Retek® Merchandising System 10.1

Installation Guide (full)

Retek

Customer Support

Customer Support hours:

Customer Support is available 7x24x365 via e-mail, 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.

Contact Method	Contact Information
Internet (ROCS)	www.retek.com/support Retek's secure client Web site to update and view issues
E-mail	support@retek.com
Phone	US & Canada: 1-800-61-RETEK (1-800-617-3835) World: +1 612-587-5800 EMEA: 011 44 1223 703 444 Asia Pacific: 61 425 792 927
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 – Hardware and software requirements

Before you install any Retek Retail Solutions product, you need to make sure that your information systems can adequately run the software that you are installing, as well as process the amount of data that you expect to manage. This section lists the suggested hardware and software requirements for Retek Merchandising products. The following are suggested hardware and software requirements:

Supported Media – Retek Merchandising is available on CD-ROM only. Tape is not available.

Database Server – database software requirements.

Application Server – including operating system software and development tools, and a list of hardware choices.

Web Browser – including the requirements that a Web browser must meet and a list of Web browsers and versions from which you can choose. It is important that you choose to install operating system and Web browser version combinations on your users' computers that can run Oracle's JInitiator. JInitiator is the Java runtime environment necessary for viewing and interacting with Retek's Web-enabled products. The following table lists acceptable operating system versions and Web browser versions.

	Database Server	**Application Server**	**Client**
Vendor	Oracle RDBMS 9I – Enterprise Edition	Oracle Application Server (Web Server) Oracle Developer	Web Browser (IE or Netscape) JInitiator
Retek (RMS 10.1)	Batch Libraries DDL (Views, Triggers, Tables etc.) Database Objects (Procedures, Packages) Control Scripts Data Scripts	Forms Reports Toolset Help Files	

Because you need to choose hardware that has sufficient random access memory (RAM) and program and data storage capacity for the products you choose, each section lists criteria you can use to "size" your hardware selections. The totals you discover after factoring in sizing issues are approximate.

Retek Merchandising System

Database server

General requirements for a database server capable of running RMS include:

- Unix (or Unix variant) based OS certified with Oracle 9i
- ANSI compliant C compiler
- Perl Compiler 5.0 or later
- Oracle RDBMS 9i Enterprise Edition
- Oracle Partitioning
- Oracle Pro*C Precompiler 9.x
- Oracle Net services
- x-Windows interface

For development:

- OCI
- Oracle XML Developers Kit
- Oracle XML SQL Utility

Hardware/OS options as used for development (see Oracle's Web site for certified platforms):

- Sun/Solaris 2.8
- IBM/AIX 4.3.3.x
- Hewlett Packard/HP UX 11.I

Note: Oracle bug #2200335 regarding table inserts is resolved with patch #1970629.

Application server

General requirements for an application server capable of running RMS include:

- UNIX (or UNIX variant) or Windows NT or Windows 2000 server
- Oracle Application Server (9IAS) 1.0.2.2.x
- x-Windows interface (only if UNIX OS)

Sizing factors and other suggestions to factor into your selection of an application server include:

- CD-ROM drive
- 1 Gbit network adapter
- ~2 GB Free disk space for 9IAS
- ~1 GB Free disk space for RMS forms, reports, gif files and help files.

Hardware/OS options as used for development:

- Sun/Solaris 2.6, 2.7, 2.8
- IBM/AIX 4.3.3 or AIX 5.1
- Hewlett Packard/HP UX 11.0 or 11.11

Web browser and client requirements

General Requirements for client capable of running RMS include:

JRE Plugin:

- Oracle JInitiator 1.1.8.xx

Client PCs:

- Pentium Processor
- Use Windows 98, 2000, XP or NT 4.0 with service pack 5 or higher
- Have the resolution set to 1024x768 pixels

Sizing factors and other suggestions to factor into your selection of a PC or network configuration include:

- Bandwidth/Speed
- PC Configuration (minimum 64 MB RAM, 200MHZ processor)

Browser options to factor into your selection include:

- Internet Explorer 5.0 or higher
- Netscape Navigator 4.7 or higher

Chapter 2 – Database installation instructions

Database Server Installation Instructions

Follow these steps to install the database server component of the RMS 10.1 software.

Getting started

Creating a UNIX user account

1   Create the following UNIX groups:

  - dba

  - rtk

2   Create the following UNIX user, using ksh as the default shell:

  - Oracle   - dba group (owns the Oracle RDBMS)

  - Retek   - dba and rtk group (owns the RMS app)

The Retek user will install and compile the Retek Merchandising 10.1 Database Server and Application Server objects on UNIX systems. The Oracle account should create the Oracle 9i database.

Modify the init.ora file in the $ORACLE_HOME/dbs directory

1   Install Oracle 9i as the Oracle account.

2   Place the following in the init.ora:

  - nls_date_format = "DD-MON-RR"

  - job_queue_processes = <number of CPUs + 1>

  - open_cursors=900

3   Create a 9i database (see Appendix D).

Verify the existence of Oracle packages

The **DBMS_SESSION**, **DBMS_RANDOM**, **DBMS_ALERT**, **DBMS_PIPE**, and **DBMS_JOB** packages must be created in each database that RMS 10.1 will be run against.

These Oracle packages are provided with the ORACLE software, and are normally created by the catproc.sql script as part of the Oracle installation process.

1   Log into the database and query the USER_OBJECTS view to verify whether or not the packages have been created, and that Oracle user sys owns these packages.

> **Note:** The source for these packages are located in the $ORACLE_HOME/rdbms/admin directory.

2   If necessary, re-create the packages by running the catproc.sql script while logged in as the Oracle user sys.

Create ORACLE tablespaces

RMS 10.1 requires that three tablespaces be created initially for the RMS installation.

1   Create the tablespaces, **retek_data**, **index_data** and **lob_data**.

> **Note:** These tablespace names are referred to in the table and index creation scripts, so their existence is required.

2   The size of all of these tablespaces will vary from client to client, depending on how much data the client intends on having in their environment. The recommended minimums are: 500MB for **retek_data**, **index_data** and 50 MB **lob_data** – the system tablespace should have at least 400MB free for each installation of the RMS 10.1 schema.

> **Note:** Analysis of additional tablespaces and sizing parameters should be done prior to setting up the production environment.

Create the file structure

1   As the retek user determine where RMS 10.1 install scripts will be installed.
There should be about 500MB of disk space available

2   Make sure your ORACLE_SID and ORACLE_HOME environment
variables are set correctly before installation. Also make sure that
ORACLE_HOME/bin is in your PATH.

3   Mount the CD on the database server and log into UNIX as Retek.

There are four directories on the RMS 10.1 CD: appserverunix, dbserverunix,
appservernt and Retek Workbench. The dbserverunix directory contains the
files for the database server install.

4   Change directories to dbserverunix.

Note: At this point, you may complete the install using the automated install
scripts, or by following the manual steps found in Appendix A.

To complete the install using the automated install scripts:

5   As the retek user run **builddb_rms.run** from the CD while in the <cd mount
point>/dbserverunix directory. The installation script must be run on the
database server.

▪   This script prompts you for a path in which to install the RMS 10.1
database server files on the system. If the entire path does not currently
exist, it creates it for you. **This is referred to as <INSTALL_DIR> in
these installation instructions.**

▪   The builddb_rms.run script copies a tarred and compressed file
containing the database files to the specified directory on your server.

▪   The builddb_rms.run script decompresses and untars the file to produce
the directory structure and files required for the remainder of the
installation process. The directory structure is described in Appendix A.

▪   The builddb_rms.run script cleans up any extra files produced.

▪   The builddb_rms.run script calls another script (install.rms) to finish the
remainder of the Database Server installation. See the *Install RMS*
section for more information.

Note: The install.rms is called by the builddb_rms.run script, but can also be
called from the command line if the tar file was decompressed and untarred
manually. To start install.rms, CD to <INSTALL_DIR>/install and run
install.rms.

Install RMS

The install.rms script walks you through most of the manual install processes described in Appendix A. The install.rms script creates the Oracle schema owner for RMS 10.1 and uses scripts from the **<INSTALL_DIR>**/install directory structure to build the database objects. The basic prompt responses throughout this script are:

- <Y> for Yes

- <N> for No

- <Q> for Quit

- <S> for Skip

- <Enter> to accept the default

The actions that are allowed at each prompt are noted and all choices can be entered in upper or lower case. Each prompt has our suggested answer as default, where clicking **Enter** will accept the default and continue the process.

All of these actions are logged to **<INSTALL_DIR>**/install/logfiles/install.log.

Each of the following bullets is a primary prompt in the script. Refer to the manual instructions in Appendix A for additional information. The install.rms script does the following:

- Ensures the database has been set up to prior specifications.

- Creates the Oracle RMS user that serves as the schema owner. A name, password and a temporary tablespace for this user are required.

- Grants the necessary privileges to the RMS schema owner.

- Generates ddl in the RMS schema.

- Starts the database objects in the RMS schema for the toolset 4.5.

- Creates all other database objects for RMS 10.1. This includes packages, procedures, and functions.

- Creates the views and triggers that rely on the database objects to compile successfully.

- Inserts required data from sql scripts. There is a list in the manual instructions in Appendix A describing the content and function for each of these scripts. The NAV_ROLE script will prompt you for a role. In the initial environment (rms10), you will likely want to use the developer role. This role will grant access to all elements in the Retek Start Tree.

Note: This step will prompt for the following:

'Is Multi-channel on? Y or N:'

'Is Vat on? Y or N:'

'Is Bracket Costing on? Y or N:'

'What is the owner schema?'<RMS 10 Schema Owner>

'What is primary currency?'USD

'How many characters are your country codes? 2 or 3:'

- Imports required data for multiview_default_45, multiview_saved_45 and rtk_errors RMS tables.

Note: This export was done with NLS_LANG set to AMERICAN_AMERICA.UTF8

- Optional: Applies generic data found in the rmsdemo script. This is data that provides fictitious demo style data that can be used to familiarize client staff with RMS 10.1 prior to complete implementation in your own environment.

Note: If you do not run rmsdemo you need to do the following before continuing:

Insert values into SYSTEM_OPTIONS table

- Runs in additional data for the system.

- Makes changes in ddl for 10.1

- Imports required data for multiview_default_45, multiview_saved_45 and rtk_errors RMS tables

- Inserts required data from sql scripts. The NAV_ROLE script will prompt you for a role. In the initial environment (rms10), you will likely want to use the developer role. This role will grant access to all elements in the Retek Start Tree.

- Updates GENERAL_DATA_INSTALL and RMSDEMO

Note: This step will prompt for the following:

'If System Level Vat is on, then is Vat specified at the Class level? Y or N:'

'If Vat is specified at the class level, then is Class Level Vat defaulted on?Yor N:'

- Disables triggers based on weather Multi Channel is on or off

- Updates packages and procedures

- Update RIB doctypes

- Validates invalid objects.

If at any point you choose to exit the install.rms script, the next time it is run, it will ask you if you wish to continue where you last left off. Answering Yes causes the script to pick up where you last left off. Answering No causes the script to start at the beginning. The install.rms script is located at <INSTALL_DIR>/install/install.rms.

Note: The install.rms script was written to install the RMS 10.1 components in a particular order. Use the <S> Skip option with caution!

Copy profiles for the RMS user (Retek)

1   In UNIX, go to the **<INSTALL_DIR>**/*sample_profiles* directory.

In this directory, you will find a sample profile for the initial install environment (rms10).

2   Review the *rms10* sample profile. Edit it for accuracy.

3   Copy the *rms10* sample profile to *.profile* in the home directory of the RMS *rms10* user (Retek).

As new environments and corresponding new UNIX users are created, use this sample profile as a guide to create the new UNIX user's profile.

4   After the profile has been copied and invoked, check the following settings:

▪   MMHOME (using the example above, MMHOME would be **<INSTALL_DIR>**/rms for the Retek user)

▪   MMUSER (RMS 10.1 schema owner associated with the rms10 environment)

▪   PASSWORD (MMUSER [RMS 10.1 schema owner's] password)

▪   ORACLE_SID

Note: The profile should only be available to its owner. You may have to change permissions using the following UNIX command while in the owner's home directory:

```
> chmod 700 ~/.profile
```

This will prohibit others from viewing your Oracle PASSWORD that is set in the profile.

Note to AIX USERS: You must also set the following environment variables:

export OBJECT_MODE=64

export LINK_CNTRL=L_PTHREADS_D7

Additional Note for AIX users

Because of an Oracle bug (2143531) you will need to do the following:

In $ORACLE_HOME/rdbms/lib/env_rdbms.mk, put quotes around the $(OBJS)

parameter on two lines:

generate_import_list "$(OBJS)" $(SHARED_LIBNAME).imp; \

Complete platform-specific preparations to compile the RMS batch libraries and programs

1 Change directories to **<INSTALL_DIR>**/rms/oracle/lib/src.

   Several platform-specific make files have been shipped with this release. They are: platform_sunsol.mk, platform_hpux.mk, platform_aix.mk, platform_dynx.mk, and platform_other.mk.

2 Copy and rename the appropriate platform-specific make file for your UNIX system to platform.mk.

3 If there is not a platform-specific make file for your UNIX system, rename the platform_other.mk to platform.mk, and note that you will need to make modifications to this file.

4 If you are using the Forte cc compiler on a Solaris machine, run patch.ksh to modify the env_precomp.mk that already exists on your system. At the OS prompt, enter:

```
patch.ksh
```

   Note: To run this script you should be the Oracle user.

5 Run the oramake script from the **<INSTALL_DIR>**/rms/oracle/lib/src directory. This will use your server's configurations to create a file called oracle.mk and copy an Oracle-supplied make file (demo_rdbms.mk) to the lib/src directory. At the OS prompt, enter:

```
> oramake
```

   Note for ALL users: Before compiling the batch, make sure the path for the make, makedepend, and compiler commands are in your $PATH environment variable. Also, the LD_LIBRARY_PATH environment variable should include $ORACLE_HOME/lib, as well as $MMHOME/oracle/lib/bin. AIX user should use LIBPATH instead of LD_LIBRARY_PATH.

6 Three sql files must be run into the database after your MMHOME has been set.

   a Change directories to <INSTALL_DIR>/install/db_objects.

   b Log in to SQLPLUS as the <RMS schema owner>.

   c From SQLPLUS, enter the following commands:

```
SQL> @createordlib.sql
SQL> @dealordlib.sql
SQL> @scllib.sql
```

Compile the RMS batch libraries and programs

> **Note:** At this point you can change directories to <INSTALL_DIR>/install and run the batch_compiler script by entering:
>
> ```
> batch_compiler
> ```
>
> at the Unix prompt. This script will run all the following make commands for you. After it has finished, you should still check all logfiles for errors.

1   Change directories to **<INSTALL_DIR>**/rms/oracle/lib/src.

2   Make library dependencies:

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

3   Check the libdpnd.log file for errors.

4   Make batch libraries:

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

5   Check the libretek.log file for errors

6   Install batch libraries:

```
make –f retek.mk install
```

You should now have the batch libraries installed in the **<INSTALL_DIR>**/rms/oracle/lib/bin directory.

7   Change directories to **<INSTALL_DIR>**/rms/oracle/proc/src.

8   Make dependencies:

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

9   Check the srcdpnd.log file for errors.

10 Compile batch programs:

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

11 Check the srcall.log file for errors.

12 Install batch programs:

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

You should now have all of the batch executables in the **<INSTALL_DIR>**/rms/oracle/proc/bin directory.

Set up additional users (optional)

If additional ORACLE users are to be set up at this time, permissions will need to be granted to them so they can run RMS. During integration, there should be multiple roles set up and assigned appropriately to users, based on user responsibilities.

1   Until that point, however, use the following grant command as a sample for what privileges should be granted to users:

```
SQL> grant create session, create table, create
procedure,create view, delete any table, insert any
table, select any table, update any table, select any
sequence, execute any procedure, create any procedure,
drop any procedure, execute any procedure, create any
table, drop any table to <userid>;

SQL> grant developer to <userid>;
```

2   After users are set up, create synonyms to the owner schema for all tables, views, sequences, functions, procedures, and packages that the user will have access to.

Oracle Net Services

Refer to Oracle's install guide for information on configuring Net Services.

Note: See Appendix C for a sample listener.ora and tnsnames.ora files. Pay special attention to the exproc entry, which is required for the scaling functionality.

Chapter 3 – Application server installation instructions

UNIX (Sun Solaris/ HPUX/AIX)

Note: <INSTALL_DIR> is the directory where the RMS 10 files will be extracted from its tar file. 9IAS_ORACLE_HOME is the directory chosen as the ORACLE_HOME to be used for the 9IAS installation.

Install and Configure Oracle 9IAS

Oracle9i Application Server (9IAS) 1.0.2.X – __UNIX__

Note: The Oracle installation tools vary by platform. The essential information is given below, but additional questions/options may be presented during the installation. In these cases, use Oracle's default setting or consult Oracle support. Oracle also recommends staying current on the patches for Developer 6i so you should check with Oracle support for the latest patch level.

1   Create a UNIX user to be used as the 9IAS Administrator account. The 9IAS administrator user must be in the dba group. Log into the application server as the 9IAS Administration user (example: oracle).

2   Insert the Oracle9i Application Server CD into CD-ROM.

3   Read the readme.txt file and ensure that the server configuration meets Oracle's requirements. Make sure to complete all pre-installation requirements.

4   Start the Oracle Installer.

   Note: Run the Oracle Installer from a location other than /cdrom

5   On the Welcome page, click **Next**.

6   Check that the source and destination settings are correct (your ORACLE_HOME for 9IAS).

   Note: 9IAS cannot share an ORACLE_HOME with other Oracle products.

7   Select Enterprise Edition.

8   Select the following the products to install:

   ▪   Forms and Reports Server

   ▪   Oracle HTTP Server

9   On the Database Access Descriptor (DAD) for Oracle9i as Portal page, do not enter any information. Click **Next**.

10 On the Database Access Descriptor (DAD) for the Login Server page, do not enter any information. Click **Next**.

11 On the Wireless Edition repository information page, do not enter any information. Click **Next**.

12 On the Wireless Edition schema information page, do not enter any information. Click **Next**.

13   On the System Password page, do not enter any information. Click **Next**.

14   On the summary page, check the product list again.

15   Click **Install** to begin installation.

16   Change CDs when prompted.

17   Log in as the `root` user.

18   Run `/9IAS_ORACLE_HOME/root.sh` as prompted.

Compile RMS Oracle forms and reports

Setup

To compile the RMS 10.1 Oracle Forms, do the following:

Set up your environment variables

1   Set and export your DISPLAY variable to the IP address of the machine you are
    using to do the installation.

    ```
    Example: export DISPLAY=10.1.2.153:0.0
    ```

2   Set the following variables: INSTALL_DIR is the location where you are planning
    on installing RMS 10.

    ```
    export
    FORMS60_PATH=<INSTALL_DIR>/toolset/bin:<INSTALL_DIR>/rms/form
    s/bin
    ```

    ```
    export REPORTS60_PATH=<INSTALL_DIR>/rms/reports/bin
    ```

    (fill in the appropriate value for `<INSTALL_DIR>`)

    In the following, db_user will refer to the RMS 10 schema owner while oracle_db
    is the Oracle SID where the RMS 10 schema was created.

    ```
    export UP=<db_user>/<db_user_password>@<Oracle_db>
    ```

    Note: On HP-UX you may need to set the UP variable using the following
    command syntax:

    ```
    export UP=<db_user>/<db_user_password>\@<Oracle_db>
    ```

    Set the ORACLE_HOME variable to the Oracle Home used when installing Oracle
    9IAS.

    ```
    export ORACLE_HOME=9IAS_ORACLE_HOME/6iserver
    ```

    ```
    export PATH=9IAS_ORACLE_HOME/6iserver/bin:$PATH
    ```

    ```
    export LD_LIBRARY_PATH=
    9IAS_ORACLE_HOME/6iserver/lib:9IAS_ORACLE_HOME/6iserver/netwo
    rk/jre11/lib/<platform>/native_threads
    ```

Replace `<platform>` with the correct value for your application server operating system.

OS	Value
Solaris	sparc
HP	PA_RISC
AIX	aix

Note: For HP use SHLIB_PATH instead of LD_LIBRARY_PATH.

Create the file structure

1 Insert the RMS 10.1 CD-ROM into the Application Server.

2 Log in as user retek.

3 Change directories to the appserverunix directory on the CD.

4 Determine where you want to install the RMS 10.1 application server files.

Note: RMS 10.1 application files require 1 GB of disk space.

5 Run the script buildapp_rms.run. This will prompt you for the path where RMS 10.1 is to be installed. This will be referred to as <INSTALL_DIR> in the remainder of the documentation.

```
cd appserverunix

./buildapp_rms.run
```

The resulting file structure, located at <INSTALL_DIR>, will contain directories for one RMS environment. The /rms and /toolset directories contain the RMS 10.1 source code. Additional environments can be created as necessary.

Note: Your environment variables must be set correctly for the following automatic install to work correctly.

Once it has copied the RMS 10 files to the proper location, the buildapp_rms.run script will give you a prompt asking if you'd like to continue with the automatic installation of RMS. If you choose 'Y', it will run installapp.rms to automate the compilation of the toolset and forms. If you choose 'N', refer to appendix B for manual instructions.

The installapp.rms script will prompt you for the number of threads to use in the compilation process. This will vary by machine. If in doubt, enter 1.

Installapp.rms walks you through most of the manual install processes described in Appendix B. It compiles libraries, forms and menus.

The installapp.rms script does the following:

- compiles toolset plls

- compiles toolset forms

- compiles toolset menus

- compiles plls

- compiles reference forms

- compiles forms

- compiles menus

- compiles reports plls

- compiles reports

After compilation is complete, check <INSTALL_DIR>/rms/forms/bin for fm_link.fmb. If this file is missing, copy it from <INSTALL_DIR>/rms/forms/src to the /bin directory.

```
> cp <INSTALL_DIR>/rms/forms/src/fm_link.fmb
<INSTALL_DIR>/rms/forms/bin
```

Refer to the manual instructions in Appendix B for additional information on each of these tasks.

Once installapp.rms has completed, view the log file at <INSTALL_DIR>/install/logfiles to check for errors. Errors that did not prevent the generation of an fmx file for non-reference forms (reference forms start with fm_) can be ignored.

Note: The fm_link.fmb will error out with an error such as:

FRM-30173: Module contains no canvases.

Form: FMLINK

FRM-30085: Unable to adjust form for output.

Careful examination of the log files will show that the compile for fm_link.fmb was successful but that the generation of the form executable (.fmx) fails with the above error. This is because this form does not contain a canvas - hence the compiler cannot create a runtime fmx for the form. Similar errors in other reference forms can be ignored.

Configuring 9IAS (UNIX)

Note: The 9IAS configuration steps should be done by the Oracle 9IAS administrator account.

1 Copy the following files at <INSTALL_DIR>/web_html/samplefiles/ to a temporary directory in the Oracle 9IAS administrator's home directory. The Oracle 9IAS administrator is the Unix user that installed 9IAS.

   - ias_web_start – used to start http server and Developer 6i server

   - ias_web_stop – used to stop the http server and Developer 6i server.

   - rms_env – Contains environment variable information used by ias_web_start. It is used by both scripts to set environment variables necessary for execution. You will have to edit this file to make sure the correct LD_LIBRARY_PATH setting is uncommented depending on your server's operating system. Solaris is the default. Oracle Reports also requires that you set the DISPLAY variable to the IP address of a valid Xwindows device. See Oracle documentation for more information.

2 Edit the above scripts:

   a Replace 9IAS_ORACLE_HOME with the Oracle Home used during the installation of 9IAS.

   b Replace **RMS_INSTALL_DIR** with the directory where RMS 10 was installed.

   c Uncomment the DISPLAY variable and set it's value to a valid Xwindows device.

   d Include the location of these files in the 9IAS administrator's PATH variable setting.

3 Copy the file <INSTALL_DIR>/web_html/samplefiles/rmsunix.conf to 9IAS_ORACLE_HOME/Apache/Apache/conf. rmsunix.conf contains the RMS-specific settings that need to be added to the httpd.conf configuration file that was generated during the installation of 9IAS and is located at 9IAS_ORACLE_HOME/Apache/Apache/conf.

4 In rmsunix.conf, replace all occurrences of 9IAS_ORACLE_HOME and **RMS_INSTALL_DIR** with your environment's information.

   Note: It is good practice to backup original Oracle files, ie: httpd.conf

5 Append the contents of rmsunix.conf to the end of httpd.conf.

6 Rename httpd.conf to rms.conf.

7   Look through rms.conf and make the following settings (or verify that they are set correctly): HTTP_PORT should be set to a port that is not being used.

```
Port          HTTP_PORT

ServerAdmin   <set to an admin email account>

ServerName    SERVER_NAME

DocumentRoot  <INSTALL_DIR>/web_html

<Directory <INSTALL_DIR>/web_html>  (must be the same value
as DocumentRoot)
```

8   Copy the file <INSTALL_DIR>/web_html/samplefiles/Tk2Motif.rgb (if case sensitivity was lost during the ftp, rename the file back to Tk2Motif.rgb during the copy) to 9IAS_ORACLE_HOME/6iServer/guicommon6/tk60/admin/. This file allows the forms server to run using the Oracle UTF8 font.

Check the Web environment directory structure

1   Go to the directory <INSTALL_DIR>/web_html.

2   Verify that the following directories exist:

- temp

- log

- jinitiator

- gif

- reptemp

- help

- helpfiles

Miscellaneous configuration tasks

1   SQLPLUS into the RMS schema as the schema owner and run the following commands. You will need to substitute the name of your reports server for REPORTS_SERVER_NAME. Check your ias_web_start script to verify the name of your reports server.

```
select * from lang where lang=1;

update lang set DESCRIPTION='english' where lang=1;

update lang set WEBHELP_SERVER='http://SERVER_NAME:HTTP_PORT'
where lang=1;

update lang set REPORTS_SERVER=REPORTS_SERVER_NAME where
lang=1;

update lang set WEBREPORTS_SERVER='dev60cgi/rwcgi60' where
lang=1;

commit;
```

Substitute your environment's SERVER_NAME, HTTP_PORT (from rms.conf) and REPORT_SERVER_NAME (example REPORTS_SERVER_NAME: repserver.world)

2   Add an entry for both the database and the reports server into the two tnsnames.ora files at:

```
9IAS_ORACLE_HOME/network/admin/tnsnames.ora
```

```
9IAS_ORACLE_HOME/6iserver/network/admin/tnsnames.ora
```

Here are samples for both of the entries:

```
DB_SID=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(hos
t=DB_SERVER_NAME)(Port=DB_LISTENER_PORT)))(CONNECT_DATA=(SID=
DB_SID)(GLOBAL_NAME=SID.world)))
```

```
REPORTS_SERVER_NAME=(ADDRESS=(PROTOCOL=tcp)(HOST=SERVER_NAME)
(PORT=REPORTS_SERVER_PORT)).
```

Edit the netscape_11814.html file

The file is located in <INSTALL_DIR>/web_html/jinitiator.

1   Fill in the correct values for Server_Name and Port.

2   Save the file.

This file will allow JInitiator to be dynamically installed on clients when accessed for the first time.

Modify the following file

9IAS_ORACLE_HOME/6iserver/forms60/java/oracle/forms/registry/Registry.dat:

• Set the default.icons.iconpath entry near the end of the file:

```
default.icons.iconpath=/web_gif/
```

Copy the keyboard mapping configuration file to the forms60 admin directory

• Copy the file fmrweb.res, found in <INSTALL_DIR>/web_html/samplefiles to 9IAS_ORACLE_HOME/6iserver/forms60/admin/resource/US/.

Create the Retek HTML Start Page

1   Copy rms.html from <INSTALL_DIR>/web_html/samplefiles to <INSTALL_DIR>/web_html.

2   Modify the serverPort setting in this file to point to the port that the forms server is starting on (refer to ias_web_start – the default is 10000).

Install the Oracle JInitiator Component on the Server

JInitiator 1.1.8.14 is included on the RMS10.1 file structure in the directory <INSTALL_DIR>/web_html/jinitiator. Check to make sure the file jinit11814.exe is there.

Browser requirements:

You will need IE 5.0 or Netscape 4.7 (or higher versions) as your Web browser to use RMS 10.1.

Test the System

Run ias_web_stop then run ias_web_start to bounce the Web processes. Connect the client to the server by issuing:

```
http://SERVER_NAME:HTTP_PORT/rms.html
```

The first time that you connect to the server, *jinitiator* will download and install. The *jinitiator* download will occur the first time that each machine accesses RMS. Restart the browser after *jinitiator* is installed.

Windows (NT, Windows 2000)

Install Oracle9i Application Server (9IAS) 1.0.2.X– **NT-Windows 2000**

> **Note:** The Oracle installation tools vary by platform. The essential information is given below, but sometimes, additional questions/options may be presented during the installation. In these cases, use Oracle's default setting or consult Oracle support. Oracle also recommends staying current on the patches for Developer 6i so you should check with Oracle support for the latest patch level.

1   Log in to the machine as the local administrator.

2   Insert the Oracle9i Application Server CD into CD-ROM.

3   Read the readme.txt file and ensure the server configuration meets Oracle's requirements. Make sure to complete all pre-install requirements.

4   The Installer will run automatically.

5   On the Welcome page, click **OK**.

6   Select Enterprise Edition.

7   Select the ORACLE HOME NAME and ORACLE HOME LOCATION for 8.1.7 RSF-based products (this page is displayed if this is the first Oracle product installed on the machine).

8   Enter the ORACLE HOME NAME and ORACLE HOME LOCATION for the Oracle9iAS installation.

> **Note:**   9IAS cannot share an ORACLE_HOME with other Oracle products.

9   Select the ORACLE HOME NAME and ORACLE HOME LOCATION for 8.0.6 RSF-based products (this page is displayed if this is the first Oracle product install on the machine this will be the directory… accept the default setting, which is different than that for 8.1.7 RSF-based products).

10 Choose to install Forms and Reports Server and Oracle HTTP Server.

11 On the Database Access Descriptor [DAD] page for Oracle9iAS Portal, do not enter any information. Click **Next**.

12 On the Database Access Descriptor [DAD] page for Login Server, do not enter any information. Click **Next**.

13   On the Wireless Edition repository information… page, do not enter any information. Click **Next**.

14   On the Wireless Edition schema information… page, do not enter any information. Click **Next**.

15   On the …SYSTEM Password for Wireless Edition page, do not enter any information. Click **Next**.

16   On the summary page, check the product list again.

17   Click **Install** to begin installation.

18   Change CDs when necessary.

19   The installation is complete.

Configure Oracle9i Application Server (9IAS) 1.0.2.X – **NT**

Note: Oracle 9IAS on NT/2000 does not include the Oracle Developer tools suite. It only contains the runtime components of Developer 6i server. If you wish to compile/modify forms, you will need to install Developer 6i.

1   Copy the file appservernt.exe from the appservernt directory on your installation cd to the directory where you are planning on installing RMS. Execute the file to build the RMS directory structure. This will be referred to as <INSTALL_DIR> in the rest of this document.

2   After the installation above, your 9IAS HTTP listener might have automatically been started. Follow these instructions to shut down the http listener.

   a   By default, the Oracle HTTP server will be installed under 9IAS_ORACLE_HOME\iSuites; and Developer 6i server, which is bundled with 9IAS, will be installed under 9IAS_ORACLE_HOME\806. Make sure the PATH system property contains the following entries:

   ```
   9IAS_ORACLE_HOME\iSuites\Apache\Apache

   9IAS_ORACLE_HOME\iSuites\Apache\Apache\bin

   9IAS_ORACLE_HOME\iSuites\BIN

   9IAS_ORACLE_HOME\806\BIN
   ```

   b   At a DOS prompt, use the command "apache –k shutdown" to stop the http process.

3   Copy <INSTALL_DIR>\samplefiles\rmsnt.conf to 9IAS_ORACLE_HOME\iSuites\Apache\Apache\conf.

   rmsnt.conf contains the RMS-specific settings that need to be added to the httpd.conf configuration file that was generated during the installation of 9IAS and is located at 9IAS_ORACLE_HOME\iSuites\Apache\Apache\conf.

4   In rmsnt.conf, replace all occurrences of 9IAS_ORACLE_HOME and **RMS_INSTALL_DIR** with your environment's information.

   Note: It is good practice to backup original Oracle files, for example: httpd.conf

5   Append the contents of rmsnt.conf to the end of httpd.conf.

6   Rename httpd.conf to rms.conf.

7   Look through rms.conf and make the following settings (or verify that they are set correctly):

```
Port          HTTP_PORT

ServerAdmin   <set to an admin email account>

ServerName    SERVER_NAME

DocumentRoot  <INSTALL_DIR>\web_html

<Directory <INSTALL_DIR>\web_html>  (must be the same value
as DocumentRoot)
```

8   Modify the file 9IAS_ORACLE_HOME\806\forms60\java\Oracle\forms\registry\Registry.dat:

- Near the end of file, add "/web_gif/" so that the iconpath setting looks like "default.icons.iconpath=/web_gif/"

9   Copy apache_start, apache_stop, and rms_form.bat from <INSTALL_DIR>\web_html\samplefiles to the directory on your server that will be used to start and stop the web processes.

10 In these files, replace any references to 9IAS_ORACLE_HOME and **RMS_INSTALL_DIR** with your environment's values. You can choose which port you would like your forms server to run on, if you wish, by modifying run_form.bat – the default port is 10000.

11 Install a reports server as a service using the command "rwmts60 –install name=REPORTS_SERVER_NAME". Make the following entries in the registry at HKEY_LOCAL_MACHINE|SOFTWARE|ORACLE|HOME0:

```
REPORTS60_PATH            <INSTALL_DIR>\rms\reports\bin

REPORTS60_PHYSICAL_MAP    <INSTALL_DIR>\web_html\temp

REPORTS60_SHARED_CACHE    YES

REPORTS60_VIRTUAL_MAP     /reptemp

REPORTS60_WEBLOC          /reptemp

REPORTS60_WEBLOC_TRANSLATED       <INSTALL_DIR>\web_html\temp
```

12 Copy rms.html from <INSTALL_DIR>\web_html\samplefiles to <INSTALL_DIR>\web_html. Modify the serverPort setting in this file to point to the port that the forms server is starting on (refer to rms_form.bat).

13 SQLPLUS into the RMS schema as the schema owner and run the following commands:

```
select * from lang where lang=1;

update lang set DESCRIPTION='english' where lang=1;

update lang set WEBHELP_SERVER='http://SERVER_NAME:HTTP_PORT'
where lang=1;

update lang set REPORTS_SERVER=REPORTS_SERVER_NAME where
lang=1;

update lang set WEBREPORTS_SERVER='dev60cgi/rwcgi60' where
lang=1;

commit;
```

Substitute your environment's SERVER_NAME, HTTP_PORT (from rms.conf) and REPORT_SERVER_NAME (example REPORTS_SERVER_NAME: repserver.world)

14 Add an entry for both the database and the reports server into the two tnsnames.ora files at

```
9IAS_ORACLE_HOME\iSuites\network\admin\tnsnames.ora

9IAS_ORACLE_HOME\806\net80\admin\tnsnames.ora
```

Here are samples for both of the entries – substitute your environment's setting for DB_SID, SERVER_NAME, DB_LISTENER_PORT, REPORTS_SERVER_NAME and REPORTS_SERVER_PORT.

```
DB_SID=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=tcp)(hos
t=DB_SERVER_NAME)(Port=DB_LISTENER_PORT)))(CONNECT_DATA=(SID=
DB_SID)(GLOBAL_NAME=DB_SID.world)))

REPORTS_SERVER_NAME=(ADDRESS=(PROTOCOL=tcp)(HOST=SERVER_NAME)
(PORT=REPORTS_SERVER_PORT)).
```

15 Edit the file netscape_11814.html located at <INSTALL_DIR>\web_html\jinitiator. Replace SERVER_NAME and HTTP_PORT with the values for your environment.

16 Copy the file fmrweb.res from <INSTALL_DIR>\web_html\samplefiles to 9IAS_ORACLE_HOME\806\forms60\admin\resource\US. This is the file that controls keyboard mapping for the RMS application.

17 Test your environment: Start up your RMS environment by running apache_start.bat and rms_form.bat from the directory chosen in #3 above. You can access the application by going to http://SERVER_NAME:HTTP_PORT/rms.html

Appendix A – Manual database installation instructions

Before beginning these steps, verify that the following procedures are complete: *Create a UNIX user account*, *Modify the init.ora file in the $ORACLE_HOME/dbs directory*, *Verify the existence of Oracle packages*, and *Create ORACLE tablespaces* in *Chapter 2 – Installation Instructions*. Complete either the steps in this appendix, or the procedures under *Create the File Structure* and *Install RMS* in *Chapter 2 – Installation Instructions*.

After mounting the CDROM:

1   Create a directory for the RMS 10.1 software. It will be referred to as **<INSTALL_DIR>** for the remainder of this document.

2   Copy the dbserverunix.Z to the **<INSTALL_DIR>**.

3   Uncompress the file:

```
> uncompress dbserverunix.Z
```

4   Untar the file:

```
> tar xvf  dbserverunix
```

This creates the directory structure in which the RMS 10.1 files will reside. The tarfile will not retain the permission settings they had when leaving Retek. Verify that the source code is protected by altering the permissions with the chmod command. Keep in mind there will be some directories that need to be written to during this install process.

The directory structure will look like this:

./sample_profiles	A profile to aid environment setup
./install	Various directories and the install.rms script.
./install/db_objects	Packages, procedures, functions, shared libraries.
./install/ddl	DDL files.
./install/import	Scripts to populate RMS tables.
./install/logfiles	Installation log directory.
./install/sqlplus	Scripts to populate required data.
./install/utility	Various useful scripts.
./install/installer_scripts	Used by install script to hold information/scripts.
./install/toolset_dbo	Toolset packages.
./rms	Parent directory for various environment directories.
./rms/oracle	
./rms/oracle/lib	
./rms/oracle/lib/bin	Batch library executables.
./rms/oracle/lib/src	Batch library source code.
./rms/oracle/proc	
./rms/oracle/proc/bin	Batch program executables
./rms/oracle/proc/src	Batch source code.

The files necessary for the server portion of the database installation reside in the directory structure above.

5   Verify that all files are owned by Retek and belong to the dba group. Make ownership and group changes if necessary.

6   Create RMS 10.1 Schema owner in Oracle Instance:

Each ORACLE database user needs to have a role named *developer* granted to it initially in order to run the RMS. The RMS menu needs to be customized during integration to set up appropriate roles. For now, all ORACLE users created need to have this role granted to them.

a   Log in to SQLPLUS as system.

b   Enter the following commands:

```
SQL> create role developer;
```

7   Create the ORACLE user that will correspond to the *rms* environment:

    a   Log into SQLPLUS as the user system.

    b   Enter the following commands, replacing the text in the brackets with the appropriate names:

```
SQL> create user <RMS 10.1 Schema Owner> identified by
<password> default tablespace Retek_data temporary
tablespace <temporary tablespace name>;
```

The ORACLE user <RMS 10.1 Schema Owner> will serve as the owner of the database objects and should be granted the following permissions.

8   Log in to SQLPLUS as the user system.

9   Enter the following command:

```
SQL> GRANT

DEVELOPER, SELECT_CATALOG_ROLE,

ALTER SESSION, ANALYZE ANY,

CREATE ANY SYNONYM, CREATE DATABASE LINK,

CREATE LIBRARY, CREATE PROCEDURE,

CREATE PUBLIC DATABASE LINK,

CREATE PUBLIC SYNONYM, CREATE SEQUENCE,

CREATE SESSION,CREATE SYNONYM, CREATE TABLE,

CREATE TRIGGER, CREATE VIEW, DROP ANY SYNONYM,

EXECUTE ANY PROCEDURE, SELECT ANY SEQUENCE,

SELECT ANY TABLE TO  <RMS 10.1 Schema Owner>

/

alter user <RMS 10.1 Schema Owner>  quota unlimited on
Retek_data

/

alter user <RMS 10.1 Schema Owner>  quota unlimited on
index_data

/


alter user <RMS 10.1 Schema Owner>  quota unlimited on
lob_data

/
```

The following three commands should be run as sys:

```
grant select on sys.dba_roles to <RMS 10.1 Schema Owner>

/

grant select on sys.dba_role_privs to <RMS 10.1 Schema
Owner>

/

grant select on sys.dba_jobs to <RMS 10.1 Schema Owner>

/
```

10 Create DDL for RMS 10.1 owner:

   a   Log into UNIX as the retek user.

   b   Change directories to: **<INSTALL_DIR>**/install/ddl. This directory contains the scripts required in order to create the tables, indexes, constraints, sequences, triggers and views within the RMS.

   Reminder: Partitioning is required for all database objects to create successfully.

   c   To start the scripts, log in to SQLPLUS as the <RMS 10.1 Schema Owner> and enter the following command:

   ```
   SQL> @rms100.sql
   ```

   ```
   This runs all of the scripts needed to create tables,
   indexes, constraints, sequences, triggers, and views for
   the user.  A spool file will be created named rms100.log
   which is in <INSTALL_DIR>/install/logfiles.
   ```

   d   Upon completion, check the spool file to verify that no errors were received.

   Note: Not all of the views and triggers will be created at this time; only the views in the file rms100.vw will be started as well as the triggers in file rms100.trg. The reason for this is that some of the views and triggers refer to functions that have not been created yet. Until the functions are created, these views and triggers cannot be created successfully. You will be instructed later in the document as to when to create the remaining views (rms100.vw2) and triggers (rms100.trg2).

11 Create packages for the toolset 4.5.

   On the server, change directories to **<INSTALL_DIR>**/install/toolset_dbo. All packages within the toolset must be started in the database in the appropriate order. The script **toolset45startallpackages.sql** will start the toolset 4.5 packages automatically and will spool to the t*oolset45startallpackages.log* file which is located in **<INSTALL_DIR>**/install/logfiles.

   a   Log in to SQLPLUS as <RMS 10.1 Schema Owner>.

   b   Enter the following command to run the script:

   ```
   SQL> @toolset45startallpackages.sql
   ```

   c   When completed, view the spool file created to verify that no errors were found.

12 Create packages, stored procedures, and functions.

   a   On the server, change directories to
       <INSTALL_DIR>install/db_objects.

This directory contains a copy of all procedures, packages, and functions that are used within RMS 10.1.

All of these objects must be started in the database in the appropriate order. The script `rms100startallpackages.sql` will do this automatically. The rms100startallpackages.sql script will spool to the rms100startallpackages.log file which is located in the **<INSTALL_DIR>**/install/logfiles directory.

   b   Log in to SQLPLUS as <RMS 10.1 Schema Owner>.

   c   Enter the following command to run the script:

```
SQL> @rms100startallpackages.sql
```

This creates each of the packages, procedures, and functions used within RMS 10.1. This takes approximately 10-30 minutes to complete. (The amount of time this requires varies, depending on hardware and configuration.)

   d   When completed, exit out of SQLPLUS and view the spool file to verify that no errors were found.

13 Create remaining views and triggers.

Now that all of the stored objects exist in the database, the rest of the views and triggers can be created. The rms100.sql2 script (which runs rms100.vw2 and rms100.trg2) will spool to the rms100.log2 file in the **<INSTALL_DIR>**/install/logfiles directory.

   a   Change directories to **<INSTALL_DIR>**/ddl.

   b   Log in to SQLPLUS as the user <RMS 10.1 Schema Owner>.

   c   From SQLPLUS, enter the following commands:

```
SQL> @rms100.sql2
```

   d   When completed, view the spool files to verify that no errors were found.

14 Load required data.

a   Log into UNIX as Retek.

b   Change directories to **<INSTALL_DIR>**/install/sqlplus.

c   In this directory there are a number of files that contain data required for RMS 10.1 to run properly. Below is an alphabetical listing and explanation of each of the SQL scripts. The order in which they are applied is noted in the rms100startallctlscripts.sql script.

SQL Script Name	Description
addplcy	Runs the procedures necessary for RMS 10.1 security.
ari_interface_test_data.sql	Creates data required for ari_interface_test table.
codes.sql	Creates data required for the code_head and code_detail table.
drpplcy	This is not run initially, but is provided as a tool.
elc_comp_htsupld.sql	This inserts initial estimated landed cost expense data for RMS 10.1.
rms100startallinstallscripts.sql	Starts all the primary installation scripts for RMS 10.1.
navigate.sql	Creates data for the tree within the Retek start form.
navrole.sql	Quick way to grant access to every element in the Retek start tree. This script will prompt you to enter a role. The developer role can be selected at this time if no other roles have been created.
populate_form_links.sql	This script and populate_forms_links_role.sql should always be run together. This script populates the FORM_LINKS table. This table contains information about the links (similar to hyperlinks commonly used on the internet) in several item forms (itemmaster, itemchildren, item supplier, item supplier country, and item location). The links allow for navigation throughout the item dialog.
populate_form_links_role.sql	This script needs to be run immediately following populate_forms_links.sql. It populates FORM_LINKS_ROLE, which contains information about which roles have access to which links. This script will insert data so that the DEVELOPER role has access to all links.

SQL Script Name	Description
restart.sql	Creates data used for restart/recovery in batch programs.
rtk_reports.sql	Creates data used for mass generation of reports.
secuser.sql	Creates data to setup the schema owner with full access to all tables within Retek security.
staticin.sql	Creates all of the static data that is required for the RMS 10.1 to run.
tl_columns.sql	Creates data used for mass translation.
uom_x_conversion.sql	Required data for unit of measure conversions.
var_upc_ean_load.sql	This script populates the VAR_UPC_EAN table, which holds data that identifies how to interpret the layout of a variable weight UPC code that is attached to an item.
wizard.sql	Creates data used for instructions for wizards within the RMS 10.1.

15 Log in to SQLPLUS as the user <RMS 10.1 Schema Owner>.

16 From SQLPLUS, enter the following commands:

```
SQL> @retek10_data_populations.sql
```

Note: When running this script, there is a prompt that will ask 'How many characters would you like your country codes to be? 2 or 3'. You must either enter 2 or 3 for this.

Run the rms100startallctlscripts.sql file

Running the rms100startallctlscripts.sql file will start all of the scripts listed above. To do this:

17 Log in to SQLPLUS as the user <RMS 10.1 Schema Owner>.

18 Enter the following:

```
SQL> @ rms100startallctlscripts.sql
```

A spool file named rms100startallctlscripts.log will be created in **<INSTALL_DIR>**/install/logfiles.

19 Verify that this file has no errors upon completion.

Load control data

20 While in the **<INSTALL_DIR>**/install/db_objects run the following command:

```
$sqlldr <RMS 10 Schema Owner>/<password>
control=rib_doctypes_rms.ctl
```

Run rmsdemo.sql

If you require a small set of data at this time, then run the file named rmsdemo.sql from SQLPLUS. It creates a minimal amount of data in the organizational and merchandise hierarchies, etc. This script will prompt for the following:

```
'Is Multi-channel on? Y or N:'

'Is Vat on? Y or N:'

'Is Bracket Costing on? Y or N:'

'What is the owner schema?'<RMS 10 Schema Owner>

'What is primary currency?'USD

'How many characters are your country codes? 2 or 3:'
```

Note: When running this script, the customer dba is responsible for setting these values to ensure that the data is correct.

21 From the **<INSTALL_DIR>**/install/sqlplus, log in to SQLPLUS as the user <RMS 10.1 Schema Owner>.

22 Enter the following:

```
SQL> @rmsdemo.sql
```

It creates a spool file named rmsdemo.log in the **<INSTALL_DIR>**/install/logfiles directory.

Run import_data

Note: Before starting this step ensure you ORACLE_SID is set correctly.

Note: This export was done with NLS_LANG set to AMERICAN_AMERICA.UTF8

There is also some data that needs to be imported into the system during installation. The tables requiring imports are: rtk_errors, multiview_default_45, and multiview_saved_45. This data can be automatically imported by using the import_data script.

23 Change directories to:

<INSTALL_DIR>/install/import.

24 Enter the following:

```
$import_data
```

25 Enter the RMS 10.1 Schema Owner's connect string (userid/password@instance) when prompted.

You will also be prompted for the system connect string – The script drops constraints on the rt_errors and the multiview default45 table as the schema owner and then does the import of the schema owner's data as sys. The script will produce an output file in **<INSTALL_DIR>**/install/logfiles named import.log

26 Check this log to ensure a successful import.

27 If you have not run rmsdemo:

- Insert values into SYSTEM_OPTIONS table

- Run the <INSTALL_DIR>/install/sqlplus/rms100startall2.sql

If you did run rmsdemo:

- Run the <INSTALL_DIR>/install/sqlplus/rms100startall2.sql

Update DDL

28 Change directories to `<INSTALL_DIR>/install/upgrade/`

29 Log in to SQLPLUS as the <RMS 10.1 Schema Owner>.

30 Enter the following command:

```
SQL> @patch101dbc.sql
```

31 View the spool file `patch101dbc.log` when finished to verify that no errors were found.

Import data

32 Change directories to `<INSTALL_DIR>/install/upgrade`.

33 Log in to SQLPLUS as the <RMS 10.1 Schema Owner>.

34 Enter the following command to truncate table:

```
SQL> @rms_truncate.sql

SQL> @disable_con.sql
```

35 Exit SQLPlus.

36 Edit the rms_import.par file

```
set TOUSER=<schema owner>
```

37 At the Unix prompt, enter the following:

```
$ imp parfile=rms_import.par
```

Update Control tables

38 Change directories to `<INSTALL_DIR>/install/upgrade`.

39 Log in to SQLPLUS as the <RMS 10.1 Schema Owner>.

40 Enter the following command to update control tables:

```
SQL> @patch101ctl.sql
```

41 View the spool file `patch101ctl.log` when finished to verify that no errors were found.

Disable Triggers

42 Change directories to `<INSTALL_DIR>/install/upgrade`.

43 Log in to SQLPLUS as the <RMS 10.1 Schema Owner>.

44 Enter one of the following commands to disable the appropriate triggers.

45   To select `multichannel_ind` from `system_options,` run the following command:

   `select multichannel_ind from system_options;`

46   If `multichannel_ind` is set to 'N', run the following command:

   `SQL>@`**`disable_mult_n.sql`**

47   If `multichannel_ind` is set to 'Y' run the following commands:

   `SQL>@disable_mult_y.sql`

Update packages, stored procedures and functions

48   Change directories to `<INSTALL_DIR>/install/upgrade`

49   Log in to SQLPLUS as the <RMS 10.1 Schema Owner>.

50   Enter the following command to update packages, procedures, and functions:

   `SQL> @patch101rms.sql`

51   Exit SQLPlus.

52   View the spool file `patch101rms.log` when finished to verify that no errors were found.

53   While in the `<INSTALL_DIR>/install/upgrade directory run the following command:`

   `$sqlldr <RMS 10.1 schema Owner>/<password>`

   `control=rib_doctypes_rms.ctl`

54   Change directories to **<INSTALL_DIR>**/install/utility

55   Log onto SQLPLUS as the RMS schema owner and run @inv_obj_comp.sql until all invalid objects compile.

Appendix B – Manual application server installation instructions

Compile RMS Oracle forms and reports

Setup

To compile the RMS 10.1 Oracle Forms, do the following:

Set up your environment variables

1   Set and export your DISPLAY variable to the IP address of the machine you are using to do the installation.

```
Example: export DISPLAY=10.1.2.153:0.0
```

2   Set the following variables: INSTALL_DIR is the location where you are planning on installing RMS 10.

```
export
FORMS60_PATH=<INSTALL_DIR>/toolset/bin:<INSTALL_DIR>/rms/for
ms/bin
```

```
export REPORTS60_PATH=<INSTALL_DIR>/rms/reports/bin
```

(fill in the appropriate value for <INSTALL_DIR>)

In the following, db_user will refer to your RMS 10 schema owner while oracle_db is the Oracle SID where the RMS 10 schema was created.

```
export UP=<db_user>/<db_user_password>@<Oracle_db>
```

NOTE: On HP-UX you may need to set the UP variable using the following command syntax:

```
export UP=<db_user>/<db_user_password>\@<Oracle_db>
```

Set the ORACLE_HOME variable to the Oracle Home used when installing Oracle 9IAS and append /6iserver at the end.

export ORACLE_HOME=9IAS_ORACLE_HOME/6iserver

Note: The ORACLE_HOME setting is different than the setting for the automatic install. ORACLE_HOME needs to be set to the location of Developer 6i – this is located at 9IAS_ORACLE_HOME/6iserver.

```
export PATH=9IAS_ORACLE_HOME/6iserver/bin:$PATH
```

```
export LD_LIBRARY_PATH=
9IAS_ORACLE_HOME/6iserver/lib:9IAS_ORACLE_HOME/6iserver/netw
ork/jre11/lib/<platform>/native_threads
```

Replace `<platform>` with the correct value for your application server operating system.

OS	Value
Solaris	sparc
HP	PA_RISC
AIX	aix

Note: For HP use SHLIB_PATH instead of LD_LIBRARY_PATH

Create the file structure

1   Insert the RMS 10.1 CD-ROM into the Application Server.

2   Log in as user retek.

3   Change directories to the appserverunix directory on the CD.

4   Determine where you want to install the RMS 10.1 application server files.

Note: RMS 10.1 application files require 1 GB of disk space.

5   Run the script buildapp_rms.run. This will prompt you for the path where RMS 10.1 is to be installed. This will be referred to as <INSTALL_DIR> in the remainder of the documentation.

```
cd appserverunix

./buildapp_rms.run
```

The resulting file structure, located at <INSTALL_DIR>, will contain directories for one RMS environment. The /rms and /toolset directories contain the RMS 10.1 source code. Additional environments can be created as necessary.

Note: Your environment variables must be set correctly for the following manual install to work correctly.

Once it has copied the RMS 10 files to the proper location, the buildapp_rms.run script will give you a prompt asking if you'd like to continue with the automatic installation of RMS. Choose 'N' to do the manual installation of RMS.

Compile Toolset Libraries (*.pll)

1   Change directories to <INSTALL_DIR>/toolset/src.

2   Use the `pld2pll45` script to convert all files in the directory from a file with a .pld extension to a file with a .pll extension.

```
> chmod 755 pld2pll45 (if necessary).

> ./pld2pll45
```

3   Check to make sure all files with a .pld extension now have a corresponding file with a .pll extension.

4   Move all of the libraries (.pll files) in the <INSTALL_DIR>/toolset/src directory to the <INSTALL_DIR>/toolset/bin directory.

5   Change directories to <INSTALL_DIR>/toolset/bin directory.

6   Start the Form Builder tool to compile all libraries for the RMS toolset

```
> f60desm &
```

a   A blue GUI interface is displayed. On the Welcome page, click **Cancel**.

b   Choose File → Connect. Log into the database as the RMS 10 schema owner schema owner.

c   Compile the libraries in the following order:

- messge45.pll

- ariiflib.pll

- stand45.pll

- calend45.pll

- find45.pll

- item45.pll

- tools45.pll

- mblock45.pll

- mview45.pll

- nav45.pll

- work45.pll

- itnumtype.pll

7   For each library file:

a   Choose File > Open.

b   Select <INSTALL_DIR>/toolset/bin/FILENAME.pll.

c   Click **OK**.

d   Once the library is "loaded", select the library name, select Program, and choose Compile > All.

e   After successful compilation, click **OK**.

f   Save and close the library.

Compile Toolset Reference Forms (fm_*.fmb)

1   Change directories to <INSTALL_DIR>/toolset/src

2   Move all reference forms (fm_*.fmb) from <INSTALL_DIR>/toolset/src to <INSTALL_DIR>/toolset/bin.

```
> mv  fm_*.fmb ../bin
```

3   Using your Form Builder session (f60desm &), navigate to
    <INSTALL_DIR>/toolset/bin/ and compile each reference form in the
    <INSTALL_DIR>/toolset/bin/ directory:

a   Choose File > Open.

b   Select <INSTALL_DIR>/toolset/bin/fm_<FILENAME>.fmb.

c   Click **OK**.

d   Once the reference form is "loaded", select the form name, select Program,
    and choose Compile > All.

e   After successful compilation, click **OK**.

f   Save and close the reference form.

The toolset reference forms have been compiled and should now reside in the bin
directory.

Compile Toolset Forms (*.fmb)

1   Change directories to <INSTALL_DIR>/toolset/src.

2   Use the fmb2fmx45 script located in that directory to compile and generate the
    executable forms (fmx).

```
> chmod 755 fmb2fmx45 (if necessary).

> ./fmb2fmx45
```

3   Check to make sure each .fmb file has a corresponding .fmx file. If a form fails to
    compile (there is no .fmx file), you may have to manually compile the form by
    launching the form builder tool (f60desm &).

4   All resulting .fmx files need to be moved to the <INSTALL_DIR>/toolset/bin
    directory. From the <INSTALL_DIR>/toolset/src directory, issue the following
    command:

```
> mv *.fmx ../bin
```

Compile toolset menus (*.mmb)

1   Change directories to <INSTALL_DIR>/toolset/src

2   Use the mmb2mmx45 script located in that directory to compile and generate the
    executable form menus with extension "mmx":

```
> chmod 755 mmb2mmx45 (if necessary).

> ./mmb2mmx45
```

3   Check to make sure each .mmb file has a corresponding .mmx file.

4   All resulting .mmx files need to be moved to the <INSTALL_DIR>/toolset/bin
    directory. From the <INSTALL_DIR>/toolset/src directory, issue the following
    command:

```
> mv *.mmx ../bin
```

Compile RMS forms libraries. (*.pll)

1   cd to **<INSTALL_DIR>**/rms/forms/src

2   Use the `pld2pll` script located in that directory to convert all files with a .pld extension to a .pll file.

```
> chmod 755 pld2pll (if necessary).
> ./pld2pll
```

3   Check to make sure all files with a .pld extension have a corresponding .pll file.

4   All resulting .pll files need to be moved to the <INSTALL_DIR>/rms/forms/bin directory. From the <INSTALL_DIR>/rms/forms/src directory, issue the following command:

```
> mv *.pll ../bin
```

5   Using your Form Builder session (f60desm &), navigate to <INSTALL_DIR>/rms/forms/bin/ and compile each form library in the <INSTALL_DIR>/rms/forms/bin/ directory:

   a   Choose File > Open.

   b   Select <INSTALL_DIR>/rms/forms/bin/<FILENAME>.pll.

   c   Click **OK**.

   d   Once the form library is "loaded", select the form library name, select Program, and choose Compile > All.

   e   After successful compilation, click **OK**.

   f   Save and close the form library.

The form libraries have been compiled and should now reside in the bin directory.

Compile RMS reference forms (fm_*.fmb)

1   Change directories to <INSTALL_DIR>/rms/forms/src

2   Move all reference forms (fm_*.fmb) from <INSTALL_DIR>/rms/forms/src to <INSTALL_DIR>/rms/forms/bin.

```
> mv fm_*.fmb ../bin
```

3   Using your Form Builder session (f60desm &), navigate to <INSTALL_DIR>/rms/forms/bin/ and compile each reference form in the <INSTALL_DIR>/ rms/forms/bin/ directory:

   a   Choose File > Open.

   b   Select <INSTALL_DIR>/rms/forms/bin/fm_<FILENAME>.fmb.

   c   Click **OK**.

   d   Once the reference form is "loaded", select the form name, select Program, and choose Compile > All.

   e   After successful compilation, click **OK**.

   f   Save and close the reference form.

The RMS reference forms have been compiled and should now reside in the bin directory.

Compile RMS forms (*.fmb)

1   Change directories to <INSTALL_DIR>/rms/forms/src.

2   Use the `fmb2fmx` script located in that directory to compile the forms.

```
> chmod 755 fmb2fmx (if necessary).
```

```
> ./fmb2fmx
```

3   Check to make sure each .fmb file has a corresponding .fmx file. If a form fails to compile (there is no .fmx file), you may have to manually compile the form by launching the form builder tool (f60desm &).

4   All resulting .fmx files need to be moved to the <INSTALL_DIR>/rms/forms/bin directory. From the <INSTALL_DIR>/rms/forms/src directory, issue the following command:

```
> mv *.fmx ../bin
```

Compile RMS menus (*.mmb)

1   Change directories to <INSTALL_DIR>/rms/forms/src.

2   Use the `mmb2mmx` script in that directory to compile the menus:

```
> chmod 755 mmb2mmx (if necessary).
```

```
> ./mmb2mmx
```

3   Check to make sure each .mmb has a corresponding .mmx file.

4   All resulting .mmx files need to be moved to the <INSTALL_DIR>/rms/forms/bin directory. From the <INSTALL_DIR>/rms/forms/src directory, issue the following command:

```
> mv *.mmx ../bin
```

Compile the reports library (*.pll)

1   Change directories to <INSTALL_DIR>/rms/reports/src

2   Move the file `rep25lib.pll` to the <INSTALL_DIR>/rms/reports/bin directory:

```
> mv rep25lib.pll ../bin
```

3   Using your Form Builder session (f60desm &), navigate to
    <INSTALL_DIR>/rms/reports/bin/ and compile the reports library in the
    <INSTALL_DIR>/rms/reports/bin/ directory:

    a   Choose File > Open.

    b   Select <INSTALL_DIR>/rms/reports/bin/rep25lib.pll.

    c   Click **OK**.

    d   Once the report library is "loaded", select the report library name, select
        Program, and choose Compile > All.

    e   After successful compilation, click **OK**.

    f   Save and close the report library.

The report libraries have been compiled and should now reside in the bin directory.

Compile reports (*.rdf)

1   Change directories to <INSTALL_DIR>/rms/reports/src

2   Use the rdf2rep script located in that directory to compile the reports:

    ```
    > chmod 755 rdf2rep (if necessary).
    > ./rdf2rep
    ```

3   Check to make sure each .rdf file has a corresponding .rep file.

4   All resulting .rep files need to be moved to the <INSTALL_DIR>/rms/reports/bin
    directory. From the <INSTALL_DIR>/rms/reports/src directory, issue the
    following command:

    ```
    > mv *.rep ../bin
    ```

5   Using your Form Builder session (f60desm &), navigate to
    <INSTALL_DIR>/rms/forms/bin/ and compile each form library in the
    <INSTALL_DIR>/rms/forms/bin/ directory:

    a   Choose File > Open.

    b   Select <INSTALL_DIR>/rms/forms/bin/<FILENAME>.pll.

    c   Click **OK**.

    d   Once the form library is "loaded", select the form library name, select
        Program, and choose Compile > All.

    e   After successful compilation, click **OK**.

    f   Save and close the form library.

The form libraries have been compiled and should now reside in the bin directory.

Compile RMS reference forms (fm_*.fmb)

1   Change directories to <INSTALL_DIR>/rms/forms/src

2   Move all reference forms (fm_*.fmb) from <INSTALL_DIR>/rms/forms/src to <INSTALL_DIR>/rms/forms/bin.

```
> mv fm_*.fmb ../bin
```

3   Using your Form Builder session (f60desm &), navigate to <INSTALL_DIR>/rms/forms/bin/ and compile each reference form in the <INSTALL_DIR>/ rms/forms/bin/ directory:

a   Choose File > Open.

b   Select <INSTALL_DIR>/rms/forms/bin/fm_<FILENAME>.fmb.

c   Click **OK**.

d   Once the reference form is "loaded", select the form name, select Program, and choose Compile > All.

e   After successful compilation, click **OK**.

f   Save and close the reference form.

The RMS reference forms have been compiled and should now reside in the bin directory.

Compile RMS forms (*.fmb)

1   Change directories to <INSTALL_DIR>/rms/forms/src.

2   Use the `fmb2fmx` script located in that directory to compile the forms.

```
> chmod 755 fmb2fmx (if necessary).
```

```
> ./fmb2fmx
```

3   Check to make sure each .fmb file has a corresponding .fmx file. If a form fails to compile (there is no .fmx file), you may have to manually compile the form by launching the form builder tool (f60desm &).

4   All resulting .fmx files need to be moved to the <INSTALL_DIR>/rms/forms/bin directory. From the <INSTALL_DIR>/rms/forms/src directory, issue the following command:

```
> mv *.fmx ../bin
```

Compile RMS menus (*.mmb)

1   Change directories to <INSTALL_DIR>/rms/forms/src.

2   Use the `mmb2mmx` script in that directory to compile the menus:

```
> chmod 755 mmb2mmx (if necessary).
```

```
> ./mmb2mmx
```

3   Check to make sure each .mmb has a corresponding .mmx file.

4   All resulting .mmx files need to be moved to the <INSTALL_DIR>/rms/forms/bin directory. From the <INSTALL_DIR>/rms/forms/src directory, issue the following command:

```
> mv *.mmx ../bin
```

Compile the reports library (*.pll)

1   Change directories to <INSTALL_DIR>/rms/reports/src

2   Move the file `rep25lib.pll` to the <INSTALL_DIR>/rms/reports/bin directory:

```
> mv rep25lib.pll ../bin
```

3   Using your Form Builder session (f60desm &), navigate to <INSTALL_DIR>/rms/reports/bin/ and compile the reports library in the <INSTALL_DIR>/rms/reports/bin/ directory:

   a   Choose File > Open.

   b   Select <INSTALL_DIR>/rms/reports/bin/rep25lib.pll.

   c   Click **OK**.

   d   Once the report library is "loaded", select the report library name, select Program, and choose Compile > All.

   e   After successful compilation, click **OK**.

   f   Save and close the report library.

The report libraries have been compiled and should now reside in the bin directory.

Compile reports (*.rdf)

1   Change directories to <INSTALL_DIR>/rms/reports/src

2   Use the rdf2rep script located in that directory to compile the reports:

```
> chmod 755 rdf2rep (if necessary).
```

```
> ./rdf2rep
```

3   Check to make sure each .rdf file has a corresponding .rep file.

4   All resulting .rep files need to be moved to the <INSTALL_DIR>/rms/reports/bin directory. From the <INSTALL_DIR>/rms/reports/src directory, issue the following command:

```
> mv *.rep ../bin
```

Appendix C – Sample NET 8 files for the server

listener.ora

Below is a sample listener.ora file.

Retek01 specifies the name of the server where the listener is located.

RETEK specifies the name of the Oracle instance that contains the Retek schema.

```
###############################################################
#  File:  listener.ora
#  Desc:  Oracle Net8 listener file.
###############################################################

CONNECT_TIMEOUT_LISTENER = 20
LOG_FILE_LISTENER = LISTENER.log
STARTUP_WAIT_TIME_LISTENER = 0
#--------------------------------------------------------#
# Valid trace levels are:  OFF | USER | ADMIN | SUPPORT  #
#--------------------------------------------------------#
TRACE_LEVEL_LISTENER = OFF
TRACE_FILE_LISTENER = LISTENER.trc
USER_PLUG_AND_PLAY_LISTENER = OFF
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (PROTOCOL_STACK =
        (PRESENTATION = TTC)(SESSION = NS)
      )
      (ADDRESS =
        (PROTOCOL = tcp)(HOST = Retek01)(PORT = 1521)
      )
      (ADDRESS =
        (PROTOCOL = IPC)(KEY = RETEK)
      )
    )
  )
#-----------------------------------------------------------------
-----#
```

```
# The following SID_LIST_LISTENER entry is required only if you
are    #

# connecting to an Oracle database version lower than 8.1.5.
#

#-----------------------------------------------------------------
-----#



SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = RETEK)
      (<ORACLE_HOME >= /files0/Oracle/product/8.0.5)
      (PRESPAWN_MAX = 99)
      (PRESPAWN_LIST =
        (PRESPAWN_DESC =
          (PROTOCOL = TCP)
          (POOL_SIZE = 0)
          (TIMEOUT = 1)
        )
      )
    )
  )



################################################################
#

#  Seperate listener process used to handle external procedure

#  calls.  All of the following entries are required and may
require

#  some changes to match your system.  Oracle suggests that the

#  LISTENER_EXTPROC be started by a Unix account other than
Oracle.

#
################################################################


CONNECT_TIMEOUT_LISTENER_EXTPROC = 20

LOG_FILE_LISTENER_EXTPROC = LISTENER_EXTPROC.log

STARTUP_WAIT_TIME_LISTENER_EXTPROC = 0

#-------------------------------------------------------#
```

```
# Valid trace levels are:  OFF | USER | ADMIN | SUPPORT  #
#-------------------------------------------------------#
TRACE_LEVEL_LISTENER_EXTPROC = OFF
TRACE_FILE_LISTENER_EXTPROC = LISTENER_EXTPROC.trc
USER_PLUG_AND_PLAY_LISTENER_EXTPROC = OFF


LISTENER_EXTPROC =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (PROTOCOL_STACK =
        (PRESENTATION = TTC)
        (SESSION = NS)
      )
      (ADDRESS =
        (PROTOCOL = tcp)(HOST = Retek01)(PORT = 1522)
)
      (ADDRESS =
        (PROTOCOL = IPC)(KEY = extproc_key)
      )
    )
  )



SID_LIST_LISTENER_EXTPROC =
  (SID_LIST =
    (SID_DESC =
      (PROGRAM = extproc)
      (GLOBAL_DBNAME = extproc_agent.world)
      (SID_NAME = extproc_agent)
      (<ORACLE_HOME >= /files0/Oracle/product/8.1.5)
      (PRESPAWN_MAX = 99)
    )
  )
```

tnsnames.ora

A tnsnames.ora file is required to connect to any Oracle database on your network. A sample tnsnames.ora is illustrated below. You will need to modify it appropriately to your environment. The extproc_connection_data entry is required along with the LISTENER_EXTPROC entry in the listener.ora file to allow Oracle to access a Unix shell library that is required by one of the stored procedures in the database.

Retek01 specifies the name of the server where the listener is located.

RETEK specifies the name of the Oracle instance that contains the Retek schema.

```
##############################################################
#  File:  tnsnames.ora
#  Desc:  Oracle Net8 TNS Names file.
##############################################################
RETEK =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = Retek01)(PORT = 1521))
    (CONNECT_DATA = (SID = RETEK))
  )


RETEK.WORLD =
(DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = Retek01)(PORT = 1521))
    (CONNECT_DATA = (SID = RETEK))
  )


EXTPROC_CONNECTION_DATA =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = IPC)(Key = extproc_key))
    (CONNECT_DATA = (SID = extproc_agent))
```

Appendix D – Database creation

The following is a sample script that creates the database necessary for the RMS 10.1.   Some new 9i features are being used like the UNDO tablespace and specifying the TEMP file at creation time. If you chose not to use this feature, be sure to create a temp and rollback tablespace.

Note that there are some outstanding Oracle bugs with the new 9i features. Research new features with Oracle prior to implementing. You may decide not to implement these new features.

Note that a different character set may be required for your database. Check with Oracle regarding compatibility of the character set with Developer 6i, as not all character sets will work with Developer 6i. All scripts following the database creation must be run.

Sample database create script

As the Oracle owner run all the following as sys.

```
startup nomount pfile=${ORACLE_HOME}/dbs/initRETEK.ora

create database "RETEK"

        maxdatafiles 1000

        character set UTF8

    datafile

        '/files0/oradata/RETEK/system01.dbf' size 100M
autoextend on next 100m maxsize 2000m

    logfile

       group 1 ('/files0/oradata/RETEK/redo1a.log') size
10M,

       group 2 ('/files0/oradata/RETEK/redo2a.log') size
10M,

       group 3 ('/files0/oradata/RETEK/redo3a.log') size
10M

default temporary tablespace temp tempfile
'/files0/oradata/RETEK/temp01.dbf' size 300M

undo tablespace undo_ts datafile
'/files0/oradata/RETEK/undo_ts01.dbf' size 300M;
```

Mandatory database setup scripts

Run the following scripts as Sys. (These are mandatory scripts.)

Install data dictionary views

```
@$ORACLE_HOME/rdbms/admin/catalog.sql

@$ORACLE_HOME/rdbms/admin/catproc.sql

@$ORACLE_HOME/rdbms/admin/catblock.sql – optional but
useful
```

Grant these privs to all

This is due to 9i security changes:

```
grant select_catalog_role to public;

grant execute_catalog_role to public;

grant execute on dbms_lock to public;

grant execute on dbms_rls to public;
```

The following should be run as system:

```
@$ORACLE_HOME/sqlplus/admin/pupbld.sql
```

Install XDK and XSU

```
ALTER SYSTEM SET "_system_trig_enabled"=FALSE
SCOPE=MEMORY;
```

Install Java objects

```
@$ORACLE_HOME/javavm/install/initjvm.sql
```

Install XML and XSU

```
@$ORACLE_HOME/rdbms/admin/initxml.sql
```

Create public synonyms and grants

```
CREATE PUBLIC SYNONYM XMLDOM for SYS.XMLDOM;

CREATE PUBLIC SYNONYM XMLPARSER for SYS.XMLPARSER;

CREATE PUBLIC SYNONYM XSLPROCESSOR for SYS.XSLPROCESSOR;

CREATE PUBLIC SYNONYM XMLTYPE for SYS.XMLTYPE;

GRANT EXECUTE ON XMLDOM TO PUBLIC;

GRANT EXECUTE ON XMLPARSER TO PUBLIC;

GRANT EXECUTE ON XMLTYPE TO PUBLIC;

GRANT EXECUTE ON XSLPROCESSOR TO PUBLIC;
```

Validate all invalid Java objects

```
spool javascript.sql


select '"alter java class "'||object_name||'" compile;'

from dba_objects

where object_type = 'JAVA CLASS' and owner = 'SYS' and
status = 'INVALID';


spool off

@javascript.sql
```

Appendix E – Retek Trade Management 10.1 system expectations

Install scripts

There is a set of install scripts needed to implement Retek Trade Management (RTM) 10.1. It is very important that these scripts be run in the order listed below. In between the `elc_comp_pre_htsupld.sql` and `elc_comp_post_htsupld.sql` scripts, run the HTS Upload program (`htsupld.pc`) to insert the HTS codes and associated data. The 9.0 release included the initial 2000 HTS tape from US Customs.

Note: All of the following can be run by running the installrtm_scripts_10.sql from the <INSTALL_DIR>/install/sqlplus directory.

1 `oga.sql`

2 `tariff_treatment.sql`

3 `quota_category.sql`

4 `country_tariff_treatment.sql`

5 `hts_headings.sql`

6 `elc_comp_pre_htsupld.sql`

7 Run `htsupld.pc` using the `htsXXXXin.1` file, which is supplied with the install scripts. (See the HTS Upload - Mass Update.doc document for a detailed description of the HTS Upload program and its Mass Update functionality.)

   a Check out the appropriate input file (for 2000 Customs Tape, 2000in.1, for 2001 Customs Tape, 2001in.1, etc.).

   b Refresh the program. Ask someone for help if you do not know how to do this.

   c Execute the program. If you do not know the appropriate input parameters, enter the program name (htsupld, with no '.pc') and press the Enter key. The needed inputs will be displayed. For htsupld the inputs are:

```
htsupld  userid/password@environment  <input file>
<reject file>   <import country>.
```

   For example:

```
htsupld  stjohnn/retek@dev8i.world  2000in.1   reject
US
```

   d Once the program has finished running, confirm if it completed successfully by checking the log. You should see the program name and that it terminated successfully.

```
tail $h/log/Feb_15.log    (instead of Feb_15, use the
current date).
```

e   If it did not run successfully, view the errors:

```
view $h/error/err.htsupld_1.Feb_15   (again, Feb_15
should be replaced with the current date)
```

Ask someone to help you if there are errors.

8   `elc_comp_post_htsupld.sql`

Important: After running this script, you must update the MPFXX and HMFXX components to have the appropriate rates. Currently for US, MPFUS will have a component rate of .025 and HMFUS will have a rate of .125.

These scripts are discussed in more detail below.

Install scripts for cost components

There are many Cost Components that are automatically inserted into the system via two scripts:

- elc_comp_pre_htsupld.sql

- elc_comp_post_htsupld.sql

Elc_comp_pre_htsupld.sql

This is the elc_comp script that is required for RMS and RTM. It is run in the basic install scripts one time only, at installation. It is *not* a refresh script. See the most recent `install_scriptsXX.sql` for the order in which it should be run. This script inserts the Expense Cost Components that are used to store totals and subtotals. These include:

- TEXPC (Total Expense Country) – This component is used to store the total of the expenses for each Item/Supplier/Origin Country/Lading Port/Discharge Port combination. It totals all expenses that have an Expense Type of 'Country' within the Item Expenses dialog. This total is added together with the Total Expense Zone, the Total Duty, and the Item's Unit Cost to come up with the Total Estimated Landed Cost of the Item/Supplier/Country combination. This component should not be modified.

- TEXPZ (Total Expense Zone) – This component is used to store the total of the expenses for each Item/Supplier/Zone/Discharge Port combination. It totals all expenses that have an Expense Type of 'Zone' within the Item Expenses dialog. This total is added together with the Total Expense Country, the Total Duty, and the Item's Unit Cost to come up with the Total Estimated Landed Cost of the Item/Supplier/Country combination. This component should not be modified.

- TEXP (Total Expense) – This component is used to store the total of the expenses for each Order/Item/Zone combination. It totals all expenses regardless of Expense Type within the Ordering dialog. This total is added together with the Total Duty and the Item's Order Cost to come up with the Total Estimated Landed Cost of the Order/Item combination. This component should not be modified.

- UNCST (Unit Cost) – This component is used to store the Unit or First Cost of each item. This only needs to be used if needed in the calculation of a component that needs be calculated as a percent of Unit Cost in addition to other components. For example, if Agent Commission is calculated as 3 percent of sum of Unit Cost and Freight, a CVB would be created that contained two details, UNCST and FREIGHT. Then Agent Commission would be created to be 3 percent of the CVB. This component should never have the In Exp., In Duty, or In ALC flags set, and should not be modified.

- ORDCST (Order Cost) – This component is used to store the Order Cost of each item. This only needs to be used if needed in the calculation of a component that needs be calculated as a percent of Order Cost in addition to other components. For example, if the Value for Duty is a sum of the Order Cost and Seller's Commission, a CVB called 'VFD' would be created that contained two details, ORDCST and SELLCOM. Then all duty components would be created to be a percentage of 'VFD'. This component should never have the In Exp., In Duty, or In ALC flags set, and should not be modified.

Elc_comp_post_htsupld.sql

This script is for the RTM product only. This needs to be applied only after all static install scripts have been run, oga, tariff_treatment, quota_category, country_tariff_treatment and hds_headings scripts have all been run followed by running the htsupld.pc program. The last step is running this script. This script will insert the Expense and Assessment Cost Components.   This script will need to be run once for each country of import that the client is using.

Note: This script is expecting two parameters to be passed in (the user will be prompted for the parameters). The first parameter is country ID, this is the Import Country. The second parameter is Currency Code, this is the code of the currency that corresponds to the entered Import Country. Most likely this script will be run using the Base Country and the Primary Currency as defined in the System Variables form.

The inserted components include:

- MPFXX (Merchandise Processing Fee XX) – This component is used to store Merchandise Processing Fee. In place of the 'XX' is the country code that is passed into the script. So if the Country is 'US', then there is one component created, 'MPFUS', with a description of 'Merchandise Processing Fee US'. This leaves the client with the ability to create additional MPF components for each of the countries that they intend to import into. This component is inserted with a Component Rate of 100 percent. This rate should be modified to be the appropriate rate for the Import Country. This component is also set up as an 'Always Default' which means that it will be defaulted to every Item/HTS combination.

- HMFXX (Harbor Maintenance Fee XX) – This component is used to store Harbor Maintenance Fee. In place of the 'XX' will be the country code that is passed into the script. So if the Country is 'US', then there will be one component created, 'HMFUS', with a description of 'Harbor Maintenance Fee US'. This leaves the client with the ability to create additional HMF components for each of the countries that they intend to import into. This component is inserted with a Component Rate of 100 percent. This rate should be modified to be the appropriate rate for the Import Country.

- TDTYXX (Total Duty XX) – This component is used to store the total of the duty for each Item/HTS or Order/Item/HTS combination. It totals all duties, taxes, and fees within the Ordering dialog. This total is added together with the Total Expense and the Item's Cost to come up with the Total Estimated Landed Cost of the Item or Order/Item combination. This component should not be modified.

- VFDXX (Value For Duty XX) – This Computation Value Base (CVB) is used to store the value that duty should be calculated from. In place of the 'XX' is the country code that is passed into the script. So if the Country is 'US', then there is one CVB created, 'VFDUS', with a description of 'Value for Duty US'. This leaves the client with the ability to create additional VFD CVBs for each of the countries that they intend to import into. Upon insert here, this CVB will only have one detail, which is 'ORDCST' (Order Cost). If the client needs additional expenses (we are making the assumption that only 'Expense' components will make up 'Value for Duty') to be used in the Value For Duty, they will need to be added to VFDXX through SQL Plus. All automatically inserted Assessment components with a Calculation Basis of 'Value' will have 'VFDXX' as their CVB.

- VFDXXXX (XX% of Value For Duty XX) – This component is used to store a percent of the CVB, Value For Duty. This is used in the case when I have an Item that is classified with multiple HTS codes. For example, a button-down shirt may have one HTS code for the cotton material that is 75 percent of the cost, and a second HTS code for the buttons that make up the other 25 percent. The duty components associated with the first HTS code would be need to be calculated from 75 percent of the entire Value for Duty. To accomplish this, the associated components would use 'VFD75XX' as their CVB instead of 'VFDXX'. The detail component would be 'VFD75XX' and would have a Component Rate of 75 and a CVB of 'VFDXX', therefore, the component 'VFD75XX' would be 75% of the Value for Duty. More generically speaking, 'VFDXXXX' will be the only detail in an inserted CVB called 'VFDXXXX', where the first 'XX' is replaced with the percentage.   In place of the second 'XX' will be the country code that is passed into the script. So if the Country is 'US', then there will be one component created, 'VFD25US', with a description of '25% of Value for Duty US'. This leaves the client with the ability to create additional VFD components for each of the countries that they intend to import into. The script will insert 'VFD25XX', 'VFD50XX', and 'VFD75XX', these are meant to be used as a guide if the client needs additional components with different percentages. These components should not be modified.

- DTYXXXX (DTYXXXX) – These components are used to calculate duty for each HTS code. In place of the first 'XX' is the HTS code's Duty Component Code concatenated with an 'A', 'B', or 'C' as needed for duty calculation. In place of the second 'XX' is the country code that is passed into the script. So if the Country is 'US', then there is one component created, 'DTYXXUS', with a description of 'DTYXXUS'. This leaves the client with the ability to create additional components for each of the countries that they intend to import into. The Import Country for these components will be the country code of the Base Country that is defined on the System Options table. This component is inserted with a Component Rate of 100 percent. This rate will be overwritten with the appropriate Tariff Treatment rate upon calculation within the Item and Ordering dialogs. These components should not be modified.

- DUTYXX(DUTYXX) – This component is used as a sub-total. In place of the 'XX' is the country code that is passed into the script. So if the Country is 'US', then there is one component created, 'DUTYUS', with a description of 'DUTYUS'. This leaves the client with the ability to create additional components for each of the countries that they intend to import into. It will contain the sum of all 'DTYXXXX' components each HTS code. This component will have a CVB called 'DUTYXX' that contains every 'DTYXXXX' component as its details. This component should not be modified.

- XXXXXXX (XXXXXXX) – Fees and Taxes are created using a concatenation of information. The Component ID consists of the Fee or Tax Class Code concatenated with the Fee or Tax Component Code, and an 'A' or 'B' as needed for calculation, and then the import country. For example, there is an existing Fee Class Code (also referred to as Fee Type) which is '053', its Fee Component Code is '1', and importing into the US, so there will be a component created that has an ID of '0531AUS'. The descriptions will be the same as the Component ID and can/should be modified to be clearer. Other than the description, these components should not be modified.

- ADXX (Anti-Dumping XX) – This component contains the Anti-Dumping charge for each Item/HTS code. In place of the 'XX' is the country code that is passed into the script. So if the Country is 'US', then there is one component created, 'ADUS', with a description of 'Anti-Dumping US'. This leaves the client with the ability to create additional components for each of the countries that they intend to import into. This component should not be modified.

- CVDXX (Countervailing Duty XX) – This component contains the Countervailing Duty charge for each Item/HTS code. In place of the 'XX' will be the country code that is passed into the script. So if the Country is 'US', then there will be one component created, 'CVDUS', with a description of 'Countervailing Duty US'. This component should not be modified.

HTS upload/mass update

There are several install scripts that must be run prior to HTS Upload to populate the following tables (note: these are one-time installs upon implementation of the product and must be maintained by the client):

- ELC_COMP, CVB_HEAD, CVB_DETAIL (via the elc_comp_pre_htsupld.sql script)

- QUOTA_CATEGORY (via the quota_category.sql script)

- OGA (via the oga.sql script)

- COUNTRY_TARIFF_TREATMENT (via the country_tariff_treatment.sql script)

- HTS_CHAPTER (via the hts_headings.sql script)

- TARIFF_TREATMENT (via the tariff_treatment.sql script)

After the initial load of the HTS data from executing the HTS Upload program. One additional install script must be run to populate the following tables with additional information:

- ELC_COMP, CVB_HEAD, CVB_DETAIL (via the elc_comp_post_htsupld.sql script)

The initial load of HTS information using a Customs provided tape and subsequent execution of the HTS Upload program will populate and update the following tables:

- HTS

- HTS_TARIFF_TREATMENT

- HTS_OGA

- HTS_FEE

- HTS_TAX

- HTS_TT_EXCLUSIONS

The following tables will need to be populated by the client, but will be updated via the HTS Upload program:

- HTS_AD

- HTS_CVD

- HTS_REFERENCE

The following tables will need to be populated and maintained by the client:

- HTS_CHAPTER_RESTRAINTS

Calculation of merchandise processing fee

This particular cost component is the only Cost Component that is calculated with a Min/Max Range for each Customs Entry. This range is defined on the MPF_MIN_MAX table (note: this table does not have a corresponding form and will need to be populated by the client via SQL Plus. In order to process MPF the MPF_MIN_MAX table must be populated for the import country or else the calculation function will error out during processing.). If a client does not use Merchandise Processing Fee, but has a similar component, they can use the MPF_MIN_MAX table and the MPFXX component to accomplish the same result. They simply need to change the Component Description and Rate. Within the Customs Entry dialog, MPFXX will be defaulted in along with all other assessments that are associated with each Order/Item combination. Once associated with the Entry, MPF will be recalculated and checked to see if the value falls within the Min/Max Range. If not, the value will be modified to be within the range and then allocated across all of the items on the Entry. Because this value is being calculated by the system, the user will not be allowed to modify the rate or value of any MPF components within the Customs Entry dialog.

Unit of measure conversions

The internal process that calculates and distributes MPF charges on-line will require Unit of Measure (UOM) conversions in multiple instances. If a particular UOM conversion is missing the processing will stop and a message will be displayed indicating that there is insufficient UOM information to continue. If this should occur, you must exit the dialog that generated the error add the missing conversion information and re-enter the dialog for the MPF charges to be processed.

Customs entry ref. status

There are 4 possible CE Ref. Statuses for each Customs Entry. They are 'Worksheet', 'Send', 'Downloaded', and 'Confirmed'. In general when an Entry is created it will be in 'Worksheet' status. Once all of the necessary information has been added, the user will set the Status to 'Send', indicating that the Entry is ready to be sent to the Broker. That night in the nightly batch run, the Entry will be downloaded to the Broker (cednld.pc). Once the download process is complete, the Status will automatically be set to 'Downloaded'; a user can never set the Status to this value manually. At that point once the user receives confirmation from the Broker, makes any necessary changes, and is sure that the information is correct, they can set the CE Ref. Status to 'Confirmed'. From that point on the Status cannot be changed, however most of the fields on the CE Header form will remain editable. All information on the CE Shipment form will be view only. Also, all information on the CE Order/Item form will be view only except for the Cleared Quantity, Cleared Quantity UOM, Apply button, and Comments fields. And finally all information in the CE Charges form will be view only as well.

Since some clients may prefer not to download their Entries to a Broker, the user will have the ability to set the CE Ref. Status from 'Worksheet' directly to 'Confirmed'.

Customs entry totals

- Total Duty contains the sum of the duty charges (any component beginning with 'DTY') for each item times the associated item's Manifest Item quantity, summed together for all items on the entry.

- Total Taxes contains the sum of the tax charges (any component beginning with a tax type (see attached document for a description of taxes)) for each item times the associated item's Manifest Item quantity, summed together for all items on the entry.

- Total Other contains the sum of all other charges (including fees) for each item times the associated item's Manifest Item quantity, summed together for all items on the entry.

- Total VFD contains the Value for Duty (which can be made up of order cost plus other dutiable expenses such as selling commission, royalties, etc.) times the associated item's Manifest Item quantity, summed together for all items on the entry.

- Total Est. Assessments contains the sum of the estimated duty/fees/taxes for each item, calculated from the Purchase Order/Item HTS Assessments, times the associated item's Manifest Item quantity, summed together for all items on the entry.

- Total Act. Assessments contains the sum of the Total Duty, Total Taxes, and Total Other values.