

Install Guide - SetupWizard - SQL Server -RED 10.3+

WhereScape Enablement Pack for SQL Server - RED 10.3+

This is a guide to installing the WhereScape Enablement Pack for SQL Server for WhereScape RED10

Table of Contents

[Prerequisites For PostgreSQL Metadata](#)
[Prerequisites Target Database](#)
[Installation Through Setup Wizard](#)
[Upgrade Of Existing Repository](#)
[Setting Extended Properties for BCP Load for SQL Server as source:](#)
[SSIS - Connection String Generation](#)
[Post Install Steps – Optional](#)
[Source Enablement Pack Support](#)
[Troubleshooting and Tips](#)

Prerequisites For PostgreSQL Metadata

Before you begin the following prerequisites must be met:

- **Create Database and ODBC DSN :**
 - Supported* version of PostgreSQL (PostgreSQL 12 or higher)
 - A database to house the RED Metadata Repository.
 - A database for the Range Table DB (Optional)
 - A database to house scheduler (Optional)
- **Software Installations**
 - WhereScape RED10 with valid license key entered and EULA accepted
 - WhereScape Enablement Pack for target database version RED10
- **Windows Powershell (64 bit) version 4 or higher**
 - To check Windows Powershell Version:
 - Run below command in Windows Powershell

```
1 Get-Host|Select-Object Version
```

- Run below command in Command Prompt

```
1 powershell $psversiontable
```

- **Run the following command using PowerShell**

- The security protocol TLS 1.0 and 1.1 used by PowerShell to communicate with PowerShell gallery has deprecated and TLS 1.2 has been made mandatory

```
1 [Net.ServicePointManager]::SecurityProtocol = [Net.ServicePointManager]::SecurityProtocol -bor [Net.SecurityProtocolType]::Tls12
```

```
1 Register-PSRepository -Default -Verbose
```

```
1 Set-PSRepository -Name PSGallery -InstallationPolicy Trusted
```

- **Progress bar placeholder info line**

```
1 | Install-Module -Name PoshProgressBar -SkipPublisherCheck -Force
```

* : RED supports the following versions for the metadata repository: PostgreSQL 12 or higher

Prerequisites Target Database

- Access to an SQL Server (Target Environment), with the following connectivity information :

1. Server Name
2. Database Name
3. User Name
4. Password
5. At least one schema available to use as a RED Data Warehouse Target

- SQL Server software installed
 - SQL Server ODBC driver (64-bit)
 - i. Add ODBC DSN for SQL Server Target Database

- **SSIS Runtime Prerequisite**

- SQL Server Integration Services (SSIS) runtime must be installed.
- The SSIS **Text File Log Provider** must be available and enabled.
- This load relies on package-defined SSIS logging and will not generate log files if the required SSIS components are missing.

Note: If SSIS was installed without logging components, this feature must be enabled through SQL Server setup modification.

Before you begin the following prerequisites must be met:

- [Python 3.11 or higher](#)
 - Select "Add Python 3.11 to PATH" from installation Window
 - Pip Manager Install with command : `python -m pip install --upgrade pip`

Note: Some of the libraries might be deprecated in the newer versions of python (Python 3.13 and higher)

- To install Python modules offline for supported python versions - [Offline Python Package Installation Guide - Enablement Packs - Product Documentation](#)
- The Setup_Validator script included with the WhereScape Enablement Pack (EP). The script is a standalone module that verifies your system meets all prerequisites before the Enablement Pack installation begins. This script is supported for Red10.* versions only

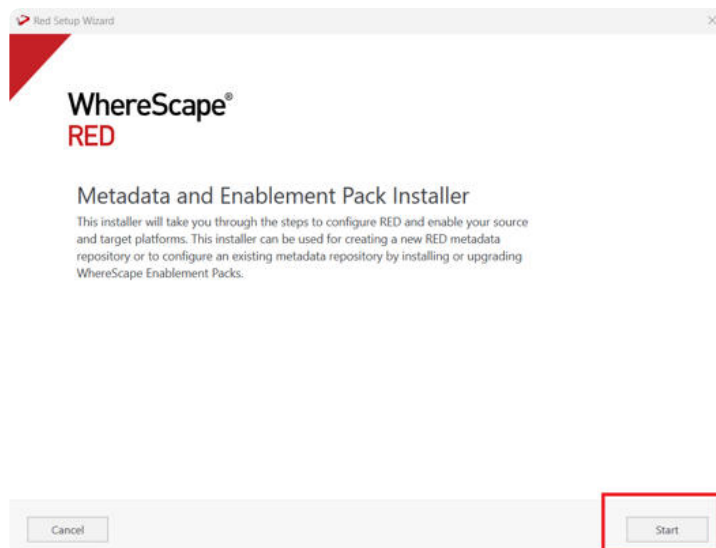
To Execute Setup_Validator Script - [Guide for Setup_Validator Script Execution - Enablement Packs - Product Documentation](#)

- This SQL Server target EP supports both RED 10 and Wherescape 3D

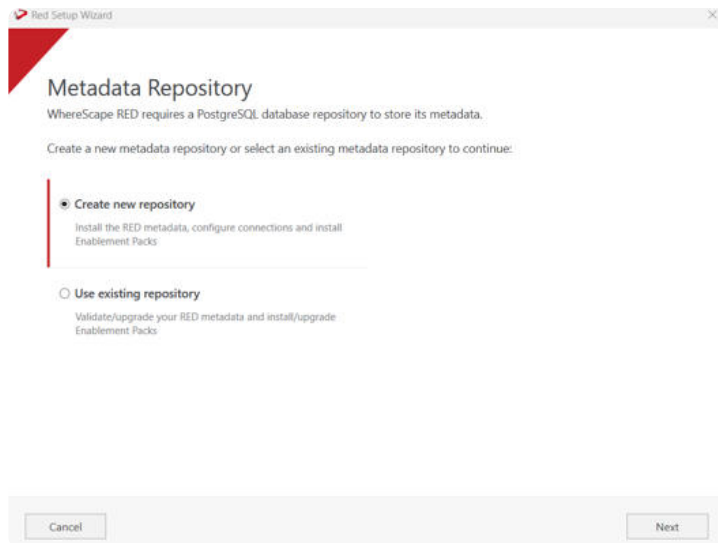
To install the enablement pack on Wherescape 3D - [WhereScape Enablement Pack for SQL Server 3D - Enablement Packs - Product Documentation](#)

Installation Through Setup Wizard

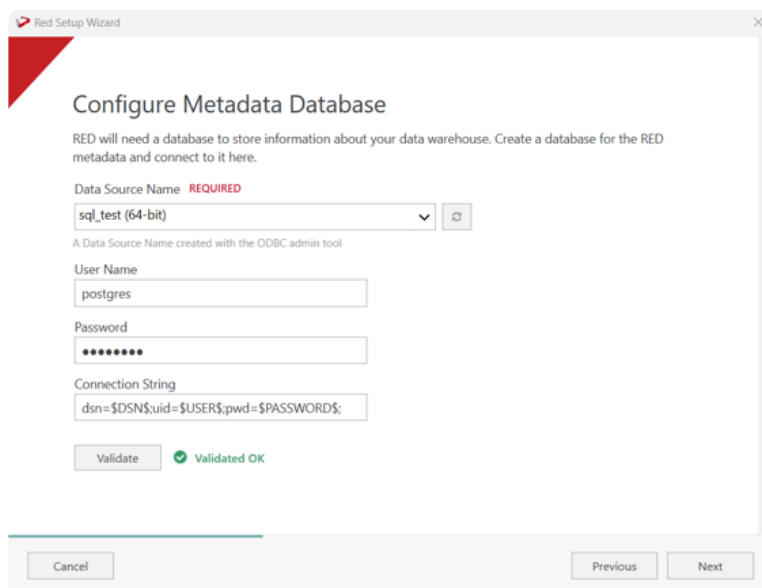
Run Setup Wizard as administrator



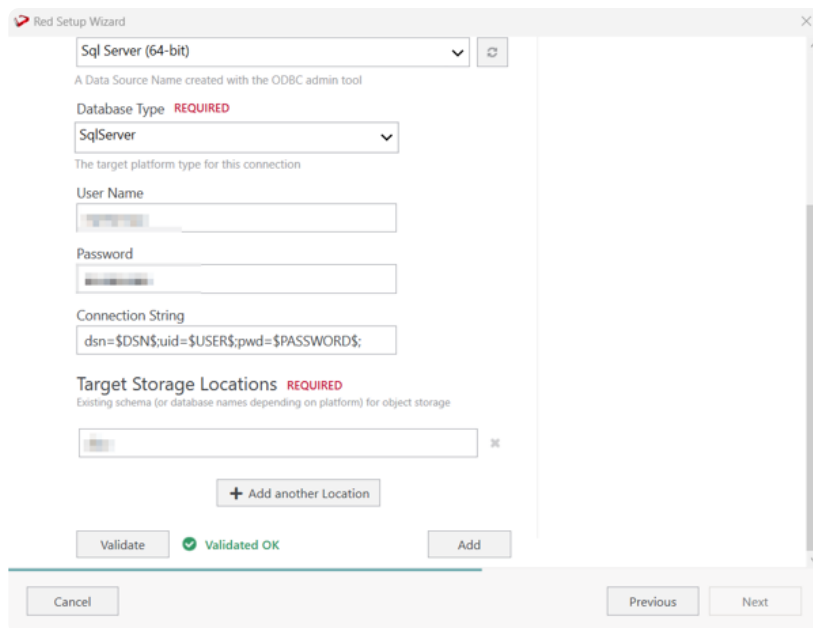
Create new repository or upgrade already existing repository.



Select the created ODBC DSN, input login details and then select "Validate". Press Next



Select the directory that contains unzipped Enablement Pack for installation. Press Next



Red Setup Wizard

Sql Server (64-bit)

A Data Source Name created with the ODBC admin tool

Database Type **REQUIRED**

SqlServer

The target platform type for this connection

User Name

Password

Connection String

dsn=\$DSN;uid=\$USER;pwd=\$PASSWORD;

Target Storage Locations **REQUIRED**

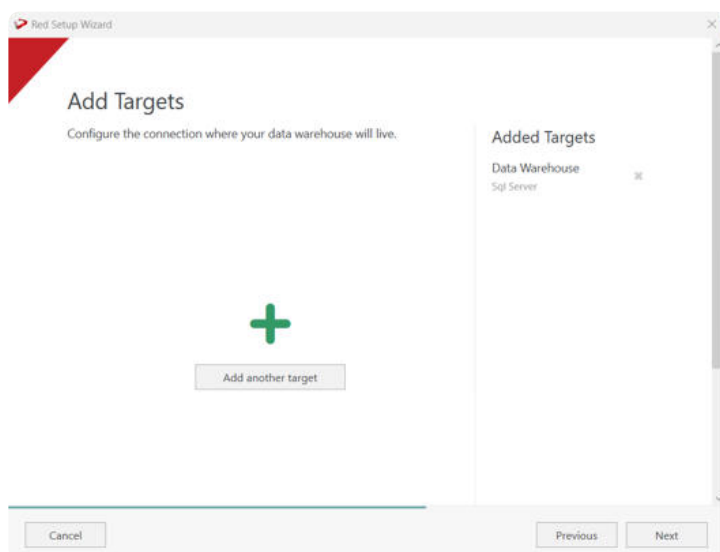
Existing schema (or database names depending on platform) for object storage

+ Add another Location

Validate Validated OK Add

Cancel Previous Next

When done, press ADD and then Press Next to advance.



Red Setup Wizard

Add Targets

Configure the connection where your data warehouse will live.

+ Add another target

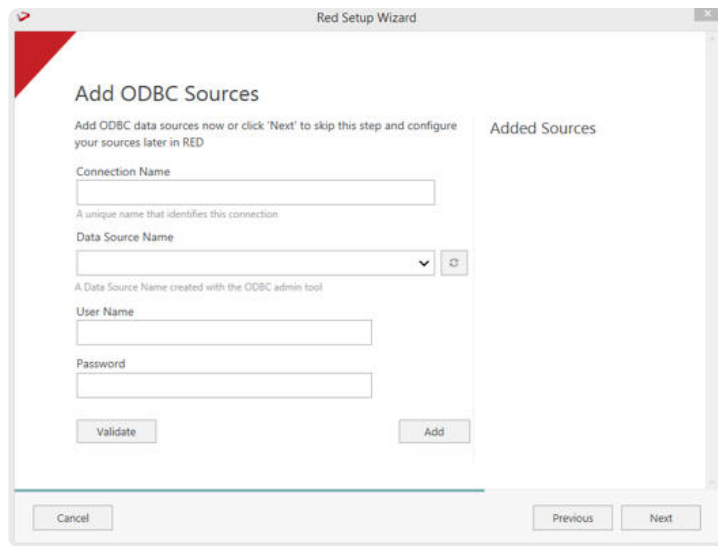
Added Targets

Data Warehouse

Sql Server

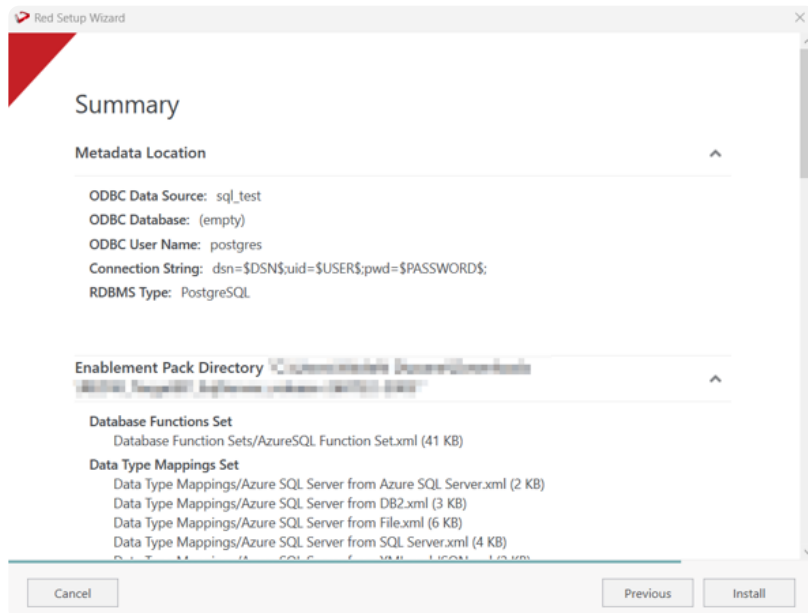
Cancel Previous Next

Configure a data source connection (optional) and its target locations. Validate and press ADD. Press Next to advance.



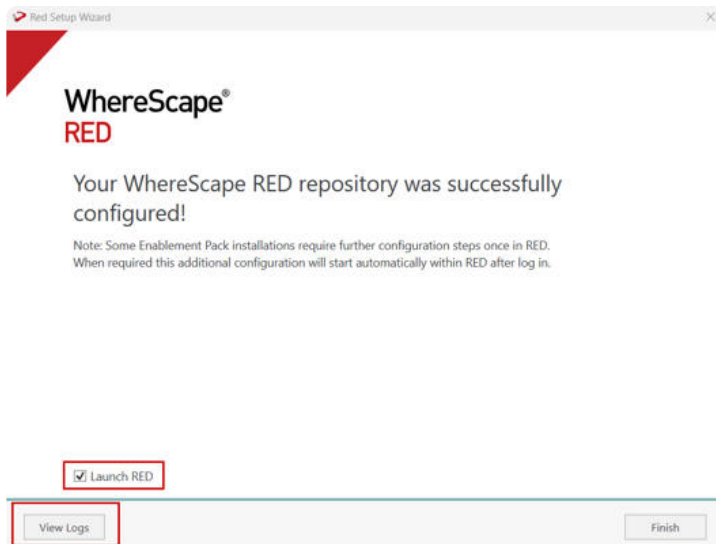
The screenshot shows the 'Add ODBC Sources' step of the Red Setup Wizard. The window has a title bar 'Red Setup Wizard' and a red logo in the top-left corner. The main area is divided into two sections. The left section, titled 'Add ODBC Sources', contains instructions: 'Add ODBC data sources now or click \'Next\' to skip this step and configure your sources later in RED'. Below this are input fields for 'Connection Name' (with a description: 'A unique name that identifies this connection'), 'Data Source Name' (with a dropdown arrow and a plus icon, and a description: 'A Data Source Name created with the ODBC admin tool'), 'User Name', and 'Password'. At the bottom of this section are 'Validate' and 'Add' buttons. The right section, titled 'Added Sources', is currently empty. At the bottom of the window are 'Cancel', 'Previous', and 'Next' buttons.

Review the installation summary and click Install

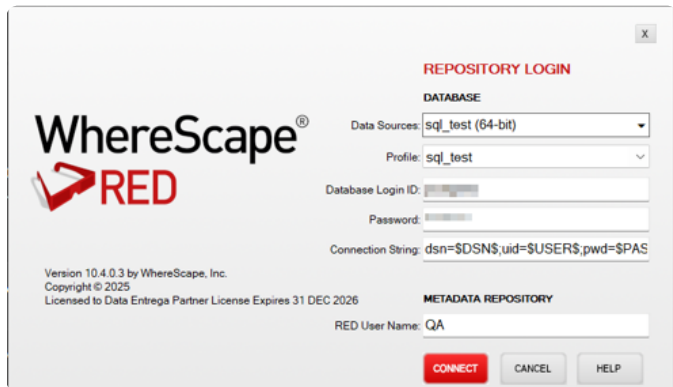


The screenshot shows the 'Summary' step of the Red Setup Wizard. The window has a title bar 'Red Setup Wizard' and a red logo in the top-left corner. The main area is titled 'Summary' and contains two expandable sections. The first section, 'Metadata Location', is expanded and shows: 'ODBC Data Source: sql_test', 'ODBC Database: (empty)', 'ODBC User Name: postgres', 'Connection String: dsn=\$DSN;uid=\$USER;pwd=\$PASSWORD;', and 'RDBMS Type: PostgreSQL'. The second section, 'Enablement Pack Directory', is also expanded and shows a file path. Below these sections are three sub-sections: 'Database Functions Set' (Database Function Sets/AzureSQL Function Set.xml (41 KB)), 'Data Type Mappings Set' (Data Type Mappings/Azure SQL Server from Azure SQL Server.xml (2 KB), Data Type Mappings/Azure SQL Server from DB2.xml (3 KB), Data Type Mappings/Azure SQL Server from File.xml (6 KB), and Data Type Mappings/Azure SQL Server from SQL Server.xml (4 KB)). At the bottom of the window are 'Cancel', 'Previous', and 'Install' buttons.

Clicking on the View Logs will take to the installation log. Click on Finish once the installation is completed successfully.

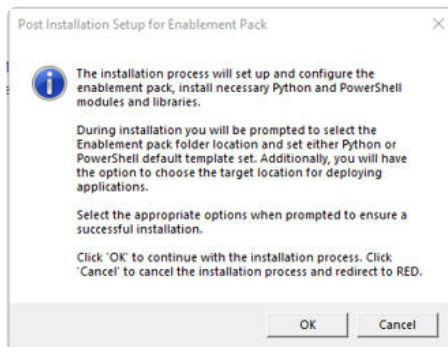


Login to WhereScape RED.



Note: There is a post-install script that will run at the first login to RED10 to complete the post setup wizard installation process.

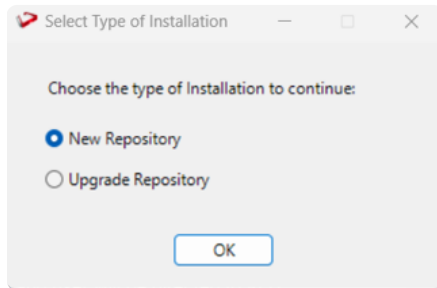
You will be directed to below PowerShell window which will give brief explanation about post installation process.



Press OK to start the post installation. If pressed Cancel installation will stop and user will be directed to RED.

The user will be directed to the window below, where they have to select the type of installation

Select “New Repository “for new repository Installation or Select “Upgrade Repository” option for upgrading the existing repository



The user will be directed to the window below, where they have to select the target connection to be configured.

“Path to Enablement Pack” - will be auto populated by the path that was selected during the selection of unzipped Enablement Pack

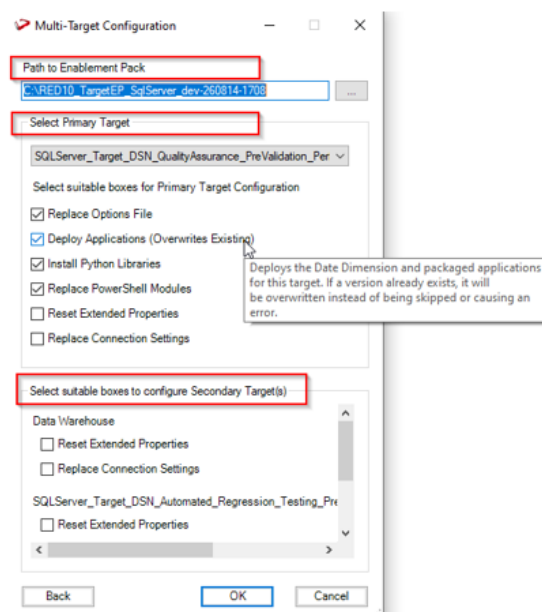
For Multi-target configuration one Target will be selected as Primary target and other targets will be selected as Secondary target.

If Installation type is selected as Upgrade Repository only below options will be selected by default

Additionally, by deselecting the provided options, the user can choose not to install a particular option.

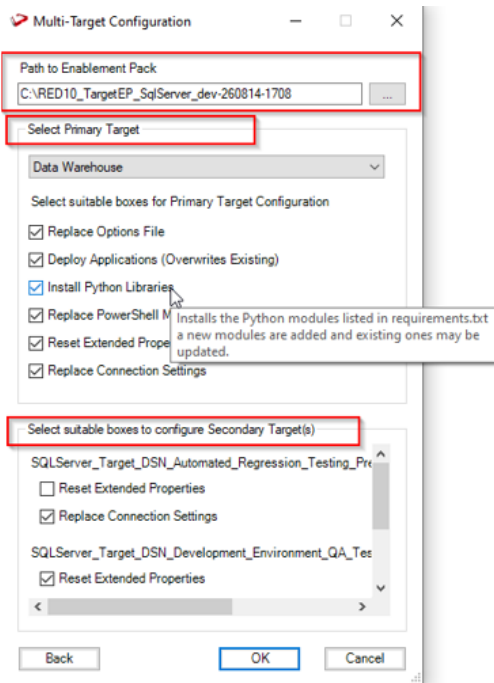
"Reset Extended Properties" and “Connection Configuration“ is deselected by default.

Tooltips are available for each option to provide additional information about its functionality.



If Installation type is selected as New Repository all the options will be selected by default

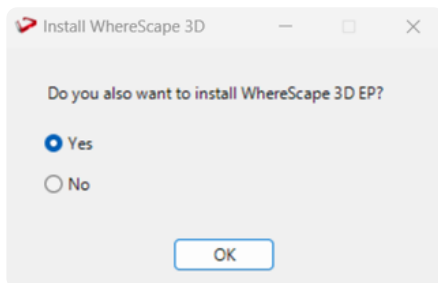
Additionally, by deselecting the provided options, the user can choose not to install a particular option.



Press OK

The user will be prompted to select Yes or No for whether they want the Enablement Pack installation on 3D repo.

- If Yes is selected, at the end of the post-installation process the 3D installation will be triggered, and the required Python libraries will be installed by default irrespective of check box selection for Install Python Libraries on previous window.
- If No is selected, the 3D installation will not take place.



The user will be directed to the window below, where they have to select the Create new profile or use existing one option.

Note: For fresh installation RED will create profile file with same name as DSN, which the user can use or choose to create new profile file.

For "Create new one" option , user will have to enter the required details below.

Choose Profile File

☒ Create new one ☐ Use existing one

Enter Details For Data Warehouse_target2

User ID:

Password:

Connection String: DSN=\$DSN\$;UID=\$USER\$;PWD=\$PASSWORD\$; ...

DSN:

Profile File Name:

Select the Profile File

Profile File:

OK

If the user choose "Use existing one" option. user can select the existing profile file.

Choose Profile File

☐ Create new one ☒ Use existing one

Enter Details For Data Warehouse_target2

User ID:

Password:

Connection String: DSN=\$DSN\$;UID=\$USER\$;PWD=\$PASSWORD\$; ...

DSN:

Profile File Name:

Select the Profile File

Profile File:

OK

Press OK

The progress bar will show the post installation progress.

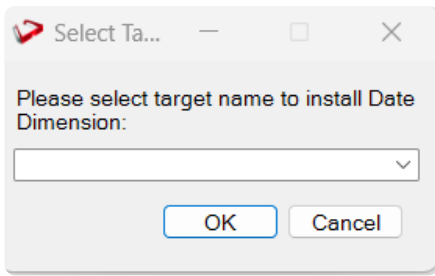
Post-Installation Steps

35%

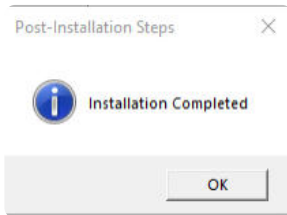
Enablement Pack Setup

Setting RED options

User will have to choose the schema for the target setting that were provided. The pop up will appear for setting default target schema for Date Dimension

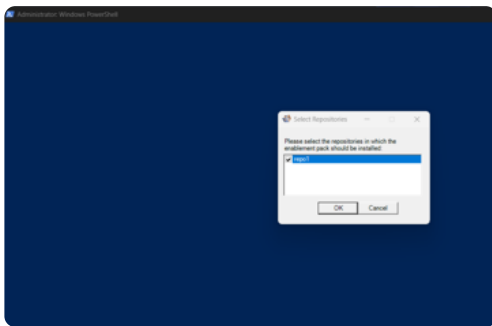


After selecting the target schema progress bar will show the progress for the installation and once it's completed, you will get the below pop up.

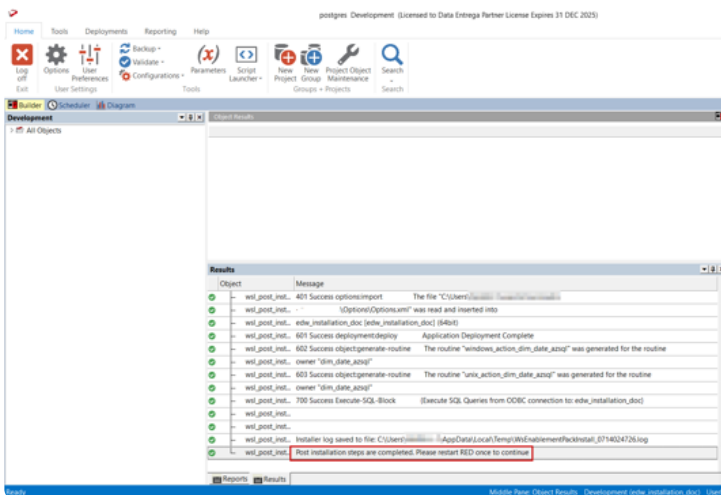


After pressing OK

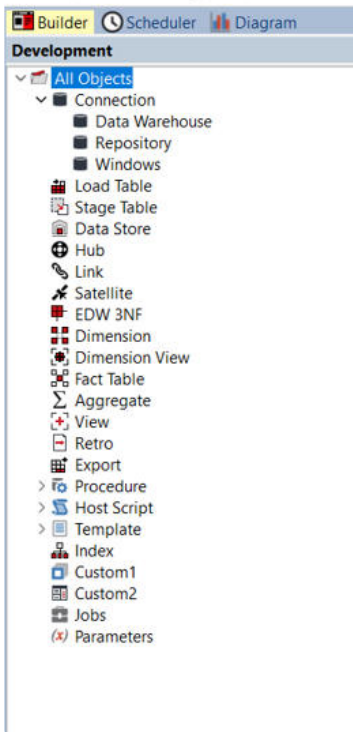
- If 3D installation was selected, a PowerShell terminal will open, and the 3D installation will start. To install the enablement pack on Wherescape 3D use the installation guide link mentioned above



- If 3D installation was not selected, RED will open directly.



User will need to restart RED once.



Upgrade Of Existing Repository

For upgrade of existing repository

- From host script set script type of `wsl_post_install_enablement_pack` as Auto Execute - PowerShell Script



⚠ Important Upgrade Notes

If RED upgrade the repository option is chosen.

This enablement pack will overwrite any existing Source Enablement Pack UI Configs:

Connection UI Config	Load UI Config
Amazon S3	Load From Amazon S3
Azure Data Lake Storage Gen2	Load From Azure Data Lake Storage Gen2
Google Cloud	Load From Google Cloud

⚠ To ensure existing Source Enablement Pack connections and associated Load Tables continue to browse and load:

Go into UI Configuration Maintenance in RED prior to installing this Enablement Pack and rename the affected UI Configurations. While the updated Load Template will work with previous Source Enablement Pack's we recommend moving these previous versions of Load Tables to newly created Parser based connections following this install. The earlier versions of the Source Enablement Pack will be deprecated following this release.

Kill script target connection — During a fresh repository installation, the kill script's connection name is automatically set to the **primary target connection** configured during installation.

If you want kill functionality to apply to a different target connection, you'll need to manually update it.

Note: this autoconfiguration only happens on a **new repository install** if you're **upgrading** an existing repository, the kill script retains whatever connection name was already set.

Setting Extended Properties for BCP Load for SQL Server as source:

The load Template supports 2 load methods for Sql Server as Source:-

1)Bulk Insert

2)BCP Load

BULK_LOAD_ROUTINE - Controls which load method is used.

BULK_TSQL	Standard T-SQL BULK INSERT
BCP	Use the bcp utility
Default (blank)	default (BULK_TSQL) method to load
Invalid value	If any invalid value is provided the load will use the default method for execution

SERVER_NAME - Set explicitly if BCP needs a different hostname or port than the ODBC connection string.

Example: `myserver.database.windows.net`

-s	SQL Server or Azure SQL server name passed to BCP
Default (Blank)	Value of <code>WSL_TGT_SERVER</code> environment variable

BCP_AUTH_METHOD - detects the authentication method

blank	Auto-detect (recommended)
-T	Windows integrated (trusted) authentication. No password required.
-G	Azure Active Directory authentication. Requires BCP 17.6 or later.

BCP_CMD_OPTIONS - The value entered is simply appended to the BCP command line.

Default	blank (recommended)
-u	to disable certificate validation.

Field enclosure with load method

blank (default)	BCP - extraction produces fully unquoted output BULK_TSQL BULK_TSQL -extraction produces output in double quotes("") i.e. default value of enclosure
double quotes (")	BCP - produces uniformly enclosed output quotes all the fields with double quotes("") BULK_TSQL - produces uniformly enclosed output quotes all the fields with double quotes("")
single quotes (')	BCP - produces uniformly enclosed output quotes all the fields with single quotes (') BULK_TSQL - produces uniformly enclosed output quotes all the fields with single quotes (')

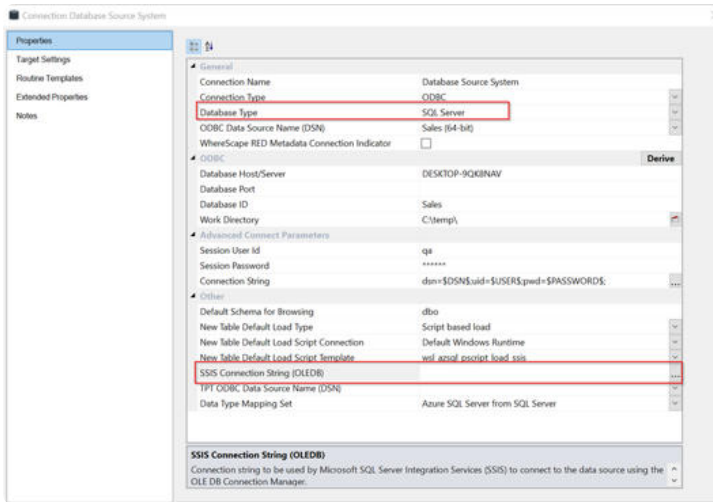
SSIS - Connection String Generation

• Prerequisites: -

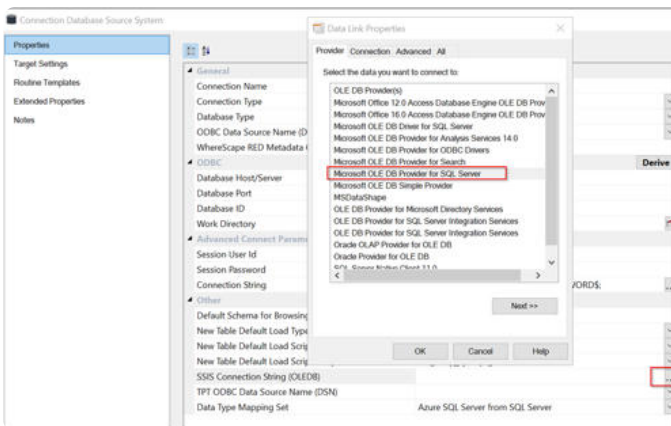
Install Visual Studio and Add Integration service Exception.

• With SQL Server As Source: -

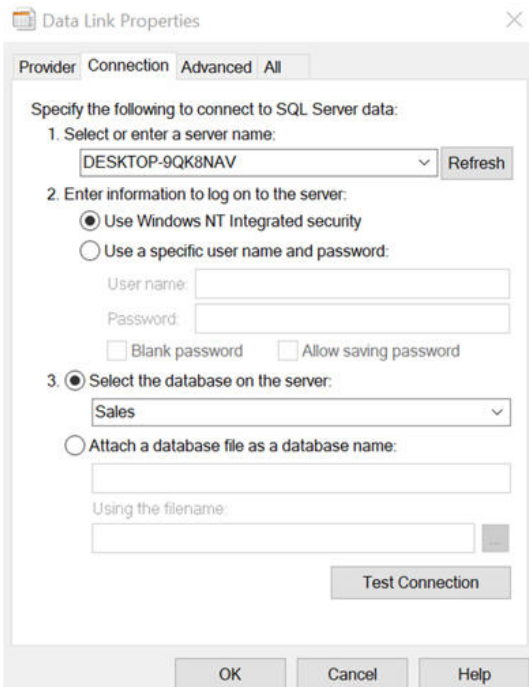
In the Source connection with database type as SQL Server, click on three dots in the SSIS connection string (OLEDB) option as below:



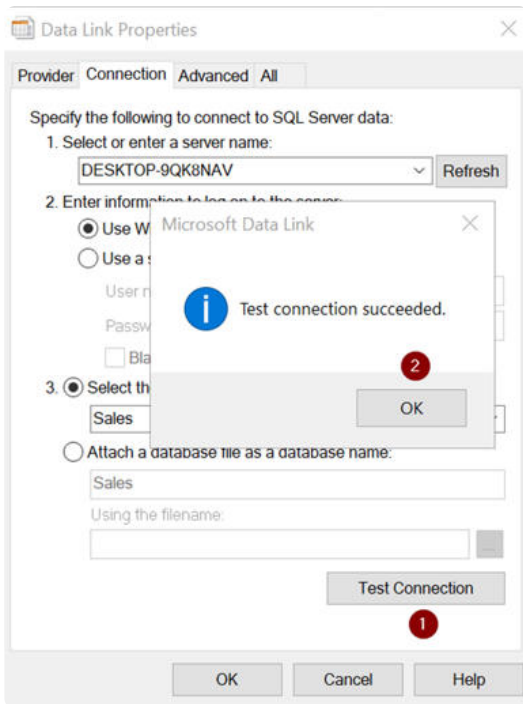
In the Data Link Properties window , select Microsoft OLE DB Provider for SQL Server:



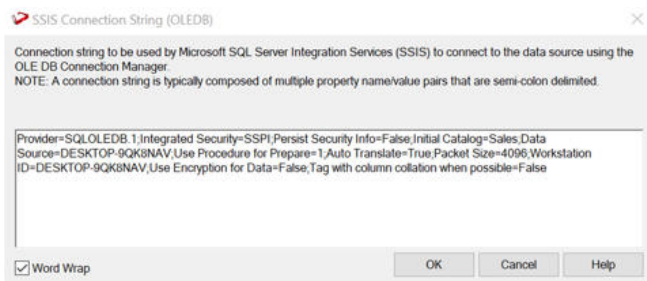
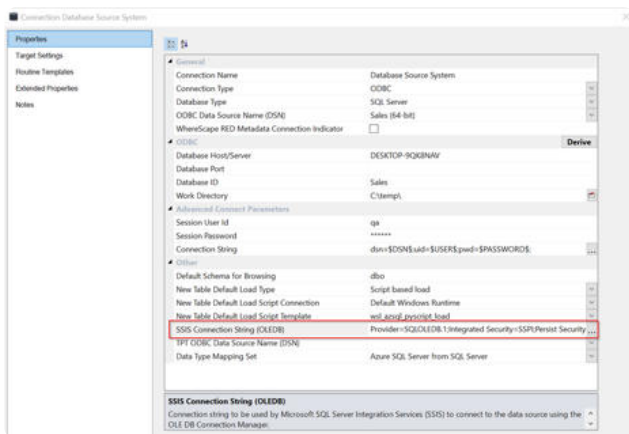
Provide Server and database details:



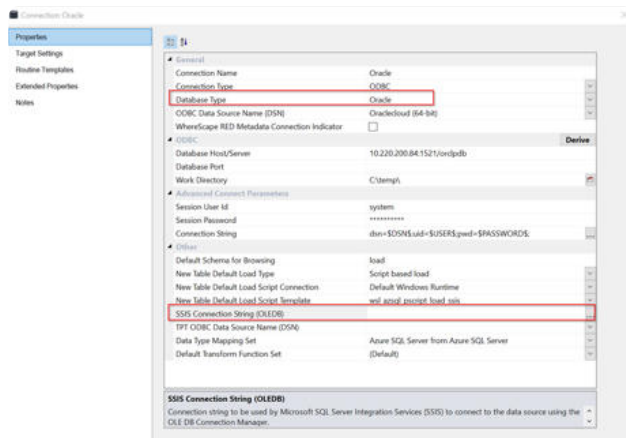
Test the connection:



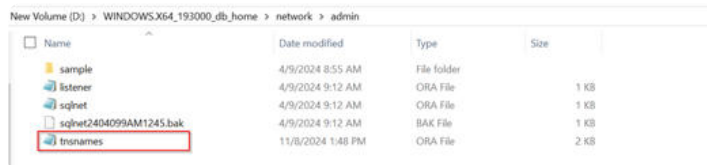
SSIS Connection String generated successfully.



With Oracle As Source: -

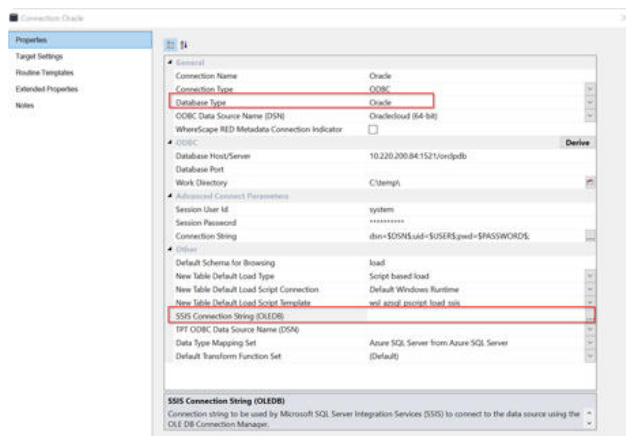


In tnsnames.ora file , define database connection details as below:

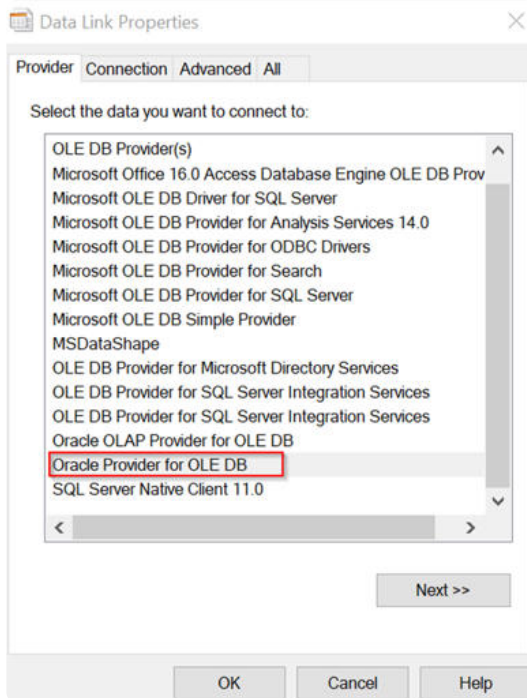


```
ORACLE =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = 10.220.200.84)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = orclpdb)
  )
)
```

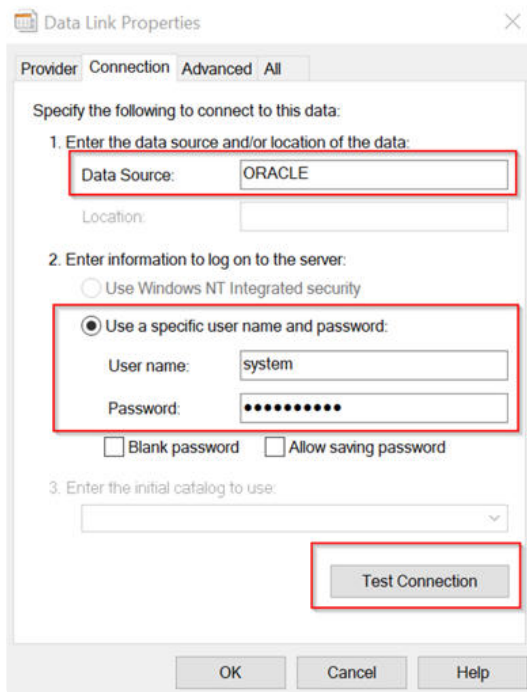
Open Oracle Source connection and click on three dots in SSIS Connection string option:



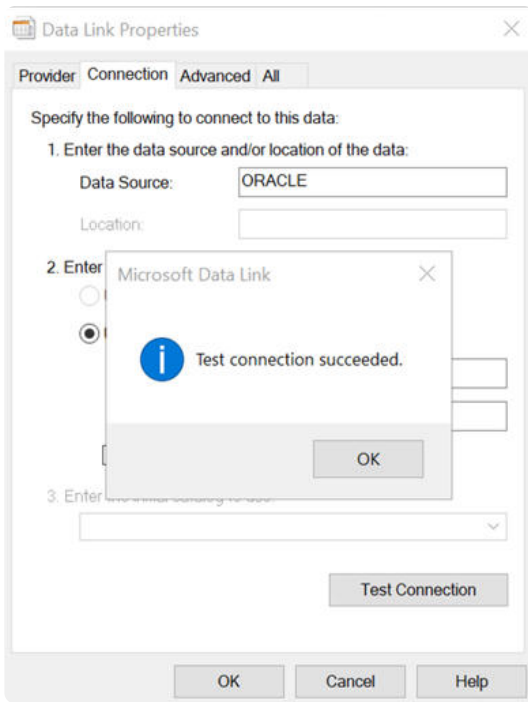
Select Oracle Provider for OLE DB option in Data Link Properties window:



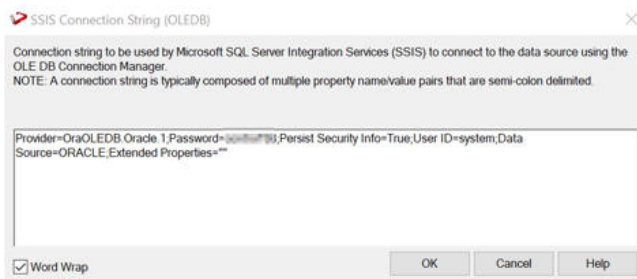
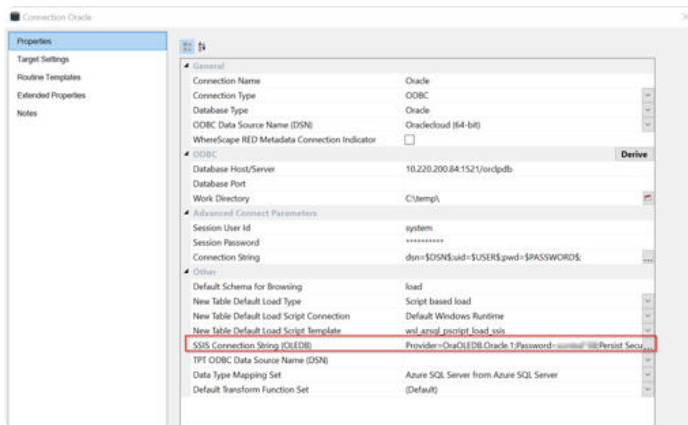
Provide the credentials as below:



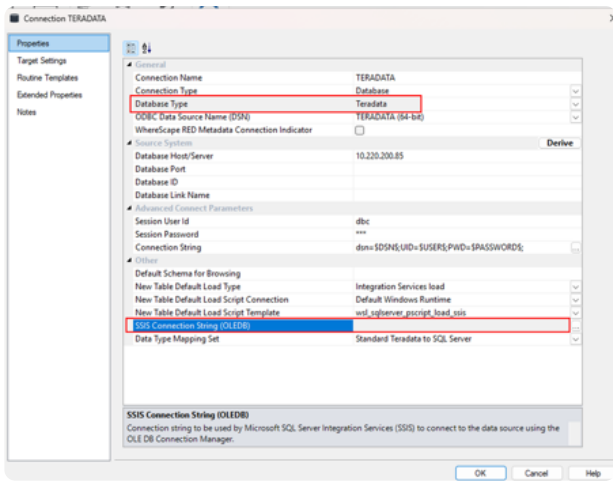
Test the connection:



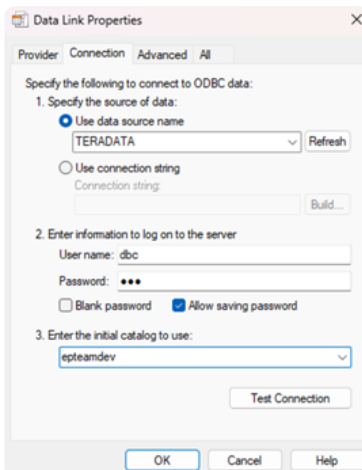
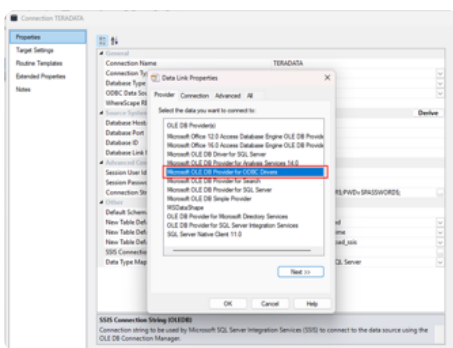
Connection string generated successfully:



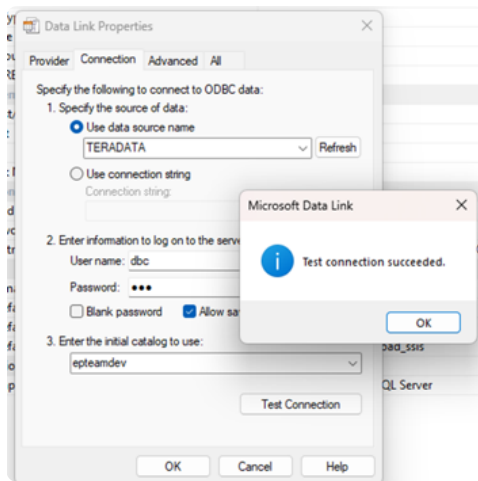
With Teradata As Source: -



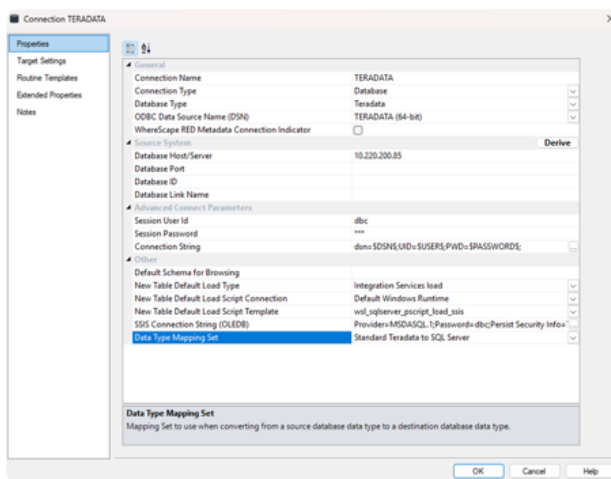
In the Data Link Properties window, select Microsoft OLE DB Provider for ODBC drivers:



Provide the Credentials as below and test the connection:



Connection string generated successfully



Post Install Steps – Optional

If you used the script Setup Wizard for installation then the following optional post install steps are available.

Configure Connections

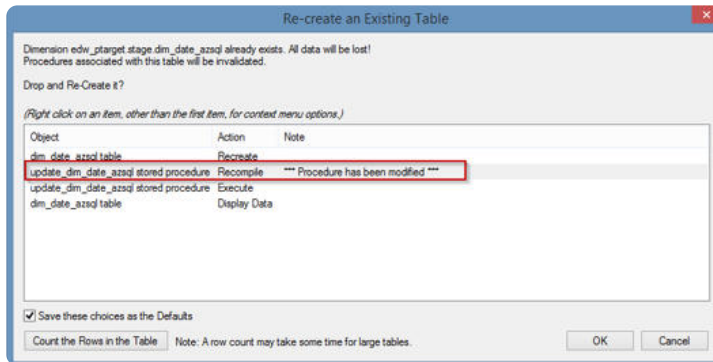
There were two connections added that will optionally require your attention:

Below two are common for SQL Server

1. Connection: 'Database Source System' - this connection was setup as an example source connection,
 - open it's properties and set it up for a source DB in your environment

- or you can remove it if not required

2. Execute Update Procedure for dim_date_azsql to **Initialize Date Dimension for Target**



NOTE: Please avoid regenerating the procedure update_dim_date_azsql as it is a custom procedure.

Source Enablement Pack Support

Source Pack Name	Support ed By SQL Server	Supported Features	Prerequisites
Cloud File Parser 1. Amazon S3 2. Azure Data Lake Storage Gen2 3. Google Cloud Storage	1. CSV 2. Excel 3. JSON 4. XML 5. AVRO 6. ORC 7. PARQUE T	Bulk load	Refer to Windows Parser Guide

Windows Parser	1. CSV 2. Excel 3. JSON 4. XML 5. AVRO 6. ORC 7. PARQUET	Load Template, Source Properties will have option to select parser type to load the files.	Refer to Windows Parser Guide
----------------	--	--	-------------------------------

Troubleshooting and Tips

Run As Administrator

Press the Windows Key on your keyboard and start typing cmd.exe, when the cmd.exe icon shows up in the search list right click it to bring up the context menu, select "Run As Administrator"

Now you have an admin prompt navigate to to the folder where you have unpacked your WhereScape Red Enablement Pack to using the 'cd' command:

```
C:\Windows\system32> cd <full path to the unpacked folder>
```

Run Powershell (.ps1) scripts from the administrator prompt by typing the Powershell run script command, for example:

```
C:\temp\EnablementPack>Powershell -ExecutionPolicy Bypass -File .\Setup_Enablement_Pack.ps1
```

Notes: In the event you can not bypass the Powershell execution policy due to group policies you can instead try "-ExecutionPolicy RemoteSigned" which should allow unsigned local scripts.

Windows Powershell Script Execution

On some systems Windows Powershell script execution is disabled by default. There are a number of workarounds for this which can be found by searching the term "Powershell Execution Policy".

Here is the most common workaround which WhereScape suggests, which does not permanently change the execution rights:

Start a Windows CMD prompt as Administrator, change directory to your script directory and run the WhereScape Powershell scripts with this command:

- cmd:>Powershell -ExecutionPolicy Bypass -File .\<script_file_name.ps1>

Restarting failed scripts

Some of the setup scripts will track each step and output the step number when there is a failure. To restart from the failed step (or to skip the step) provide the parameter "-startAtStep <step number>" to the script.

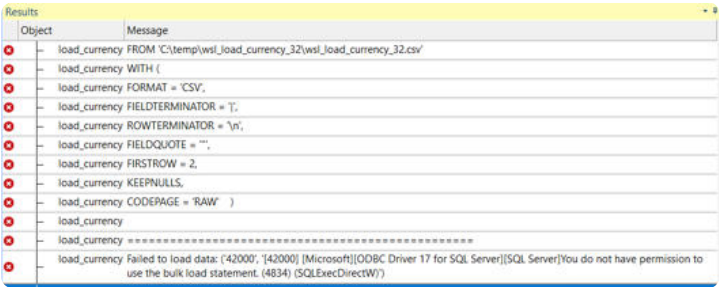
Example:

Powershell -ExecutionPolicy Bypass -File .\<script_file_name.ps1> -startAtStep 123

Tip: to avoid having to provide all the parameters again you can copy the full command line with parameters from the first "INFO" message from the beginning of the console output.

Bulk load error on SQL SERVER

Interactive load

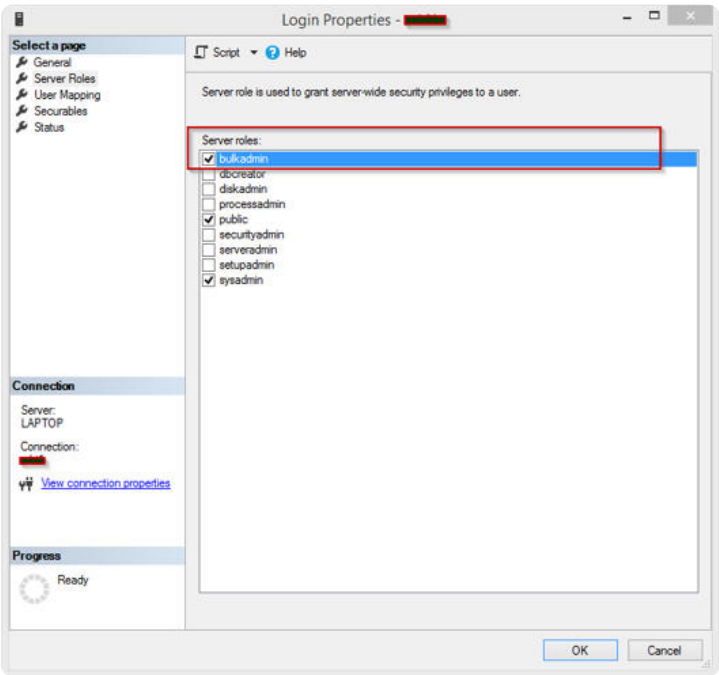


Scheduler load



For bulk load error, the 'bulkadmin' permission is required.

SQL SERVER → SECURITY → LOGINS → USER → PROPERTIES →SERVER ROLES



For bulk load error Operating system error code 5(Access is denied)

```
Failed to load data: ('42000', '[42000] [Microsoft][ODBC Driver 17 for SQL Server][SQL Server]Cannot bulk load because the file "C:\WhereScape\Temp\...\xxxx.csv" could not be opened. Operating system error code 5(Access is denied.). (4861) (SQLExecDirectW)')
```

1. Grant SQL Server access to the bulk load folder by adjusting folder permissions. Navigate to the folder, right-click, select "Properties," go to the "Security" tab, click "Edit," then "Add." In the new window, select "Advanced," click "Find Now," and locate the appropriate SQL Server user (e.g., SQLServerMSSQLUser\$UserName\$SQLEXPRESS). Confirm by clicking "OK" through all opened dialogs.
2. If the first fix doesn't resolve the issue, verify if permissions are applied to subfolders as well. Navigate to the advanced settings, and check the 'Inheritance' status. If it's 'Disabled,' enable it by using the "Allow Inheritance" button.

Row Terminator For Linux

Task_ID	1	2024-02-05 11:52:04.025488	*****
Task_ID	1	2024-02-05 11:52:04.025488	*****
Task_ID	1	2024-02-05 11:52:04.025488	*****
Task_ID	1	2024-02-05 11:52:04.025488	*****

Error Message:

```
Failed to load data: ('42000', '[42000] [Microsoft][ODBC Driver 17 for SQL Server][SQL Server]Cannot obtain the required interface ("IID_IColumnsInfo") from OLE DB provider "BULK" for linked server "(null)". (7301) (SQLExecDirectW)')
```

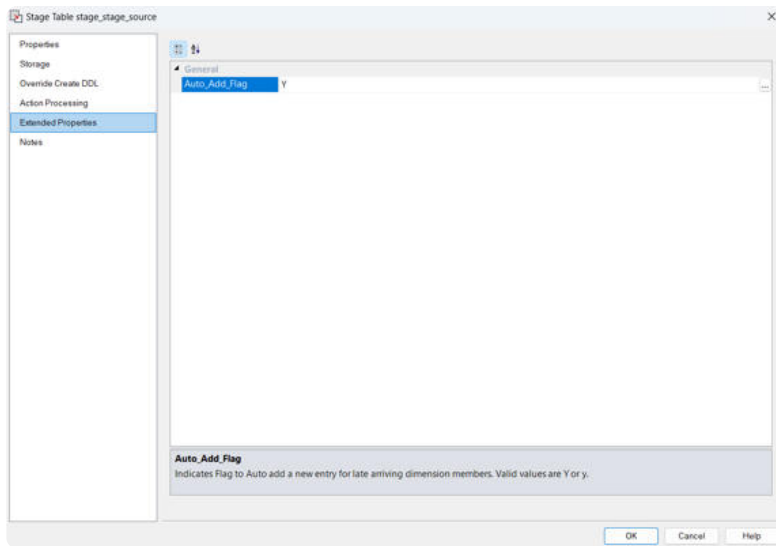
When you specify '\n' as a row terminator for bulk import in *UNLOAD_RECORD_CHAR* extended property or source properties, or implicitly use the default row terminator, BULK INSERT statement expects a carriage return-line feed combination (CRLF) as the row terminator. If your source file uses a line feed character only (LF) as the row terminator - as is typical in files generated on Unix and Linux computers - use hexadecimal notation to specify the LF row terminator. For example use '0x0A' in *UNLOAD_RECORD_CHAR* extended property or source properties for Unix and Linux machines.

Note: The hexadecimal notation is not required for Parquet file for Unix and Linux machines

Auto add a new entry for late arriving dimension members

Set This Extended property of stage table Named **Auto_Add_flag** to Y/y to add a new entry for late arriving dimensions.

If this flag holds any thing other than Y/y , it works as a normal stage table.



Python requirements for offline install

Additionally to the base Python installation being required, the WhereScape Python Template set also requires certain additional Python libraries. The install scripts uses the PIP (package manager) to download these libraries, however for offline installs you will need to install the required libraries yourself.

Required Python libraries/add-ons:

- pywin32-ctypes
- python-tds
- pywin32
- glob2
- gzip-reader
- regex
- pyodbc

If a valid RED installation can not be found

If you have RED 10.x or higher installed but the script (Setup_Enablement_Pack.ps1) fails to find it on your system then you are most likely running PowerShell (x86) version which does not show installed 64 bit apps by default. Please open a 64 bit version of Powershell instead and re-run the script.

Python 3.13/3.14 Modules Compatibility:

If you experience load failures or "ModuleNotFoundError: No module named 'cgi'" when using Python 3.13 or 3.14, it is due to the deprecation of the `cgi` library from the Python standard library

To resolve this without changing your Python version, install the `legacy-cgi` package to provide the necessary backward compatibility for WhereScape Enablement Packs.

Run the following command in your Python environment: `pip install legacy-cgi`