



WHERESCAPE RED ADMINISTRATION GUIDE

6.8.4.0

WhereScape RED Installation and Administration Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Copyright Notice

Copyright © 2002-2015 WhereScape Software Limited. All rights reserved. This document may be redistributed in its entirety and in this electronic or printed form only without permission; all other uses of this document and the information it contains require the explicit written permission of WhereScape Software limited.

Due to continued product development this information may change without notice. WhereScape Software Limited does not warrant that this document is error-free.

Trademarks

WhereScape and WhereScape RED are trademarks or registered trademarks of WhereScape Software Limited. Other brands or product names are trademarks or registered trademarks of their respective companies.

WhereScape USA, Inc

1915 NW AmberGlen Parkway
Suite 400, Beaverton
Oregon 97006
United States
T: 503-466-3979
F: 503-466-3978

WhereScape Limited

P.O.Box 56569, Auckland 1446
Level 3, WhereScape Towers
38 Wyndham Street
Auckland 1010, New Zealand
T: +64-9-358-5678
T: US toll free 1-877-237-3980
F: +64-9-358-5679

WhereScape Europe

Reading Enterprise Centre
Earley Gate, Whiteknights Rd
Reading RG6 6BU
United Kingdom
T: +44-118-914-4509
F: +44-118-914-4508

WhereScape Asia Pte. Ltd

300 Tampines Avenue 5
#09-02 Singapore 529653
T: +65-6679-5728

TABLE OF CONTENTS

1. Overview	5
1.1 Documentation Roadmap	6
1.2 How to use this Manual	7
2. Installing WhereScape RED	9
2.1 Installing WhereScape RED from the Command Line	14
3. Adding Your License	15
4. Checking Your Windows Version	18
5. Checking Your ODBC Drivers	19
6. Creating an ODBC Connection	21
7. SQL Server Installation and Configuration	25
7.1 Installation and Configuration Introduction	25
7.2 Installing RED for SQL Server	26
7.3 SQL Server Database and Login Setup	27
7.4 Configuring a SQL Server Metadata Connection (DSN)	28
7.5 Creating a Metadata Repository for SQL Server	34
7.6 Installing a Windows Scheduler for SQL Server	36
7.7 SQL Server Quick Start	42
7.8 SQL Server Quick Application	45
8. Teradata Installation and Configuration	50
8.1 Installation and Configuration Introduction	50
8.2 Installing RED for Teradata	52
8.3 Creating a Teradata Environment	53
8.4 Configuring a Teradata Metadata Connection (DSN)	54
8.5 Creating a Metadata Repository for Teradata	58
8.6 Installing a Windows Scheduler for Teradata	60
8.7 Installing a Unix Scheduler for Teradata	65
8.8 Installing a Teradata Unix Loader	71
9. Oracle Installation and Configuration	74

9.1 Installation and Configuration Introduction	74
9.2 Installing RED for Oracle	76
9.3 Creating an Oracle User	78
9.4 Configuring an Oracle Metadata Connection (DSN)	84
9.5 Validating Oracle User Privileges	92
9.6 Creating a Metadata Repository for Oracle	96
9.7 Installing a Windows Scheduler for Oracle	99
9.8 Installing a Unix Scheduler for Oracle	106
10. DB2 Installation and Configuration	115
10.1 Installation and Configuration Introduction	115
10.2 Installing RED for DB2	117
10.3 Creating a DB2 Schema	118
10.4 Configuring a DB2 Metadata Connection (DSN)	122
10.5 Creating a Metadata Repository for DB2	124
10.6 Installing a Windows Scheduler for DB2	126
10.7 Installing a Unix Scheduler for DB2	130
11. Tutorial Data	136
11.1 Creating Tutorial Data	137
11.2 Validating Tutorial Data	138
12. Applications	140
12.1 Loading an Application	142
12.2 Creating and Loading Applications from the Command Line	153
12.3 Changing the Application Directory	155
12.4 Listing Applications	157
12.5 Testing Applications	158
12.6 XML Files for Application Loads	159
13. Upgrading WhereScape RED	162
14. Checking the Unix Connection	166
15. Testing an ODBC Source	168
16. Validating a Meta Repository	170
16.1 Validating a Metadata Repository using the Command Line	172
17. Maintaining the Windows Scheduler	174

18. Languages	177
18.1 Loading Languages	178
18.2 Creating a Language File	179
18.3 Changing the Languages Directory	180
19. Tools	181
19.1 Start/Stop logging	181
Index	i

CHAPTER 1 OVERVIEW

It is envisaged that WhereScape RED will be used by experienced data warehouse designers and implementers.

WhereScape RED provides an environment to facilitate the rapid production of prototype data warehouse analysis areas.

It also provides the functionality to move that prototype into a production environment and support the day-to-day running of a data warehouse.

This online help version of WhereScape RED's Setup Administrator documentation is for RED 6.8.4.0, last updated Friday, July 3, 2015.

IN THIS CHAPTER

1.1 Documentation Roadmap	6
1.2 How to use this Manual.....	7

1.1 DOCUMENTATION ROADMAP

The WhereScape RED **Installation and Administration Guide**, the **online help**, and the WhereScape RED **User Guide** assume that you are proficient in the use of the Windows operating system.

WhereScape RED often provides multiple ways to accomplish a task. In some cases, you can use the main menu, the right-click menu, or a toolbar, or a key combination (e.g. Alt/M and Ctrl/M to raise menus).

Instructions in this documentation generally include only the most convenient method of accomplishing a task.

The following sources of information are available with WhereScape RED:

WhereScape RED Installation and Administration Guide

The **Installation and Administration Guide** is available either as online help, as a PDF, or in printed format. The Installation and Administration Guide provides the information needed to:

- Install the WhereScape RED software
- Validate the various software components required by WhereScape RED
- Install the WhereScape RED metadata
- Install a Scheduler
- Optionally install third-party data warehouse applications.

WhereScape RED User Guide

The WhereScape RED **User Guide** is available either as online help, as a PDF, or in a printed manual. The User Guide provides information on how to use WhereScape RED to build a data warehouse. It includes a set of tutorials to assist in the familiarization of the terms and methodologies embodied in WhereScape RED.

WhereScape RED Teradata User Guide

The WhereScape RED **Teradata User Guide** is available either as online help, as a PDF, or in a printed manual. The Teradata User Guide provides information on how to use WhereScape RED to build a data warehouse in Teradata. It includes a set of tutorials to assist in the familiarization of the terms and methodologies embodied in WhereScape RED.

Sql Admin User Guide

The **Sql Admin User Guide** is available either as online help, or as a PDF. It provides documentation in the use of the stand alone SQL query tool shipped as part of the WhereScape RED product.

WhereScape Forum

A **web forum** is available at <http://www.wherescape.com>. This forum contains information on the latest version, and bug reports that may be relevant for installation. In addition the WhereScape Blog is available at <http://www.wherescape.com> which may provide additional information.

1.2 HOW TO USE THIS MANUAL

WhereScape RED is designed to build data warehouse analysis areas within a database environment. The installation and configuration process is very rarely a simple task as it involves a number of layered products external to WhereScape RED. For the more complex cases a number of hours or days should be set aside to complete this task. Therefore, it is important that the reader begins the installation process with the correct expectations. Highlighted below are the major steps in the installation and setup of WhereScape RED. In many environments some of these steps will have been completed independently of the installation process, or may need to be completed by a third party.

Note: This document does not cover the creation and configuration of the database required to run the data warehouse. It does not cover the procedures required to create and maintain database users; nor does it cover the installation of the database client software. Refer to the database vendor manuals for assistance.

The **Installation and Configuration** chapters provide a logical outline for the installation and setup of WhereScape RED for the following databases:

- For SQL Server, see **SQL Server Installation and Configuration** (see "**7. SQL Server Installation and Configuration**" on page 25)
- For Teradata, see **Teradata Installation and Configuration** (see "**8. Teradata Installation and Configuration**" on page 50)
- For Oracle, see **Oracle Installation and Configuration** (see "**9. Oracle Installation and Configuration**" on page 74)
- For IBM DB2, see **DB2 Installation and Configuration** (see "**10. DB2 Installation and Configuration**" on page 115)

The rest of the manual deals with the following topics:

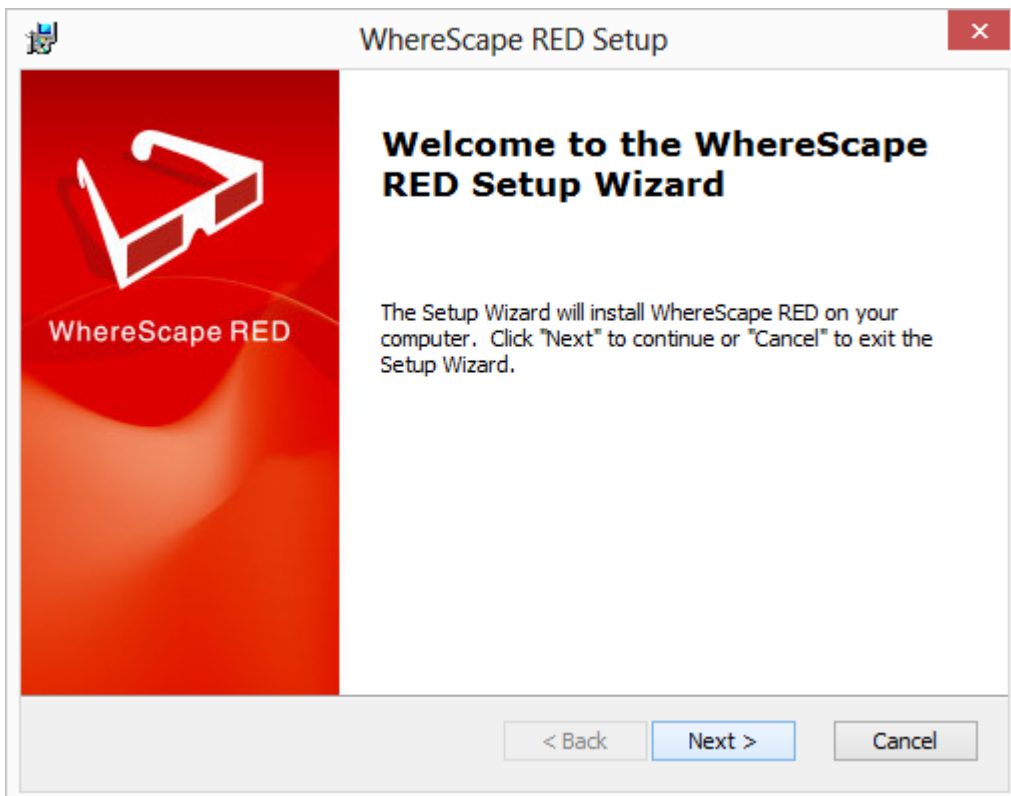
- **Tutorial Data** (see "**11. Tutorial Data**" on page 136) - creating and validating the tutorial data.
- **Applications** (see "**12. Applications**" on page 140) - loading the data warehouse applications.
- **Upgrading** (see "**13. Upgrading WhereScape RED**" on page 162) - upgrading WhereScape RED.
- **Unix Connection** (see "**14. Checking the Unix Connection**" on page 166) - checking the Unix connection.
- **Test ODBC Source** (see "**15. Testing an ODBC Source**" on page 168) - testing an ODBC source.
- **Validate Repository** (see "**16. Validating a Meta Repository**" on page 170) - validating a meta repository.
- **Maintain Scheduler** (see "**17. Maintaining the Windows Scheduler**" on page 174) - maintaining the Windows scheduler.
- **Languages** (see "**18. Languages**" on page 177) - creating and loading language files.

CHAPTER 2

2. INSTALLING WHERESCAPE RED

Use the following procedure to install the WhereScape RED software:

- 1 Stop any WhereScape RED schedulers running on the PC.
- 2 Close any other applications that are running.
- 3 Double-click the downloaded file RED_xxxx.exe (For example: the downloaded file for version 5.6.4, would be RED_56400.exe).
- 4 A dialog box displays, welcoming you to the WhereScape RED installation process. Click **Next**.

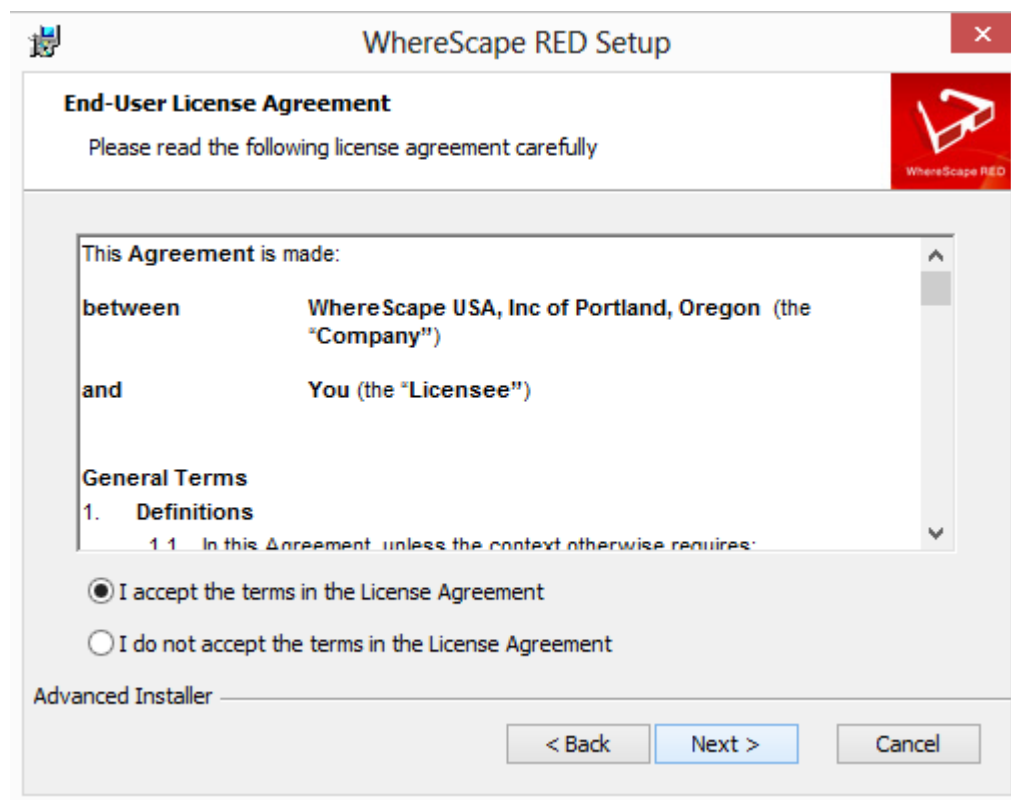


NOTE: Prerequisites Wizard screen

The installer will check for certain prerequisites that are needed to run some components of RED. If any of the prerequisites are missing, clicking **Next** will take users to the list of required prerequisites and install them.

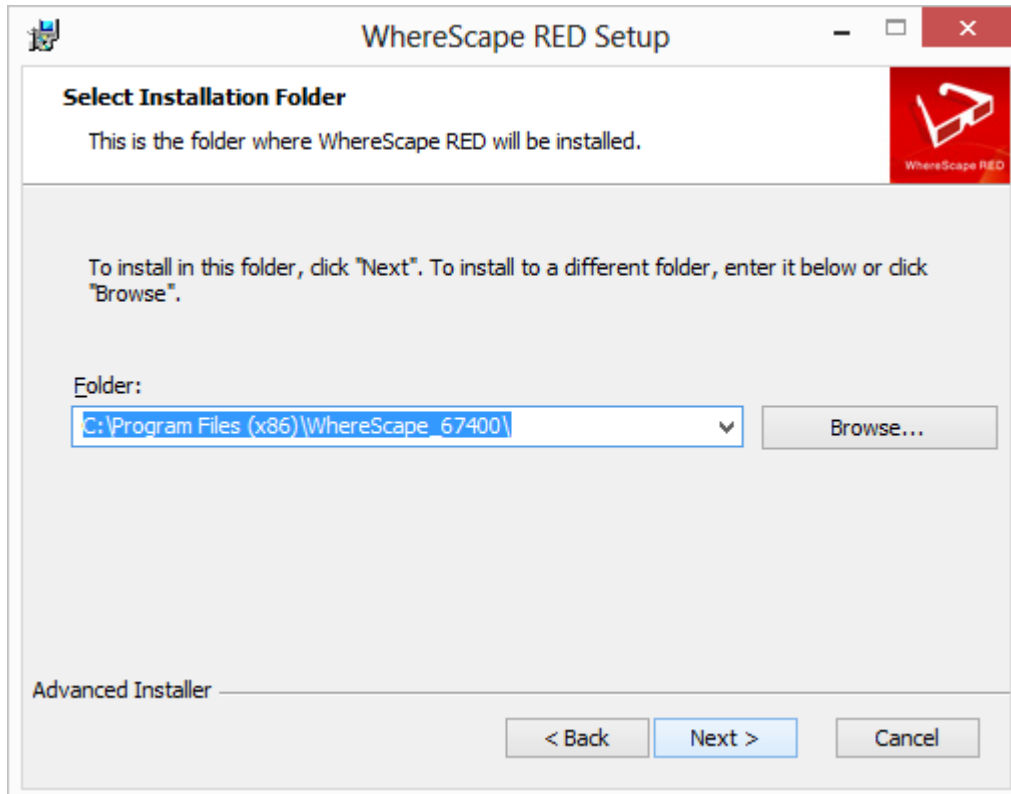
To skip this step and jump directly to the installation of RED, click **Finish**.

- 5 Read the license terms. If you do not accept the license agreement, you cannot continue with the installation. Select the **I Accept the terms in the License Agreement** option and click **Next**.



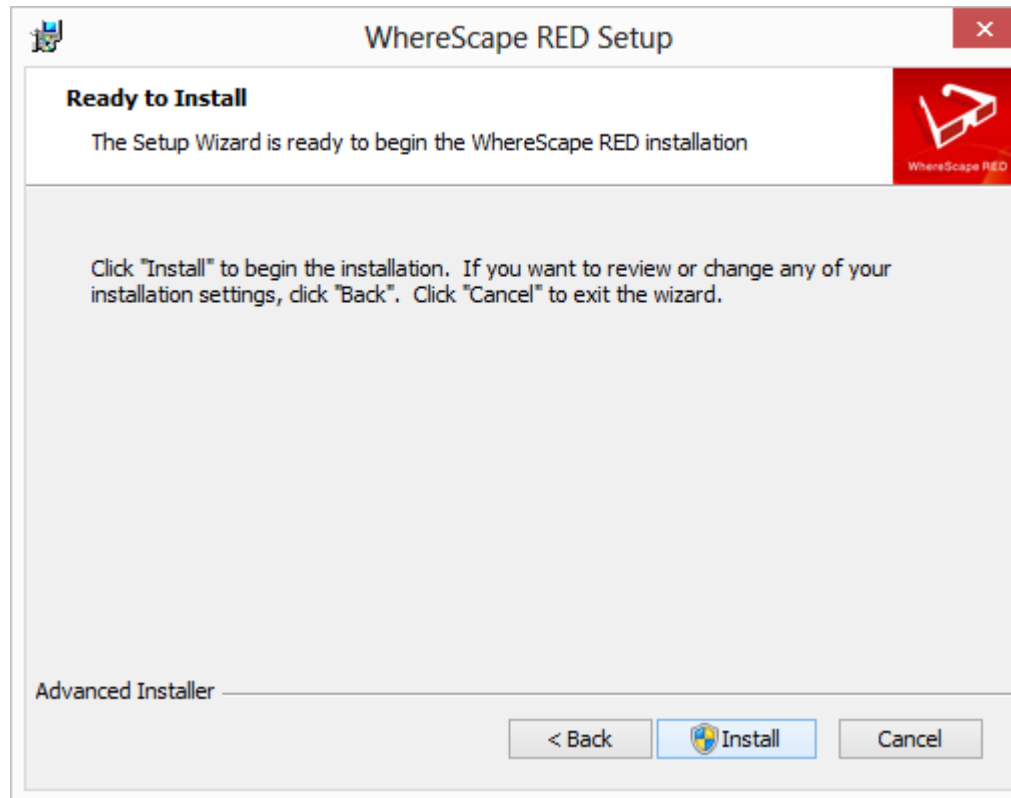
- The Installation Folder dialog box displays, showing the directory where the WhereScape RED software will be installed.

To change this directory, click **Browse...** Once you have selected the required installation folder, click **Next**.

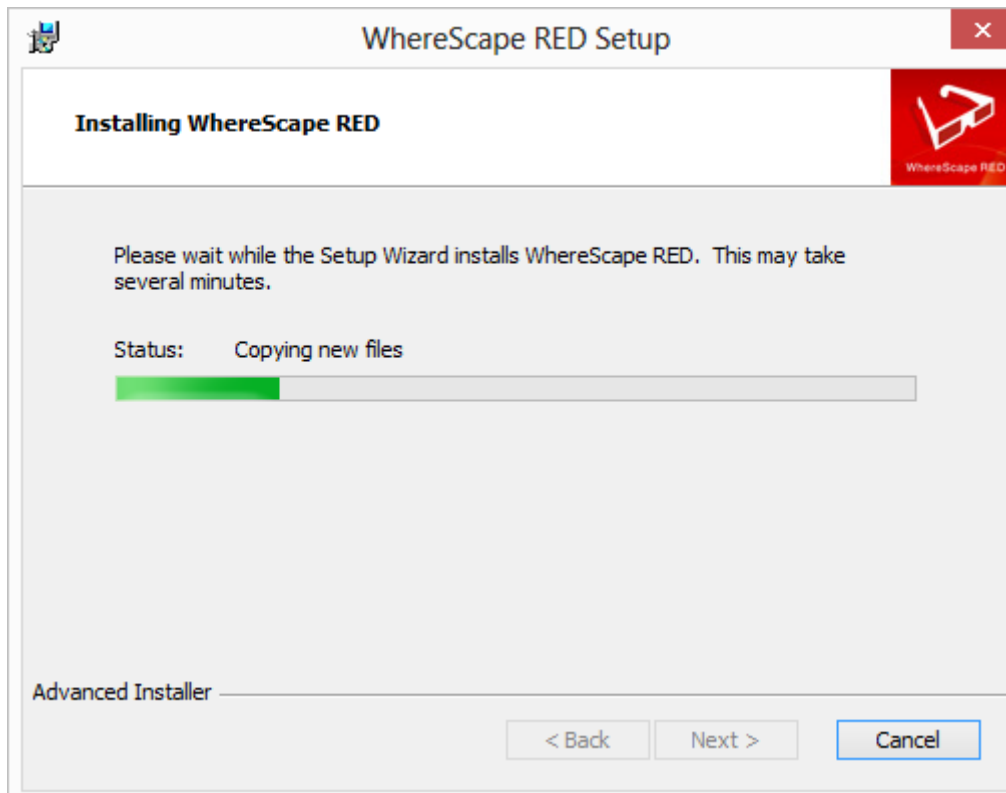


- 7 The Ready to Install dialog box displays, giving you the opportunity to go back and change any of the information you have entered so far.

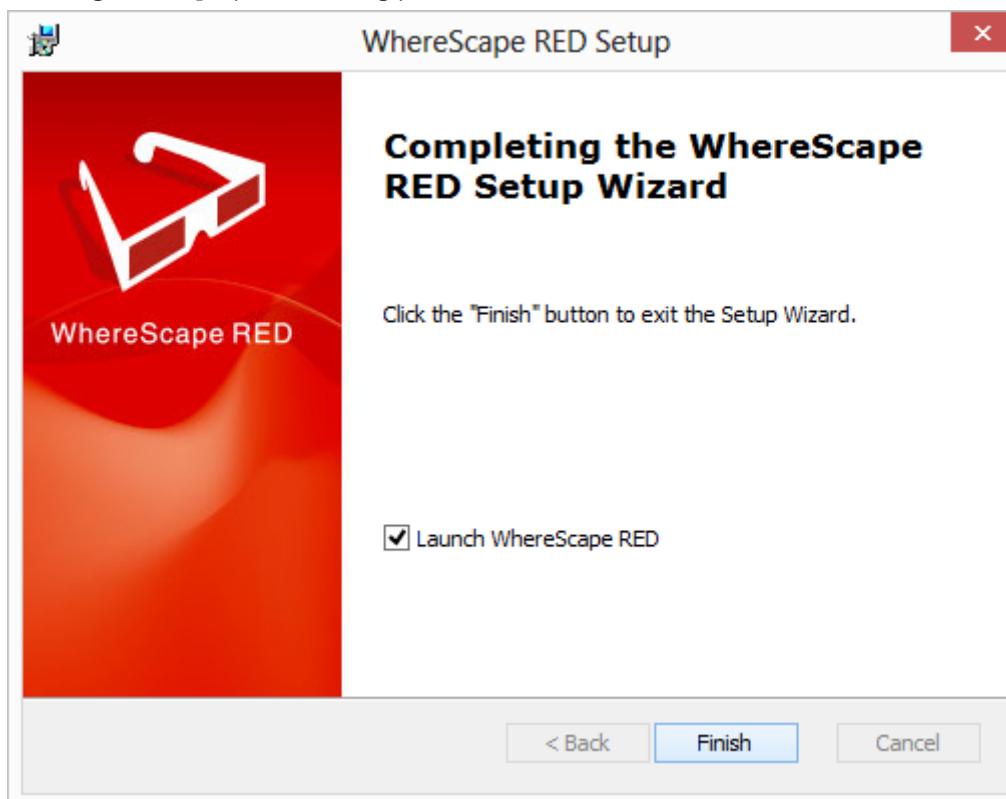
Click **Install** to install the WhereScape RED software.



- 8 The Installing dialog box displays, showing you the progress as the files are copied.



- 9 A dialog box displays, informing you that the installation was successful. Click **Finish**.



Note: If a scheduler was running or a program open (when doing an upgrade), a dialog will display, advising you that a reboot is required to finish the install.

IN THIS CHAPTER

2.1 Installing WhereScape RED from the Command Line . 14

2.1 INSTALLING WHERESCAPE RED FROM THE COMMAND LINE

It is possible to install WhereScape RED from the command line and specify a custom installation directory using the APPDIR parameter by running a bat file.

This process can be used for all metadata database types where the exe file is configured to copy the APPDIR property into all other relevant properties.

Example

The following script example will load RED with full verbose logging:

```
@echo off
cd c:\users\user\downloads\
"RED_68400.exe" APPDIR="c:\WhereScape\install\" ALLUSERS=1 RebootYesNo="No"
Reboot="Suppress" /qn+ /1*vx "c:\WhereScape\log.txt"
exit
```

Please see Advanced Installer documentation for more details about command-line options.

To install **versions of WhereScape RED older than 6.8.3.4** using the command line, please use the following script instead:

```
@echo off
cd c:\users\user\downloads\
msiexec /i "RED_68300.msi" APPDIR="c:\WhereScape\install\" ALLUSERS=1 RebootYesNo="No"
Reboot="Suppress" /qn+ /1*vx "c:\WhereScape\log.txt"
exit
```

Please see Microsoft Documentation for more details about Msiexec command-line options.

CHAPTER 3

3. ADDING YOUR LICENSE

WhereScape RED requires a valid licensee and license key. This information and key are normally supplied in a separate email.

- 1 In WhereScape's Setup Administrator, click on the **Computer Setup** tab in the left pane;
 - Then click on the **WhereScape RED License** icon.



- This menu option allows the entry of the license information. Once validated the license information is stored in the registry and validated by the WhereScape RED program on startup.

WhereScape RED License Key

WhereScape[®]
RED

Welcome to WhereScape RED.
Please enter the "Licensed To" and "License Key" information in the boxes below.
You will then be asked to confirm your license information before the key is accepted.

If you do not know your license information, please contact your site administrator or
WhereScape Software Limited sales contact.

Licensed To: Evaluate until 30-Sept-2013

License Key: ML13 PQER 123L F194 AG33 GLHN

OK Cancel

- The example above shows the entry of an evaluation license and key. If the key fails to validate contact support@wherescape.com.
 - Enter the license fields and click **OK**.

The license type dialog now displays:

This Agreement is made:

between WhereScape USA, Inc of Portland, Oregon (the "Company")

and You (the "Licensee")

General Terms

1. Definitions

1.1 In this Agreement, unless the context otherwise requires:

Confidential Information means all information exchanged between the parties to this Agreement, whether in writing, electronically or orally, including terms of this Agreement, but does not include information which is, or becomes publicly available other than through unauthorized disclosure by the other party. In the case of the Company, the term Confidential Information includes the Software and the Documentation;

Data Warehouse Repository means a single data warehouse consisting of one or more object sets (e.g. dimensional, normalized or data store) as defined in the License Fee invoice and confirmed by you when using the WhereScape RED License Key, and underlying support tables and utilities on one or more hardware devices sharing common objects (including but not limited to dimension tables) and may include copies or near copies used for development and testing purposes;

Documentation means software user guides (including those in electronic form), handbooks, maintenance libraries, educational materials and all other related material supplied by the Company to the Licensee for aiding the use of the Software;

Initial Support Period means the first 12 months Support Services, commencing on the date of the invoice for the License Fee and ceasing on the same date in the following year;

License Fee means the sum invoiced to the Licensee by the Company for the Licensee's use of the Software and for provision of Support Services for the Initial Support Period;

Non Production Purposes means the Licensee using the Software for the purposes of evaluation and testing but not production;

Proprietary Rights means all rights to patents, trade marks, service marks, trade names, inventions, trade secrets, copyright, moral rights, rights in a design, know-how, confidential information and all or any other intellectual property rights whether or not registered (together with the right to seek any such registration) and the right to enforce such rights which relate to the origin, manufacture, programming, operating and/or servicing of the Software and Documentation and any related enhancements or modifications;

Restricted Software means the Company's limited functionality version of the Software;

Software means the object code version of the WhereScape RED software application;

Support Agreement means the software support agreement set out in Annex A;

ACCEPT DECLINE

- 4 Click **ACCEPT** to confirm the type of license.

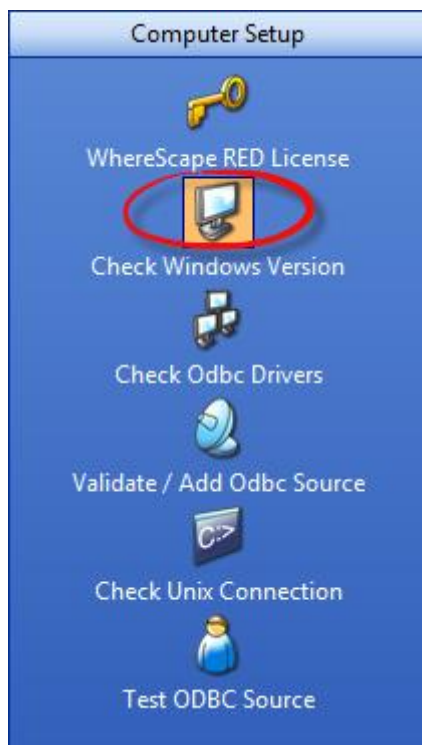
CHAPTER 4

4. CHECKING YOUR WINDOWS VERSION

WhereScape RED requires a Windows 2008, Windows 2008 R2, Windows Vista, Windows 7 or Windows 8 environment with a screen size set to at least 1024x768. This step simply validates that the above requirements have been met. The WhereScape RED front end will run in other Windows environments, but the scheduler may not. WhereScape RED will not run under Windows 95, 98 or NT. Please note that if the minimum screen size of 1024x768 is used then the standard font size must also be used.

- 1 In WhereScape's Setup Administrator, click on the **Computer Setup** tab in the left pane.

 - Then click on the **Check Windows Version** icon.



CHAPTER 5

5. CHECKING YOUR ODBC DRIVERS

Both the SQL Server and Microsoft for Oracle ODBC drivers are shipped as normal in Windows 2008, Windows 2008 R2, Windows Vista, Windows 7 and Windows 8. This step will usually be a validation of the pre-installed drivers.

WhereScape RED connects to the data warehouse database using ODBC. This step checks that the ODBC driver is present and up to date.

- 1 In WhereScape's Setup Administrator, click on the **Computer Setup** tab in the left pane.
 - Then click on the **Check Odbc Drivers** icon.



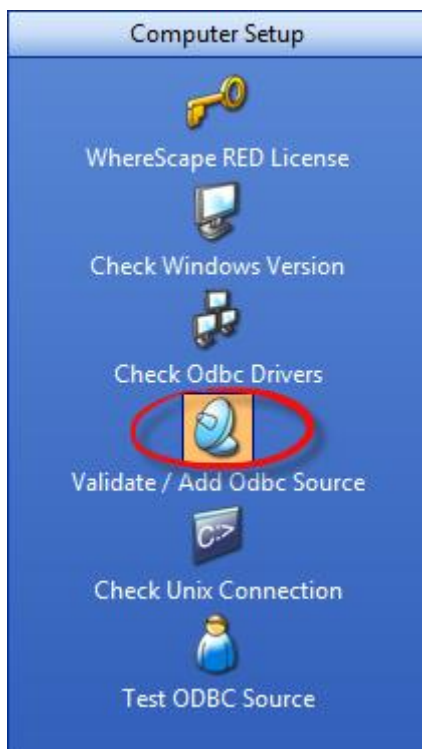
- A list of ODBC drivers will be displayed provided the computer user has enough privileges to access the registry.

ODBC Driver			
Driver Name	Driver File	Version	Status
Microsoft Access-Treiber (*.mdb)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Driver do Microsoft Paradox (*.db)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Driver do Microsoft Excel(*.xls)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft Text Driver (*.txt; *.csv)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Driver da Microsoft para arquivos texto (...)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft dBase-Treiber (*.dbf)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
SQL Server	C:\Windows\system32\SQLSRV32.dll	6.2.9200.16384 (...)	OK
Microsoft Excel Driver (*.xls)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Driver do Microsoft dBase (*.dbf)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft Paradox-Treiber (*.db)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft ODBC for Oracle	C:\Windows\system32\msorc132.dll	6.2.9200.16384 (...)	OK
Microsoft Text-Treiber (*.txt; *.csv)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft Excel-Treiber (*.xls)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft Access Driver (*.mdb)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Driver do Microsoft Access (*.mdb)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft Paradox Driver (*.db)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Microsoft dBase Driver (*.dbf)	C:\Windows\system32\odbcjt32.dll	6.2.9200.16384 (...)	
Teradata	C:\Program Files (x86)\Teradata\Client\...	14.10.00.00	OK
NetezzaSQL	C:\Windows\SysWOW64\nsqlodbc.dll	7.0.2.27689	
IBM DB2 ODBC DRIVER - DB2COPY1	C:\PROGRA~1\IBM\SQLLIB\BIN\DB2CL...		
IBM DB2 ODBC DRIVER	C:\Program Files\IBM\SQLLIB\BIN\DB2...		
PostgreSQL ANSI	C:\Program Files (x86)\Greenplum\gree...	08.04.0200	
PostgreSQL Unicode	C:\Program Files (x86)\Greenplum\gree...	08.04.0200	
SQL Server Native Client 11.0	C:\Windows\SysWOW64\sqlncli11.dll	2011.0110.3000.0...	OK
Oracle in OraClient11g_home1_32bit	C:\oracle\product\11.2.0\client_2\BIN\...	11.2.0.3.0	OK
Hortonworks Hive ODBC Driver	C:\Program Files (x86)\Hortonworks Hi...		

CHAPTER 6

6. CREATING AN ODBC CONNECTION

- 1 In WhereScape's Setup Administrator, click on the **Computer Setup** tab in the left pane.
 - Then click on the **Validate/ Add Odbc Source** icon.



2 An entry needs to exist in the ODBC tables for the data warehouse database. This step lists all entries found that use an ODBC driver. If no entry exists for the data warehouse database then you will need to add one.

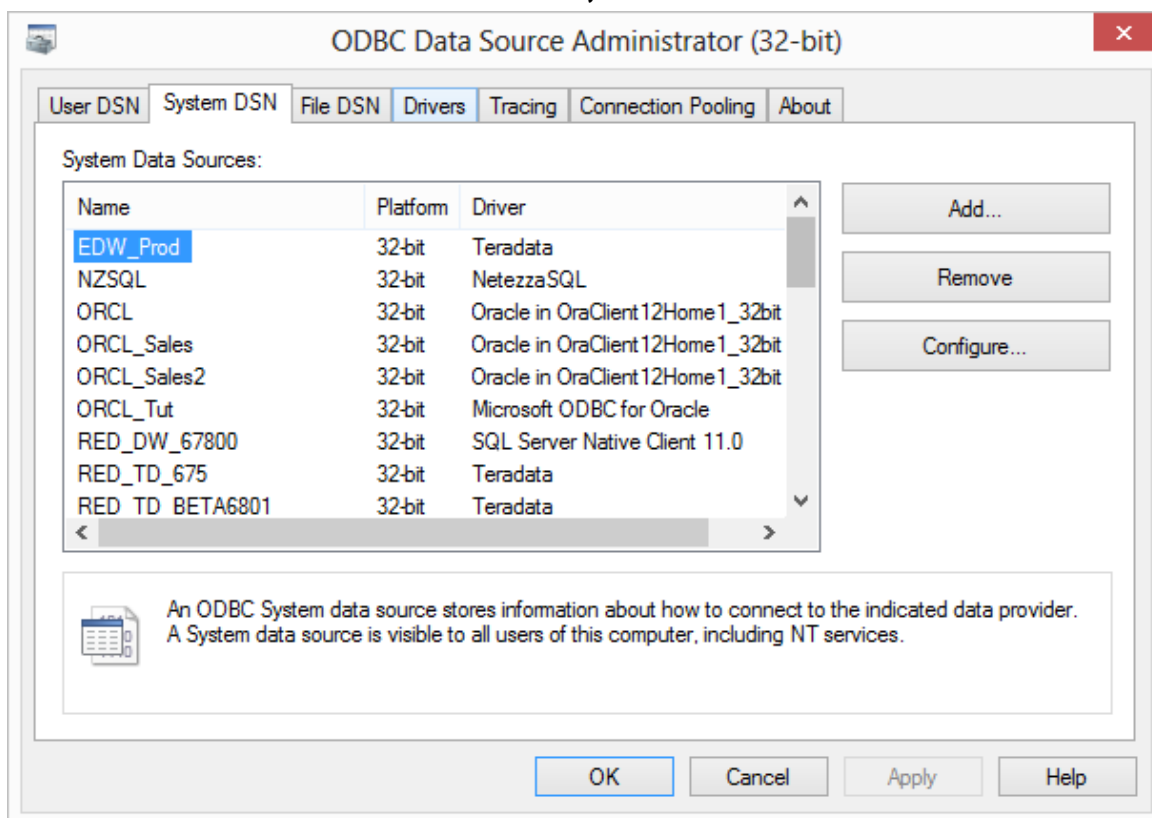
- Right-click in the anywhere on the right frame to bring up the **Launch ODBC Data Source Administrator** menu option.

The screenshot shows the ODBC Source Administrator window. On the left is a 'Computer Setup' sidebar with various options. The main area is titled 'ODBC Source' and contains a table with three columns: 'Odbc Connectio...', 'Driver Name', and 'Driver File'. Below the table is a button labeled 'Launch ODBC Data Source Administrator'.

Odbc Connectio...	Driver Name	Driver File
NZSQL	NetezzaSQL	C:\Windows\SysWOW64\nsqlo...
Sample Hortonw...	Hortonworks Hive ODBC Driver	C:\Program Files (x86)\Hortonw...
WSCAPE	IBM DB2 ODBC DRIVER - DB2C...	C:\PROGRA~1\IBM\SQLLIB\BIN...
WSCAPE_Tut	IBM DB2 ODBC DRIVER - DB2C...	C:\PROGRA~1\IBM\SQLLIB\BIN...
WSCAPE_Sales	IBM DB2 ODBC DRIVER - DB2C...	C:\PROGRA~1\IBM\SQLLIB\BIN...
WSCAPE_Sales2	IBM DB2 ODBC DRIVER - DB2C...	C:\PROGRA~1\IBM\SQLLIB\BIN...
ORCL_Tut	Microsoft ODBC for Oracle	C:\Windows\system32\msorcl3...
test1	SQL Server	C:\Windows\system32\SQLSRV...
tutorial1	SQL Server Native Client 11.0	C:\Windows\SysWOW64\sqlIncli...
td_dbc	Teradata	C:\Program Files (x86)\Teradata...
WslWarehouse	SQL Server Native Client 11.0	C:\Windows\SysWOW64\sqlIncli...
Sales_Forecasts	Microsoft Excel Driver (*.xls)	C:\Windows\system32\odbcjt3...
WslWarehouse O...	Microsoft ODBC for Oracle	C:\Windows\system32\msorcl3...
Sales	SQL Server Native Client 11.0	C:\Windows\SysWOW64\sqlIncli...
Sales2	SQL Server Native Client 11.0	C:\Windows\SysWOW64\sqlIncli...
WslWarehouse_TD	Teradata	C:\Program Files (x86)\Teradata...
Sales New ORCL	Microsoft ODBC for Oracle	C:\Windows\system32\msorcl3...
Sales2 New OCRL	Microsoft ODBC for Oracle	C:\Windows\system32\msorcl3...
EDW_Prod	SQL Server Native Client 11.0	C:\Windows\SysWOW64\sqlIncli...
tdSales	Teradata	C:\Program Files (x86)\Teradata...
TD_14_00	Teradata	C:\Program Files (x86)\Teradata...

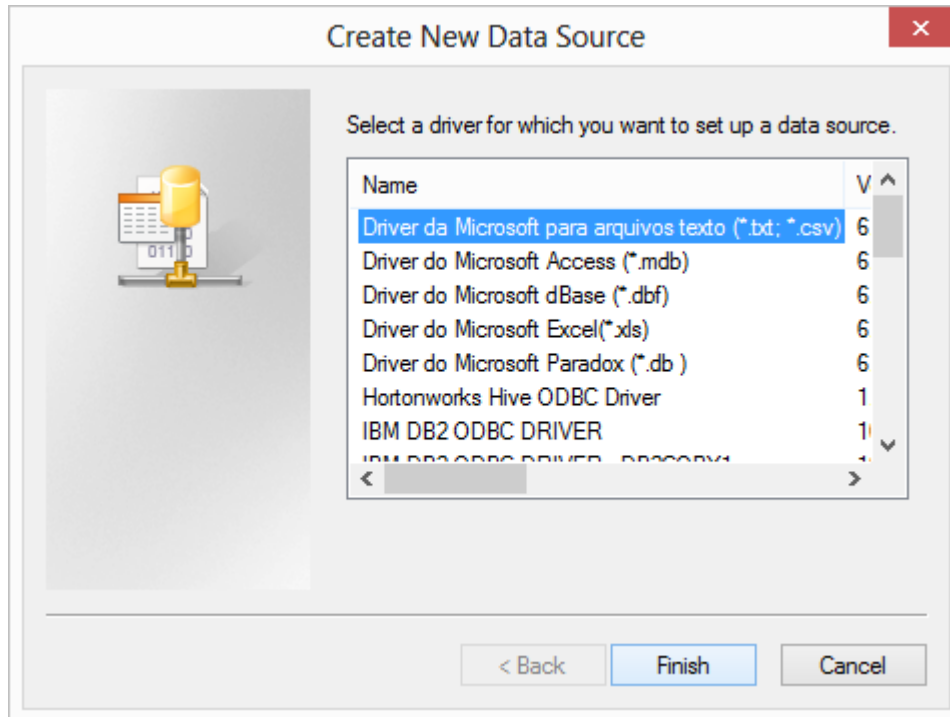
Launch ODBC Data Source Administrator

- 3 Click on the **System DSN** tab to view the system wide ODBC entries.
 - Click on the 'Add...' button to add a new entry.



NOTE: ODBC connections must be either **User DSN** or **System DSN**. **File DSN** connections are **not** supported.

- 4 Select the relevant driver and click **Finish**.



- 5 Enter the relevant details to create a new ODBC data source.

CHAPTER 7

7. SQL SERVER INSTALLATION AND CONFIGURATION

IN THIS CHAPTER

7.1 Installation and Configuration Introduction	25
7.2 Installing RED for SQL Server	26
7.3 SQL Server Database and Login Setup	27
7.4 Configuring a SQL Server Metadata Connection (DSN)	28
7.5 Creating a Metadata Repository for SQL Server	34
7.6 Installing a Windows Scheduler for SQL Server	36
7.7 SQL Server Quick Start	42
7.8 SQL Server Quick Application	45

7.1 INSTALLATION AND CONFIGURATION INTRODUCTION

The steps provided here are an indication of the normal process required to install and configure WhereScape RED.

In some environments additional processes will be required.

- 1 Install the WhereScape RED software onto your PC. See **Installing RED for SQL Server** (see "7.2 Installing RED for SQL Server" on page 26)
- 2 Add your license key. See **Adding your License** (see "3. Adding Your License" on page 15)
- 3 Check your Windows version. See **Checking your Windows Version** (see "4. Checking Your Windows Version" on page 18)
- 4 Check your ODBC Drivers. See **Checking your ODBC Drivers** (see "5. Checking Your ODBC Drivers" on page 19)
- 5 Designate a database that is to be used as the data warehouse database. See **Creating a SQL Server Database** (see "7.3.2 SQL Server Login Setup" on page 27)
- 6 Add the definition of an ODBC data source to connect to the data warehouse database. See **Configuring a SQL Server Metadata Connection (DSN)** (see "7.4 Configuring a SQL Server Metadata Connection (DSN)" on page 28)
- 7 Create the WhereScape metadata repository. See **Creating a Metadata Repository for SQL Server** (see "7.5 Creating a Metadata Repository for SQL Server" on page 34)
- 8 Install a Windows Scheduler. See **Installing a Windows Scheduler for SQL Server** (see "7.6 Installing a Windows Scheduler for SQL Server" on page 36)
- 9 If required, load the WhereScape tutorial tables into a separate database. It is highly recommended that the tutorials be undertaken. See **Creating Tutorial Data** (see "11.1 Creating Tutorial Data" on page 137)

Note: A Quickstart option is available for SQL Server. This option will perform all the required tasks (except the entering of a license key) when provided with a valid database server.

Problem Solving

If any problems happen in your install that cannot be resolved proceed as follows when running WhereScape's Setup Administrator utility:

- 1 Create a log file by selecting **Log/Start Log** from the menu bar. Enter a file name.
- 2 Step through at least:
 - Computer Setup --> Check Windows version
 - Computer Setup --> Check ODBC driver
 - Computer Setup --> Validate/Add Odbc source
 - Create/Validate Meta Repository --> Validate meta repository
- 3 Select **Log/Stop Log** to close off the log
- 4 Edit the log and remove any passwords that appear
- 5 Send your log file to support@wherescape.com

7.2 INSTALLING RED FOR SQL SERVER

Installation Prerequisites

WhereScape RED requires certain client and server components.

Client Prerequisites

- Windows Vista, Windows 7 or Windows 8.
- 250MB available disk space.
- ODBC Version 3.0 (normally shipped as part of the Operating System).
- For building a data warehouse, the appropriate Microsoft ODBC driver for SQL Server is required.

Server Prerequisites

- One of the following data warehouse database platforms is required: Microsoft SQL Server 2005, 2008, 2008R2, 2012, 2012R2 or 2014.
- To use the WhereScape RED Scheduler, a UNIX, Linux or Windows based server is also required (this may be the database server). WhereScape RED only supports sh and ksh under UNIX and bash under Linux.

Installation of WhereScape RED

To install WhereScape RED, see *Installing WhereScape RED* (see "2. Installing WhereScape RED" on page 9)

7.3 SQL SERVER DATABASE AND LOGIN SETUP

7.3.1 SQL SERVER DATABASE

Create a SQL Server database to be used as the RED metadata repository and data warehouse. To use object placement across multiple schemas in RED, create the required schemas in the SQL database.

NOTE: The database name used for the RED repository must follow SQL Server rules for "Regular Identifiers". For example, the database name should not contain any spaces. Please refer to SQL Server documentation for more details on rules for regular identifiers.

7.3.2 SQL SERVER LOGIN SETUP

Create a SQL Server login that will be used to set up the RED metadata repository and for access by the RED scheduler service.

The minimum SQL Server permissions are as follows:

Server Level Permissions

- **bulkadmin** - this role is required if you want to load files or use Native ODBC loads.

Permissions to kill a process when killing jobs in the scheduler

- VIEW SERVER STATE
- ALTER ANY CONNECTION

SQL Server 2012 roles

For SQL Server 2012 and later only, users can create server roles and grant them to the RED scheduler and RED users.

Database Permissions

Database roles (on the database with the RED metadata in it)

- db_datareader
- db_datawriter
- db_ddladmin

Database grants (on the database with the RED metadata in it)

- SELECT
- INSERT
- UPDATE
- DELETE
- EXECUTE

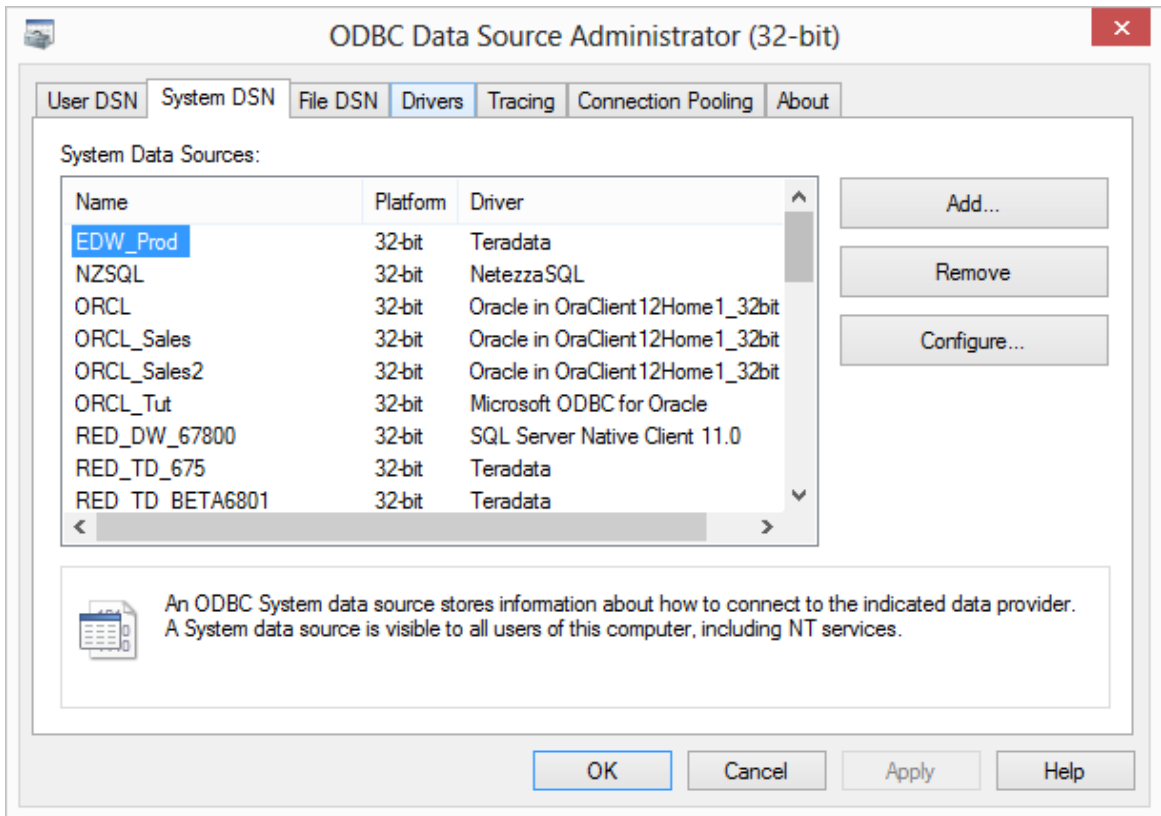
NOTE: When creating a new repository or validating the metadata in an existing repository, the SQL Server login used must have SYSADMIN or the ALTER TRACE permission. This is because almost all the WhereScape procedures have a catch all with log file option exception handler. This ensures that in the event that the scheduler cannot log errors to the database, it can log errors to SQL Server logs and Application logs.

The SQL Server login for RED must have a default schema of **dbo**.

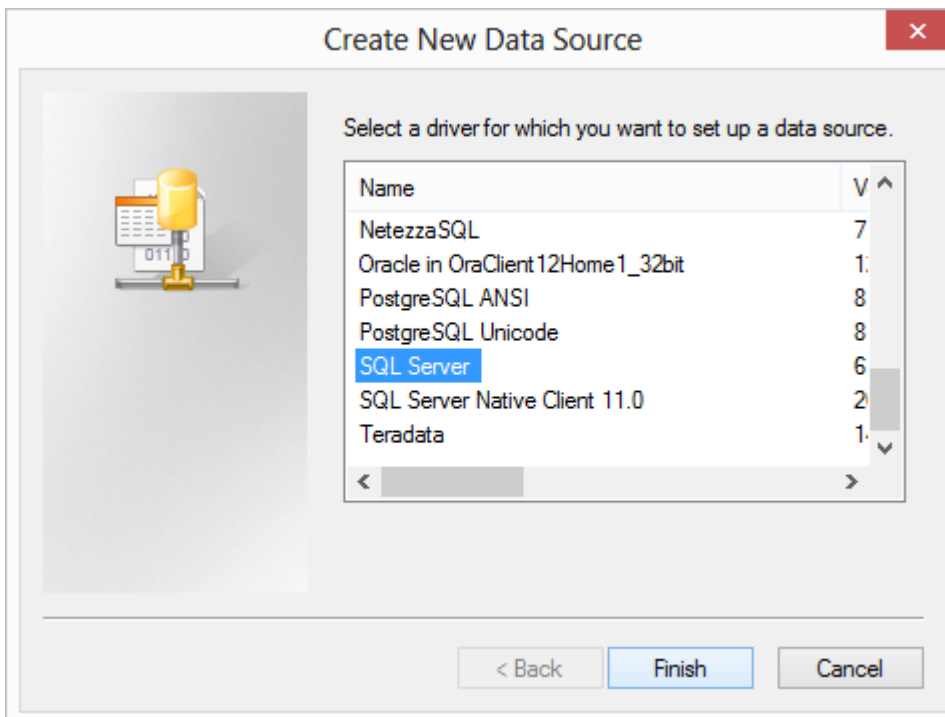
7.4 CONFIGURING A SQL SERVER METADATA CONNECTION (DSN)

To begin adding an ODBC Source from WhereScape's Setup Administrator, see *Creating an ODBC Connection* (see "6. *Creating an ODBC Connection*" on page 21)

- 1 In ODBC Data Source Administrator, click on the **System DSN** tab to view the system wide ODBC entries.
 - Click the 'Add...' button to add a new entry.



- 2 Select a **SQL Server** driver from the driver list.



- 3 Enter a **Name** for the ODBC Source and select or enter the **Server**.
 - Click **Next**.

Create a New Data Source to SQL Server

This wizard will help you create an ODBC data source that you can use to connect to SQL Server.

What name do you want to use to refer to the data source?

Name:

How do you want to describe the data source?

Description:

Which SQL Server do you want to connect to?

Server:

- 4 Another pop-up will appear with authentication details.
 - Change as required or leave as shown if you are unsure.
 - Click **Next**.

Create a New Data Source to SQL Server

How should SQL Server verify the authenticity of the login ID?

With Windows NT authentication using the network login ID.

With SQL Server authentication using a login ID and password entered by the user.

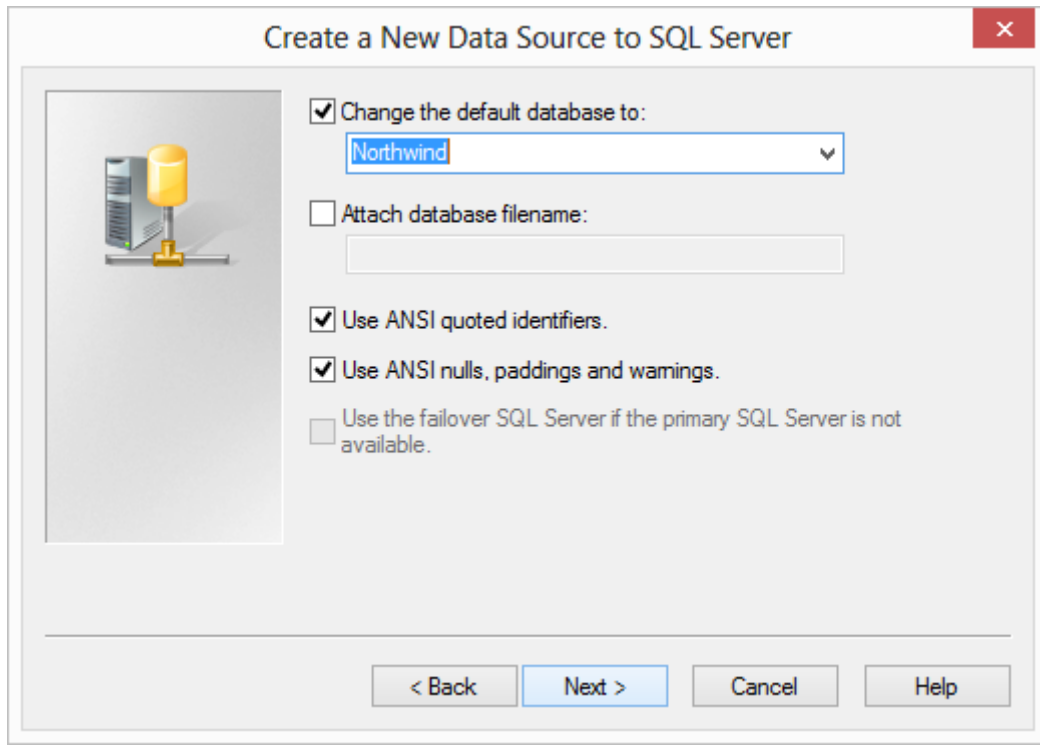
To change the network library used to communicate with SQL Server, click Client Configuration.

Connect to SQL Server to obtain default settings for the additional configuration options.

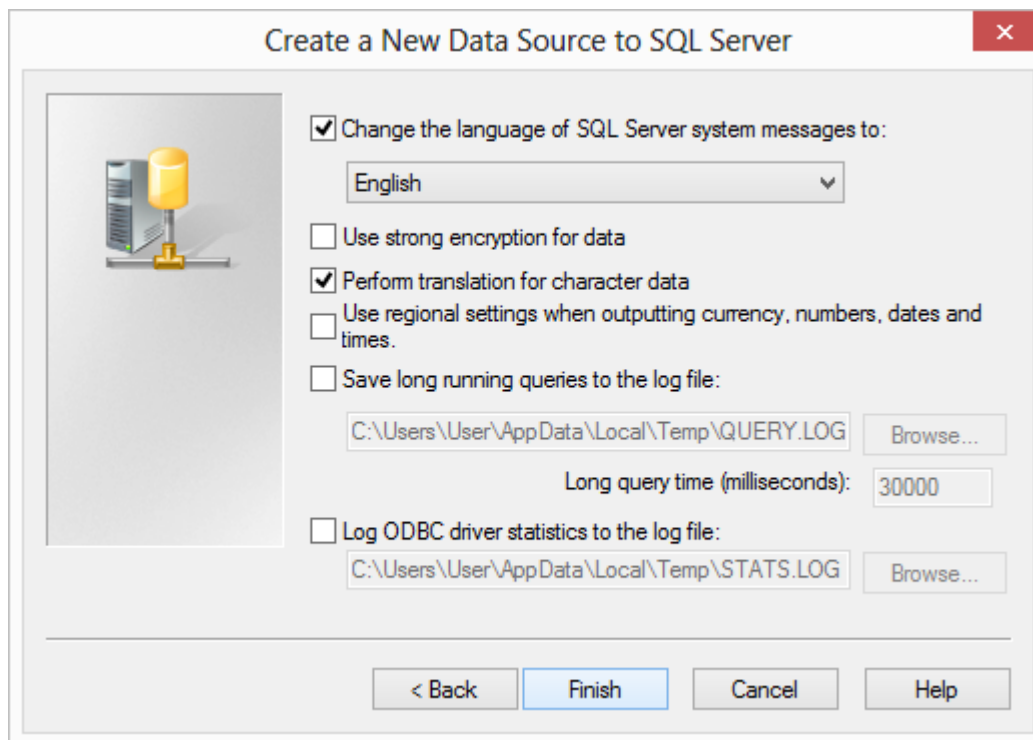
Login ID:

Password:

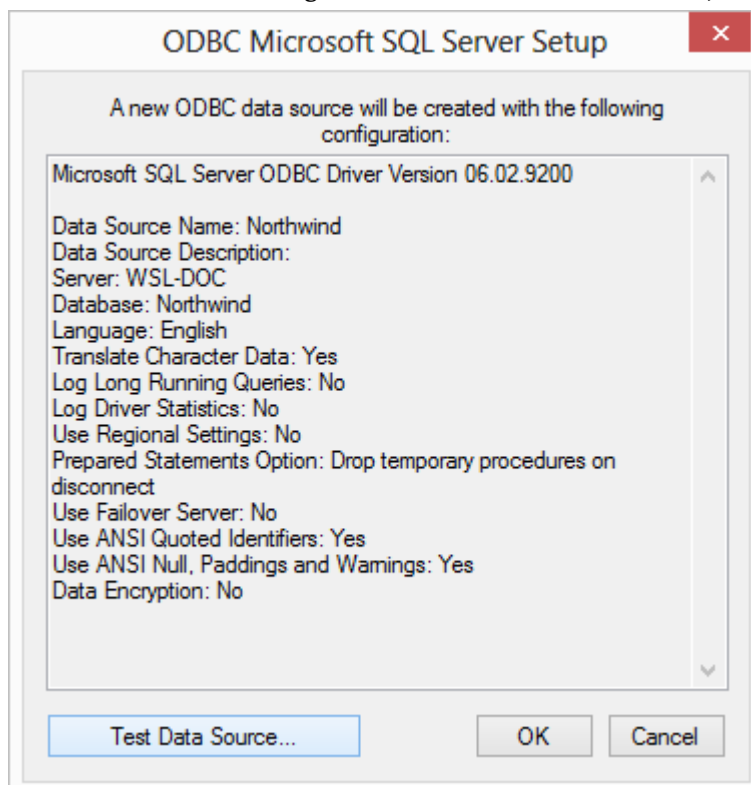
- 5 The next pop-up will provide the opportunity to select a database. The following example shows the selection of the Northwind database.
 - Select a database from the drop-down-list and click **Next**.



- The last pop-up allows setting of other features if desired.
 - Ensure the **regional settings** check-box is **not used**. If this is used then date and number formats will be in regional standards rather than native form.
 - Click **Finish**.



- Test the data source using the 'Test Data Source...' button, otherwise just click **OK**.





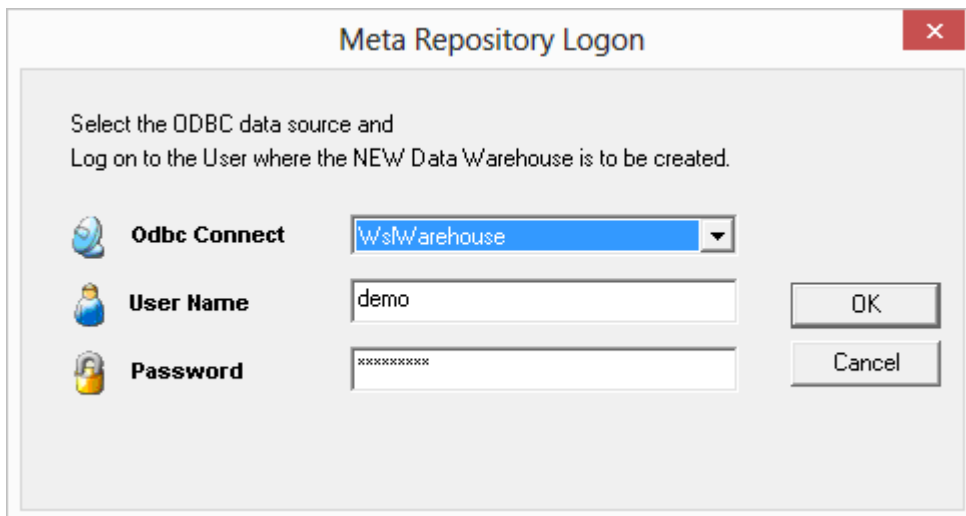
TIP: If multiple PCs will access the data warehouse they must all use the same data source name as entered here.

7.5 CREATING A METADATA REPOSITORY FOR SQL SERVER

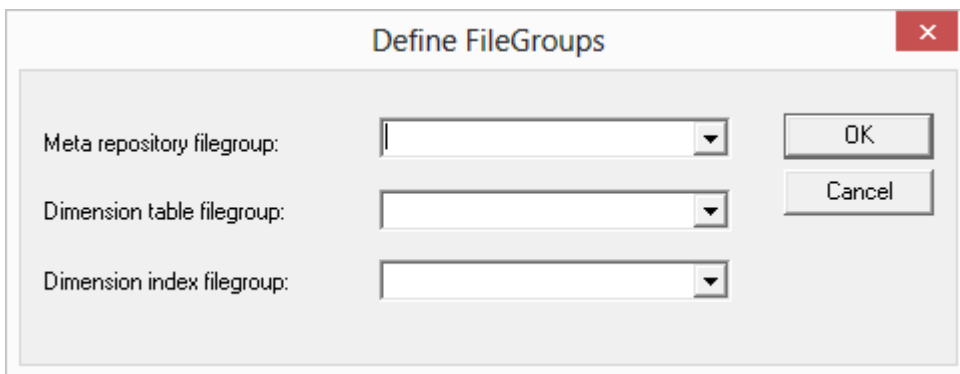
- 1 In WhereScape's Setup Administrator, click on the **Metadata Repository** tab in the left pane.
 - Then click on the **Create Metadata Repository** icon.



- 2 This option will create the WhereScape meta tables, sequences and procedures in the specified schema. It checks to ensure that the schema, database or user does not contain an existing metadata repository before starting the metadata install.
 - Click on **Create Metadata Repository**.
 - Enter the **Odbc Connect**, **User Name** and **Password** and click **OK**.

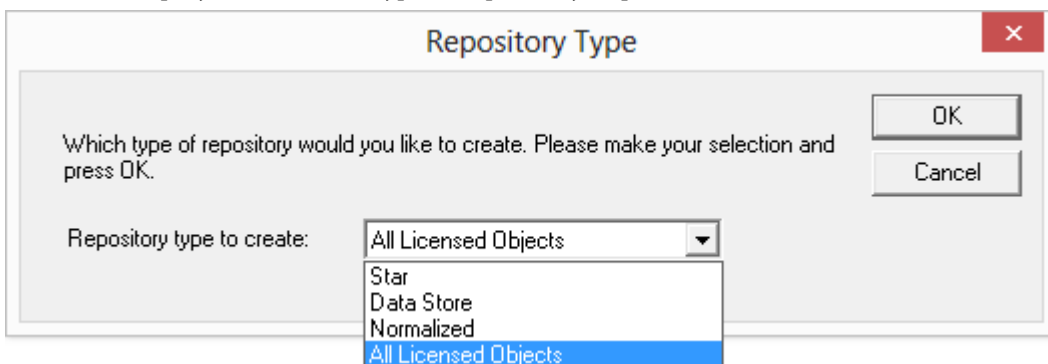


- When creating a SQL Server metadata repository, an additional screen is displayed requesting **Filegroup** information.
 - Select the required **Filegroups** and click **OK**.
 - If Filegroups are not being used, leave the Filegroups fields blank and click **OK**.



The 'Define FileGroups' dialog box contains three dropdown menus for selecting filegroups: 'Meta repository filegroup:', 'Dimension table filegroup:', and 'Dimension index filegroup:'. To the right of these fields are 'OK' and 'Cancel' buttons.

- Depending on the type of license you have, the following dialog may be displayed:
 - If it is displayed, select the type of repository required and click **OK**.



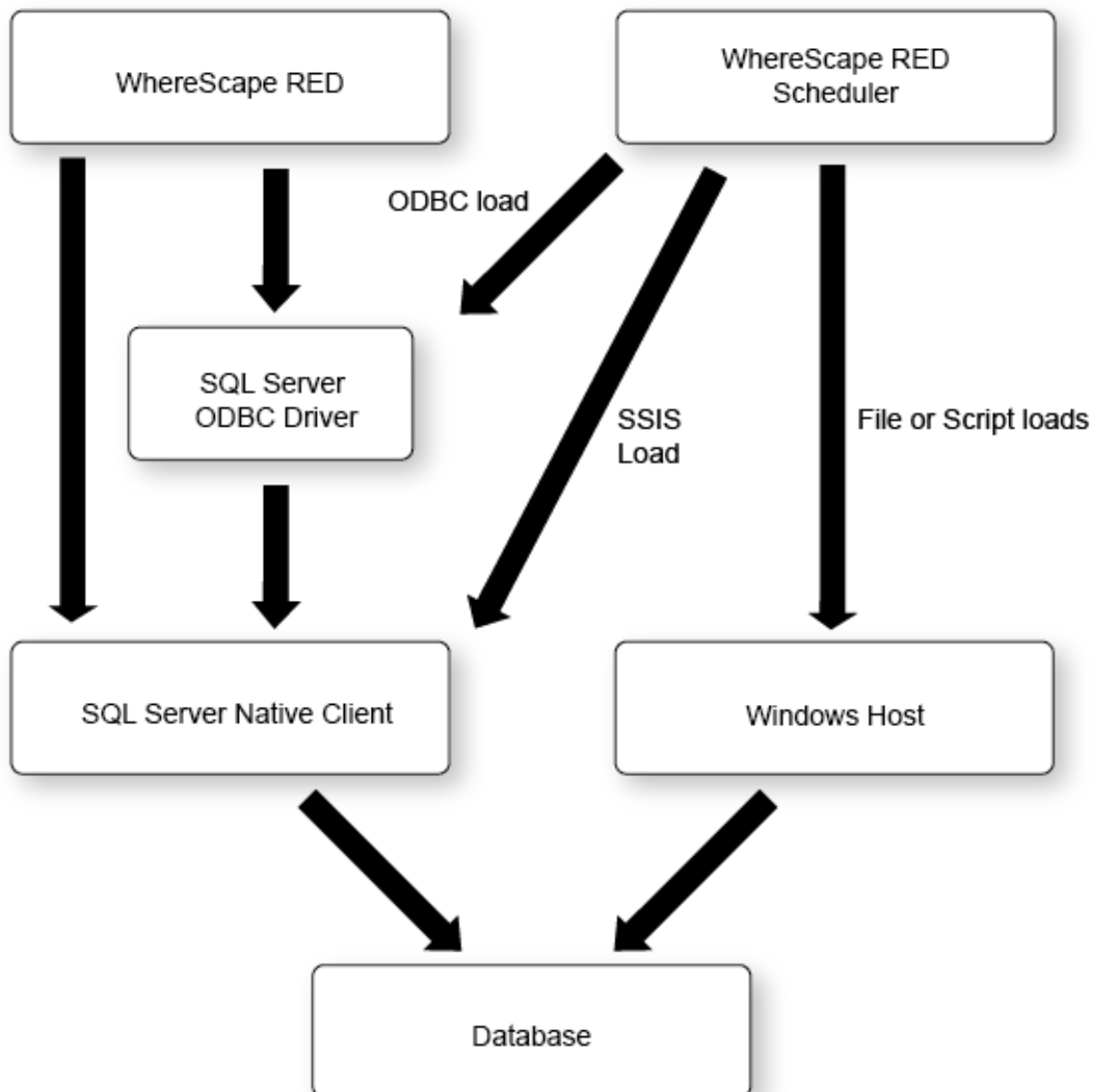
The 'Repository Type' dialog box contains a text prompt: 'Which type of repository would you like to create. Please make your selection and press OK.' Below this is a dropdown menu labeled 'Repository type to create:' with a list of options: 'All Licensed Objects', 'Star', 'Data Store', 'Normalized', and 'All Licensed Objects'. To the right are 'OK' and 'Cancel' buttons.

NOTE1: The metadata repository and data warehouse tables must both be created under the **dbo** schema when using SQL Server as the data warehouse server. Ensure that the user creating the meta repository has the required privileges to create objects in **dbo**.

NOTE2: This is just setting the default objects visible in RED. They can later be changed under **Tools/Options**.

7.6 INSTALLING A WINDOWS SCHEDULER FOR SQL SERVER

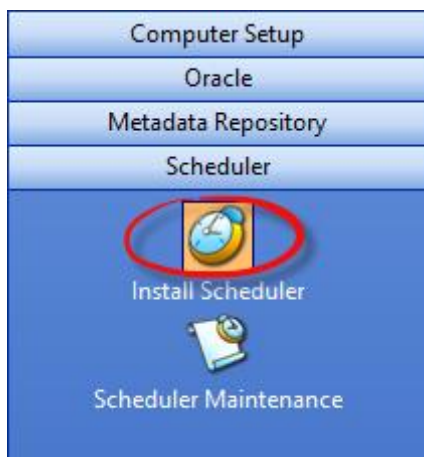
This option installs and starts a copy of the WhereScape Windows scheduler. The scheduler runs as a system service and can also be maintained through the normal Windows services functions. The following diagram provides an overview of how WhereScape RED and the scheduler access a **SQL Server** database.



The scheduler makes use of the ODBC layer to perform the routine tasks involved in scheduling and running jobs. WhereScape RED and the scheduler can also use SSIS to load data into SQL Server.

For file or script based loads, the scheduler actions the load on the host system utilizing BULK INSERT.

- 1 In WhereScape's Setup Administrator, click on the **Scheduler** tab in the left pane.
 - Then click on the **Install Scheduler** icon.



NOTE: To install or maintain the Scheduler you need to run RED's Setup Administrator as an **Administrator**.

- 2 On selection of this menu option the following dialog box will appear:

Windows Scheduler Definition

Unique Identifier (e.g. SID_user):	<input type="text" value="WsWarehouse"/>
Scheduler Name (e.g. WINnnnn):	<input type="text" value="WIN0001"/>
Service Display Name:	<input type="text" value="WhereScape WsWarehouse"/>
Service logged on under User:	<input type="text" value="System Account"/>
Work Directory:	<input type="text" value="C:\ProgramData\WhereScape\Work\"/>
Log File:	<input type="text" value="C:\ProgramData\WhereScape\Scheduler\WsISched_log"/>
Poll Interval (default 30) seconds:	<input type="text" value="30"/>
Log Level (0=none, 9=debug)	<input type="text" value="2"/>
Direct Path Load or ODBC	<input type="text" value="Odbc sql load"/>
TNS Service (Oracle):	<input type="text"/>
ODBC Connection:	<input type="text" value="WsWarehouse"/>
Database User id:	<input type="text" value="WsWarehouse"/>
Database Password:	<input type="password" value="xxxx"/>
	<input type="checkbox"/> Password is TDWallet string.
MetaData Schema (DB2):	<input type="text"/>
MetaData Database (Teradata):	<input type="text"/>

Unique Identifier

Each scheduler requires a unique name (possibly the database name and user e.g. WslWarehouse_sa). This unique name is the name by which the service is referenced.

Scheduler Name

When scheduling a job with WhereScape RED it is possible to assign the job to a specific scheduler. Each scheduler should therefore have a unique name. The default name is WIN0001. Most users change this to be the scheduler machine's name followed by a number, e.g.: DWHP01, where DWHP is the name of the windows scheduler server.

Service Display Name

This field displays the name that the scheduler will appear under when viewed through the Windows services. It is a combination of the string 'WhereScape' and the unique identifier.

Service Logged on Under User

This is the name of the Windows local user account that the scheduler service will use when logging on. The default setting is **System Account**, also known as the **Local System** service account. To set the scheduler to use a Windows domain user account with a password, use the Service Account option from the right-click menu. If a domain user account is specified then the contents of this field (on the **Windows Scheduler Definition** dialog) are ignored.

NOTE: A domain user account will require the **log on as a service** privilege in order for it to work. This is assigned automatically when setting the log on account using the Services console in Windows.

Work Directory

The directory used to store some temporary files created by the scheduler.



TIP: Two schedulers should not use the same work directory

Log File

The full path where the scheduler service will write a log of the various actions undertaken. This file may get large if log level 9 is set.

Poll Interval

This defines the frequency of the scheduler polling the database, looking for work. The default is 30 seconds. If set to a very low interval such as 1 then the impact on the database may be noticeable.

Log Level

This defines the level of detail written to the log file. The following logging levels are available:

- 0 = will result in no information being written.
- 1 = fatal errors only
- 2 = major events
- 3 = major events and return stats to caller
- 9 = full debug



TIP: Setting Log Level 3 will return timing stats during ODBC based loads

Direct Path load or Odbc load

This should always be **ODBC sql load** for SQL Server schedulers.

TNS Service

This is not set for SQL Server schedulers.

ODBC Connection

The ODBC connection field requires the ODBC source name that will provide a connection to the data warehouse system. The scheduler uses ODBC to connect to the data warehouse.

Database User Id

For Windows Authentication, leave this field empty. For SQL Server Authentication, enter a database username.

Database Password

For Windows Authentication, leave this field empty. For SQL Server Authentication, enter the password for the database username entered in the **Database User Id** field.

Metadata Schema (DB2) / Metadata Database (Teradata)

This is not set for SQL Server schedulers.

NOTE: A windows scheduler must be run as either the system service account or as a user with administrator privileges in windows.

7.7 SQL SERVER QUICK START

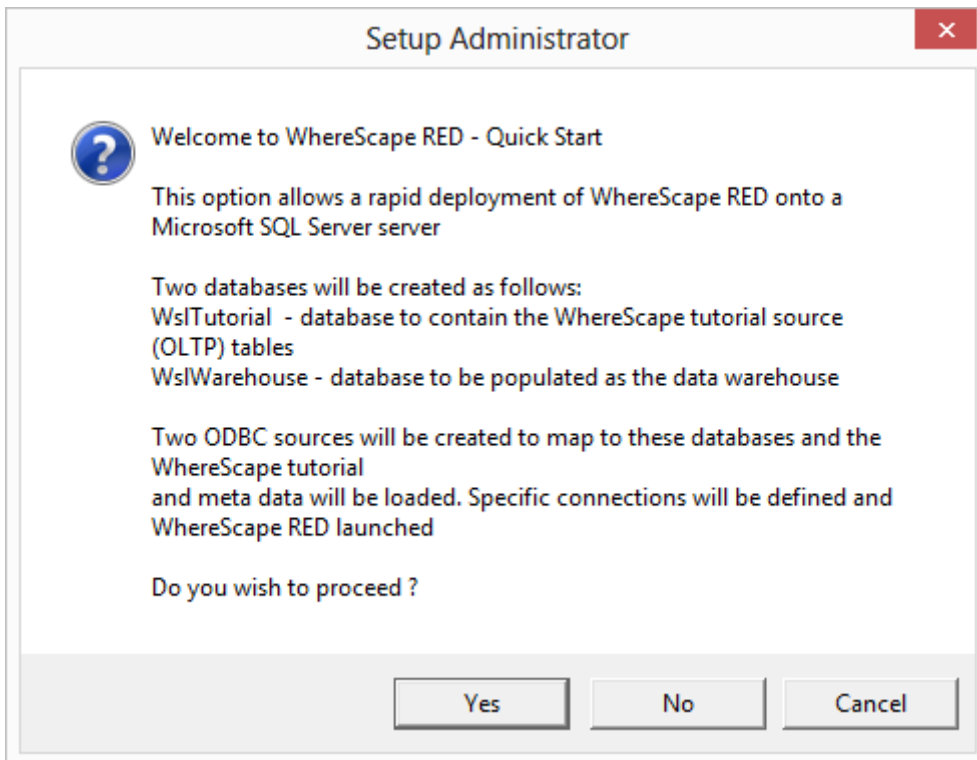
The quick start component of WhereScape's Setup Administrator attempts to do a rapid install of WhereScape RED onto a SQL Server database server.

It creates two databases on this server and then installs the RED software and the tutorial; finally invoking WhereScape RED and the on-line tutorial help.

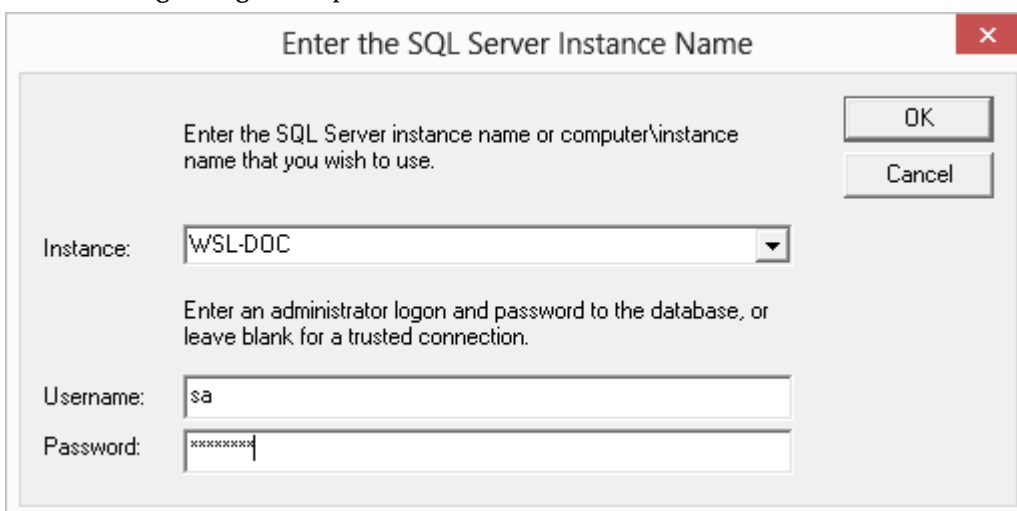
- 1 On the **Quick Start** tab, click on the **Auto install Metadata** icon to start.



- The following dialog is displayed.
 - Click **Yes** to proceed with the creation of the databases.



- You will first be asked for the SQL Server instance in which to load the metadata repository and tutorial.
The following dialog box is presented.



RED will attempt to locate all SQL Server servers available and list them in the drop-down. This process can fail for a number of reasons in which case the **Instance** name will need to be ascertained and entered.

Leave the **Username** and **Password** blank if using a trusted connection and no password is required. Click **OK** when complete.

Once the install has completed RED will allow you to logon to the newly created data warehouse.

7.8 SQL SERVER QUICK APPLICATION

The quick application option for SQL Server attempts to create a Data Warehouse database, populate that database with the WhereScape metadata, load an application, install a scheduler and optionally run a batch job.

It provides a means of quickly installing a RED application and getting the contents of the tables loaded and populated.

- 1 On the **Quick Start** tab, click on the **Auto Install Metadata and Application** icon to start.



2 The following dialog is displayed.

- Click **Yes** to proceed.



3 The following dialog is displayed.

- Once all the values have been entered, click **OK** to proceed with the creation of the databases and the loading of the application.

Choose an Application and Database Name etc

A WhereScape RED data warehouse repository will be created in a new database on the chosen instance of SQL Server. An application will be loaded into this repository and all the objects created.

A scheduler for this repository will be installed on the local machine and a job queued to run. An ODBC source DSN will also be created to be used by the data warehouse source connection.

Select an instance of Sql Server in which to install the new data warehouse.

WSL-DOC

Enter an administrator logon and password to the database, or leave blank for a trusted connection.

Username: sa

Password: wsl

Enter the database name which will become the new data warehouse repository.

newDW

Please enter the location and name of the application id file (app_id_*.wst)

C:\ProgramData\WhereScape\Application\app_id_SQLT_tutorial_1.wst Browse

Enter the name of a job to queue to run.

Daily Update

An ODBC source (DSN) to be used by the data warehouse source connection will be created.

Enter the ODBC source DSN.

WslTutorial

Select the instance for the database containing the source data.

WSL-DOC

Enter the source database name.

WslTutorial OK

Trusted connection Cancel

The quick application can also be initiated from the command line or from a Windows batch file.

Windows Command Line Invocation

The syntax for calling from the command line is as follows:

```
adm.exe -QA [-SWITCH "switch value"]
```

The switches that can be applied supply information to the Setup Administrator utility. When all switches are provided the application install will occur automatically. Where some switches are left out a partially completed dialog will appear. The missing information can be provided and the install will then proceed.

Note: Where a parameter for a switch contains spaces, the parameter must be enclosed in double quotes.

The switches are:

Switch	Parameter to be supplied
-QA	Indicates that this is a "Quick Application". No parameter is required.
-SS	The SQL Server instance name where the Data Warehouse database will reside
-SL	The administrator user name for the SQL Server instance defined with the -SS switch. If this switch is not provided a Trusted connection is assumed.
-SP	The administrator password for the SQL Server instance defined with the -SS switch.
-SN	The Data Warehouse database name. This database will be created in the instance defined by -SS. An ODBC Source DSN of the same name will also be created on the PC.
-AN	The RED application name. This is the name of the 'id' file and is typically of the structure app_id_XXXX_vvvvvv.wst where XXXX is the application identifier and vvvvvv is the application version. For example the tutorial application would be app_id_SQLT_tutorial_6.wst
-AD	The RED application directory. This is the directory in which the file defined by the -AN switch resides.
-LI	The SQL Server instance that contains the source database.
-LD	The ODBC source DSN name as defined in the Connection object in the data warehouse application. This ODBC source will be created.
-LN	The database name on the SQL Server instance as defined by the -LI switch. This is the database that contains the source data.

-LT	This switch does not have a parameter. If set it flags the ODBC source DSN as defined in the -LD switch as a trusted connection.
-JN	The name of a job in the data warehouse that is to be run once all other tasks are completed.
-CC	The name of the cube connection if appropriate. If set then all cubes will be updated to have this as the Analysis Services connection.
-DC	The name of the Data Warehouse connection if appropriate. If set then all cubes will be updated to have this as their Source connection.
-CD	The name of the cube database in Analysis Services if appropriate. If set then all cubes will be updated to have this database as their Analysis Services database.

For Example a batch file to call an application load of the tutorial may look as follows:

```
@echo off
c:\
cd \Program Files\WhereScape\
adm -QA -SS "WSL-TEST" -SN new_dw -LT -AN app_id_SQLT_tutorial_3.wst -AD "c:\Program
Files\WhereScape\" -LI WSL-TEST -LD tutorial -LN tutorial -JN "Daily Update" -CC Cubes -CD
My_database -DC DataWarehouse
exit
```

CHAPTER 8

8. TERADATA INSTALLATION AND CONFIGURATION

IN THIS CHAPTER

8.1 Installation and Configuration Introduction	50
8.2 Installing RED for Teradata.....	52
8.3 Creating a Teradata Environment	53
8.4 Configuring a Teradata Metadata Connection (DSN) ..	54
8.5 Creating a Metadata Repository for Teradata	58
8.6 Installing a Windows Scheduler for Teradata	60
8.7 Installing a Unix Scheduler for Teradata	65
8.8 Installing a Teradata Unix Loader	71

8.1 INSTALLATION AND CONFIGURATION INTRODUCTION

The steps provided here are an indication of the normal process required to install and configure WhereScape RED. In some environments additional processes will be required.

- 1 Install Teradata Tools and Utilities 13.10 (or greater) or on a suitable computer. These tools and utility components are required:
 - CLI
 - Teradata ODBC Driver
 - BTEQ
 - Fast Load / Multiload or TPT
 - Fast Export

- 2 Install the WhereScape RED software onto your PC. See *Installing RED for Teradata* (see "8.2 *Installing RED for Teradata*" on page 52)
- 3 Add your license key. See *Adding your License* (see "3. *Adding Your License*" on page 15)
- 4 Check your Windows version. See *Checking your Windows Version* (see "4. *Checking Your Windows Version*" on page 18)
- 5 Check your ODBC Drivers. See *Checking your ODBC Drivers* (see "5. *Checking Your ODBC Drivers*" on page 19)
- 6 Designate a database or user that is to be used as the meta data database. For testing or evaluation purposes this database or user may be a Teradata Express database server. WhereScape RED requires the Teradata database version 13 or greater as the data warehouse database server. Record the database and server names for this database. Create a database user to hold the metadata tables.
- 7 Allocate the privileges required by WhereScape RED See *Creating a Teradata Environment* (see "8.3 *Creating a Teradata Environment*" on page 53).
- 8 Add appropriate entries to the hosts file on the computer for the server.
- 9 Add a definition of an ODBC data source to connect to the data warehouse database. See *Configuring a Teradata Metadata Connection (DSN)* (see "8.4 *Configuring a Teradata Metadata Connection (DSN)*" on page 54)
- 10 Create the WhereScape metadata repository under the data warehouse user that you created. See *Creating a Metadata Repository for Teradata* (see "8.5 *Creating a Metadata Repository for Teradata*" on page 58)
- 11 Install a Windows Scheduler. See *Installing a Windows Scheduler for Teradata* (see "8.6 *Installing a Windows Scheduler for Teradata*" on page 60)
- 12 Install a Unix Scheduler. See *Installing a Unix Scheduler for Teradata* (see "8.7 *Installing a Unix Scheduler for Teradata*" on page 65)
- 13 Install a Teradata Unix Loader. See *Installing a Teradata Unix Loader* (see "8.8 *Installing a Teradata Unix Loader*" on page 71)
- 14 If required, load the WhereScape tutorial tables into a separate database user. It is highly recommended that the tutorials be undertaken. See *Create Tutorial Data* (see "11.1 *Creating Tutorial Data*" on page 137)

Problem Solving

If problems occur in your install that cannot be resolved, proceed as follows when running WhereScape's Setup Administrator utility:

- 1 Create a log file by selecting **Log/Start Log** from the menu bar. Enter a file name.
- 2 Step through at least:
 - Computer Setup --> Check Windows version
 - Computer Setup --> Check ODBC driver
 - Computer Setup --> Validate/Add Odbc source
 - Create/Validate Meta Repository --> Validate meta repository
- 3 Select **Log/Stop Log** to close off the log
- 4 Edit the log and remove any passwords that appear.
- 5 Send your log file to support@wherescape.com

8.2 INSTALLING RED FOR TERADATA

Installation Prerequisites

WhereScape RED requires certain client and server components.

Client Prerequisites

- Windows Vista, Windows 7 or Windows 8
- 250MB available disk space
- ODBC Version 3.0 (normally shipped as part of the Operating System).
- For building a data warehouse, Teradata Tools and Utilities 13.10 or greater is required.

Server Prerequisites

- One of the following data warehouse database platforms is required: 13.10, 14.x or 15.
- To use the WhereScape RED Scheduler, a UNIX, LINUX or Windows based server is also required. WhereScape RED only supports sh and ksh under Unix and bash under Linux.
- When installing a Unix/Linux scheduler, the server will need to have Teradata Tools and Utilities for Linux/UNIX installed and configured as a prerequisite before the scheduler can be installed.

Installation of WhereScape RED

To install WhereScape RED, see *Installing WhereScape RED* (see "2. Installing WhereScape RED" on page 9)

8.3 CREATING A TERADATA ENVIRONMENT

The creation of the Teradata repository and user(s) that will be used to administer and maintain the data warehouse must be performed outside of this utility.

The RED repository is recommended to be a Teradata database for a more secure database environment.

WhereScape recommends following Teradata best practices related to ROLES and PROFILES, but does not mandate they be used.

RED repositories require these permissions for administration:

```
GRANT EXECUTE, SELECT, INSERT, UPDATE, DELETE, STATISTICS, SHOW, TABLE, VIEW,  
MACRO ON RED_REPOSITORY TO RED_REPOSITORY;  
GRANT EXECUTE PROCEDURE ON RED_REPOSITORY TO RED_REPOSITORY WITH GRANT  
OPTION;
```

RED developers do not need TABLE or VIEW permissions to the repository database. RED developers only require INSERT, UPDATE, DELETE and SELECT on metadata repository views. Please contact WhereScape should you need assistance in configuring a Teradata secure repository environment.

For DBC permissions the following is required:

GRANT SELECT ON DBC.ErrorMsgs; – to log error messages into error logs. This is always accessed through the locking for access modifier.

GRANT SELECT ON DBC.TablesV; – to describe Tables in a database within databases the user has permission to.

GRANT SELECT ON DBC.ColumnsV; – to describe columns associated with a table within databases the user has permission to.

As an example, for our data warehouse, or decision support system, we will create a user, 'dssdemo'. An example sql script to create this 'dssdemo' user is shipped and can be found in the WhereScape RED program directory:

- ..\Teradata\Recreate_dssdemo.sql

Please note that **this script will need to be modified** to reflect the Teradata environment created at your site.

NOTE: Timezone Settings

Teradata can generate CURRENT_TIMESTAMP, CURRENT_TIME and CURRENT_DATE to reflect the session time and session time zone (as opposed to the current date and time on the Teradata server).

This can result in date/time/timestamp comparisons behaving unexpectedly and impact the RED scheduler. A Teradata Customer Service Engineer can configure this using a DBSCONTROL field called TimeDateWZControl.

The values for TimeDateWZControl are:

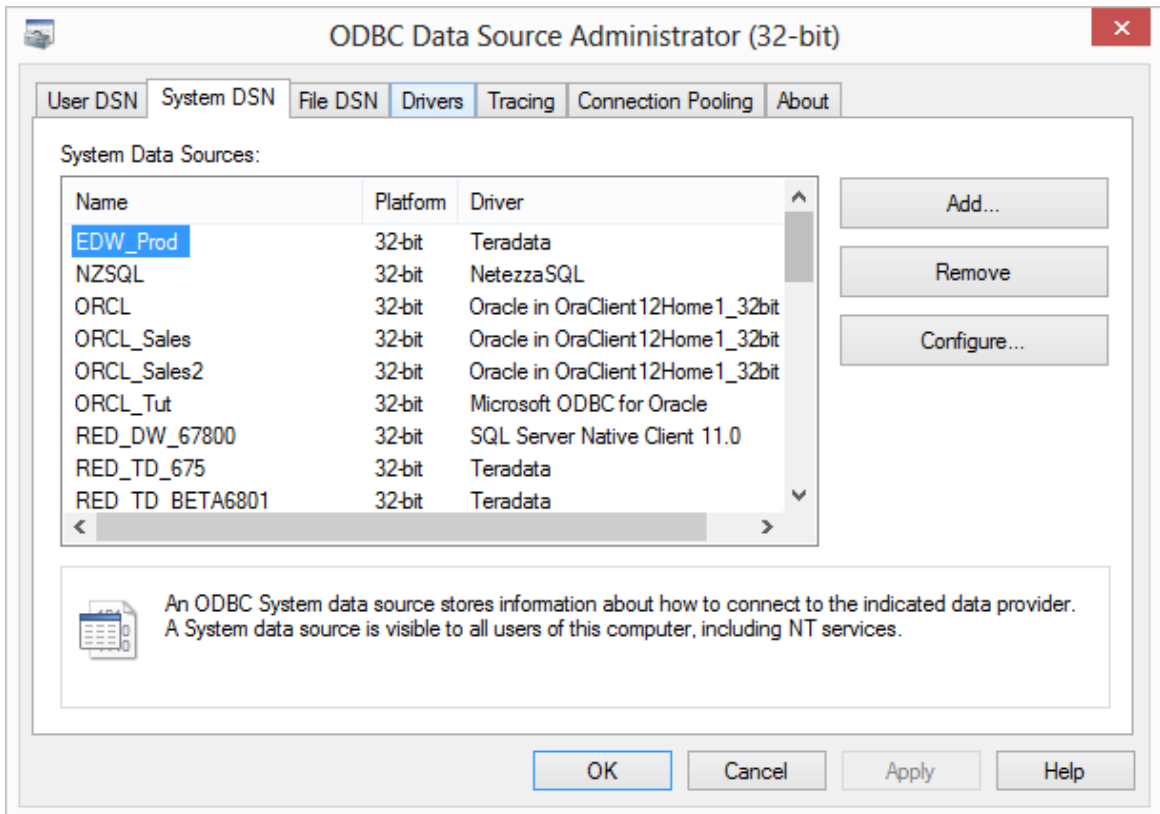
- 1 (disabled)
- 2 (enabled with UTC i.e. all datetime data is stored in UTC on disk)
- 3 (enabled with LOCAL i.e. all datetime data is stored in system LOCAL time on disk)
- 0 (returns the field to disabled)

Please ensure TimeDateWZControl is disabled if you are not using temporal fields.

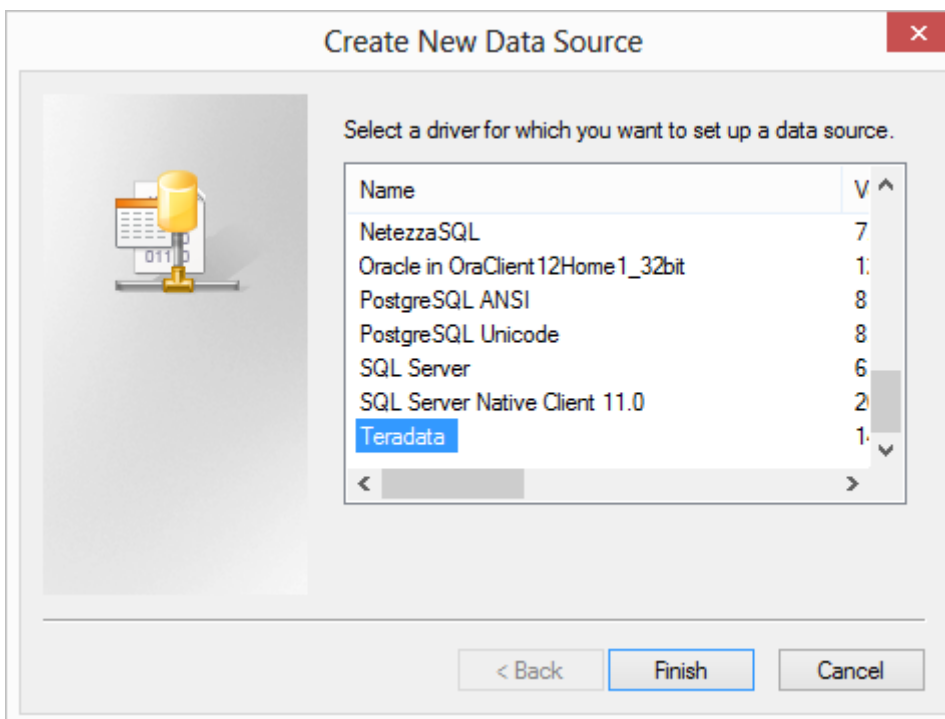
8.4 CONFIGURING A TERADATA METADATA CONNECTION (DSN)

To begin adding an ODBC Source from WhereScape's Setup Administrator, see **Creating an ODBC Connection** (see "6. Creating an ODBC Connection" on page 21).

- 1 In ODBC Data Source Administrator, click on the **System DSN** tab to view the system wide ODBC entries.
 - Click on the 'Add...' button to add a new entry.



- 2 Select a **Teradata** driver from the driver list.



- 3 A pop-up as shown below will appear.
 - Enter a **Name** for the Data Source.
 - Next, enter the **Name or IP Address** for the Teradata Server Info.
 - The **Session Character Set** field must be set to **ASCII** - this is the only character-encoding scheme supported for Teradata.

ODBC Driver Setup for Teradata Database

Data Source

Name: EDW_Prod

Description:

OK

Cancel

Help

Teradata Server Info

Name or IP address: TD_14_10

Authentication

Use Integrated Security

Mechanism:

Parameter: Change...

Username:

Password

Teradata Wallet String

Optional

Default Database:

Account String: Options >>

Session Character Set:

ASCII

NOTE: The name entered for the **Teradata Server Name** should appear in the **hosts** file on the computer in this form:

(IP address) (Teradata Server Name) (Teradata Server Name)Cop1

For the example in the ODBC screen shot above for TD_14_00, this entry would present if the server had an IP address of 10.12.123.60 and a name of "TD_14_00":

10.12.123.60 TD_14_00 TD_14_00Cop1

An alternative approach is to add the same entry to a dynamic name server (DNS).

- 4 If using the ODBC source for a RED Scheduler, click on **Run in Quiet Mode**.
 - Click **OK**.

Teradata ODBC Driver Options

Use Column Names Session Mode: System Default ▾

Disable Async. DateTime Format: AAA ▾

Use X Views Return Generated Keys: No ▾
(V2R6.2 and above)

No HELP DATABASE

Ignore Search Patterns

Enable Reconnect

Run in Quiet Mode

Disable Parsing

Log Error Events

Disable CALL to EXEC Conversion

Use Regional Settings for Decimal Symbol

Use Native Large Object Support

Return Output Parameters As ResultSet

Enable Data Encryption

Enable Extended Statement Information

OK

Cancel

Help

Warning

Changing the advanced options may cause the ODBC driver to perform improperly.

Advanced

- 5 Click **OK** again.



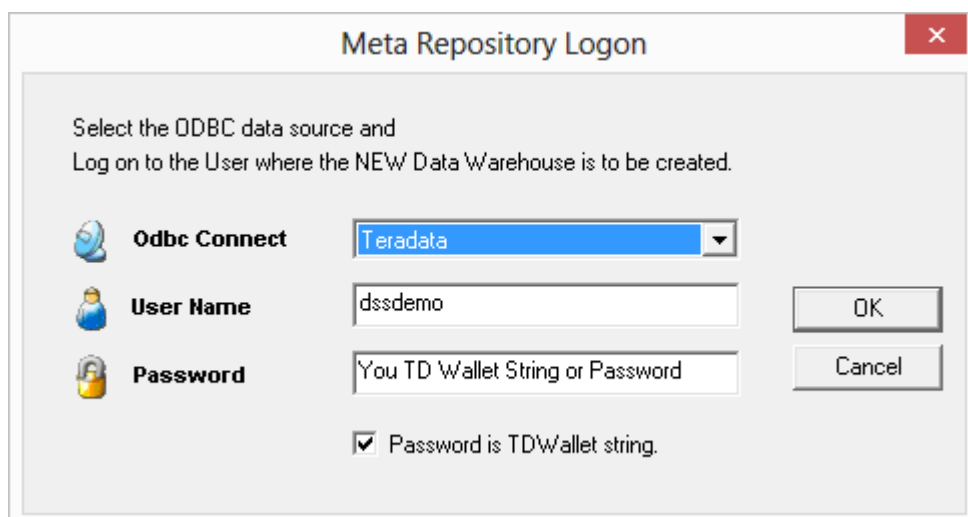
TIP: If multiple PCs will access the data warehouse they must all use the same data source name as entered here.

8.5 CREATING A METADATA REPOSITORY FOR TERADATA

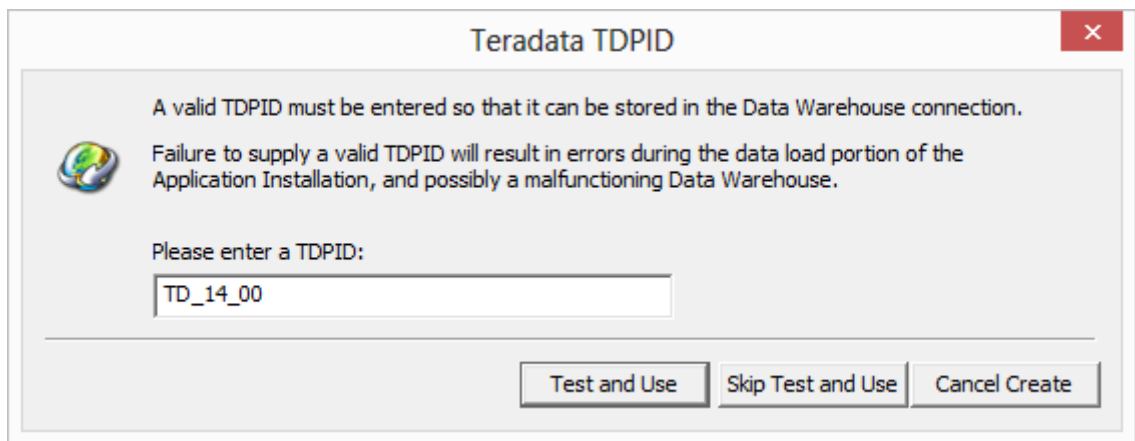
- 1 In WhereScape's Setup Administrator, click on the **Metadata Repository** tab in the left pane;
 - Then click on the **Create Metadata Repository** icon.
This option will create the WhereScape meta tables, sequences and procedures in the specified schema.
It checks to ensure that the schema, database or user does not contain an existing metadata repository before starting the metadata install.



- 2 After clicking on **Create Metadata Repository**, enter the **Odbc Connect**, **User Name** and **Password or TD Wallet String**.
 - Click **OK**.



- When creating a Teradata metadata repository, two additional screens are displayed:
 - First, a dialog is displayed for the Teradata server name (TDPID). Enter the **TDPID**.
 - Click **Test and Use**. If you skip the test and provide an invalid or wrong **Teradata TDPID**, native loads will not work in WhereScape RED.



Teradata TDPID

A valid TDPID must be entered so that it can be stored in the Data Warehouse connection.

Failure to supply a valid TDPID will result in errors during the data load portion of the Application Installation, and possibly a malfunctioning Data Warehouse.

Please enter a TDPID:

TD_14_00

Test and Use Skip Test and Use Cancel Create

NOTE: The name entered for the **Teradata Server Name** should appear in the **hosts** file on the computer in this form:

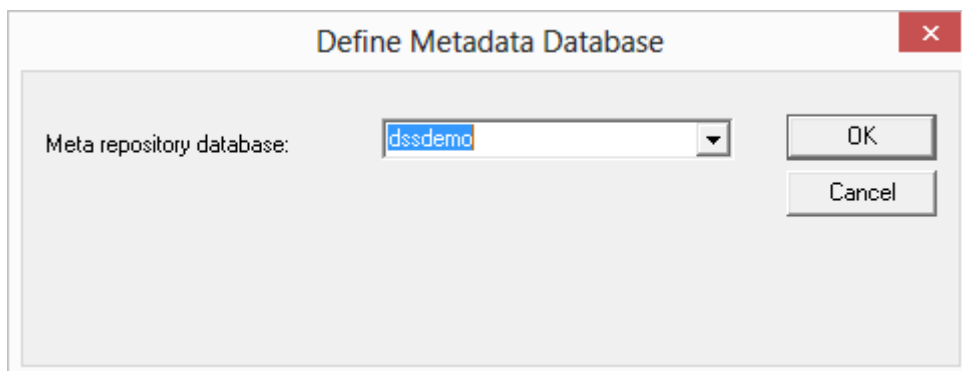
(IP address) (Teradata Server Name) (Teradata Server Name)Cop1

For the example in the ODBC screen shot above for TD_14_00, this entry would present if the server had an IP address of 10.12.123.60 and a name of "TD_14_00":

10.12.123.60 TD_14_00 TD_14_00Cop1

An alternative approach is to add the same entry to a dynamic name server (DNS).

- A second additional dialog is displayed for the Metadata Repository database.
 - Enter the name of the user or database in which the metadata repository is being created.
 - Click **OK**.



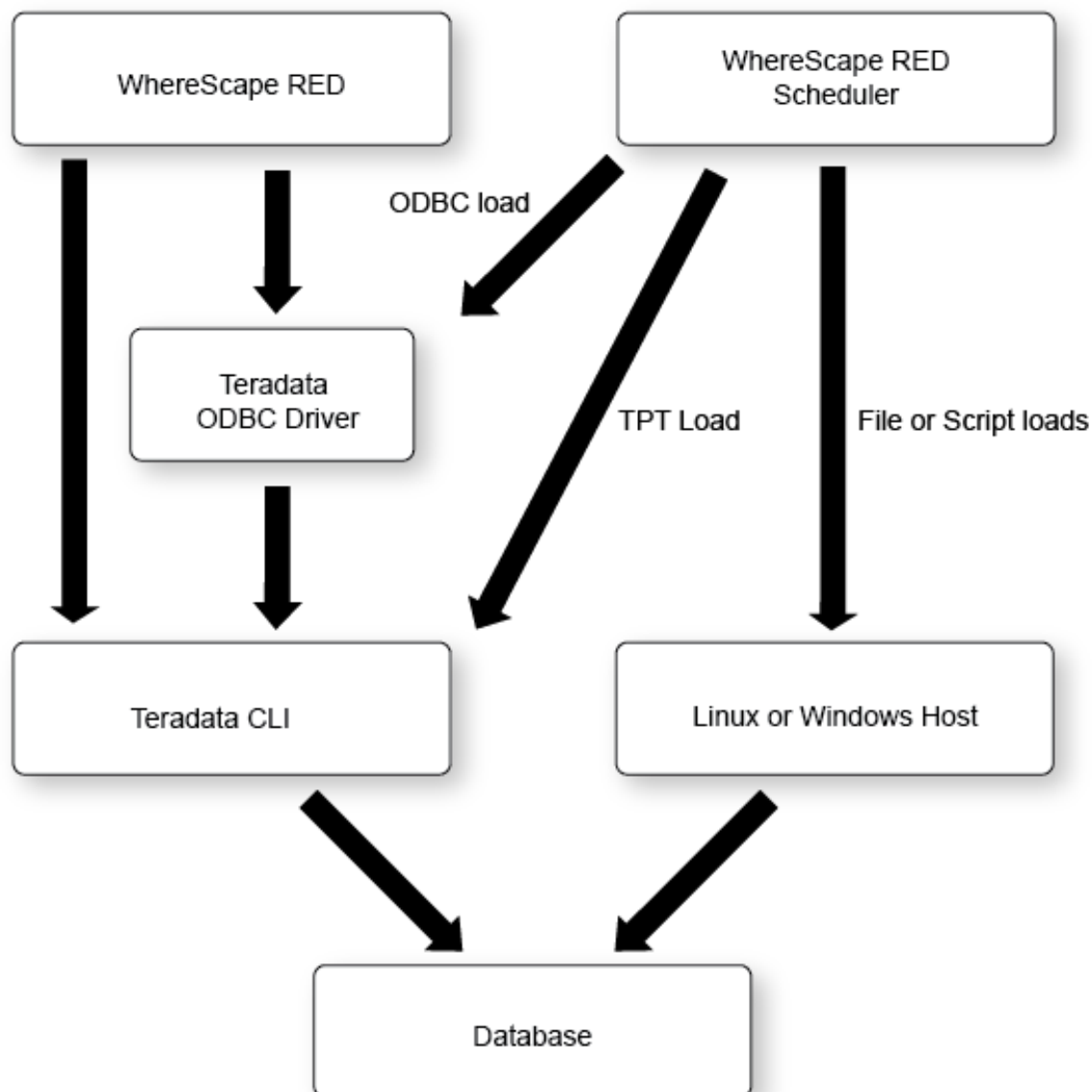
Define Metadata Database

Meta repository database: dssdemo

OK Cancel

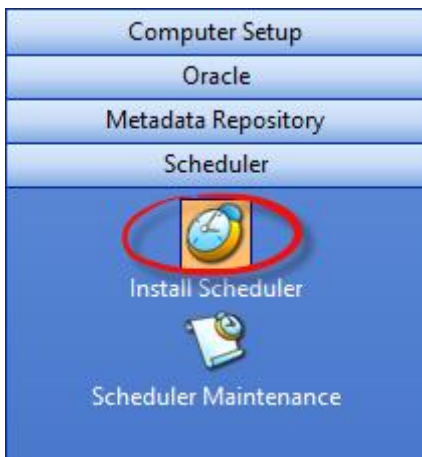
8.6 INSTALLING A WINDOWS SCHEDULER FOR TERADATA

This option installs and starts a copy of the WhereScape Windows scheduler. The scheduler runs as a system service and can also be maintained through the normal Windows services functions. The following diagram provides an overview of how WhereScape RED and the scheduler access a **Teradata** database.

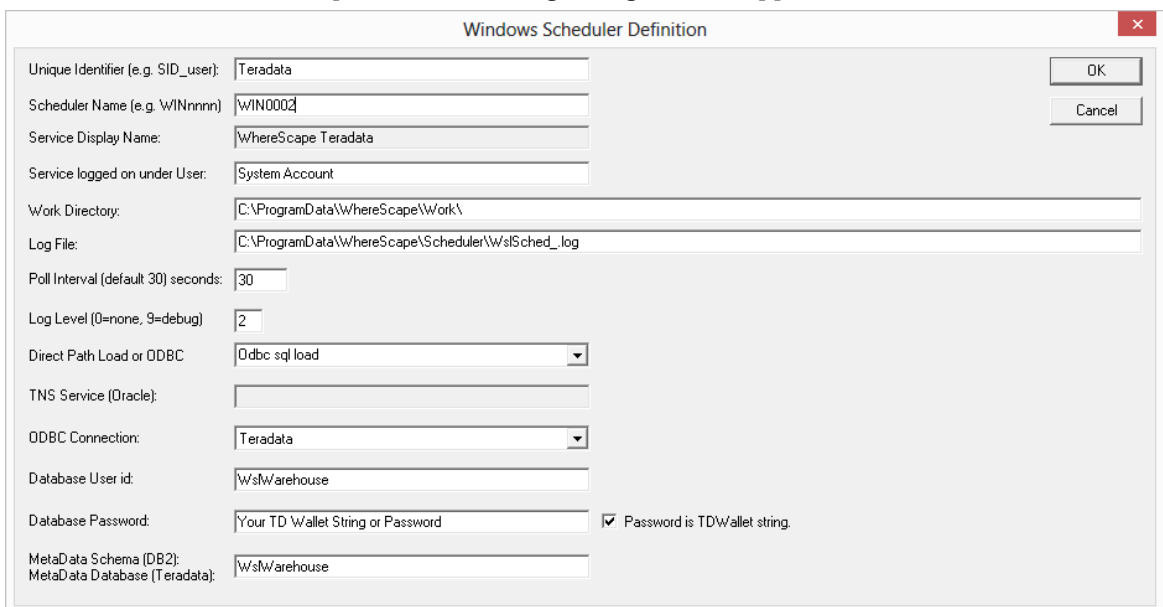


The scheduler makes use of the ODBC layer to perform the routine tasks involved in scheduling and running jobs. WhereScape RED and the scheduler can also use the TPT ODBC operator to load data into Teradata. For file or script based loads, the scheduler actions the load on the host system utilizing fastload, multiload and TPT (load and update operators).

- 1 In WhereScape's Setup Administrator, click on the **Scheduler** tab in the left pane.
 - Then click on the **Install Scheduler** icon.



- 2 On selection of this menu option the following dialog box will appear:

A screenshot of the 'Windows Scheduler Definition' dialog box. The fields are filled with the following values: Unique Identifier (e.g. SID_user): Teradata; Scheduler Name (e.g. WINnnnn): WIN0002; Service Display Name: WhereScape Teradata; Service logged on under User: System Account; Work Directory: C:\ProgramData\WhereScape\Work\; Log File: C:\ProgramData\WhereScape\Scheduler\WstSched_log; Poll Interval (default 30) seconds: 30; Log Level (0=none, 9=debug): 2; Direct Path Load or ODBC: Odbc sql load; TNS Service (Oracle): (empty); ODBC Connection: Teradata; Database User id: WsWarehouse; Database Password: Your TD Wallet String or Password; Metadata Schema (DB2): WsWarehouse; Metadata Database (Teradata): WsWarehouse. The 'Password is TDWallet string.' checkbox is checked. 'OK' and 'Cancel' buttons are visible in the top right.

Unique Identifier

Each scheduler requires a unique name (possibly the environment name and metadata owner e.g. prod_dssdemo). This unique name is the name by which the service is referenced.

Scheduler Name

When scheduling a job with WhereScape RED it is possible to assign the job to a specific scheduler. Each scheduler should therefore have a unique name. The default name is WIN0001. Most users change this to be the scheduler machine's name followed by a number, e.g.: DWHP01, where DWHP is the name of the windows scheduler server.

Service Display Name

This field displays the name that the scheduler will appear under when viewed through the Windows services. It is a combination of the string 'WhereScape' and the unique identifier.

Service Logged on Under User

This is the name of the Windows local user account that the scheduler service will use when logging on. The default setting is **System Account**, also known as the **Local System** service account. To set the scheduler to use a Windows domain user account with a password, use the Service Account option from the right-click menu. If a domain user account is specified then the contents of this field (on the **Windows Scheduler Definition** dialog) are ignored.

Note: A domain user account will require the **log on as a service** privilege in order for it to work. This is assigned automatically when setting the log on account using the Services console in Windows.

Work Directory

The directory used to store some temporary files created by the scheduler.



TIP: Two schedulers should not use the same work directory

Log File

The full path where the scheduler service will write a log of the various actions undertaken. This file may get large if log level 9 is set.

Poll Interval

This defines the frequency of the scheduler polling the database, looking for work. The default is 30 seconds. If set to a very low interval such as 1 then the impact on the database may be noticeable.

Log Level

This defines the level of detail written to the log file. The following logging levels are available:

- 0 - results in no information being written
- 1 - fatal errors only
- 2 - major events
- 3 - major events and return stats to caller
- 9 - full debug



TIP1: Setting Log Level 3 will return timing stats during ODBC based loads



TIP2: Setting Log Level 9 will create an additional detailed log file for each thread of each job that is run

Direct Path load or Odbc load

This should always be **ODBC sql load** for Teradata schedulers.

TNS Service

This is not set for Teradata schedulers.

ODBC Connection

The ODBC connection field requires the ODBC source name that will provide a connection to the data warehouse system. The scheduler uses ODBC to connect to the data warehouse.

Database User Id

The User name the Scheduler will use when connecting to the data warehouse Teradata database.

Database Password

The password or Teradata Wallet String for the User the Scheduler will use when connecting to the data warehouse Teradata database.

Metadata Schema (DB2) / Metadata Database (Teradata)

The Database Name or User Name where the metadata repository is stored. This must be entered for Teradata.

Note: A windows scheduler must be run as either the system service account or as a user with administrator privileges in windows.

8.7 INSTALLING A UNIX SCHEDULER FOR TERADATA

The Unix scheduler must be manually installed and maintained.

The scheduler consists of a series of shell scripts that are controlled by the cron process. The bulk of the work undertaken by the WhereScape scheduler takes place in the WhereScape procedures within the database. To obviate the need to provide a port to each Unix platform this series of shell scripts are used. They should run with no modification on all Unix platforms and have been tested under Unix offerings from Sun, HP, IBM and Linux.

The Unix scheduler can process every type of load except for ODBC loads, Native ODBC loads and XML loads. It cannot process OLAP Cubes.

WARNING: For UNIX/Linux scheduler processing, the **Encrypt User** and **Password** options in RED's Repository Privacy Settings cannot be used. Encrypt options are only supported when using a Windows scheduler.

Installation

The installation of the Unix scheduler needs to be undertaken by someone familiar with the Unix environment and the running of database processes from Unix. The tasks that need to be undertaken are:

1. Unix User and Shell

Acquire a Unix user name that has access to cron and the database data warehouse environment. A reasonable quantity of disk space is also required if the automated backups of the metadata repository are used. The user should log on to Unix without passing through a menu system if possible. Log on to the user.

- This user will need to have a .profile file in "/home/<user>/profile" otherwise ws_sched_check_600.sh will not work. If this file doesn't exist it will need to be created.

The WhereScape scheduler consists of a number of shell scripts. By default some of these shell scripts will run in the Bourne shell and some will run in the Korn shell. If Linux is being used then the Bash shell is used exclusively.

Note: During setup it is recommended that the Korn shell be used as the default shell for the user installing the scheduler (or Bash if running Linux).

2. Create directories

Create the following directories (if not already present) to be used by the WhereScape scheduler. If the directory names are not appropriate and others must be used then it will be necessary to alter the WhereScape supplied scripts.

- wsl (top level directory for WhereScape directories)
- wsl/bin (used to store the WhereScape scripts)
- wsl/sched (used for the scheduler job and log files)
- wsl/sched/job (used for the scheduler job files)
- wsl/sched/joblog (used for the scheduler job logs)
- wsl/sched/log (used for the scheduler logs)
- wsl/export (used for the automated exports of the metadata)
- wsl/expback (used to hold compressed copies of the metadata exports)
- wsl/log (used for export and archive logs)
- wsl/mon (used for the monitoring process)
- wsl/mon/db (used for database monitoring scripts and control files)
- wsl/mon/job (used for job monitoring scripts)
- wsl/mon/log (used for monitoring log files)

3. Copy supplied Unix files

FTP the scheduler files shipped with WhereScape RED to the wsl/bin directory on the Unix system (use ASCII mode). All files should be copied over.

WhereScape RED ships with several different versions of the unix files for the scheduler. The appropriate version of the files should be copied to the server. The following lists the files shipped:

- ...\\WhereScape\\Teradata\\Linux\\Version_nnn\\...

Note: There may be several different versions of the scheduler files for a given database and platform (Unix or Linux). These are identified by version numbers nnn.

For example, there may be two different folders in ...\\WhereScape\\Teradata\\Linux\\: Version_560 and Version_600. The highest version number script less than or equal to the version of RED in use should always be used.

In the table that follows, the nnn in file names refers to the version number of the script. Some scripts might have 010, some will have 411 and others 600, etc.

A definition of the files follows:

File	Purpose
tdenv (for Teradata)	Contains the template of a script for setting environmental variables. This needs to reside in the home directory of the Unix user and must be tailored for the database user and environment. It may be renamed to reflect the actual environment. (e.g. tdenv_TDPID_user)
crontab	Contains the template of the required cron commands. This file needs to be modified to include the HOME directory of the Unix user, and the name of the environment file. It is then used to update the crontab tables.
ws_start_nnn.sh	Script to start the scheduler. This script is normally called by the ws_sched_check_nnn.sh script and is passed the database environment file name as a parameter.
ws_sched_check_nnn.sh	Script to check that the scheduler is running. This script is normally executed by cron every 20 minutes. If the scheduler is found to be absent it attempts to start a new iteration. By default the scheduler will terminate if its database is shut down. This script will therefore start the scheduler after a system start-up or a database start-up.
ws_sched_nnn.sh	The actual scheduler script. This script looks for new scheduler jobs every x seconds. The default is every 30 seconds. This script must be modified to change the default behavior. It calls the ws_job_wait_010.sh script to perform the actual checking and initiating of a job. It terminates when ws_job_wait_010.sh returns an error, and relies upon cron for a restart.
ws_sched_status_nnn.sh	This script is called by ws_sched_nnn.sh to update the scheduler status in the scheduler metadata. It's a wrapper of the metadata database procedure ws_sched_status (i.e. it calls the database procedure ws_sched_status and returns it's results to the calling script).
ws_job_wait_nnn.sh	This script performs the bulk of the work. It is called by ws_sched_nnn.sh on a regular basis. It checks for any job waiting to run. If it locates a job that is ready to run it will create and start a script in the \$HOME/wsl/sched/job directory for each thread of that job. These job threads in turn perform the actual processing and log their results to \$HOME/wsl/sched/joblog. Under normal operation it is not necessary to check or inspect these files.

ws_job_check_nnn.sh	This script is called to check the status of running jobs. It is called by ws_sched_nnn.sh on a regular basis (the interval is defined in ws_sched_nnn.sh). It checks that any job running in the scheduler is actually running in the Unix environment. If a job is not found under Unix then it is aborted in the scheduler. This scenario can occur as the result of a database or machine crash. If a job abort is required the results are logged to \$HOME/wsl/sched/log/job_check_'env file name'.log. Under normal operation it is not necessary to check or inspect this file.
cleanup_nnn.sh	This script is normally executed by cron and cleans up files created by the WhereScape scheduler. It deletes old versions of the jobs, job logs, and backups. It will also delete any temporary scripts created in the running of host scripts or file loads from within the scheduler. The location of these temporary scripts is defined within the WhereScape RED Unix Connection object. The default value in the cleanup script is /tmp/ and this must be altered if a different temporary directory is used for a Connection.

4. Set permissions

Change the permissions on the copied scripts so that they are executable. (e.g. chmod 750 \$HOME/wsl/bin/*)

5. Create an environment file

Using the environment file (tdenv) as a template and adhering to the variable names within, create an environment file for the database environment and user under which the scheduler is to run. The environment file must be copied to the \$HOME directory of the user selected in Step (1). It may be called anything but a meaningful name may be tdenv_TDPID_user for a given Teradata scheduler. Check the users .profile and database environment file to see the normal settings.

Make the following replacements in the environment file:

- Replace all occurrences of `_HOME_` with the full pathname of your home directory
- Replace all occurrences of `_TDPID_` with the host name of the data warehouse database server, used by the scheduler as the database server name for Teradata load and unload utilities
- Replace all occurrences of `_BTEQDB_` with the host name or ip address of the data warehouse database server, used by the scheduler as the database server name/address for reading and writing WhereScape RED Metadata in Teradata (may be set to the same value as `_TDPID_`)
- Replace `_USER_` with the data warehouse database user name
- Replace `_PASSWORD_` with the data warehouse database user password
- If using a Teradata Wallet String, comment out the first Password line and comment back in the second line containing `_WALLETSTRING_`. Replace all occurrences of `_WALLETSTRING_` with the data warehouse user wallet string.
- Replace all occurrences of `_METABASE_` with the user or database where the metadata resides
- Replace all occurrences of `_WORKDIR_` with the name of the directory that will be used as the work directory by default (e.g. /tmp/)

6. Setup crontab

Ascertain if there are any existing crontab entries, and if so save them to a file for re-inclusion later. (e.g. crontab -l)

Edit the crontab file and perform the following changes. Also check all entries and make sure you are happy with the times and actions:

- Change all occurrences of `_HOME_` with the full pathname of the Unix users home directory
- Change all occurrences of the environment file name (tdenv) to the name of the file created in step (4)
- Remove comment markers from the beginning of the second line of code (the line containing this: `ws_sched_check_nnn.sh`)

7. Initialize crontab entries

Load the crontab file into cron (e.g. crontab crontab).

Perform a crontab list to ensure it all looks OK (e.g. crontab -l)

8. Start Scheduler

Manually start the scheduler by issuing the `ws_sched_check_nnn.sh` script with the environment file name from step (4) as the parameter (e.g. `ws_sched_check_600.sh tdenv`). If the scheduler is not running it will be started. Execute the script a few more times to ensure that only one copy of the scheduler is started. Check that the scheduler is running by executing a `'ps -ef'` statement and checking that the `ws_sched_nnn.sh` process is present.

9. Test the Scheduler

Run a job from WhereScape RED and ensure that it works. In the event that the job does not start or goes into and remains in a pending state, the following investigations should be undertaken:

- Ensure all WhereScape procedures are compiled. Use the **Validate Metadata Repository** option in the Setup Administrator to confirm that all procedures are valid.
- Review the job logs in `$HOME/wsl/sched/joblog`.
- Review the logs in `$HOME/wsl/sched/log`.
- Isolate the name of the job and locate the relevant files for this job in `$HOME/wsl/sched/job` and `$HOME/wsl/sched/joblog` if they exist. Mail these files together with the files in `$HOME/wsl/sched/log` to support@wherescape.com. It may be necessary to remove password references from the file(s) in `$HOME/wsl/sched/job`.

NOTE: It is possible to stop a Linux/UNIX Scheduler from within WhereScape RED. See the Teradata WhereScape RED User Guide for details.

8.8 INSTALLING A TERADATA UNIX LOADER

Sometimes the Teradata database server has a Linux non-tpa node or a MP-RAS UNIX non-tpa node used for loading data from files. This can provide a significantly better platform for loading files into Teradata than a network connected Windows scheduler server. The WhereScape RED Teradata UNIX loader provides a mechanism for moving file loads from a windows scheduler to a non-TPA node in certain circumstances.

The Unix loader engine must be manually installed and maintained.

The loader engine consists of a shell script that is controlled by the cron process. The loader should run without modification on all Unix platforms and has been tested under Unix offerings from Sun, HP, IBM and Linux.

Installation

The installation of the Unix loader needs to be undertaken by someone familiar with the Unix environment and the running of Teradata Bteq and fastload processes from Unix. The tasks that need to be undertaken are:

1. Unix User and Shell

Acquire a Unix user name that has access to cron and the Teradata data warehouse environment. The user should log on to Unix without passing through a menu system. Log on to the user.

The WhereScape Teradata Unix Loader consists of a number of shell scripts. By default the shell script will run in the Korn shell on unix. If Linux is being used then the Bash shell is used.

Note: During setup it is recommended that the Korn shell be used as the default shell for the user installing the loader (or Bash if running Linux).

2. Teradata User

Acquire a Teradata user name that has read, write and execute procedure privileges for the Teradata database/user containing the WhereScape RED metadata as well as read and write access to the databases tables to be loaded exist in.

3. Create directories

Create the following directories (if not already present) to be used by the WhereScape loader.

- wsl (top level directory for WhereScape directories)
- wsl/bin (used to store the WhereScape script)
- wsl/data (used for the data files being loaded)
- wsl/log (used for log files)
- wsl/work (used for temporary files)

4. Copy supplied Unix files

FTP the files in the WhereScape installation\Unix directory (or \Linux for Linux installations) to the wsl/bin directory on the Unix system (use ASCII mode). All files should be copied over. A definition of the files follows:

File	Purpose
tdenv	Contains the template of a script for setting environmental variables. This needs to reside in the home directory of the Unix user and must be tailored for the Teradata user and environment. It may be renamed to reflect the actual environment. (e.g. tdenv_env_user)
ws_sched_tera_550.sh	Script to check to see if any files require loading and create and spawn new scripts for each file to be loaded. This script is normally executed by cron every 5 minutes.

5. Set permissions

Change the permissions on the copied files so that they are executable. (e.g. `chmod 750 $HOME/wsl/bin/ws_sched_tera_550.sh`)

6. Create an environment file

Using the tdenv file as a template and adhering to the variable names within, create an environment file for the Teradata environment and user under which the unix loader is to run. This file should be located in the \$HOME directory. It may be called anything, but a meaningful name may be: (tdenv_env_user). Check the users .profile to see the normal settings.

As a minimum you will need to make the following changes to the template oraenv file:

- Replace all occurrences of `_TPID_` with the Teradata server name
- Replace all occurrences of `_USER_` with the Teradata user name the unix load will run loads as
- Replace all occurrences of `_PASSWORD_` with the password for `_USER_`
- If using a Teradata Wallet String, comment out the first Password line and comment back in the second line containing `_WALLETSTRING_`. Replace all occurrences of `_WALLETSTRING_` with the data warehouse user wallet string
- Replace all occurrences of `_METABASE_` with the data warehouse database containing the WhereScape RED metadata
- Replace all occurrences of `_WORKDIR_` with the full pathname of the work directory
- Replace all occurrences of `_DATA_DIR_` with the full pathname of the data directory
- Replace all occurrences of `_LOG_DIR_` with the full pathname of the log directory.

7. Setup crontab

Add the following to the users' crontab:

```
#  
## Run the WhereScape file loader  
##  
0,5,10,15,20,25,30,35,40,45,50,55 * * * * *  
_HOME_/wsl/bin/ws_sched_tera_550.sh tdenv >>_HOME_/wsl/log/sched_tera.log  
2>&1
```

making the following changes:

- Change all occurrences of `_HOME_` with the full pathname of the Unix users home directory.
- Change all occurrences of `tdenv` to the name of the file created in step (4).

8. Test the Scheduler

Process a load table in WhereScape RED that has been set up to use the Unix File Load without any post clean up actions (for an example, see *Native Loads using UNIX and Linux* in the **WhereScape RED Teradata User Guide**) and ensure that it works. Check to see if the file has been moved to the data directory in Unix and that a temporary files exist in the work directory.

CHAPTER 9

9. ORACLE INSTALLATION AND CONFIGURATION

IN THIS CHAPTER

9.1 Installation and Configuration Introduction	74
9.2 Installing RED for Oracle	76
9.3 Creating an Oracle User	78
9.4 Configuring an Oracle Metadata Connection (DSN)	84
9.5 Validating Oracle User Privileges.....	92
9.6 Creating a Metadata Repository for Oracle	96
9.7 Installing a Windows Scheduler for Oracle	99
9.8 Installing a Unix Scheduler for Oracle	106

9.1 INSTALLATION AND CONFIGURATION INTRODUCTION

The steps provided here are an indication of the normal process required to install and configure WhereScape RED. In some environments additional processes will be required.

- 1 Install the Oracle 10g, 11g or higher database client software on a suitable PC.
- 2 Install the WhereScape RED software onto your PC. See *Installing RED for Oracle* (see "9.2 *Installing RED for Oracle*" on page 76)
- 3 Add your license key. See *Adding your License* (see "3. *Adding Your License*" on page 15)
- 4 Check your Windows version. See *Checking your Windows Version* (see "4. *Checking Your Windows Version*" on page 18)
- 5 Check your ODBC Drivers. See *Checking your ODBC Drivers* (see "5. *Checking Your ODBC Drivers*" on page 19)
- 6 Designate a database that is to be used as the data warehouse database. For testing or evaluation purposes this may be a PC based database. WhereScape RED requires the Oracle database version 10.1 or greater as the data warehouse database. Record the SID for this database. Create the database user (schema) that is to be used to hold the metadata and data warehouse tables. Allocate the privileges required by WhereScape RED. See *Creating an Oracle DSS User* (see "9.3 *Creating an Oracle User*" on page 78)
- 7 Establish the Oracle Net connections required to allow access to the data warehouse database from the PC. Record the **Server** name for the connection.
- 8 Add the definition of an ODBC data source to connect to the data warehouse database. See *Configuring an Oracle Metadata Connection (DSN)* (see "9.4 *Configuring an Oracle Metadata Connection (DSN)*" on page 84)
- 9 Create the WhereScape metadata repository under the data warehouse schema that you created. See *Creating a Metadata Repository for Oracle* (see "9.6 *Creating a Metadata Repository for Oracle*" on page 96)
- 10 Install a Windows Scheduler. See *Installing a Windows Scheduler for Oracle* (see "9.7 *Installing a Windows Scheduler for Oracle*" on page 99)
- 11 Install a Unix Scheduler. See *Installing a Unix Scheduler for Oracle* (see "9.8 *Installing a Unix Scheduler for Oracle*" on page 106)
- 12 If required, load the WhereScape tutorial tables into a separate database schema. It is highly recommended that the tutorials be undertaken. See *Creating Tutorial Data* (see "11.1 *Creating Tutorial Data*" on page 137)

Problem Solving

If problems occur in your install that cannot be resolved proceed as follows when running WhereScape's Setup Administrator utility:

- 1 Create a log file by selecting **Log/Start Log** from the menu bar. Enter a file name.
- 2 Step through at least:
 - Computer Setup --> Check Windows version
 - Computer Setup --> Check ODBC driver
 - Computer Setup --> Validate/Add Odbc source
 - Oracle --> Validate Oracle user privileges (if an Oracle DataWarehouse)
 - Create/Validate Meta Repository --> Validate meta repository
- 3 Select **Log/Stop Log** to close off the log
- 4 Edit the log and remove any passwords that appear.
- 5 Send your log file to support@wherescape.com

9.2 INSTALLING RED FOR ORACLE

Installation Prerequisites

WhereScape RED requires certain client and server components.

Client Prerequisites

- Windows Vista, Windows 7 or Windows 8.
- 250MB available disk space
- ODBC Version 3.0 (normally shipped as part of the Operating System).
- For building a data warehouse, a version 10.1 database client or greater is required. A 32-bit Oracle client installation will be necessary.
- WhereScape RED supports the Microsoft ODBC for Oracle and DataDirect's ODBC Driver for Oracle.

Server Prerequisites

- One of the following data warehouse database platforms is required: Oracle version 10g, 11g or higher.
- To use the WhereScape RED Scheduler, a UNIX, LINUX or Windows based server is also required (this may be the database server). WhereScape RED only supports sh and ksh under Unix and bash under Linux.

Installation of WhereScape RED

To install WhereScape RED, see *Installing WhereScape RED* (see "2. Installing WhereScape RED" on page 9)

NOTE: The WhereScape RED installer selects the following installation directory as the default location for RED on a 64-bit windows server: C:\Program Files (x86)\WhereScape. This is where 32-bit applications normally reside on a 64-bit Windows platform.

However, if Oracle 10g is involved, either as the data warehouse repository, or as a source system, installing RED in "Program Files (x86)" can cause an issue. Oracle does not like directories that have a parenthesis in their name. Applications using Oracle cannot be located in such a directory. This includes WhereScape RED using ODBC to connect to Oracle. The result is that WhereScape RED will not be able to connect to Oracle. (This issue does not usually occur with Oracle 11g). To prevent this, change the installation directory of WhereScape RED. Options include:

C:\WhereScape

D:\Program Files\WhereScape (if a second drive is available)

It is not good practice to install 32-bit applications in C:\Program Files\WhereScape on a 64-bit machine.

9.3 CREATING AN ORACLE USER

The creation of the Oracle database user that will be used to administer and maintain the data warehouse must be performed outside this utility. The **SQL Admin** utility can be used to execute the sql required to create such a user, or the database administrator should be contacted.

A list of the required privileges is displayed when this option is selected. Many of these privileges must be allocated as specific privileges and cannot be granted as part of a role. For example if a 'grant dba to dssuser' command is executed WhereScape RED will still need these privileges specifically granted as 'dba' is a roll.

Note: WhereScape RED makes use of the `dbms_lock` and `dbms_sql` packages supplied by Oracle. When using Oracle, privileges obtained by the owner via a role cannot be referenced in a procedure. Therefore RED requires that these privileges be granted to the user. This is an Oracle restriction and not something imposed by WhereScape.

For our data warehouse, or decision support system, we will create a user, 'dssdemo'. An example sql script to create this 'dssdemo' user is shipped and can be found in the WhereScape RED program directory:

- `..\Oracle\Recreate_dssdemo.sql`

Please note that **this script will need to be modified** to reflect the data warehouse schema chosen at your site. An example script follows below.

Note that this script would normally need to be run under the sys user.

1. The dss user created is dssdemo in this case:

```
drop user dssdemo cascade;
create user dssdemo identified by wsl;
alter user dssdemo default tablespace DEMODATA;
alter user dssdemo temporary tablespace TEMP;

grant create session to dssdemo;
grant resource to dssdemo;
grant create table to dssdemo;
grant create view to dssdemo;
grant create sequence to dssdemo;
grant create procedure to dssdemo;
grant select any table to dssdemo;
grant create materialized view to dssdemo;
grant create database link to dssdemo;
grant query rewrite to dssdemo;
grant select on sys.v_$database to dssdemo;
grant alter session to dssdemo;
grant select on sys.v_$session to dssdemo;
grant execute on sys.dbms_lock to dssdemo;
grant execute on sys.dbms_sql to dssdemo;
grant create any view to dssdemo;
grant unlimited tablespace to dssdemo;
grant alter system to dssdemo;
```

NOTE:

select any table - this is only required to gain access to tables in other schema. It is not required if specific grants have been provided to the tables that are required, such as with the tutorial.

alter system - some dbas do not like granting the ALTER SYSTEM privilege as it is very powerful. With this privilege, you can make wide reaching changes including altering initialization parameters. RED only uses ALTER SYSTEM to kill jobs. Rather than granting ALTER SYSTEM, you can compile Ws_User_Abort (contained in C:\Program Files\WhereScape\Oracle) under the SYS user. Then grant execute on SYS.Ws_User_Abort to dssdemo.

2. To use object placement **across multiple schemas** in WhereScape RED, the RED user also needs be granted the following privileges:

```
grant select any table to dssdemo;  
grant create any view to dssdemo;  
grant drop any view to dssdemo;  
grant create any table to dssdemo;  
grant drop any table to dssdemo;  
grant delete any table to dssdemo;  
grant insert any table to dssdemo;  
grant update any table to dssdemo;  
grant alter any table to dssdemo;  
grant global query rewrite to dssdemo;  
grant create any materialized view to dssdemo;  
grant drop any materialized view to dssdemo;  
grant alter any materialized view to dssdemo;  
grant create any index to dssdemo;  
grant drop any index to dssdemo;  
grant alter any index to dssdemo;  
grant select any sequence to dssdemo;  
grant create any sequence to dssdemo;  
grant drop any sequence to dssdemo;  
grant analyze any to dssdemo;
```

3. To do loads from **Hadoop** using multiple schemas, the RED user also needs be granted the following privilege:

```
grant select_catalog_role to dssdemo;
```

9.3.1 CREATING ORACLE INDIVIDUAL USERS

For Oracle users wanting each developer to log into Oracle using their own username and password, they can use the **Oracle Individual User** Logon Method.

The Oracle Individual User option allows a user, with the appropriate permissions, to log into WhereScape RED and then operate as the RED schema.

This method enables customers to configure RED ensuring that users logged in via this method cannot edit the metadata user password in the connections, and also ensuring that the password is encrypted in **Tools>Options**.

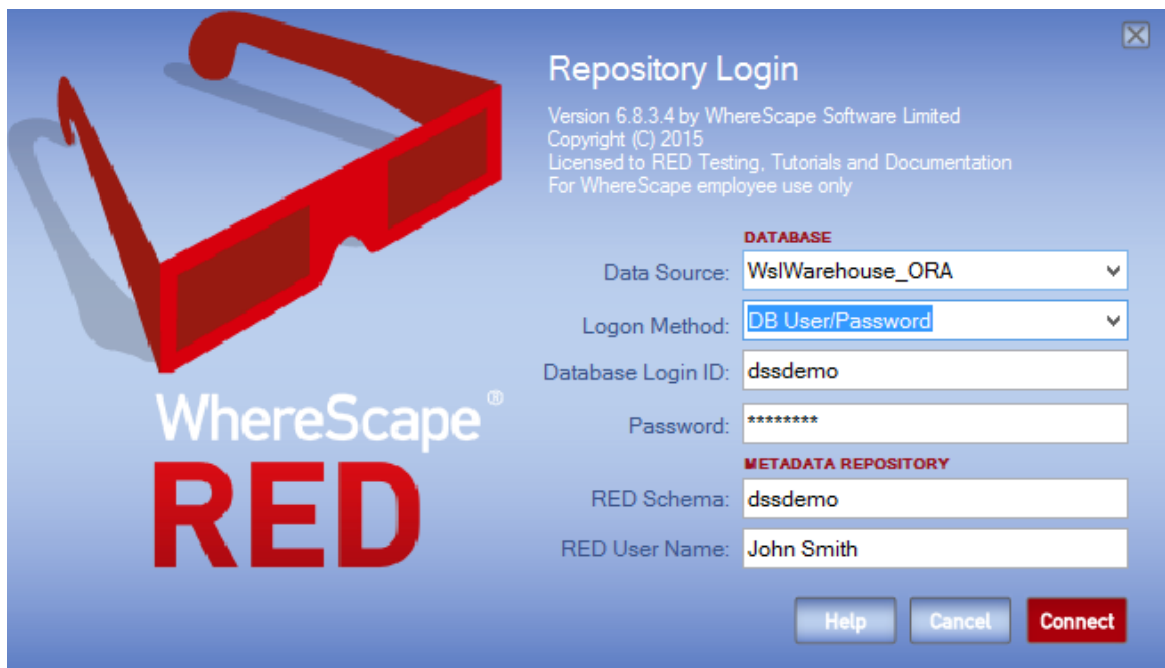
To authorize users to log in to RED using the Oracle Individual User method, a DBA will need to grant users select on the **ws_dbc_connect** table, as well as the **CREATE SESSION** system privilege.

This method is also useful for tracking individual WhereScape RED user activity in the Oracle database. Their authenticating user name is now stored in the CLIENT_IDENTIFIER field of v\$session which can be used by Oracle's auditing, functioning to track individual developer activity.

Oracle individual users will need to be created and granted certain privileges. Alternatively, for existing Oracle database users, they just need to be granted the following privileges below:

```
grant create session to oracle_user1
grant select on dssdemo.ws_dbc_connect to oracle_user1
```

- 1 To ensure the **Repository Privacy Settings** is locked in **Tools>Options**, the RED metadata user needs to log in to RED before any individual user. To log in:
 - Select the **DB User/Password** option from the Logon Method drop-down menu.
 - Enter the user name and password for the metadata user.
 - Enter the RED schema.

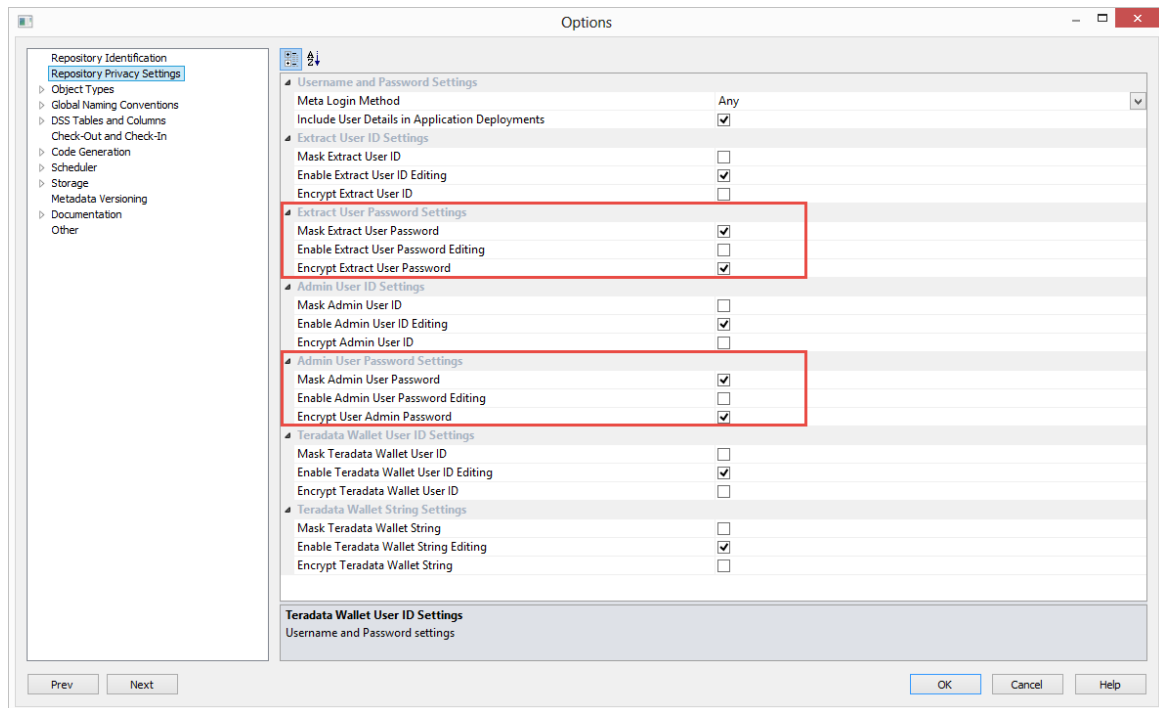


- 2 Once logged in to RED, the metadata user needs to do the following in **Tools>Options>Repository Privacy settings**:
 - Check the option to enable the **Mask Extract User Password**.
 - Un-check the option to disable the **Extract User Password Editing** (this will lock down the metadata user password in the connections).
 - Check the option to enable the **Mask Admin User Password**.
 - Un-check the option to disable the **Enable Admin User Password Editing** (this will lock down the Admin/Dss User Password in the connection properties).

NOTE: Encrypt User/Admin User Password Options

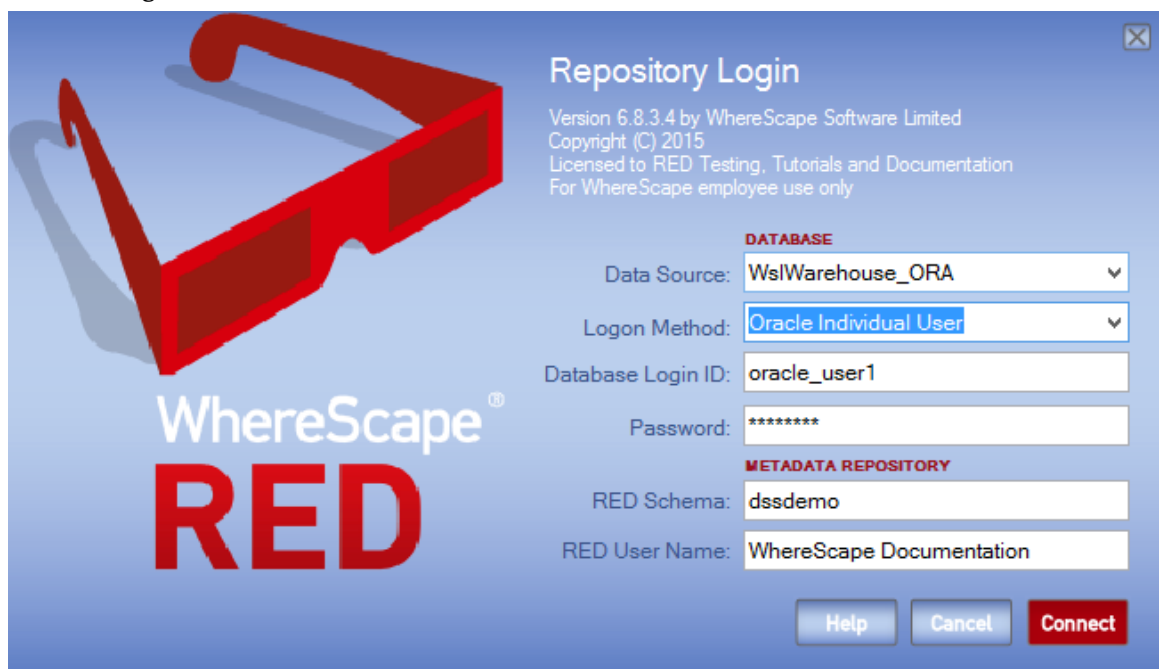
Users should also check the options to enable **Encrypt User Password** and **Encrypt Admin User password** as shown in the example screenshot below.

For **UNIX/Linux scheduler** processing User and Password Encryption implications, please see section 5.1.1.2 Settings - Repository Privacy Settings in the RED User Guide.

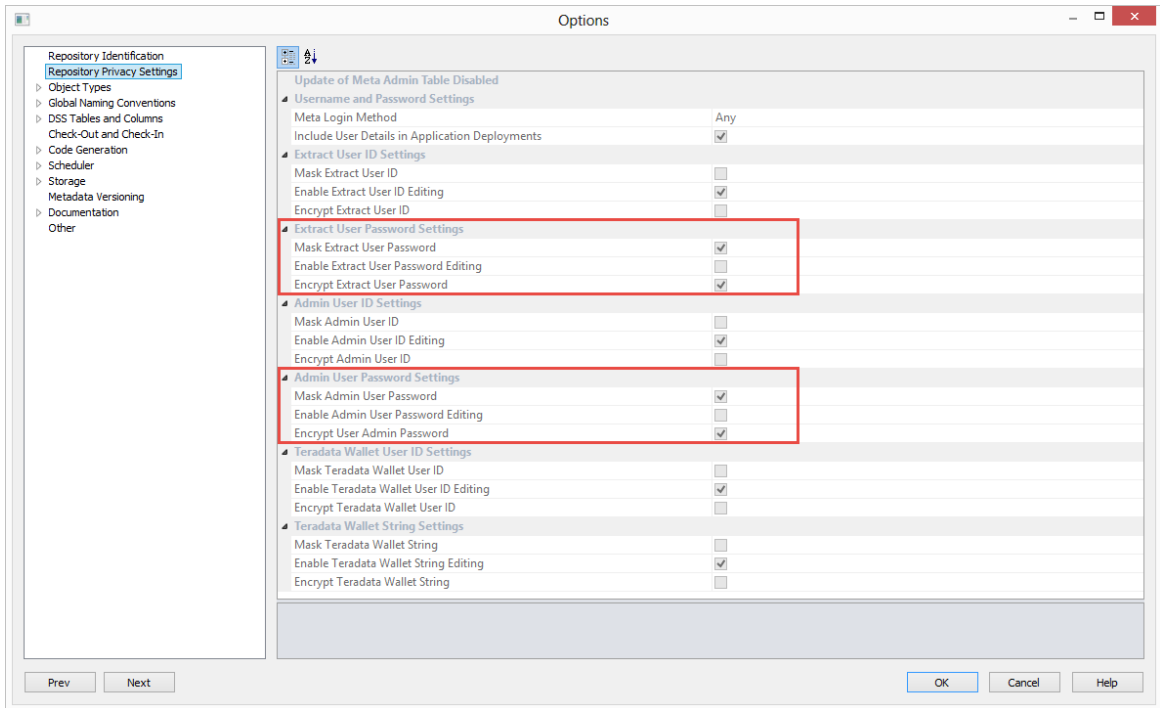


3 Multiple users can then login to RED using their own credentials by:

- Selecting the **Oracle Individual User** option from the **Logon Method** drop-down menu.
- Entering their user name and password.
- Entering the RED schema.



- 4 Users that log in to RED via the **Oracle Individual User** method will have their **Repository Privacy Settings** completely disabled.





WhereScape RED TIP: Oracle Individual User System Information

The permissions below are required to list and see user session, session locks and user locks information for Oracle individual users.

```
grant select on sys.v_$sql to dssdemo;  
grant select on sys.v_$sess_io to dssdemo;  
grant select on sys.v_$locked_object to dssdemo;  
grant select on sys.v_$lock to dssdemo;
```

The screenshot shows the 'Output' window in the WhereScape RED SQL Editor. The table displays the following data:

USERNAME	STATUS	OSUSER	CLIENT_IDENTIFIER	PROGRAM	SQL_TEXT (WARNING: Output truncated to 256 characters)	SID
DSSDEMO	ACTIVE	User		Connecting ... - []	select distinct username, status ,osuser, client_identifier, program, sql_te...	72
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		24
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		26
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		36
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		45
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		48
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		49
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		50
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		53
DSSDEMO	INACTIVE	User	dssdemo	WhereScape RED		55
DSSDEMO	INACTIVE	User	oracle_user1	WhereScape RED		47
DSSDEMO	INACTIVE	User	oracle_user1	WhereScape RED		67
DSSDEMO	INACTIVE	User	oracle_user1	WhereScape RED		73
DSSDEMO	INACTIVE	User		Connecting ... - []		17
DSSDEMO	INACTIVE	User		Connecting ... - []		41

9.4 CONFIGURING AN ORACLE METADATA CONNECTION (DSN)

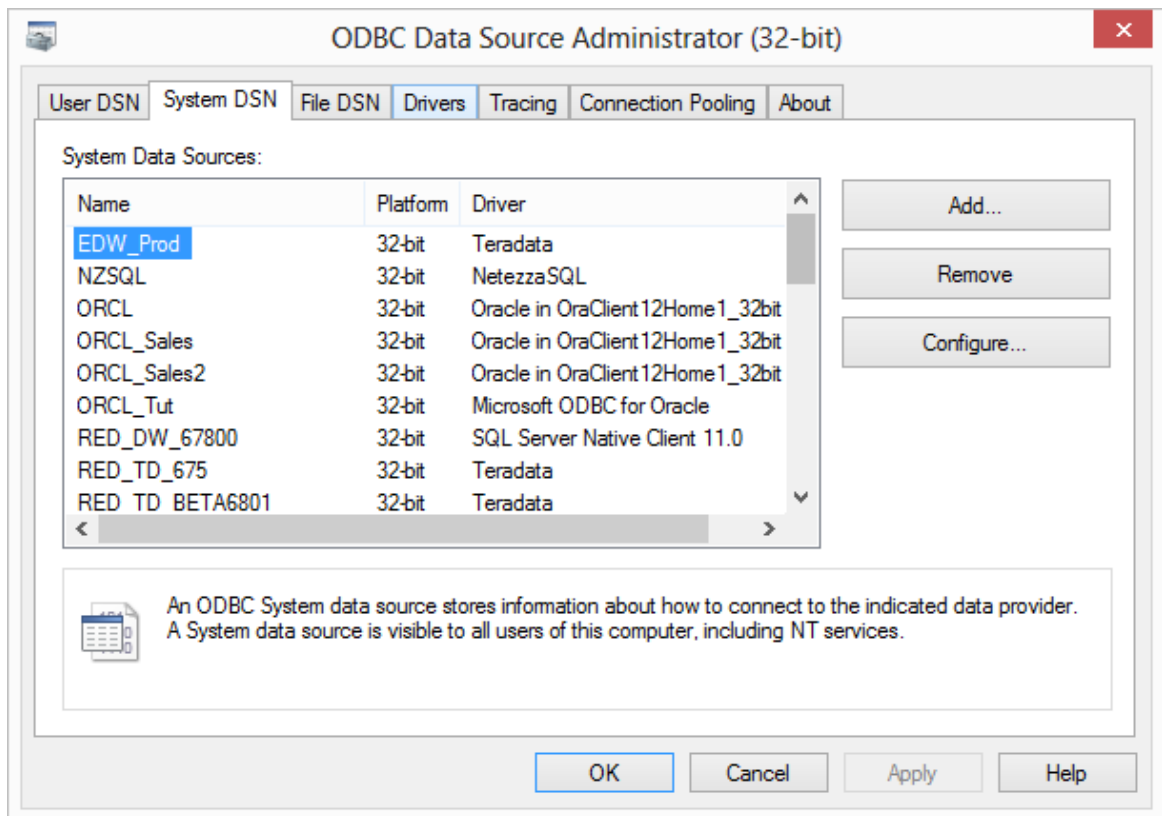
WhereScape RED supports both the **Microsoft** ODBC Driver for Oracle and the **DataDirect** ODBC Driver for Oracle.

See below instructions for setting up either one of these drivers.

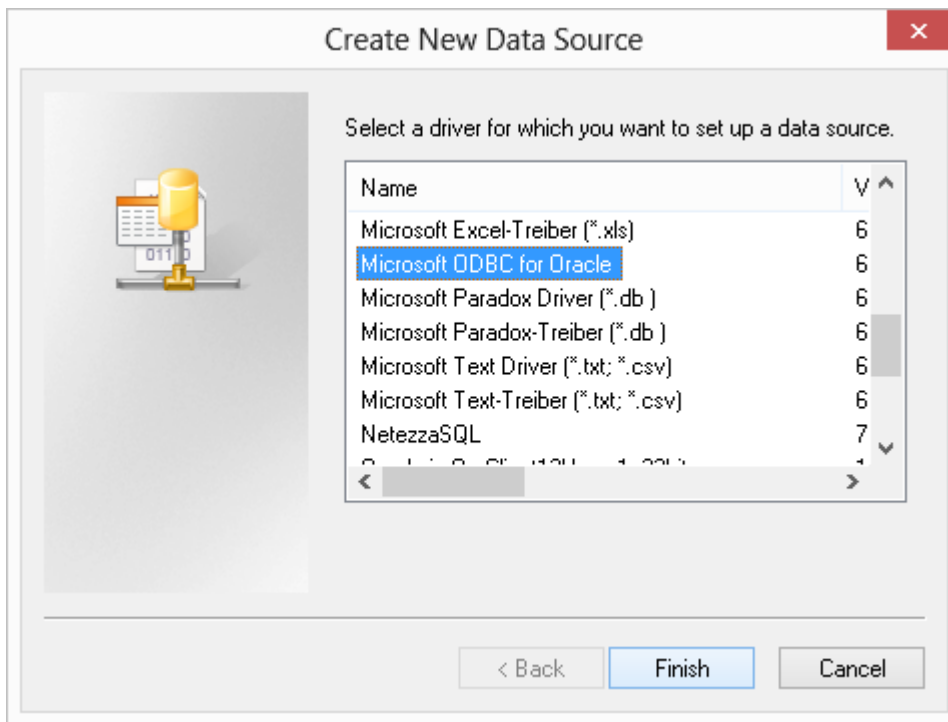
To begin adding an ODBC Source from WhereScape's Setup Administrator, see **Creating an ODBC Connection** (see "6. Creating an ODBC Connection" on page 21).

Setting up a Microsoft ODBC Driver for Oracle:

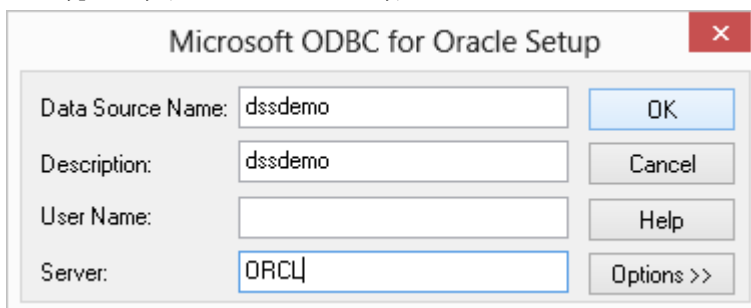
- 1 In ODBC Data Source Administrator, click on the **System DSN** tab to view the system wide ODBC entries.
 - Click on the 'Add...' button to add a new entry.



- 2 Select the **Microsoft ODBC for Oracle** driver from the driver list.
 - Click **Finish**.



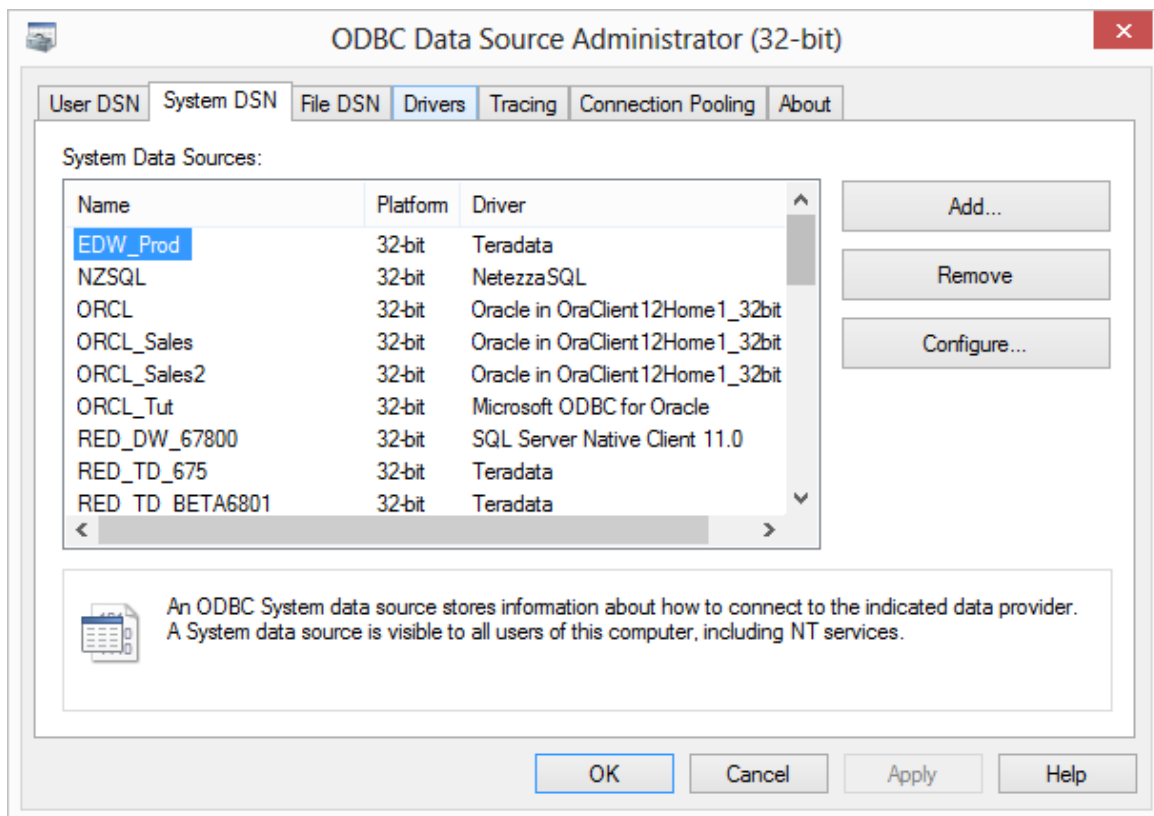
- 3 At this stage the connection to the Oracle client is checked, so if a failure occurs it is probably due to the Oracle client not being present or installed incorrectly.
 - In the dialog box that follows, the **Data Source Name** and **Description** can be set to the SID for simplicity.
 - The **User Name** is not required.
 - The **Server** is the Oracle client connection to the data warehouse database, and is typically (but not necessarily) the SID.



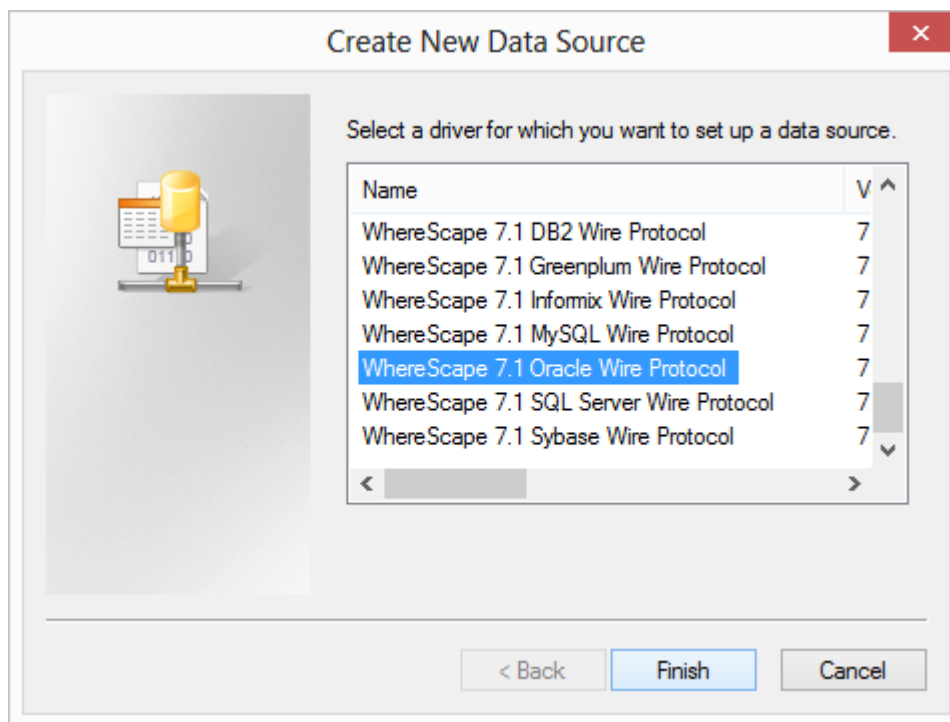
Note: The Server must be a valid Tnsnames entry.

Setting up a Data Direct ODBC Driver for Oracle:

- 1 In ODBC Data Source Administrator, click on the **System DSN** tab and click on the 'Add...' button to add a new entry.



- 2 Select the **WhereScape 7.1 Oracle Wire Protocol** driver from the list.
 - Click **Finish**.



- 3 Enter the the **Data Source Name** and **Description** - these can be set to the SID for simplicity.
 - The **Server** is the Oracle client connection to the data warehouse database, and is typically (but not necessarily) the SID.
 - The remaining fields are usually not required.

ODBC Oracle Wire Protocol Driver Setup

Pooling Bulk Client Monitoring Advanced Security About
General Advanced Security Performance Failover

Data Source Name: Help

Description:

Standard Connection

Host:

Port Number:

SID:

Service Name:

TNSNames Connection

Server Name:

TNSNames File

Edition Name:

Test Connect OK Cancel Apply

4 Switch to the **Advanced** tab and tick the following check-boxes:

- Enable Timestamp With Timezone
- Application Using Threads
- Enable SQLDescribeParam
- Enable N-CHAR Support

The screenshot shows the 'ODBC Oracle Wire Protocol Driver Setup' dialog box with the 'Advanced' tab selected. The 'Advanced' tab is highlighted with a red box. Several checkboxes are also highlighted with red boxes: 'Enable Timestamp With Timezone', 'Application Using Threads', 'Enable SQLDescribeParam', and 'Enable N-CHAR Support'. Other visible options include 'Local Timezone Offset', 'Default Buffer Size for Long/LOB Columns (in Kb): 1024', 'Describe at Prepare', 'Catalog Options', 'Report Recycle Bin', 'Enable Server Result Cache', 'Fetch TSWTZ as Timestamp', 'Timestamp Escape Mapping', 'Report Codepage Conversion Errors', 'Server Process Type', 'Initialization String', 'Login Timeout: 15', 'Query Timeout: 0', and 'Extended Options'. Buttons at the bottom include 'Test Connect', 'OK', 'Cancel', and 'Apply'.

5 To test this connection, click on **Test Connect**, otherwise click **OK**.

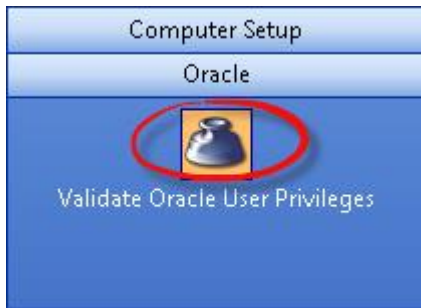
NOTE: When using the *SQL Admin* tool in conjunction with DataDirect's Oracle ODBC driver the *Enable SQLDescribeParam* option must be enabled in the ODBC driver setup menu.



TIP: If multiple PCs will access the data warehouse they must all use the same data source name as entered here.

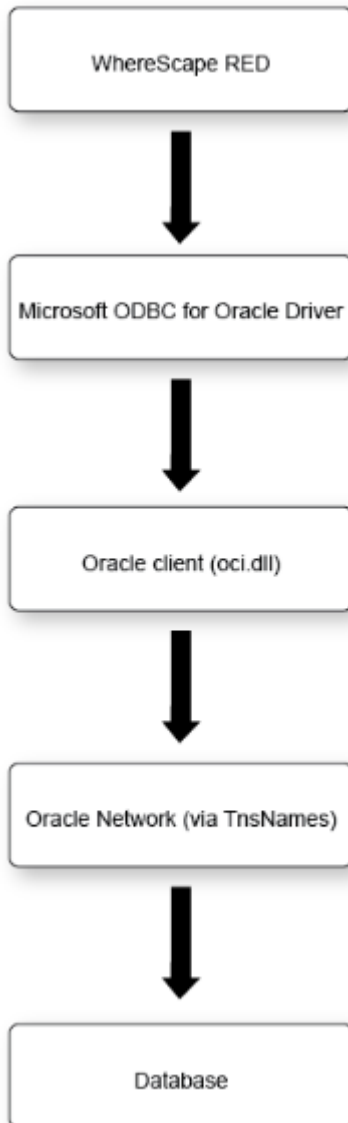
9.5 VALIDATING ORACLE USER PRIVILEGES

- 1 To validate the Oracle user privileges, click on the **Oracle** tab.
 - Then click on the **Validate Oracle User Privileges** icon.



This option tests that an ODBC connection can be made to the Oracle database running the data warehouse.

The diagram below shows some of the layers involved in making up this type of connection.



If this connection fails then we need to isolate where the failure is occurring. The Oracle documentation can provide assistance in resolving the problem. Starting from the bottom and working back up the layers we can perform the following tests:

- 1 Log on to the host running the database, and check that a direct connection to the database works via a tool such as sqlplus. This will test that the database is active and receiving connections.
- 2 Issue a Tnsping from a dos shell on the PC (e.g. tnsping ORCL) where ORCL is the tnsnames connection entry for the data warehouse database. This will check the Oracle network layer. If the Tnsping responds with an **OK** then the database server is visible through the network layer.
- 3 Connect to the data warehouse database using sqlplus from the PC. Again this test checks the network layer and a small part of the Oracle client software.
- 4 Attempt an ODBC connection using the Oracle supplied ODBC tool (normally called **Oracle ODBC Test** and located within the Oracle program menu). This tool is normally loaded as part of the Oracle client. It provides an independent test of the ODBC, Oracle client and network layers. Initially attempt to connect using the Oracle supplied ODBC driver. This will confirm that all layers apart from the ODBC driver are working correctly. If a failure occurs consult the Oracle documentation.
- 5 Perform the same test as step (4) but this time using the Microsoft ODBC for Oracle driver.

NOTE: A connect error **NA000** usually means the ODBC driver is not functioning correctly. This may be due to a Microsoft service pack dependency. Acquire the ODBC drivers from Microsoft and try again. The file is mdac_typ.exe and is a free download from Microsoft.

Some combinations of the Oracle client software and the Microsoft ODBC driver for Oracle result in a connection error **NA000**. If the connection fails and all other possibilities have been checked then the following may address the problem on Windows 2000 systems.

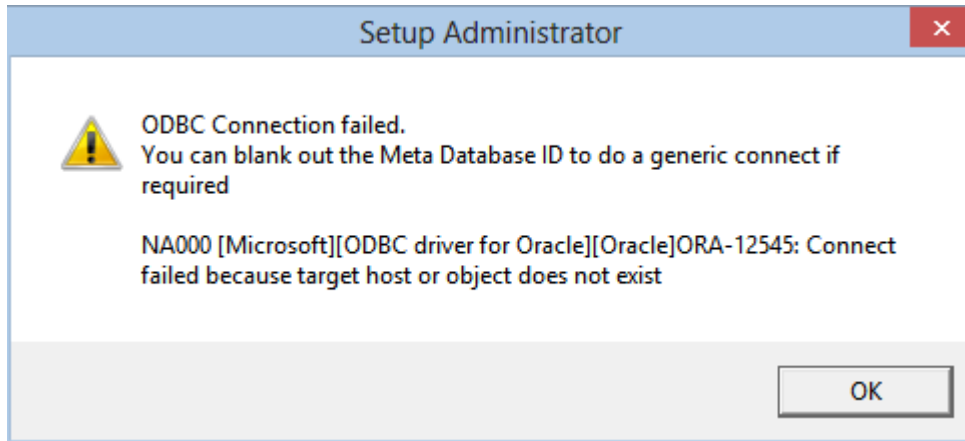
Run RegEdit

Under the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSDTC\MTxOCI entry edit OracleOciLib and set it to Oci.dll (The default is Ociw32.dll).

Another common problem is where multiple Oracle homes are in use (especially where Oracle Discoverer has been installed). Ensure that the PATH variable references an 8.1.7 or later Oracle directory. The OCI used by the Microsoft odbc driver utilizes features not found in older versions of the OCI.

If you are still unable to connect record the error message as displayed on connection failure and contact support@wherescape.com.

The following example shows the error message "2800 (Microsoft)(ODBC driver for Oracle)ORA_01017: invalid username/password; logon denied"



9.6 CREATING A METADATA REPOSITORY FOR ORACLE

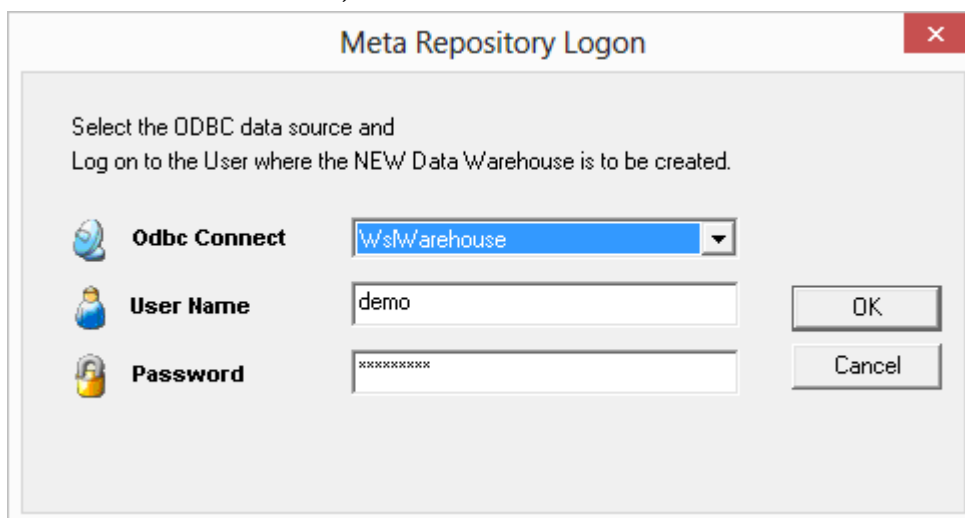
- 1 In WhereScape's Setup Administrator, click on the **Metadata Repository** tab in the left pane.
 - Then click on the **Create Metadata Repository** icon.



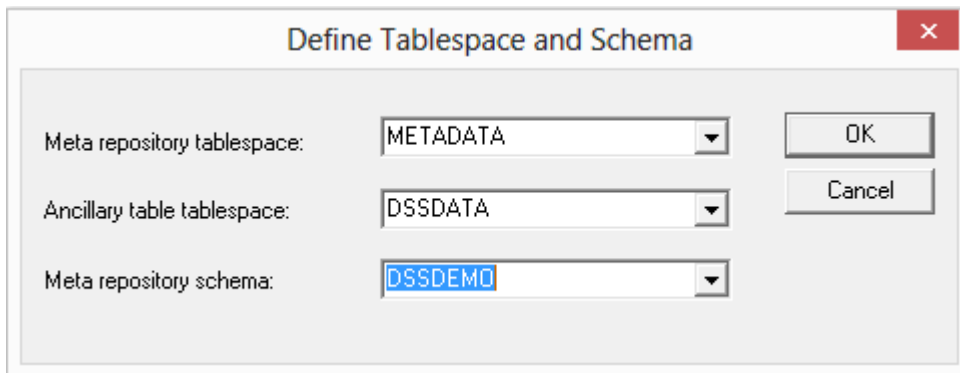
- 2 This option will create the WhereScape meta tables, sequences and procedures in the specified schema.

It checks to ensure that the schema, database or user does not contain an existing metadata repository before starting the metadata install.

- Click on Create Metadata Repository.
- Enter the **ODBC Connect**, **User Name** and **Password** and click **OK**.

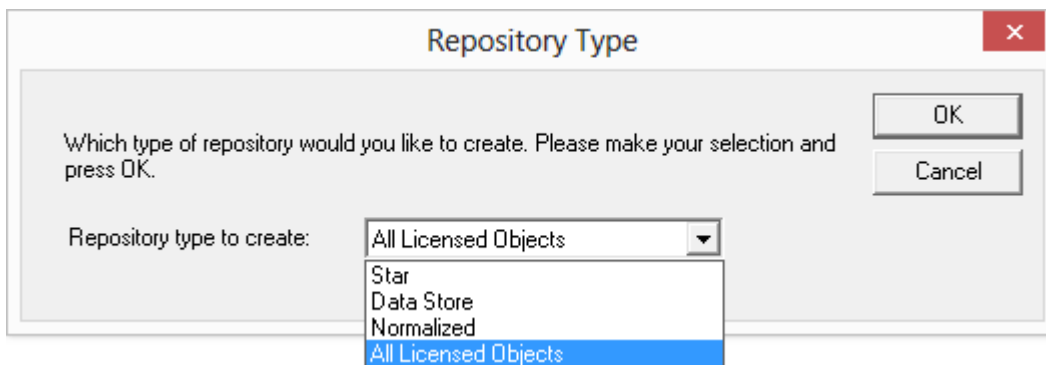
A screenshot of the 'Meta Repository Logon' dialog box. The title bar says 'Meta Repository Logon' with a close button. The main area contains the text: 'Select the ODBC data source and Log on to the User where the NEW Data Warehouse is to be created.' Below this are three fields: 'Odbc Connect' with a dropdown menu showing 'WsWarehouse', 'User Name' with a text box containing 'demo', and 'Password' with a text box containing '*****'. There are 'OK' and 'Cancel' buttons on the right.

- When creating an Oracle metadata repository, an additional screen is displayed requesting **Tablespace** information.
 - Select the required Tablespaces and click **OK**.
 - If Tablespaces are not specified, the user's default Tablespace will be used.



The dialog box titled "Define Tablespace and Schema" contains three dropdown menus and two buttons. The first dropdown menu is labeled "Meta repository tablespace:" and has "METADATA" selected. The second dropdown menu is labeled "Ancillary table tablespace:" and has "DSSDATA" selected. The third dropdown menu is labeled "Meta repository schema:" and has "DSSDEMO" selected. To the right of the dropdowns are "OK" and "Cancel" buttons.

- Depending on the type of license you have, the following dialog may be displayed:



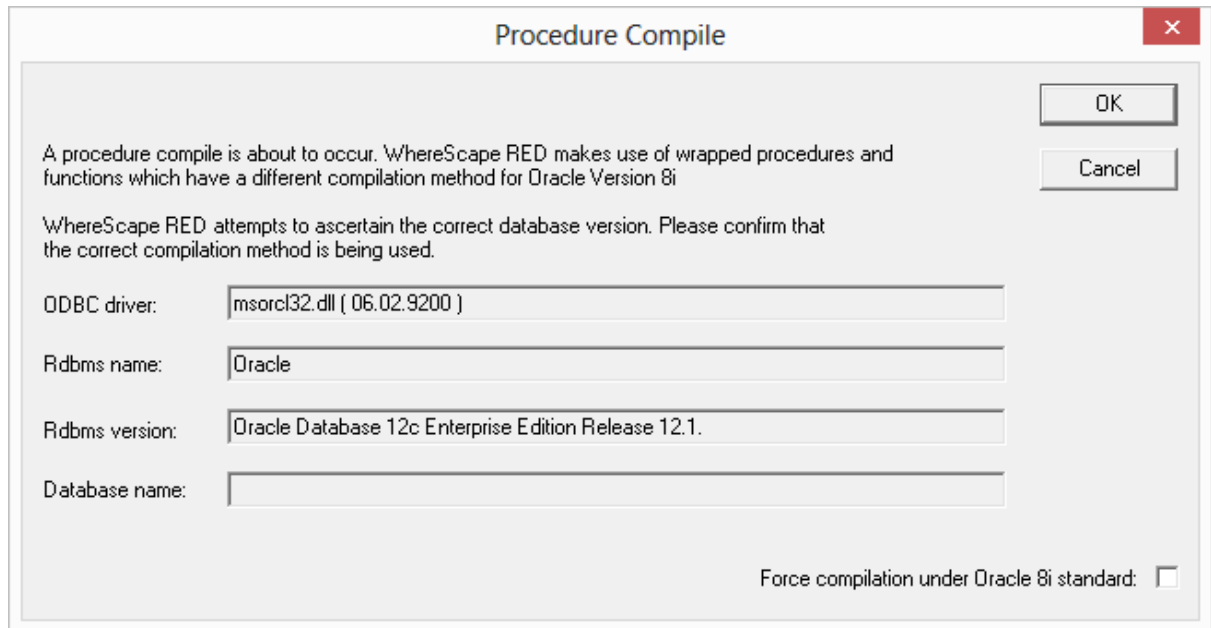
The dialog box titled "Repository Type" contains a text prompt and a dropdown menu. The text prompt says "Which type of repository would you like to create. Please make your selection and press OK." Below the text is a dropdown menu labeled "Repository type to create:" with a list of options: "All Licensed Objects", "Star", "Data Store", "Normalized", and "All Licensed Objects". The "All Licensed Objects" option at the bottom of the list is highlighted in blue. To the right of the dropdown menu are "OK" and "Cancel" buttons.

- If it is displayed, select the type of repository required and click **OK**.

Note: This is just setting the default objects visible in RED. They can later be changed under **Tools/Options**.

6 The procedure compile dialog is displayed:

- To force the compilation under the Oracle 8i standard, tick the check-box on the rightmost bottom corner of the Procedure Compile dialog. Otherwise, click **OK**.



If a procedure fails to compile then scroll to the right in the right pane to see if any of the returned error codes help in ascertaining what the problem is.

For **Oracle** common causes of procedure compilation failures are:

- Running an Oracle version prior to 8.1.7
- Insufficient room in the SYSTEM table space.
- Insufficient privileges. (See notes below)
- Memory parameters set too small in the Oracle instance.

Insufficient privileges on procedure compile (Oracle)

Compiles of the WhereScape procedures may fail where the required privileges have not been granted. The bulk of the WhereScape procedures are used by the scheduler, so they can be left uncompiled if the scheduler is not to be used. The following procedures however must successfully compile for WhereScape RED to function correctly:

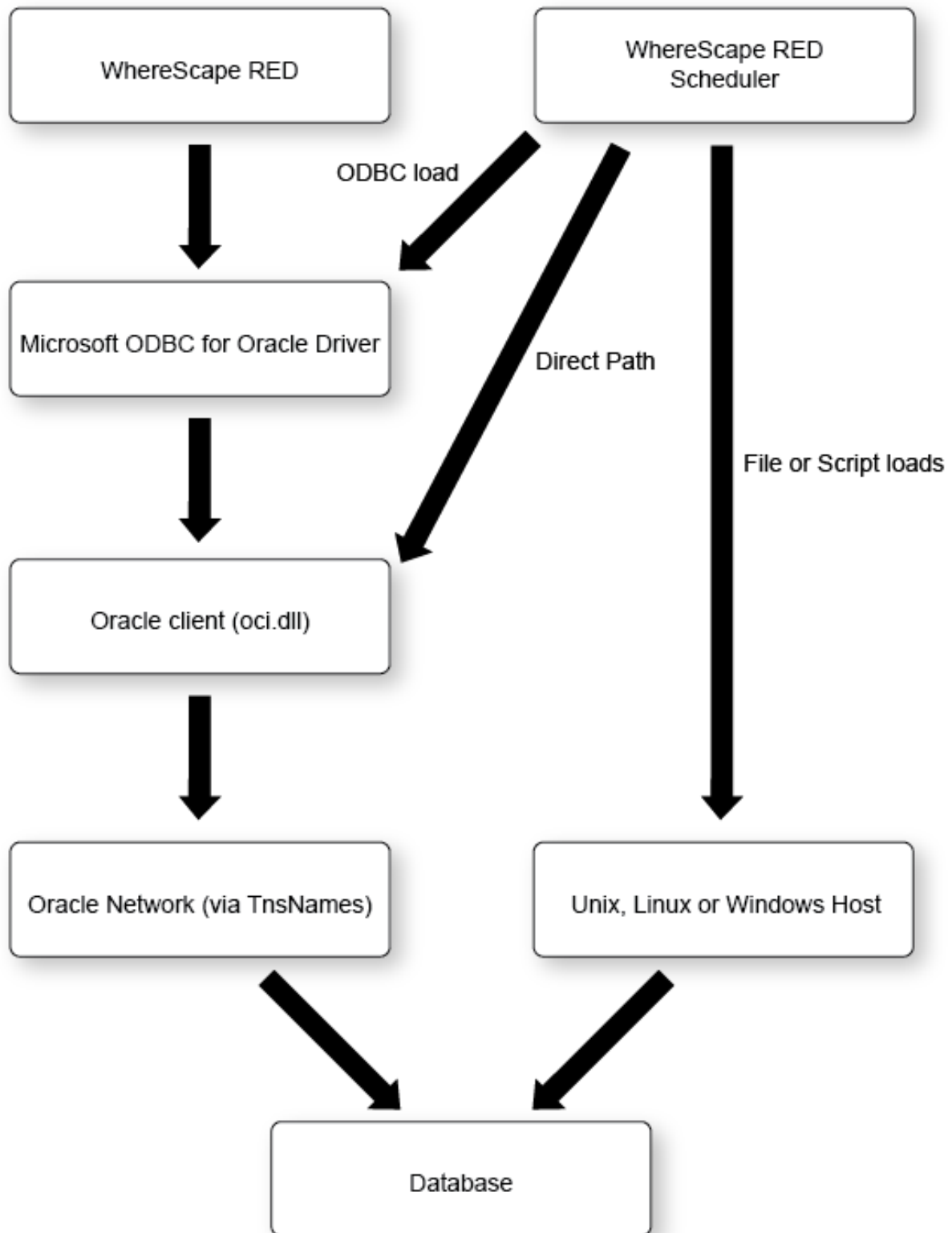
Ws_Procedure_Save, Ws_Procedure_Compile, WsWrkAudit, WsWrkError, WsParameterRead and WsParameterWrite

If one or more procedures fail to compile then use the **Validate Metadata Repository** option to compile the procedures once the problem has been rectified.

9.7 INSTALLING A WINDOWS SCHEDULER FOR ORACLE

NOTE: The WhereScape windows scheduler for Oracle requires the Oracle client 8.1.7 or later software. The Oracle client software should be a close version match to the Oracle Database version, that is the same major release or one release either side.

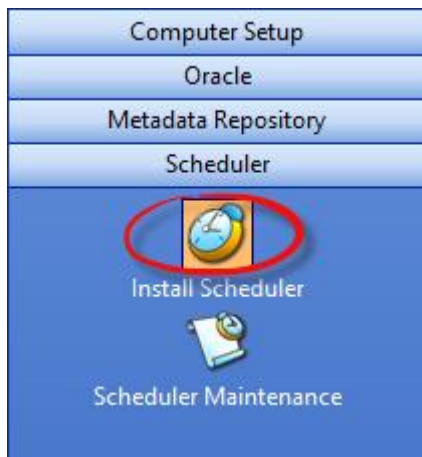
This option installs and starts a copy of the WhereScape Windows scheduler. The scheduler runs as a system service and can also be maintained through the normal Windows services functions. The following diagram provides an overview of how WhereScape RED and the scheduler access an **Oracle** database.



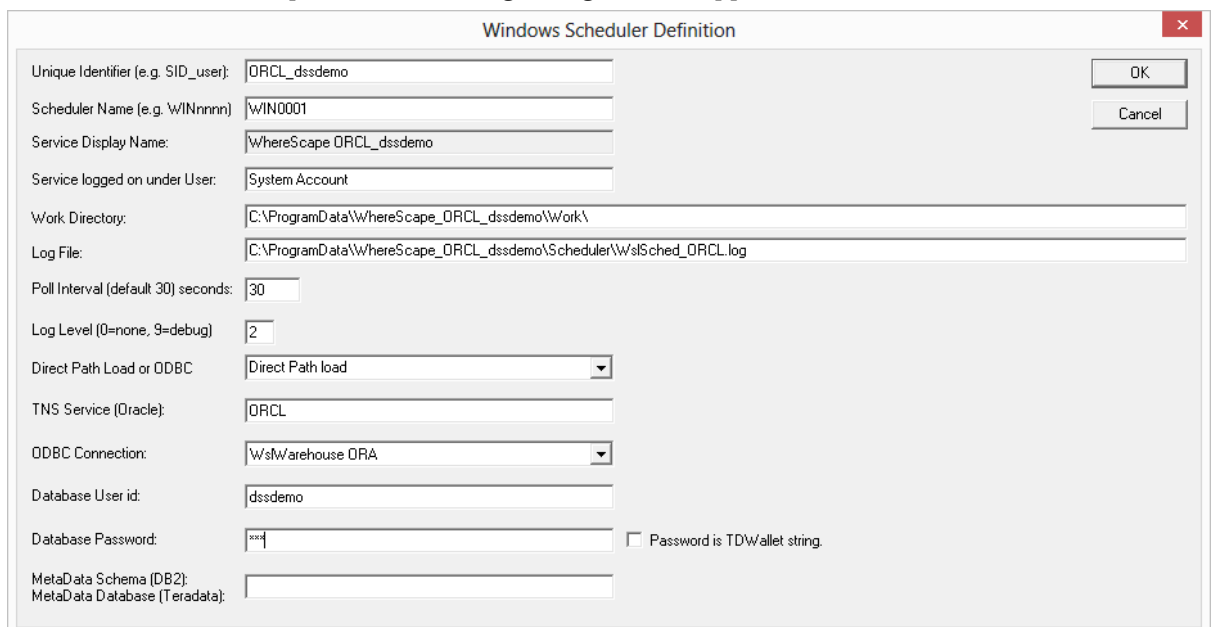
The scheduler makes use of the ODBC layer to perform the routine tasks involved in scheduling and running jobs. When loading data from an ODBC source the scheduler can either use the **ODBC layer** or the **Oracle direct path** load facility.

For file or script based loads, the scheduler actions the load on the host system utilizing Oracle SQL* Loader.

- 1 In WhereScape's Setup Administrator, click on the **Scheduler** tab in the left pane.
 - Then click on the **Install Scheduler** icon.



- 2 On selection of this menu option the following dialog box will appear:

A screenshot of the 'Windows Scheduler Definition' dialog box. The fields are filled with the following values:

- Unique Identifier (e.g. SID_user): ORCL_dssdemo
- Scheduler Name (e.g. WINnnnn): WIN0001
- Service Display Name: WhereScape ORCL_dssdemo
- Service logged on under User: System Account
- Work Directory: C:\ProgramData\WhereScape_ORCL_dssdemo\Work\
- Log File: C:\ProgramData\WhereScape_ORCL_dssdemo\Scheduler\W\Sched_ORCL.log
- Poll Interval (default 30) seconds: 30
- Log Level (0=none, 9=debug): 2
- Direct Path Load or ODBC: Direct Path load
- TNS Service (Oracle): ORCL
- ODBC Connection: WsWarehouse ORA
- Database User id: dssdemo
- Database Password: ***
- Metadata Schema (DB2):
- Metadata Database (Teradata):

The 'Password is TDWallet string.' checkbox is unchecked. 'OK' and 'Cancel' buttons are visible in the top right corner.

Unique Identifier

Each scheduler requires a unique name (possibly the SID and user e.g. ORCL_dssdemo). This unique name is the name by which the service is referenced.

Scheduler Name

When scheduling a job with WhereScape RED it is possible to assign the job to a specific scheduler. Each scheduler should therefore have a unique name. The default name is WIN0001. Most users change this to be the scheduler machine's name followed by a number, e.g.: DWHP01, where DWHP is the name of the windows scheduler server.

Service Display Name

This field displays the name that the scheduler will appear under when viewed through the Windows services. It is a combination of the string 'WhereScape' and the unique identifier.

Service Logged on Under User

This is the name of the Windows local user account that the scheduler service will use when logging on. The default setting is **System Account**, also known as the **Local System** service account. To set the scheduler to use a Windows domain user account with a password, use the Service Account option from the right-click menu. If a domain user account is specified then the contents of this field (on the **Windows Scheduler Definition** dialog) are ignored.

NOTE: A domain user account will require the **log on as a service** privilege in order for it to work. This is assigned automatically when setting the log on account using the Services console in Windows.

Work Directory

The directory used to store some temporary files created by the scheduler.



TIP: Two schedulers should not use the same work directory.

Log File

The full path where the scheduler service will write a log of the various actions undertaken. This file may get large if log level 9 is set.

Poll Interval

This defines the frequency of the scheduler polling the database, looking for work. The default is 30 seconds. If set to a very low interval such as 1 then the impact on the database may be noticeable.

Log Level

This defines the level of detail written to the log file. The following logging levels are available:

- 0 = will result in no information being written.
- 1 = fatal errors only
- 2 = major events
- 3 = major events and return stats to caller
- 9 = full debug



TIP: Setting Log Level 3 will return timing stats during ODBC based loads.

A common Oracle error is NA000, object invalid which implies that the `Ws_Job_Exec_44` procedure was not compiled successfully during the latest install or upgrade, or that the procedure has become invalid, due to a change in one of the WhereScape meta tables or procedures. Validate the metadata to ensure that this procedure has been compiled.

Direct Path load or ODBC

This option is only relevant for ODBC based data loads in Oracle. When ODBC loads write to the data warehouse database there are two load types available to be selected from the drop-down list:

- Oracle Direct Path loading using the OCI library - **Direct Path loads** are considerably faster and should be used in most situations. Normally if a Direct Path load can not be undertaken the scheduler will revert to an ODBC based load.
- Indirect load using ODBC - the `Odbc sql` load option is presented here primarily as a means of disabling Direct Path loads in the event of problems.

NOTE: Direct Path loads have strict date format requirements. For more information about this, see section **9.4 ODBC Based Load** in the WhereScape RED User Guide.

TNS Service (Oracle)

If you select the **Direct Path load** option from the Direct Path load or ODBC drop-down-list, enter the database server as defined in the tnsnames.ora file on this field. Normally this will be the same name as entered for the 'server' in an ODBC source definition.

ODBC Connection

The ODBC connection field requires the ODBC source name that will provide a connection to the data warehouse system. The scheduler uses ODBC to connect to the data warehouse.

Database User Id

The User Name the Scheduler will use when connecting to the data warehouse database.

Database Password

The password for the User the Scheduler will use when connecting to the data warehouse database.

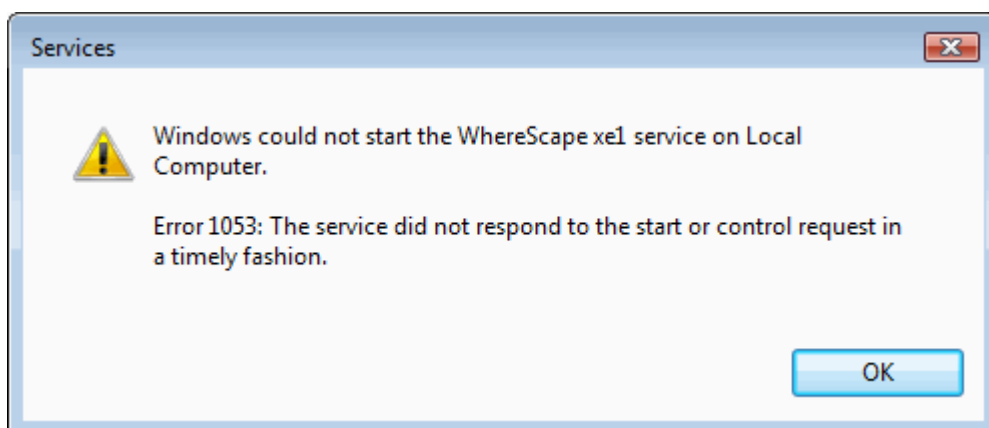
Metadata Schema (DB2) / Metadata Database (Teradata)

This is not set for Oracle schedulers.

Note: A windows scheduler must be run as either the system service account or as a user with administrator privileges in windows.

Oracle Error Messages:

The following error message may appear as the scheduler starts. In such an event the scheduler will not function until the problem is resolved.



This error may be the result of an old version of the Oracle Call Interface (OCI) dynamic link library (dll) being in the path. The WhereScape scheduler requires Oracle client 8.1.7 or above. This situation can occur when an Oracle product such as Discoverer is installed. Discoverer may install an older version of the oci.dll file. If it does, its location will be put in the Windows path in front of the 8.1.7+ client version.

A work-around is to copy an 8.1.7 or later version of the oci.dll file into the WhereScape program directory. To see if an oci.dll is valid, right-click on the file and select properties and then version. A valid oci.dll will have version 8.1.7 or later.

9.8 INSTALLING A UNIX SCHEDULER FOR ORACLE

The Unix scheduler must be manually installed and maintained.

The scheduler consists of a series of shell scripts that are controlled by the cron process. The bulk of the work undertaken by the WhereScape scheduler takes place in the WhereScape procedures within the database. To obviate the need to provide a port to each Unix platform this series of shell scripts are used. They should run with no modification on all Unix platforms and have been tested under Unix offerings from Sun, HP, IBM and Linux.

The Unix scheduler can process every type of load except for ODBC loads, Native ODBC loads and XML loads. It cannot process OLAP Cubes.

WARNING: For UNIX/Linux scheduler processing, the **Encrypt User** and **Password** options in RED's Repository Privacy Settings cannot be used. Encrypt options are only supported when using a Windows scheduler.

Installation

The installation of the Unix scheduler needs to be undertaken by someone familiar with the Unix environment and the running of database processes from Unix. The tasks that need to be undertaken are:

1. Unix User and Shell

Acquire a Unix user name that has access to cron and the database data warehouse environment. A reasonable quantity of disk space is also required if the automated backups of the metadata repository are used. The user should log on to Unix without passing through a menu system if possible. Log on to the user.

The WhereScape scheduler consists of a number of shell scripts. By default some of these shell scripts will run in the Bourne shell and some will run in the Korn shell. If Linux is being used then the Bash shell is used exclusively.

Note: During setup it is recommended that the Korn shell be used as the default shell for the user installing the scheduler (or Bash if running Linux).

2. Create directories

Create the following directories (if not already present) to be used by the WhereScape scheduler. If the directory names are not appropriate and others must be used then it will be necessary to alter the WhereScape supplied scripts.

- wsl (top level directory for WhereScape directories)
- wsl/bin (used to store the WhereScape scripts)
- wsl/sched (used for the scheduler job and log files)
- wsl/sched/job (used for the scheduler job files)
- wsl/sched/joblog (used for the scheduler job logs)
- wsl/sched/log (used for the scheduler logs)
- wsl/export (used for the automated exports of the metadata)
- wsl/expback (used to hold compressed copies of the metadata exports)
- wsl/log (used for export and archive logs)
- wsl/mon (used for the monitoring process)
- wsl/mon/db (used for database monitoring scripts and control files)
- wsl/mon/job (used for job monitoring scripts)
- wsl/mon/log (used for monitoring log files)

3. Copy supplied Unix files

FTP the scheduler files shipped with WhereScape RED to the wsl/bin directory on the Unix system (use ASCII mode). All files should be copied over.

WhereScape RED ships with several different versions of the unix files for the scheduler. The appropriate version of the files should be copied to the server. The following lists the files shipped:

- ...\\WhereScape\\Oracle\\Unix\\Version_nnn\\...
- ...\\WhereScape\\Oracle\\Linux\\Version_nnn\\...

NOTE: There may be several different versions of the scheduler files for a given database and platform (Unix or Linux). These are identified by version numbers **nnn**. For example, there may be three different folders in ...\\WhereScape\\Oracle\\Linux\\: Version_010, Version_410 and Version_560. The highest version number script less than or equal to the version of RED in use should always be used.

In the table that follows, the **nnn** in file names refers to the version number of the script. Some scripts will have **010**, some will have **411**, others **560** or **680**, etc.

A definition of the files follows:

File	Purpose
oraenv (for Oracle)	Contains the template of a script for setting environmental variables. This needs to reside in the home directory of the Unix user and must be tailored for the database user and environment. It may be renamed to reflect the actual environment. (e.g. oraenv_SID_user)
crontab	Contains the template of the required cron commands. This file needs to be modified to include the HOME directory of the Unix user, and the name of the environment file. It is then used to update the crontab tables.
ws_start_nnn.sh	Script to start the scheduler. This script is normally called by the ws_sched_check_nnn.sh script and is passed the database environment file name as a parameter.
ws_sched_check_nnn.sh	Script to check that the scheduler is running. This script is normally executed by cron every 20 minutes. If the scheduler is found to be absent it attempts to start a new iteration. By default the scheduler will terminate if its database is shut down. This script will therefore start the scheduler after a system start-up or a database start-up.
ws_sched_nnn.sh	The actual scheduler script. This script looks for new scheduler jobs every x seconds. The default is every 30 seconds. This script must be modified to change the default behavior. It calls the ws_job_wait_010.sh script to perform the actual checking and initiating of a job. It terminates when ws_job_wait_010.sh returns an error, and relies upon cron for a restart.
ws_sched_status_nnn.sh	This script is called by ws_sched_nnn.sh to update the scheduler status in the scheduler metadata. It's a wrapper of the metadata database procedure ws_sched_status (i.e. it calls the database procedure ws_sched_status and returns it's results to the calling script).
ws_job_wait_nnn.sh	This script performs the bulk of the work. It is called by ws_sched_nnn.sh on a regular basis. It checks for any job waiting to run. If it locates a job that is ready to run it will create and start a script in the \$HOME/wsl/sched/job directory for each thread of that job. These job threads in turn perform the actual processing and log their results to \$HOME/wsl/sched/joblog. Under normal operation it is not necessary to check or inspect these files.

ws_job_check_nnn.sh	<p>This script is called to check the status of running jobs. It is called by ws_sched_nnn.sh on a regular basis (the interval is defined in ws_sched_nnn.sh). It checks that any job running in the scheduler is actually running in the Unix environment. If a job is not found under Unix then it is aborted in the scheduler. This scenario can occur as the result of a database or machine crash. If a job abort is required the results are logged to \$HOME/wsl/sched/log/job_check_'env file name'.log. Under normal operation it is not necessary to check or inspect this file.</p>
meta_backup_nnn.sh	<p>This script is normally called by the cron process once or maybe more times a day to perform an automated export backup of the metadata tables. It creates a number of command files and a database export in the \$HOME/wsl/export directory named with the date/time of the export.</p> <p>It is recommended that users copy the new meta_backup_680.sh file to use Datapump's expdb instead the deprecated exp tool. This version uses the data pump export executable expdp.</p> <p>It assumes that the scheduler and the Oracle database reside on the same server.</p>
meta_archive_nnn.sh	<p>This script is normally called by cron once a day to compress and tar the files in the \$HOME/wsl/export directory and relocate them to the \$HOME/wsl/expback directory. This action is performed purely to save on disk space usage.</p>
meta_restore_nnn.sh	<p>This script can be executed to restore from a meta repository backup. It will overwrite any existing metadata. It can be used to restore to a schema/database/user other than the one from which the backup was taken. It does not however restore the WhereScape procedures. If a full recovery or a new user create is required then the WhereScape metadata install would normally be performed before actioning this restore. If the restore is from an export that has been moved to the expback directory you will need to uncompress the export and move it and associated files back to the export directory.</p> <p>It is recommended that users copy the new meta_restore_680.sh file to use Datapump's impdp instead the deprecated imp tool. This version uses the data pump import executable impdp.</p> <p>It assumes that the scheduler and the Oracle database reside on the same server.</p>

cleanup_nnn.sh	This script is normally executed by cron and cleans up files created by the WhereScape scheduler. It deletes old versions of the jobs, job logs, and backups. It will also delete any temporary scripts created in the running of host scripts or file loads from within the scheduler. The location of these temporary scripts is defined within the WhereScape RED Unix Connection object. The default value in the cleanup script is /tmp and this must be altered if a different temporary directory is used for a Connection.
ws_mon_nnn.sh	The main monitoring script. This script in turn calls the database and job monitoring scripts to perform the monitoring. By default the monitor process checks the jobs every 15 minutes and the database every 60 minutes. This script can be edited to change the frequency of the checks if required. Also a log level of 9 can be set to produce debug logging.
ws_mon_db_nnn.sh	This script checks the database status and if necessary calls the notify script to notify of a database down. Also checks to see if the criteria for database monitoring has changed and if changed calls the refresh script.
ws_mon_notify_nnn.sh	This script checks that notifications are valid for the day/time and if valid sends a notification.
ws_mon_refresh_nnn.sh	This script refreshes the scripts and command files stored in the mon/db directory. These scripts and command files are used to ascertain the action should the database be unavailable.
ws_mon_jobs_nnn.sh	This script performs the checking of jobs for which monitoring is defined. If a job has a notification required then this script will issue the notification.
ws_mon_start_nnn.sh	This script starts the monitor process
ws_mon_check_nnn.sh	This script checks that the monitor process is running. If not running the script will start the monitor process. This script is normally called via cron every 30 minutes.

4. Set permissions

Change the permissions on the copied scripts so that they are executable. (e.g. `chmod 750 $HOME/wsl/bin/*.sh`)

5. Create an environment file

Using the environment file (oraenv) as a template and adhering to the variable names within, create an environment file for the database environment and user under which the scheduler is to run. This file should be located in the \$HOME directory. It may be called anything but a meaningful name may be oraenv_SID_user for a given Oracle scheduler. Check the users .profile and database environment file to see the normal settings. For oracle, check to see if the TNS_ADMIN variable is in use and if so, include it in the environment file.

Make the following replacements in the environment file:

- Replace all occurrences of `_HOME_` with the full pathname of your home directory
- Replace `_USER_` with the data warehouse database user name
- Replace `_PASSWORD_` with the data warehouse database user password.
- Replace all occurrences of `_SID_` with the sid of the data warehouse database
- Replace all occurrences of `_ORAHOME_` with the full pathname of the Oracle home directory

6. Setup crontab

Ascertain if there are any existing crontab entries, and if so save them to a file for re-inclusion later. (e.g. `crontab -l` `crontab -l >/tmp/my_crontab`)

Edit the crontab file and perform the following changes. Also check all entries and make sure you are happy with the times and actions:

- Change all occurrences of `_HOME_` with the full pathname of the Unix users home directory.
- Change all occurrences of the environment file name (oraenv, dbenv or tdenv) to the name of the file created in step (4).
- Include any existing crontab entries as recorded above.

7. Initialize crontab entries

Load the crontab file into cron (e.g. `crontab crontab`).

Perform a crontab list to ensure it all looks OK (e.g. `crontab -l`)

8. Start Scheduler

Manually start the scheduler by issuing the `ws_sched_check_nnn.sh` script with the environment file name from step (4) as the parameter (e.g. `ws_sched_check_010.sh oraenv`). If the scheduler is not running it will be started. Execute the script a few more times to ensure that only one copy of the scheduler is started. Check that the scheduler is running by executing a 'ps -ef' statement and checking that the `ws_sched_560.sh` process is present.

9. Start the Monitor process

Oracle only. Manually start the monitor by issuing the `ws_mon_check_010.sh` script with the environment file name from step (4) as the parameter (e.g. `ws_mon_check_010.sh oraenv`). If the monitor is not running it will be started. Execute the script a few more times to ensure that only one copy of the scheduler is started. Check that the monitor is running by executing a `'ps -ef'` statement and checking that the `ws_mon_010.sh` process is present.

10. Test the Scheduler

Run a job from WhereScape RED and ensure that it works. In the event of the job not starting or going into and remaining in a pending state then the following investigations should be undertaken:

- Ensure all WhereScape procedures are compiled. Use the Validate metadata option in the Setup Administrator to confirm that all procedures are valid.
- Review the job logs in \$HOME/wsl/sched/joblog.
- Review the logs in \$HOME/wsl/sched/log.
- Isolate the name of the job and locate the relevant files for this job in \$HOME/wsl/sched/job and \$HOME/wsl/sched/joblog if they exist. Mail these files together with the files in \$HOME/wsl/sched/log to support@wherescape.com. It may be necessary to remove password references from the file(s) in \$HOME/wsl/sched/job.

NOTE: It is possible to stop a Linux/UNIX Scheduler from within WhereScape RED. See the WhereScape RED User Guide for details.

UNIX SCHEDULER SETUP FOR HADOOP LOADS USING ORACLE'S BIG DATA CONNECTORS

To set up the UNIX scheduler for Oracle loads from Hadoop using Oracle's Big Data connectors with OSCH and OLH, the following environment variables need to be edited in oraenv:

Example

```
export JAVA_HOME=/usr/lib/jvm/java-openjdk
export HADOOP_HOME=/home/oracle/hadoop-2.4.1

export PATH=$ORACLE_HOME/bin:$HADOOP_HOME/bin:$PATH
export JAVA_LIBRARY_PATH=$HADOOP_HOME/lib/native:$JAVA_LIBRARY_PATH

export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="$HADOOP_OPTS -Djava.library.path=$HADOOP_HOME/lib/native"
```

For either OSCH or OLH, add one of the following to oraenv:

```
#OSCH
export OSCH_HOME=/home/oracle/orahdfs-3.1.0
export PATH=$OSCH_HOME/bin:$PATH
export HADOOP_CLASSPATH=$OSCH_HOME/jlib/*:$HADOOP_CLASSPATH
```

```
#OLH
```

```
export OLH_HOME=/home/oracle/oraloder-3.2.0-h2
export HADOOP_CLASSPATH=$OLH_HOME/jlib/*:$HADOOP_CLASSPATH
```

NOTE1: Please restart the scheduler after you modified environment variables in oraenv.

NOTE2: The following Oracle directory objects will need to be created with certain privileges:

OSCH

OSCH directory objects

OSCH bin Directory

OSCH External Tables Directory

OSCH Log Directory

Example

```
CREATE DIRECTORY osch_bin_path AS '/opt/orahdfs-3.1.0/bin';
GRANT EXECUTE ON DIRECTORY osch_bin_path TO c##oschuser;
GRANT READ ON DIRECTORY osch_bin_path TO c##oschuser;
CREATE DIRECTORY exttab_default_directory AS '/home/oracle/exttab';
GRANT READ ON DIRECTORY exttab_default_directory TO c##oschuser;
GRANT WRITE ON DIRECTORY exttab_default_directory TO c##oschuser;
CREATE DIRECTORY osch_log_path AS '/home/oracle/temp'
GRANT READ ON DIRECTORY osch_log_path TO c##oschuser;
GRANT WRITE ON DIRECTORY osch_log_path TO c##oschuser;
```

CHAPTER 10

10. DB2 INSTALLATION AND CONFIGURATION

IN THIS CHAPTER

10.1 Installation and Configuration Introduction	115
10.2 Installing RED for DB2	117
10.3 Creating a DB2 Schema	118
10.4 Configuring a DB2 Metadata Connection (DSN).....	122
10.5 Creating a Metadata Repository for DB2.....	124
10.6 Installing a Windows Scheduler for DB2.....	126
10.7 Installing a Unix Scheduler for DB2	130

10.1 INSTALLATION AND CONFIGURATION INTRODUCTION

The steps provided here are an indication of the normal process required to install and configure WhereScape RED. In some environments additional processes will be required.

- 1 Install the IBM DB2 9.7 or 10.1 client access software on a suitable PC.
- 2 Install the WhereScape RED software onto your PC. See *Installing RED for IBM DB2* (see "10.2 Installing RED for DB2" on page 117)
- 3 Add your license key. See *Adding your License* (see "3. Adding Your License" on page 15)
- 4 Check your Windows version. See *Checking your Windows Version* (see "4. Checking Your Windows Version" on page 18)
- 5 Check your ODBC Drivers. See *Checking your ODBC Drivers* (see "5. Checking Your ODBC Drivers" on page 19)
- 6 Designate a database that is to be used as the data warehouse database. For testing or evaluation purposes this may be a PC based database. WhereScape RED requires the DB2 database version 9.7 or greater as the data warehouse database. Create the database schema that is to be used to hold the metadata and data warehouse tables. Allocate the privileges required by WhereScape RED. See *Creating a DB2 Dss Schema* (see "10.3 Creating a DB2 Schema" on page 118)
- 7 Add the definition of an ODBC data source to connect to the data warehouse database. See *Configuring a DB2 Metadata Connection (DSN)* (see "10.4 Configuring a DB2 Metadata Connection (DSN)" on page 122)
- 8 Create the WhereScape metadata repository under the data warehouse schema that you created. See *Creating a Metadata Repository for IBM DB2* (see "10.5 Creating a Metadata Repository for DB2" on page 124)
- 9 Install a Windows Scheduler. See *Installing a Windows Scheduler for IBM DB2* (see "10.6 Installing a Windows Scheduler for DB2" on page 126)
- 10 Install a Unix Scheduler. See *Installing a Unix Scheduler for IBM DB2* (see "10.7 Installing a Unix Scheduler for DB2" on page 130)
- 11 If required, load the WhereScape tutorial tables into a separate database schema. It is highly recommended that the tutorials be undertaken. See *Creating Tutorial Data* (see "11.1 Creating Tutorial Data" on page 137)

Problem Solving

If problems occur in your install that cannot be resolved proceed as follows when running the WhereScape's Setup Administrator utility:

- 1 Create a log file by selecting **Log/Start Log** from the menu bar. Enter a file name.
- 2 Step through at least:
 - Computer Setup --> Check Windows version
 - Computer Setup --> Check ODBC driver
 - Computer Setup --> Validate/Add Odbc source
 - Create/Validate Meta Repository --> Validate meta repository
- 3 Select **Log/Stop Log** to close off the log
- 4 Edit the log and remove any passwords that appear.
- 5 Send your log file to support@wherescape.com

10.2 INSTALLING RED FOR DB2

Installation Prerequisites

WhereScape RED requires certain client and server components.

Client Prerequisites

- Windows Vista, Windows 7 or Windows 8.
- 250MB available disk space
- ODBC Version 3.0 (normally shipped as part of the Operating System).
- For building a data warehouse, a version 9.7 database client or greater is required.

Server Prerequisites

- One of the following data warehouse database platforms is required: IBM DB2 9.7, 10.1. or 10.5.
- To use the WhereScape RED Scheduler, a UNIX, LINUX or Windows based server is also required (this may be the database server). WhereScape RED only supports sh and ksh under Unix and bash under Linux.

Installation of WhereScape RED

To install WhereScape RED, see *Installing WhereScape RED* (see "2. Installing WhereScape RED" on page 9)

10.3 CREATING A DB2 SCHEMA

The creation of the DB2 database and schema that will be used to administer and maintain the data warehouse must be performed outside this utility. If required, the DB2 command window can be used to execute the required sql, or the database administrator should be contacted.

NOTE1: Users and Groups in DB2 are not managed by DB2, rather they're mappings from operating system users and groups. The following example assumes DB2 is installed locally on windows. It also assumes WhereScape RED is running on the same windows machine. For more complex scenarios, support@wherescape.com or your database administrator.

NOTE2: WhereScape metadata tables should not be compressed. Possible tables for compression in DB2 are tables that have many more SELECT queries compared to INSERT, UPDATE and DELETE queries. Most large metadata tables are accessed by the WhereScape RED scheduler in the form of regular INSERT statements, so are not candidates for compression. Compressing metadata tables may adversely affect the performance of the WhereScape RED scheduler.

To complete the tutorials, the following database objects are required:

- one database
- two schemas (one for the data warehouse and one for the tutorial data)
- one buffer pool
- one system temporary tablespace
- one (or more) tablespaces

For our data warehouse, or decision support system, we will create a user, 'dssdemo'. An example sql script to create this 'dssdemo' user is shipped and can be found in the WhereScape RED program directory:

- ...\\DB2\\Recreate_dssdemo.sql

Please note that **this script will need to be modified** to reflect the data warehouse database and schema chosen at your site.

Note: If the user creating the metadata is not the same user who will sign into WhereScape RED, additional grants to those below will be required.

An example script follows. The database created is **wscape**, the data warehouse schema is **dssdemo**, the schema for tutorials is **tutorial**, the buffer pool is **RED**, the system temporary tablespace is **REDtemp01**, the two tablespaces created are **metadata** and **dssdata**.

```
CREATE DATABASE wscape
AUTOMATIC STORAGE YES
ON 'c:\'
DBPATH ON 'c:\'
USING CODESET IBM-1252
TERRITORY US COLLATE
USING SYSTEM PAGESIZE 4096;

CONNECT TO wscape;

CREATE SCHEMA dssdemo;

CREATE SCHEMA tutorial;

CREATE BUFFERPOOL RED PAGESIZE 32768;

CREATE SYSTEM TEMPORARY TABLESPACE REDtemp01
PAGESIZE 32 K
MANAGED BY AUTOMATIC STORAGE
EXTENTSIZE 32 OVERHEAD 10.67 PREFETCHSIZE 32 TRANSFERRATE 0.04
BUFFERPOOL RED;

CREATE TABLESPACE metadata
PAGESIZE 32 K
MANAGED BY SYSTEM
USING ('C:\DB2TS\METADATA')
EXTENTSIZE 32 OVERHEAD 10.67 PREFETCHSIZE 32 TRANSFERRATE 0.04
BUFFERPOOL RED
DROPPED TABLE RECOVERY OFF;

CREATE TABLESPACE dssdata
PAGESIZE 32 K
MANAGED BY SYSTEM
USING ('C:\DB2TS\WHOUSE')
EXTENTSIZE 32 OVERHEAD 10.67 PREFETCHSIZE 32 TRANSFERRATE 0.04
BUFFERPOOL RED
DROPPED TABLE RECOVERY OFF;

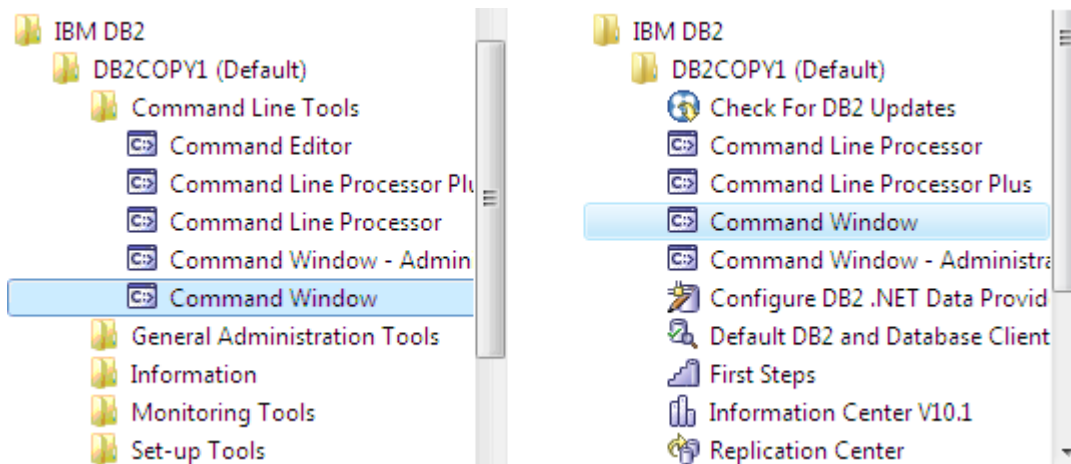
GRANT DBADM ON DATABASE TO USER username;
```

Where username is:

- the name of any users creating new metadata repositories
- the name of any users validating any existing metadata repositories
- the name of any windows users running the WhereScape RED windows scheduler

The default user running a windows scheduler is "system". You can change the windows scheduler to run as a different user via the services control panel in windows.

The best way to run this script is to use the **IBM DB2 Command Window**. Start the command window from the start menu in Windows:



Start the command line processor by entering:

```
db2 -td;
```

You should see this output:

```
Administrator: DB2 CLP - DB2COPY1 - db2 -td;
C:\Program Files\IBM\SQLLIB\BIN>db2 -td;
(c) Copyright IBM Corporation 1993,2007
Command Line Processor for DB2 Client 10.1.0

You can issue database manager commands and SQL statements from the command
prompt. For example:
  db2 => connect to sample
  db2 => bind sample.bnd

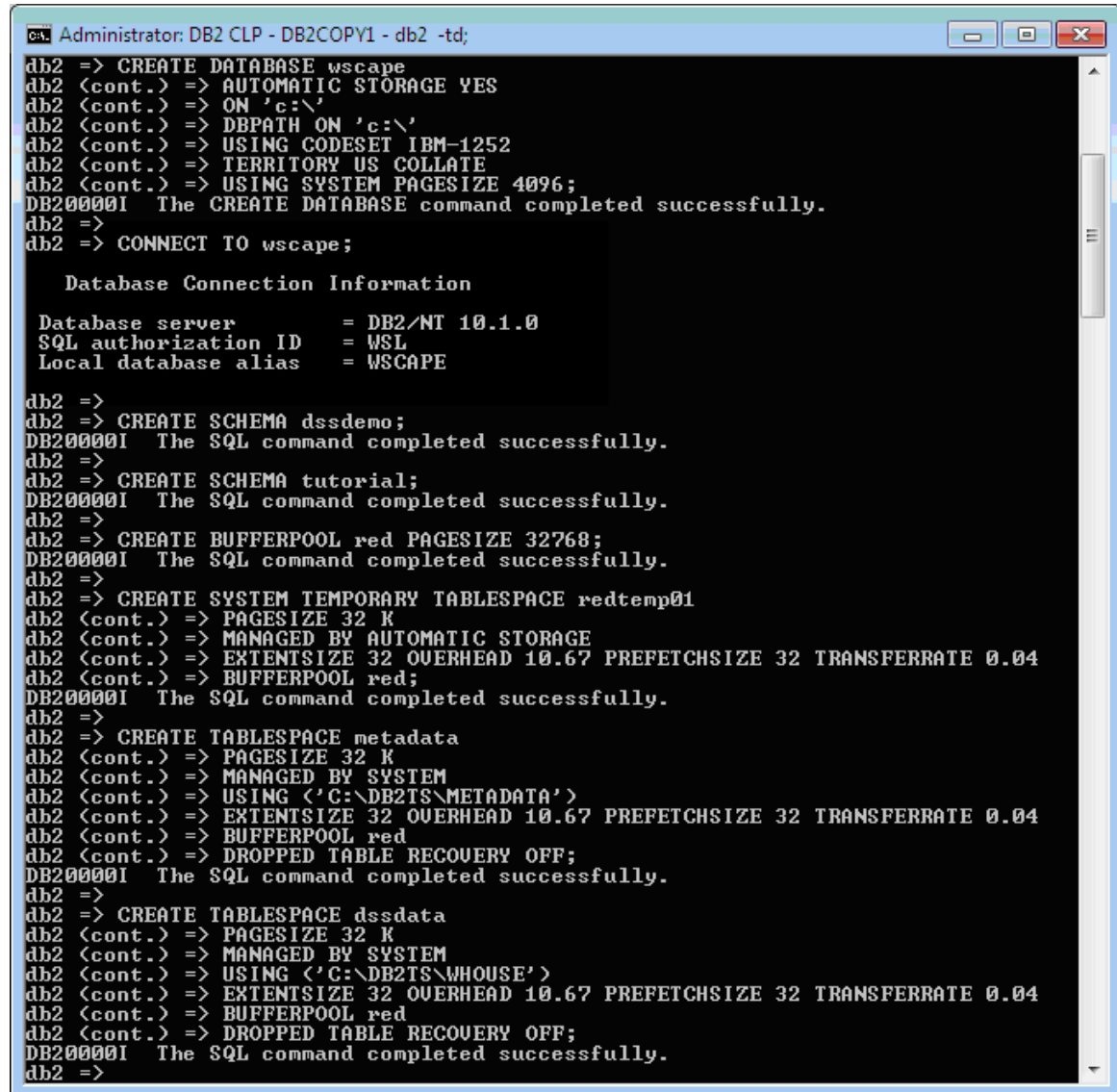
For general help, type: ?.
For command help, type: ? command, where command can be
the first few keywords of a database manager command. For example:
? CATALOG DATABASE for help on the CATALOG DATABASE command
? CATALOG           for help on all of the CATALOG commands.

To exit db2 interactive mode, type QUIT at the command prompt. Outside
interactive mode, all commands must be prefixed with 'db2'.
To list the current command option settings, type LIST COMMAND OPTIONS.

For more detailed help, refer to the Online Reference Manual.

db2 => _
```

Now paste in the script above:



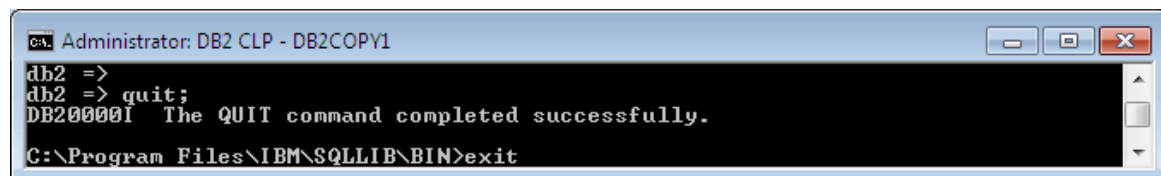
```
Administrator: DB2 CLP - DB2COPY1 - db2 -td;
db2 => CREATE DATABASE wscape
db2 <cont.> => AUTOMATIC STORAGE YES
db2 <cont.> => ON 'c:\'
db2 <cont.> => DBPATH ON 'c:\'
db2 <cont.> => USING CODESET IBM-1252
db2 <cont.> => TERRITORY US COLLATE
db2 <cont.> => USING SYSTEM PAGESIZE 4096;
DB20000I The CREATE DATABASE command completed successfully.
db2 =>
db2 => CONNECT TO wscape;

Database Connection Information

Database server          = DB2/NT 10.1.0
SQL authorization ID    = WSL
Local database alias    = WSCAPE

db2 =>
db2 => CREATE SCHEMA dssdemo;
DB20000I The SQL command completed successfully.
db2 =>
db2 => CREATE SCHEMA tutorial;
DB20000I The SQL command completed successfully.
db2 =>
db2 => CREATE BUFFERPOOL red PAGESIZE 32768;
DB20000I The SQL command completed successfully.
db2 =>
db2 => CREATE SYSTEM TEMPORARY TABLESPACE redtemp01
db2 <cont.> => PAGESIZE 32 K
db2 <cont.> => MANAGED BY AUTOMATIC STORAGE
db2 <cont.> => EXTENTSIZE 32 OVERHEAD 10.67 PREFETCHSIZE 32 TRANSFERRATE 0.04
db2 <cont.> => BUFFERPOOL red;
DB20000I The SQL command completed successfully.
db2 =>
db2 => CREATE TABLESPACE metadata
db2 <cont.> => PAGESIZE 32 K
db2 <cont.> => MANAGED BY SYSTEM
db2 <cont.> => USING ('C:\DB2TS\METADATA')
db2 <cont.> => EXTENTSIZE 32 OVERHEAD 10.67 PREFETCHSIZE 32 TRANSFERRATE 0.04
db2 <cont.> => BUFFERPOOL red
db2 <cont.> => DROPPED TABLE RECOVERY OFF;
DB20000I The SQL command completed successfully.
db2 =>
db2 => CREATE TABLESPACE dssdata
db2 <cont.> => PAGESIZE 32 K
db2 <cont.> => MANAGED BY SYSTEM
db2 <cont.> => USING ('C:\DB2TS\WHOUSE')
db2 <cont.> => EXTENTSIZE 32 OVERHEAD 10.67 PREFETCHSIZE 32 TRANSFERRATE 0.04
db2 <cont.> => BUFFERPOOL red
db2 <cont.> => DROPPED TABLE RECOVERY OFF;
DB20000I The SQL command completed successfully.
db2 =>
```

Once all the database objects have been created you may choose to **quit** the command processor and **exit** the *IBM DB2 Command Window*, as follows:



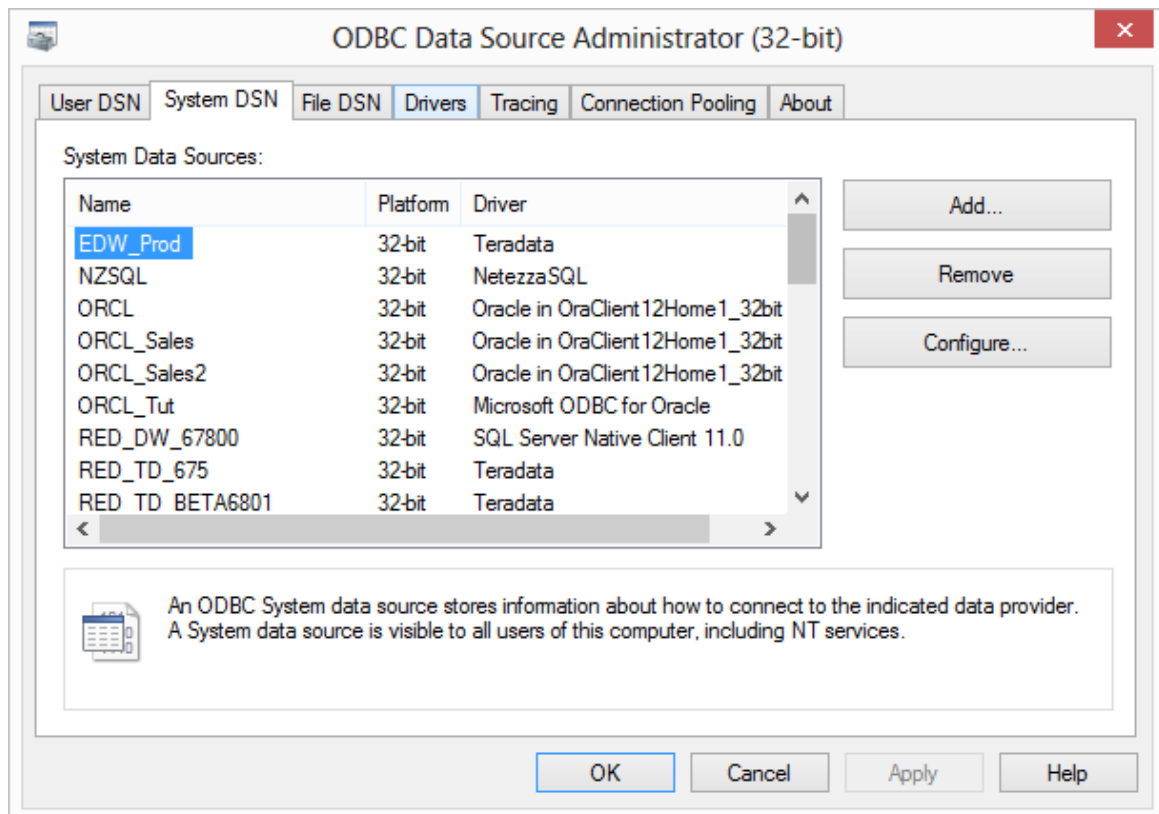
```
Administrator: DB2 CLP - DB2COPY1
db2 =>
db2 => quit;
DB20000I The QUIT command completed successfully.
C:\Program Files\IBM\SQLLIB\BIN>exit
```

Note: The create database statement cannot be run using SQL Admin or any other tool that connects to DB2 using ODBC. It should be run using the DB2 command window. The rest of the commands (from create schema onwards) can be run using SQL Admin, once the database has been created. Also, any commands run using the DB2 command window cannot include carriage returns or line feeds. The supplied sample script has each command on a single line.

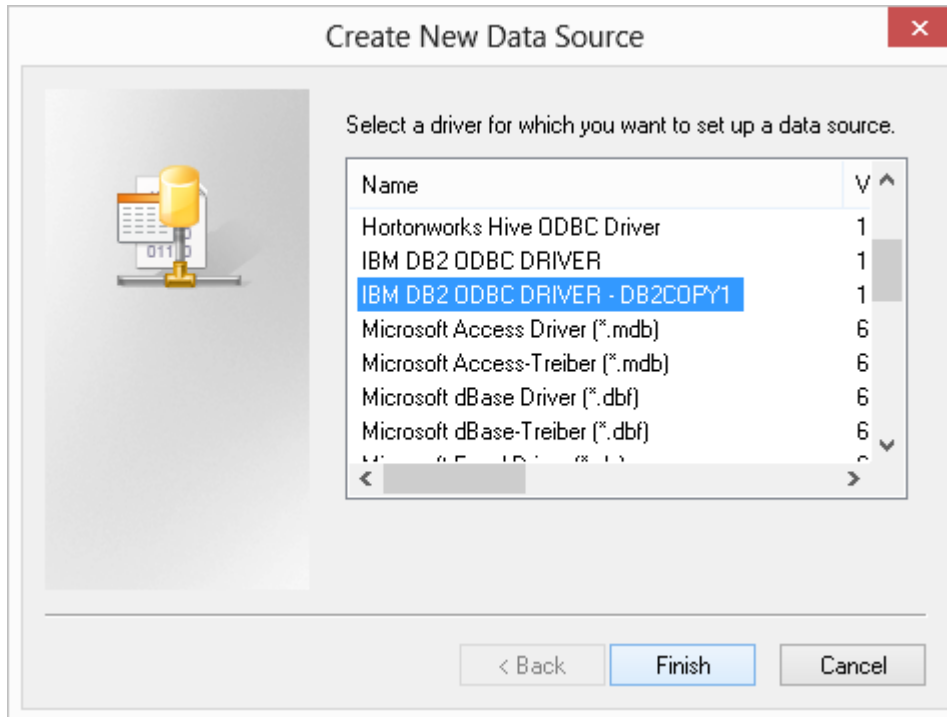
10.4 CONFIGURING A DB2 METADATA CONNECTION (DSN)

To begin adding an ODBC Source from WhereScape's Setup Administrator, see *Creating an ODBC Connection* (see "6. Creating an ODBC Connection" on page 21).

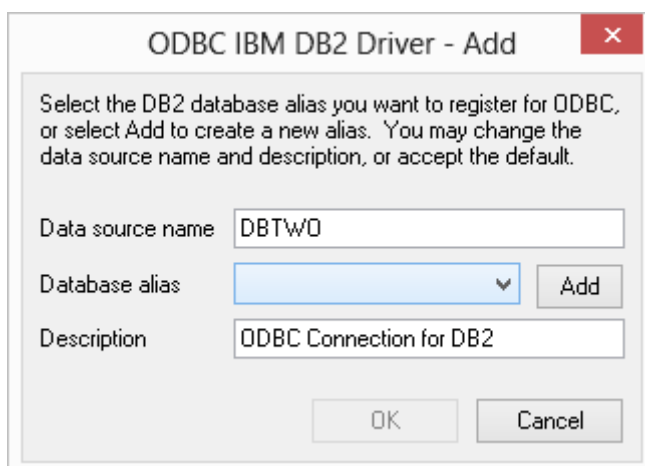
- 1 In ODBC Data Source Administrator, click on the **System DSN** tab to view the system wide ODBC entries.
 - Click on the 'Add...' button to add a new entry.



- 2 Select an **IBM DB2 ODBC DRIVER** driver from the driver list and click **Finish**.



- 3 A pop-up as shown below will appear.
 - Enter a name for the **Data source name**.
 - Choose the **Database alias** from the drop-down list.
 - Enter a **Description**.
 - Click **OK**.



TIP: If multiple PCs will access the data warehouse they must all use the same data source name as entered here.

10.5 CREATING A METADATA REPOSITORY FOR DB2

- 1 In WhereScape's Setup Administrator, click on the **Metadata Repository** tab in the left pane.
 - Then click on the **Create Metadata Repository** icon.

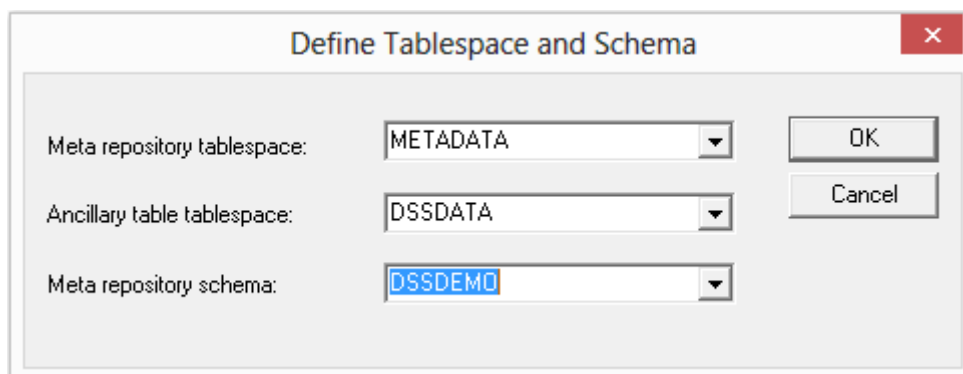


- 2 This option will create the WhereScape meta tables, sequences and procedures in the specified schema. It checks to ensure that the schema, database or user does not contain an existing metadata repository before starting the metadata install.
 - Click on Create Metadata Repository.
 - Enter the **ODBC Connect**, **User Name** and **Password** and click **OK**.

A dialog box titled 'Meta Repository Logon' with a close button (X) in the top right corner. The dialog contains the following text: 'Select the ODBC data source and Log on to the User where the NEW Data Warehouse is to be created.' Below this text are three input fields: 'Odbc Connect' with a dropdown menu showing 'WsfWarehouse', 'User Name' with a text box containing 'demo', and 'Password' with a text box containing 'xxxxxxxx'. To the right of the input fields are two buttons: 'OK' and 'Cancel'.

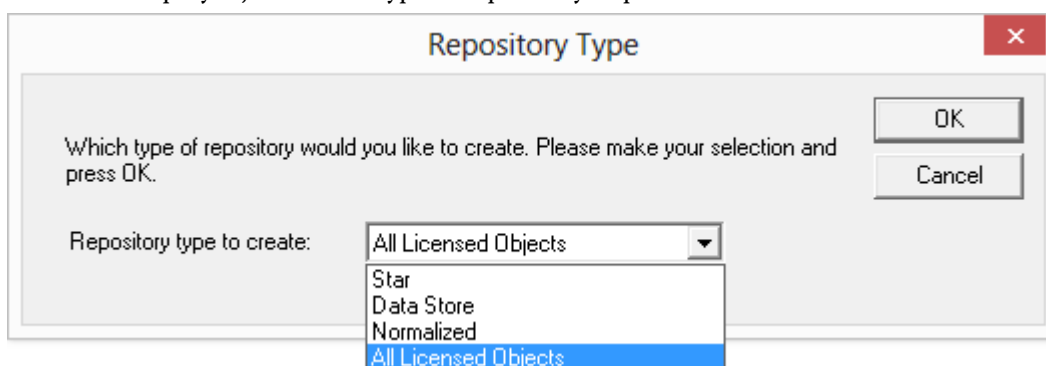
- 3 When creating a DB2 metadata repository, an additional screen is displayed requesting **Tablespace** and **Schema** information.
Select the following fields:

- Tablespace where the metadata repository will reside
- Tablespace where initial dimensions will be created
- Schema used for the metadata repository
- Click **OK**.



The dialog box titled "Define Tablespace and Schema" contains three dropdown menus and two buttons. The first dropdown menu, labeled "Meta repository tablespace:", is set to "METADATA". The second dropdown menu, labeled "Ancillary table tablespace:", is set to "DSSDATA". The third dropdown menu, labeled "Meta repository schema:", is set to "DSSDEMO". To the right of the dropdowns are "OK" and "Cancel" buttons.

- 4 Depending on the type of license you have, the following dialog may be displayed:
- If it is displayed, select the type of repository required and click **OK**.

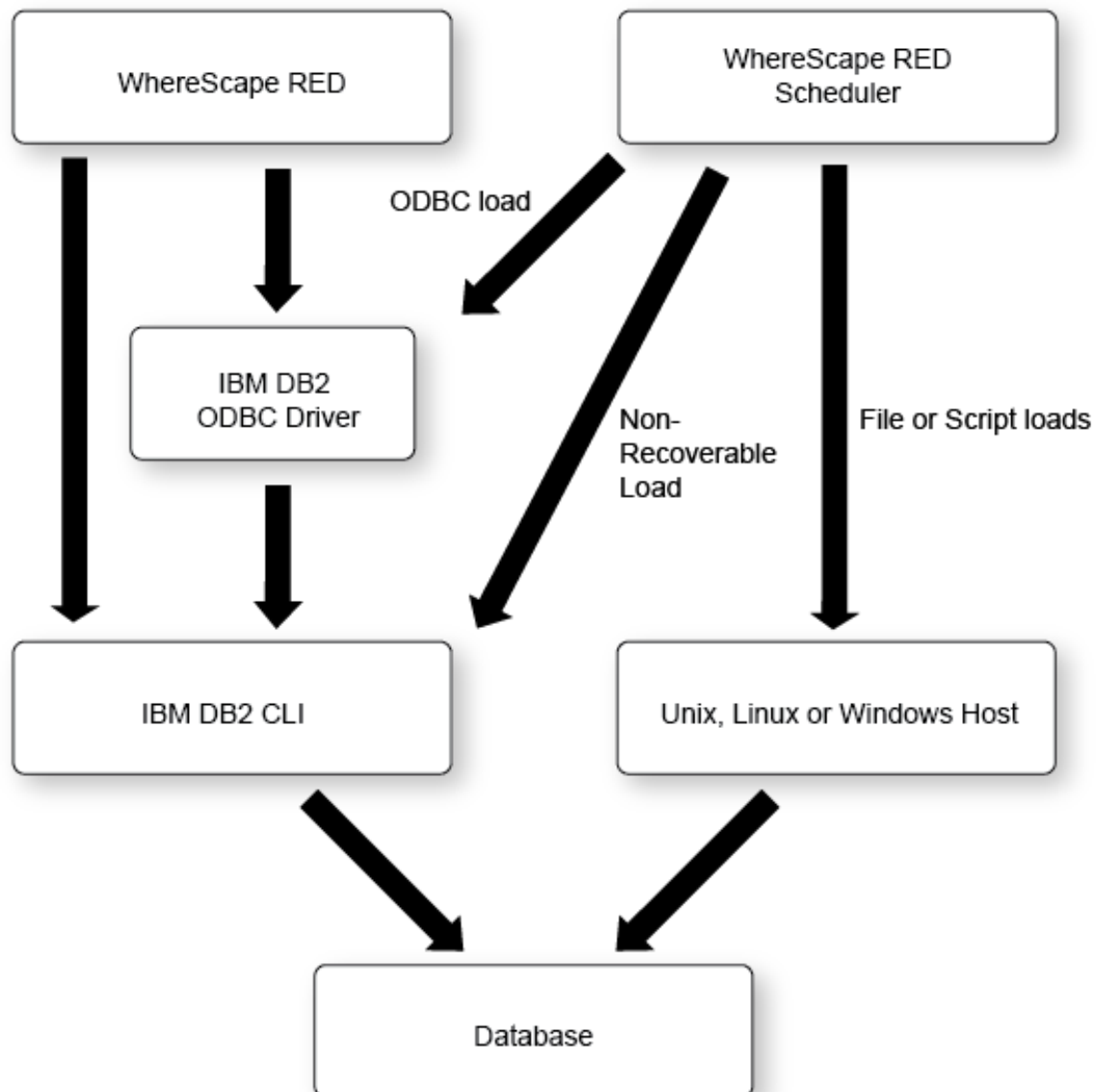


The dialog box titled "Repository Type" contains a text prompt and a dropdown menu. The text prompt reads: "Which type of repository would you like to create. Please make your selection and press OK." Below the prompt is a dropdown menu labeled "Repository type to create:" with a list of options: "All Licensed Objects", "Star", "Data Store", "Normalized", and "All Licensed Objects". The "All Licensed Objects" option at the bottom of the list is highlighted in blue. To the right of the dropdown menu are "OK" and "Cancel" buttons.

Note: This is just setting the default objects visible in RED. They can later be changed under **Tools/Options**.

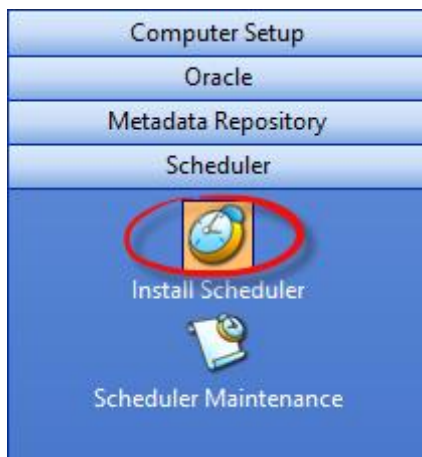
10.6 INSTALLING A WINDOWS SCHEDULER FOR DB2

This option installs and starts a copy of the WhereScape Windows scheduler. The scheduler runs as a system service and can also be maintained through the normal Windows services functions. The following diagram provides an overview of how WhereScape RED and the scheduler access a DB2 database.



The scheduler makes use of the ODBC layer to perform the routine tasks involved in scheduling and running jobs. WhereScape RED and the scheduler can also use CLI non-recoverable loads to load data into DB2. For file or script based loads, the scheduler actions the load on the host system utilizing the DB2 Load statement.

- 1 In WhereScape's Setup Administrator, click on the **Scheduler** tab in the left pane.
 - Then click on the **Install Scheduler** icon.



On selection of this menu option the following dialog box will appear:

Windows Scheduler Definition	
Unique Identifier (e.g. SID_user):	db2
Scheduler Name (e.g. WINnnnn):	WIN0001
Service Display Name:	WhereScape db2
Service logged on under User:	System Account
Work Directory:	C:\ProgramData\WhereScape\Work\db2\dssdemo\
Log File:	C:\ProgramData\WhereScape\Scheduler\WslSched_db2.log
Poll Interval (default 30) seconds:	30
Log Level (0=none, 9=debug)	2
Direct Path Load or ODBC	Odbc sql load
TNS Service (Oracle):	
ODBC Connection:	dbTWO
Database User id:	dssdemo
Database Password:	****
MetaData Schema (DB2):	
MetaData Database (Teradata):	dssdemo

Unique Identifier

Each scheduler requires a unique name (possibly the instance name and metadata schema e.g. DB2COPY1_dssdemo). This unique name is the name by which the service is referenced.

Scheduler Name

When scheduling a job with WhereScape RED it is possible to assign the job to a specific scheduler. Each scheduler should therefore have a unique name. The default name is WIN0001. Most users change this to be the scheduler machine's name followed by a number, e.g.: DWHP01, where DWHP is the name of the windows scheduler server.

Service Display Name

This field displays the name that the scheduler will appear under when viewed through the Windows services. It is a combination of the string 'WhereScape' and the unique identifier.

Service Logged on Under User

This is the name of the Windows local user account that the scheduler service will use when logging on. The default setting is **System Account**, also known as the **Local System** service account. To set the scheduler to use a Windows domain user account with a password, use the Service Account option from the right-click menu. If a domain user account is specified then the contents of this field (on the **Windows Scheduler Definition** dialog) are ignored.

Note: A domain user account will require the **log on as a service** privilege in order for it to work. This is assigned automatically when setting the log on account using the Services console in Windows.

Work Directory

The directory used to store some temporary files created by the scheduler.



TIP: Two schedulers should not use the same work directory

Log File

The full path where the scheduler service will write a log of the various actions undertaken. This file may get large if log level 9 is set.

Poll Interval

This defines the frequency of the scheduler polling the database, looking for work. The default is 30 seconds. If set to a very low interval such as 1 then the impact on the database may be noticeable.

Log Level

This defines the level of detail written to the log file. The following logging levels are available:

- 0 = will result in no information being written.
- 1 = fatal errors only
- 2 = major events
- 3 = major events and return stats to caller
- 9 = full debug



TIP: Setting Log Level 3 will return timing stats during ODBC based loads

Direct Path load or Odbc load

This should always be **ODBC sql load** for DB2 schedulers.

TNS Service

This is not set for DB2 schedulers.

ODBC Connection

The ODBC connection field requires the ODBC source name that will provide a connection to the data warehouse system. The scheduler uses ODBC to connect to the data warehouse.

Database User Id

The User Id the Scheduler will use when connecting to the data warehouse database.

Database Password

The password for the User the Scheduler will use when connecting to the data warehouse database.

Metadata Schema (DB2) / Metadata Database (Teradata)

The Schema in DB2 where the metadata repository is stored. This must be entered for DB2.

DB2 Grants

From DB2 9.7, additional grants are required to run the WhereScape RED Windows Scheduler as the system service account in Windows. See *Creating a Metadata Repository for IBM DB2* (see "10.5 Creating a Metadata Repository for DB2" on page 124) for more details.

Note: A windows scheduler must be run as either the system service account or as a user with administrator privileges in windows.

10.7 INSTALLING A UNIX SCHEDULER FOR DB2

The Unix scheduler must be manually installed and maintained.

The scheduler consists of a series of shell scripts that are controlled by the cron process. The bulk of the work undertaken by the WhereScape scheduler takes place in the WhereScape procedures within the database. To obviate the need to provide a port to each Unix platform this series of shell scripts are used. They should run with no modification on all Unix platforms and have been tested under Unix offerings from Sun, HP, IBM and Linux.

The Unix scheduler can process every type of load except for ODBC loads, Native ODBC loads and XML loads. It cannot process OLAP Cubes.

WARNING: For UNIX/Linux scheduler processing, the **Encrypt User** and **Password** options in RED's Repository Privacy Settings cannot be used. Encrypt options are only supported when using a Windows scheduler.

Installation

The installation of the Unix scheduler needs to be undertaken by someone familiar with the Unix environment and the running of database processes from Unix. The tasks that need to be undertaken are:

1. Unix User and Shell

Acquire a Unix user name that has access to cron and the database data warehouse environment. A reasonable quantity of disk space is also required if the automated backups of the metadata repository are used. The user should log on to Unix without passing through a menu system if possible. Log on to the user.

The WhereScape scheduler consists of a number of shell scripts. By default some of these shell scripts will run in the Bourne shell and some will run in the Korn shell. If Linux is being used then the Bash shell is used exclusively.

Note: During setup it is recommended that the Korn shell be used as the default shell for the user installing the scheduler (or Bash if running Linux).

2. Create directories

Create the following directories (if not already present) to be used by the WhereScape scheduler. If the directory names are not appropriate and others must be used then it will be necessary to alter the WhereScape supplied scripts.

- wsl (top level directory for WhereScape directories)
- wsl/bin (used to store the WhereScape scripts)
- wsl/sched (used for the scheduler job and log files)
- wsl/sched/job (used for the scheduler job files)
- wsl/sched/joblog (used for the scheduler job logs)
- wsl/sched/log (used for the scheduler logs)
- wsl/export (used for the automated exports of the metadata)
- wsl/expback (used to hold compressed copies of the metadata exports)
- wsl/log (used for export and archive logs)

3. Copy supplied Unix files

FTP the scheduler files shipped with WhereScape RED to the wsl/bin directory on the Unix system (use ASCII mode). All files should be copied over.

WhereScape RED ships with several different versions of the unix files for the scheduler. The appropriate version of the files should be copied to the server. The following lists the files shipped:

- ...\\WhereScape\\DB2\\Unix\\Version_nnn\\...
- ...\\WhereScape\\DB2\\Linux\\Version_nnn\\...

Note: There may be several different versions of the scheduler files for a given database and platform (Unix or Linux). These are identified by version numbers **nnn**. For example, there may be three different folders in ...\\WhereScape\\DB2\\Linux\\: Version_010, Version_410 and Version_560. The highest version number script less than or equal to the version of RED in use should always be used. In the table that follows, the **nnn** in file names refers to the version number of the script. Some scripts will have **010**, some will have **411** and other **560**, etc.

A definition of the files follows:

File	Purpose
dbenv (for DB2)	Contains the template of a script for setting environmental variables. This needs to reside in the home directory of the Unix user and must be tailored for the database user and environment. It may be renamed to reflect the actual environment. (e.g. dbenv_SID_user)
crontab	Contains the template of the required cron commands. This file needs to be modified to include the HOME directory of the Unix user, and the name of the environment file. It is then used to update the crontab tables.
ws_start_nnn.sh	Script to start the scheduler. This script is normally called by the ws_sched_check_nnn.sh script and is passed the database environment file name as a parameter.
ws_sched_check_nnn.sh	Script to check that the scheduler is running. This script is normally executed by cron every 20 minutes. If the scheduler is found to be absent it attempts to start a new iteration. By default the scheduler will terminate if its database is shut down. This script will therefore start the scheduler after a system start-up or a database start-up.
ws_sched_nnn.sh	The actual scheduler script. This script looks for new scheduler jobs every x seconds. The default is every 30 seconds. This script must be modified to change the default behavior. It calls the ws_job_wait_010.sh script to perform the actual checking and initiating of a job. It terminates when ws_job_wait_010.sh returns an error, and relies upon cron for a restart.
ws_sched_status_nnn.sh	This script is called by ws_sched_nnn.sh to update the scheduler status in the scheduler metadata. It's a wrapper of the metadata database procedure ws_sched_status (i.e. it calls the database procedure ws_sched_status and returns it's results to the calling script).

ws_job_wait_nnn.sh	This script performs the bulk of the work. It is called by ws_sched_nnn.sh on a regular basis. It checks for any job waiting to run. If it locates a job that is ready to run it will create and start a script in the \$HOME/wsl/sched/job directory for each thread of that job. These job threads in turn perform the actual processing and log their results to \$HOME/wsl/sched/joblog. Under normal operation it is not necessary to check or inspect these files.
ws_job_check_nnn.sh	This script is called to check the status of running jobs. It is called by ws_sched_nnn.sh on a regular basis (the interval is defined in ws_sched_nnn.sh). It checks that any job running in the scheduler is actually running in the Unix environment. If a job is not found under Unix then it is aborted in the scheduler. This scenario can occur as the result of a database or machine crash. If a job abort is required the results are logged to \$HOME/wsl/sched/log/job_check_'env file name'.log. Under normal operation it is not necessary to check or inspect this file.
meta_backup_nnn.sh	This script is normally called by the cron process once or maybe more times a day to perform an automated export backup of the metadata tables. It creates a number of command files and a database export in the \$HOME/wsl/export directory named with the date/time of the export.
meta_archive_nnn.sh	This script is normally called by cron once a day to compress and tar the files in the \$HOME/wsl/export directory and relocate them to the \$HOME/wsl/expback directory. This action is performed purely to save on disk space usage.
meta_restore_nnn.sh	This script can be executed to restore from a meta repository backup. It will overwrite any existing metadata. It can be used to restore to a schema/database/user other than the one from which the backup was taken. It does not however restore the WhereScape procedures. If a full recovery or a new user create is required then the WhereScape metadata install would normally be performed before actioning this restore. If the restore is from an export that has been moved to the expback directory you will need to uncompress the export and move it and associated files back to the export directory.

cleanup_nnn.sh	This script is normally executed by cron and cleans up files created by the WhereScape scheduler. It deletes old versions of the jobs, job logs, and backups. It will also delete any temporary scripts created in the running of host scripts or file loads from within the scheduler. The location of these temporary scripts is defined within the WhereScape RED Unix Connection object. The default value in the cleanup script is /tmp and this must be altered if a different temporary directory is used for a Connection.
----------------	--

4. Set permissions

Change the permissions on the copied scripts so that they are executable. (e.g. `chmod 750 $HOME/wsl/bin/*.sh`)

5. Create an environment file

Using the environment file (dbenv) as a template and adhering to the variable names within, create an environment file for the database environment and user under which the scheduler is to run. This file should be located in the \$HOME directory. It may be called anything but a meaningful name may be `dbenv_SID_user` for a given DB2 scheduler. Check the users .profile and database environment file to see the normal settings.

Make the following replacements in the environment file:

- Replace all occurrences of `_HOME_` with the full pathname of your home directory
- Replace `_USER_` with the data warehouse database user name
- Replace `_PASSWORD_` with the data warehouse database user password.
- Replace all occurrences of `_DBID_` with the database id of the data warehouse database
- Replace all occurrences of `_SCHEMA_` with the schema where the metadata resides

6. Setup crontab

Ascertain if there are any existing crontab entries, and if so save them to a file for re-inclusion later. (e.g. `crontab -l` `crontab -l >/tmp/my_crontab`)

Edit the crontab file and perform the following changes. Also check all entries and make sure you are happy with the times and actions:

- Change all occurrences of `_HOME_` with the full pathname of the Unix users home directory.
- Change all occurrences of the environment file name (oraenv, dbenv or tdenv) to the name of the file created in step (4).
- Include any existing crontab entries as recorded above.

7. Initialize crontab entries

Load the crontab file into cron (e.g. `crontab crontab`).

Perform a crontab list to ensure it all looks OK (e.g. `crontab -l`)

8. Start Scheduler

Manually start the scheduler by issuing the `ws_sched_check_nnn.sh` script with the environment file name from step (4) as the parameter (e.g. `ws_sched_check_010.sh oraenv`). If the scheduler is not running it will be started. Execute the script a few more times to ensure that only one copy of the scheduler is started. Check that the scheduler is running by executing a `'ps -ef'` statement and checking that the `ws_sched_560.sh` process is present.

9. Test the Scheduler

Run a job from WhereScape RED and ensure that it works. In the event of the job not starting or going into and remaining in a pending state then the following investigations should be undertaken:

- Ensure all WhereScape procedures are compiled. Use the Validate metadata option in the Setup Administrator to confirm that all procedures are valid.
- Review the job logs in `$HOME/wsl/sched/joblog`.
- Review the logs in `$HOME/wsl/sched/log`.
- Isolate the name of the job and locate the relevant files for this job in `$HOME/wsl/sched/job` and `$HOME/wsl/sched/joblog` if they exist. Mail these files together with the files in `$HOME/wsl/sched/log` to `support@wherescape.com`. It may be necessary to remove password references from the file(s) in `$HOME/wsl/sched/job`.

NOTE: It is possible to stop a UNIX/Linux Scheduler from within WhereScape RED. See the **WhereScape RED User Guide** for details.

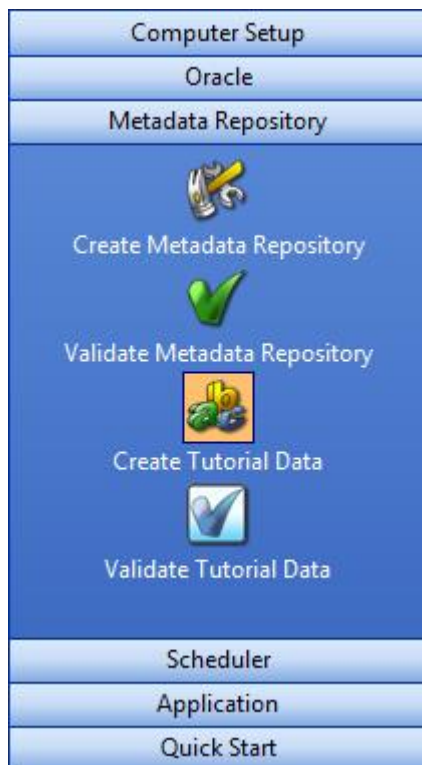
CHAPTER 11

11. TUTORIAL DATA

The tutorial data component of WhereScape's Setup Administrator attempts to validate, upgrade or perform an initial install of the WhereScape tutorial tables.

The tutorial tables are only required if the tutorial is to be undertaken. They can be ignored if desired.

- 1 Click on the **Metadata Repository** tab in the left pane, then click on either:
 - The **Create Tutorial Data** (see "**11.1 Creating Tutorial Data**" on page 137) option to create (or re-create) the set of tables in the specified schema (user).
 - The **Validate Tutorial Data** (see "**11.2 Validating Tutorial Data**" on page 138) option to check for the existence of WhereScape RED tutorial tables.



IN THIS CHAPTER

11.1 Creating Tutorial Data	137
11.2 Validating Tutorial Data.....	138

11.1 CREATING TUTORIAL DATA

- 1 In WhereScape's Setup Administrator, click on the **Metadata Repository** tab in the left pane.
 - Then click on the **Create Tutorial Data** icon.



This option will create, or re-create, the WhereScape tutorial tables within the schema (user) or database provided. The normal practice is to locate these tables in a different schema from the data warehouse user for Oracle and in a different database for SQL Server. The data warehouse user will need select access to these tables to perform the tutorial. The tutorial documentation assumes these tutorial tables will be loaded under the user 'wtutorial' for Oracle or in the Tutorial database for SQL Server. Another user or database may be used if desired. The user chosen must have sufficient privileges to allow the creation of the tables. When this option is chosen an ODBC connection dialog will appear, and if a valid user is entered the tables will be created and populated under the specified user.

Please ensure that no tables that match those shown in the Validate tutorial tables section exist under the specified user. If existing tables are found, a prompt will be issued, and if proceed is chosen then all data will be deleted from the tables and the tutorial data loaded.

Teradata

When creating tutorial tables for Teradata, an additional dialog is displayed for the user or database name that the tutorial data will be created in.

- Enter the name of the user or database in which the tutorial data will be created and click **OK**.

11.2 VALIDATING TUTORIAL DATA

- 1 In WhereScape's Setup Administrator, click on the **Metadata Repository** tab in the left pane.
 - Then click on the **Validate Tutorial Data** icon.



This option validates that the tutorial tables exist in the given schema. The tables are based around a simple sales model and are as follows:

Table	Contents
product	A list of products including a product code (sku), description, group and subgroup
prod_group	A description of each product group
prod_subgroup	A description of each product sub group
prod_line	A description of each product line
code_table	A table containing the codes and descriptions for product groups, sub groups and lines. A code type of 'pg', 'sg' and 'pl' is used to differentiate the different codes.
customer	A list of customers containing names and addresses.
order_line	The lines on a sale. Includes the specific product code, quantities and values
order_header	The header for a sale. Includes the customer code, order data and ship date
kpi	Used to generate a KPI dimension

budget	Budget data for sales of products to customers
forecast	Demand forecast data for products and customers

CHAPTER 12

12. APPLICATIONS

The application component of WhereScape's Setup Administrator attempts to list, upgrade or perform an initial install of a third-party supplied data warehouse application.

- 1 Click on the **Application** tab in the left pane; then click:
 - **Application Load** (see "12.1 Loading an Application" on page 142) to load the metadata, tables procedures, scripts etc. provided as part of the application.
 - **Change Application Directory** (see "12.3 Changing the Application Directory" on page 155) to change the directory of applications to be installed.
 - **List applications in Repository** (see "12.4 Listing Applications" on page 157) to list all applications that have been applied to the chosen metadata repository.

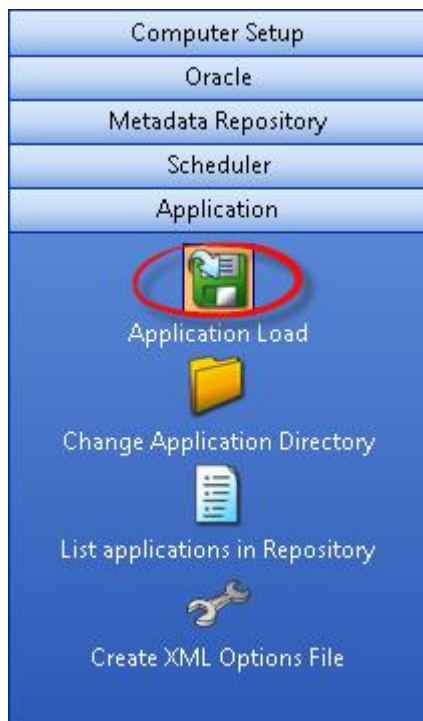


IN THIS CHAPTER

12.1 Loading an Application	142
12.2 Creating and Loading Applications from the Command Line	153
12.3 Changing the Application Directory	155
12.4 Listing Applications	157
12.5 Testing Applications	158
12.6 XML Files for Application Loads	159

12.1 LOADING AN APPLICATION

- 1 In WhereScape's Setup Administrator, click on the **Application** tab in the left pane, then click on the **Application Load** icon.



Loading an application inserts various objects into the chosen metadata repository. An application is best defined as a set of objects that are shipped to allow inclusion of those objects into a remote repository.

NOTE: An application can only be loaded into a metadata repository running on the same database type as that of the application creator. (e.g. An Oracle application cannot be loaded into a SQL Server metadata repository).

During the load of an application a number of cross environment mappings need to be resolved. These are:

Connections

Connections are normally unique for each metadata repository. They provide the path to the source data and this nearly always varies. Even though an application can ship a set of connections, these connections will nearly always need modification. Alternatively a connection shipped with an application may map to an existing connection, and this can be done as part of the load process. If you are unsure whether or not a connection is required, it can be added and later modified.

Tablespaces (Filegroups in SQL Server and Databases in Teradata)

If an application ships table or index objects then these objects will typically have a tablespace (filegroup in SQL Server and database in Teradata) associated with them. During the load it will be necessary to map these tablespaces (filegroups in SQL Server and databases in Teradata) to those available for the metadata repository.

Object Changes

The application may ship objects that already exist in the metadata. In such a case the process during the load is to create a version of the existing object and then replace it with the new object. At the start of the load a check is made to see if any objects will be replaced. If the application is a patch or an upgrade of an existing application version then this will be normal. The load can be stopped at this point to allow for further investigation.

Table Changes

If the load replaces table objects, it may in turn need to alter the physical table in the database to match the new definition. Again, if this is a patch or an application upgrade, this may be desirable. If not, the proposed actions should be reviewed before proceeding.

Procedure Changes

New procedures, scripts, or modifications may be loaded as part of the application load. In the case of procedures these procedures will need to be recompiled. The load can perform this action, but any failures or issues will need to be resolved after the load has completed.

The following section covers the actual application load process.

NOTE1: It is always a good idea to backup the WhereScape RED metadata before running an application load. See the section *Backing Up the Metadata before Loading Applications* for details.

NOTE2: Some database operations, such as converting existing non-partitioned tables to partitioned tables, cannot be done using a deployment application. In these cases some manual intervention may be required to update the target databases to match the new metadata.

12.1.1 BACKING UP THE METADATA BEFORE LOADING APPLICATIONS

It is always a good idea to backup the WhereScape RED metadata before running an application load.

A metadata backup can be used to restore the metadata to the state before the application was loaded.

There are three options for backing up the metadata:

- 1 In RED, use the **Backup/Unload the metadata to disk** option to unload the metadata to a WhereScape Proprietary backup. See the section **Unloading Metadata** in the WhereScape RED User Guide for more information.
- 2 In RED, use the **Backup/Export the metadata** option to backup the metadata using the data warehouse database's backup utility. See the section **Backup using DB Routines** in the WhereScape RED User Guide for more information.
- 3 In RED, use the **Tools/Build application tables** option to build an application of the objects being replaced. This is achieved by selecting the new application files you're about to load in the **Previous application** box of the Define and build an application distribution dialog. See the section **Application Creation** in the WhereScape RED User Guide for more information.

Note: If the table changes in the application are being applied to the data warehouse database, then it is also a good idea to backup the data warehouse tables being changed.

12.1.2 APPLICATION LOAD PROCESS

An application consists of five files that are loaded from a directory. The default directory is the WhereScape program directory. Another directory may be specified by clicking on the **Change Application Directory** button. The last selected application directory is the default directory for the next application load.

Once the directory has been selected, the application load is started by clicking on the **Application Load** button. This displays the available applications in the right pane.

NOTE: If the **Change Application Directory** button was used, then the list of available applications will already be displayed and the **Application Load** button doesn't need to be clicked.

If no applications are present, the list in the right pane will be empty and no application load will be possible. The tutorial applications are nearly always present. An example application list follows:

Available Applications					
App Id 'Right mouse to edit'	Version	Description	Metadata Version	Directory	Comments
SQLT	tutorial_1	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 1 - Basic star schema fact table
SQLT	tutorial_2	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 2 - Rollup fact tables, ASCII file loads an...
SQLT	tutorial_3	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 3 - Scheduling and dependencies ;
SQLT	tutorial_4	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 4 - Complex dimensions and hierarchies
SQLT	tutorial_5	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 5 - Analysis Services Cubes
SQLT	tutorial_6	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 6 - Kpi fact tables
TERT	tutorial_1	WhereScape Tutorials ...	5.5	C:\Program ...	Tutorial 1 - Basic normalized model
WSLT	tutorial_1	WhereScape Tutorials	5.5	C:\Program ...	Tutorial 1 - basic star schema fact table
WSLT	tutorial_2	WhereScape Tutorials	5.5	C:\Program ...	Tutorial 2 - Rollup fact tables, ASCII file loads an...
WSLT	tutorial_3	WhereScape Tutorials	5.5	C:\Program ...	Tutorial 3 - Scheduling and dependencies
WSLT	tutorial_4	WhereScape Tutorials	5.5	C:\Program ...	Tutorial 4 - Complex dimensions and hierarchies
WSLT	tutorial_5	WhereScape Tutorials	5.5	C:\Program ...	Tutorial 5 - Analysis Services Cubes
WSLT	tutorial_6	WhereScape Tutorials	5.5	C:\Program ...	Tutorial 6 - KPI fact table

- To load an application, right-click on the required Application and select **Install Application** from the pop-up menu.
- Once this menu option is chosen, enter the ODBC data source and logon to the meta repository and click **OK**.

1: Teradata Applications

NOTE1: When loading an application into Teradata, the **TDPID** will be tested. If the TDPID test fails at this stage, a dialog will appear asking you to enter the correct TDPID.

Click **Test and Use** after entering the TDPID and relevant credentials. If you skip the test and provide an invalid TDPID, native loads will not work in WhereScape RED.

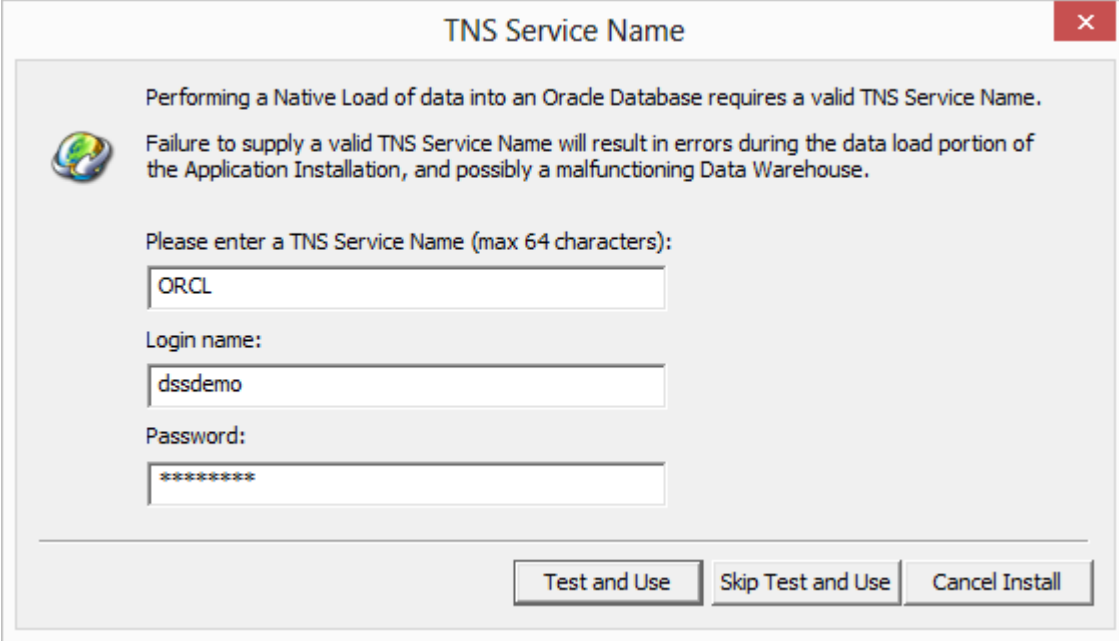
Teradata native loads use bteq which connects natively to the database using its own internal protocols without using the ODBC driver. Therefore ODBC settings are not relevant in this process and a valid Teradata **TDPID** is required to load application procedure and scripts successfully using native load.

NOTE2: After the TDPID check, an additional dialog is displayed for selecting the metadata repository database. Select the database and click **OK**.

2. Oracle applications

When loading an application into Oracle, the **TNS Name** will be tested. If the TNS Name test fails at this stage, a dialog will appear asking you to enter the correct TNS Name.

Click **Test and Use** after entering the TNS Name. If you skip the test and provide an invalid TNS Name, native loads will not work in WhereScape RED.



TNS Service Name

Performing a Native Load of data into an Oracle Database requires a valid TNS Service Name.

Failure to supply a valid TNS Service Name will result in errors during the data load portion of the Application Installation, and possibly a malfunctioning Data Warehouse.

Please enter a TNS Service Name (max 64 characters):

ORCL

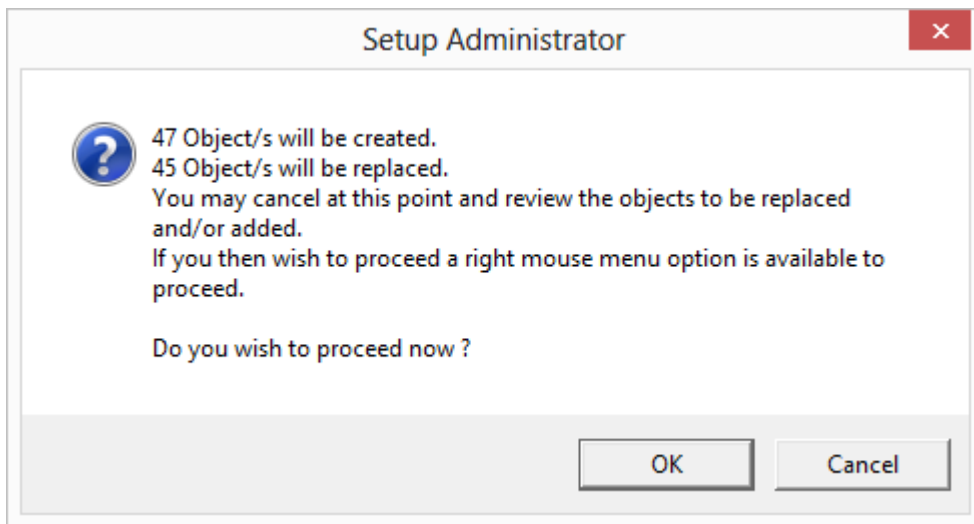
Login name:

dssdemo

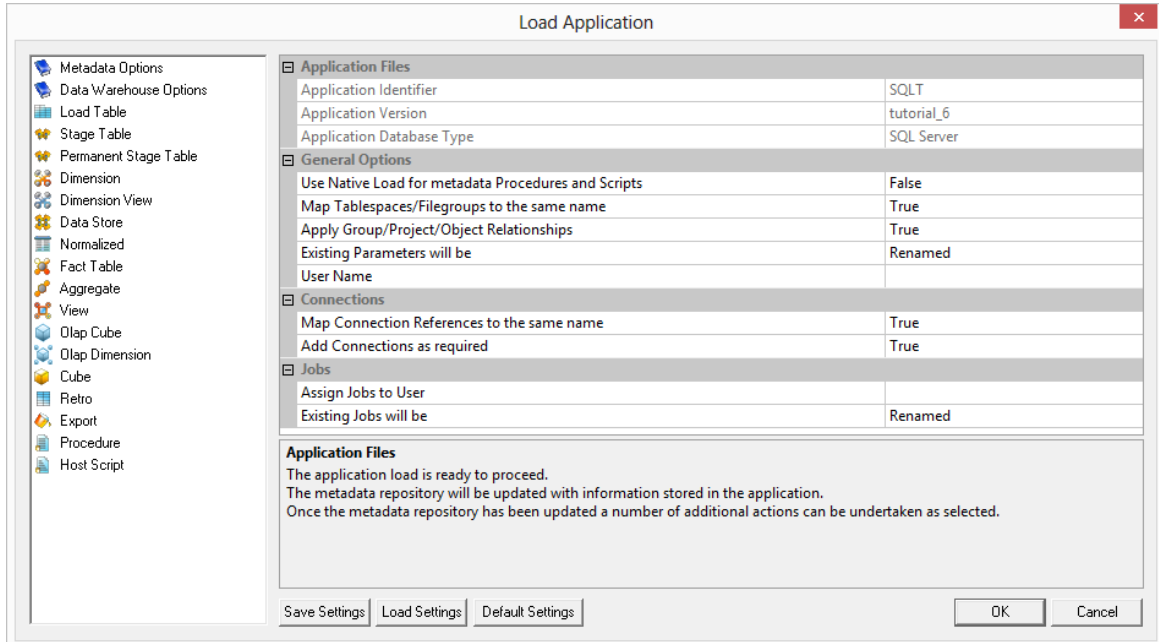
Password:

Test and Use Skip Test and Use Cancel Install

A list of all the objects in the application is displayed in the right pane. Each object will be identified as either new to the metadata, or as replacing an existing object. For example:



A dialog will appear after the list has been displayed advising of what is to occur. You may cancel the application load at this point and review the objects to be replaced. The load can then be continued by using the right-click menu option when positioned over any object. Click **OK** to proceed with the application load. A new dialog will appear providing options for the application load:



If you click **OK**, the objects will be inserted into the metadata.

METADATA OPTIONS

[-] Application Files	
Application Identifier	SQLT
Application Version	tutorial_1
Application Database Type	SQL Server
[-] General Options	
Use Native Load for Metadata Procedures and Scripts	False
Map Tablespaces/Filegroups to the same name	True
Apply Group/Project/Object Relationships	True
Existing Parameters will be	Renamed
[-] Connections	
Map Connection References to the same name	True
Add Connections as required	True
[-] Jobs	
Assign Jobs to User	ETL Quickstart
Existing Jobs will be	Renamed

Use Native Load for Metadata Procedures and Scripts

This option causes the metadata for procedures and scripts to be loaded into the target metadata repository using the target repository's load utility instead of ODBC. This can have significant performance advantages when loading large applications, when loading applications into a slow target database or over a slow network. The loaders used are: **SQL*Loader** for Oracle, **bcp** for SQL Server and **BTEQ** for Teradata data warehouses.

Map Tablespace/Filegroups to the same name

Every table in Oracle and DB2 is located in a **tablespace**. SQL Server tables may be in **filegroups**.

If this option is set, existing tablespaces and filegroups in the target data warehouse database will automatically be matched to their same ones on the source repository. If the tablespace/filegroup does not exist in the the target data warehouse database, a Tablespace selection dialog will pop-up so a tablespace/filegroup can be selected. This dialog appears once for each tablespace/filegroup that cannot be matched.

Note: This option is not available for Teradata

Apply Group/Project/Object Relationships

If this option is marked as true, the project group relationships are updated in the target data warehouse database.

Existing Parameters will be

The options are

- **Updated/Overwritten**
- **Retained**
- **Renamed**

Map Connection References to the same name

Use this option to ensure that any objects with linked connections map to an existing connection using the same name as the connection that linked to in the metadata repository where the application was built. If a connection does not exist in the target metadata repository and the option *Add new connections as required by the application* is not selected, then an alternative connection must be selected using the **Connection Build** dialog. This dialog appears once for each tablespace/filegroup that cannot be matched.

Add Connections as required

If this option is set, all connection objects included in the application that do not exist in the target metadata repository and that are used by objects in the application will be added to the target metadata repository.

Assign Jobs to User

All jobs in the WhereScape scheduler belong to a user. If this option is marked as true, the user name specified in the **User name for any jobs added by the application** drop-down is set as the user name of all jobs loaded by the application. If the specified user name does not exist, it is then created.

Existing Jobs will be

The options are

- Updated/Overwritten
- Retained
- Renamed

DATA WAREHOUSE OPTIONS

Data Warehouse Options	
Apply Metadata Changes to Data Warehouse Objects	True
Log Metadata DML and Data Warehouse DDL to a File	True
DDL File	C:\Users\User\Documents\WhereScape\app_SQLT_tutorial_1_ddl.sql
DDL File Statement Terminator	;
Terminate lines with carriage return as well as new line	True
Compile New and Changed Procedures	True

Apply Metadata Changes to Data Warehouse Objects

If this option is marked as True, the changes contained in the application are applied to the metadata according to the options selected for each object type.

Log Metadata DML and Data Warehouse DDL to a File

If this option is marked as True, the changes contained in the application are written to a file.

- **DDL File** - Option to enter the file name/path for the DDL file.
- **DDL File Statement Terminator** - Option to specify the statement terminator.
- **Terminate lines with carriage return as well as new line** - Option to terminate lines with a carriage return.

Compile New and Changed Procedures

If this option is marked as true then all procedure objects loaded by the application load are compiled in the target data warehouse database.

NOTE: Depending on the type of database, invalid procedures may not compile.

Data Warehouse Table/View Options

Load Table	
New Load Tables will be	Created
New Load Tables will be Versioned	False
New Load Table Indexes will be	Created
Existing Load Tables will be	Recreated
Existing Load Tables will be Versioned	True
Existing Load Table Indexes will be	Recreated

For each table and view object loaded by the application load, it is possible to specify whether or not the metadata changes should be applied to the data warehouse table or view; and to the indexes on these objects. The change action can be specified by object type. New objects can have a different action to existing objects of the same type. Possible actions are:

- **Created** - available for new tables and indexes of any object type
- **Recreated** - available for existing tables and indexes of any object type
- **Altered** - available for existing tables and indexes of most object types (not available for load tables or views)
- **Nothing** - available for all new and existing object types and indexes.

This dialog also enables you to set whether or not new or existing objects are to be versioned. Set to **True** to use versioning, else set to **False**.

Stage Table	
New Stage Tables will be	Created
New Stage Tables will be Versioned	False
New Stage Table Indexes will be	Created
Existing Stage Tables will be	Recreated
Existing Stage Tables will be Versioned	True
Existing Stage Table Indexes will be	Recreated
Alter allows columns to be dropped	False

Another option enables you to set whether or not columns are allowed to be dropped by object type, for tables being altered. Set to **True** to allow columns that are not in the metadata, to be dropped; else set to **False**. This option applies to Stage, Permanent Stage, Dimension/Model, ODS, Normal, Fact and Aggregate Tables.

Object action settings may be saved and reloaded later for another application using the **Save Settings** and **Load Settings** buttons. Supplied default settings may also be loaded by clicking on the **Default Settings** button.

THE CONNECTION BUILD DIALOG

If connections are encountered and are not automatically mapped (see connection options above), a dialog is issued to ascertain whether the connection can be mapped to an existing connection or whether the connection needs to be added. The application vendor will need to advise of the action required. The dialog is as follows:

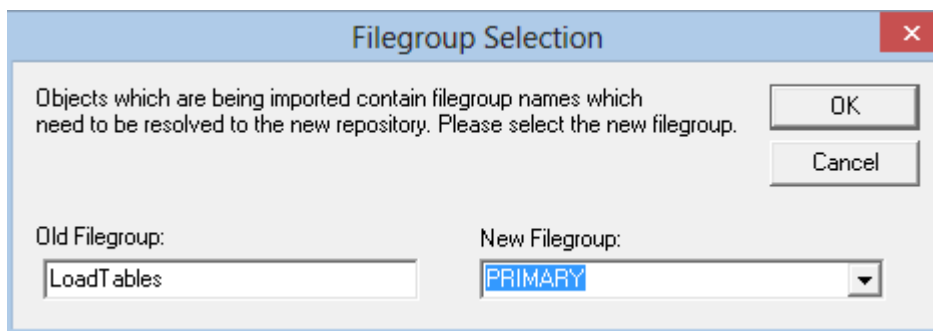
Either a connection is selected from the **Existing connections** drop-down list or the **Add New** button is clicked. If a new connection is added then it will need to be modified to match the database and user names of the new environment.



TIP: Added connections will need to be modified by selecting their **Properties** in WhereScape RED.

THE FILEGROUP /TABLESPACE SELECTION DIALOG

Filegroups/Tablespaces may need to be manually remapped to the corresponding Filegroup/Tablespace available to the metadata repository. A dialog box as follows will appear for each Filegroup/Tablespace encountered that is not automatically mapped (See Filegroup/Tablespace options above).



The drop-down list box **New Filegroup** provides a list of all filegroups visible to the repository. A dialog box will appear when the application load has completed. The full results should be reviewed by scrolling back, or you can right-click and click **Excel It** to send the results to Excel.

Computer Setup	Application Installation	
Oracle	Object	Status
Metadata Repository	fact_forecast_idx_3	Index created
Scheduler	fact_forecast_idx_A	Index created
Application	fact_sales_analysis	Table created
Application Load	fact_sales_analysis_idx_1	Index created
Change Application Directory	fact_sales_analysis_idx_2	Index created
List applications in Repository	fact_sales_analysis_idx_3	Index created
	fact_sales_detail	The table fact_sales_detail already exists, and contains 21 rows of data
	agg_sa_customer	Table created
	agg_sa_customer_idx_1	Index created
	agg_sa_customer_idx_2	Index created
	agg_sa_product	Table created
	agg_sa_product_idx_1	Index created
	agg_sa_product_idx_2	Index created
	dim_order_date	The view dim_order_date already exists, and contains 0 rows of data
	dim_order_date	View (re)created
	dim_ship_date	The view dim_ship_date already exists, and contains 0 rows of data
	dim_ship_date	View (re)created
	get_dim_customer_key	Compiled
	update_dim_customer	Compiled
	get_dim_date_key	Compiled
	update_dim_date	Compiled
	get_dim_order_date_key	Compiled
	get_dim_product_key	Compiled
	update_dim_product	Compiled

12.2 CREATING AND LOADING APPLICATIONS FROM THE COMMAND LINE

It is possible to create and load applications from the command line by running a bat file. The WhereScape RED Application directory contains an example bat file

WSL_Application_Create_Restore_Point_and_Load.bat for creating and loading.

If you right-click on this file and select **Edit** you will see the steps outlined as well as the details on the options available.

The first step **creates** a restore point application (R) based on objects about to be loaded. This process calls the command line functionality of RED and creates the application file.

The second step **loads** an application (A) into a test WhereScape RED repository. It uses an xml file to specify various options, calls the command line automation functionality of Setup Administrator and loads the application (A).

The WhereScape RED Application directory contains example xml files:

- WSL_Application_Load_SQL.xml for SQL Server
- WSL_Application_Load_DB2.xml for DB2
- WSL_Application_Load_ORA.xml for Oracle

The tags in the xml file need to be edited as the login/connection details etc need setting.

The Batch Application Create options and the Batch Application Load options are listed at the end of the bat file. The values for these variables will also need to be customized before running this file.

The process typically involves the following steps:

- 1 Create an application (A) in RED containing your data warehouse changes.
- 2 RUN WSL_Application_Create_Restore_Point_and_Load.bat from the command line; which creates a restore point application (R) and applies application (A) to a test WhereScape RED repository.
- 3 If the changes are incorrect, they can be undone by loading the restore point application (R).

Batch Application CREATE options

/BA selects batch application create.

/U username.

/P password.

/C DSN name.

/M metaschema/database for DB2/Teradata logon.

/D directory to save application files to.

/I new application files identifier.

/V new application files version.

/AP Project name - all objects in a Project; and associated jobs.

/AG Group name - all objects in a Group; and associated jobs.

/ALL All objects - all objects, jobs and parameters.

/RC Remove connections - drop all connections from the application.

/RJ Remove jobs - drop all jobs from the application.

/RP Remove parameters - drop all parameters from the application.

/AF absolute application id file name which restore point is being created for.

If using a trusted connection, the user and password are not necessary.

Batch Application LOAD options

/AL select application load

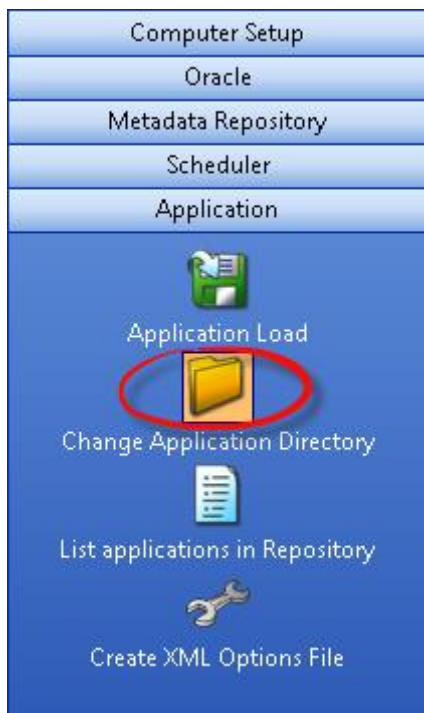
/AF absolute application id file name

/LF absolute log file name

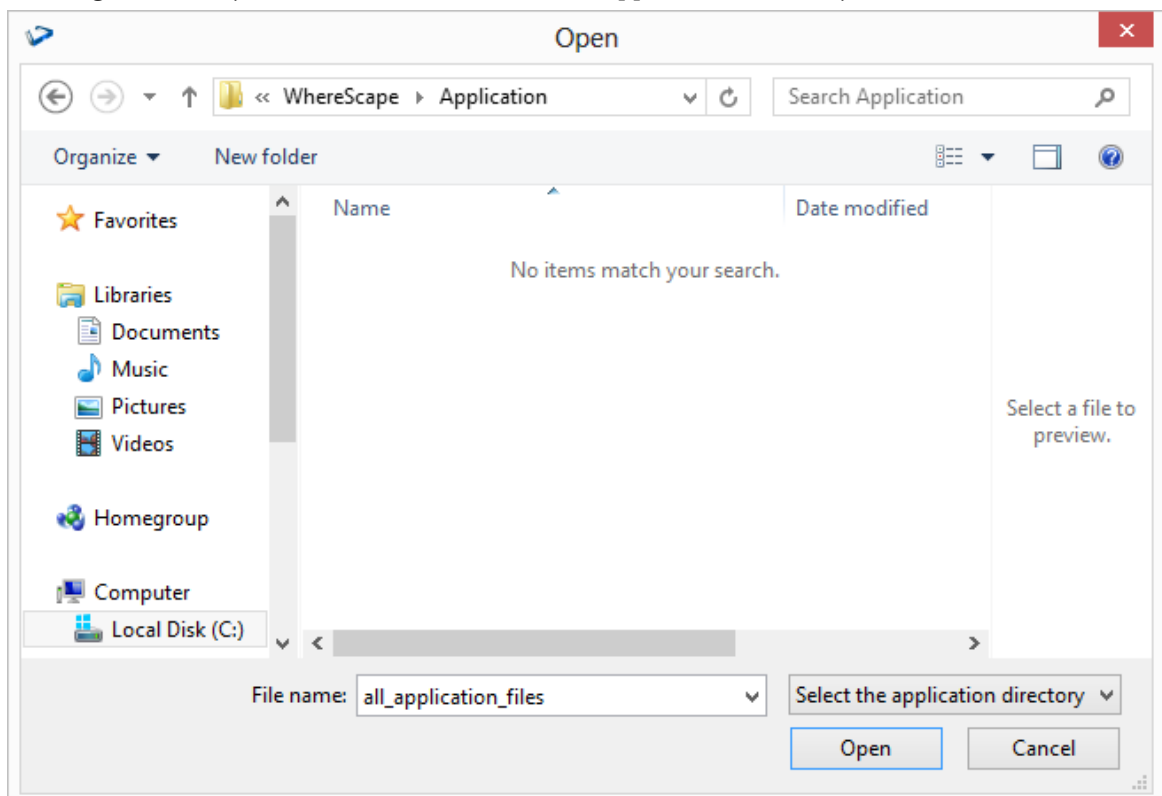
/PF absolute xml parameter file name

12.3 CHANGING THE APPLICATION DIRECTORY

- 1 In WhereScape's Setup Administrator, click on the **Application** tab in the left pane, then click on the **Change Application Directory** icon.

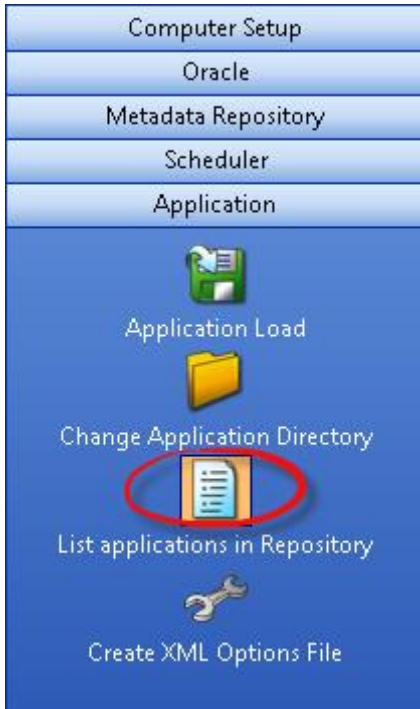


- 2 A dialog will allow you to browse and select a new application directory.



12.4 LISTING APPLICATIONS

- 1 In WhereScape's Setup Administrator, click on the **Application** tab in the left pane, then click on the **List applications in Repository** icon.



- 2 The **List applications in Repository** menu option will prompt for a logon to the required metadata repository. Applications are applied to a specific metadata repository. Once logged on a list of the applications that have been installed into that repository will be shown. If no applications have been loaded then the list will be empty. An example output is as follows:

List Installed Applications				
App Id	Name	Version	Status	Date installed
SQLT	WhereScape Tutorials (SQL Server)	tutorial_2	Completed	2007-02-13 17:32:16.257

This example shows that the tutorial_2 application has been loaded into the metadata.

12.5 TESTING APPLICATIONS

A testing application set consists of a Procedure and an XML script and provides the ability to define a series of tests against data warehouse objects; either comparing them to an expected value or to the results of a query.

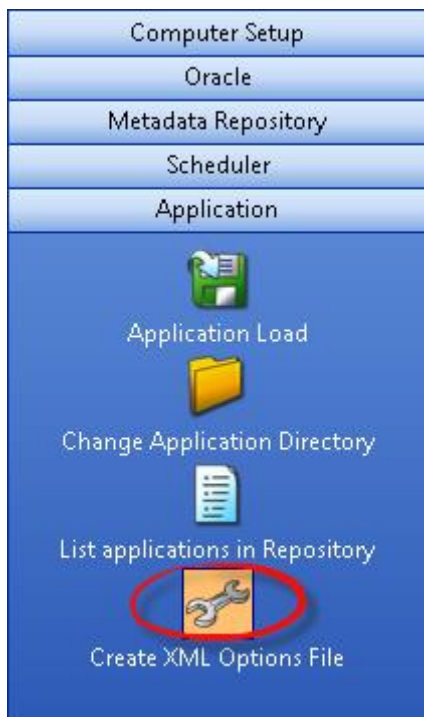
The XML script contains the test definitions. Each test is a new XML node in the XML script and contains a name, a test query, an expected value, or a comparison query.

The procedure simply runs the tests and determines whether the tests are passed or not. This is most likely to be run as a scheduled job within WhereScape RED.

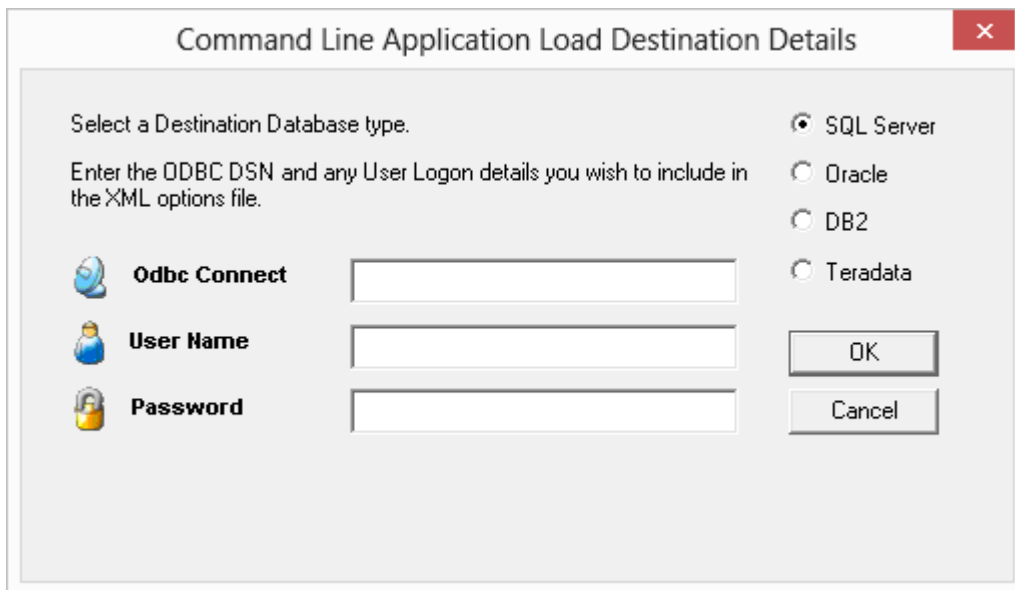
12.6 XML FILES FOR APPLICATION LOADS

If applications are being loaded using the command line, the saved application load options in the local registry can be outputted to an xml file for use in the application load.

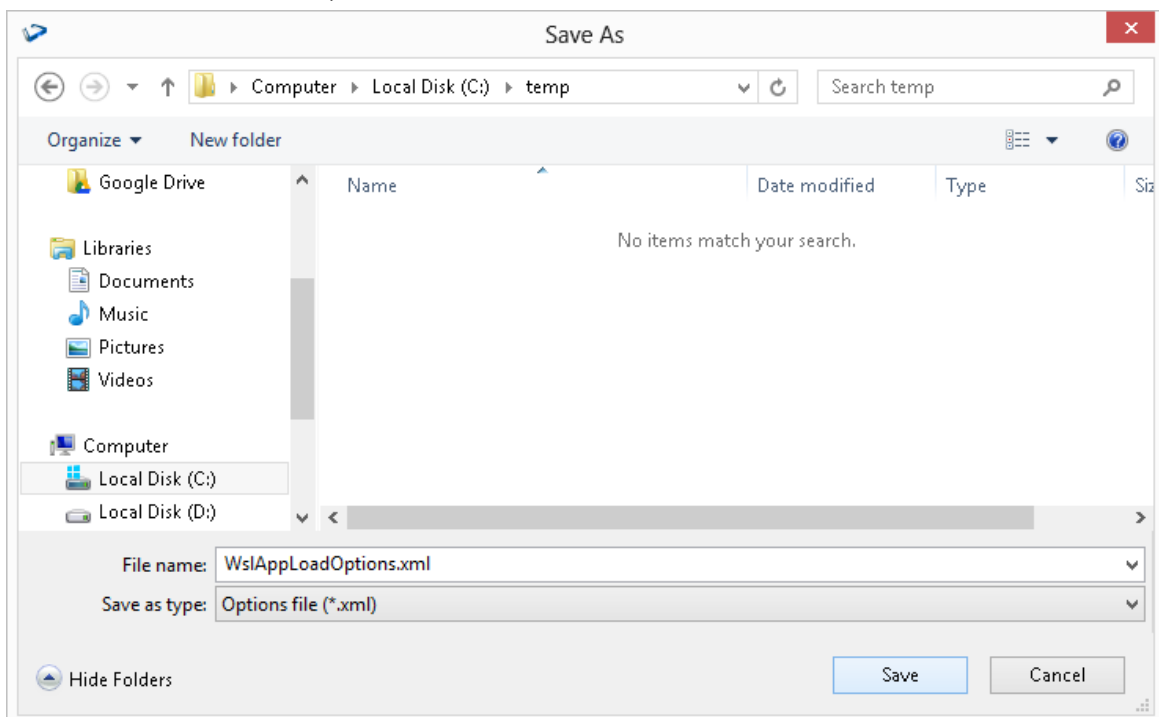
- 1 To create the xml file click on **Create XML Options File** in the left pane.







- 2 If the load options have not yet been saved into the local registry, a message will appear; prompting you to select the application to be loaded and to saved the load settings for that application.
 - Enter in the **Odbc Connection**, **User Name** and **Password** to add the destination details to the XML options file.



- 3 Choose the location where you want to save the XML.



- 4 The results will confirm that the XML file has been created.

Computer Setup	Application Load XML File
Oracle	Results
Metadata Repository	XML file C:\temp\Wsl\ApplLoadOptions.xml created.
Scheduler	
Application	
 Application Load	
 Change Application Directory	
 List applications in Repository	
 Create XML Options File	

CHAPTER 13

13. UPGRADING WHERESCAPE RED

WhereScape Versions

WhereScape RED has a four part version number normally shown as xx.xx.xx.xx. An example of this may be 6.0.6.0. The first number represents the major release. The second number represents the metadata repository version. The third and fourth numbers relate to application specific releases.

From the example above we see that the current version is version 6 of WhereScape RED. We are on version 6.0 of the metadata repository.

Metadata Changes

A change from a 6.0.. release to a 6.5.. release would indicate a change in the metadata tables. All metadata changes are non destructive. They simply add new columns or new meta tables. In this way they can be applied without harming an existing metadata repository. The impact of a metadata change is that the associated applications (namely the RED executable, the Scheduler, security module etc.) will need to be at the same metadata version. Therefore a 6.0.6.0 version of RED may not successfully run against a version 6.5 metadata repository.

Application changes

The final two numbers in the version represent application releases. Applications are deemed to be all of the executable images supplied with WhereScape RED as well as the UNIX scheduler scripts and the stored procedures. Application changes reflect enhancements and bug fixes. A change in the first number indicates a major enhancement in one of the application components.

Upgrading RED

Upgrading WhereScape RED consists of the following steps:

- 1 Allow any active jobs to complete. Halting active jobs will allow running tasks to complete with no new tasks starting. Aborting active jobs will kill any running tasks and stop running jobs.
- 2 Stop all schedulers. Windows schedulers can be stopped with WhereScape's Setup Administrator. To stop a UNIX or Linux scheduler, kill the active scheduler process and comment their crontab entries (to stop the scheduler re-starting itself).
- 3 Close any WhereScape programs that are running on your machine.
- 4 Install the new version of RED on your machine.
- 5 Back up your metadata.
- 6 In RED Setup Administrator, select the Validate Metadata Repository option. This function may re-compile existing or create new metadata procedures; metadata tables may be altered or new tables created.

When performing an upgrade of RED:

- Click **OK** when prompted to validate metadata tables.
- Click **Yes** when prompted to re-create metadata views.
- Click **Yes** when prompted to re-compile metadata procedures.

Warning: If the above steps are not all completed during an upgrade the metadata may be left in an inconsistent state.

- 7 For **Oracle** metadata updates see the note below on **recompiling Oracle invalid procedures** during a RED upgrade.
- 8 Back up your metadata again (just in case).
- 9 If using a UNIX or Linux scheduler and this is a major application enhancement then rename the wsl/bin directory to say wsl/bin_versionxxxxxx. Create a new bin directory and ftp over the files under the unix directory (see the main install instructions). Change the protections on the files (chmod 750 *.sh).
- 10 Restart all schedulers. Windows schedulers can be restarted with WhereScape's Setup Administrator. For any UNIX or Linux schedulers, uncomment any commented out crontab entries - this is enough to restart the schedulers.
- 11 Restart any halted or aborted jobs.

NOTES:

1. There are different versions of the scripts for each database and for UNIX versus Linux. You should also be looking in the sub directory with the highest version number.



When using a UNIX/Linux scheduler for Oracle, it is recommended that users copy the new meta_backup_680.sh file to use Datapump's expdb/impdp instead the deprecated exp/imp tools.

This version uses the data pump export executables expdp/impdp. It assumes that the scheduler and the Oracle database reside on the same server.

2. Metadata tables do not change between minor releases, but metadata procedure often and usually do change.

3. RED will not let you sign into an old repository version using a newer version of RED

4. RED will let you sign into a new repository version using an older version of RED, but it will warn you that this may potentially cause issues

5. It is very important when using the windows scheduler to have the installed RED version on the scheduler server EXACTLY matching the stored procedures.

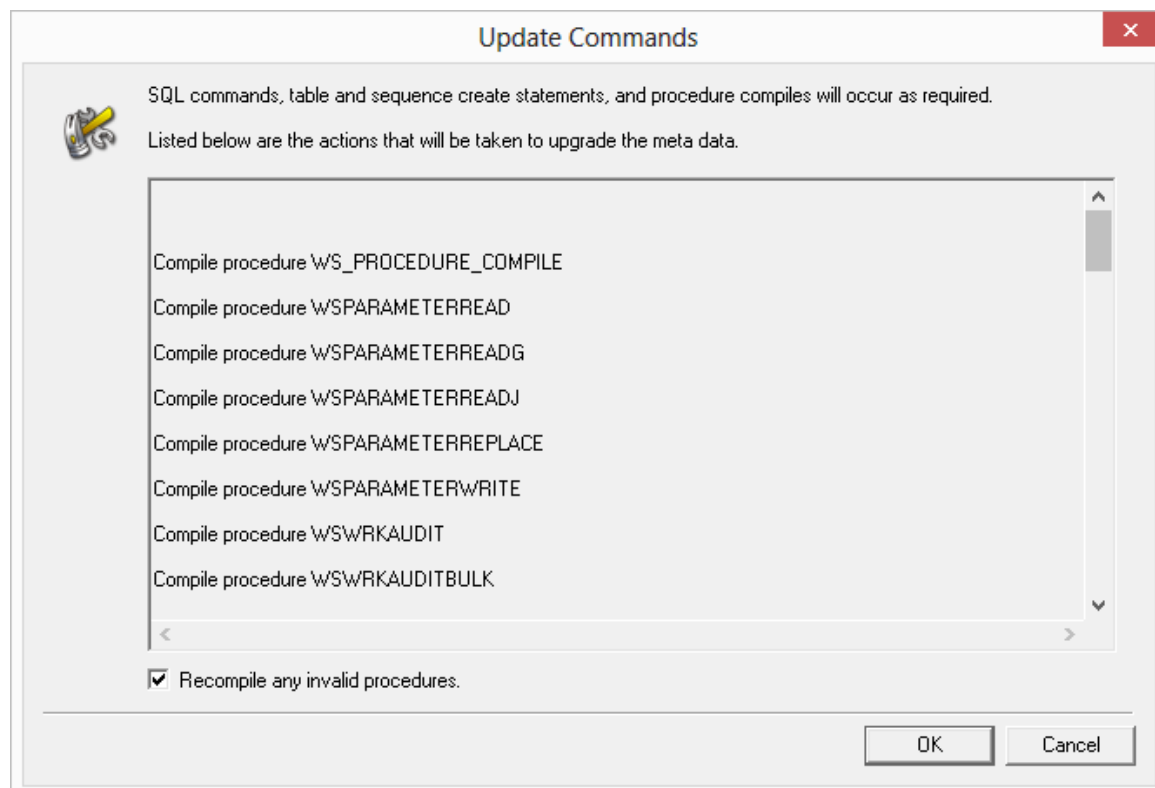
6. Side by side installations are possible (two versions in two directories on the same machine), but be careful with schedulers. If you install the new version of RED in a new directory, you will have to remove and reinstall all windows schedulers in order for the scheduler(s) to use the new version of RED.

7. As a general rule, the UNIX and Linux scheduler scripts, metadata tables/procedures and RED front end should all be in sync.

8. If the scripts for the UNIX or Linux scheduler have changed, you should replace files in your old and new shell scripts in the WhereScape Program Files directory and see if any have changed.

Oracle invalid procedures

When validating an Oracle Metadata Repository, tick the **Recompile any invalid procedures** check-box to have any invalid user procedures recompiled during a metadata validate.

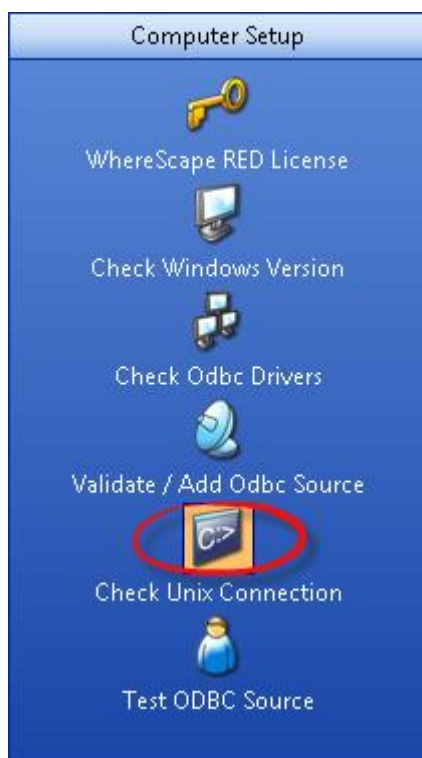


CHAPTER 14

14. CHECKING THE UNIX CONNECTION

This section applies to **IBM DB2** and **Oracle** data warehouses only.

- 1 To check the UNIX connection, click on the **Computer Setup** tab.
 - Then click on the **Check Unix Connection** icon.



WhereScape RED can allow the user to browse a Unix file system and utilize the drag and drop functionality to setup the loading of files from a Unix file system. It can also perform interactive file loads. To facilitate this and to obviate the need to port code to every Unix environment the Telnet protocol is used for the Unix connectivity.

This step is not necessary for the successful running of a Unix based data warehouse environment. As mentioned above it is provided to simplify the process of setting up the loading of Unix files. This step is totally unnecessary for a Windows based data warehouse environment.

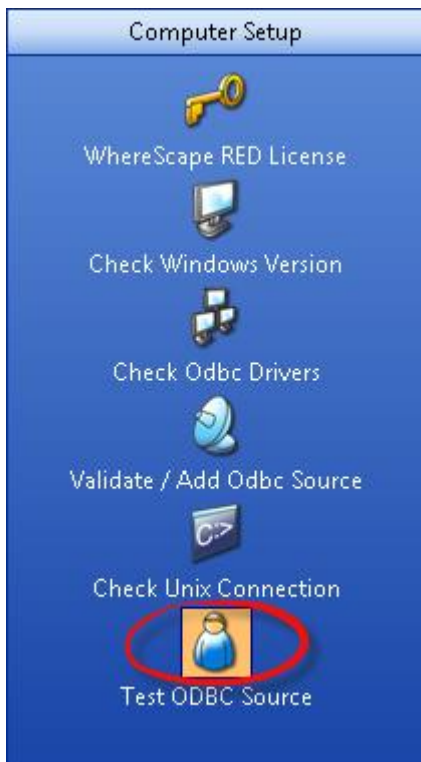
If you want to use these Unix drag and drop assistants then consult the WhereScape RED User Guide for an explanation of Unix connections and perform the following tasks:

- 1 Identify a Unix user that allows direct logon to a Unix shell prompt without passing through a menu system.
- 2 Ensure that the Unix user has the Oracle 'sqlldr' program in the path after a normal logon. If not then interactive loads will not be possible. Scheduled loads should still function correctly.
- 3 Complete the dialog box presented specifying the Unix user and password identified in step (1).
- 4 Click **OK** to proceed. A timeout of 20 seconds is enabled, so that if connection is not achieved within that time the process will terminate. The Telnet window will be displayed to assist in debugging any problems. It will be closed on completion or timeout.
- 5 Review the results and attempt to address any problems. A file called 'WslTelnet.log' is created in the installation directory containing the full dialog of the connection attempt. The message 'Telnet connection worked. All OK' will be displayed if the connection worked. If successful record the parameters used as these will need to be defined when setting up a Connection within WhereScape RED itself. If a connection cannot be achieved then remove all reference to a username and password from the file mentioned above and mail it to support@wherescape.com.

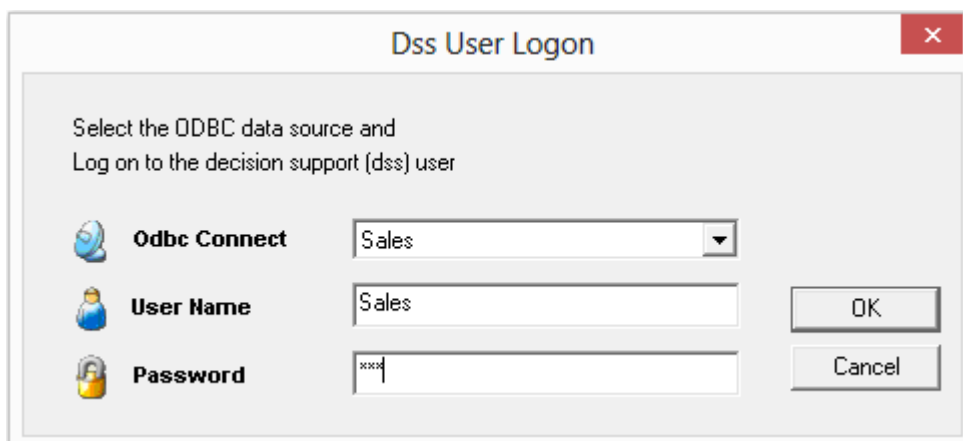
CHAPTER 15

15. TESTING AN ODBC SOURCE

- 1 To test an ODBC source, click on the **Computer Setup** tab.
 - Then click on the **Test ODBC Source** icon.



- 2 Select the **ODBC connection** from the drop-down list and enter the **User Name** and **Password**.
 - Click **OK**.



- 3 If the logon using the ODBC connection was successful, the following message will be displayed.

Computer Setup	Test ODBC connection		
 <p>WhereScape RED License</p>  <p>Check Windows Version</p>  <p>Check Odbc Drivers</p>  <p>Validate / Add Odbc Source</p>  <p>Check Unix Connection</p>  <p>Test ODBC Source</p>	<table border="1"><thead><tr><th data-bbox="715 396 1445 432">Result</th></tr></thead><tbody><tr><td data-bbox="715 432 1445 468">Sales logged on using Odbc connection Sales successfully. All OK</td></tr></tbody></table>	Result	Sales logged on using Odbc connection Sales successfully. All OK
Result			
Sales logged on using Odbc connection Sales successfully. All OK			

CHAPTER 16

16. VALIDATING A META REPOSITORY

- 1 To validate a Metadata Repository, click on the **Metadata Repository** tab.
 - Then click on the **Validate Metadata Repository** icon.



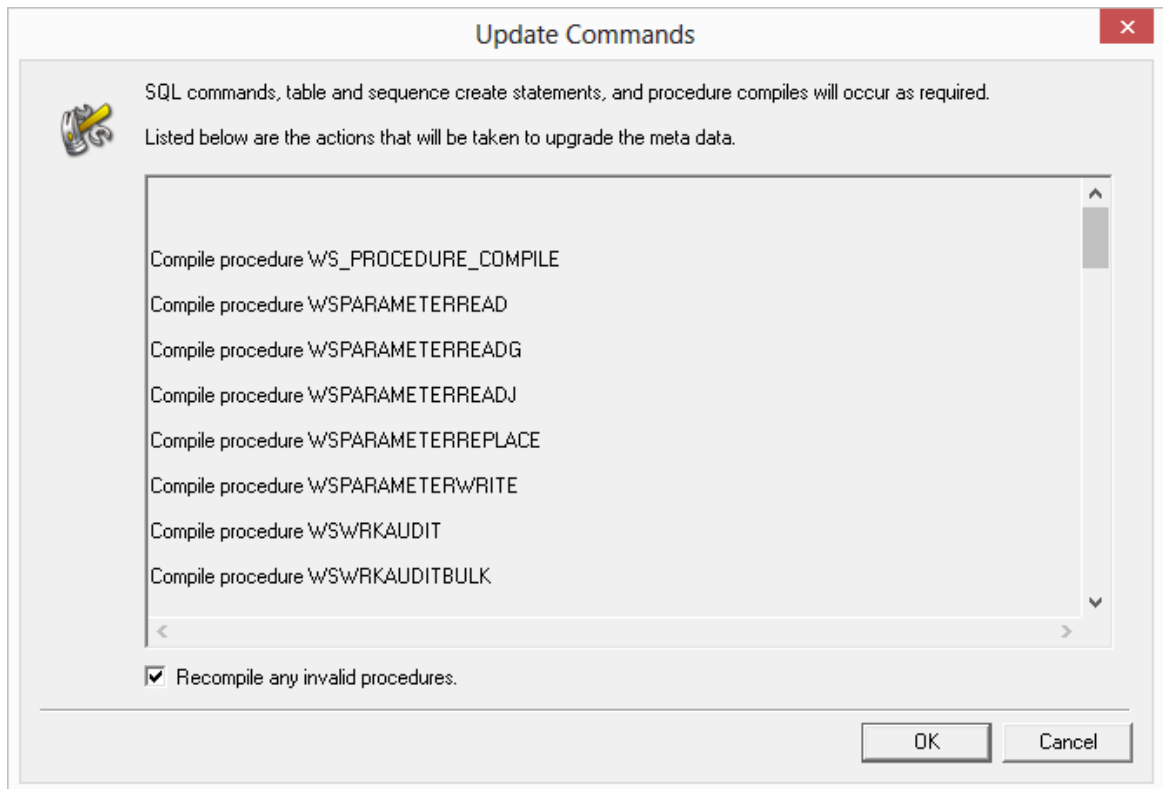
This option validates the version of the meta tables installed in the specified SQL Server database, Oracle schema, DB2 schema or Teradata Database/User. If requested it goes on to validate the tables, sequences and procedures against the installation media. A validate can be run at any stage to confirm that all tables and procedures are up to date and that the procedures are in a compiled and valid state. This option can be used to recompile a procedure that has become invalid. This option should be used when installing a new version of RED and for SQL Server all procedures should be recompiled.

If a mismatch is found or a procedure is found to be invalid, the user is given the option of updating the metadata. If chosen the metadata will be upgraded to the latest version.

NOTE: Ensure the scheduler is not running and no users are accessing the metadata before doing an upgrade.

Oracle invalid procedures

When validating an Oracle Metadata Repository, tick the **Recompile any invalid procedures** check-box to have any invalid user procedures recompiled during a metadata validate.



IN THIS CHAPTER

16.1 Validating a Metadata Repository using the Command Line 172

16.1 VALIDATING A METADATA REPOSITORY USING THE COMMAND LINE

Is it possible to Validate the Metadata Repository using the Command Line by running a bat file. This process can be used for all metadata database types: it validates the tables and alters or creates/copies any tables that require changes.

All the process steps are logged to a file: **My Documents\WhereScape\WhereScape_quick_validate.log**.

During this process:

- All Teradata metadata views are recreated
- Oracle sequences and procedures are validated and updated if required
- All non Oracle procedures are recompiled
- All metadata indexes are recreated
- Any Oracle non metadata procedures that are invalid are compiled
- Finally the metadata version in ws_meta_tables.mt_version is updated

To have a metadata repository validated from the command line, create a bat file using the following batch application quick validate options and check the example below:

Batch Application Quick Validate options

ADM -QV

-SN DSN
-SL LoginId
-SP Password

-DS
//SQLServer Dimension filegroup
//Teradata – not used
//Oracle Dimension Table Tablespace
//DB2 Ancillary Table Tablespace

-IS.....
//SQLServer Dimension Index Filegroup
//Teradata – not used
//Oracle Dimension Index Tablespace
//DB2 Metadata Repository Schema

-MS.....
//SQLServer Metadata Filegroup
//Teradata Metadata Repository Database
//Oracle Metadata Repository Tablespace
//DB2 Metadata Repository Tablespace

Example

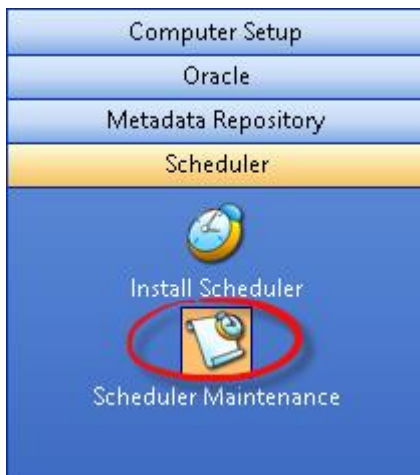
The following example validates an Oracle repository named **WslWarehouse_ORA**, using a connection with username **dssdemo** and password **wsl**.
It recreates the Oracle Dimension Index Tablespace, validates the Oracle Metadata Repository Tablespace and validates/updates the Oracle Dimension Table Tablespace, recompiling any procedures that may be invalid, during the validate process.

```
@echo off
cd "c:\Program Files (x86)\wherescape\"
adm -QV -SN WslWarehouse_ORA -SL dssdemo -SP wsl -IS RED_METADATA -MS
RED_METADATA -DS RED_METADATA
exit
```

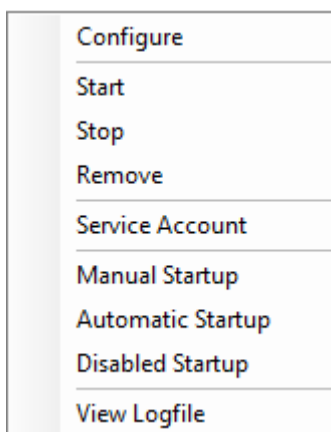
CHAPTER 17

17. MAINTAINING THE WINDOWS SCHEDULER

- 1 To maintain the Windows scheduler, click on the **Scheduler** tab.
 - Then click on the **Scheduler Maintenance** icon.



- 2 This option lists all currently active WhereScape schedulers. By using the right-click menu when positioned over a scheduler name it is possible to start, stop, remove, change the settings, or view the log of a scheduler. The menu is as follows:



3 The options are:

Option	Description
Configure	Allows the alteration of the service configuration. You can change the level of logging. The default is 2. Level 0 provides no log information and level 9 provides full debug. See Scheduler Creation for more detail.
Start	Starts a stopped scheduler. This action requires service control manager (SCM) privileges on the PC.
Stop	Stop a running scheduler. SCM privilege required.
Remove	Removes and deletes the scheduler. SCM required.
Service Account	By default the scheduler will start in the system account. If you wish to change this you can specify an account and password. However, you may have to remove the service and enter the account information when creating a new scheduler.
Manual Startup	Changes the startup profile (on PC startup) of the scheduler. The scheduler will not be automatically started when the PC starts and must be started manually. SCM required.
Automatic Startup	Sets the scheduler to start automatically when the PC starts. SCM required.
Disabled Startup	Sets the scheduler startup profile to disabled. No startup will occur unless the profile is changed to either Manual or Automatic. SCM required.
View Logfile	Displays the scheduler service log file. The level of logging is defined in the configuration of the scheduler.

NOTE: If an error occurs while a scheduled job is running, you should check the following:

- the audit log of the failed task, for any messages of type E or F.
- the audit log of the failed job, for any messages of type E or F.
- the scheduler log file for any error messages.
- the windows event logs, specifically the **Application** and **System** event logs, for any Error or Warning messages at the time of the job failure.

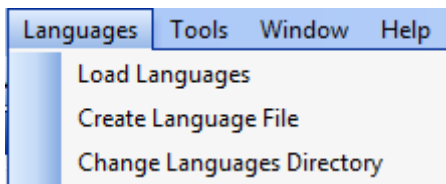
How to view and manage event logs depends on which version of Windows you are using. If you need further assistance, please contact support at support@wherescape.com.

CHAPTER 18

18. LANGUAGES

The Languages component of WhereScape's Setup Administrator allows you to create a Language file or to load a Language File into RED.

- 1 Click on the **Languages** menu item in the command bar at the top of the screen.
 - Click on any of the options under this level to action them. The Languages menu is as follows:



The **Load Languages** (see "**18.1 Loading Languages**" on page 178) option will list the Language files that have been created. A file can then be selected and loaded into the target Metadata Repository.

The **Create Language File** (see "**18.2 Creating a Language File**" on page 179) option allows you to create a Language file for a particular Metadata Repository. A translator will then need to replace the English definitions in the file with the translations relevant to the chosen language.

The **Change Languages Directory** (see "**18.3 Changing the Languages Directory**" on page 180) option allows you to change the language directory.

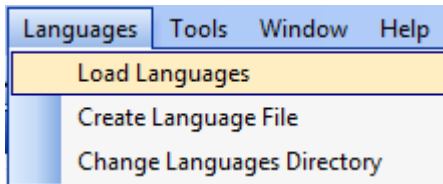
IN THIS CHAPTER

18.1 Loading Languages	178
18.2 Creating a Language File.....	179
18.3 Changing the Languages Directory	180

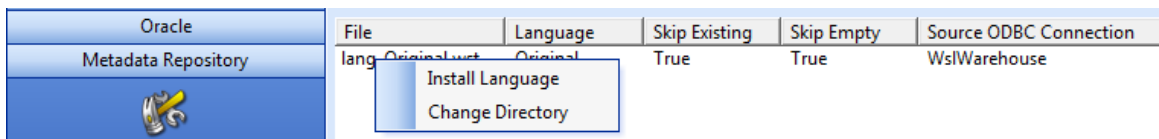
18.1 LOADING LANGUAGES

Note: Before you install a language, you will first need to logon to the repository in RED; go to **Tools/Language Options** and add the language.

- 1 Click on the **Languages** menu item in the command bar at the top of the screen, then choose **Load Languages**.



- 1 The **Load Languages** menu option will list the Language Files available.
 - Right-click on a particular file and select **Install Language**.



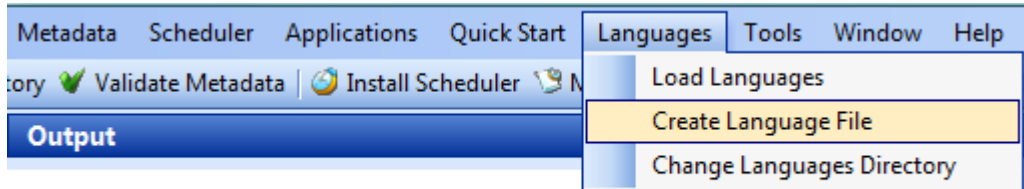
File	Language	Skip Existing	Skip Empty	Source ODBC Connection
lang_Original.wst	Original	True	True	WsIWarehouse

You will then be prompted to logon to the required metadata repository and to select the Language to be updated in RED.

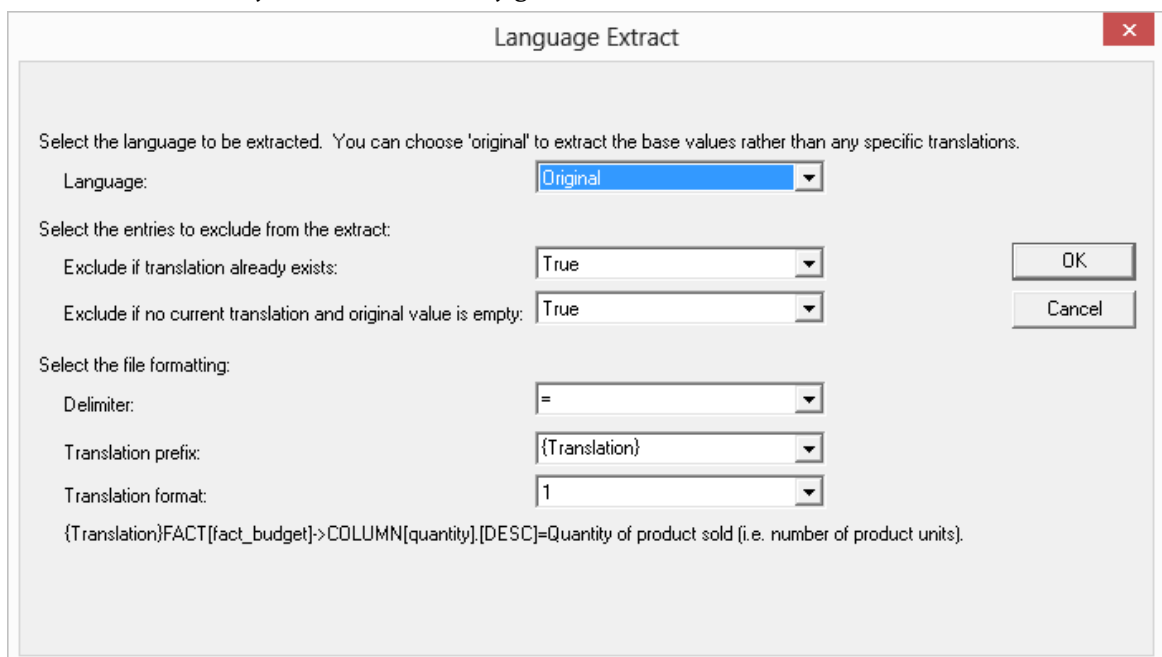
The results will be shown in the middle pane.

18.2 CREATING A LANGUAGE FILE

- 1 Click on the **Languages** menu item in the command bar at the top of the screen.
 - Select the **Create Language File** option.



- 2 You will then be prompted to logon to the required metadata repository. Language Files are applied to a specific metadata repository. Once logged on, you will be prompted for:
 - The **Language** - select from the drop-down list. Select a Language or choose **Original** to create a Language file containing the original values.
 - The **entries to exclude** from the extract.
 - The file delimiter, translation prefix and translation format.
 - The **file delimiter** provides an easy means of breaking up the file after extract.
 - The **translation prefix** must be something that will NEVER appear in the data itself. This is used to identify the start/end of any given field translation.

A screenshot of the 'Language Extract' dialog box. The title bar says 'Language Extract' with a close button (X) on the right. The dialog contains several sections:

- Select the language to be extracted.** You can choose 'original' to extract the base values rather than any specific translations. The 'Language:' dropdown is set to 'Original'.
- Select the entries to exclude from the extract:**
 - 'Exclude if translation already exists:' dropdown is set to 'True'.
 - 'Exclude if no current translation and original value is empty:' dropdown is set to 'True'.
- Select the file formatting:**
 - 'Delimiter:' dropdown is set to '='.
 - 'Translation prefix:' dropdown is set to '{Translation}'.
 - 'Translation format:' dropdown is set to '1'.

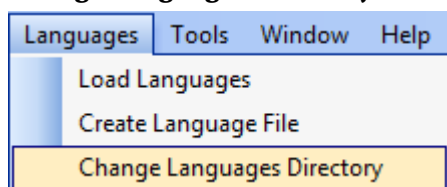
At the bottom, there is a preview of the translation format: `{Translation}FACT[fact_budget]->COLUMN[quantity],[DESC]=Quantity of product sold (i.e. number of product units).` On the right side, there are 'OK' and 'Cancel' buttons.

A Language extraction file will then be created and saved into **C:\ProgramData\RED\Language**

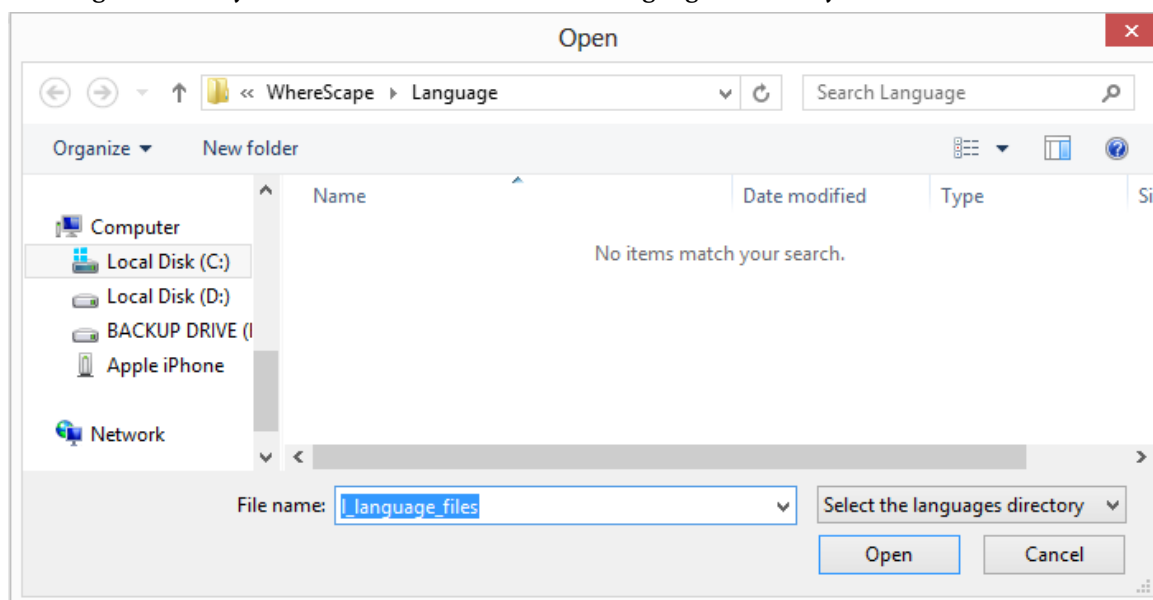
A translator will need to replace the English definitions in the file with the translations relevant to the chosen language.

18.3 CHANGING THE LANGUAGES DIRECTORY

- 1 Click on the **Languages** menu item in the command bar at the top of the screen and select **Change Languages Directory**.



- 2 A dialog will allow you to browse and select a new languages directory.



CHAPTER 19

19. TOOLS

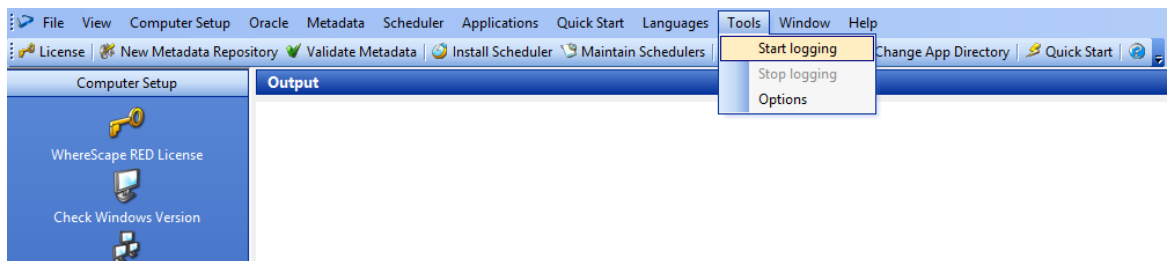
IN THIS CHAPTER

19.1 Start/Stop logging 181

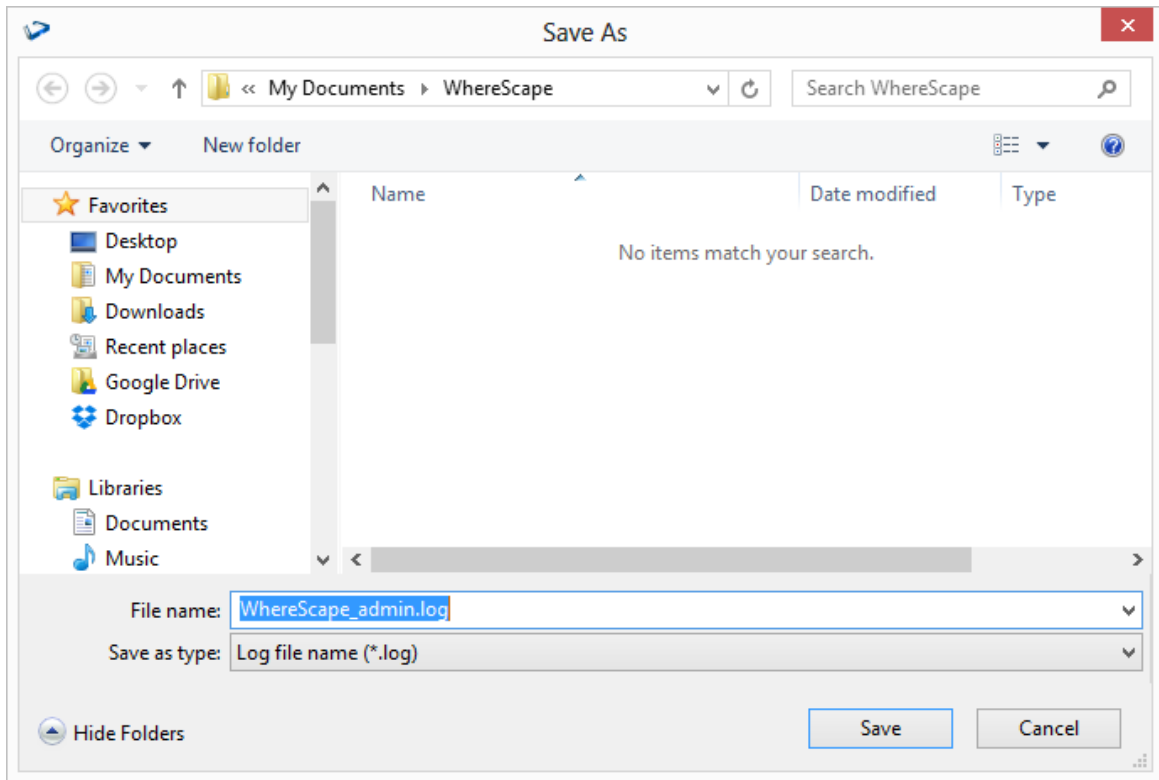
19.1 START/STOP LOGGING

In WhereScape Administrator, users have the option of logging all output to a file by selecting the **Start logging** function in the **Tools** menu.

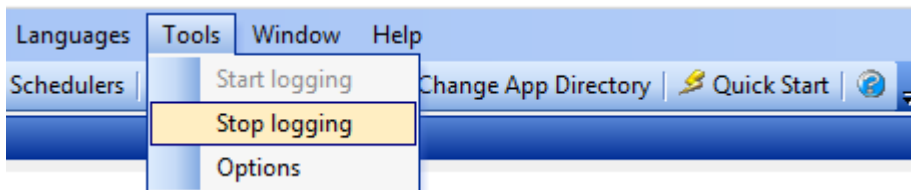
- 1 To starting logging output from WhereScape Administrator, click on **Tools>Start logging** in the top bar.



- 2 When this option is selected, users are prompted to create a file in a specified location. Once this is done, all the output from WhereScape Administrator starts getting logged to that file.



- 3 To cancel logging, click **Tools>Stop logging**.



INDEX

1

1. Overview	5
1.1 Documentation Roadmap	6
1.2 How to use this Manual	7
10. DB2 Installation and Configuration ..	8, 100
10.1 Installation and Configuration	
Introduction	100
10.2 Installing RED for DB2	101, 102
10.3 Creating a DB2 Schema	101, 103
10.4 Configuring a DB2 Metadata Connection (DSN)	101, 106
10.5 Creating a Metadata Repository for DB2	101, 109, 114
10.6 Installing a Windows Scheduler for DB2	101, 111
10.7 Installing a Unix Scheduler for DB2 ...	101, 114
11. Tutorial Data	8, 120
11.1 Creating Tutorial Data	21, 43, 66, 101, 120, 121
11.2 Validating Tutorial Data.....	120, 122
12. Applications	8, 123
12.1 Loading an Application	123, 124
12.1.1 Backing Up the Metadata before Loading Applications.....	125, 126
12.1.2 Application Load Process	127
12.2 Creating and Loading Applications from the Command Line	135
12.3 Changing the Application Directory ..	123, 136
12.4 Listing Applications	123, 138
12.5 Testing Applications	139
12.6 XML Files for Application Loads	140
13. Upgrading WhereScape RED	8, 142
14. Checking the Unix Connection	8, 146
15. Testing an ODBC Source	8, 148
16. Validating a Meta Repository.....	8, 150
16.1 Validating a Metadata Repository using the Command Line	151
17. Maintaining the Windows Scheduler	8, 153
18. Languages.....	8, 156
18.1 Loading Languages.....	156, 157
18.2 Creating a Language File	156, 158

18.3 Changing the Languages Directory	156, 159
--	----------

19. Tools	160
19.1 Start/Stop logging	160

2

2. Installing WhereScape RED	9, 23, 44, 68, 102
2.1 Installing WhereScape RED from the Command Line	13

3

3. Adding Your License	14, 21, 43, 66, 101
------------------------------	---------------------

4

4. Checking Your Windows Version .	16, 21, 43, 66, 101
------------------------------------	---------------------

5

5. Checking Your ODBC Drivers .	17, 21, 43, 66, 101
---------------------------------	---------------------

6

6. Creating an ODBC Connection	19, 24, 46, 75, 106
--------------------------------	---------------------

7

7. SQL Server Installation and Configuration	8, 21
--	-------

7.1 Installation and Configuration	
Introduction	21

7.2 Installing RED for SQL Server	21, 22
---	--------

7.3 SQL Server Database and Login Setup	23
--	----

7.3.1 SQL Server Database	23
---------------------------------	----

7.3.2 SQL Server Login Setup.....	21, 23
-----------------------------------	--------

7.4 Configuring a SQL Server Metadata Connection (DSN)	21, 24
--	--------

7.5 Creating a Metadata Repository for SQL Server	21, 29
---	--------

7.6 Installing a Windows Scheduler for SQL Server	21, 31
---	--------

7.7 SQL Server Quick Start.....	36
---------------------------------	----

7.8 SQL Server Quick Application.....	38
---------------------------------------	----

8

8. Teradata Installation and Configuration ...8,	42
8.1 Installation and Configuration	
Introduction	42
8.2 Installing RED for Teradata.....	43, 44
8.3 Creating a Teradata Environment	43, 45
8.4 Configuring a Teradata Metadata	
Connection (DSN).....	43, 46
8.5 Creating a Metadata Repository for	
Teradata.....	43, 50
8.6 Installing a Windows Scheduler for	
Teradata.....	43, 52
8.7 Installing a Unix Scheduler for Teradata	43,
56	
8.8 Installing a Teradata Unix Loader	43, 62

9

9. Oracle Installation and Configuration .. 8,	65
9.1 Installation and Configuration	
Introduction	65
9.2 Installing RED for Oracle	66, 67
9.3 Creating an Oracle User	66, 69
9.3.1 Creating Oracle Individual Users	71
9.4 Configuring an Oracle Metadata	
Connection (DSN).....	66, 75
9.5 Validating Oracle User Privileges.....	80
9.6 Creating a Metadata Repository for Oracle	
.....	66, 83
9.7 Installing a Windows Scheduler for Oracle	
.....	66, 86
9.8 Installing a Unix Scheduler for Oracle ...	66,
91	

U

Unix Scheduler setup for Hadoop loads using	
Oracle's Big Data Connectors	98