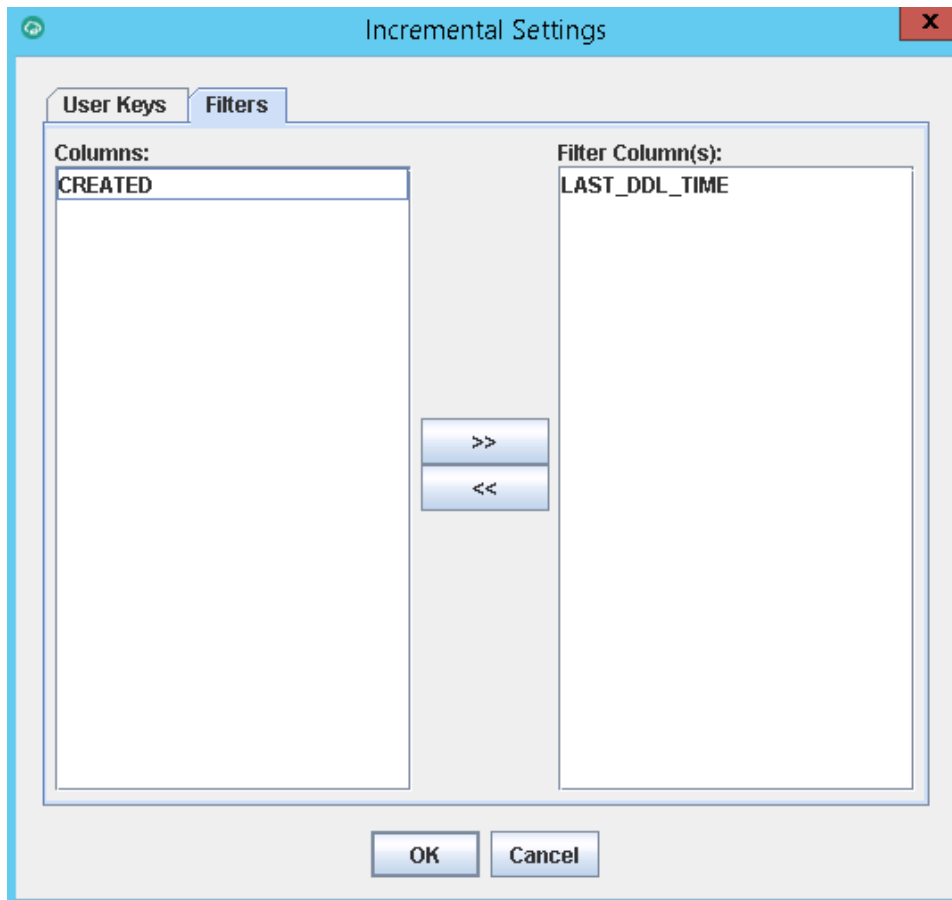
- The **Update table** option selected earlier allows incremental loading on the table.

Incremental loads are based on a date/timestamp filter column. **Data Sync** records the time of the previous run and filters on a date column to pick rows that are greater than the recorded time.

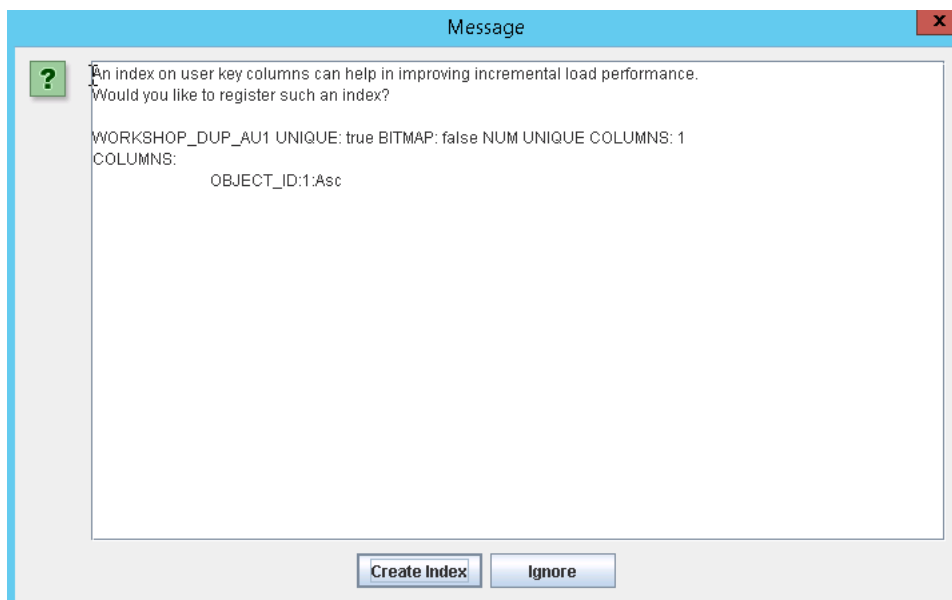
- Click **OK**.
- Configure the incremental load options on the **Incremental Settings** pop-up. Under the **User Keys** tab, choose **OBJECT_ID** and click ">>". **OBJECT_ID** will be the unique id based on which duplicate columns will be removed.



- Click the **Filters** tab, choose **LAST_DDL_TIME** and click **>>** . If rows are updated on the source table, Data from the table will move be loaded to Target if the **Filter** column is updated as well.



- Click **OK**.
- A pop-up will suggest that building additional indexes can improve performance of incremental loads. Click **Ignore** to ignore the suggestion for now.

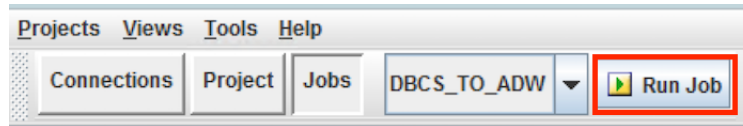


- Similarly, you can navigate to the other tables and investigate the load strategy.. By default all tables will

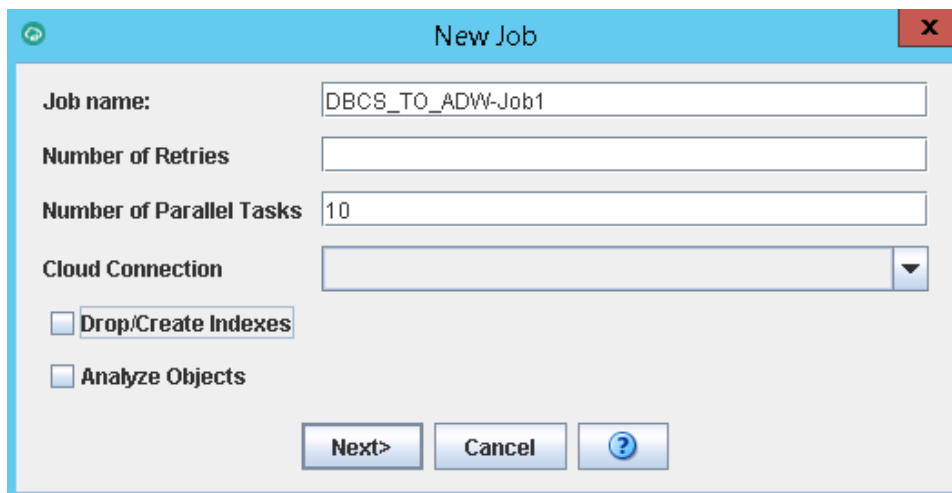
have a **replace data in table** option set.

Step 5: Run the Initial Load

- Create a new **Job** that will load the tables you have just defined. Click on **Run Job** at the top.



- Since this is the first Job we like to run, a **New Job** window will pop-up.
- Uncheck the **Drop/Create Indexes** checkbox. Checking this column will attempt to create a unique index to speed up de-duplication. Click **Next**.

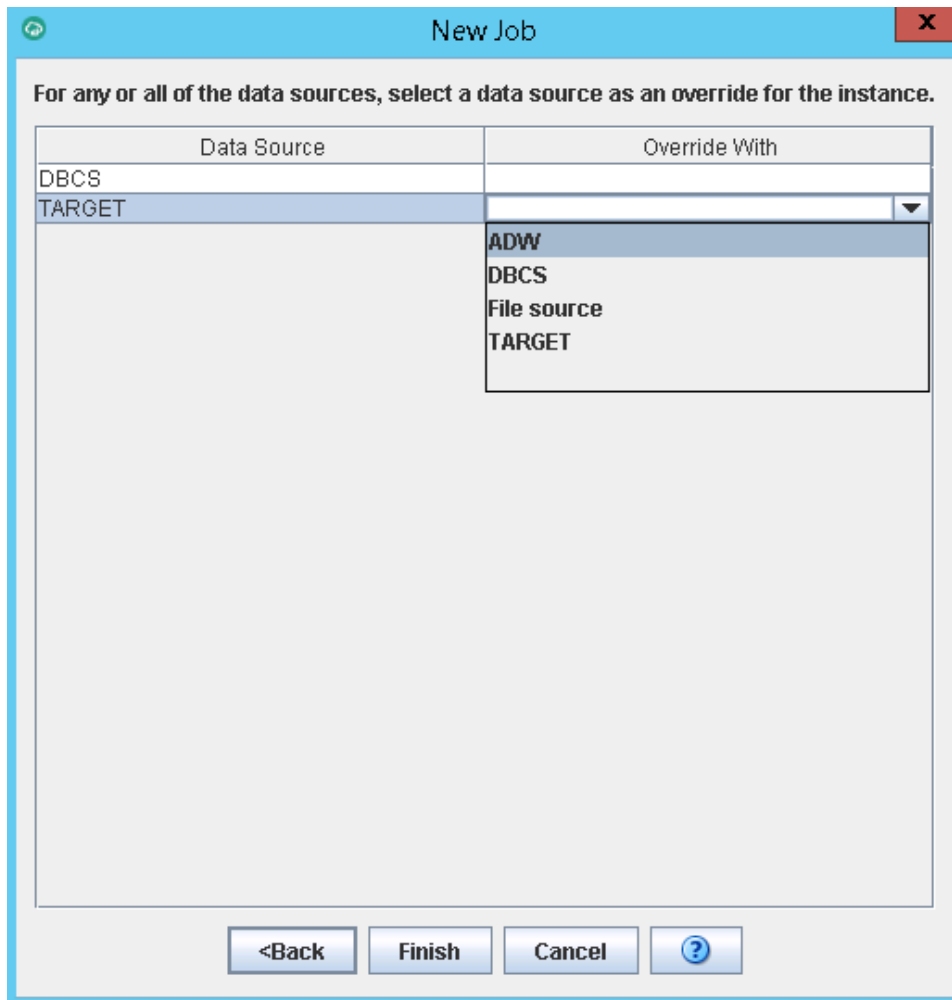
A screenshot of a 'New Job' dialog box. The title bar says 'New Job' with a close button (X). The dialog contains the following fields and options:

- Job name:** DBCS_TO_ADW-Job1
- Number of Retries:** (empty text box)
- Number of Parallel Tasks:** 10
- Cloud Connection:** (dropdown menu)
- Drop/Create Indexes**
- Analyze Objects**

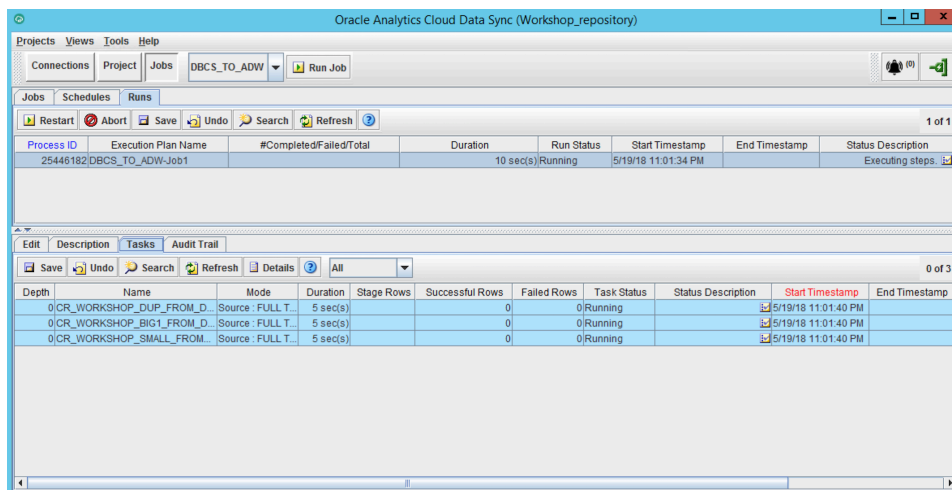
At the bottom, there are three buttons: 'Next>', 'Cancel', and a help button (question mark icon).

IMPORTANT If you forget to uncheck this option, Data Sync will try to create a unique index in Target ADWC to speedup incremental loading. However since ADWC currently does not support the creation of indexes, You will see an error in the **Job Details**

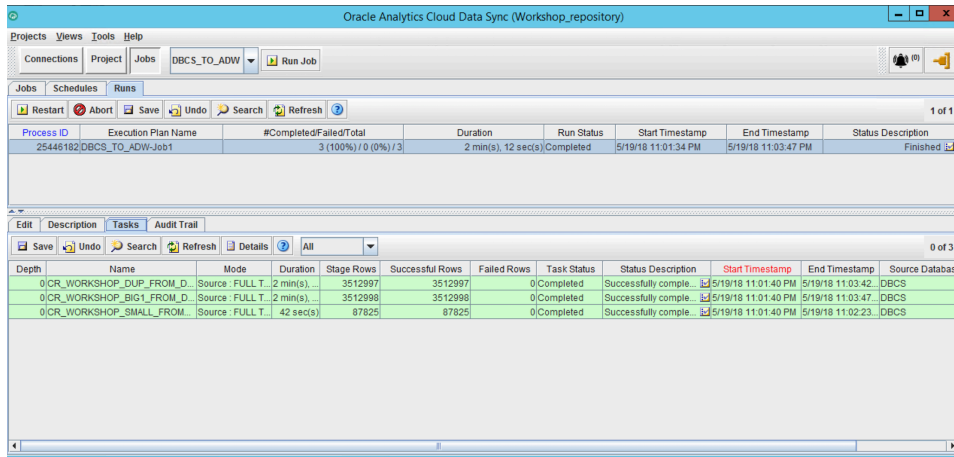
- For the **Target** datasource, choose your **ADW** connection. Click **Finish**.



- Your first job is being compiled and run.



- In a few seconds, your Job will complete and you will observe the following three tasks in the lower **Tasks** tab:



- You have now successfully loaded the 3 tables.
 - **WORKSHOP_BIG1** has about 3.5 million rows and is about 500 MB in size.
 - **WORKSHOP_SMALL** is a small table with about 87K rows.
 - **WORKSHOP_DUP** has about 3.5 million rows in the source, but only about 87K rows in target as we had opted to eliminate duplicate rows.

Depth	Name	Mode	Duration	Successful Rows
0	CR_WORKSHOP_BIG1_FROM_DBCS-Pluggab...	Source : FULL Target : FULL	2 min(s), 51 ...	3510278:
0	CR_WORKSHOP_SMALL_FROM_DBCS-Pluggab...	Source : FULL Target : FULL	29 sec(s)	87757:
0	CR_WORKSHOP_DUP_FROM_DBCS-Pluggab...	Source : FULL Target : FULL	1 min(s), 41 ...	87757:

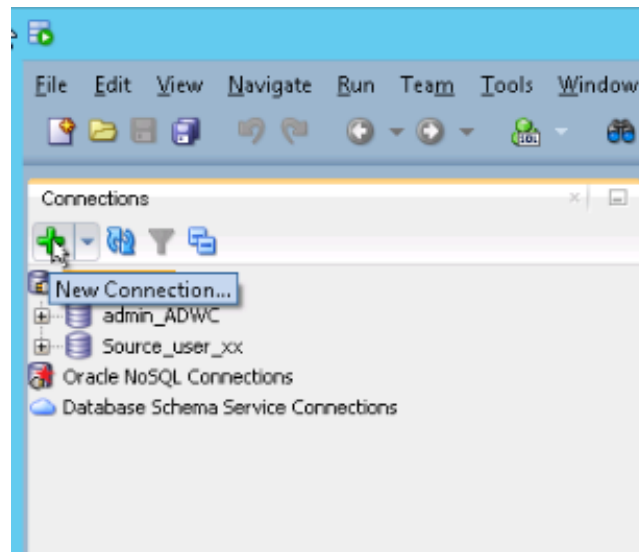
A single project can have multiple jobs. Usually, you would configure a single project to load a single Target, but the Sources can be multiple.

Step 6: Run the Incremental Load

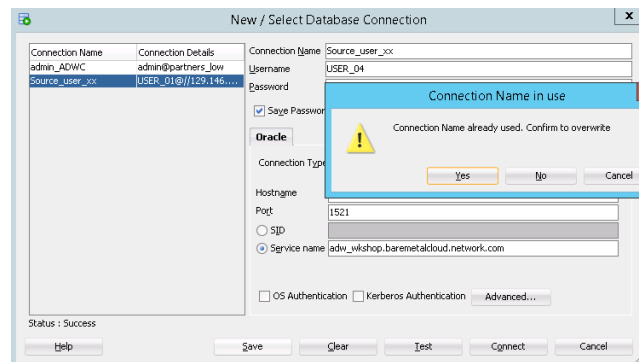
One of the use cases for **Data Sync** is to keep the ADWC service in sync with on-prem or other cloud services.

In the previous step, the **Load Strategy** for **WORKSHOP_DUP** table was set to support incremental loads. We will load additional rows on the **WORKSHOP_DUP** table and use Data Sync to replicate the new rows to ADWC.

- Start **Oracle SQL Developer** and Click on **New Connection** to create a new Connection to the Oracle Database.



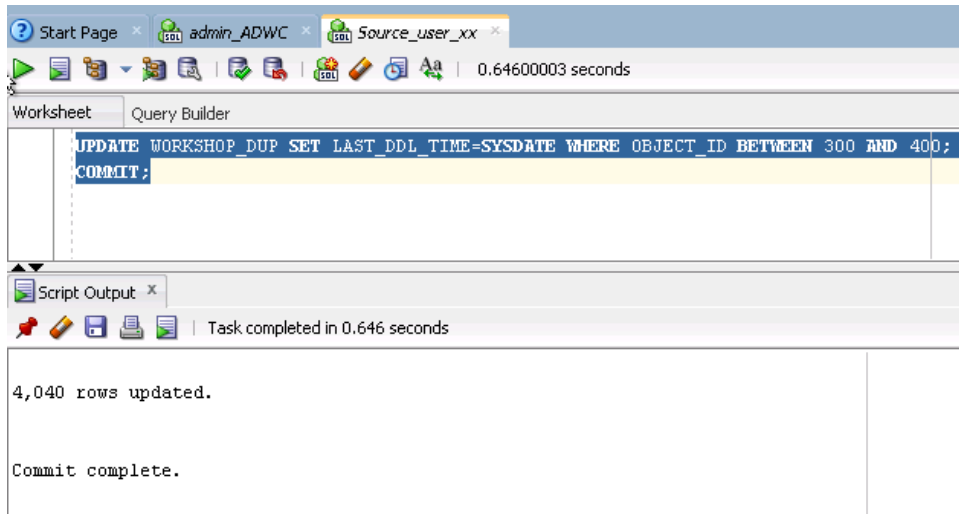
- In the **New/Select Database Connection** pop-up click on the previously saved **Source_User_XX** connection in the left pane.
- Change the username from USER_XX to your assigned User name. Keep the same password.
- Click **Test**. If the connection is successful, **Status : Success** will appear on the lower left of the pop-up.
- Click **Save** and confirm **Yes** to overwrite the connection.



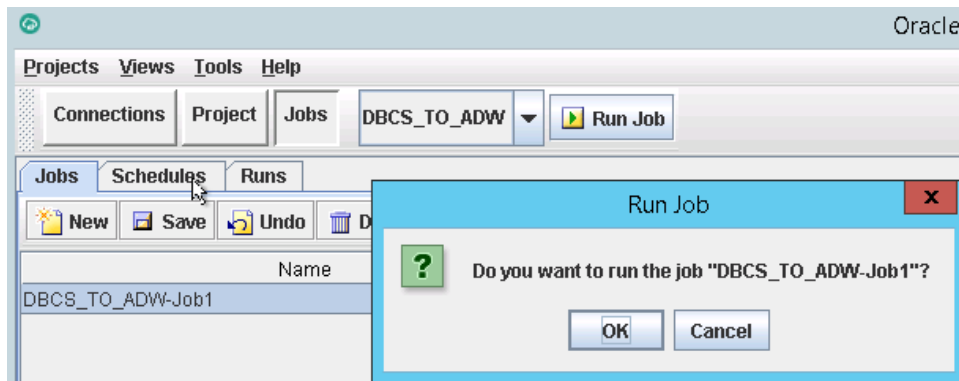
- Next, Click **Connect** and new **SQL Worksheet** will open.
- Run the script below to update and commit the rows.

```
UPDATE WORKSHOP_DUP
SET LAST_DDL_TIME=SYSDATE
WHERE OBJECT_ID BETWEEN 300 AND 400;

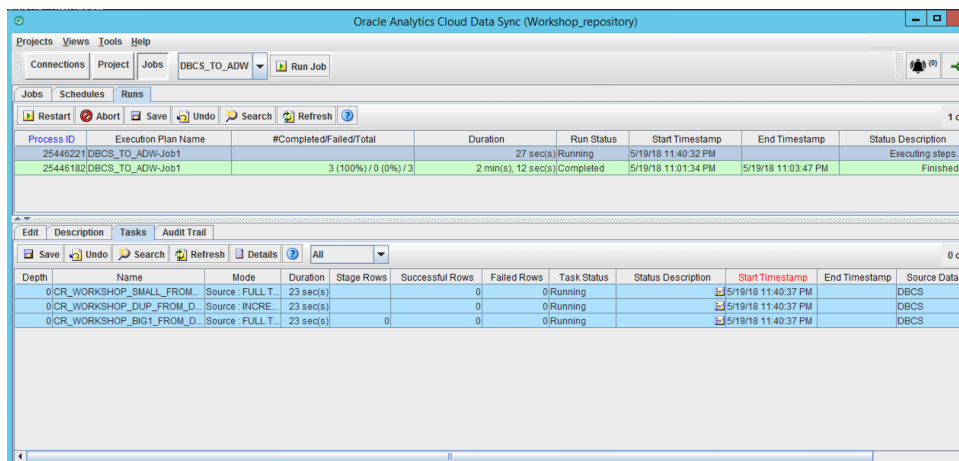
COMMIT;
```



- We have now simulated some incremental data change in the source. The source table has about 3.5 million rows. The update modifies about 4K rows.
- Now switch to **Data Sync** and click on **Jobs** then on **Run Job**.
- Click **OK** on **Run Job** pop-up.



- Wait for your Job to Complete.



- Ensure that the **Tasks** tab is selected on the lower half of the window and observe 4040 rows were updated in the **Incremental** mode. The other two tables were appended with additional rows as the incremental settings were not in effect.

The screenshot shows the Oracle Analytics Cloud Data Sync interface. The top section displays a job run summary for 'DBCS_TO_ADW'. The bottom section shows a table of tasks with the following data:

Depth	Name	Mode	Duration	Stage Rows	Successful Rows	Failed Rows	Task Status	Status Description	Start Timestamp	End Timestamp	Source Database
0	CR_WORKSHOP_SMALL_FROM_DBCS-Plu...	Source: FULL Target: FULL	34 sec(s)	87825	87825	0	Completed	Successfully comple...	5/19/18 11:40:37 PM	5/19/18 11:41:11...	DBCS
0	CR_WORKSHOP_DUP_FROM_DBCS-Plu...	Source: INCREMENTAL Target: INC...	33 sec(s)	4040	4040	0	Completed	Successfully comple...	5/19/18 11:40:37 PM	5/19/18 11:41:10...	DBCS
0	CR_WORKSHOP_BIG1_FROM_DBCS-Plu...	Source: FULL Target: FULL	1 min(s), ...	3512998	3512998	0	Completed	Successfully comple...	5/19/18 11:40:37 PM	5/19/18 11:42:04...	DBCS

- Click on the row that was loaded with **Incremental Target Mode** and click on **Details**.

The screenshot shows the Oracle Analytics Cloud Data Sync interface with the 'Details' tab selected for the task '0CR_WORKSHOP_DUP_FROM_DBCS-Plu...'. The 'Details' tab shows the following information:

Depth	Name	Show details	Mode	Duration	Stage Rows	Successful Rows	Failed Rows	Task Status	Status Description
0	CR_WORKSHOP_SMALL_FROM_DBCS-Plu...		Source: FULL Target: FULL	34 sec(s)	87825	87825	0	Completed	Successfully comple...
0	CR_WORKSHOP_DUP_FROM_DBCS-Plu...		Source: INCREMENTAL Target: INC...	33 sec(s)	4040	4040	0	Completed	Successfully comple...
0	CR_WORKSHOP_BIG1_FROM_DBCS-Plu...		Source: FULL Target: FULL	1 min(s), ...	3512998	3512998	0	Completed	Successfully comple...

- Observe other details on the Import Job, including **Duration**, **Throughput** and **Errors**.

The screenshot shows a window titled "Details: CR_WORKSHOP_DUP_FROM_DBCS-PluggableObject.WORKSHOP_DUP". It contains a "Task Details" tab and an "Audit Trail (RO)" section. Below this is a toolbar with "Save", "Undo", "Search", "Refresh", and a dropdown menu set to "All". A status indicator shows "0 of 1". The main area is a table with the following data:

Sequence	Name	Task Run Type	Status	Stage Rows	Successful Rows	Failed Rows	Duration	Read Through
1	INSERT_UPDATE_DATA...	Data Copy	Completed	4040	4040	0	3 sec(s)	

- You have successfully moved data from source DB to ADWC and configured the Job to do incremental loads to ADWC.