Project Documentation: Oracle to MSSQL Data Transfer Pipeline

Project Overview:

This document provides comprehensive instructions for setting up Mage, configuring the Oracle to MSSQL Data Transfer Pipeline, and running the pipeline to migrate data from an Oracle database to a Microsoft SQL Server (MSSQL) database. It covers the installation steps for Mage, the pipeline setup, and the execution process.

1. Pre-request:

To run this pipeline and must install ODBC Driver for MS SQL server.

You can do it by following below steps:

In terminal

sudo yum update

sudo yum install unixODBC

Download the ODBC driver package for Linux from the Microsoft website and Make sure to download the appropriate version (32-bit or 64-bit) based on your CentOS installation.

curl https://packages.microsoft.com/config/rhel/7/prod.repo | sudo tee /etc/yum.repos.d/mssql-release.repo

sudo yum remove unixODBC-utf16 unixODBC-utf16-devel

sudo ACCEPT\_EULA=Y yum install -y msodbcsql18

1. Setting Up Mage:

1.1. Open the terminal and execute the following command:

```bash

git clone <https://github.com/mage-ai/compose-quickstart.git> mage-quickstart \&& cd mage-quickstart \&& cp dev.env .env && rm dev.env \

&& docker compose up

```

1.2. After the server starts running, navigate to [http://localhost:6790/pipelines?\_limit=30 HYPERLINK "http://localhost:6790/pipelines?\_limit=30&tab=all"& HYPERLINK "http://localhost:6790/pipelines?\_limit=30&tab=all"tab=all](http://localhost:6790/pipelines?_limit=30&tab=all).

\*port number(6790) is not unique.

Double click "active" or "inactive" of oracle\_to\_sql section in etl-demo>Pipelines.

2. Setting Up the Oracle to MSSQL Data Transfer Pipeline:

* 1. Pipeline Name:

Table migration pipeline :oracle\_to\_sql

View migration pipeline : oracle\_view

* 1. Pipeline Components:

2.2.1. oracle\_to\_sql

- `load\_oracle`: Data loader component responsible for loading data from the Oracle database.

- `oracle\_data\_exporter`: Data exporter component for exporting the loaded data to the MSSQL database.

2.2.2. orcle\_view

- `oracle\_view`: Data loader component responsible for loading view text from the Oracle database.

- `oracle\_view\_exporter`: Data exporter component for creating the view to the MSSQL database.

2.3. Update io\_config.yaml file with MSSQL Server Credentials:

```io\_config.yaml

# Ms\_SQL

MSSQL\_DATABASE: Database Name that you want to migrate

MSSQL\_SCHEMA: Schema name in Database

MSSQL\_DRIVER: ODBC Driver 18 for SQL Server(if you install ODBC for SQL Server by using above command, write this directly)

MSSQL\_HOST: SQL Server Name is hosting

MSSQL\_PASSWORD: your password

MSSQL\_PORT: 1433

MSSQL\_USER: SA

```

2.4. Migrating the any data from oracle to your mssql database.

2.4.1 Table migrationning

- Open the `load\_oracle` data loader component.

- Locate the `owner\_name` variable and update it with the desired owner of the table to be moved.

2.4.2 View migrationning

- Open the `load\_view` view text loader component.

- Locate the `owner\_name` variable and update it with the desired owner of the view to be moved.

3. Running the Pipeline:

3.1. Load Data from Oracle:

3.1.1 Table structure loading

- Run the `load\_oracle` data loader to load data from the Oracle database.

3.1.2 Load view text from oracle

- Run the `load\_view` data loader to load text of view from the Oracle database.

3.2. Export Data to MSSQL:

3.2.1 Table Export

- Run the `oracle\_data\_exporter` component to export the loaded data to the MSSQL database.

3.2.2

- In case view migration, you must edit the SQL query of view according to text of oracle view.

This work contains that make the “migration\_view.txt” file manually edit and verifying then place this file on “/etl-demo/etl-demo/table” folder.

- Run the `oracle\_view\_exporter` component to create the view to the MSSQL database.

1. Importing input files:

-Place the input files to “etl-demo/etl-demo/db-data” folder

1. Where to configure SQL server and Oracle connection properties.

As I mentioned front section, when you set the io\_config.yarml file, connection from SQL to Oracle is created by interaction data\_loader and data\_exporter.

1. Pipeline Setting

5.1 Readying to Data

5.1.1 Setting in case of Table migration

You must get the table structure according to text file that must be migration to database.

To do this you must change the value of "owner\_name" that you want to migrate owner of tables in load\_oracle loader.

- Place the files that contains all tables for the views on “etl-demo/etl-demo/db-data” directory

If you place the input data to this directory, you need nothing that you must do another.

5.1.2 Setting in case View Migration

-You must change the value of "owner\_name" that you want to migrate owner of views in load\_view loader.

- Then After run load\_view and get the text of view.

- And you must convert the text to SQL query of view manually and verify the logic.

- Place the file that contains all queries for the views on “etl-demo/etl-demo/table” directory

-Then you must change the value of file\_name in oracle\_view\_exporter.

- In example, You can see the file\_name variable in oracle\_view\_exporter same following  
 file\_name = "migration\_view.txt"

5.2. Exporting data

5.2.1 Table exporting

Run the “oracle\_data\_export”

5.2.2 View exporting

Run the “oracle\_view\_exporter”

I think you don’t understand well about it but you can invite me when you need my helping.  
Thank you.

if you set this values in correctly, you can migrate the data by running the exporter.