

WhereScape Enablement Pack for SQL Server

This is a guide to installing the WhereScape Enablement Pack for SQL Server for WhereScape RED 8.6.6.1 or higher

Prerequisites

Before you begin the following prerequisites must be met:

- A supported* version of SQL Server with
 - A database to house the RED Metadata Repository
 - (optionally) a database for the Range Table DB
 - ODBC DSN's created for these DB's
 - RED Metadata Repository Database permissions- SELECT,INSERT,UPDATE and EXECUTE – Optional
 - Note: The above permissions are for Scheduler installation,this can be done manually later.
- 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)
 1. Add ODBC DSN for SQL Server Target Database
 - .Net Framework 4.8 or higher
 - [bcp Utility](#) installed and enabled
 - Windows Powershell version 5 or higher
 1. Note: make sure this is done using a 64-bit powershell terminal
- WhereScape RED version 8.6.6.1 or higher
 - Must be pre-installed with valid license key entered and EULA accepted
- [Python 3.8 or higher](#)- For File Parser
 - Select "Add Python 3.x to PATH" from installation Window
 - Pip Manager Install with command : `python -m pip install --upgrade pip`
- WhereScape Enablement Pack for SQL Server version 8.6.6.1 or higher
 - Downloaded and unpacked to a local temp folder

* : RED supports the following versions for the metadata repository: MS SQL SERVER 2012 to 2019 and Azure SQL DB

Enablement Pack Setup Scripts

The Enablement Pack Install process is entirely driven by scripts. The below table outlines these scripts, their purpose and if "Run as Administrator" is required.

#	Enablement Pack Setup Scripts	Script Purpose	Run as Admin	Intended Application
1	Setup_Enablement_Pack.ps1	Setup and configure a RED Metadata Repository for SQL Server If RED repository exists then updates the repository with 1.Templates 2.Scripts 3.Extended Properties 4.Datatype Mappings 5.UI Configurations	Yes	New and Existing installations
2	install_WslPowershell_Modules.bat	Installs or updates WslPowershell Modules on this machine	Yes	New and Existing installations
3	install_WslPython_Modules.bat	Installs or updates WslPython Modules and required Python libraries on this machine	Yes	New and Existing installations

* Note that on some systems executing Windows Powershell scripts is disabled by default, see [troubleshooting for workarounds](#)

Each Powershell script in the list above provides some help at the command line, this can be output by passing the "-help" parameter to the script.For Example > .\Setup_Enablement_Pack.ps1 -help

Step-By-Step Guide

Setup and configure a new RED Metadata Repository for SQL Server

Run Powershell as Administrator:

```
Script 1 > Powershell -ExecutionPolicy Bypass -File .\Setup_Enablement_Pack.ps1
```

Install or Update WhereScape Powershell Modules

Run Script As Administrator

```
Script 2 > install_WslPowershell_Modules.bat
```

Install or Update WhereScape Python Modules

Run Script As Administrator

```
Script 3 > install_WslPython_Modules.bat
```

There are two steps in this script:

1. Installs the WhereScape WslPython modules to C:\Program Data\WhereScape\Modules\
2. PIP to download/update required Python libraries - for offline install please see the required library list for Python in the Troubleshooting section.

Note: The upgrade will not overwrite the existing Data Type Mappings, UI configs. The user either needs to manually delete them or use startAtstep for continuing the script execution.

Important Upgrade Notes



If RED repository exists, it will prompt to upgrade the repository. This enablement pack will overwrite any existing Source Enablement Pack UI Configs:

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:

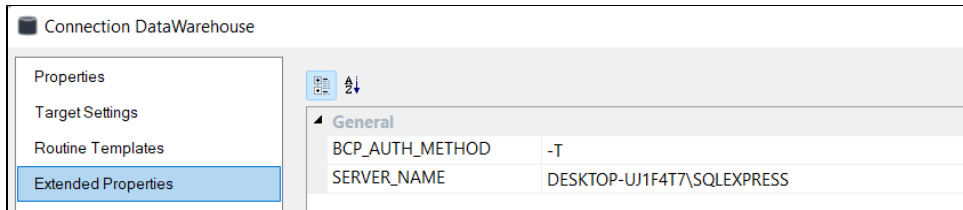
Go into UI Configuration Maintenance in RED prior to installing this Enablement Pack and rename the affected UI Configurations. While the updated Load Template will work with previous Source Enablement Pack's we recommend moving these previous versions of Load Tables to newly created Parser based connections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

Post Install Steps - Optional

Configure Connections

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

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. Target Connection Extended Property - SERVER_NAME
 - Add a complete server name for SQL server or SQL Server instance.
3. Target Connection Extended Property - BCP_AUTH_METHOD
 - -T for trusted Connections
 - -G for Azure Active Directory authentication
 - Default blank



4. Add exact record terminators for the files from windows source connection, please note \r\n and \n are considered as different record terminators.

NOTE: For record terminator '\n' use hexadecimal notation '0x0A'

Enable Script Launcher Toolbar

There are a number of stand-alone scripts which provide some features, 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 'View->Toolbars->Script Launcher'

Source Enablement Pack Support

Source Pack Name	Supported By SQL Server	Supported Features
Amazon S3	Yes	<p>Download to local and load using bcp</p> <p>Install Powershell package:</p> <ul style="list-style-type: none"> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12 Install-Module -Name AWSPowerShell.NetCore <p>Note: Complete system restart is recommended after installation of AWS modules. In case of "The term 'Set-AWSCredential' is not recognized" error, please refer to troubleshoot section.</p>
Azure Data Lake Storage Gen2	Yes	<p>Load file from Azure Data Lake Storage Gen2 bucket to SQL Server</p> <p>Note: Files must be contained within a folder; if a file is found in the root directory, it will not be loaded.</p> <p>Required Properties:</p> <ul style="list-style-type: none"> Azure Data Lake Storage Gen2 Account Name Azure Data Lake Storage Gen2 Access Key Azure Data Lake Storage Gen2 SAS Token Azure Data Lake Storage Gen2 Container Name <p>Install Powershell packages:</p> <ul style="list-style-type: none"> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12 Install-Module -Name Az.Storage -Force -AllowClobber <p>Note: Complete system restart is recommended after installation of Azure modules.</p>
Google Drive	No	None
Google Cloud	Yes	<p>Please refer the official link for steps to download gsutil: https://cloud.google.com/storage/docs/gsutil_install</p>

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

NOTE:

File Types : AVRO,ORC and PARQUET

Load to SQL Server : If the source file data contains field delimiter (Default ", "(comma)), change it in the source properties of load table to load data successfully.

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 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 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 on the internet 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.

If a valid RED installation can not be found

If you have Red 8.6.6.1 or higher installed but the script (Setup_Enablement_Pack) 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.