

# Retek<sup>®</sup> Data Warehouse<sup>™</sup> 11.0

## Database Installation Guide



---

**Corporate Headquarters:**

Retek Inc.  
Retek on the Mall  
950 Nicollet Mall  
Minneapolis, MN 55403  
USA  
888.61.RETEK (toll free US)  
Switchboard:  
+1 612 587 5000  
Fax:  
+1 612 587 5100

**European Headquarters:**

Retek  
110 Wigmore Street  
London  
W1U 3RW  
United Kingdom  
Switchboard:  
+44 (0)20 7563 4600  
Sales Enquiries:  
+44 (0)20 7563 46 46  
Fax:  
+44 (0)20 7563 46 10

The software described in this documentation is furnished under a license agreement, is the confidential information of Retek Inc., and may be used only in accordance with the terms of the agreement.

No part of this documentation may be reproduced or transmitted in any form or by any means without the express written permission of Retek Inc., Retek on the Mall, 950 Nicollet Mall, Minneapolis, MN 55403, and the copyright notice may not be removed without the consent of Retek Inc.

Information in this documentation is subject to change without notice.

Retek provides product documentation in a read-only-format to ensure content integrity. Retek Customer Support cannot support documentation that has been changed without Retek authorization.

The functionality described herein applies to this version, as reflected on the title page of this document, and to no other versions of software, including without limitation subsequent releases of the same software component. The functionality described herein will change from time to time with the release of new versions of software and Retek reserves the right to make such modifications at its absolute discretion.

Retek® Data Warehouse™ is a trademark of Retek Inc.

Retek and the Retek logo are registered trademarks of Retek Inc.

This unpublished work is protected by confidentiality agreement, and by trade secret, copyright, and other laws. In the event of publication, the following notice shall apply:

©2004 Retek Inc. All rights reserved.

All other product names mentioned are trademarks or registered trademarks of their respective owners and should be treated as such.

Printed in the United States of America.

## Customer Support

### Customer Support hours

Customer Support is available 7x24x365 via email, phone, and Web access.

Depending on the Support option chosen by a particular client (Standard, Plus, or Premium), the times that certain services are delivered may be restricted. Severity 1 (Critical) issues are addressed on a 7x24 basis and receive continuous attention until resolved, for all clients on active maintenance. Retek customers on active maintenance agreements may contact a global Customer Support representative in accordance with contract terms in one of the following ways.

Contact Method	Contact Information
----------------	---------------------

E-mail	support@retек.com
--------	-------------------

Internet (ROCS)	<a href="https://rocs.retek.com">rocs.retek.com</a> Retek's secure client Web site to update and view issues
-----------------	---

Phone	+1 612 587 5800
-------	-----------------

Toll free alternatives are also available in various regions of the world:

Australia	+1 800 555 923 (AU-Telstra) or +1 800 000 562 (AU-Optus)
France	0800 90 91 66
Hong Kong	800 96 4262
Korea	00 308 13 1342
United Kingdom	0800 917 2863
United States	+1 800 61 RETEK or 800 617 3835

Mail	Retek Customer Support Retek on the Mall 950 Nicollet Mall Minneapolis, MN 55403
------	---

### When contacting Customer Support, please provide:

- Product version and program/module name.
- Functional and technical description of the problem (include business impact).
- Detailed step-by-step instructions to recreate.
- Exact error message received.
- Screen shots of each step you take.

# Contents

<b>Chapter 1 – Introduction .....</b>	<b>1</b>
Audience .....	1
Database independence .....	1
Typographic conventions.....	1
Command syntax .....	1
<b>Chapter 2 – Data Warehouse server and system requirements ..</b>	<b>3</b>
Data warehouse server (target server).....	3
Hardware requirements .....	4
Software requirements and configuration matrix .....	5
Capacity planning .....	7
Client system requirements.....	7
<b>Chapter 3 – RDW Unix environment setup .....</b>	<b>9</b>
Installation instructions.....	9
Create Unix user accounts .....	9
Create Unix directories.....	10
Alter directory ownership and privileges .....	11
Copy profiles for Retek users .....	12
<b>Chapter 4 – RDW Oracle RDBMS installation and configuration</b>	<b>13</b>
General installation notes.....	13
Oracle Concepts for the Retek Data Warehouse.....	13
Installation instructions.....	13
Oracle database configuration .....	13
<b>Chapter 5 – RDW DB2 RDBMS installation and configuration...</b>	<b>19</b>
General installation notes.....	19
DB2 Concepts for the Retek Data Warehouse.....	20
Installation instructions.....	21
DB2 database configuration .....	21

<b>Chapter 6 – RDW Teradata RDBMS installation and configuration</b>	<b>25</b>
General installation notes.....	25
Teradata Concepts for the Retek Data Warehouse .....	25
Installation instructions .....	26
Teradata database configuration.....	26
<b>Chapter 7 – Default data population .....</b>	<b>29</b>
ETL installation and configuration .....	29
Data population .....	29
Populate datamart default data .....	29
Populate time dimension .....	31
Populate static dimension data .....	33
<b>Appendix A – Oracle information.....</b>	<b>35</b>
Sample Oracle Net Services File for the Server .....	35
<b>Appendix B – DB2 information.....</b>	<b>37</b>
Sample database configuration parameters on DB2 .....	37
<b>Appendix C – Additional information .....</b>	<b>39</b>
Development Database Schema implementation.....	39
RDW Code Tree .....	40

# Chapter 1 – Introduction

## Audience

This document is intended for database administrators responsible for installing the Retek Data Warehouse. The database administrator is expected to have significant expertise in creating databases with the chosen database platform.

## Database independence

This guide contains instructions on how to install RDW for Oracle, DB2, and Teradata. Please note, at the time of the 12/31/04 RDW 11 release, RDW is currently only certified on Oracle. Please contact Retek Support about availability of RDW 11 for DB2 and Teradata.

## Typographic conventions

### **monospace**

Monospace type indicates UNIX commands, directory names, usernames, pathnames, and filenames.

### **brackets [ ]**

Words enclosed in brackets indicate key names (for example: Press [Return]).



**Note:** Brackets have a different meaning when used in command syntax.

### ***italics***

Italic type indicates a variable, including variable portions of filenames. It is also used for emphasis.

### **UPPERCASE**

Uppercase letters indicate Structured Query Language (SQL) reserved words, initialization parameters, and environment variables.

## Command syntax

UNIX command syntax appears in monospace font and assumes the use of the Korn shell. The "%" character at the beginning of UNIX command examples should not be entered at the prompt.

**backslash \** A backslash indicates a command that is too long to fit on a single line. Enter the line as printed (with a backslash) or enter it as a single line without a backslash:

```
imp system/password log=rdw11_md.log \ file=rdw11md.dmp ignore=y
grants=n \
fromuser=rdw11md touser=rdw11md
```

In the following instructions, the character '\ ' in a path name indicates a DOS-based file structure, for example the CD. The character '/' in a path name indicates a Unix-based file structure.



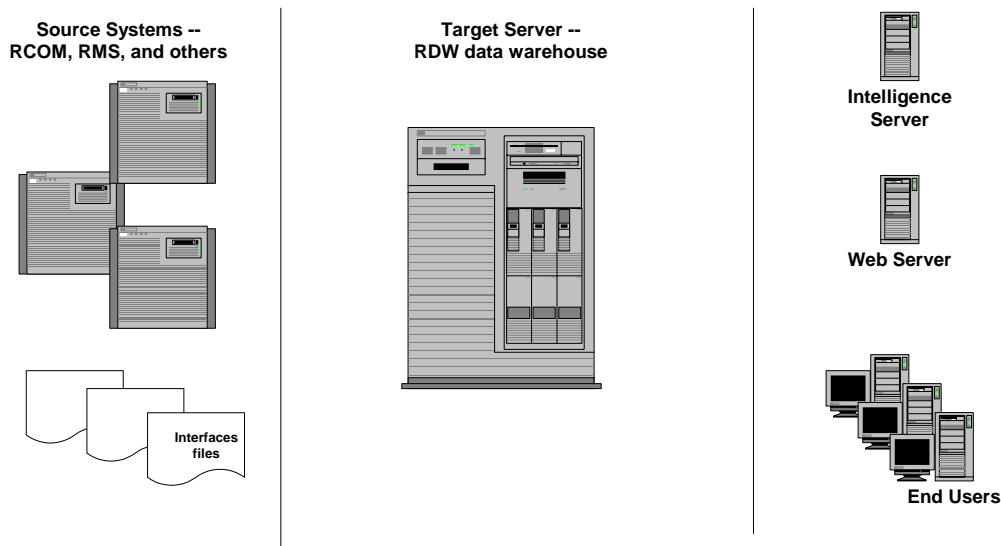


# Chapter 2 – Data Warehouse server and system requirements

## Data warehouse server (target server)

Retek Data Warehouse (RDW) version 11.0 architecture supports a database independent platform. RDW supports three database platforms: Oracle, DB2 or Teradata. This release of RDW incorporates optional interfaces with Retek Merchandising System (RMS), Retek Sales Audit, Retek Invoice Matching (ReIM), Retek Customer Order Management (RCOM), Retek Top Plan and Retek Price Management (RPM), which can be sources for the data warehouse.. The data warehouse can also operate as a standalone product and be fed from other legacy systems. If you are using Retek applications as the source systems, follow the guidelines set forth in each applications documentation regarding the server requirements for the source server. The data warehouse server or RDW server is considered the target server. It is recommended that the source systems be on a separate server from the data warehouse.

### RDW Architecture on Oracle and DB2 (see chapter 6 for Teradata architecture)



These are the hardware and software requirements necessary to install and operate the Retek Data Warehouse (RDW) software.

General requirements for a data warehouse server capable of running RDW include:

- Unix based OS
- TCP/IP Network
- Standard ANSI C compiler

## Hardware requirements

Hardware requirements for the data warehouse database server or target server include:

General Hardware Requirements	Minimum	Recommended
Memory	8GB	12GB+
Multi-processors	4 – highest MHz	8+ – highest MHz
CD-ROM drive (either local or network)		
Disk Space	300 GB	Site specific (refer to your RDW sizing information)
Swap Space	Equal to physical RAM	2.5 times the physical RAM



**Note:** This reflects the amount of space needed to install the database software, RETL software, as well as the RDW software, and still maintain an acceptable amount of usage on the Unix file system. This may reflect external storage as well as internal, such as 50GB of internal space and a 250+ GB external RAID storage array.

Hardware requirements for the load server for Teradata implementations include:

General Hardware Requirements	Minimum
Memory	4GB
Multi-processors	4 – highest MHz
Disk Space	100 GB
Swap Space	Equal to physical RAM
Network Bandwidth	100Mbit/s at full duplex mode in a private domain



**Note:** This reflects the amount of space need to install the RDW software and RETL software and still maintain an acceptable amount of usage on the Unix file system. This may reflect external storage as well as internal, such as 50GB of internal space and a 250+ GB external RAID storage array.

Sizing Factors and other suggestions to factor into the selection of a data warehouse server or Teradata load server include:

- Concurrent front-end user base. More concurrent users will require more database processing power from the server.
- DASD connectivity. Typically, it is better if the fiber channel connectivity provides the maximum throughput to disk.
- Backup/Recovery requirements. Extra disks may be required for backup and recovery procedures.
- Server backplane speeds. Depending on the architecture of the server backplane (or comparable links), memory access and/or CPU utilization may be a factor in performance.
- Overall database size for the RDW. A capacity plan should be done for the database server and DASD requirements in general to assess how large the environment will need to be.



**Note:** These data warehouse server requirements should only be used as guidelines because they reflect the hardware used to run a small environment (approximately 200 gigabytes of data). Actual requirements can be somewhat smaller, or typically much larger, depending on the intended size of the RDW environment upon full implementation. Retek can assist with making these determinations by providing information on database size estimates, server architecture, and so on.

## Software requirements and configuration matrix

The RDW 11.0 Configuration Matrix shows the currently supported configurations at the time of release. If a required configuration that is not supported, verify with Retek Customer Support to see if the configuration is now supported.

Database*	Operating System - Target Database Server**	Operating System - Source Database Server	Operating System - Load Server	ETL tool
IBM DB2 Universal Database - Extended Enterprise Edition 7.2 (32-bit)	AIX 4.3.3	Optional interface with RMS, use guidelines recommended by RMS documentation.  If standalone source, no server requirements.		RETL for DB2.
NCR Teradata V2R4.0 or R4.1	WorldMark 5xxx series – UNIX WorldMark 4xxx series – UNIX S2x series – UNIX	Optional interface with RMS, use guidelines recommended by RMS documentation.  If standalone source, no server requirements.	Sun Solaris 8 HP-UX 11i AIX 4.3.3	RETL for Teradata. The RETL application will be on the load server along with the Teradata load utilities.

Database*	Operating System - Target Database Server**	Operating System - Source Database Server	Operating System - Load Server	ETL tool
Oracle RDBMS Enterprise Edition with Partitioning Option 9.2 (64-bit)	Sun Solaris 9 HP-UX 11.11 AIX 5L	Optional interface with RMS, use guidelines recommended by RMS documentation.  If standalone source, no server requirements.		RETL for Oracle.



## Notes:

\*Review the database documentation for minimum guidelines and requirements.

\*\* Other configurations may exist call Retek Customer Support for currently supported configurations. Future availability will depend on demand and testing required prior to support.

Other software requirements for the data warehouse include: Please refer to the *RDW 11.0 Middle Tier Installation Guide* for the full installation instructions of the MicroStrategy components.

- MicroStrategy Intelligence Server 7.5.2
- MicroStrategy Web 7.5.2
- MicroStrategy Desktop 7.5.2
- MicroStrategy Administrator 7.5.2
- RETL 11.2.1

Retek product interfaces to the data warehouse include: Please refer to Chapter 6 – RDW Interfaces of the *RDW 11.0 Operations Guide* for more details about each interface.

- RMS 11.0
- ReSA 11.0
- RCOM 11.0
- Top Plan
- ReIM 11.0
- RPM 11.0

## Capacity planning

If you have not yet estimated the minimum disk space for your Retek Data Warehouse, contact the Retek Sizing team at [sizings@rettek.com](mailto:sizings@rettek.com), to request a RDW Database Sizing Questionnaire kit. When you have completed the Questionnaire return it to the Retek Sizing Team for an estimate on the minimum disk space required for your Retek Data Warehouse. The typical turnaround time for this estimate is 48 hours or 2 business days. Refer to your RDW sizing information to determine tablespace, table, index, and sizes as appropriate for planning your production environment. The database size created by the install scripts in this document is based on a typical development environment.

Retek Sizing can provide detailed Sizing information, iterative sizings and assistance in determining Technical Infrastructure requirements, on a Time & Materials basis, either remotely or onsite when arranged through the Retek Program Manager.

## Client system requirements

General requirements for end-user client workstation software requirements include:

- See specific database platform documentation for client requirements.
- See the Retek Data Warehouse Middle Tier Installation Guide for other client requirements.
- X-term station or application (such as Exceed) for Oracle installation. Oracle 9.2.x requires an X-term for GUI installation.



# Chapter 3 – RDW Unix environment setup

## Installation instructions

### Create Unix user accounts

- 1 Log in as the `root` user.
- 2 Create the following groups:
  - `dba` - admin group which controls RDW access
  - `dev` - development group
- 3 Create the following user accounts, using `ksh` as the default shell:

- `retex` - `dba` group
- `rdw11dev` - `dev` group

The following users are only needed for DB2 installations:

- `rdw11dm` – `dev` group (DB2 only)
- `rdw11md` – `dev` group (DB2 only)
- `rdw11sys` – `dev` group (DB2 only)



**Note:** These steps must be completed exactly as shown, since these user accounts are referenced later in installation scripts. To successfully run the RDW modules, environment variables are needed that are set in the profiles of these Unix user accounts. These profiles are set in a later step.

- 4 Identify the user that will own the database software. This user will be referred to as *rdwdba*.
  - DB2 – DB2 instance owner, typically `db2inst1`
  - Oracle – Oracle software and database owner, typically `oracle`
  - Teradata – Teradata database owner

### Create Unix directories

Create Unix directories for the RDW database structure and the code directory structure.



**Note:** The database directories are used to provide scripts to aid in the creation of tablespaces and users. The code directory structure holds the source code, error and log directories. Refer to Appendix C for further descriptions of the code directories.

- 1 Create RDW database directory structure:
  - a Extract the files from the downloaded zip file.
  - b Log in to the **RDW Database server** as *rdwdba* account, identified in the step above. This user will own the database directory structure.
  - c FTP (in binary mode) the tar file, *rdw11\_db.tar.Z*, to *<database\_directory>* on the Unix server where the *rdw11* database will be created. The *<database\_directory>* directory is a working directory that you should designate at this time.
  - d Uncompress and extract the tar file to create RDW database admin directories in the *<database\_directory>*. As the *rdwdba* user, extract the *rdw11\_db.tar* file by typing:

```
%uncompress rdw11_db.tar.Z
%tar -xvf rdw11_db.tar
```

Once expanded, the directory structure looks as follows:

```
<database_directory>/rdw11/create
                        /ddl/md/ora
                        /db2/full
                        /ter
                        /dm
```

- 2 Create RDW code directory structure:
  - a Log in to the **RDW database server** as *retrek*. Retek will own the code directory structure.
  - b FTP (in binary mode) the file *rdw11\_code.tar.Z* to a base directory where you wish to permanently store the RDW code directory. This base directory should be on the Unix server where the RDW 11.0 database will be created. The base directory is a working directory that you should designate at this time.
  - c Uncompress and extract the tar file to create the RDW code directory. As the *retrek* user, extract the *rdw11\_code.tar.Z* file in the base directory specified above by entering:

```
%uncompress rdw11_code.tar.Z
%tar -xvf rdw11_code.tar
```



Once expanded, the directory structure looks as follows:

```
<base_directory>/rdw11.0/dbasql
                        /dev/batch
                        /data
                        /error
                    /install
                    /log
                    /rfx/bin
                        /bookmark
                    /etc
                        /include
                        /lib
                        /schema
                        /src
                /retek/sample_profiles
```



**Note:** We highly recommend that clients have a separate server that is for production only. When the time comes for a production environment, create a prd directory, then copy the dev directory structure to your production server and use the <base\_directory>/rdw11.0/prd as your code directory structure on that server. A prd group is needed and a separate prd user should be created as well.

## Alter directory ownership and privileges

For added security, alter the directory ownership and privileges.

- 1 Log in as the root user (in the <base\_directory>/rdw11.0 directory).
- 2 Change the following as indicated in the table below:
  - the ownership (chown -R <owner> <directory>)
  - the group (chgrp -R <user group> <directory>)
  - the privileges (chmod -R <privs> <directory>)

Directory	Owner	Group	Privilege
retek	retek	dba	775
dev	rdw11dev	dev	775
dbasql	rdwdba -Db2, Oracle, Teradata	dba	774

For example: `cd <base_directory>/rdw11.0`  
`chown -R rdw11dev dev`  
`chgrp -R dev dev`  
`chmod -R 775 dev`

### Copy profiles for Retek users

For each of the Retek users, copy the profiles.

- 1 At the Unix prompt, change directories to  
`<base_directory>/rdw11.0/retrek/sample_profiles`.  
  
In this directory, you can find a sample of the profile that needs to be set up for each Unix user account.
- 2 Copy the profile to the home directory of each user. For example, for a Teradata installation, the `rdw_profile_td` file should be copied to `.profile` in the home directory of the users that were created. Edit the `.profile` to correspond with your file system, RETL installation and Unix environment.
- 3 Copy the profile to the `rdw11dev`, `rdw11md`, and `rdw11dm` users.
- 4 When the time comes to create a test and/or production schema, create `rdwtst`, `rdwprd` and other Unix users. A copy of the corresponding development user's profile is often a good place to start.



**Note:** The `sample_profiles` by default are set up for a development environment. Change the indicated variables as necessary for other environments. Any variable that may require modification are noted in the `sample_profile` comments. Be sure to set up these Unix accounts to automatically run the `.profile` within their home directory upon login.

Continue to the appropriate chapter for your database platform.

# Chapter 4 – RDW Oracle RDBMS installation and configuration

## General installation notes

This release of RDW incorporates an optional interface with Retek Merchandising System (RMS), Retek Sales Audit (ReSA), Retek Invoice Matching (ReIM), Retek Customer Order Management (RCOM), Retek Top Plan and Retek Price Management (RPM).



**Note:** You should become familiar with the Retek Data Warehouse development environment before setting up a production system.

## Oracle Concepts for the Retek Data Warehouse

Oracle supports multiple partitioning options. Partitioning is a strategy that you should consider for your compressed datamarts to enhance performance. To take full advantage of the Oracle partitioning feature, you should partition large historical tables. To determine an appropriate partitioning strategy, consider the database size and business requirements. For example, the amount of history to be held at various levels, and the various functional areas that might be used in RDW should be referenced when determining a partitioning strategy. Large non-compressed fact tables can be partitioned for ease of rolling off history. Before creating the RDW datamart tables in a production environment, determine your partitioning strategy. See the RDW Operations Guide to fully understand the Partitioning strategy for both compressed and non-compressed fact tables that can be used when implementing the data warehouse on an Oracle platform.

## Installation instructions

### Oracle database configuration

The following instructions are recommended for the development and test environments only. Refer to your RDW sizing information to determine tablespace, table, index, and sizes as appropriate for planning the production environment. The installation scripts provided will need to be modified with your production environment for your tablespace requirements, table and index storage parameters, and so on.

The scripts to create the RDW database, users, and tablespaces are in **<database\_directory>/rdw11/create** directory which was created in Chapter 3 Unix environment setup.

- Create an Optimal Flexible Architecture (OFA) compliant directory structure to store all Oracle files before creating the RDW database. The files have already been moved to the Unix server and untarred in a previous step.

### Set up Parameter file

- 1 Before creating the rdw11 database, the parameter file must be set up. A sample parameter file is provided.

See the file `<database_directory>/rdw11/create/initrdw11.ora` for guidance on setting up the Oracle database parameters. This file should serve as a guideline for setting up the rdw11 database. It contains unformatted notes; therefore, it will not be valid to start a database without modification first.

- a Modify the `nls_language` parameter to suit your environment. Some examples of languages are: American, French, and Japanese.
  - b Modify the `nls_territory` parameter to suit your environment. Some examples of territories are: America, France, and Japan.
  - c Modify remaining parameters to suit your environment.
- 2 Once this file has been updated and placed in the proper directory, alter and run the actual database creation scripts.

### Create database, tablespaces, and users

- 1 Log in to the **RDW database server** as Oracle. Change directories to `<database_directory>/rdw11/create`.
- 2 Verify your Unix session is set up correctly for creating an Oracle database. Refer to the Oracle Documentation for the correct settings.
- 3 Modify the sample database creation scripts listed below to contain the appropriate Unix file system paths, database name, number and size of redo logs, undo tablespace name and size, temporary tablespace name and size, and user account passwords.
- 4 The script `crusers_ora.sql` can then be used to create the default RDW development Oracle users and grant the appropriate system privileges. Refer to the Development Database Schema Implementation section in Appendix C for descriptions of each user.

- `crdb_ora.1`
- `crdb_ora.2`
- `crtbspc_ora.sql`
- `crusers_ora.sql`



**Note:** These scripts assume that the RDW database name is `rdw11`, so change the path names and datafile names to reflect the actual RDW database name if necessary.

The default sizing of the tablespaces in these create scripts assume they are being used to set up a development environment

- 5 At a Unix prompt enter:  

```
% ./run_crdb_ora.sh
```
- 6 A log file will be produced for each script. Review the log files for errors, if the script executed properly, continue with the database object creation. If not, make the appropriate corrections and rerun the necessary portions. The database creation scripts may take up to 30 minutes to complete.

- 7 To secure the passwords in the create users script, change the privileges on the script by entering:

```
%chmod 700 crusers_ora.sql
```

### Database connectivity

Verify that you have connectivity and the necessary Net Services components on both the server and client machines. Samples are located in the *Sample Oracle Net Services File for the Server* section in Appendix A.

### Create RDW Datamart-level tables

- 1 Change directories to <database\_directory>/rdw11/ddl/dm.
- 2 Log in to SQL\*Plus as rdw11dm and execute the following script:

```
SQL>@rdw11dm_ora.sql
```

This script creates the RDW datamart-level tables. It will create all default base non-partitioned tables. If you are implementing a partitioning strategy at this time, modify the DDL to reflect your partitioning.

- 3 Review the rdw11dm\_ora.lst file to verify that the objects were all created successfully.

### Install MicroStrategy metadata

In order to run RDW reports from MicroStrategy, a metadata layer is needed in the database. The metadata tables hold information about reports, filters, metrics, and so on. The tables for MicroStrategy have already been set up and exist in export files. The steps below outline how to install the MicroStrategy repository including all RDW reporting objects. This approach will install all objects with the Oracle import utility, which is faster for an initial installation of all four RDW Workbenches.

- 1 Change directories to  
<database\_directory>/rdw11/ddl/md/ora.
- 2 Set your NLS\_LANG environment variable to match your language and character set. For example for an English installation your NLS\_LANG should be AMERICAN\_AMERICA.UTF8.
- 3 The script, import\_rdw11md.sh will import the metadata objects using an export file. At a Unix prompt enter: ./import\_rdw11md.sh
- 4 At the prompt enter the username where the metadata should be created, rdw11md.
- 5 At the prompt choose the appropriate language for your implementation.
- 6 Enter the username and password to execute the import. We recommend you use system or oracle as the user names for the import.
- 7 Review the rdw11md\_ora.log file for errors and make appropriate corrections if needed and rerun.
- 8 Run the grants and synonyms scripts for the rdw11md user outlined below.

### Set up RDW user permissions

- 1 Change directories on the Unix server to <base\_directory>/rdw11.0/dbasql.

This directory contains scripts that will set up all necessary synonyms and grants for the RDW environment.

The header of each file contains information regarding which user should execute the script, when to run them and additional details about the scripts.

The scripts will spool to the /tmp directory, which should already exist, but verify that this directory exists and can be written to before running.

- 2 The rdw\_users.sql script has been set up to reference the default RDW usernames, so verify that they are correct and change to suit your environment if necessary.
- 3 When prompted by any of the scripts for an object name, press the **[Enter]** key to create permissions for all objects in the schema.
- 4 Connect to SQL\*Plus as rdw11dm and execute the following script to setup permissions to the datamart objects:

```
SQL>@gs_rdw11_dm_ora.sql
```

```
SQL>@invalids.sql
```

- 5 Verify the scripts executed correctly by reviewing the following files:

```
g_rdw11_dm.lst, s_rdw11_dm.lst
```

- 6 Connect to SQL\*Plus as rdw11md and execute the following script to setup permissions to the MicroStrategy database objects:

```
SQL>@gs_rdw11_md_ora.sql
```

- 7 Verify the scripts executed correctly by reviewing the following files:

```
g_rdw11_md.lst, s_rdw11_md.lst
```

Ignore any text that is generated similar to the following.

```
SP2-0734: unknown command beginning "new          7:
..." - rest of line ignored
```

Ignore any text that is generated from invalids.sql if no objects need to be validated.

```
no rows selected
```

```
SP2-0734: unknown command beginning "no rows se..." - rest of line
ignored.
```



**Note:** These scripts are also used for maintenance. If you need to drop or add a table in the database, run these scripts again to give the appropriate privileges to your users. If a table has been dropped and recreated for maintenance, you also need to verify that all objects in the database are **VALID**. For a test or a production environment, you need to modify the rdw\_users.sql script to alter the usernames for the environment you are creating.

### Configure RETL

- 1 Log in to the Unix server with the `rdw11dev` account.
- 2 Change directories to `<base_directory>/rdw11.0/dev/rfx/etc`.
- 3 Modify the `rdw_config.env` script to match your environment:
  - a Change the `DBNAME` variable to the name of RDW database.
  - b Verify the `DB_ENV` variable is set to `ORA`.
  - c Change the `DM_OWNER` variable to the username of the RDW Data Mart schema owner.
  - d Change the `BA_OWNER` variable to the username of the RDW batch user.
  - e Change the `LOAD_TYPE` to direct or conventional based on the requirements for SQL Loading. Refer to the Operations Guide Chapter Program Flow Diagrams for more details.
  - f Change the `LANGUAGE` variable to the appropriate two-letter language code for your environment. For an English installation the `LANGUAGE` variable should be set to `en`.
- 4 Continue to Chapter 7, Default Data Population.





# Chapter 5 – RDW DB2 RDBMS installation and configuration

## General installation notes

This release of RDW incorporates an optional interface with Merchandising System (RMS), Retek Sales Audit (ReSA), Retek Invoice Matching (ReIM), Retek Customer Order Management (RCOM), Retek Top Plan and Retek Price Management (RPM).



**Note:** You should become familiar with the Retek Data Warehouse development environment before setting up a production system.

Minimum requirements for DB2 database partitions and other configuration options, depending on transaction throughput desired:

Hardware	Minimum	Recommendations are site specific
Multi-processors	8	12+
DB2 Database Partitions	4 nodes	6+
Memory	8 GB	12GB+

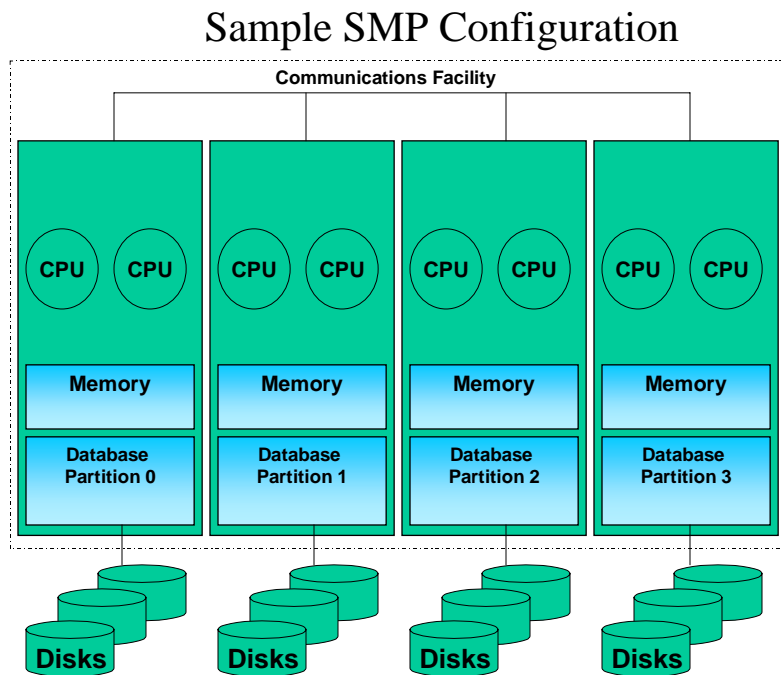


**Note:** The minimum recommended memory is 1 GB of memory per CPU. You most likely will see significant performance benefits with more than 1 GB of memory per CPU. All production sites are unique.

## DB2 Concepts for the Retek Data Warehouse

The default DB2 installation will assume the minimum configuration of eight CPUs, four database partitions, and eight GB of memory running on a symmetric multi-processor (SMP) server. All installation scripts reflect this configuration. If you have a configuration other than the default one, modify the scripts to suit your environment.

Sample configuration on SMP:

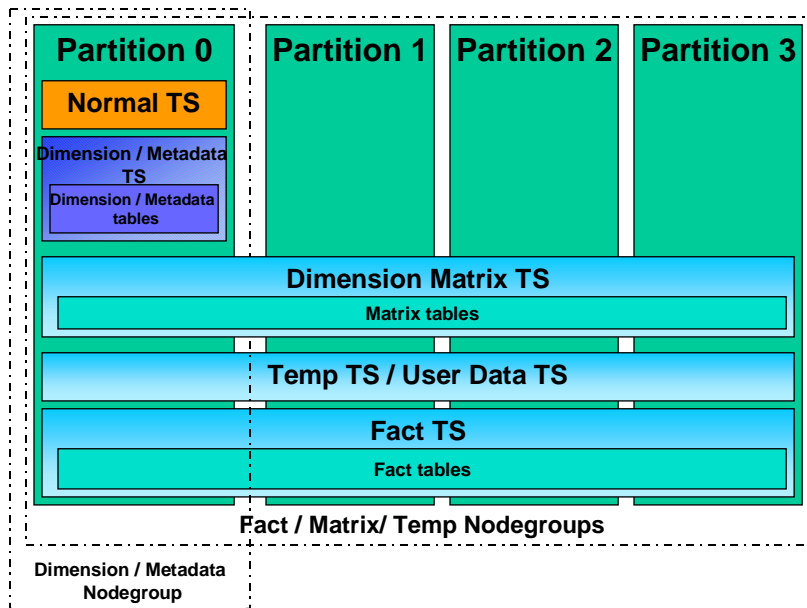


As an alternative the following should be considered and carefully planned for your data warehouse environment: additional CPUs, partitions, memory, multiple symmetric multi-processor -SMP servers clustered or massively parallel processing – MPP server. At this time an MPP configuration is not supported. If you are considering an MPP configuration, contact Retek for an estimated timeline for future support of MPP architecture.

Tablespace placement should also be considered before creating your production system. This is a typical tablespace layout on four partitions with two node groups. By adding partitions, you can gain parallel processing power and scalability. The dimension and metadata tablespaces and tables are located in the first partition. The dimension matrix, fact, temp, and user tablespaces and tables are spread across all the nodes. By spreading your large fact tables across all your partitions, you can perform your queries in parallel across all the partitions defined in that group. All installation scripts are based on this default configuration. If you have a different configuration, modify the installation scripts to suit your environment.

Default tablespace configuration:

### Default tablespace layout



## Installation instructions

### DB2 database configuration

We recommend the following instructions for development and test environments only. Refer to your RDW sizing information to determine tablespace, table, index, and sizes as appropriate for planning the production environment. The installation scripts provided will need to be modified with your production environment for your tablespace requirements, table and index storage parameters, and so on.

#### Create database and objects (tablespaces/containers, users, and so on)

The scripts to create the RDW database, users, and tablespaces/containers are in Unix subdirectories of the `<database_directory>/rdw11/create` directory. The files to create your database base directory structure have already been moved to the Unix server and untarred in chapter 3.

The scripts used to create and maintain the RDW database must be executed exactly the same as indicated in the instructions. Dynamic scripts, \*.dyn, create and execute SQL scripts, \*.sql. These SQL scripts are generated from the data dictionary and contain the commands to be run against the database. The output of the SQL scripts is written to a \*.log file.

### Create database, tablespaces and users

- 1 Modify the sample database creation scripts. The following scripts will need to be modified to contain the appropriate Unix file system paths, database name, number and size of logs, and tablespace size.

These scripts create a four-node database with the default tablespaces and buffer pools. The default sizing of the tablespaces in these create scripts assume they are being used to set up a development environment.

The `crusers_db2.sql` script can then be used to create the default RDW development users and grant the appropriate system privileges. Refer to the Development Database Schema Implementation section for descriptions of each user.

Several database configuration parameters will be modified in this script. Use these parameter settings as a baseline and modify as needed for your specific environment. The parameter settings are also listed as a guideline in the Appendix B section. The scripts are located in `<database_directory>/rdw11/create`.

- `crdb_db2.sql`
- `crtbspd_db2.sql`
- `crtbspi_db2.sql`
- `upd_cfg_db2.sh`
- `crusers_db2.sql`
- `crtbsp_privs_db2.dyn`



**Note:** These scripts assume that the RDW database name is `rdw11`, so change the path names to reflect the actual database name if desired.

- 2 Log in as the DB2 instance owner. Change directories to `<database_directory>/rdw11/create`. At the Unix prompt enter:

```
% ./run_crdb_db2.sh
```

A log file will be produced for each script. The log naming convention is `<script_name>.log`. Check the logs and verify there were no errors generated. If the script executed properly, continue with the database object creation. If not, make the appropriate corrections and rerun the necessary portions. The database creation scripts may take up to 30 minutes to complete.

- 3 Verify the database configuration parameters are set correctly using the following command for each node.

```
%db2 get db cfg for rdw11 | more
```

### Create RDW Datamart-level tables

- 1 Change directories to <database\_directory>/rdw11/ddl/dm.

- 2 At a Unix prompt enter:

```
%db2 connect to rdw11 user rdw11dm
```

```
%db2 -tvf rdw11dm_db2.sql > rdw11dm_db2.log
```

This script creates the RDW tables, views, primary keys, constraints and indexes. It creates all fact tables with a default partitioning key defined the same as the primary key. The partitioning key can be modified to suit your environment.

- 3 Review the rdw11dm\_db2.log file to verify that the objects were all created successfully.

### Install MicroStrategy Metadata

In order to run RDW reports from MicroStrategy, a metadata layer is needed in the database. The metadata tables hold information about reports, filters, metrics, and so on. The tables for MicroStrategy have already been set up and exist in the following export files. The steps below outline how to install the MicroStrategy repository including all RDW reporting objects. This approach will install all objects with the DB2 import utility, which is faster for an initial installation of all four RDW Workbenches.

- 1 Change directories to <database\_directory>/rdw11/ddl/md/db2/full.
- 2 The import script will call subsequent files in the same directory and create the metadata objects.

#### File to import

```
<database_directory>/rdw11/ddl/md/db2/full/rdw11md_db2.sql
```

```
<database_directory>/rdw11/ddl/md/db2/full/mstr7_db2.trg
```

- 3 Execute the following commands to start the import.

```
%db2 connect to rdw11 user rdw11md
```

```
%db2 -tf rdw11md_db2.sql > rdw11md_db2.log
```

```
%db2 -tvf mstr7_db2.trg > mstr7_db2.log
```

- 4 Review the log files for errors and make appropriate corrections if needed and rerun the necessary portions.
- 5 Run the grants and synonyms scripts for the rdw11md user outlined below.

### Set up RDW user permissions

- 1 Change directories on the Unix server to <base\_directory>/rdw11.0/dbasql.

This directory contains scripts that set up all necessary synonyms/aliases and grants for the RDW environment.

- The header of each file contains information regarding which user should execute the script, when to run them, and additional details about the scripts.
- All scripts will create a SQL file with the necessary SQL statements, then the SQL file will be executed and output to a log file. The command syntax must be exactly as shown below.

- 2 Log in to the Unix server with the `rdw11dm` account.
- 3 All scripts have been set up to reference the default RDW usernames, so verify that they are correct and change to suit your environment if necessary.
- 4 Connect to `rdw11` as `rdw11dm` and execute the following script to grant permissions to the datamart objects:  

```
%db2 connect to rdw11 user rdw11dm  
%db2 -tf gs_rdw11_dm_db2.dyn > gs_rdw11_dm_db2.sql
```
- 5 Review the log file, `gs_rdw11_dm_db2.log`, to verify the script executed successfully.
- 6 Log in to the Unix server with the `rdw11md` account.
- 7 Connect to `rdw11` as `rdw11md` and execute the following script to grant permissions to the MicroStrategy metadata objects:  

```
%db2 connect to rdw11 user rdw11md  
%db2 -tf gs_rdw11_md_db2.dyn > gs_rdw11_md_db2.sql
```
- 8 Review the log file, `gs_rdw11_md_db2.log`, to verify the script executed successfully.



**Note:** These scripts are also used for maintenance. If you need to drop or add a table in the database, you need to run these scripts again to give the appropriate privileges to your users. If a table has been dropped and recreated for maintenance, you also need to verify that all objects in the database are valid. For a test or a production environment, you need to modify these scripts to alter the usernames for the environment you are creating.

### Configure RETL

- 1 Log in to the Unix server with the `rdw11dev` account.
- 2 Change directories to `<base_directory>/rdw11.0/dev/rfx/etc`.
- 3 Modify the `rdw_config.env` script to match your environment:
  - a Change the `DBNAME` variable to the name of RDW database.
  - b Change the `DB_ENV` to `DB2`.
  - c Change the `DM_OWNER` variable to the username of the RDW Data Mart schema owner, `RDW11DM`.
  - d Change the `BA_OWNER` variable to the username of the RDW batch user, `RDW11DEV`.
  - e Change the `LOAD_TYPE` to direct or conventional based on the requirements for data Loading. Refer to the Operations Guide Chapter Program Flow Diagrams for more details.
  - f Change the `SCHEDULE_TYPE` to sequential or parallel based on your batch processing schedule. Refer to the Operations Guide Chapter Program Flow Diagrams for more details.
  - g Change the `LANGUAGE` variable to the appropriate two-letter language code for your environment. For an English installation the `LANGUAGE` variable should be set to `en`.
- 4 Continue on to Chapter 7, Default Data Population.

# Chapter 6 – RDW Teradata RDBMS installation and configuration

## General installation notes

This release of RDW incorporates an optional interface with Retek Merchandising System (RMS), Retek Sales Audit (ReSA), Retek Invoice Matching (ReIM), Retek Customer Order Management (RCOM), Retek Top Plan and Retek Price Management (RPM)

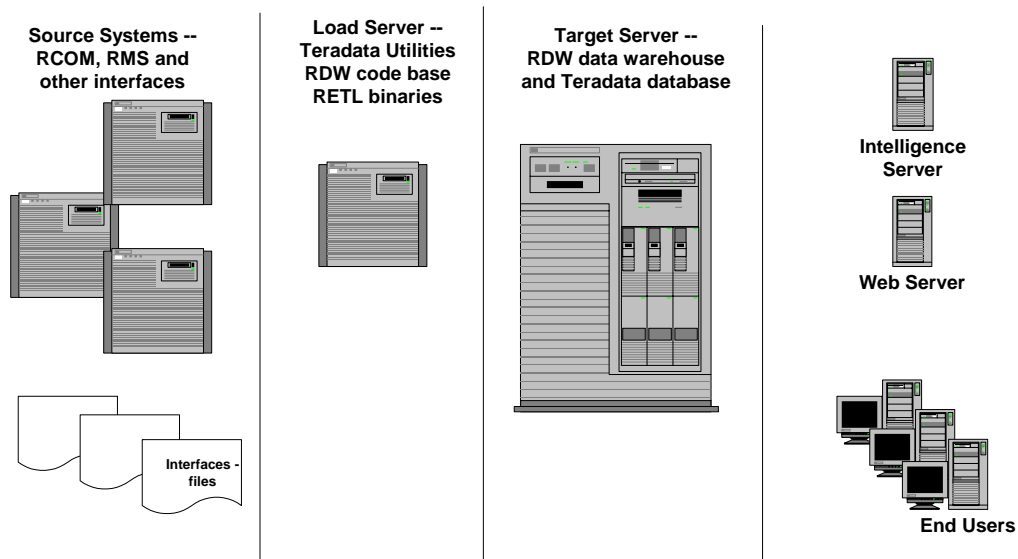


**Note:** You should become familiar with the Retek Data Warehouse development environment before setting up a production system.

## Teradata Concepts for the Retek Data Warehouse

The Teradata environment has another component in addition to the source and target servers. A load server will be part of the environment, which will hold the Teradata Utilities, RDW code base, and the RETL; and will connect to the source and target server via ODBC and CLI connections. The Teradata server is the target server and is where the RDW data warehouse is located.

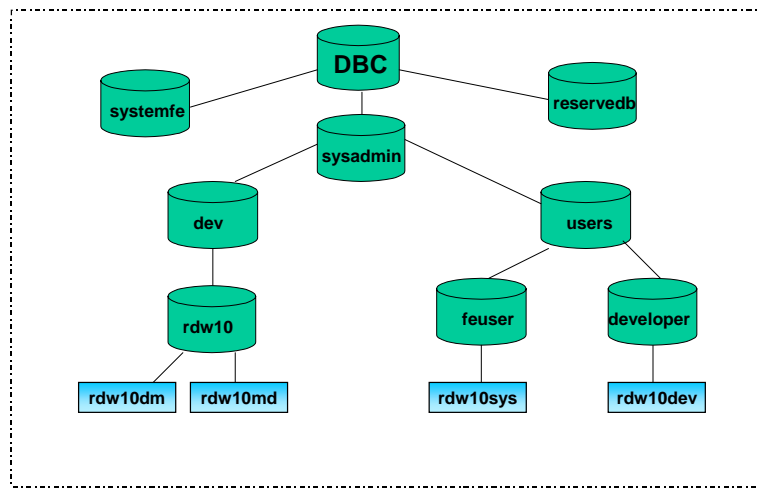
### RDW Architecture on Teradata



This installation assumes that the sysadmin and systemfe have already been created. Teradata also recommends creating a reservedb, which contains approximately 15 to 20% of the database space for backup spool space.

Default RDW configuration:

Sample RDW Teradata Database Configuration



## Installation instructions

### Teradata database configuration

We recommend the following instructions for development and test environments only. Refer to your RDW sizing information to determine table, index, and sizes as appropriate for planning the production environment.

#### Create database and objects

The scripts to create the RDW database and users are in the `<database_directory>/rdw11/create` directory which was created in an earlier step.

- The files to create your database base directory structure have already been moved to the Unix server and untarred in chapter 3.

#### Create database and users:

- 1 Log in to Unix as the Teradata user specified in Chapter 3. Change directories to `<database_directory>/rdw11/create`.
- 2 Modify the sample database creation scripts `<database_directory>/rdw11/create/crdb_td.sql` to contain the appropriate database name, perm and spool size. The default sizing of the databases in these create scripts assume they are being used to set up a development environment. These scripts can then be used to create the default RDW development users and grant the appropriate system privileges. Refer to the Development Database Schema Implementation section for descriptions of each user. These scripts assume that the RDW database name is `rdw11`, modify if necessary.



- 3 Modify the logon in the script, `run_crdb_td.sh`, to include the correct password for dbc.

```
%bteq <run_crdb_td.sh> run_crdb_td.log
```

A log file will be produced – `run_crdb_td.log`. Check the log and verify there were no errors generated. If the script executed properly, continue with the database object creation. If not, make the appropriate corrections and rerun only the necessary portions.

### Create RDW Datamart-level tables

- 1 Change directories to `<database_directory>/rdw11/ddl/dm`.

- 2 Execute the script as the `rdw11dm` database user.

```
%bteq .logon rdw11dm,password <rdw11dm_td.sql> \ rdw11dm_td.log
```

This script creates the RDW datamart-level tables, views, primary keys, constraints and indexes.

- 3 Review the `rdw11dm_td.log` file to verify that the objects were all created successfully.

### Install MicroStrategy Metadata

In order to run RDW reports from MicroStrategy, a metadata layer is needed in the database. The metadata tables hold information about reports, filters, metrics, and so on. The tables for MicroStrategy have already been set up and exist in an export file. This section will outline how to install the MicroStrategy repository including all RDW reporting objects. This approach will install all objects with the Teradata copy utility, which is faster for an initial installation of all four RDW Workbenches.

- 1 Change directories to `<database_directory>/rdw11/ddl/md/ter`.
- 2 Modify the script, `rdw11md_full_td.arc`, to include the Teradata server name and password. The script has been set up to reference the default RDW usernames, change to suit your environment if necessary.

#### Files to import

```
<database_directory>/rdw11/ddl/md/ter/rdw11md_full_td.arc
```

```
<database_directory>/rdw11/ddl/md/ter/mstr7_td.trg
```

- 3 Execute the following commands to start the import.

```
%arcmain < rdw11md_full_td.arc > rdw11md_td.log
```

```
%bteq .logon rdw11md,password < mstr7_td.trg > \ mstr7_td.log
```

- 4 Review the log files for errors and make appropriate corrections if needed and rerun.
- 5 Run the grants scripts for the `rdw11md` user outlined below.

### Set up RDW user permissions

- 1 Change directories on the Unix server to <base\_directory>/rdw11.0/dbasql.  
This directory contains scripts that set up all necessary grants for the RDW environment.
  - The header of each file contains information regarding which user should execute the script, when to run them, and additional details about the scripts.
- 2 All scripts have been set up to reference the default RDW usernames, so verify that they are correct and change to suit your environment if necessary.

- 3 At a Unix Prompt, enter:

```
%bteq .logon rdw11dm,password < \  
gs_rdw11_dm_td.dyn> gs_rdw11_dm_td.log  
%bteq .logon rdw11md,password < \  
gs_rdw11_md_td.dyn > gs_rdw11_md_td.log
```

- 4 Review the log files for errors and make appropriate corrections if needed and rerun the necessary portions.



**Note:** These scripts are also used for maintenance. If you need to drop or add a table in the database, you need to run these scripts again to give the appropriate privileges to your users. For a test or a production environment, you need to modify these scripts to alter the usernames for the environment you are creating.

### Configure RETL

- 1 Log in to the Unix server with the rdw11dev account.
- 2 Change directories to <base\_directory>/rdw11.0/dev/rfx/etc.
- 3 Modify the rdw\_config.env script to match your environment:
  - a Change the DBNAME variable to the name of Teradata server.
  - b Change the DB\_ENV to TER.
  - c Change the DM\_OWNER variable to the username of the RDW Data Mart schema owner, RDW11DM.
  - d Change the BA\_OWNER variable to 'DEVELOPER', the default database for the RDW batch user.
  - e Change the LANGUAGE variable to the appropriate two-letter language code for your environment. For an English installation the LANGUAGE variable should be set to en.
- 4 Continue on to Chapter 7, Default Data Population.

# Chapter 7 – Default data population

## ETL installation and configuration

This release of RDW incorporates the RETL. Take this time now to install and configure this on the target server. See the RETL Programmers Guide to install and configure this tool. Once configured, populate the default data as described in the rest of this chapter.

## Data population

Some of the RDW tables require one or more default records before setting up the batch cycle.

Log in to a Unix session as `rdw11dev`. If you are setting up a test or production environment, log in with the appropriate Unix account – `rdw11tst` for the test system, `rdw11prd` for the production system.

If the account is set up with the proper profile information in the `.profile`, the environment variables will be set up by logging into the appropriate Unix user. The key environment variables for running RETL programs are: `$MMUSER`, `$PASSWORD`, `$PATH`, `$RFX_HOME` and `$MMHOME`.



**Note:** All SQL scripts, error and log files are encoded UTF-8. Any changes or modifications to these SQL scripts should be made with your UNIX session set to UTF8. Refer to your operating system documentation for more details on your session settings.

### Populate datamart default data

- 1 Log in to a Unix session on the **RDW database server** as `rdw11dev` and run the `rdw` profile created in Chapter 3. For creating the default data you need to change the `MMUSER` and `PASSWORD` variables to the data mart owner, **`rdw11dm`** and the appropriate password. The `MMUSER` variable should be set to `rdw11dev` for Teradata. Verify the RETL executable is in the path of your Unix session by typing:

```
%which rfx
```

- 2 Change directories on the Unix server to `<base_directory>/rdw11.0/dev/install`. This directory contains the code for all default data installation scripts.

- 3 At the Unix prompt, enter:

```
%default_load.ksh
```



**Note:** Read the comments on every install scripts called within `default_load.ksh` specifically `load_maint_dim_key_dm.sql` because the initial date for `curr_load_dt` might be different for every client.

This script executes the following SQL scripts:

- `alter_dt_fmt.sql` - alters the current session's date format (Oracle only).
- `load_program_control_dm.sql` - populates the program status data for the RDW modules.
- `load_dummyrows.sql` - populates necessary 'dummy' records for low level dimensions.
- `load_maint_dim_key_dm.sql` - populates the dimension surrogate key information and initial maintenance date.
- `load_indicator_dm.sql` - populates the default values for a dummy table for front-end reporting.
- `load_cde_dtl_dm.sql` - populates the code abbreviation table for RDW batch modules to reference.
- `load_prod_diff_type_dm.sql` - populates the diff type dimension with default diff type values.
- `load_rtl_type_dm.sql` - populates the retail type dimension table with default retail types.
- `load_cde_dtl_com_dm.sql` - populates the customer order code abbreviation table for RDW batch modules to reference.
- `load_tsf_type_dm.sql` - populates the transfer type dimension table with default transfer types.

4 Change directories on the Unix server to:

`<base_directory>/rdw11.0/dev/log`. Review the log file that was created or modified.

5 Change directories on the Unix server to

`<base_directory>/rdw11.0/dev/error`. Review the error files that were created for default\_load.

5 Change directories on the Unix server to

`<base_directory>/rdw11.0/dev/install`.

6 Rerun the RDW profile and verify the MMUSER and PASSWORD variables are set to the batch user, rdw11dev and the appropriate password. At the Unix prompt, enter:

```
%time_static_load.ksh
```

7 `time_static_load.ksh` - populates the following tables with default values:

- `time_minute_dm`
- `time_hour_dm`
- `time_half_hour_dm`
- `time_qtr_hour_dm`

- 8 Change directories on the Unix server to  
`<base_directory>/rdw11.0/dev/log`. Review the log file that was created or modified.
- 9 Change directories on the Unix server to `<base_directory>/rdw11.0/dev/error`. Review the error files that were created for `time_static_load`.
- 10 Refer to the RDW Operations Guide for more information on the log files and error files.

## Populate time dimension

This section describes the initial load and loading of additional time into RDW. The time dimension can be loaded with a 454 calendar, 13 period time calendar or a 454 with Gregorian calendar. The client must populate these tables according to the client's specifications and business requirements. To populate the time dimension with 454-calendar time or 454 with Gregorian calendar, for clients using RMS, the calendar information can be extracted from RMS. See the RMS Operations Guide for details of how to extract time. After the calendar information has been extracted, proceed to section B 'Create Time in RDW' to load it in RDW. To populate time for 454, 13 period or 454 with Gregorian calendar standalone, proceed to section A 'Create Standalone Time'. For information on the tables loaded for the Time dimension refer to the RDW Data Model documentation.

### A. Create Standalone time

Modify the sample text files that will serve as the data to populate the time calendar dimension with appropriate time for your environment. If you are implementing a 454-time calendar or a 454-time calendar with Gregorian calendar, modify the `time_454*.txt`. If you are implementing a 13 period time modify the `time_13.txt` file with your time calendar data. Modify the file, `wkday*.txt`, with the description of the first day of the fiscal week, for example 'SUNDAY'. The entry in `wkday.txt` must be in all capital letters. A sample of these text files has been provided in `<base_directory>/rdw11.0/dev/install`. See the RDW Operations Guide Appendix A 'API Flat File Specifications' for instructions on how to modify the text files. Proceed to step B 'Create Time in RDW' to load time into the Data Warehouse.



**Note:** You can use the sample text files to get started in a development environment. These files need to be modified to match your fiscal calendar before creating time in your production environment

### B. Create Time in RDW

If you need to add more time data, modify the parameters below for the new time period and run the module again. To minimize the load time for adding additional time, enter only the first year to be added as the response for the first year prompt in number 5.

- 1 If you are not using RMS as the source of your time calendar, move on to step 2, otherwise after running time extraction in RMS (see RMS Operations Guide for details), FTP `time_454.txt`, `start_of_half_month.txt`, and `wkday.txt` from RMS install directory to the RDW directory `<base_directory>/rdw11.0/dev/install`.
- 2 Login to a Unix session on the **RDW database server** as `rdw11dev` and run the RDW profile. Verify the RETL executable is in the path of your Unix session by typing:  
`%which rfx`
- 3 Change directories to `<base_directory>/rdw11.0/dev/install`.

- 4 Verify the C compiler is in the path of your Unix session and the C compiler is in your Unix library path. At the Unix prompt, enter:

```
%which cc
```

- 5 Compile the module `cr_time_454`, `cr_time_13` and `cr_time_g` with a standard ANSI C compiler. At the Unix prompt, enter:

```
%cc -g -I. -o cr_time_454 c_utils.c cr_time_454.c
```

```
%cc -g -I. -o cr_time_13 c_utils.c cr_time_13.c
```

```
%cc -g -I. -o cr_time_g c_utils.c cr_time_g.c
```

- 6 Execute the `cr_time.ksh` module. At the Unix prompt enter:

```
%cr_time.ksh
```

This script will prompt for your calendar type. Choose 1 for 454 time, 2 for 13 period time and 3 for 454 with Gregorian time.

At the prompt enter the 4-digit year for the beginning and ending of your time calendar:

```
%Please enter first year to be loaded:
```

```
%Please enter last year to be loaded:
```



**Note:** To determine your beginning and ending fiscal year, refer to the text file modified above. Verify all months or periods are included in the text file for the first year; no partial years are allowed.

One text file will be generated in the install directory for each dimension table.

- 7 At the Unix prompt, enter:

For 454 time calendar or 13 period time calendar:

```
%time_load.ksh
```

```
%time_trnsfrm_load.ksh
```

For 454 time with Gregorian time calendar:

```
%time_load.ksh
```

```
%g_time_load.ksh
```

```
%time_trnsfrm_load.ksh
```

```
%g_time_trnsfrm_load.ksh
```

- 8 Change directories on the Unix server to `<base_directory>/rdw11.0/dev/log`. Review the log file that was created or modified.
- 9 Change directories on the Unix server to `<base_directory>/rdw11.0/dev/error`. Review the error files that were created for `time_load` and `time_trnsfrm_load`. Review also the error files that were created for `g_time_load` and `g_time_trnsfrm_load` if 454 time with Gregorian calendar is used.
- 10 Refer to the RDW Operations Guide for more information on the log files and error files.



**Note:** `DAY_IDNT` serves as a surrogate key for day of the time. It can represent 454 time, 13 period time or Gregorian time. Since it is a surrogate key, it does not have meaning to it as long as it is unique across all the days of the time used in RDW.

## Populate static dimension data

There are several dimension tables in RDW that must to be loaded once before loading Voucher Facts and executing some MicroStrategy Reports. In order to determine the information needed for these tables, refer to the RDW Operations Guide, 'API Flat File Specifications'. Sample flat files have been provided in the <base\_directory>/rdw11.0/dev/install directory, but the client must populate these tables according to individual specifications and business requirements.

The tables to be populated are time\_last\_yr\_by\_day\_lfl\_dm, time\_last\_yr\_by\_wk\_lfl\_dm, vchr\_age\_band\_dm. See the RDW Data Model documentation for more information on the tables and the column descriptions to be loaded.

- 1 Login to a Unix session on the **RDW database server** as rdw11dev. Verify the RETL executable is in the path of your Unix session by typing:

```
%which rfx
```

- 2 Change directories on the Unix server to <base\_directory>/rdw11.0/dev/install. This directory contains the code for all default data install scripts.

- 3 The script timelfldm.ksh loads the following Time 'Like for Like' tables that are used exclusively by the MicroStrategy reports:

```
time_last_yr_by_day_lfl_dm
time_last_yr_by_wk_lfl_dm
```

Load the Time 'Like for Like' tables by executing the script. At the Unix prompt, enter:

```
%timelfldm.ksh
```

- 4 Change directories on the Unix server to <base\_directory>/rdw11.0/dev/log. Review the log file that was created or modified.
- 5 Change directories on the Unix server to <base\_directory>/rdw11.0/dev/error. Review the error files that were created.
- 6 Change directories on the Unix server to <base\_directory>/rdw11.0/dev/install. The script, vchragedm.ksh, loads the table, vchr\_age\_band\_dm, which is used by the voucher fact load batch module. At the Unix prompt, enter:
 

```
%vchragedm.ksh
```
- 7 Change directories on the Unix server to <base\_directory>/rdw11.0/dev/log. Review the log file that was created or modified.
- 8 Change directories on the Unix server to <base\_directory>/rdw11.0/dev/error. Review the error files that were created.
- 9 Refer to the RDW Operations Guide for more information on the log files and error files.





# Appendix A – Oracle information

## Sample Oracle Net Services File for the Server

### Net Services Setup

- 1 Install the necessary Net Services components on both the server and client machines. If Net Services already exists and is operational on the server, manually alter the files  
`$ORACLE_HOME/network/admin/listener.ora` and  
`$ORACLE_HOME/network/admin/tnsnames.ora` according to the Oracle 9.2 documentation.
- 2 Check to see whether the Net Services listener is running by logging into Unix as the user `oracle` and then issuing the following command:  
`%lsnrctl stat`
- 3 If the listener is running, stop it by issuing the following command:  
`%lsnrctl stop`
- 4 Start the listener by issuing the following command:  
`%lsnrctl start`

### Modify the client-side `tnsnames.ora` file

- 1 Edit the `$ORACLE_HOME\network\admin\tnsnames.ora` file on the client to match the settings on the Unix server.
- 2 After you have altered this file on one PC and have tested the connectivity, copy this file to other PCs, if necessary.

The `tnsnames.ora` file for client PCs may be on a LAN. In this case, it is not necessary to copy the file to each PC.

Samples of a typical `tnsnames.ora` file is below.

```
retek =
  (DESCRIPTION=
    (ADDRESS=(PROTOCOL=tcp)(host=<IP address>)
      (Port=1521))
    (CONNECT_DATA=(SERVICE_NAME = retek)
  )
)
```



## Appendix B – DB2 information

### Sample database configuration parameters on DB2

Use the following parameters as a starting point for DB2 databases running RDW. The parameters can be modified for your data warehouse and your environment.



**Note:** There may be differences per specific environments, as these are the parameters specifically used with DB2 version 7.2.

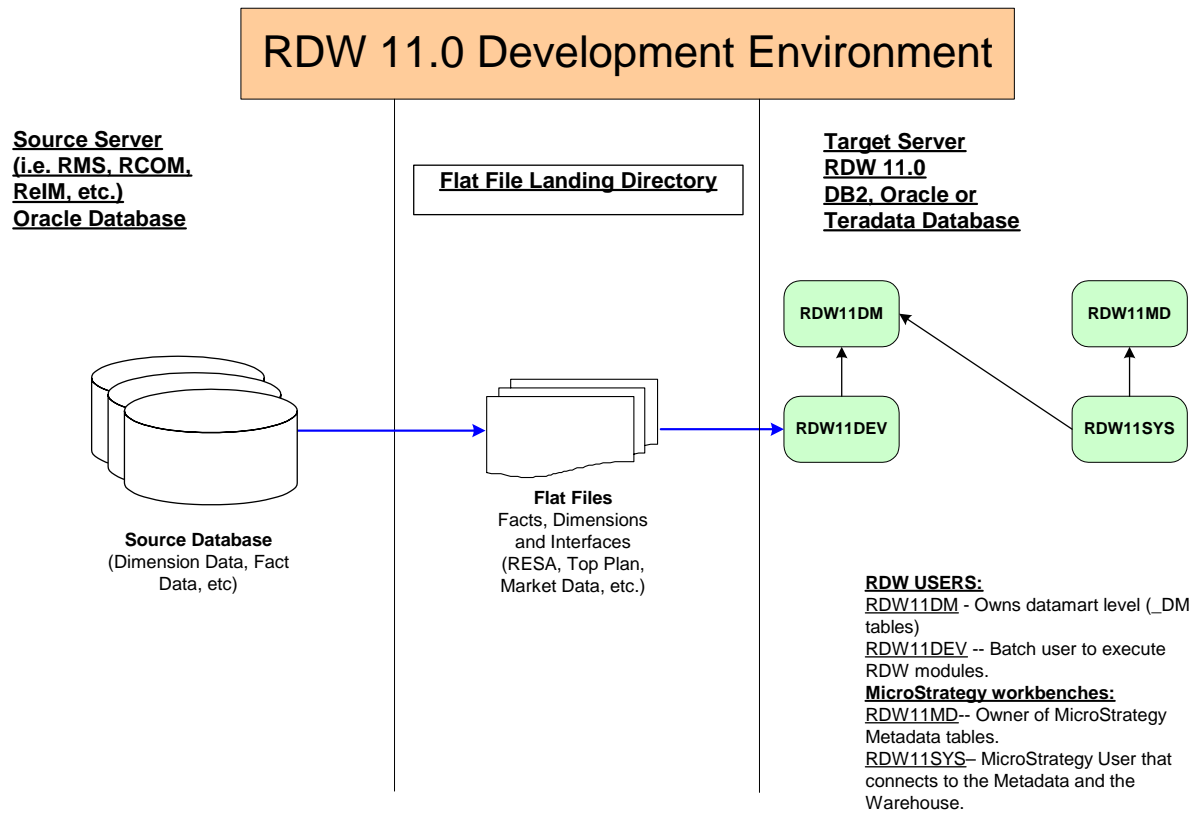
Parameter	Value
BUFFPAGE	60000 (general guideline: 20000 * # of nodes)
CATALOGCACHE_SZ	512
CHNGPGS_THRESH	80
DBHEAP	8000
LOGBUFSZ	512
LOGFILSIZ	10000
LOGPRIMARY	10
LOGSECOND	50
LOCKLIST	1024
MAXAPPLS	100
MAXFILOP	250
NEWLOGPATH	Designate a separate log path for each node.
NUM_IOCLEANERS	12 (general guideline: # of CPUs)
NUM_IOSERVERS	12
SOFTMAX	700
SORTHEAP	1024
APPLHEAPSZ	32768
MIN_DEC_DIV_3	YES



# Appendix C – Additional information

## Development Database Schema implementation

The following figure shows the source system and RDW development schemas that are referred to in this installation guide.



## RDW Code Tree

The following table describes the contents of each of the RDW code tree directories created during the RDW installation.

Path	Directory	Description
<base_directory>	dev	Directory that contains all code, error, log, etc subdirectories. The environment variable \$MMHOME is set to point here.
<base_directory>	dbasql	This directory contains all SQL scripts necessary to maintain the permissions for the database users.
<base_directory>	rettek	The subdirectory, sample_profiles contains the sample unix profiles needed to set up the unix accounts for batch execution. The environment variable \$RETEK_HOME is set to point to this directory.
<base_directory>dev	batch	Empty directory used for development and testing purposes only.
<base_directory>dev	data	This directory contains the text files that serve as the input to RDW RETL load modules. For clients with Retek applications such as RMS, RCOM, etc, these text files can be generated in these source applications. The text files should be FTP from the source application to the RDW data directory. Directory is empty on installation.
<base_directory>dev	error	This directory holds all program error files, and status files. Directory is empty on installation.
<base_directory>dev	install	This directory contains all RETL modules and SQL scripts needed only at installation.
<base_directory>dev	log	This directory holds log files of program execution. Directory is empty on installation.
<base_directory>/dev	rfx	This directory contains subdirectories for all the code and files related directly to RETL
<base_directory>dev/rfx	bin	Currently not used.
<base_directory>dev/rfx	bookmark	This directory contains a file created during execution of each module to track the execution of the module. Files are deleted upon successful completion of module. Directory is empty on installation.
<base_directory>dev/rfx	etc	This directory contains files that hold variables used by RDW batch modules. The configuration file is found in this directory.
<base_directory>dev/rfx	include	This directory contains files that hold string language translations used by RDW batch modules.

Path	Directory	Description
<base_directory>dev/rfx	lib	This directory contains all RDW library code
<base_directory>dev/rfx	schema	This directory contains all RDW schema files used with each module.
<base_directory>dev/rfx	src	This directory contains RDW source code.