

Install Guide - Setup Wizard - Microsoft Fabric

WhereScape Enablement Pack for Microsoft Fabric-RED 10.2+

This is a guide to installing the WhereScape Enablement Pack for Microsoft Fabric in WhereScape RED10

Table of Contents

[Prerequisites For Microsoft Fabric](#)

[Installation Through Setup Wizard](#)

[Upgrade Of Existing Repository](#)

[Setting Extended Properties for ADF and Load templates:](#)

[Post Install Steps – Optional](#)

[Configure Connections](#)

[Source Enablement Pack Support](#)

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  
2 | Register-PSRepository -Default -Verbose  
3 | 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 For Microsoft Fabric

Before you begin the following prerequisites must be met:

- For **Windows**
 - Install the latest Azure-CLI package - [Latest MSI of the Azure CLI \(64-bit\)](#)
- For **Linux**
 - Install latest Azure CLI package using command:

```
1 | curl -sL https://aka.ms/InstallAzureCLIDeb | bash
```

1. Open the Command prompt :

- Execute this command for **Azure Active Directory Password authentication using a login ID and password**

az login command:

```
1 az login
```

- Execute this command for **Service Principal authentication**

az login command:

```
1 az login --service-principal --username <Client ID> --password <Client Secret Value> --tenant <Tenant ID>
```

Note: If the current Azure login session cannot be reused or has expired, you may be prompted to run `azd auth login`. Execute the command if prompted.

- Follow the below steps to configure a DSN using Azure Active Directory Password authentication using a login ID and password

Install ODBC Driver for SQL Server and configure the DSN by adding below details

Server: Enter your Server name

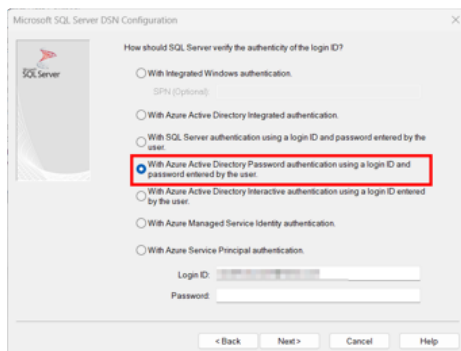
Authentication: Azure Active Directory Password authentication using a **login ID and password**

Authentication

Login ID: Enter your user ID.

Password: Enter your password.

Click Next and **Test Connection**.



- Follow the below steps to configure a DSN using Service Principal authentication

Install ODBC Driver for SQL Server and configure the DSN by adding below details

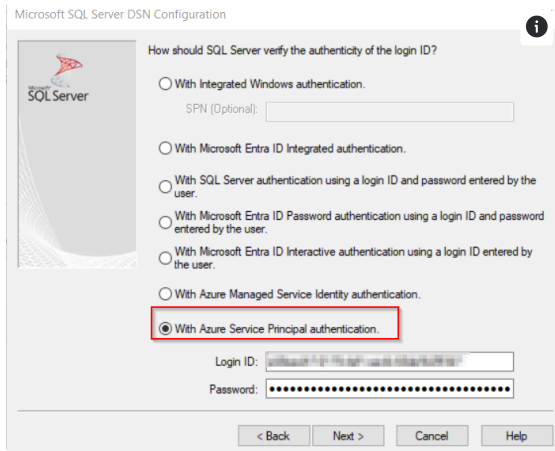
Server: Enter your Server name

Authentication: Select **Azure Service Principal Authentication**

Login ID: Enter your **Client ID**.

Password: Enter your **Client Secret value**.

Click Next and **Test Connection**.



Note: A detailed setup document "[Setup Guide_Microsoft Fabrics_Service Principal Authentication](#)" is available in the Enablement Pack folder for reference and at below location - [Setup Guide_Microsoft Fabrics_Service Principal Authentication - Enablement Packs - Product Documentation](#)

2. [Python 3.11 or higher](#)

- Select "Add Python 3.11 to PATH" from installation Window
- Pip Manager Install with command:

```
1 python -m pip install --upgrade pip
```

* We recommend using case-insensitive (CI) collations during the creation of your Fabric Warehouse

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](#)

3. Setting up Environment Variables:

Add System Variables as below

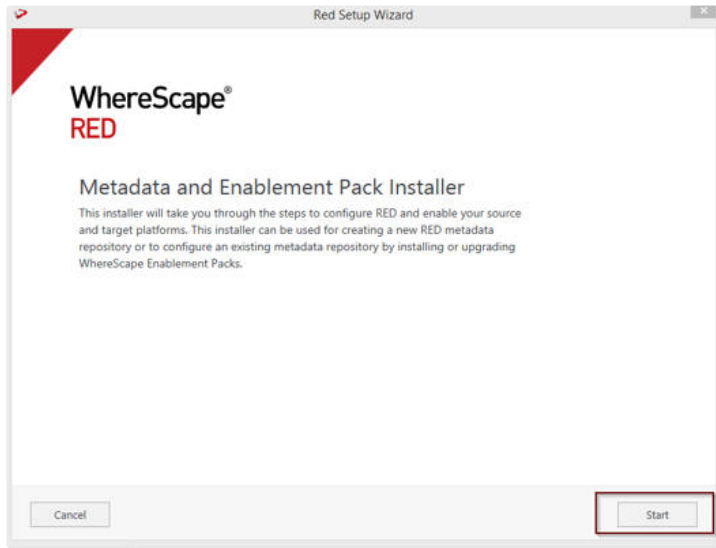
- AZURE_CLIENT_ID ← Client ID
- AZURE_CLIENT_SECRET ← Client Secret Value
- AZURE_TENANT_ID ← Tenant ID

System variables	
Variable	Value
AZURE_CLIENT_ID	ad88a8d7-0878-4687-9a6d-815a5e0c01e3
AZURE_CLIENT_SECRET	ad88a8d7-0878-4687-9a6d-815a5e0c01e3
AZURE_TENANT_ID	ad88a8d7-0878-4687-9a6d-815a5e0c01e3

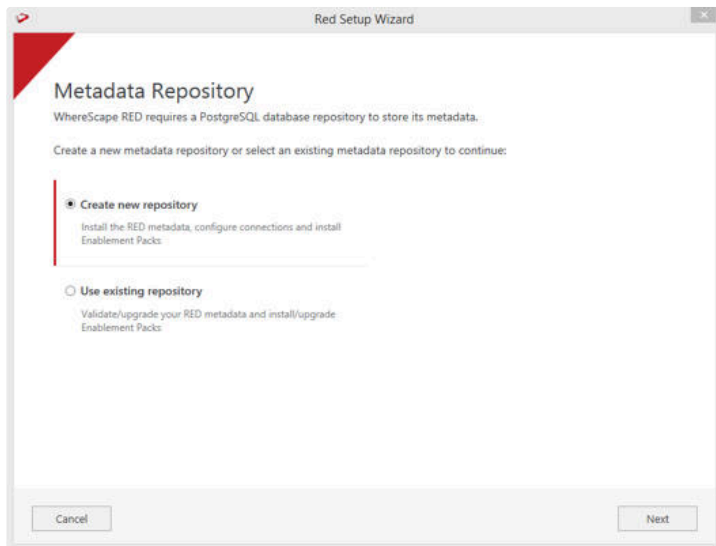
- These steps are required only if Service Principal authentication is used
- The authentication method used here first checks for the required environment variables. If they are not set, authentication will still succeed as long as `az login` has been executed with the appropriate credentials.
- Authentication can be done either by setting the environment variables or authenticate using `az login` —both approaches provide the same authentication outcome.

Installation Through Setup Wizard

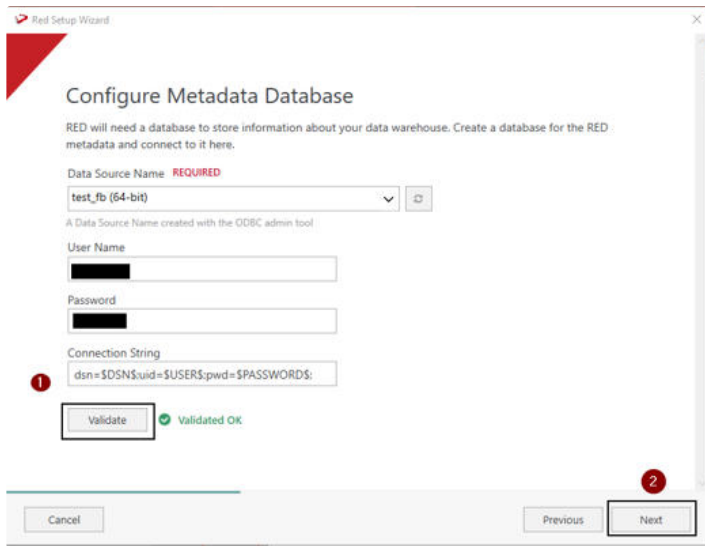
Run Setup Wizard as administrator



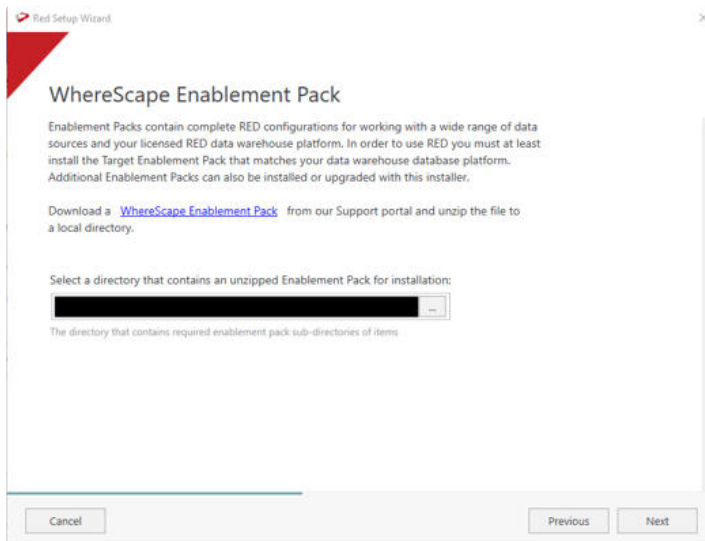
Create new repository or upgrade already existing repository.



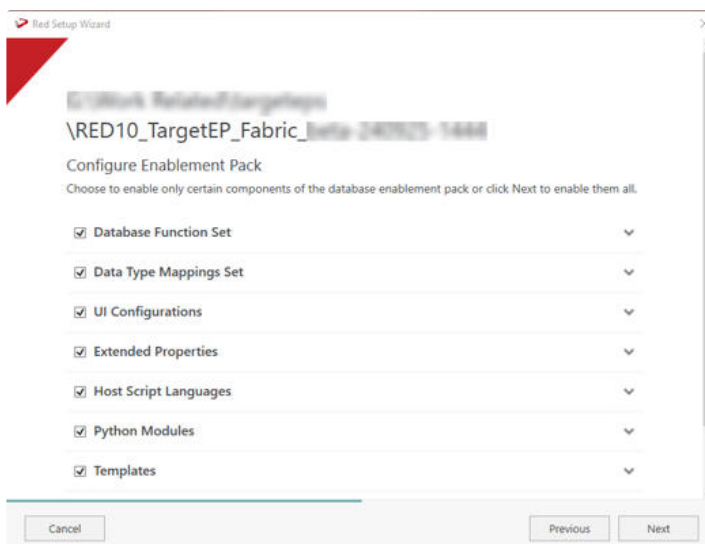
Select the created ODBC DSN, input login details with connection string and then select "Validate". Press Next



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



Using the check boxed list, include or exclude the components that are to be installed. Press Next



To configure a target connection using Service Principal Authentication, enter the **Client ID** as the **username** and the **Client Secret Value** as the **password**. Add target location. Validate and press ADD.

Red Setup Wizard

Connection Name **REQUIRED**
Data Warehouse

A unique name that identifies this connection

Data Source Name **REQUIRED**
Fabrics_mfa (64-bit)

A Data Source Name created with the ODBC admin tool

Database Type **REQUIRED**
CUSTOM

The target platform type for this connection

User Name
[REDACTED] ← ClientID

Password
[REDACTED] ← Client Secret Value

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

Target Storage Locations **REQUIRED**
Existing schema (or database names depending on platform) for object storage
dev_load

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.

+

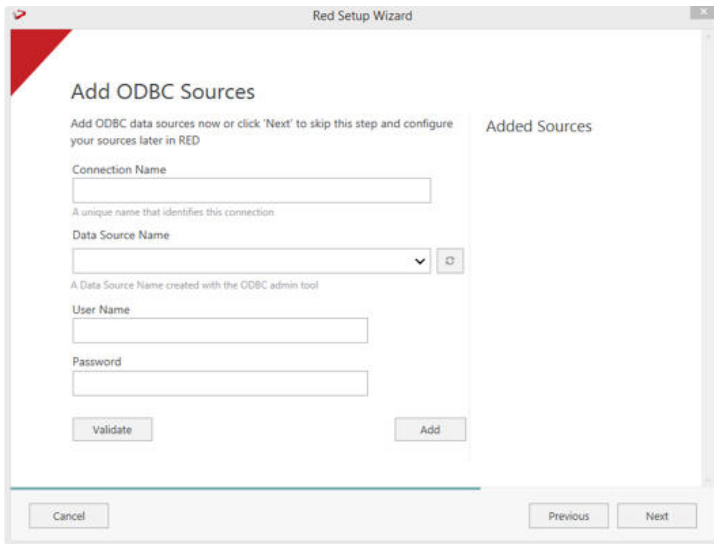
Add another target

Added Targets

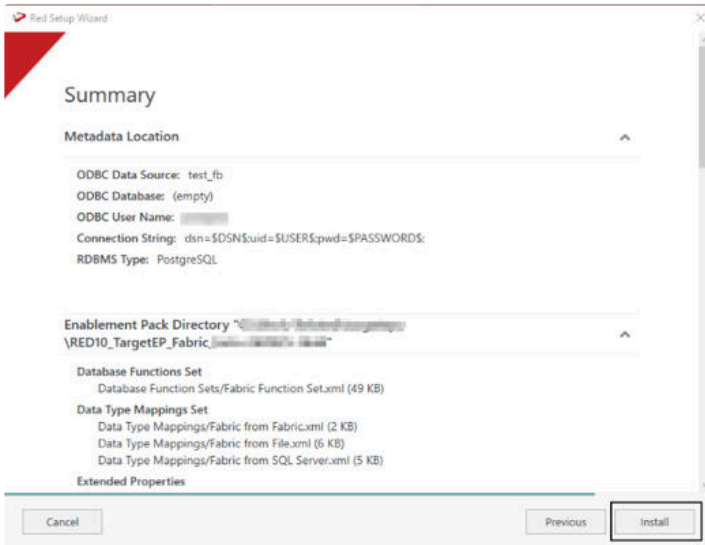
- Data Warehouse Fabrics

Cancel Previous Next

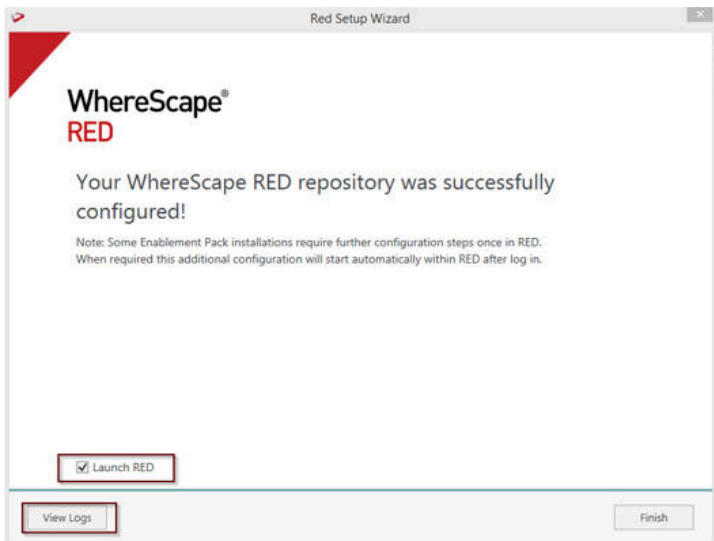
Configure a data source connection (optional) and its target locations. Validate and press ADD. Press Next to advance.



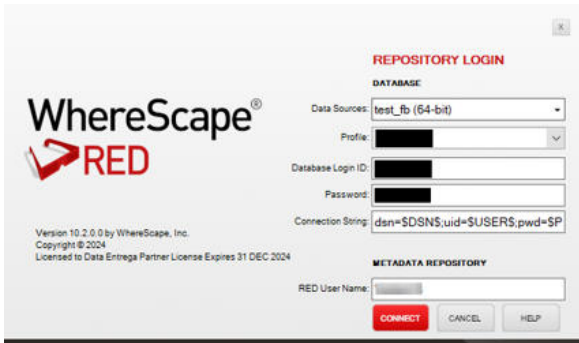
Review the installation summary and press Install



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

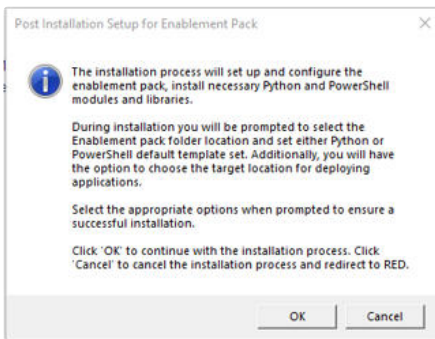


Login to WhereScape RED.



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.

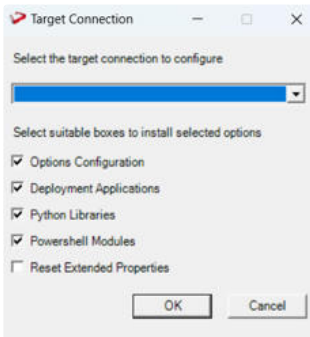


Press OK to start the post installation.

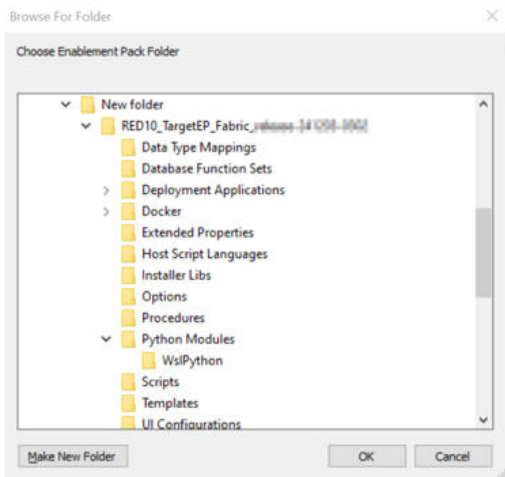
If pressed Cancel installation will stop and user will be directed to RED.

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

Additionally, by deselecting the provided options, the user can choose not to install a particular option. "Reset Extended Properties" is deselected by default.

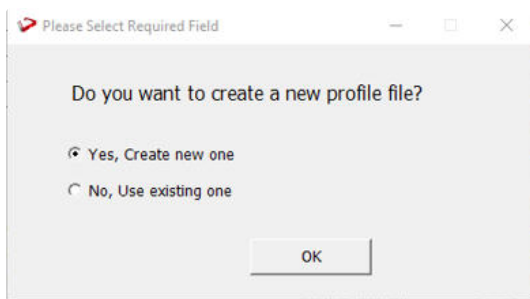


You will be directed to below PowerShell window. Provide the directory that contains unzipped Enablement Pack.



Press OK

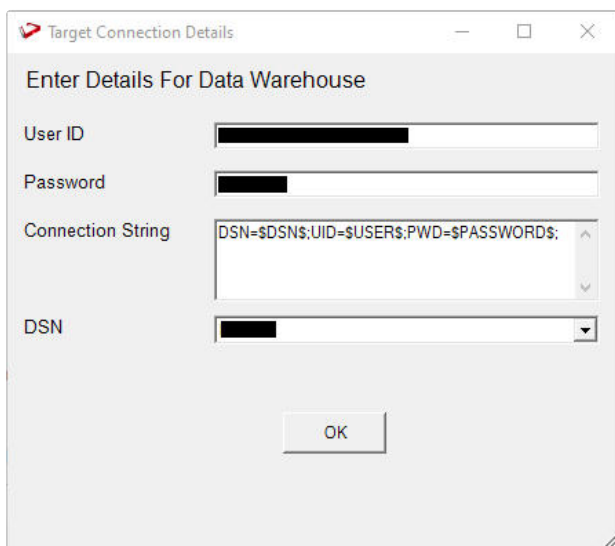
The user will be directed to the window below, where they have to select the Create new profile or use existing one option.



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.

Press Ok.

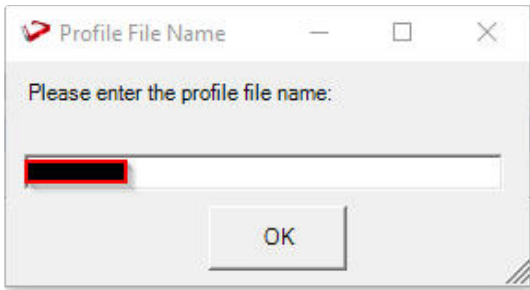
For "Yes, Create new one" option , user will be directed to the window below.



Note: User can use default connection string or input new one.

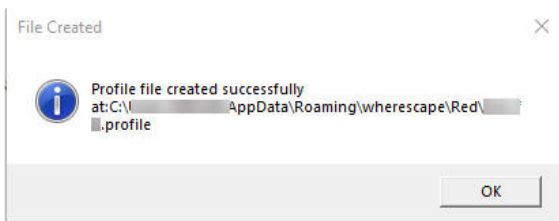
Press OK

The user will be directed to the window below, where user can add profile name.



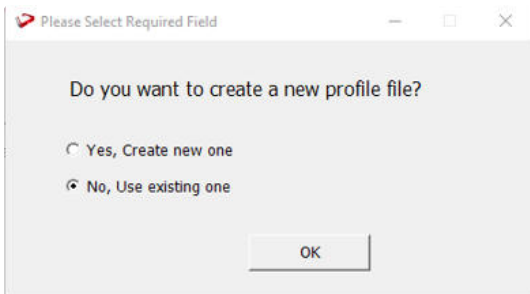
Press Ok.

The below pop up will come to confirm the user that profile is created at that location



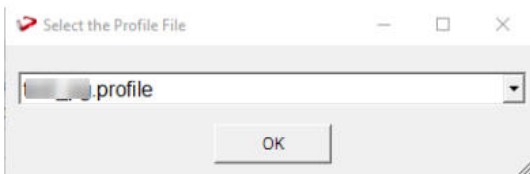
Press OK.

If the user choose "No, Use existing one" option.



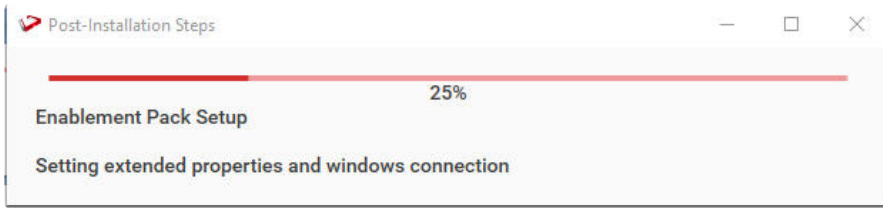
Press OK

The user will be directed to the window below ,where user can select the exiting profile file.

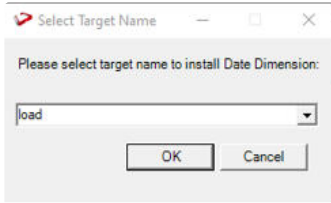


Press OK.

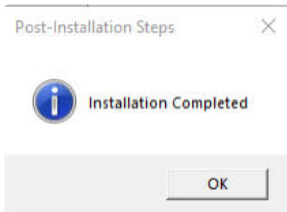
The progress bar will show the post installation progress.



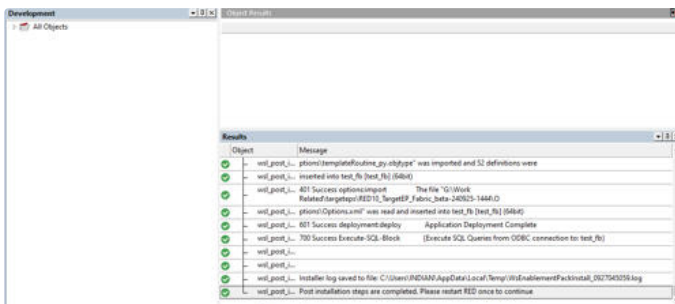
User will have to choose the schema for the target setting that were provided. One pop up will come for setting default target schema for Date Dimension.



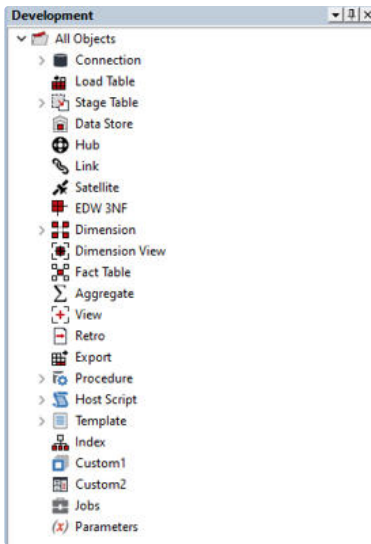
After selecting the target schema progress bar will show the progress for the installation and once it's completed, you will get the below pop up.



After pressing OK RED10 will open automatically.



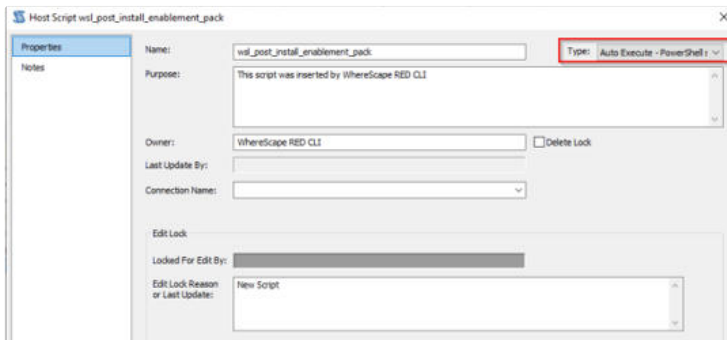
User will need to refresh the All Objects tree once.



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



Setting Extended Properties for ADF and Load templates:

The EP contains 2 load templates:

1. *wsl_fabric_pyscript_load* will support load for Database, OneLake and traditional file parser.
2. *wsl_fabric_pyscript_load_df* template is for Fabric Data Factory.

ADF Template:

Fabric Data Factory template currently supports OneLake and database sources, including SQL Server, Azure SQL Server, Snowflake, Oracle, and Azure Synapse.

When creating a connection for file browsing, the Lakehouse name must be specified in the source connection. After selecting a file through the browser, dragging the table into the flow automatically populates all relevant source properties.

In addition to the standard properties, **four extended properties** are available specifically for pipeline execution:

1. Recreate Pipeline on Run

This property determines how pipelines are handled during execution.

- When set to True (default), a new pipeline is created every time the job runs.

Any existing pipeline with the same name is deleted before recreation.

- When set to **False**, the **existing pipeline is retained**, and only a new execution run is triggered.

However, if a pipeline fails due to schema or property mismatches, the pipeline must be manually deleted and recreated, because **its JSON definition cannot be modified** when this property is False.

- If the value is set to **DROP**, the existing pipeline is deleted and **not** recreated.

Once a pipeline successfully loads data, turning this property to **False** allows that pipeline to be preserved as a stable starting point for other future operations.

2. Pipeline Timeout Duration

This property defines how long RED will wait for the pipeline to complete.

- The default timeout is **10 minutes**.
 - If the timeout is reached, the pipeline continues running in the background, and RED alerts the user that execution is still in progress.
 - The user can then monitor the run through the **Fabric portal**.
 - If the job finishes earlier (e.g., in 1–2 minutes), RED immediately marks the run as successful and displays the result.
-

3. Max Retries to Recreate Pipeline

This setting controls reliability during pipeline recreation.

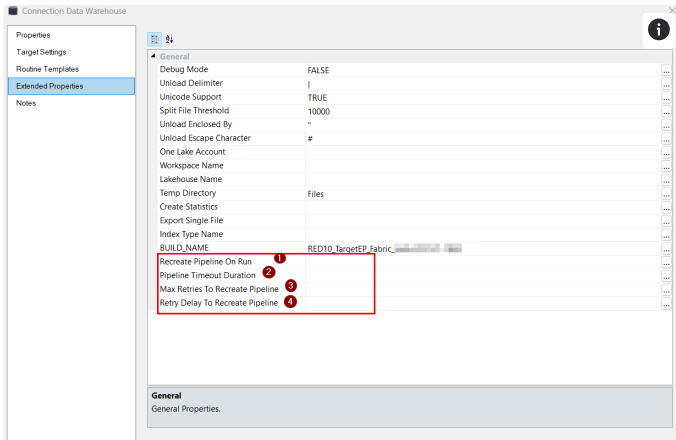
- It specifies the **maximum number of retry attempts** allowed when pipeline creation fails (e.g., due to transient issues).
 - The default retries count is **6**.
 - If recreation continues to fail after the allowed attempts, the process stops and the error is surfaced.
-

4. Retry Delay to Recreate Pipeline

This property introduces a delay between retries or recreation attempts.

- The value represents the **delay in seconds** before attempting to recreate the pipeline.

- Default delay = **60 seconds**.
- This helps ensure cleanup is completed and avoids stale or corrupted pipeline configurations.



Handling Data Type Issues

While loading data through Fabric Data Factory, type mismatches may occur. In such cases, the pipeline displays an error and suggests updating the affected column—commonly recommending **varchar (8000)**.

To resolve:

1. Modify the table's column data type as recommended.
2. Recreate the table.
3. Regenerate the script.
4. Execute the pipeline again with **"Recreate Pipeline on run" = True** (or leave it blank to use the default behavior).

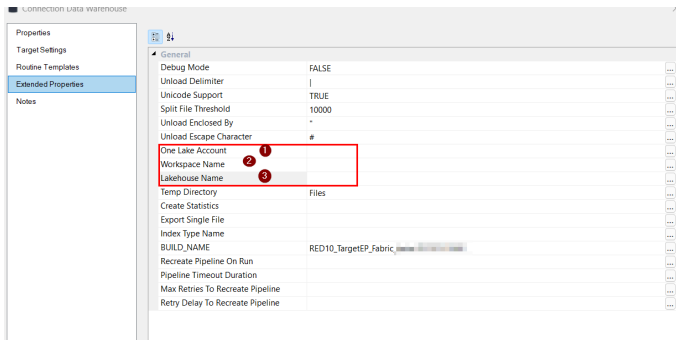
Load Template:

The load template supports loading data from **Databases, OneLake**, and the **traditional file parser**.

Target Connection – Required Extended Properties

You must configure the following extended properties in the **Target connection** to enable OneLake-based staging:

1. **OneLake Account:** Specify the OneLake account where temporary staging files will be uploaded during the load process.
2. **Workspace Name:** Required for script-based loads to ensure the load operation is routed to the correct Fabric workspace.
3. **Lakehouse Name:** Provide the name of the Lakehouse within the workspace where staging or target data will reside.

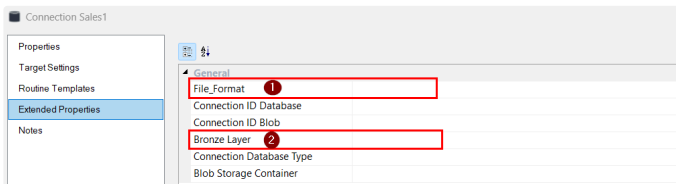


These properties collectively allow the system to connect to OneLake for staging operations.

Source Connection – Additional Extended Properties

Set the following extended properties in the **Source connection** when using OneLake staging:

1. **File Format:** Defines the file format generated when the source is a database. This indicates the file type created in Fabric for data extraction.
Acceptable values: **parquet** or **csv**.
2. **Bronze Layer:** Controls whether files in the OneLake staging area should be retained or removed.
 - o Set **TRUE** to retain the staged files.
 - o Set **FALSE** to delete them after processing.



Post Install Steps – Optional

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

Configure Connections

These connections added that will optionally require your attention:

1. Connection: Data Warehouse ('Fabrics')- This connection was setup as per parameters provided in Setup Wizard
 - a. open Properties and click derive button for Database Host/Server and Database ID.
 - b. open it's properties and check extended properties tab, set it up for **One Lake Account, Workspace Name and Lakehouse Name**.
2. Connection: 'Database Source System' - this connection was setup as an example source connection,
 - a. open its properties and set it up for a source DB in your environment
 - b. or you can remove it if not required

Enable Script Launcher Toolbar

There are a number of stand-alone scripts which provide some features such as "Ranged Loading", these scripts have been added to the Script Launcher menu but you will need to enable the menu toolbar item to see them.

To enable the Script Launcher menu in RED: Select menu item 'Home→Script Launcher'

Source Enablement Pack Support

Source Pack Name	Supported By Microsoft Fabric	Supported Features	Prerequisites/Permissions Required for Microsoft Fabric
Google Cloud Storage	Yes	Download to local and load	None
Azure Data Lake Storage Gen2	Yes	Download to local and load	None
Amazon S3	Yes	Download to local and load	None
Windows Parser	Yes	Load Template, Source Properties will have option to select parser type to load the files.	Refer to Windows Parser Guide
Azure One Lake	Yes	Download to local and load	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 batch (.bat) scripts from the administrator prompt by simply typing the name at the prompt and hit enter, for example:

- C:\temp\EnablementPack>[install_WslPowershell_Modules.bat](#)

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 cannot 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>**

Re-install Python Libraries

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 batch (.bat) scripts from the administrator prompt by simply typing the name at the prompt and hit enter, for example:

- C:\temp\EnablementPack>[uninstall_WslPython_Modules.bat](#)

For installation of Python libraries there are two methods

• Method 1

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 batch (.bat) scripts from the administrator prompt by simply typing the name at the prompt and hit enter, for example:

```
C:\temp\EnablementPack>install_WslPython_Modules.bat
```

• Method 2

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 the below command

```
python -m pip install -r requirements.txt
```

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

Restarting failed scripts

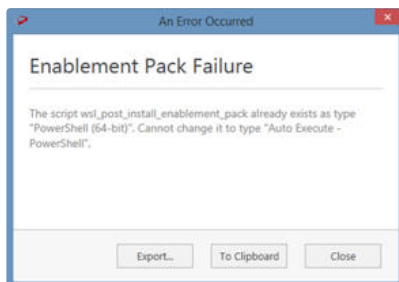
Some of the setup scripts will track each step and output the step number when there is a failure. To restart from the failed step (or to skip the step) provide the parameter "-startAtStep <step number>" to the script.

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.

For upgrade of existing repository



In upgrade of exiting repository if the user gets above error then it means the script type of *wsl_post_install_enablement_pack* is set to PowerShell(64-bit) change the script type to Auto Execute-PowerShell before upgrade or manually run the *wsl_post_install_enablement_pack* script from host script from RED after upgrade.

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 you 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

Table name should be given in lowercase only

While loading the table, the table name should be given in lowercase, for example, `load_tablename`; otherwise, loaded data will now get displayed.

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`