Retek® Merchandising System™ 11.0.2 French

Installation Guide

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@retek.com
Internet (ROCS)	rocs.retek.com 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

Chapter 1 –Database Installation Instructions

Before you apply the RMS 11.0.2 patch:

• Make a backup of all your objects and database schema.

• Check that RMS 11.0 French is installed.

• This release contains code for both RMS 11.0.1 and 11.0.2 French

• Review the enclosed RMS 11.0.2 Patch Release Notes (rms-1102-rn.pdf).

• Review each of the enclosed SIR documents.

• The rsm folder in staging area/pricing1/ will be used during the RSM 11 installation.

Before copying over any files:

• Note whether customizations have been made to the module. If so, then the customizations must be reapplied over the new version of the module (or the fix may need to be applied to the custom version of the code).

• Copy the original files to a different directory before copying over them in case they need to be referred to at a later date.

  Note: These instructions refer to RMS11DEV as the Oracle owning schema.

Mount CD-ROM on the Database Server

1 Copy the rms1102dbpatch.zip file from the CD /dbserverunix directory to a newly created staging directory on your UNIX server.

2 Unzip the file by entering:

```
unzip rms1102dbpatch.zip
```

Update RIB Objects for RMS

  Note: The following directories are included but not used

CastorPayloadTyped - Contains typed, serialized java beans representing message families. A configuration file (payload.properties) maps each bean to a specific message family/message type.

CastorPayloadUntyped - Contains untyped, serialized java beans representing message families. Each class member is represented by a String (as opposed to the data type the member represents). A configuration file (payload.properties) maps each bean to a specific message family/message type.

Retek_Pub_Trans - Contains a class that maps an oracle object to an XML formatted string for every family represented in a database by an oracle object. Each translator handles all message types within a single family

Create RIB tables and types

1   Change directories to staging area/rib_objects1101/xml

2   Log into sqlplus as RMS11DEV and run the following command:

    SQL> @rms1101xml.sql

3   Check the log file rms1101xml.log for any errors noting that ORA-04043 errors and warnings are to be ignored.

Create RIB Objects

1   Change directories to staging area/rib_objects1101/Oracle_Objects.

2   Log into sqlplus as RMS11DEV and run the following command:

    SQL> @rms1101rib.sql

3   Check the log file rms1101rib.log for any errors noting that ORA-04043 errors are to be ignored.

Loading RIB data

1   Change directories to staging area/rib_objects1101/xml.

2   Run the following command at the UNIX prompt:

    sqlldr RMS11DEV/SCHEMA_PASSWORD control=rib_doctypes_rms.ctl

3   Check the log file rib_doctypes_rms.log for any errors.

Update RMS types

1   Change directories to staging area /xml1

2   Log into sqlplus as RMS11DEV and run the following command:

    SQL> @patch1101xml.sql

3   Check the log file patch1101xml.log for any errors.

Partitioning for new tables

The following four tables are being added and need to be partitioned:

- DEAL_ACTUALS_ITEM_LOC

- DEAL_ITEM_LOC_EXPLODE

- DEAL_ITEMLOC

- RPL_NET_INVENTORY_TMP

These files get created in the script add_partitioning.sql in staging area/dbcs.

Partitioning can be added by either modifying the staging area/dbcs/part/partition_attributes.cgf file and running the staging area/dbcs/part/partition.ksh script or by copying the staging area/dbcs/add_partition.sql to staging area/dbcs/add_partition_final.sql and modifying add_partition_final.sql to include you partitioning.

Update RMS tables

  Note: Some triggers will give compilation warnings. These are OK. The warnings are caused by dependencies. The warnings will be cleared when objects are revalidated later in the install.

1 Change directories to staging area/dbcs1

2 Log into sqlplus as RMS11DEV and run the following command:

  ```SQL> @patch1101dbcs.sql```

3 Check the log file patch1101dbcs.log for any errors.

Update RMS database objects

1 Change directories to staging area /db_objects1

2 Log into sqlplus as RMS11DEV and run the following command:

  ```SQL> @patch1101rms.sql```

3 Check the log file patch1101rms.log for any errors.

Update data for RMS

  Note: This script will prompt for the schema owner.

1 Change directories to staging area /sqlplus1

2 Log into sqlplus as RMS11DEV and run the following command:

  ```SQL> @patch1101ctl.sql```

3 Check the log file patch1101ctl.log for any errors.

Update additional Pricing xml

1 Change directories to staging area/pricing1/xml

2 Log into sqlplus as RMS11DEV and run the following command:

  ```SQL> @pricing1101xml.sql```

3 Check the log file pricing1102xml.log for any errors.

Update Pricing tables

1 Change directories to staging area/pricing1/dbcs

2 Log into sqlplus as RMS11DEV and run the following command:

  ```SQL> @pricing1101dbcs.sql```

3 Check the log file pricing1102dbcs.log for any errors.

Update Pricing database objects

1 Change directories to staging area /pricing1/db_objects

2 Log into sqlplus as RMS11DEV and run the following command:

```
SQL> @pricing1101dbo.sql
```

3 Check the log file pricing1102dbo.log for any errors.

Update data for Pricing

1 Change directories to staging area /pricing1/sqlplus

2 Log into sqlplus as RMS11DEV and run the following command:

```
SQL> @pricing1102ctl.sql
```

3 Check the log file pricing1102ctl.log for any errors.

Update additional RIB Objects for RMS

    Note: The following directories are included but not used

CastorPayloadTyped - Contains typed, serialized java beans representing message families. A configuration file (payload.properties) maps each bean to a specific message family/message type.

CastorPayloadUntyped - Contains untyped, serialized java beans representing message families. Each class member is represented by a String (as opposed to the data type the member represents). A configuration file (payload.properties) maps each bean to a specific message family/message type.

Retek_Pub_Trans - Contains a class that maps an oracle object to an XML formatted string for every family represented in a database by an oracle object. Each translator handles all message types within a single family

Create additional RIB Objects

1 Change directories to staging area/rib_objects1102/Oracle_Objects.

2 Log into sqlplus as RMS11DEV and run the following command:

```
SQL> @rms1102rib.sql
```

3 Check the log file rms1102rib.log for any errors noting that ORA-04043 errors are to be ignored.

Loading additional RIB data

1 Change directories to staging area/rib_objects1102/xml.

2 Run the following command at the UNIX prompt:

```
sqlldr RMS11DEV/SCHEMA_PASSWORD control=rib_doctypes_rms.ctl
```

3 Check the log file rib_doctypes_rms.log for any errors.

Update additional RMS tables

    📖   **Note:** Some triggers will give compilation warnings. These are OK. The warnings are caused by dependencies. The warnings will be cleared when objects are revalidated later in the install.

1   Change directories to staging area/dbcs2

2   Log into sqlplus as RMS11DEV and run the following command:

    `SQL> @patch1102dbcs.sql`

3   Check the log file patch1102dbcs.log for any errors.

Update additional RMS database objects

1   Change directories to staging area /db_objects2

2   Log into sqlplus as RMS11DEV and run the following command:

    `SQL> @patch1102rms.sql`

3   Check the log file patch1102rms.log for any errors.

Update additional data for RMS

1   Change directories to staging area /sqlplus2

2   Log into sqlplus as RMS11DEV and run the following command:

    `SQL> @patch1102ctl.sql`

3   Check the log file patch1102ctl.log for any errors.

Update additional Pricing xml

1   Change directories to staging area/pricing2/xml

2   Log into sqlplus as RMS11DEV and run the following command:

    `SQL> @pricing1102xml.sql`

3   Check the log file pricing1102xml.log for any errors.

Update additional Pricing tables

1   Change directories to staging area/pricing2/dbcs

2   Log into sqlplus as RMS11DEV and run the following command:

    `SQL> @pricing1102dbcs.sql`

3   Check the log file pricing1102dbcs.log for any errors.

Update additional Pricing database objects

1   Change directories to staging area /pricing2/db_objects

2   Log into sqlplus as RMS11DEV and run the following command:

    `SQL> @pricing1102dbo.sql`

3   Check the log file pricing1102dbo.log for any errors.

Update additional data for Pricing

1   Change directories to staging area /pricing2/sqlplus

2   Log into sqlplus as RMS11DEV and run the following command:

    ```
    SQL> @pricing1102ctl.sql
    ```

3   Check the log file pricing1102ctl.log for any errors.

Validate all invalid objects

   📖   **Note:** Deadlocked objects may appear when running this script. This is expected. Run the script until no more invalid objects remain.

1   Change directories to INSTALL_DIR/utility

2   Log into sqlplus as RMS11DEV and run the following command:

    ```
    SQL> @inv_obj_comp.sql
    ```

3   This script may need to be run more than once.

Update RETL

1   Change directories to staging area /retl/rfx/etc

2   Copy all the files from this directory INSTALL_DIR/retl/rfx/etc

    `cp * ` INSTALL_DIR/retl/rfx/etc

3   Change directories to staging area /retl/rfx/schema

4   Copy all the files from this directory INSTALL_DIR/retl/rfx/schema

    `cp * ` INSTALL_DIR/retl/rfx/schema

5   Change directories to staging area /retl/rfx/src

6   Copy all the files from this directory INSTALL_DIR/retl/rfx/src

    `cp * ` INSTALL_DIR/retl/rfx/src

Compile RMS batch libraries and programs

   📖   **Note:** Warning messages may appear during the compilation of the batch. These warnings can be ignored if the batch executables are successfully generated.

Setting Environment Variables

1   As the retek user,make sure the following variables are set:

    📖   **Note:** INSTALL_DIR is the location where RMS 11 was installed.

    📖   **Note:** Make sure the path for make, makedepend, and the compiler are in $PATH environment variable.

- MMHOME=INSTALL_DIR/rms
- MMUSER=RMS Schema Owner
- PASSWORD=RMS Schema Owner Password
- ORACLE_HOME=Location of Oracle install
- ORACLE_SID=The Oracle Sid for the RMS database

AIX only:

- LIBPATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$LDPATH
- OBJECT_MODE=64
- LINK_CNTRL=L_PTHREADS_D7

HP only:

- SHLIB_PATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$SH_LIBPATH

Solaris only:

- LD_LIBRARY_PATH=$ORACLE_HOME/lib: $MMHOME/oracle/lib/bin:$LD_LIBRARY_PATH

2   Copy the files from staging are/batch/lib/src to INSTALL_DIR/rms/oracle/lib/src

3   Change directories to INSTALL_DIR/rms/oracle/lib/src and run the following commands

4   To make library dependencies

```
make –f retek.mk depend 2>&1 | tee libdpnd.log
```

5   Check the libdpnd.log file for errors

6   To make batch libraries

```
make –f retek.mk retek rms resa 2>&1 | tee libretek.log
```

7   Check the libretek.log file for errors

8   To install batch libraries

```
make –f retek.mk install
```

9   The batch libraries should now be in INSTALL_DIR/rms/oracle/lib/bin

10 Copy the file resa2rdw from INSTALL_DIR/rms/oracle/lib/src to INSTALL_DIR/rms/oracle/lib/bin

11 Copy the files from staging are/batch/proc/src to INSTALL_DIR/rms/oracle/proc/src

12 Change directories to INSTALL_DIR/rms/oracle/proc/src and run the following commands

13 To make dependencies

   make –f mts.mk rms-depend recs-depend rtm-depend resa-depend 2>&1 | tee srcdpnd.log

14 Check the srcdpnd.log file for errors

15 To make batch programs

   Because of an additional make command the following command must be run first

   ```
   make –f rms.mk PRODUCT_PROCFLAGS=dynamic=ansi ditinsrt
   ```

   To make the rest of the batch programs run the following command

   ```
   make –f mts.mk rms-ALL recs-ALL resa-ALL rtm-ALL 2>&1 | tee
   srcall.log
   ```

16 Check the srcall.log file for errors

17 To install batch programs

   ```
   make –f mts.mk rms-install recs-install resa-install rtm-install
   ```

18 The batch programs should now be in INSTALL_DIR/rms/oracle/proc/bin

Chapter 2 – Data Warehouse interface (DWI) UNIX environment setup

Installation instructions

The Retek Merchandising System (RMS) Extracts (for RDW) is a module released with RMS 11.0.2 that allows this release of the RMS to interface with the Retek Data Warehouse (RDW). The interface extracts must be installed on the same server and in the same database as RMS.

This code also contain code for the RPM and ReIM interface to RDW

Create Unix user accounts

1   Log in as the `root` user.

2   Create Unix groups for the following environments:

- dba - admin group which controls RMS Extracts (for RDW) access

- dev - development group

3   Create the following Unix users, using ksh as the default shell:

- retek - dba group

- dwidev - dev group

  Note: These steps must be completed exactly as described, since these Unix accounts are referenced later in installation scripts. To successfully run the DWI batch modules, environmental variables are needed and are set in the profiles of these Unix user accounts. These profiles are set in a later step.

Create Unix directories

Create Unix directories for the code directory structure.

  Note: The code directory structure stores source code, error and log directories. Refer to Appendix for further descriptions of the code directories.

1   Create the RMS Extracts (for RDW) code directory structure:

a   Copy the file, `dwi11_code.tar.Z,` from staging area/retlforRDW to a base directory, the permanent location for the RMS Extraction (RDW) code directory , on the Unix server where the RMS database is located. The base directory is a working directory that you should designate at this time. This base directory is only for extracts to RDW and should be different with other base directories that will be used for extracts to other Retek products.

b   Uncompress and extract the tar file to create RMS Extracts (RDW) code directory. As the `retek` user, extract the dwi11.tar file in the base directory by typing:

```
%uncompress dwi11_code.tar.Z
%tar -xvf dwi11_code.tar
```

Once expanded, the directory structure looks as follows:

```
<base_directory>/dwi11.0/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 the production server and use the `<base_directory>/dwi11.0/prd` as the 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

1   Log in as the `root` user in the `<base_directory>/dwi11.0` directory.

2   Change the following as indicated in the table below:

   ▪   the ownership (chown –R <owner> <directory>)

   ▪   the group (chgrp –R <user group> <directory>)

   ▪   and the privileges (chmod –R <privs> <directory>)

Directory	Owner	Group	Privilege
retek	retek	Dba	775
dev	dwidev	Dev	775
dbasql	oracle	Dba	774

For example: cd <base_directory>/dwi11.0
            chown –R dwidev 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, enter
    `%cd <base_directory>/dwi11.0/retek/sample_profiles`

    In this directory is 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, `dwi_profile` should be copied to `.profile` in the home directory of the `dwidev` user that was created.

    📖   **Note:** The `dwi_profile` by default is set up for a development environment. Change the indicated variables as necessary for other environments, such as dwitst or dwiprd. Any variable that may require modification is 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.

Configure RETL

1   Log in to the Unix server with the `dwidev` account.

2   Change directories to `<base_directory>/dwi11.0/dev/rfx/etc`.

3   Modify the dwi_config.env script to match your environment:

    a   Change the DBNAME variable to the name of RMS database.

    b   Change the RPM_OWNER and RIM_OWNER variables to the username of the RPM table owner and ReIM table owner.

    c   Verify the DB_ENV variable is set to ORA.

    d   Change the RMS_OWNER variable to the username of the RMS table owner.

    e   Change the BA_OWNER variable to the username of the RMS batch user.

    f   Change the LOAD_TYPE to direct or conventional based on the requirements for SQL Loading.

    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`.

Chapter 3 – Application Server Installation Instructions

    📖   **Note:** INSTALL_DIR is the directory where the RMS 11.x files were extracted to. 9iAS10G_ORACLE_HOME is the location where Oracle 9iAS 10g Forms and Reports Services (9iAS 10g) was installed.

Mount CD-ROM on the Database Server

1   Copy the rms1102apppatch.zip file from the CD /appserverunix directory to a newly created staging directory on your UNIX server.

2   Unzip the file by entering:

```
unzip rms1102apppatch.zip
```

Setup

1   As the retek user, set the DISPLAY variable to the IP address plus ":0.0" (ie: 10.1.1.1:0.0) of the machine that is being used to perform the compilation from.

2   As the retek user, set the following variables:

    📖   **Note:** INSTALL_DIR is the location where RMS 11 was installed.

    📖   **Note:** 9iAS10G_ORACLE_HOME is the location where Oracle 9iAS 10g was installed.

ORACLE_HOME=9iAS10G_ORACLE_HOME

PATH=$ORACLE_HOME/bin:INSTALL_DIR/forms9i_scripts:$PATH

Solaris only:
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/jdk/jre/lib/sparc:$ORACLE_HOME/jdk/jre/lib/sparc/native_threads

HP-UX only:
SHLIB_PATH=$ORACLE_HOME/lib32:$ORACLE_HOME/lib:$ORACLE_HOME/jdk/jre/lib/PA_RISC:$ORACLE_HOME/jdk/jre/lib/PA_RISC/server

AIX only:
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/lib32:$ORACLE_HOME/jdk/jre/lib

```
LIBPATH=$LD_LIBRARY_PATH
```

All:
CLASSPATH=$ORACLE_HOME/jlib/debugger.jar:$ORACLE_HOME/jlib/utj90.jar:$ORACLE_HOME/jlib/ewt3.jar:$ORACLE_HOME/jlib/share.jar

FORMS90_BUILDER_CLASSPATH=$CLASSPATH

FORMS90_PATH=INSTALL_DIR/toolset/bin:INSTALL_DIR/rms/forms/bin:$ORACLE_HOME/forms90

REPORTS_PATH=INSTALL_DIR/rms/reports/bin:$ORACLE_HOME/forms90

Solaris/AIX only:
UP=<RMS schema owner>/<RMS schema password>@<RMS database>

HP-UX only:
UP=<RMS schema owner>/<RMS schema password>\@<RMS database>

Toolset

1   Copy all the files from staging area/toolset/src to INSTALL_DIR/toolset/src

2   Copy all forms (*.fmb files) in the INSTALL_DIR/toolset/src directory to the INSTALL_DIR/toolset/bin directory.

3   Run fmb2fmx9i_fm (in INSTALL_DIR/toolset/bin) to compile the Toolset reference forms.

4   Remove all newly created fm_*.fmx files (reference forms should not have executable files).

5   Run fmb2fmx9i (in INSTALL_DIR/toolset/bin) to generate Toolset runtime forms – .fmx's.

6   Check to make sure that each non-reference form (.fmb file) has a corresponding .fmx file. If a form fails to compile (there is no .fmx file), it will have to be manually compiled with Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions.

    📖   **Note:** Disregard fm_*.fmx files should they be created. These files should be removed. They should NOT exist in the INSTALL_DIR/toolset/bin directory.

7   Remove all non-reference form forms from INSTALL_DIR/toolset/bin; the following syntax will leave all reference forms (fm_*.fmb) in the bin directory, while removing all other forms:

```
> for PROG in `ls *.fmb | grep -v fm_`
> do PROGNAME=`echo $PROG`
> rm $PROGNAME
> done
```

8   Copy all menus (*.mmb files) in the INSTALL_DIR/toolset/src directory to the INSTALL_DIR/toolset/bin directory.

9   Run mmb2mmx9i (in INSTALL_DIR/toolset/bin) to generate Toolset runtime menus – .mmx's.

10   Check to make sure that each .mmb file has a corresponding .mmx file. If a menu fails to compile (there is no .mmx file), it will have to be manually compiled with Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions.

    📖   **Note:** Should .err files be created by the compilation scripts above, these files are logs of the compilation process and can be removed.

11   Remove all .mmb files from INSTALL_DIR/toolset/bin.

Forms

1   Copy all the files from staging area/forms/src to INSTALL_DIR/rms/forms/src

2   Copy all libraries (.pll files) in the INSTALL_DIR/rms/forms/src directory to the directories to the INSTALL_DIR/rms/forms/bin directory.

3   Change directories to INSTALL_DIR/rms/forms/bin.

4   Run pll2plx9i_forms to compile all RMS .pll's.

5   Check to make sure that each .pll file has a corresponding .plx (to ensure that all .pll's compiled successfully). If a library fails to compile (there is no .plx file), it will have to be manually compiled with Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions

6   Remove all newly created .plx files.

7   Copy all forms (*.fmb files) in the INSTALL_DIR/rms/forms/src directory to the INSTALL_DIR/rms/forms/bin directory.

8   Run fmb2fmx9i_fm (in INSTALL_DIR/rms/forms/bin) to compile the RMS reference forms.

9   Remove all newly created fm_*.fmx files (reference forms should not have executable files).

10 Run fmb2fmx9i (in INSTALL_DIR/rms/forms/bin) to generate RMS runtime forms – .fmx's.

11 Check to make sure that each non-reference form .fmb file has a corresponding .fmx file. If a form fails to compile (there is no .fmx file), it will have to be manually compiled with Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions.

    📖   **Note:** Disregard fm_*.fmx files should they be created. These files should be removed. They should NOT exist in the INSTALL_DIR/rms/forms/bin directory.

12 Remove all non-reference form forms from INSTALL_DIR/rms/forms/bin; the following syntax will leave all reference forms (fm_*.fmb) in the bin directory, while removing all other forms:

```
> for PROG in `ls *.fmb | grep -v fm_`
> do PROGNAME=`echo $PROG`
> rm $PROGNAME
> done
```

13 Copy all menus (*.mmb files) in the INSTALL_DIR/rms/forms/src directory to the INSTALL_DIR/rms/forms/bin directory.

14 Run mmb2mmx9i (in INSTALL_DIR/rms/forms/bin) to generate RMS runtime menus – .mmx's.

15 Check to make sure that each .mmb file has a corresponding .mmx file. If a form fails to compile (there is no .mmx file), it will have to be manually compiled with Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions.

16 Remove all .mmb files from INSTALL_DIR/rms/forms/bin.

    📖   **Note:** Should .err files be created by the compilation scripts above, these files are logs of the compilation process and can be removed.

Reports

1   Copy all the files from staging area/reports/src to INSTALL_DIR/rms/reports/src

2   Copy the reports library (rep25lib.pll) in the INSTALL_DIR/rms/reports/src directory to the INSTALL_DIR/rms/reports/bin directory.

3   Change directories to INSTALL_DIR/rms/reports/bin.

4   Run pll2plx9i_reports to compile rep25lib.pll. If rep25lib.pll fails to compile (there is no .plx file), it will have to be manually compiled with Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions

5   Remove the newly created rep25lib.plx file.

6   Copy all reports (*.rdf files) in the INSTALL_DIR/rms/reports/src directory to the INSTALL_DIR/rms/reports/bin directory

7   Run rdf2rep9i (in INSTALL_DIR/rms/reports/bin) to generate Reports runtime reports – .rep's.

  📖 **Note:** The following error messages may appear when running rdf2rep9i; these errors can be ignored if report (.rep) generation was successful:

   REP-0759: One or more PL/SQL libraries have been modified since the reports was saved. The PL/SQL will be recompiled.

   REP-0202: Attempt to free a null pointer

   REP-0759 is generated by the r25conv program. The error appears any time a report is converted.

   REP-0202 is due to an Oracle bug with rwconverter and can be ignored

8   Check to make sure that each .rdf file has a corresponding .rep file. If a report fails to compile (there is no .rep file), it will have to be manually compiled with Reports Builder in Oracle 9iDS 10g. See Appendix F of the RMS 11 Install Guide for manual compilation instructions.

9   Remove all .rdf files from INSTALL_DIR/rms/reports/bin.

  📖 **Note:** Should .err files be created by the compilation scripts above, these files are logs of the compilation process and can be removed.

Appendix A – RMS RETL instructions

This Appendix summarizes the RETL program features utilized in the RMS Extractions (RMS ETL). More information about the RETL tool is available in the latest RETL Programmer's Guide. More information about RMS ETL is available in the RMS ETL operations guide.

Configuration

RETL

Before trying to configure and run RMS ETL, install RETL version 11.0 or later which is required to run RMS ETL. Run the "verify_retl" script (included as part of the RETL installation) to ensure that RETL is working properly before proceeding.

RETL users and permissions

It is recommended that a Unix user is created for the installation and execution of the RMS RETL modules. This user must own a home directory in which the RMS RETL tar file will be extracted. The RMS RETL user must have the permissions to locate and execute the RETL tool executable. The RMS RETL user must also have the permissions to create directories and files, and to delete, write to, and execute files in the RMS RETL file structure. An existing Unix user may be used, as long as the aforementioned criteria are met.

In order to run the RMS RETL modules, it is also recommended that an RMS RETL-specific database user is created. This database user must have the permissions to connect to the RMS database via SQL*Plus, drop tables, create tables, analyze tables, and finally to update and insert into tables. An existing RMS database user may be used, as long as the aforementioned criteria are met.

RMS RETL installation

Log into the Unix server from which the RMS RETL extractions will be run. It is recommended that the RMS RETL modules be installed on the same Unix server that hosts the RMS database. FTP the RMS RETL tar file into the directory structure where the RMS RETL modules will be permanently stored (i.e. the home directory of the RMS RETL Unix user). Uncompress and extract the tar file as the RMS RETL user. The resultant directory structure is as follows :

```
<base_directory>/data/
        /error/
        /install/
        /log/
        /rfx/bin/
                /rfx/bookmark
        /rfx/etc/
        /rfx/include/
        /rfx/lib/
        /rfx/schema/
        /rfx/src/
```

Once the RMS RETL tar file has been uncompressed and extracted, ensure all files and directories have the required permissions for the RMS RETL user (please see the RETL users and permissions section).

Environment variables

In addition to the RETL environment variables (please see the Programmer's Guide for version of RETL), it is necessary to set the MMHOME environment variable to the base directory for RMS ETL. This is the top level directory that was selected during the installation process. In the .kshrc (or .profile) file of the RMS RETL user, a line similar to the following must be added :

> export MMHOME=<base directory for RMS ETL>

Also, the RFX_HOME environment variable must be added to the .kshrc (or .profile) file of the RMS RETL user, similar to the following :

> export RFX_HOME=<base directory of the RETL tool install>

Finally, it is necessary to include "${RFX_HOME}/bin" to the PATH variable of the RMS RETL user, in order for the RETL executable to be found.

rmse_config.env

There are a couple variables will need to change depending upon local settings:

> export DBNAME=<ORACLE_SID for the database>
>
> export RMS_OWNER=<database schema owning account name>
>
> export BA_OWNER=<RMS RETL database user name>
>
> export ORACLE_PORT=<Port for connection to the RMS database>
>
> export ORACLE_HOST=<machine name of the RMS database host>
>
> export LANGUAGE=<Valid values are : en, fr, ja, or es>
>
> export DB_ENV=<ORA, etc.>
>
> export RFX_OPTIONS=<options for the RFX executable>
>
> export MMUSER=<RMS RETL database user name>
>
> export PASSWORD=<password for RMS RETL database user account> **

** This Unix environment variable may be set elsewhere for security purposes, but it needs to be defined in order to allow the RETL tool to connect to the database.

Appendix B – Additional RETL RDW information

RMS Extracts (RDW) Code Tree

The following table describes the contents of each of the DWI code tree directories created during the DWI 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>	retek	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 output data files created during batch execution, text files containing rejected records and input files exported from Retek Sales Audit. 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 DWI 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 DWI batch modules.
<base_directory>dev/rfx	lib	This directory contains all DWI library code.

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