Retek Merchandising 9.0.0.0

Installation Guide

Retek Merchandising System 9.0.0.0

Retek Trade Management 9.0.0.0

Retek Sales Audit 9.0.0.0

Retek Invoice Matching 9.0.0.0

Retek Competitive Shopping 9.0.0.0

Retek API to Oracle Financials 9.0.0.0

RMS-9.0.0.0-IG

Retek Merchandising™

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

Copyright Notice

Copyright © 2000 Retek Inc.

All rights reserved.

No part of this documentation may be reproduced or transmitted in any form or by any means without the express written permission of Retek Inc., 801 Nicollet Mall, Suite 1100, Minneapolis, MN 55402.

Information in this documentation is subject to change without notice.

Trademarks

Retek Merchandising is a trademark of Retek Inc.

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

Printed in the United States of America.

Contents

Hardware and Software Requirements....1

Retek Merchandising System 2

Installation Instructions....6

Database Server Installation Instructions....6
Application Server Software Installation Instructions UNIX (Sun Solaris/ HPUX/AIX/SEQUENT)....15
Application Server Software Installation Instructions Windows NT 4.0 w/ServicePack 5 36
Upgrading from a Previous Release of the RMS (optional)....50
Data Dictionary Installation Instructions (optional) 53

Appendix A 54

Manual Instructions for Database Server Installation Process 54

Appendix B 62

Sample NET 8 Files for the Server....62

Appendix C 65

Customizing the Date Format....65

Appendix D 66

Troubleshooting Database Setup 66

Appendix E 68

Script (move_user.sql) to Create Private Synonyms 68

Appendix F....69

Makefiles and How to Use Them in the Retek Environment....69

Appendix G 79

Modules and their associated batch programs 79

Appendix H 84

Product security, Location security, and Price zone security 84

Appendix I 85

Financial Interface Installation Modification 85
Create Synonyms From Rms 9.0.0.0 To Oracle Financials Objects 91

Appendix J 97

Setting up multiple Retek Merchandising 9.0.0.0 Environments 97

Appendix K 98

Installation Instructions Retek Sales Audit 9.0.0.0 - Oracle Site Fuels Management 98

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 for your company's:

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

Database Server—including operating system, database software types and versions, and a list of hardware options.

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 support 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 8I – Enterprise Edition, with Partitioning, Bit Mapped Indexes and Fine Grained Access Control	Oracle Application Server (Web Server) Oracle Developer	Web Browser (IE or Netscape) JInitiator
Retek (RMS 9.0.0.0)	Batch Libraries DDL (Views, Triggers, Tables etc.) Database Objects (Procedures, Packages) Control Scripts Data Scripts	Forms Reports Toolset Help Files	

Hardware and Software Requirements

Because you need 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

Modules included in the Retek Merchandising System (RMS) are Retek Trade Management (RTM), Retek Sales Audit (ReSA), Invoice Matching (IM), Retek Competitive Shopping (CS), and Retek API to Oracle Financials (FIF).

Database Server

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

- Unix (or Unix variant) based OS
- ANSII compliant C compiler
- Perl Compiler 5.0 or later (Required for Trade Management Letter of Credit functionality).
- POSIX *if installing Sales Audit*
- Oracle RDBMS 8.1.5 Enterprise Edition with Partitioning (database patch 913504 required for Sun platforms)
- Oracle Pro*C Precompiler 8.x
- Oracle Net8

Sizing Factors and Other Suggestions to factor into your selection of a database server include:

- 1 GB RAM (dependent upon how much memory Oracle background processes consume)
- Tape backup
- CD-ROM drive
- Sufficient disk space to load:
  - Oracle for your platform (likely ~600MB)
  - RMS base tablespaces (~1 GB)

Hardware/OS Options for your database server include:

- Sun/Solaris 2.6
- IBM/AIX 4.3.x
- Hewlett Packard/HP UX 11.0
- Sequent/DYNIX 4.4.x

Application Server

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

- UNIX (or UNIX variant) or NT based OS
- Oracle Application Server (OAS) 4.0.8.1
- Oracle Developer Server 6.0
- x-Windows interface (only if UNIX OS)

Sizing Factors and Other Suggestions to factor into your selection of an application server include:

- ~40 MB RAM per client connection
- CD-ROM drive
- ~600 MB Free disk space for OAS
- ~1 GB Free disk space for Developer 6.0
- ~1 GB Free disk space for RMS forms, reports, gif files and help files.

Hardware/OS Options for your application server include:

- Sun/Solaris 2.6
- IBM/AIX 4.3.x
- Hewlett Packard/HP UX 11.0
- Microsoft/Windows NT 4.0 with ServicePack 5

Web Browser and Client Requirements

General Requirements for

Web Browsers:

- Must be certified to run with Oracle Application Server (OAS) 4.0.8.1
- Must run in conjunction with Oracle JInitiator 1.1.7.18

Client PCs:

- Have a Pentium Processor
- Use Windows 95, 98 or NT 4.0 with service pack 4 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 4.0 with Service Pack 1 (4.72.3110) or higher
- Netscape Navigator 4.61 or higher

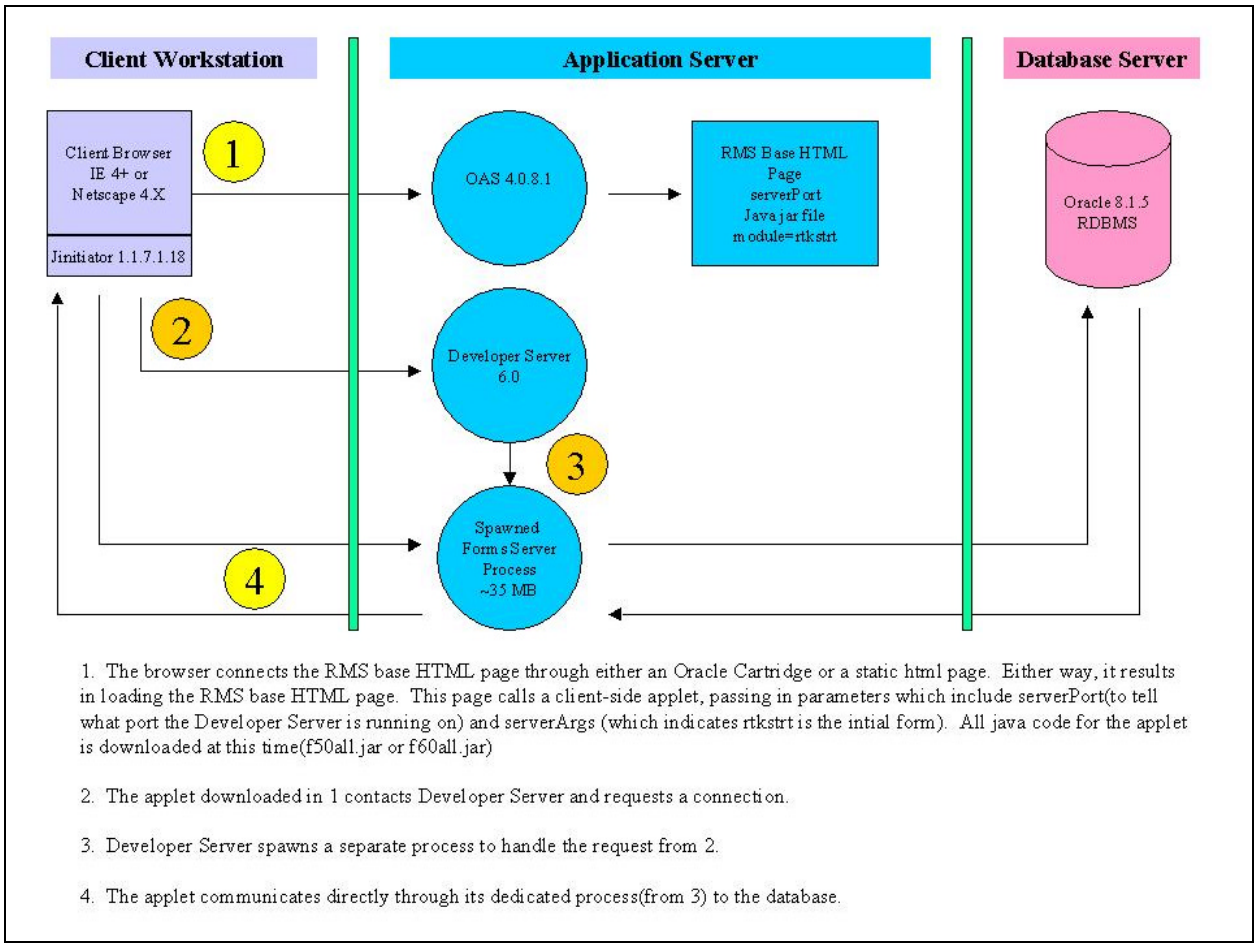
Supported OS/Browser Combinations for JInitiator version 1.1.7.18

Platform	IE 4.0 sp 4	IE 5.0	Navigator 4.61 or Higher
Windows 95*	Yes	Yes	Yes
Windows 98	N/A	Yes	Yes
Windows NT*	Yes	Yes	Yes

* Oracle JInitiator 1.1.7.18 does not support the use of the Microsoft Active Desktop.

MTS Web Architecture Overview

The following diagram depicts the architecture that allows Retek MTS products to be deployed in a three-tier environment:



1. The browser connects the RMS base HTML page through either an Oracle Cartridge or a static html page. Either way, it results in loading the RMS base HTML page. This page calls a client-side applet, passing in parameters which include serverPort(to tell what port the Developer Server is running on) and serverArgs (which indicates rtkstrt is the intial form). All java code for the applet is downloaded at this time(f50all.jar or f60all.jar)

2. The applet downloaded in 1 contacts Developer Server and requests a connection.

3. Developer Server spawns a separate process to handle the request from 2.

4. The applet communicates directly through its dedicated process(from 3) to the database.

Installation Instructions

Database Server Installation Instructions

Follow these steps to install the database server component of the RMS 9.0.0.0 software.

1. Create a UNIX user account.

   a. Create the following groups:

   - dba
   - rtk

   b. Create the following user, using ksh as the default shell:

   - retek - dba and rtk group

The retek user will install and compile the Retek Merchandising 9.0.0.0 Database Server and Application Server objects on UNIX systems.

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

After you have created your database (see Appendix D for sample database create scripts), add the following lines to each of the database initialization files (init.ora) that will be running the RMS. The database must be restarted after adding these lines to the *init.ora* file. See Appendix C for additional date format information.

- nls_date_format = "DD-MON-RR"
- job_queue_processes = <number of CPUs + 1>
- open_cursors=900

3. 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 9.0.0.0 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. You can log in to the server manager and query the USER_OBJECTS view to verify whether or not the packages have been created, and that Oracle user sys owns these packages. The source for these packages are located in the $ORACLE_HOME/rdbms/admin directory, and they can be re-created by running the catproc.sql script while logged into a server manager session as the Oracle user sys.

4. Create ORACLE tablespaces.

RMS 9.0.0.0 requires that two tablespaces be created initially for the RMS installation. The names of these tablespaces must be **retek_data** and **index_data**. These tablespace names are referred to in the table and index creation scripts, so their existence is required. We recommend that a separate tablespace be set up for rollback segments and another separate tablespace be set up for a temporary tablespace. 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: 400MB for **retek_data** and **index_data** – the system tablespace should have at least 50MB free for each installation of the RMS 9.0.0.0 schema. See Appendix D for sample database creation and tablespace creation scripts.

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

5. Create the file structure.

Note: Check to make sure there is enough disk space available when determining where RMS 9.0.0.0 will be installed. The RMS 9.0.0.0 Database Server file structure needs about 500 MB of space for install scripts, install modules and batch modules.

- Make sure your ORACLE_SID and ORACLE_HOME environment variables are set correctly before installation.

Log into UNIX as *retek* and mount the CD on the database server. There are five directories on the RMS 9.0.0.0 CD: appserverunix, appservernt, dbserverunix, d2000, CompShop. The dbserverunix directory contains the files for the database server install. Change directories to dbserverunix. Decide where you will be installing RMS database files.

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, run **builddb_90.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 9.0.0.0 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 the installation instructions.
- The program then copies a tarred and compressed file containing the database files to the specified directory on your server.
- Next, it 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 script then cleans up any extra files produced.
- Finally, it calls another script (install.rms) to finish the remainder of the Database Server installation. This is described in the next step.

Note: install.rms is called by the builddb_90.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.

6. Install RMS.

The install.rms script walks you through most of the manual processes in Appendix A. The script creates the Oracle schema owner for RMS 9.0.0.0 and utilizes all the information from the <INSTALL_DIR>/rms9000 structure. If additional RMS 9.0.0.0 environments are needed, refer to Appendix I for a short discussion on setting up multiple environments.

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 hitting <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 4.0.
- Creates all other database objects for RMS 9.0.0.0. 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 (rms9000), you will likely want to use the *developer* role. This role will grant access to all elements in the Retek Start Tree.
- Imports required data for multiview_default_40, multiview_saved_40 and rtk_errors RMS tables.
- **Optional:** Applies generic data found in the mmdemo script. This is data that provides fictitious demo style data that can be used to familiarize client staff with RMS 9.0.0.0 prior to complete implementation in your own environment.
- 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 9.0.0.0 components in a particular order. Use the <S> Skip option with caution!

7. Oracle Site Fuels Management API

If you are using the Site Fuels Management Oracle Application Interface, see the **Database Server Update** section of Appendix J. The Site Fuels Management Oracle Application Interface is the interface from Retek Sales Audit 9.0.0.0 to Oracle Applications Site Fuels Management (SFM) module. **Unless you have this Oracle Applications Module, you don't need to install these components; therefore, ignore this step and the instructions in Appendix J**.

8. Oracle Financials Interface Installation

In Appendix I, there are instructions to integrate Oracle Financials and RMS 9.0.0.0. Use these instructions to install the required objects into the Oracle Applications Environment.

After installing the required objects for the Oracle Financials, the newly created schema owner for RMS 9.0.0.0 must have access (that is, appropriate synonyms) to the Oracle Financials objects. If the newly created RMS user does not have these synonyms, some of the packages **will compile with errors.** See Appendix I for the list of objects that require synonyms from the RMS 9.0.0.0 owning schema to the Oracle Financials owning schema.

After verifying or creating the Oracle Financials Interface synonyms, create related packages, stored procedures and functions in the RMS 9.0.0.0 owning schema. On the database server, change directories to <INSTALL_DIR>/rms9000/oapps/orafin_107/rms. Log in to SQLPLUS as the RMS 9.0.0.0 schema owner and enter the following command:

SQL> **@startorafin107.pls**

When completed, exit out of SQLPLUS and view the spool file, CHECKOA107.LST file in the <INSTALL_DIR>/rms9000/oapps/orafin_107/rms directory to verify that no errors were found.

Note: Before you can upload vendor files from the financials system on-line, one change must be made to the stand40.pll library. See Appendix I for details.

9. Verify that all database objects are valid. Change directories to <INSTALL_DIR>/install/utility. Log into Oracle as the RMS 9.0.0.0 schema owner and type:

   SQL> @inv_obj_comp.sql

This script will recompile any invalid objects in the schema. You might want to run this script several times to validate all the objects.

10. Copy profiles for the RMS user (retek).

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

In this directory, you will find a sample profile for the initial install environment (rms9000) (see Appendix J for a short discussion on setting up multiple RMS 9.0.0.0 environments). The *rms9000* sample profile should be reviewed, edited for accuracy and be copied to *.profile* in the home directory of the RMS *rms9000* user (retek). As new environments and new users are created, use this sample profile as a guide to create the new user's profile.

After the profile has been copied and invoked, be sure to check the following settings:

- MMHOME (using the example above, MMHOME would be <INSTALL_DIR>/rms9000 for the retek user)
- MMUSER (RMS 9.0.0.0 schema owner associated with the rms9000 environment)
- PASSWORD (MMUSER [RMS 9.0.0.0 schema owner's] password)
- ORACLE_SID

Note that 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.

11. Compile the RMS batch libraries and programs:

Complete platform specific prep:

- Change directories to <INSTALL_DIR>/rms9000/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. Copy and rename the appropriate platform-specific make file for your UNIX system to platform.mk. If there isn't a platform-specific make file for your UNIX system, you will have to rename the platform_other.mk to platform.mk and note that you will need to make modifications to this file.
- Run the oramake script from the <INSTALL_DIR>/rms9000/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 type:

  >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.

Note for AIX users <u>ONLY</u>: AIX users must always have the environment variable LINK_CNTRL set equal to L_PTHREADS_D7 before compiling the batch. (export LINK_CNTRL=L_PTHREADS_D7 from the command line or in the profile of the environment owner). **AIX users must** edit the demo_rdbms.mk file and change the extproc_callback target **EVERY TIME oramake IS RUN**. The extproc_callback target calls the *nm* UNIX command with an input specification of $$1. This input specification must be changed to $$@. See example below:

```
extproc_callback: $(OBJS) $(LIBCLNTSH)
      generate_export_list() \
       { \
          /bin/nm -B -h -g "$$1" | grep -v ' U ' | awk '{print $$3}' | \
        egrep -v '^\.|^TOC' | sort | uniq ; \
       }; \
```

Must be changed to:

```
extproc_callback: $(OBJS) $(LIBCLNTSH)
       generate_export_list() \
       { \
          /bin/nm -B -h -g "$$@" | grep -v ' U ' | awk '{print $$3}' | \
        egrep -v '^\.|^TOC' | sort | uniq ; \
       }; \
```

Once the platform specific preparation is completed, you may compile the batch by using the batch_compiler script located in <INSTALL_DIR>/install:

cd to <INSTALL_DIR>/install and type

> batch_compiler

The batch_compiler script asks you a series of questions about what you wish to compile. Once it gathers the necessary information, it will echo the make commands for your set up and, with your approval, invoke make to compile the necessary libraries and source code. All output from the batch_compiler script is logged to <INSTALL_DIR>/install/logfiles.

The modules and their associated batch programs are listed in Appendix G.

12. Edit files with hard coded paths.

- Change directories to <INSTALL_DIR>/install/db_objects.
- Open scllib.pls and edit the path to point to where the libsupcstrr shared library resides (<$MMHOME>/oracle/lib/bin/libsupcstrr.so - i.e. /files1/rms_install/rms9000/oracle/lib/bin/libsupcstrr.so). Save and exit the file.

Note: The full path to the shared library must be indicated in this file. Also note that libsupcstrr has different extensions on different platforms (.so, .sl etc.)

- Log into sqlplus as the RMS 9.0.0.0 schema owner and start the corrected file:

  SQL> @scllib.pls

This change will invalidate two packages. They are: SCALE_CONSTRAINTS_SQL and ORDER_RECALC_SQL, which were built by callscl.pls and ordrclcs.pls respectively.

Revalidate these objects by using <INSTALL_DIR>/install/utility/inv_obj_comp.sql to revalidate them.

13. 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. Until that point, however, the following grant command can be used as a sample for what privileges should be granted to users:

SQL> grant create session, create table, create procedure, create cluster, 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>;

After users are set up, synonyms will need to be created to the *owner* schema for all tables, views, sequences, functions, procedures, and packages that the user will have access to. See Appendix E for a sample script that creates private synonyms for an Oracle user.

14. NET8 setup.

Log on to the server as *root* and move to the */etc* directory. Using a suitable editor, edit the *services* file. Search for the string "tnslsnr". If this string exists, take note of the socket number and quit the editor. If it doesn't exist, check to see if the socket 1521 is being used by another service. If this socket is being used, select a different socket number that is not being used. Then insert the following line into the *services* file:

"tnslsnr <socket #>/<protocol>"

For example, the default entry for Oracle running over TCP is:

"tnslsnr 1521/TCP"

If NET 8 already exists and is operational on the server, manually alter the files *listener.ora* and *tnsnames.ora* to reflect the sample files listed in Appendix B, taking special note of the following parameters:

- PROTOCOL
- HOST
- PORT
- SID

If your server is an RS6000, these files will be located in the */etc* directory. If your server is a SUN, these files will be located in the /var/opt/oracle directory.

If NET 8 does not exist on the server, see Appendix B for sample *listener.ora* and *tnsnames.ora* files that can be used as an example when creating these files.

Check to see whether the NET 8 listener is running by logging into UNIX as the user *oracle* and then issuing the command *"lsnrctl stat"*. If NET 8 is running, stop it by issuing the command *"lsnrctl stop"*.

Start the NET 8 listener by issuing the command *"lsnrctl start"*.

Application Server Software Installation Instructions
UNIX (Sun Solaris/ HPUX/AIX/SEQUENT)

Note: <INSTALL_DIR> is the directory where the RMS 9.0.0.0 files will be extracted from its tar file.

The <ORACLE_HOME > setting will be different when installing OAS 4.0.8.1 and Developer 6.0. In the documentation, these will be referred to as <DEV_ORACLE_HOME>(for the ORACLE_HOME used when installing Developer Server 6.0) and <OAS_ORACLE_HOME>(for the directory used as ORACLE_HOME when installing OAS 4.0.8.1). If using the machine where the database resides, please note that each <ORACLE_HOME > used for the installation of Web software should be different than the <ORACLE_HOME > (DBA_ORACLE_HOME) used when installing the database.

Example:

Setting for Developer 6.0 install:

DEV_ORACLE_HOME

<ORACLE_HOME> = /files0/oracle/product/d2k60

Setting for OAS install:
OAS_ORACLE_HOME
<ORACLE_HOME> = /files0/oracle/product/oas4081

HP-UNIX Note:

To ensure successful compilation of Oracle Forms on HP-UNIX, it may be necessary to consult Oracle to see if any kernel parameter settings are required.

1. Install Oracle Developer 6.0.

* Please note that 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.

- Log into the server as oracle and insert the Oracle Developer 6.0 Server software CD-ROM.
- Read the readme.txt file and ensure the server configuration meets Oracle's requirements.
- Start Oracle Installer.
- Select Custom Install and click OK.
- Select Install, Upgrade or De-install Software.

- Select Install New Product – Do not create DB objects.
- If prompted, run the preinstall script as root.
- Enter the domain when prompted for the domain name.
  Example: retek.com
- Check the displayed <ORACLE_HOME >to ensure it is where you want to install Developer. This will be referred to as <DEV_ORACLE_HOME> in the remainder of the installation guide.
- Click OK to accept log file location defaults.
- Install the following Developer 6.0 server edition options:

  Oracle Net8 Products 8.0.5.0.0

  Oracle Forms Web Server

  Oracle Graphics Web Server

  Oracle Reports Web Server

  Oracle UNIX Installer 4.0.3

  SQL*Plus 8.0.5

  TCP/IP Protocol Adapter
- Select Install.
- Choose Motif Bitmapped Interface and Web Graphics Interface when asked for Graphics.
- When prompted, choose all four Forms interface choices (Character Mode Interface, Motif Bitmapped Interface, Designer & Generator Executables, Web Forms Interface).
- Choose all four Reports interface choices (Character Mode Interface, Motif Bitmapped Interface, Reports Designer & Converter Executables, Web Reports Interface).
- When prompted, choose to relink executables.
- Log in as root, and run the file called root.sh in <DEV_ORACLE_HOME>/orainst directory.
- After the install completes successfully, you will need to check the Oracle Web site for the latest patch release for Developer 6.0. You will have to register with Oracle MetaLink if you haven't already. Go to http://metalink.oracle.com and register/log in. Select Patchsets on the left side of the page. Select Oracle Developer/2000 as the product. Download the latest patchset for your UNIX operating system from the site and apply it according to instructions in the readme.txt file accompanying the patch. Call Retek Customer Support before installing the patch to verify that it will run with RMS 9.0.0.0.

- You will also need to make an entry in the tnsnames.ora file for the Oracle database installed earlier in the installation process. The file is located at <DEV_ORACLE_HOME>/network/admin/tnsnames.ora. A sample entry is as follows:

```
DEV8.world =
(DESCRIPTION=
        (ADDRESS_LIST =
                (ADDRESS =
                        (PROTOCOL=tcp)
                        (host=mspvol01)
                        (Port=1521)
                )
        )
        (CONNECT_DATA=
                (SID =DEV8)
                (GLOBAL_NAME = DEV8.world)
        )
        )
```

Where DEV8 is replaced by the Oracle SID you need to connect to, mspvol01 is replaced by the server name of the machine the database is running on, and 1521 is replaced by the port number the Oracle Instance is listening on (1521 is the default). Once the tnsnames.ora file is edited, test your entry by connecting to the database through sqlplus.

Note *(Only for HP Install):* You may receive two error messages about linking the libraries during the installation. This is an Oracle Developer 6.0 bug. Click Ignore in response to the error messages and apply Oracle's bug fix after the installation is done.

The bug fix can be obtained by going to the Oracle Web site and searching on OracleDeveloper 6.0 Bug 925165. This must run after you have ignored the errors and finished the install. The documentation accompanying the bug fix will tell you what to do.

Compile RMS Oracle Forms and Reports

Setup

To compile the RMS 9.0.0.0 Oracle Forms, do the following:

1. Build the file structure.

Insert the RMS 9.0.0.0 CD-ROM, log in as user retek, and change directories to the appserverunix directory on the cd.

Determine where you want to install the RMS 9.0.0.0 application server files.

Note: RMS 9.0.0.0 application files require 1 GB of disk space.

Run the script buildapp_90.run. This will prompt you for the path where RMS 9.0.0.0 is to be installed. This will be referred to as <INSTALL_DIR> in the remainder of the documentation.

cd appserverunix

./buildapp_90.run

The resulting file structure located at <INSTALL_DIR> will contain directories for one RMS environment. The /rms9000 directory contains the RMS 9.0.0.0 source code. Additional environments can be created as necessary.

2. Set <ORACLE_HOME > to <DEV_ORACLE_HOME>

Example: export ORACLE_HOME=/files0/oracle/product/d2k60

3. Set and export your DISPLAY variable.

Example: export DISPLAY=10.1.2.153:0.0

4. Set the following variables.

export FORMS60_PATH=
<INSTALL_DIR>/toolset40/bin:<INSTALL_DIR>/rms9000/forms/bin

(fill in the appropriate values for <INSTALL_DIR> and <DEV_ORACLE_HOME>)

export UP=<db_user>/<db_user_password>@<oracle_db>

(the db_user should be the Retek RMS 9.0.0.0 oracle schema owner.)

Be sure that <DEV_ORACLE_HOME> bin is in the PATH. Also be sure to set the NLS_LANG variable.

Compile Toolset Libraries (.pll)*

cd to <INSTALL_DIR>/toolset40/src.

Use the *pld2pll40* script to convert all files in the directory from a file with a .pld extension to a file with a .pll extension.

Make sure that the script pld2pll40 is executable by doing an ls –l and checking its permissions. You can change permissions so it is executable with the command 'chmod 755 pld2pll40'.

Run the script to do the conversion.

./pld2pll40

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

Move all of the libraries (.pll files) in the <INSTALL_DIR>/toolset40/src directory to the <INSTALL_DIR>/toolset40/bin directory.

Start Form designer to compile all libraries for the RMS toolset. Log in to a command tool as oracle, set ORACLE_HOME and FORMS60_PATH variables (see steps 2 , 3, and 4 above) and *su* to retek before starting the tool. <DEV_ORACLE_HOME>/bin must be in your PATH setting.

f60desm &

A blue GUI interface will be displayed. Click Cancel at the welcome page.

Choose File, Connect. Log into the database as the Retek oracle schema owner.

Compile the libraries **in the following order**:

1. messge40.pll
2. ariiflib.pll
3. stand40.pll
4. calend40.pll
5. find40.pll
6. item40.pll
7. tools40.pll
8. mblock40.pll
9. mview40.pll
10. nav40.pll
11. work40.pll

For each file:

Choose File, Open and select <INSTALL_DIR>/toolset40/bin/FILENAME.pll before clicking OK.

Highlight FILENAME.pll, select Program, and choose Compile | All.

After successful compilation, click OK.

After all the toolset libraries listed above have been compiled (in the correct order), select Save from the File Menu.

The toolset forms libraries have been compiled. Exit the tool by selecting Quit from the File menu.

Compile Toolset Reference Forms (fm_*.fmb)

Start Form designer to compile all reference forms for the RMS toolset. You can start form designer by typing the following command (<DEV_ORACLE_HOME>/bin must be in your PATH setting).

f60desm &

A blue GUI interface will be displayed. Click Cancel at the welcome page.

Choose File, Connect. Log into the database as the Retek oracle owner.

Move all reference forms from <INSTALL_DIR>/toolset40/src to <INSTALL_DIR>/toolset40/bin.

Mv <INSTALL_DIR>/toolset40/src/fm_*.fmb <INSTALL_DIR>/toolset40/bin/.

For each file starting with fm_ having a .fmb extension in <INSTALL_DIR>/toolset40/bin, do the following:

Choose File, Open, and select <INSTALL_DIR>/toolset40/bin/FILENAME.fmb before clicking OK.

Highlight FILENAME.fmb, select Program, and choose Compile.

After successful compilation, click OK, save the form, and compile the next one.

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

Compile Toolset Forms (.fmb)*

cd to <INSTALL_DIR>/toolset/src. Use the *fmb2fmx40* script located in that directory to compile and generate the executable forms (fmx).

chmod 755 fmb2fmx40

./fmb2fmx40

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

All resulting .fmx files need to be moved to the <INSTALL_DIR>/toolset40/bin directory.

mv *.fmx ../bin

Compile Toolset Menus (.mmb)*

Compile all the menus in <INSTALL_DIR>/toolset40/src directory with a .mmb extension.

Use the *mmb2mmx40* script located in that directory to compile and generate the executable form menus with extension “mmx”.

chmod 755 mmb2mmx40

./mmb2mmx40

Each .mmb file should have a corresponding .mmx file.

All resulting .mmx files need to be copied to the <INSTALL_DIR>/toolset40/bin directory.

mv *.mmx ../bin

Exit out of Forms designer by selecting File, Quit.

Compile RMS Forms Libraries. (.pll)*

cd to <INSTALL_DIR>/rms9000/forms/src.

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

Run the script.

chmod 755 pld2pll

./pld2pll

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

Move all pll files to the bin directory:

mv *.pll ../bin

Open form designer by typing the following command:

f60desm &

Choose File, Connect. Log into the database as the Retek schema owner.

Open each file with a .pll extension in <INSTALL_DIR>/rms9000/forms/bin, and compile and save them.

Compile RMS Reference Forms (fm_*.fmb)

Move all reference forms (files that start with fm_ and have an .fmb extension) from <INSTALL_DIR>/rms9000/forms/src to <INSTALL_DIR>/rms9000/forms/bin.

Mv fm_*../bin

f60desm &

Open fm_*.fmb, compile them and save them.

Exit out of Form Designer by selecting File, Quit.

Compile RMS Forms (.fmb)*

cd to <INSTALL_DIR>/rms9000/forms/src. Use the *fmb2fmx* script located in that directory to compile the forms. Set up UP as mentioned before.

chmod 755 fmb2fmx

./fmb2fmx

This compiles the forms and should generate an executable with the extension .fmx for each form.

Compile RMS Menus (.mmb)*

Use the mmb2mmx script in directory <INSTALL_DIR>/rms9000/forms/src to compile the menus. You must be in the same directory as the script for the compilation to work. Set up UP as mentioned before.

chmod 755 mmb2mmx

./mmb2mmx

Move compiled objects to bin directory
Ensure that each .fmb file in <INSTALL_DIR>/rms9000/forms/src has a corresponding .fmx file.

Ensure that each .mmb file in <INSTALL_DIR>/rms9000/forms/src has a corresponding .mmx file.

If some of the .fmb or .mmb do not have corresponding .fmx or .mmx, open them with f60desm, and compile them individually, then save them after successful compilation.

Move all .fmx and .mmx files to the bin directory:
<INSTALL_DIR>/rms9000/forms/bin

cd <INSTALL_DIR>/rms9000/forms/src

mv *.fmx ../bin

mv *.mmx ../bin

Note: If you are using the Site Fuels Management Oracle Application Interface, please see the Application Server Update section of Appendix J to compile the Store Day Summary Form. This needs to be done in order to interface the Retek Sales Audit 9.0.0.0 with the Oracle Site Fuels Management (SFM) API.

Compile RMS Oracle Reports

Setup:

Log into a UNIX command tool as ORACLE and ensure that <ORACLE_HOME >is set to <DEV_ORACLE_HOME>.

Example: export ORACLE_HOME=/files0/oracle/product/d2k60

Set and export your DISPLAY variable to <IP Address>:0.0 .

Example: export DISPLAY=10.1.2.153:0.0

Set FORMS60_PATH and REPORTS60_PATH

export FORMS60_PATH=<INSTALL_DIR>/toolset40/bin:<INSTALL_DIR>/rms9000/forms/bin

export REPORTS60_PATH=<INSTALL_DIR>/rms9000/reports/bin=

Set your path variable to include the forms compiler executable f60desm.

```
export PATH=${ORACLE_HOME}/bin:${PATH}
```

Compile the Reports Library (.pll)*

Move the file rep25lib.pll to the bin directory:

mv <INSTALL_DIR>/rms9000/reports/src/rep25lib.pll <INSTALL_DIR>/rms9000/reports/bin/.

Run the command to bring up the forms builder.

```
f60desm &
```

A blue GUI interface will be displayed. Click Cancel at the welcome page.

Choose File, Connect. Log into the database as the Retek oracle owner.

Choose File, Open, and select <INSTALL_DIR>/rms9000/reports/bin/ rep25lib.pll before clicking OK.

Highlight rep25lib.pll, select Program, and choose Compile.

After successful compilation, click OK. From the File Menu, choose Save.

The reports library has been compiled. Exit the tool by selecting File, Quit.

Compile Reports (.rdf)*

Change directories to <INSTALL_DIR>/rms9000/reports/src

Run the script to compile the reports:

chmod 755 rdf2rep

./rdf2rep

All reports should now have a corresponding .rep file. Move the compiled reports to the bin directory:

mv *.rep ../bin

Install Oracle Application Server 4.0.8.1

* Please note that 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.

Log into the server as oracle and insert the Oracle OAS 4.0.8.1 software CD.

Start Oracle Installer by changing directory to the CD-ROM's mount directory (CD-ROM mount names may vary) and changing directory to the oracle/orainst directory. Run the following commands:
Verify that <ORACLE_HOME> is set to a different directory then what was used when Developer 6.0 was installed (this directory will be referred to as <OAS_ORACLE_HOME>).

```
echo $ORACLE_HOME

export TERM=vt100;
export DISPLAY=<IP Address>:0.0
./orainst /c
```

Select Oracle Application Server 4.0 Enterprise.

Select Custom Install.

Select Single Node.

For options, select the following:

- JWeb Cartridge
- LiveHTML Cartridge
- PLSQL Cartridge

Enter domain name if prompted.

Enter "website40" as the Site Name with a Boot Port of 2649. If port 2649 is in use, you will need to check with your administrator to determine an alternative port.

Enter the following Node Manager Listener Settings:

Port Number: 8888
User: admin
Password: <password>
Confirm: <password>

Enter a TCP Port Number of 8889.

Enter the following Web listener information:

Web Listener: www
Port Number: 80 (an alternative port of 8880 can also be used)

Note: If port 80 is chosen, it will be necessary for the root user to start and stop the listener. It is common to choose a port above 1024 so oracle can stop and start the Web server. If this is the option chosen, the URL used to access the Web listener will need to indicate the port entered above.

Example

If port 8880 is chosen as the Web listener for a Web server running on server mspvol01, the following URL will be used to access the Web listener.

http://mspvol01:8880

Entering http://mspvol01 will result in the browser thinking the default port of 80 was chosen as the listener.

Select the following JDBC Drivers:

JDK 1.1 JDBC OCI Driver
JDK 1.1 JDBC Thin Driver

The Web node administrator listener should be started by the installation process.

Enter the node administrator by opening a browser and going to URL http://<SERVER_NAME>:8888 .

Example: http://mspvol01:8888

Log in as user admin and supply the password you entered during installation

Configuring Your System To Run Web, Report, and Form Servers

Copy the following files at <INSTALL_DIR>/web_html /samplefiles/ to a temporary directory in your home directory

rmsweb_start – used to start OAS and Developer 6.0 server

rmsweb_stop – used to stop the OAS and Developer 6.0 server.

profile_rmsdev90 – Contains environment variable information

used by rmsweb_start.

Edit these scripts, replacing <DEV_ORACLE_HOME >with the <ORACLE_HOME >used during the installation of Developer 6.0, replacing <OAS_ORACLE_HOME> with the <ORACLE_HOME> used during the install of OAS and replacing <INSTALL_DIR> with the directory where the RMS 9.0.0.0 tar file was extracted.

rmsweb_start will be used to start the Web environment while rmsweb_stop will be used to shut down the middle tier. profile_rmsdev90 is used by both scripts to set environment variables necessary for execution. All three scripts should be placed in a common directory. The location of these files should be included in user oracle's PATH setting.

Install a Web Cartridge

Go to directory <INSTALL_DIR>/*web_html*. Verify that the following directories exist:

- temp
- log
- jinitiator
- gif
- reptemp
- help
- helpfiles

Using Netscape or IE to connect to as http://your_machine_name:8888 (See browser requirements for information on supported versions).

Note: you may have to use the full machine name such as mspows02.retek.com.

The OAS administrator listener should have started on port 8888 when OAS was installed. If attempts to connect fail, stop and start the OAS server using the scripts created above.

Enter the following user information:

User Name	Password
Admin	<password>

Click OAS Manager.

Click the Web site (website40).

Select HTTP Listeners.

Expand the www listener and select Network.

There will already be an entry for port 80(if selected) for the full machine name with domain. In order to access the Web server without using the domain name extension, follow the instructions below.

Make the following entry:

Address	Port	Security	Host Name	Base Directory
Any	80	Norm	<server name>	/

This will allow the Web server to be accessed by using the machine name without its domain.

Click Apply.

While the www listener is still expanded, select Directory and clear all the existing entries.

After the entries have been cleared, add the following virtual directories (use the exact case match):

File System Directory	Flag	Virtual Directory
<INSTALL_DIR>/web_html/	NR	/
<DEV_ORACLE_HOME>/forms60/java/	NR	/web_code/
<INSTALL_DIR> /web_html/	NR	/web_html/
<DEV_ORACLE_HOME>/forms60/java/	NR	/web_jars/
<INSTALL_DIR> /web_html/temp/	NR	/web_image/
<INSTALL_DIR> /web_html/gif/	NR	/web_gif/
<INSTALL_DIR>	NR	/english/

File System Directory	Flag	Virtual Directory
/web_html/helpfiles/english/		
<INSTALL_DIR> /web_html/jinitiator/	NR	/jinitiator/
<INSTALL_DIR> /web_html/temp/	NR	/reptemp/
<INSTALL_DIR> /web_html/temp/	NR	/temp/

Click Apply. A dialog box informs you that the changes were applied successfully.

Add the Forms Application

Click Applications. A page comes up showing all of the applications configured on the OWS. Click "+" to add a new application.

From the drop-down box, select Cweb, select "Manually" as the Configure Mode, and click "Apply."

Enter the following information and click Apply:

Application Name	RMS
Display	RMS
Version	9.0

A dialog box will display, telling you that the application has been added successfully. Click Add Cartridge to This Application.

Enter the following information:

Cartridge Name: RMSDEV

Display Name: RMSDEV

Shared Object: <*DEV_ORACLE_HOME*>/lib/f60webc.so

Note: The extension for the ifwebc60 library may be different across platforms.

To determine the correct shared object name, change directories to <DEV_ORACLE_HOME>/lib and run the following command:

ls f60webc.*

Cartridge Entry Point: form_entry

Virtual Path: /RMSDEV

Leave the default entry for physical path and click Apply. A dialog box displays telling you that the cartridge was added successfully. Click OK.

Set the Forms Cartridge Parameters:
Set the RMSDEV cartridge parameters by going to the OAS Tree and clicking Applications, RMS, Cartridges, RMSDEV, Configuration, Cartridge Parameters. Enter the following parameters:

baseHTML <INSTALL_DIR>/web_html/rmsdev90.html
HTMLdelimiter %
forms_param module=rtkstrt

The parameters are case sensitive – make sure that your entries match the ones above exactly before you click Apply. The rmsdev90.html page will be created in miscellaneous tasks later.

Configuring the Java Cartridge

Create the Java application
On the OAS Tree, click Application, When the Application status page comes up, click + to add an application.

Select JWeb as the application type.

Select Manually.

Click Apply.

Enter the following information:

Application Name: JavaApps

Display Name: JavaApps

Application Version: 9.0

Click apply. When the dialog box appears, click "Add Cartridge to Application."

Enter the following information:

Cartridge Name: rhelp

Display: rhelp

Location of Class files: <INSTALL_DIR>/web_html/help

Virtual Path: /java/help

Click Apply, and when the dialog box is displayed indicating success, click OK.

Add the Reports Cartridge

In the OAS Tree, Click Applications and, when the Applications status page appears, click "+" to add an application.

Enter the application type of "CWeb", select "Add Manually" and click Apply.
Enter the following information:

Application Name: RMSReports

Display Name: RMSReports

Version 9.0

Click Apply and, when the success window shows, select "Add Cartridge to Application".

Enter the following cartridge information:

Cartridge Name: RMSRPT90
Shared Object Path: <DEV_ORACLE_HOME>/bin/rwows60.so

The extension for the rwows60 library may be different across platforms. To determine the correct shared object name, change directories to <DEV_ORACLE_HOME>/bin and run the following command:

ls rwows60.*

Cartridge Entry Point: rwows_start
Virtual Path: /rwows

Leave the Physical Path to its default setting and click Apply and, when a dialog box is displayed indicating success, click OK.

Set the Report Cartridge Parameters

In the OAS Tree select Applications, RMSReports, Cartridge, RMSRPT90, Configuration. Enter the following Cartridge parameters.

REPORTS60_SHARED_CACHE NO
REPORTS60_VIRTUAL_MAP /reptemp
REPORTS60_PHYSICAL_MAP <INSTALL_DIR>/web_html/reptemp

Click Apply. Then,when the dialog box is displayed indicating success, click OK

Miscellaneous Tasks

Make the following entries in the Oracle database

update lang set webhelp_server='http://<SERVER_NAME>:<PORT>', reports_server= 'repserver.world", webreports_server='rwows' where lang = 1;

Update the tnsnames.ora file

The tnsnames.ora files at <DEV_ORACLE_HOME>/network/admin and <OAS_ORACLE_HOME>.network/admin need to be updated to reflect the new Oracle Components.

Add an entry for the reports server to each tnsnames.ora file

repserver.world =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = mspvol01)(PORT = 1949)))

where repserver.world matches the entry made in the lang table above and mspvol01 is replaced by your server name. Port can be changed to any port approved by your network administrator.

Edit the netscape_11718.html file

The file is located in <INSTALL_DIR>/web_html/jinitiator.
Fill in the correct values for <SERVER> and <PORT> and save the file. This file will allow jinitiator to be dynamically installed on clients when accessed for the first time.

Modify the following file

<DEV_ORACLE_HOME>/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 a configuration file to the forms admin directory

Copy the file fmrweb.res, found in <INSTALL_DIR>/web_html/samplefiles, to <DEV_ORACLE_HOME>/forms60/admin/resource/US/.

cp <INSTALL_DIR>/web_html/samplefiles/fmrweb.res
<DEV_ORACLE_HOME>/forms60/admin/resource/US/.

Create the baseHTML page for the forms server

When configuring the forms cartridge, we included a baseHTML parameter of

<INSTALL_DIR>/web_html/rms90dev.html. Now we need to create the page and put it in its proper location. Get the shell from <INSTALL_DIR>/web_html/samplefiles/formbase.html and copy (rename) it to <INSTALL_DIR>/web_html/rmsdev90.html.

Edit the block of code in the first section of the file to reflect your environment.

Do the following

Your environment's form server is listening on a port indicated in rmsweb_start. Set the port number variable in the file to the same number.(default was 9010)

```
//Begin Customer Settings
//  The following variables need to be set to match your RMS Environment
//  port_no should be set to the port that the Developer Server is running on
//  This corresponds to the serverPort setting in the old HTML Page.
var port_no = '9010';
```

The next setting is the name of the initial form that is called when RMS is started. Leave this set to rtkstrt.

```
//  startup_form represents the initial RMS form being viewed - this is
//  rtkstrt in RMS and does not need to be changed.
var startup_form = 'rtkstrt';
```

Finally, if desired, set the splash screen for the application. If no splash screen is desired, leave as is.

```
//  splash_screen represents the name of the file containing your company's
//  splash screen.  This is a .gif file located in this directory.
var splash_screen = 'rtksplash.gif';


//End Customer Settings
```

Save rmsdev90.html.

Install the Oracle JInitiator Component on the Server

JInitiator 1.1.7.1.8 is included on the RMS9.0 file structure in the directory <INSTALL_DIR>/web_html/jinitiator. Check to make sure the file jinit11718.exe is there.

Browser requirements

You will need IE 4.0 with SP1 (or higher version) or Netscape 4.6 (or higher version) as your Web browser to use RMS 9.0.0.0.

If you are using IE 4.0 with SP1 (check your version by opening IE, select HELP, choose ABOUT INTERNET EXPLORER), you need to do this to check your configuration to ensure JInitiator will work:

Select VIEW, choose INTERNET OPTIONS, pick ADVANCED. Then Make sure the following options are checked:

Browsing

- Notify when downloads complete
- Disable script debugging
- Launch Channels in full screen window
- Use AutoComplete
- Use smooth scrolling
- Enable page transitions
- Browse in a new process
- Enable page hit counting
- Enable scheduled subscription updates
- Show Internet Explorer on the Desktop (requires restart)
- Update links – Always

Multimedia

- Show pictures
- Play animations
- Play videos
- Play sounds
- Smart image dithering

Security

- Enable Profile Assistant
- SSL 2.0
- SSL 3.0
- Warn if forms submit is being redirected
- Warn if changing between secure and not secure mode
- Warn about invalid site certificates
- Cookies - Always accept cookies

Java VM

- Java console enabled (requires restart)
- Java JIT compiler enabled

Searching

- Autoscan common root domain
- Search when URL files - Always ask

HTTP 1.1. Settings

- Use HTTP 1.1

Then click APPLY, and OK.

IE5.0 default options work with JInitiator.

Test the System

Run rmsweb_stop then run rmsweb_start to bounce the Web processes. Connect the client to the server by issuing:

http://<your_machine>:<selected port if not 80>/RMSDEV90

Note: "RMSDEV90" is case sensitive, and should match the cartridge name exactly.

The first time that you connect to the server, the *jinitiator* will download and install. Select the checkbox indicating you trust files from Oracle. The *jinitiator* download will occur the first time that each machine accesses the Web server. After the *jinitiator* is completely installed, close the browser and open a new browser session to use *jinitiator*.

Application Server Software Installation Instructions Windows NT 4.0 w/ServicePack 5

Note: INSTALL_DIR is the directory where the RMS9.0 file has been extracted.

<DEV_ORACLE_HOME >is the <ORACLE_HOME> used when Developer 6.0 is installed.

<OAS_ORACLE_HOME >is the <ORACLE_HOME> used when OAS 4.0.8.1 is installed AND is different than <DEV_ORACLE_HOME>.

Log into the server as administrator and insert the Oracle Developer 6.0 Server software CD.

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 case, use Oracle's default setting or consult Oracle support.

Start Oracle Installer by double-clicking the setup file. Select English. Enter Company name and set your Oracle Home to the directory where you wish to install Developer 6.0. This directory should be the default ORACLE_HOME of <drive>:\orant .

Note: The <ORACLE_HOME> used to install Developer will be referred to as <DEV_ORACLE_HOME> in the remaining documentation. A different <ORACLE_HOME> must be used during the installation of Oracle Application Server (<OAS_ORACLE_HOME>).

Select "Custom Installation."

Install Developer 6.0 server edition by installing the following:

- GUI Common Files 6.0.5.6.0
- Oracle Developer Forms 6.0.5.2
- Oracle Developer Graphics 6.0.5.8
- Oracle Developer Reports 6.0.5.28.0
- Oracle Installer 3.2.2.1.1
- Oracle Net8 Products 8.0.5.0.0
- Required Support Files 8.0.5.1.0
- SQL*Plus 8.0.5
- System Support Files 6.0.5
- Tools Utilities 6.0.5.5

Select “Install.”

If prompted that some of the options selected for installation will require files from Microsoft, select “OK” and continue with the installation. Agree to changes to PATH setting if prompted.

The installer will ask where you want to install files. If <DEV_ORACLE_HOME >was set to D:\ORANT, the locations will be as follows:

Forms	D:\ORANT\FORMS60
Graphics	D:\ORANT\GRAPH60
Reports	D:\ORANT\REPORT60
Project Builder	D:\ORANT\PJ60
Online Documents	D:\ORANT\OIN

After the install completes successfully, you need to check the Oracle Web site for the latest patch release for Developer 6.0. Download the latest patchset for Windows NT from the site and apply it according to instructions in the readme.txt file accompanying the patch. Call Retek Customer Support before installing the patch to verify its compatibility with RMS 9.0.0.0.

THIS MUST BE DONE BEFORE INSTALLING OAS 4.0.8.1 !

You will also need to make an entry in the tnsnames.ora file for the Oracle database installed earlier in the installation process. The file is located at <DEV_ORACLE_HOME>\Net80\admin\tnsnames.ora. A sample entry would be:

DEV8.WORLD=

(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(host=**mspvol01**)(Port=**1521**))

(CONNECT_DATA=(SID=**DEV8**)(srvr=dedicated)))

Where DEV8 is replaced by the Oracle SID you need to connect to, mspvol01 is replaced by the name of the machine the database is running on, and 1521 is replaced by the port number the Oracle SID Listener is listening on (1521 is the default).

Installation of Developer 6.0 is complete!

Install Oracle Web Server 4.0.8.1

Before proceeding with this installation, the patch to Developer 6.0 must be installed. Do not proceed until this is complete.

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 case, use Oracle's default setting or consult Oracle support.

Stop any Oracle processes that are running on the server. While logged on as administrator, click the setup.exe file on the OAS 4.0.8.1 CD. Enter Company Name and enter an <ORACLE_HOME> where you wish to install OAS (it MUST be different than the <ORACLE_HOME> used when installing Developer!) . This will be referred to as <OAS_ORACLE_HOME> in the remainder of the documentation.

Select English as the language and, from the list of products in the window on the left, select Oracle Application Server 4.0.8.1 and click Install.

Verify that <ORACLE_HOME> is set to a different directory from which Developer 6.0 was installed.

Select Oracle Application Server 4.0 Enterprise.

Select "Custom Install."

Select Single Node.

For options, select the following:

- JWeb Cartridge
- LiveHTML Cartridge
- PLSQL Cartridge

Enter domain name when prompted.

Enter "website40" as the Site Name with a Boot Port of 2649. If port 2649 or any of the other ports listed below are in use, you need to check with your NT administrator to determine an alternative port.

You may be asked for a Multicast port number and told to record this for future use. Multicast is not used by RMS so you can accept Oracle's default settings and continue with the installation.

Enter the following Node Manager Listener Settings:

Port Number:	8888
User:	admin
Password:	<password>
Confirm:	<password>

Enter a TCP Port Number of 8889 for the Administration Utility Listener.

Enter the following Web listener information:

Web Listener:	www
Port Number:	80 (an alternative port of 8880 can also be used)

Select the following JDBC Drivers:

JDK 1.1 JDBC OCI Driver

JDK 1.1 JDBC Thin Driver

A dialog box is displayed informing you that installation of OAS was successful.

Reboot the Server. The OAS Server starts up automatically. If it does not start up automatically, go to Start, Run and type the following commands:

owsctl start –nodemgr	Starts the node manger.
owsctl start	Starts the OAS service

Enter the node administrator by opening a browser and going to URL

http://SERVER_NAME:8888 .
Example: http://mspvol01:8888

Log in as user admin and supply the password you entered during installation.

Configuring Your System To Run Web, Report, and Form Servers

Insert the RMS 9.0.0.0 cd and change directories to appservernt. Double-click on the appserver.exe file and, when prompted, enter the directory where RMS 9.0.0.0 is to be installed. This will create the RMS file structure.

Copy the following files at <INSTALL_DIR>\web_html \samplefiles to a temporary directory on your hard drive (you may need to create the directory c:\Temporary).

- rmsdev90form.bat
- oas_start.bat
- oas_stop.bat

Copy the sample oas_start.bat file to the *startup* directory of the all users profile under WinNT (The startup directory is at C:\Winnt\Profiles\All Users\StartMenu\Programs\Startup).

The *oas_start.bat and oas_stop.bat scripts* do not need to be edited. Their text is below – move oas_stop.bat from the temporary directory to a permanent location and create a shortcut to oas_stop.bat by right-clicking your mouse when the file is selected and choosing Create Shortcut. This script is used to shut down the OAS server.

oas_start.bat:

owsctl start –nodemgr

owsctl start

oas_stop.bat

owsctl stop

owsctl stop –nodemgr

In the temporary directory, modify the rmsdev90form.bat script to fit your system.

rmsdev90form.bat

set forms60_path=RMS_TOOLSET_BIN;RMS_FORMS_BIN

*ifsrv60 port=**9010** log=**INSTALL_DIR\web_html\log\rmsint90.log***

where ***RMS_TOOLSET_BIN*** points to the directory where the RMS 9.0.0.0 toolset resides (<INSTALL_DIR>\toolset40\bin) and ***RMS_FORMS_BIN*** points to the RMS forms bin directory (<INSTALL_DIR>/rms9000\forms\bin).

The log can be put wherever desired.

The **port** number will match up with the port number set in the baseHTML page created below.

Copy the modified rmsdev90form.bat file to the *startup* directory of the all users profile under WinNT (The startup directory is at C:\Winnt\Profiles\All Users\StartMenu\Programs\Startup). This will cause the forms server to start when the system is booted.

Open your Start----Settings—Control panel—System. Under the performance tab, set the boost to None.

Install a Web Cartridge

Go to directory <INSTALL_DIR>*web_html*. Verify that the following directories exist:

- temp
- log
- jinitiator
- gif
- reptemp
- help
- helpfiles

Using your browser (see Browser Requirements section below) go to http://your_machine_name:8888 (Note you may have to use the full machine name such as mspows02.retek.com). The OAS administrator listener started on port 8888 when the machine was restarted above. If attempts to connect fail, stop and start the OAS server using the scripts created above.

Enter the following user information:

User Name	Password
Admin	<password>

Click OAS Manager.

Click the Web site (website40).

Select HTTP Listeners.

Expand the www listener and select Network.

There will already be an entry for port 80(if selected) for the full machine name with domain. In order to access the Web server without using the domain name extension, follow the instructions below.

Make the following entry:

Address	Port	Security	Host Name	Base Directory
Any	80	Norm	<server name>	/

This will allow the Web server to be accessed by using the machine name without its domain.

Click Apply.

While the www listener is still expanded, select Directory and clear all the existing entries.

After the entries have been cleared, add the following virtual directories (use the exact case match):

File System Directory	Flag	Virtual Directory
<INSTALL_DIR>\web_html\	NR	/
<DEV_ORACLE_HOME>\forms60\java\	NR	/web_code/
<INSTALL_DIR>\web_html\	NR	/web_html/
<DEV_ORACLE_HOME>\forms60\java\	NR	/web_jars/
<INSTALL_DIR>\web_html\temp\	NR	/web_image/
<INSTALL_DIR>\web_html\gif\	NR	/web_gif/
<INSTALL_DIR>\web_html\helpfiles\english\	NR	/english/
<INSTALL_DIR>\web_html\jinitiator\	NR	/jinitiator/
<INSTALL_DIR>\web_html\temp\	NR	/reptemp/
<INSTALL_DIR>\web_html\temp\	NR	/temp/

Click Apply. A dialog box informs you that the changes were applied successfully.

Adding the Forms Application

Click Applications. A page should come up showing all of the applications configured on the OWS. Click "+" to add a new application.

From the drop-down box, select "CWeb Application", select "Manually" as the Configure Mode, and click "Apply."

Enter the following information and click Apply:

Application Name	RMS
Display	RMS
Version	9.0

A dialog box is displayed telling you that the application has been added successfully. Click Add Cartridge to This Application.

Enter the following information:

- Cartridge Name: RMSDEV
- Display Name: RMSDEV
- Shared Object: <*DEV_ORACLE_HOME*>\bin\ifwebc60.dll
- Cartridge Entry Point: form_entry
- Virtual Path: /RMSDEV

Leave the default entry for physical path and click Apply. A dialog box displays, telling you that the cartridge was added successfully. Click OK.

Set the Forms Cartridge Parameters

Set the RMSDEV cartridge parameters by going to the OAS Tree and clicking Applications, RMS, Cartridges, RMSDEV, Configuration, Cartridge Parameters.

Enter the following parameters:

- baseHTML <INSTALL_DIR>\web_html\rmsdev90.html
- HTMLdelimiter %
- forms_param module=rtkstrt

The parameters are case sensitive – make sure that your entries match the ones above exactly before you click Apply. The rmsdev90.html page will be created in miscellaneous tasks later.

Configure the Java Cartridge

Create the Java application

On the OAS Tree, click Application, when the Application status page appears, click + to add an application of type JWeb.

Enter the following information:

- Application Name: JavaApps
- Display Name: JavaApps
- Application Version: 9.0

Click Apply. When the dialog box indicating success is displayed, click Add Cartridge to Application.

Enter the following information:

- Cartridge Name: rhelp
- Display: rhelp
- Location of Class files: <INSTALL_DIR>\web_html\help\
- Virtual Path: /java/help

Click Apply, and when the dialog box is displayed indicating success, click OK.

Add the Reports Cartridge

In the OAS Tree, Click Applications and, when the Applications status page appears, click + to add an application.

Enter the application type of CWeb, select Add Manually and click Apply. Enter the following information:

- Application Name: RMSReports
- Display Name: RMSReports
- Version 9.0

Click Apply and, when the success window shows, select Add Cartridge to Application.

Enter the following cartridge information:

- Cartridge Name: RMSRPT90
- Display Name: RMSRPT90
- Cartridge Shared Object: <DEV_ORACLE_HOME>\bin\rwows60.dll
- Cartridge Entry Point: rwows_start
- Virtual Path: /rwows

Leave the Physical Path to its default setting and click Apply and, when a dialog box is displayed indicating success, click OK.

Set the Report Cartridge Parameters
In the OAS Tree, select Applications, RMSReports, Cartridge, RMSRPT90, Configuration. Enter the following parameters.

- REPORTS60_SHARED_CACHE NO
- REPORTS60_VIRTUAL_MAP /reptemp
- REPORTS60_PHYSICAL_MAP
  <INSTALL_DIR>\web_html\reptemp

Click Apply. Then, when the dialog box is displayed indicating success, click OK.

Miscellaneous Tasks

Make the following entries in the Oracle database

update lang set webhelp_server='http://<SERVER_NAME>:<PORT>', reports_server= 'repserver.world', webreports_server='rwows' where lang = 1;

Update the tnsnames.ora file

The tnsnames.ora files at <DEV_ORACLE_HOME>\Net80\admin and <OAS_ORACLE_HOME> need to be updated to reflect the new Oracle Components.

Add an entry for the reports server.

```
repserver.world =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = TCP)(HOST = mspvol01)(PORT =
1949)) )
```

where repserver.world matches the entry made in the lang table above and mspvol01 is replaced by your server name. Port can be changed to any port approved by your network administrator. Port 1949 is the default.

Set up the Reports Server to run as a service

Open a command tool and type the following command to install the report server as an NT service:

rwmts60 –install repserver.world tcpip

When asked whether you want to install repserver.world as a service, answer Yes.

Go to Control Panel|Settings|Services and ensure that the Oracle Reports Server was added.

There is an issue caused by installing Developer and OAS in different ORACLE_HOMEs

To work around the issue, there needs to be an entry under the OAS Oracle Home (HOME1) that points to the reports60 directory in the Developer Oracle Home (HOME0). To achieve this, the user must manually create an entry under the OAS Oracle Home (HOME1) with the following properties:

- Entry type: String
- Entry name: RW60
- Entry value: c:\orant\report60

Run regedit and make the entry at HKEY_LOCAL_MACHINE|SOFTWARE|ORACLE|HOME1

(verify that HOME1 is the OAS_ORACLE_HOME entry).

While in regedit, add the following to the front of the REPORTS60_PATH key at HKEY_LOCAL_MACHINE|SOFTWARE|ORACLE.

REPORTS60_PATH <INSTALL_DIR>\reports\bin; …

This will ensure that the reports server can find the RMS reports. Go to Control Panel - Services and restart the Reports Server service for these changes to take effect.

Edit the netscape_11718.html file

The file is located in <INSTALL_DIR>\web_html\jinitiator.
Fill in the correct values for <SERVER> and <PORT> and save the file. This file will allow JInitiator to by dynamically installed on clients when accessed for the first time.

Copy a configuration file to the forms admin directory

Copy the file fmrweb.res, found in <INSTALL_DIR>\web_html\samplefiles, to <DEV_ORACLE_HOME>\forms60.

Modify the following file

<ORACLE_HOME>\forms60\java\oracle\forms\registry\Registry.dat

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

default.icons.iconpath=/web_gif/

Make Registry Settings in the NT Registry

Enter 'regedit' at a run command line (go to Start, Run to get a command line).

Under HK_LOCAL_MACHINE, select Software, then Oracle. Move the cursor to the right pane and click the right mouse button and select New String Value.

Name the new String NLS_DATE_FORMAT and double click on it to enter DD-MON-RR as its value.

Make the following additional registry settings using the same procedure:

FORMS60_OUTPUT	<INSTALL_DIR>\web_html\gif\
FORMS60_MAPPING	/web_gif/
COMMODE	SYNCHRONOUS
FORMS60_REPFORMAT	HTML

Create the baseHTML page for the initial Web site for RMS9.0

When configuring the forms cartridge, we included a baseHTML parameter of <INSTALL_DIR>\web_html\rms90dev.html. Now, create the page and put it in its proper location. Get the shell from <INSTALL_DIR>\web_html\samplefiles\formsbase.html and copy it to <INSTALL_DIR>\web_html\rmsdev90.html.

Edit the block of code in the first section of the file to reflect your environment.

Do the following

Your environment's form server is listening on a port indicated in rmsweb_start.

Set the port number variable in the file to the same number.(default was 9010)

```
//Begin Customer Settings
//  The following variables need to be set to match your RMS Environment
//  port_no should be set to the port that the Developer Server is running on
//  This corresponds to the serverPort setting in the old HTML Page.
var port_no = '9010';
```

The next setting is the name of the initial form that is called when RMS is started. Leave this set to rtkstrt.

```
//  startup_form represents the initial RMS form being viewed - this is
//  rtkstrt in RMS and does not need to be changed.
var startup_form = 'rtkstrt';
```

Finally, if desired, set the splash screen for the application. If no splash screen is desired, leave as is.

// splash_screen represents the name of the file containing your company's

// splash screen. This is a .gif file located in this directory.

var splash_screen = 'rtksplash.gif';

//End Customer Settings

Save rmsdev90.html.

Install the Oracle JInitiator Component on the Server

JInitiator 1.1.7.1.8 is included on the RMS9.0 CD-ROM in the directory <INSTALL_DIR>\web_html\jinitiator. Check to make sure the file is there. If the file jinit11718.exe is not there, download JInitiator from the Oracle Web site ftp://oracle.ftp.oracle.com/ into <INSTALL_DIR>\web_html\jinitiator.

Browser requirements

You will need IE 4.0 with SP1 (or higher version) or Netscape 4.6 (or higher version) as your Web browser to use RMS 9.0.0.0.

If you are using IE 4.0 with SP1 (check your version by opening IE, select HELP, choose ABOUT INTERNET EXPLORER), you need to do this to check your configuration to ensure JInitiator will work:

Select VIEW, INTERNET OPTIONS, ADVANCED. Then make sure the following options are checked:

Browsing

- Notify when downloads complete
- Disable script debugging
- Launch Channels in full screen window
- Use AutoComplete
- Use smooth scrolling
- Enable page transitions
- Browse in a new process
- Enable page hit counting
- Enable scheduled subscription updates
- Show Internet Explorer on the Desktop (requires restart)
- Update links – Always

Multimedia

- Show pictures
- Play animations
- Play videos
- Play sounds
- Smart image dithering

Security

- Enable Profile Assistant
- SSL 2.0
- SSL 3.0
- Warn if forms submit is being redirected
- Warn if changing between secure and not secure mode
- Warn about invalid site certificates
- Cookies - Always accept cookies

Java VM

- Java console enabled (requires restart)
- Java JIT compiler enabled

Searching

- Autoscan common root domain
- Search when URL files - Always ask

HTTP 1.1. Settings

- Use HTTP 1.1

Click Apply. Click OK.

IE5.0 default options work with JInitiator.

Test the System

Reboot the server computer. Ensure that the two *.bat files in startup run. Leave the command windows for ows_start.bat, rmsdev90form.bat, and rmsdev90report.bat open until they shut down automatically. Add a printer to your machine if you do not have one; this is required for the reports to work properly. To test reports after you start the forms and reports servers (using scripts rmsdev90form.bat and rmsdev90report.bat), specify your Web listener port number only if it was set to a default different than 80.

Test the reports cartridge: http://<your_machine>:port number/rwows?

e.g. http://mspvol01/rwows?

Test the reports server: http://<your_machine>:port number/rwows/showjobs?server=report_server_name

e.g. http://mspvol01/rwows/showjobs?server=repserver.world

Connect the client to the forms server by calling:

http://<your_machine>/RMSDEV90

Note: "RMSDEV90" is case sensitive, and should match the cartridge name exactly.

The first time that you connect to the server the *JInitiator* will download and install. Select the checkbox indicating you trust files from Oracle. The *JInitiator* download will occur the first time that each machine accesses the Web server. After the *JInitiator* is completely installed, close the browser and open a new browser session to use *JInitiator*.

JInitiator can be downloaded from ftp://oracle.ftp.oracle.com/dev_tools/patchsets/dev2k/Win95NT/jinitiator/ if necessary.

Upgrading from a Previous Release of the RMS (optional)

1. Create the File Structure.

   Note: Check to make sure there is enough disk space available when determining where RMS 9.0.0.0 will be installed.The RMS 9.0.0.0 Database Server file structure needs about 500 MB of space for install scripts, install modules and batch modules.

   If you haven't already, log into UNIX as *retek* and mount the CD on the database server. There are four directories on the RMS 9.0.0.0.0 CD: appserverunix, appservernt, dbserverunix and d2000. The dbserverunix directory contains the files for the database server install. Change directories to dbserverunix. Decide where you will be installing RMS database files.

To unpack the Database Server Files and create the File System run **builddb_90.run** from the CD while in the <CD mount point>/dbserverunix directory. The installation script must be run on the database server.

- The **builddb_90.run** script will prompt for a path in which to install the RMS 9.0.0.0 database server files on the system. If the entire path does not currently exist it will create it for you. This will be referred to as <INSTALL_DIR> in the installation instructions.
- The script will then copy the tarred and compressed file containing the database files to the specified directory on your server.
- Next, it will uncompress the file and untar 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 script will then clean up any extra files produced.

Note: The **builddb_90.run** script will call the install.rms script after it has copied the Database Server files and created the file structure. If you are upgrading from a previous release, do not continue running the install.rms script. The builddb_90.run script will prompt you before calling the install.rms script, at this prompt you should answer N(o) and continue on to step 2.

2. Upgrade DDL

To upgrade from a previous release of the RMS, there is a set of scripts provided in the <INSTALL_DIR>/upgrade directory that, when run in the correct order, will change the current DDL from release 8.0.3 to 9.0.0.0. A master script has been created to call each of the upgrade scripts in the proper order. The script is named rundbc.sql and is located in the <INSTALL_DIR>/upgrade directory. The upgrade scripts assume that you have Retek 8.0.3 installed. Please note that these scripts are meant as a starting point for upgrading. Each script will need to be looked at and modified appropriately based on tablespace names, storage clauses, etc. Also, if customization has been done to tables in base RMS, then all of this must be taken into account when running the scripts. Please see the release notes included with this release of RMS 9.0.0.0 for a list of all the database upgrade scripts and the associated database objects that they act upon.

These scripts are probably not the best option for converting the larger tables. For large tables, a conversion routine should be discussed and put into place. The script should, however, provide a starting point for what changes need to be made to each table.

Some scripts will not actually save data for you. For these table changes, a conversion program will need to be created if you wish to convert data from one release to the next.

Some of scripts will save data, but an exact data exchange may not be possible due to the nature of the structure change performed on the table. These scripts will need to be verified so that appropriate data can be inserted for your environment rather than the defaults that are provided.

After this task has been completed, the DDL sent in the upgrade directory should be created and compared with the DDL that you are currently running to verify that they are in sync.

To run the upgrade scripts, change directories to <INSTALL_DIR>/upgrade. Log into sqlplus as your schema owner and type the following:

>rundbc.sql

3. Upgrade owning schema privileges.

The necessary owning schema privileges and permissions have changed greatly from RMS 8.0.3 to RMS 9.0.0.0. The largest change is the removal of DBA privileges. To remove all the existing system and object privileges for your schema owner, change directories to <INSTALL_DIR>/upgrade. Log into sqlplus as sys, and run the following script:

>ownerprivchange80to90.sql

This script will prompt you for the owning schema's name, then remove all the granted permissions and apply all the necessary ones for RMS 9.0.0.0.

4. Upgrade stored objects

To upgrade stored objects from a previous release of the RMS, all existing RMS Base stored procedures, packages, and functions should be dropped and the new set should be created. The objects that are no longer used in 9.0 are removed in various database change scripts located in the <INSTALL_DIR>/upgrade directory. Please note that all customizations made to base RMS packages and procedures will need to be made again on the new version of code. To start the objects, see Appendix A. You will need to run the scripts that create the packages, stored procedures, and functions for RMS 9.0.0.0 as well as the script that creates the packages for the toolset 4.0. To update the database server packages, stored procedures, and functions for use with the Oracle Site Fuels Management API or the Oracle Financials 10.7 Interface, see Appendix I and Appendix K respectively.

5. Upgrade source code

To upgrade source code from a previous release of the RMS, all existing RMS Base source code should be dropped and the new set of source code from the tape should be copied to the appropriate directory. This includes all forms, menus, libraries as well as all the batch programs and libraries. Please note that all customizations made to base RMS source code will need to be made again on the new version of code.

Data Dictionary Installation Instructions (optional)

If you have Designer/2000 and would like to have a copy of the data dictionary online, perform the following steps. Note that this section is optional.

Note: The version of Designer/2000 required for this export is **2.1.2**.

1. FTP the export file from the server.

The rms90.dmp file resides in the /d2000 directory on the CD.

2. Import the file in Designer/2000.

Login to the Designer/2000 application. Open up the Repository Object Navigator and choose the menu option Application--Restore.

In the import box at the bottom of the window, choose to import the file that was brought over to the client in step 1. After it is imported, select that application to be restored into the new application name.

Note: Please be aware that this version of designer has not been updated to work completely in synch with Oracle 8i. There are several functional components of Oracle 8i that Retek uses that Designer cannot handle as yet. Those exceptions deal primarily with the new types of partitioning of tables and indexes.

Appendix A

Manual Instructions for Database Server Installation Process

Complete either the steps in this appendix or steps 5 and 6 from the primary installation process. Before beginning these steps, verify that steps 1 through 4 have been completed in the primary installation instructions for the database server.

(Beginning after mounting the CDROM in step 5 from the primary install process)

1. Create a directory for the system. It will be referred to as <INSTALL_DIR> for the remainder of this document.
2. Copy the dbserver.Z to the <INSTALL_DIR>.
3. Uncompress the file:

   uncompress dbserver.Z

4. Untar the file:

   tar xvf dbserver

This creates the directory structure in which the RMS 9.0.0.0 files will reside. The tarfile will not retain the permissions settings they had when leaving Retek, so 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.

5. The directory structure will look like this:

<INSTALL_DIR>/

rms9000/ /error -Batch error files print here.
/log -Batch output prints here.
/oracle
/lib/ /bin -Batch library executables.
/src -Batch library source code.

/proc /bin -Batch program executables
/src -Batch source code.

install/ db_objects/ -Packages, procedures, functions, shared libraries.
ddl/ -DDL files.
import/ -Scripts to populate RMS tables.
installer_scripts/ -Scripts used by the install. utility.
logfiles/ -Installation log directory.
sqlplus/ -Scripts to populate required data.
toolset4.0_dbo/ -Toolset packages.
utility/ -Various useful scripts.

oapps/ orafin_107 -Various Oracle Financials Interface information/scripts.
/grants - Oracle Financials setup see Appendix I for more info.
/packages - Oracle Financials setup see Appendix I for more info.
/reports - Oracle Financials setup see Appendix I for more info.
/sequences - Oracle Financials setup see Appendix I for more info.
/triggers - Oracle Financials setup see Appendix I for more info.
/views - Oracle Financials setup see Appendix I for more info.
/tables - Oracle Financials setup see Appendix I for more info.
/synonyms - Oracle Financials setup see Appendix I for more info.
/rms - RMS 9.0.0.0 to Oracle Financials
sfm/ -Site Fuels Management Interface trigger scripts see Appendix K.

sample_profiles/ -A profile to aid environment setup

upgrade/ -Scripts to aid upgrading from a previous version of the RMS.

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

   Verify that all files are owned by retek and belong to the dba group. Make ownership and group changes if necessary. Logged into Unix as *Root*, CD to <INSTALL_DIR> and use the following UNIX commands to change the owner and group:

   - chown –R retek *
   - chgrp –R dba *

7. Create RMS 9.0.0.0 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.

   Login to SQLPLUS as *sys* and enter the following commands:

   SQL> create role developer;

   Create the ORACLE user that will correspond to the *rms9000* environment by logging into SQLPLUS as the user *sys* and enter the following commands, replacing the text in the brackets with the appropriate names:

   SQL> create user <RMS 9.0.0.0 Schema Owner> identified by <password>

   default tablespace retek_data

   temporary tablespace <temporary tablespace name>;

The ORACLE user <RMS 9.0.0.0 Schema Owner> will serve as the owner of the database objects and should be granted the following permissions. Login to SQLPLUS as the user sys and enter the following command:

Note: This command can be issued by calling the /<INSTALL_DIR>/install/utility/grants.sql script when logged onto the database as sys.

```
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 9.0.0.0 Schema Owner>
    /
    alter user <RMS 9.0.0.0 Schema Owner>  quota unlimited on retek_data
    /
    alter user <RMS 9.0.0.0 Schema Owner>  quota unlimited on index_data
    /
    grant select on sys.dba_roles to <RMS 9.0.0.0 Schema Owner>
    /
    grant select on sys.dba_role_privs to <RMS 9.0.0.0 Schema Owner>
    /
    grant select on sys.dba_jobs to <RMS 9.0.0.0 Schema Owner>
    /
```

8. Create DDL for RMS 9.0.0.0 owner.

Log into UNIX as the *retek* user. 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.

To start the scripts, login to SQLPLUS as the ORACLE user (<RMS 9.0.0.0 Schema Owner>) and enter the following command:

SQL> @rms90.sql

This runs all of the scripts needed to create tables, indexes, constraints, sequences, and views for the user. A spool file will be created named rms90.log which is in <INSTALL_DIR>/install/logfiles. This file should be checked upon completion 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 *rms90.vw1* will be started as well as the triggers in file *rms90.tr1*. 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 (*rms90.vw2)* and triggers *(rms90.tr2)*.

9. Create packages for the toolset 4.0.

On the server, change directories to <INSTALL_DIR>/install/toolset4.0_dbo. All packages within the toolset must be started in the database in the appropriate order. The script START40.PLS will start the toolset 4.0 packages automatically and will spool to the *start40.log* file which is located in <INSTALL_DIR>/install/logfiles.

Login to SQLPLUS as <RMS 9.0.0.0 Schema Owner>and enter the following command to run the script:

SQL> @start40.pls

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

10. Create packages, stored procedures, and functions.

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 9.0.0.0. All of these objects must be started in the database in the appropriate order. The script *STARTALL90.PLS* will do this automatically. The *STARTALL90.PLS* script will spool to the *checkpack.log* file which is located in the <INSTALL_DIR>/install/logfiles directory.

Login to SQLPLUS as <RMS 9.0.0.0 Schema Owner>and enter the following command to run the script:

SQL> @startall90.pls

This creates each of the packages, procedures, and functions used within RMS 9.0.0.0. This takes approximately 10-30 minutes to complete. (The amount of time this requires varies, depending on hardware and configuration.) When completed, exit out of SQLPLUS and view the spool file to verify that no errors were found.

11. 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 *rms90.sql2* script (which runs rms90.vw2 and rms90.tr2) will spool to the rms90.log2 file in the <INSTALL_DIR>/install/logfiles directory. To create the remaining views and triggers, change directories to <INSTALL_DIR>/ddl.

Login to SQLPLUS as the ORACLE user <RMS 9.0.0.0 Schema Owner>. From SQLPLUS, enter the following commands:

SQL> @rms90.sql2

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

12. Load required data.

Log into UNIX as *retek* and change directories to <INSTALL_DIR>/install/sqlplus. In this directory there are a number of files that contain data required for RMS 9.0.0.0 to run properly. Listed below is an alphabetical listing and explanation of each of the SQL scripts. The order in which they are applied is noted in the install_scripts_90.sql script.

SQL Script Name	Description
addplcy	Runs the procedures necessary for RMS 9.0.0.0 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.
context_rms900_base.sql	Creates data required for RMS 9.0.0.0 online help.
context_resa900_base.sql	Creates data required for Sales Audit online help.
context_tm900_base.sql	Creates data required for Trade Management online help.
drpplcy	This is not run initially, but is provided as a tool.
elc_comp_pre_htsupld.sql	This inserts initial estimated landed cost expense data for RMS 9.0.0.0.
install_scripts_90.sql	Starts all the primary installation scripts for RMS 9.0.0.0.
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.

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.
sa_metadata.sql	Required data for Sales Audit. This script is called by sa_system_required.sql
sa_parm.sql	Required data for Sales Audit. This script is called by sa_metadata.sql.
sa_parm_type.sql	Required data for Sales Audit. This script is called by sa_metadata.sql.
sa_realm.sql	Required data for Sales Audit. This script is called by sa_metadata.sql.
sa_realm_type.sql	Required data for Sales Audit. This script is called by sa_metadata.sql.
sa_system_required.sql	Required data for Sales Audit.
sa_wizard_prompt_text.sql	Required data for Sales Audit. This script is called by sa_system_required.sql
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 9.0.0.0 to run.
tl_columns.sql	Creates data used for mass translation.
uom_conversion.sql	Required data for unit of measure conversions
uom_x_conversion.sql	Required data for unit of measure conversions.
vehrnd.sql	
wizard.sql	Creates data used for instructions for wizards within the RMS 9.0.0.0.

Running the install_scripts_90.sql file will start all these scripts. To do this, login to SQLPLUS as the user <RMS 9.0.0.0 Schema Owner> and enter the following:

SQL> @install_scripts_90.sql

A spool file named *install_script90.log*will be created in <INSTALL_DIR>/install/logfiles. Verify that this file has no errors upon completion.

Additionally, there is a file named *mmdemo.sql* that creates a minimal amount of data in the organizational and merchandise hierarchies, etc. It can be run from SQLPLUS at this point if a small set of data is desired at this time.

To run mmdemo.sql, from the <INSTALL_DIR>/install/sqlplus, login to SQLPLUS as the user <RMS 9.0.0.0 Schema Owner> and enter the following:

SQL> @mmdemo.sql

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

There is also some data that needs to be imported into the system during installation. The tables requiring imports are: *rtk_errors*, *multiview_default_40,* and *multiview_saved_40*. This data can be automatically imported by using the *import_data* script.

To run this, change directories to:

<INSTALL_DIR>/install/import.

Then, enter the following:

$ ./import_data

Enter the RMS 9.0.0.0 Schema Owner's connect string (userid/password@instance) when prompted. You will also be prompted for the sys connect string – The script drops constraints on the *rt_errors* and the *multiview default40* table as the schema owner and then does the import of the schema owner's data as sys. The script will produce two output files in <INSTALL_DIR>/install/logfiles – one for the rtk_errors import named rtk_errors.out and one for the multiview table imports named multiview.out. These logs should be checked to ensure a successful import.

13. Return to the primary dbserver installation process at step 7 to fully complete this process.

Appendix B

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 C

Customizing the Date Format

The date format is set so that it will display as 'DD-MON-RR' online but will be entered in the format 'MMDDRR' by users. If this format is not desirable, this format can be modified.

To change the format where the date is displayed, set the NLS_DATE FORMAT in the Windows Registry or in the appropriate *.bat file on the Oracle Web Application Server.

NLS_DATE_FORMAT=DD-MON-RR

To change the format where the date is entered, change the DATE_ENTRY column in the SYSTEM_OPTIONS table to the format that is desired for entry. For example, the value of this field could be 'DDMMRR'.

Note: Any date format that contains only two digits for the year should contain 'RR' rather than 'YY' so that dates will be handled appropriately by Oracle in the year 2000.

Appendix D

Troubleshooting Database Setup

The following is a sample script that creates the database necessary for the RMS 9.0.0.0. Note that a different character set may be required for your database. Please check with Oracle regarding compatibility of the character set with Developer 6.0 as not all character sets will work with Developer 6.0. To use this information, login to UNIX as *oracle*, run svrmgrl and execute the following commands:

```
connect internal

startup nomount pfile=/home/oracle/dbs/initRETEK.ora

create database RETEK
   controlfile reuse
   maxlogfiles 100
   maxdatafiles 200
   character set WE8ISO8859P1
   datafile '/files1/oradata/RETEK/retek_system01.dbf' SIZE 200M
   logfile group 1('/files1/oradata/RETEK/redo01a.log',
                   '/files2/oradata/RETEK/redo01b.log') SIZE 10M,
           group 2('/files3/oradata/RETEK/redo02a.log',
                   '/files4/oradata/RETEK/redo02b.log') SIZE 10M,
           group 3('/files5/oradata/RETEK/redo03a.log',
                   '/files6/oradata/RETEK/redo03b.log') SIZE 10M;

set echo off
@$ORACLE_HOME/rdbms/admin/catalog.sql
@$ORACLE_HOME/rdbms/admin/catproc.sql
@$ORACLE_HOME/rdbms/admin/catpart.sql   # - optional but useful
@$ORACLE_HOME/rdbms/admin/cataudit.sql  # - optional but useful
@$ORACLE_HOME/rdbms/admin/catblock.sql  # - optional but useful
set echo on

create rollback segment RBSSYS
   tablespace system
   storage(initial 16K next 16K minextents 2 maxextents 20);

alter rollback segment RBSSYS online;

create tablespace RBS
   datafile '/files1/oradata/RETEK/rbs01.dbf' size 500M reuse
   default storage(initial 1M next 1M minextents 2 maxextents 100);

create tablespace temp
   datafile '/files2/oradata/RETEK/temp01.dbf' size 100M reuse
   default storage(initial 5M next 5M minextents 1 maxextents 20 );
```

```
create rollback segment RBS01
   tablespace RBS
   storage(optimal 2M);

alter rollback segment RBS01 online;

create rollback segment RBS02
   tablespace RBS
   storage(optimal 2M);

alter rollback segment RBS02 online;

create rollback segment RBS03
   tablespace RBS
   storage(optimal 2M);

alter rollback segment RBS03 online;

alter rollback segment RBSSYS offline;

connect system/manager

set echo off
@$ORACLE_HOME/rdbms/admin/catdbsyn.sql
@$ORACLE_HOME/sqlplus/admin/pupbld.sql
set echo on

create tablespace retek_data
   datafile '/files3/oradata/RETEK/retek_data01.dbf' size 4000M
   default storage (initial 100K next 100K maxextents 255);

create tablespace index_data
   datafile '/files4/oradata/RETEK/index_data01.dbf' size 4000M
   default storage (initial 100K next 100K maxextents 255);
```

Appendix E

Script (move_user.sql) to Create Private Synonyms

The following sample script automates the creation of private synonyms for an Oracle user. The script will prompt you to enter the user that you are creating the synonyms for and the user that owns the objects.

```
declare
   string VARCHAR2(200);
   c1     INTEGER;
   c2     INTEGER;
   ret1   INTEGER;
   ret2   INTEGER;
   cursor cur_1 is
      select object_name
        from dba_objects
       where owner = upper('&&username')
         and object_type in ('SYNONYM');
   cursor cur_2 is
      select owner, object_name
        from dba_objects
       where owner = upper('&new_owner')
         and object_type in ('TABLE', 'VIEW', 'CLUSTER', 'FUNCTION', 'PACKAGE',
                             'PROCEDURE', 'SEQUENCE');
begin
  for c1_rec in cur_1 loop
    string := 'drop synonym &&username' || '.' || c1_rec.object_name;
    c1 := dbms_sql.open_cursor;
    dbms_sql.parse(c1, string, dbms_sql.v7);
    ret1 := dbms_sql.execute(c1);
    dbms_sql.close_cursor(c1);
  end loop;
  for c2_rec in cur_2 loop
    string := 'create synonym &&username' || '.' || c2_rec.object_name ||
              ' for ' || c2_rec.owner || '.' || c2_rec.object_name;
    c2 := dbms_sql.open_cursor;
    dbms_sql.parse(c2, string, dbms_sql.v7);
    ret2 := dbms_sql.execute(c2);
    dbms_sql.close_cursor(c2);
  end loop;
end;
/

undefine username;
```

Appendix F

Makefiles and How to Use Them in the Retek Environment

This document describes how makefiles are utilized to build Retek software products. You should have a minimal understanding about how to use the make utility. This software can be downloaded (for free) from ftp://prep.ai.mit.edu/gnu/make or from http://prep.ai.mit.edu.

Since Retek software is ported to many different UNIX platforms and the platforms do not all have the same version of **make**; the constructs used should be kept to the basics. A good book to learn about and to use as a reference guide to **make** is "Managing Projects with Make" by Andrew Oram and Steve Talbot, from O'Reilly & Associates.

In this document, items in **bold** are the names of specific things (commands, files, variables, etc). Items in *italics* are generalized names for things.

The **retek.mk** file contains good examples for dealing with libraries. The **resa.mk** file contains good examples for dealing with executables.

The Files Involved

Each product will have its own makefile named ***product*.mk** (ex: **retek.mk**, **rms.mk**, **resa.mk**, etc) and an "include" makefile per product, named ***product*_include.mk** (ex: **rms_include.mk**, **resa_include.mk**, etc). There are 6 included files for each ***product*.mk**. However, to build any of the MTS products, it is possible to use the **mts.mk** makefile which will then call the appropriate ***product*.mk** to build the product.

Oracle.mk	This file is initially generated with the oramake script. oramake examines the example makefiles that come with the Oracle database software and extracts the compile and link definitions. These definitions assume that the native C compiler is being used. If it is not, than the options and arguments in oracle.mk needs to be modified for the compiler to be used. oramake is a Korn shell (ksh) script and should run on any UNIX machine.
platform.mk	This file contains the definitions for platform (hardware and operating system) specific compile and link options and arguments. It also aggregates them together for ease of use in the other parts of the makefiles. This should be the only file (with the possible exception, as noted above, of oracle.mk) that needs to be modified to port Retek software to another platform.
rules.mk	This file contains the translation rules for turning source code into object files.

***product*.d**	This file contains the dependencies of the individual object modules. Initially, an empty file is created (so that make does not complain about a missing include file). All source code that pertains to this product should be copied to the same directory. Then a make –f product.mk depend is done. This causes the product.d file to be regenerated with all of the include dependencies of the source code that is present. This should be done anytime the source code is changed or source files are added. Note that a make –f mts.mk depend causes all product.d files to be generated.

Note: Some early versions of the Korn shell have a bug that prevents them from correctly interpreting in embedded **awk** script (like there is in oramake). Either get the latest patches for the Korn shell from the vendor or add a backslash (\) before all of the open ({) and close (}) braces of the embedded **awk** script.

Warning: If all of the source code for the product is not present, the newly regenerated ***product*.d** file will be missing the dependencies of the missing modules. This hampers the generation of the libraries and executables and cause much confusion during development and porting.

Environment Variables That Must be Defined

These variables are set by the Retek **chgenv.ksh** script. They are used inside the makefiles to define other variables (macros), to figure out where to install things, and to connect to the Oracle database during compilation.

MMHOME	Defines where the Retek products are installed
MMUSER	Defines the Retek user's Oracle login ID
ORACLE_HOME	Defines where the Oracle software is installed.
ORACLE_SID	Defines which Oracle database to login to.
PASSWORD	Defines the users Oracle login ID password.

The Contents of platform.mk

Variable (Macro) Definitions

PROCDEBUG
This variable is used by Oracle's C Precompiler (Pro*C) to add the appropriate #line directives to the resulting C source file. This causes the compiler (and hopefully the debugger) to report problems based on the original .pc file line number. Some debuggers have a problem with #line directives. This variable definition makes it easy to turn off the #line directives. For example:

```
make –f rms.mk posdnld PROCDEBUG=
```

GFLAG
This variable is originally defined in **oracle.mk**. During development and porting this variable should be defined to be –g (or whatever C compiler options and arguments enable symbolic debugging). This makes using a debugger easier. During production, use the GFLAG= within the build command so that the value in **oracle.mk** is used.

OPTIMIZE
This variable is originally defined in **oracle.mk**. During development and porting this variable should be defined to be empty. This makes using a debugger easier. During production, the software should be rebuilt with **OPTIMIZE** commented out so that the value in **oracle.mk** is used.

PLATFORM_CFLAGS
Some platforms may require additional options or arguments to the C compiler (beyond what **oracle.mk** provides) in order to compile the Retek software. For instance, some of the Retek software requires POSIX library calls, which some compilers do not enable by default. So POSIX must be turned on. This is usually accomplished with a –D option of some sort.

RETEK_LIB_BIN
This variable defines where libraries should be installed. It makes use of the **MMHOME** environment variable.

RETEK_LIB_SRC
This variable defines where library source files should be installed. It makes use of the **MMHOME** environment variable.

RETEK_PROC_BIN
This variable defines where executable programs should be installed. It makes use of the **MMHOME** environment variable.

RETEK_PROC_SRC
This variable defines where source code should be installed. It makes use of the **MMHOME** environment variable.

RETEK_LDFLAGS
Like **PLATFORM_CFLAGS**, but used during linking. This should not be used outside of **platform.mk**, use **LDFLAGS** instead. This uses the Retek library, **libretek.a**.

LDFLAGS
This is the variable that should be used to link a **program** with the Retek library, **libretek.a**. It combines the **PRODUCT_LDFLAGS** (defined in ***product*.mk**), **RETEK_LDFLAGS** and **ORACLE_LDFLAGS** (defined in **oracle.mk**) for ease of use.

PRODUCT_CFLAGS
This is set in the ***product*.mk** makefile. Some products may require additional options or arguments to the C compiler (beyond what **oracle.mk** provides) in order to compile the Retek product software. This should be limited to –D or –I options. Anything else probably belongs in **PLATFORM_CFLAGS**.

PRODUCT_LDFLAGS
This is set in the ***product*.mk** makefile. Like **PRODUCT_CFLAGS**, but used during linking. This should be limited to –L or –l options. Anything else probably belongs in **RETEK_LDFLAGS**.

*The Contents of product*_include.mk

Variable (Macro) Definitions

***PRODUCT*_EXECUTABLES**
This contains the name of each ***program*** that this makefile produces. In some cases, additional macros may be defined containing the specific executables in a system. For instance ACH_EXECUTABLES in **resa_include.mk** contains all the executables required if the ACH external system is used.

***PRODUCT*_LIBS**
This contains the name of each library (**lib*library*.a**) that this makefile produces.

***LIBRARY*_INCLUDES**
This contains the name of each include file that this ***library*** has.

***PRODUCT*_INCLUDES**
This contains the name of each include file that this makefile produces. This is usually made up of the ***LIBRARY*_INCLUDES** variables.

***PROGRAM*_C_OBJS**
Each ***program*** that has C modules will have one of these. It should contain all of the .o files that are derived from the .c files that will be linked into the ***program***. This variable is used so that we do not need to remember to put the filenames in all of the places that it would be required (at least 5).

***PROGRAM*_PC_OBJS**
Each ***program*** that has Pro*C modules will have one of these. It should contain all of the .o files derived from the .pc files that will be linked into the ***program***. This variable is used so that you won't need to remember to put the filenames in all of the required places (at least 5).

***LIBRARY*_C_OBJS**
Each library that has C modules will have one of these. It should contain all of the .o files derived from the .c files that will be archived into the library (**lib*library*.a**). This variable is used so that you won't need to remember to put the filenames in all of the required places (at least 5).

***LIBRARY*_PC_OBJS**
Each library that has Pro*C modules will have one of these. It should contain all of the .o files derived from the .pc files that will be archived into the library (**lib*library*.a**). This variable is used so that you won't need to remember to put the filenames in all of the required places (at least 5).

***PRODUCT*_C_SRCS**
This contains all of the C source files. It is made up of all of the ***PROGRAM*_C_OBJS** variables but the .o's are changed to .c's with the **$(*VAR*:.o=.c)** makefile construct.

***PRODUCT*_PC_SRCS**
This contains all of the Pro*C source files. It is made up of all of the ***PROGRAM*_PC_OBJS** variables but the .o's are changed to .pc's with the **$(*VAR*:.o=.pc)** makefile construct.

***PRODUCT*_TARGETS**
This contains the list of valid targets for this product makefile. This is usually made up of *PRODUCT*_EXECUTABLES, *PRODUCT*_C_SRCS variables changed to .o, *PRODUCT*_PC_SRCS changed to .o, *PRODUCT*_PC_SRCS changed to .c, *PRODUCT*_LIBS, *PRODUCT*_INCLUDES, and any additional product specific targets.

The Contents of product.mk

Variable (Macro) Definitions

PRODUCT_CFLAGS
For details, see the discussion above in the contents of platform.mk.

PRODUCT_LDFLAGS
For details, see the discussion above in the contents of platform.mk.

Standard Targets

all
This target first ensures that all of the **libs** are built. Then it builds all of the **EXECUTABLES**. This must be the first target in the makefile. The first target becomes the default target. If you do not tell **make** what to build on its command line, this is what will be built.

libs
This target first ensures that all of the **includes** are built. Then it builds all of the **LIBS**.

includes
This target builds all of the **INCLUDES**.

install
This target looks for existing **LIBS** and **EXECUTABLES** and copies them to the appropriate directory. This is like the last step of a **fcomp**.

clean
This target removes all of the .o files and all of the .c files that come from .pc files. In general, it is meant to remove all intermediate files.

clobber
This target first removes all intermediate files by ensuring that the **clean** target is invoked. Then, it should remove all **EXECUTABLES**, **LIBS** and **INCLUDES**. In general, it is meant to remove all derived objects.

lint
This target (if implemented) will perform a lint check on the source files. **Lint** catches many questionable C constructs that C compilers tend to ignore. For details about **lint**, see the UNIX **man** page. If the code is lint free, it is usually much easier to port to other systems.

depend
This target updates the ***product*.d** file with all of the dependencies of **C_SRCS** and **PC_SRCS**. For details, see the discussion above.

***module*.o**
This target builds just the object file (.o) associated with ***module*.c** or ***module*.pc**. This is useful for checking syntax for a multi-module executable.

***module*.ln**
This target does a lint check on the module associated with ***module*.c** or ***module*.pc**. This is useful for checking portability for a module.

lib*library*.a
This target builds everything needed to create a library named **lib*library*.a**.

llib-l*library*.ln
This target (if implemented) does a lint check across all of the modules that make up **lib*library*.a**. This is useful for checking portability for an executable. **llib-l*library*.ln** files are also useful when building a ***program*.lint**. Portability issues are remembered in this file and utilized while checking ***program*.lint**. For details about **lint**, see the UNIX **man** page. If the code is lint free, it is usually much easier to port to other systems.

program
This target builds everything needed to create an executable named ***program***. This is like the old **hcomp** or **sacomp** scripts except that if ***program*** consists of multiple modules, all of them will be built and linked together to form the executable.

***program*.lint**
This target (if implemented) does a lint check across all of the modules that make up ***program***. This is useful for checking portability for an executable. For details about **lint**, see the UNIX **man** page. If the code is lint free, it is usually much easier to port to other systems.

How to Add a Program or Library to a Makefile

A few notes about Retek's makefiles and makefiles in general

A long line may be continued by placing a backslash (\) at the end of it. Make sure that the backslash is right before the newline or it loses its meaning. A set of lines continued by backslashes is parsed as one line.

Leading tabs on command lines (the script that says how to build a target) are significant. If they get converted to spaces or a removed, the makefile will cease to function. Placing a tab at the beginning of any other statement (like dependency definitions) can also cause problems. Imbedded tabs are OK. This includes tabs at the beginning of lines that have been continued from the previous line.

Blank lines are ignored, provided that they do not begin with a tab.

The # character marks the start of a comment that continues to the end of the line. Comments are also ignored.

Please keep the various sections within the makefiles alphabetized. While not required, this makes it much easier to determine what additions and modifications are needed.

Use the existing makefiles as examples when making changes to them. Chances are, that what you need to add has already been done for something else.

Adding a Module to an Existing Program

Add the new module's *module*.o name to the ***PROGRAM*_C_OBJS** or ***PROGRAM*_PC_OBJS** variable definition depending on if it started life as *module*.c or *module*.pc.

If ***PROGRAM*_C_OBJS** (or ***PROGRAM*_PC_OBJS**) does not exist, than create it and add it to the ***PRODUCT*_C_SRCS** (or ***PRODUCT*_PC_SRCS**) variable definition.

Adding a new Program

Find the appropriate ***product*.mk**. Add the new *module*.o names to the ***PROGRAM*_C_OBJS** and/or ***PROGRAM*_PC_OBJS** variable definitions depending on if it started life as *module*.c or *module*.pc. Then add the new ***PROGRAM*_C_OBJS** and/or ***PROGRAM*_PC_OBJS** to the ***PRODUCT*_C_SRCS** and/or ***PRODUCT*_PC_SRCS** variable definition.

Add the new *program* to the ***PRODUCT*_EXECUTABLES** variable definition. If there are optional interfaces to other products, than there will be multiple versions of the **EXECUTABLES** variable (in the form ***INTERFACE*_EXECUTABLES**) to make building easier. These multiple versions will than be aggregated by the ***PRODUCT*_EXECUTABLES** variable. Add the new *program* to the appropriate one. If a new variable needs to be created, than also add it to the ***PRODUCT*_EXECUTABLES** aggregate and create a new target for it that is similar to the **all** target.

2 new dependency or rule definitions also need to be added. The first says how to build *program*. The second says how to run **lint** on *program*. Add the following lines to the appropriate sections (the lines are continued here for readability):

```
program: $(PROGRAM_C_OBJS) $(PROGRAM_PC_OBJS)
      $(LD) -o `basename $@` `echo $(PROGRAM_C_OBJS) \
                $(PROGRAM_PC_OBJS) | xargs -n1 basename` $(LDFLAGS)
      chmod $(MASK) `basename $@`

program.lint: $(PROGRAM_C_OBJS:.o=.c) $(PROGRAM_PC_OBJS:.o=.c)
      $(LINT) $(LINT_FLAGS) `echo $(PROGRAM_C_OBJS:.o=.c) \
             $(PROGRAM_PC_OBJS:.o=.c)| xargs -n1 basename` \
             $(LDFLAGS) >`basename $@` 2>&1
```

Adding a Module to an Existing Library

Add the new module's *module*.o name to the ***LIBRARY*_C_OBJS** or ***LIBRARY*_PC_OBJS** variable definition depending on if it started life as *module*.c or *module*.pc.

If ***LIBRARY*_C_OBJS** (or ***LIBRARY*_PC_OBJS**) does not exist, than create it and add it to the **C_SRCS** (or **PC_SRCS**) variable definition.

If a new header file is being added, than add the .h file to ***LIBRARY*_INCLUDES** variable definition.

Adding a new Library

Add the new *module*.o names to the ***LIBRARY*_C_OBJS** and/or ***LIBRARY*_PC_OBJS** variable definitions depending on if it started life as *module*.c or *module*.pc. Than add the new ***LIBRARY*_C_OBJS** and/or ***LIBRARY*_PC_OBJS** to the **C_SRCS** and/or **PC_SRCS** variable definition.

Add the new *library* to the **LIBS** variable definition. Also create a separate target for building just this *library*.

If new header files are being added, than add the .h files to the new ***LIBRARY*_INCLUDES** variable definition. Than add the new ***LIBRARY*_INCLUDES** to the **INCLUDES** variable definition.

2 new dependency or rule definitions also need to be added. The first says how to build *library*. The second says how to run **lint** on *library* and for making a lint library for checking programs that use it. Add the following lines to the appropriate sections (the lines are continued here for readability):

```
lib<library>.a: $(LIB<LIBRARY>_C_OBJS) $(LIB<LIBRARY>_PC_OBJS)
      ar -r `basename $@` `echo $(LIB<LIBRARY>_C_OBJS) \
                  $(LIB<LIBRARY>_PC_OBJS) | xargs -n1 basename`
      chmod $(MASK) `basename $@`

llib-l<library>.ln: $(LIB<LIBRARY>_C_OBJS:.o=.c) \
                  $(LIB<LIBRARY>_PC_OBJS:.o=.c)
      $(LINT) $(LINT_FLAGS) -o <library> \
                `echo $(LIB<LIBRARY>_C_OBJS:.o=.c) \
                  $(LIB<LIBRARY>_PC_OBJS:.o=.c) | xargs -n1 basename`
      chmod $(MASK) `basename $@`
```

Troubleshooting Information

If a depend file will not create, your PATH may need to be edited to include the directory where the makedepend file lives. This error was encountered when compiling on the HP. The path was edited to include /opt/imake/bin. This location may vary by server and OS type.

To clean up old files (.o, .a) you can use:

make –f retek.mk clean

Note: you can substitute resa.mk or rms.mk for retek.mk

In the event of errors during compilation, use the –k option to help determine if the problem is universal or only with select programs. Example: make –f rms.mk –k 2>&1 | tee rmsbatch.lst. If the –k is not used, the makefile will stop immediately when an error occurs. If the –k is used, the makefile will continue through the list even after encountering the error.

There is a shared library, libsupcstrr.so (the suffix will vary depending on the platform), that may need special configuration to compile correctly. This may require information from the hardware manufacturer via Web site, manuals or their support services. See the platform_hpux.mk and the platform_sunsol.mk files in the *INSTALL_DIR*/rms9000/oracle/lib/src directory that were supplied, to determine the differences necessary for differences in SUN and HP.

If you started with a platform other than a SUN Solaris or HP and have errors, begin with comparing the two platform.mk's supplied to determine the changes necessary to get the files to compile on both of the servers.

Appendix G

Modules and their associated batch programs

RMS Batch Programs		
allocupd.pc	itemdnld.pc	rplext.pc
asndnld.pc	ladprg.pc	rplprg.pc
auditprg.pc	layprg.pc	rplrsprg.pc
auditsys.pc	lclrbld.pc	rtvprg.pc
callocupd.pc	locdnld.pc	rtvupld.pc
ccprg.pc	orddscnt.pc	saaldnld.pc
cmpprg.pc	ordprg.pc	salapnd.pc
cntrmain.pc	ordrev.pc	saldly.pc
cntrordb.pc	ordupd.pc	saleoh.pc
cntrprss.pc	otbdlord.pc	salins.pc
ctniupld.pc	otbdlsal.pc	salmth.pc
dealcalc.pc	otbdnld.pc	salprg.pc
dealcls.pc	otbprg.pc	salstage.pc
dealprg.pc	otbupfwd.pc	salweek.pc
ditinsrt.pc	otbupld.pc	sccext.pc
dlyprg.pc	pccdnld.pc	schedprg.pc
dtesys.pc	pccext.pc	slocrbld.pc
edidladd.pc	pccprg.pc	soutdnld.pc
edidlcon.pc	pccrdnld.pc	sprdrbld.ctl
edidldeb.pc	pccrext.pc	sprdrbld.pc
edidlord.pc	pcdnld.pc	stkdly.pc
edidlprd.pc	pcext.pc	stkprg.pc
ediprg.pc	pcimpc.pc	stkupd.pc
ediupack.pc	pcovrl.pc	stkupld.pc
ediupadd.pc	pcovrlpq.pc	stkvar.pc
ediupasn.pc	pcprg.pc	stkxplst.pc
ediupavl.pc	pctrandn.pc	stkxplwh.pc
ediupcat.pc	pctranex.pc	stlgdnld.pc

RMS Batch Programs		
ediupinv.pc	plncupld.pc	storeadd.pc
fcslupld.pc	plndupld.pc	supcnstr.pc
fcstprg.pc	plnsupld.pc	supdnld.pc
fcstrbld.pc	poscdnld.pc	supmth.pc
fdaydnld.pc	posdnld.pc	szonrbld.pc
fisdnldf.pc	posgpdld.pc	szrtbld.pc
fisdnlds.pc	posupld.pc	tcktdnld.pc
fmednldf.pc	powodld.pc	tifposdn.pc
fmednlds.pc	prepost.pc	tsfalprg.pc
forgdnld.pc	primpdep.pc	tsfcomb.pc
fsadnldf.pc	primpsku.pc	tsfdcdld.pc
fsadnlds.pc	prmext.pc	tsfdnld.pc
ftmednld.pc	prmprg.pc	tsfiupld.pc
fwhdnld.pc	prmxpld.pc	tsfoupld.pc
gcupld.pc	promdnld.pc	tsfparse.pc
hstbld.pc	rcvext.pc	tsfprg.pc
hstprg.pc	rcvupld.pc	tsfresv.pc
hstwkfst.pc	reclsdly.pc	txrposdn.pc
hstwkfwh.pc	reqext.pc	txrtupld.pc
hstwksst.pc	rpladjf.pc	vatdlxpl.pc
hstwkswh.pc	rpladjs.pc	vrplbld.pc
invaprg.pc	rplatupd.pc	wasteadj.pc
invaupld.pc	rplbld.pc	whstrasg.pc

Invoice Matching Batch Programs		
edidldeb.pc	invclshp.pc	invprg.pc
ediupinv.pc	invmatch.pc	invcpost.pc

RMS - Logistics Batch Programs		
bolcontainer_upload.ctl	lifbolup.pc	lifstkup.pc
bolhead_upload.ctl	lifinvup.pc	Receipts_upload.ctl
bolitem_upload.ctl	lifrcvup.pc	rms2rls.awk
ftp_files.awk	lifrtvup.pc	rtv_upload.ctl

Trade Management Batch Programs		
cednld.pc	lcmdnld.pc	tranupld.pc
htsupld.pc	lcupld.pc	
lcadnld.pc	lcup798.pc	

Competitive Shopping		
cmpprg.pc		

Library Batch Programs		
4_5_4.pc	saemptyfld.c	saoranuminit.c
curconv.h	saerrcodes.sql	saoranummul.c
curconv.pc	saerror.c	saoranumsub.c
intrface.h	safldcpy.c	sapartner.c
intrface.pc	saftchstdy.pc	saprom.c
oracle.mk	saftchstdyerrcnt.pc	saputrec.c
oracle.pc	saftchstdyexp.pc	sasku.c
oramake	saftchstdyexplike.pc	sastdlen.h
platform_hpux.mk	saftchsysdate.pc	sastoreday.c
platform_other.mk	saftchsysopt.pc	sastorepos.c
platform_sunsol.mk	saftchvdate.pc	sastr2fxl.c
plsqlscl.pc	safxl2str.c	sastrsig.c
restart.h	sagetbaltot.pc	sasupplier.c
restart.pc	sagetconv.pc	satendertype.c
retek_2.h	sagetlock.pc	satent.c

Library Batch Programs		
retek_2.pc	sagetrec.c	satotavail.pc
retek.d	sagetref.h	satrat.c
retek.h	sagettot.pc	saupc.c
retek.mk	sagettotlike.pc	savaldate.c
rpl.h	saincchar.pc	savalnum.c
rules.mk	saisleap.c	savaltime.c
saatoin.c	salib.h	savupc.c
saatoln.c	samerchcode.c	sawaste.c
sacode.c	samrkstdyexp.pc	std_err.h
sacodes.sql	samrktotexp.pc	std_len.h
sacrstdy.pc	samrktrexp.pc	std_rest.h
sacvtcur.pc	saoranum.h	supcstrr.h
sadatetime.c	saoranumadd.c	supcstrr.pc
saemployee.c	saoranumdiv.c	

Oracle Financials		
fifcuru2107.p c	fifgldn3107.pc	fiftrmu2107.pc
fifgldn1107.pc	fifinvcu107.pc	Fifvndu2107.pc
fifgldn2107.pc	fifpdlp107.pc	

Sales Audit Batch Programs		
sacust.ctl	saimptlog_datastat.pc	saimptlogfin.pc
sacustatt.ctl	saimptlog_final.c	samastersfm.pc
saerror.ctl	saimptlog_init.c	samisstr.ctl
saescheat_nextesn.pc	saimptlog_manval.c	sapreexp.pc
saescheat.pc	saimptlog_nextbgsn.pc	sapurge.pc
saexpach.pc	saimptlog_nextesn.pc	sarules.pc
saexpgl107.pc	saimptlog_nextmtsn.pc	sastdycr.pc
saexpim.pc	saimptlog_nexttsn.pc	satdisc.ctl
saexprdw.pc	saimptlog_nextvsn.pc	sathead.ctl
saexprms.pc	saimptlog_output.c	satitem.ctl
saexpsfm.pc	saimptlog_proto.h	satotals.pc

Sales Audit Batch Programs		
saexpuar.pc	saimptlog_rtlog.c	sattax.ctl
sagetref.pc	saimptlog_tdup.c	sattend.ctl
saimpadj.pc	saimptlog_tdup.h	savouch.pc
saimptlog_ccval.c	saimptlog.c	
saimptlog_ccval.h	saimptlog.h	

Appendix H

Product security, Location security, and Price zone security

Security overview

The Security feature provides a method for protecting product, location, and price zone records. You can give users either full or limited access. Users with limited access can be linked to one or more groups. The users within the groups share the same permissions to view, add, edit, and delete records in specified functional areas, merchandise levels, organizational levels, and price zones. The users must be valid system users.

To secure records:

- Add the users.
- Add the security groups.
- Add users to the security groups
- Add product level security by group.
- Add location level security by group.
- Add price zone security by group.

The security restrictions that you set up at the group level are applied to all users that are linked to the group. Any changes that you make to the security settings become effective after a scheduled or ad hoc batch program rebuilds the security records.

Help Topics

The following topics concerning product security, location security, and price zone security can be found in the application's Help files:

- Security overview
- Add a user
- Edit a user
- View a user
- Delete a user
- Add a security group
- Edit a security
- Delete price zone security by group

Appendix I

Financial Interface Installation Modification

INTEGRATE ORACLE FINANCIALS 10.7

Oracle-INDUS Integration Readme File (05/13/1997) Modified to Word document May, 2000.

These are files that enable RMS 9.0.0.0 to integrate with Oracle Financials 10.7. They MUST be applied prior to enabling certain RMS 9.0.0.0 Financial functionality. If not, errors will occur.

* Make sure the concurrent manager is up and running. Run the "startmgr" command at the UNIX prompt by logging in as APPLMGR account. This command prompts for the password for AOL. Please refer to the Oracle Application Installation Manual.

* Make sure a user is setup with System Administrator responsibility access.

Fresh Install Steps

Retek Notes:

Once the tarfile for the dbserver has been unloaded successfully from the CD using the basic RMS 9.0.0.0 instructions, the following instructions will apply to the files and folders in the directory <INSTALL DIR>/oapps/orafin_107. The instructions assume that these files have not been moved and has the SQL*PLUS application in the UNIX path.

Whenever <user> is specified, that is the owner of the schema that owns those specific Oracle Application tables. <po user>, for example, owns all of the Purchase Ordering tables.

If FTPing, all files should be FTPed in ASCII mode except for two in the reports folder: poex.rep and poex.rdf. They should be FTPed in binary mode.

1. Change into the <INSTALL DIR>/oapps/orafin_107/tables directory and run the following commands :

   sqlplus <po user>/<passwd> @crpotabs.sql

   check the "crpotabs.lst" file for any errors.

   sqlplus <ap user>/<passwd> @craptabs.sql

   check the "craptabs.lst" file for any errors.

2. Change into the <INSTALL DIR>/oapps/orafin_107/sequence directory and run the following command :

   sqlplus <po user>/<passwd> @crseqs.sql

   check the "crseqs.lst" file for any errors.

3. Change into the <INSTALL DIR>/oapps/orafin_107/grants directory and run the following command :

   sqlplus <po user>/<passwd> @crpogrnt.sql

   check the "crpogrnt.lst" file for any errors.

   sqlplus <ap user>/<passwd> @crapgrnt.sql

   check the "crapgrnt.lst" file for any errors.

4. Change into the <INSTALL DIR>/oapps/orafin_107/synonyms directory and run the following command :

   sqlplus <apps user>/<passwd> @crsyns.sql

   check the "crsyns.lst" file for any errors.

5. Change into the <INSTALL DIR>/oapps/orafin_107/views directory and run the following command :

   sqlplus <apps user>/<passwd> @crviews.sql

   check the "crviews.lst" file for any errors.

6. Change into the <INSTALL DIR>/oapps/orafin_107/packages directory and run the following command :

   sqlplus <apps user>/<passwd> @crpacks.sql

   check the "crpacks.lst" file for any errors.

7. Change into the <INSTALL DIR>/oapps/orafin_107/triggers directory and run the following command :

   sqlplus <apps user>/<passwd> @crtrigs.sql

   check the "crtrigs.lst" file for any errors.

Following are the tasks to be performed after the completion of a successful install of the integration software.

Retek Notes:

1. The following steps are performed on the UNIX server. The "/" allows the user to access the menu options on the GUI tool.
2. Step 1 sets up a concurrent job that will validate the PO data, load it to the Oracle base tables, and run a report.
3. Steps 2 & 3 relate to proper setup of the reports. If you have questions with any of these 3 steps, contact Oracle technical support.

---

1. Create/set up Accounts Payables Executables and Concurrent Programs.
2. Set up Flexfield.

---

1. Log in Foundation "found" as any user with with "System Administrator" Responsibility. found

{user name}

{password}

System Administrator

Create/Setup Accounts Payables Executables and Concurrent Programs.

Define Concurrent Program Executables by enter the following information:

\ Navigate Concurrent Program Executable

Create concurrent program executables:

Application: Oracle Payables

Name: PO Validation

Description: PO Validation

Execution File: po_validate_cai.po_val_cai

Subroutine:

Execution Method: PL/SQL Stored Procedure

\ Screen Save

Application: Oracle Payables

Name: PO Load

Description: PO Load

Execution File: poload_cai.po_load_drv_cai

Subroutine:

Execution Method: PL/SQL Stored Procedure

\ Screen Save

Add PO Exception Report:

Application: Oracle Payables

Name: PO Exception Report

Description: PO Exception report

Execution File: poex

Execution Method: Oracle Reports

\ Screen Save

Define Concurrent Program

\ Navigate Concurrent Program Define

Application: Oracle Payables

Name: PO Validation

Short Name: POVALIDATION

Executable: PO Validation

\ Screen Save

Same thing for PO Load

Add PO Exception Report:

Application: Oracle Payables

Name: PO Exception Report

Short Name: POEXRPT

Executable: PO Exception Report

\ Screen Save

2. For the report sets, create the reports that make up this report set.

Choose the System Administrator responsibility.

Navigate: \ navigate report sets

Application: Oracle Payables

Name: PO Interface

\ zone next

Next enter the three reports in the sequence as follows:

Sequence: 1

Application: Oracle Payables

Name: PO Validation

More: No

Sequence: 2

Application: Oracle Payables

Name: PO Load

More: No

Sequence: 3

Application: Oracle Payables

Name: PO Exception Report

More: No

\ screen save

3. If the reports25 is not installed the following commands will

fail. Run them manually after installing reports25 :-

r25convm batch=yes dunit=character overwrite=yes source=poex

dtype=repfile stype=rdffile userid=<apps user>/<passwd>

cp poex.* $AP_TOP/srw/.

Create Synonyms From Rms 9.0.0.0 To Oracle Financials Objects

If using Oracle Applications, create synonyms to the Oracle Applications tables for the *RMS 9.0.0.0 schema owner*. The following synonyms point the RMS 9.0.0.0 schema to the Oracle Financials database objects. To reduce administration efforts, if feasible, create PUBLIC synonyms. The following is a list of the Oracle Applications tables/views that require synonyms from the *RMS 9.0.0.0* schema owner:

AP_EXPENSE_REPORT_LINES_ALL

AP_EXPENSE_REPORT_HEADERS_ALL

AP_EXPENSE_REPORT_HEADERS_S

AP_TERMS

AP_TERMS_EXPORT_CAI

AP_TERMS_V

AP_TERMS_LINES

AP_VENDORS_V

AP_VENDORS_SITES_V

FND_CLIENT_INFO

FND_ID_FLEX_SEGMENTS

FND_FLEX_VALUES

FND_FLEX_VALUE_SETS

GL_CODE_COMBINATIONS_CAI_V

GL_CODE_COMBINATIONS

GL_DAILY_CONV_RATES_CAI_V

GL_INTERFACE

GL_PERIODS

GL_SETS_OF_BOOKS

HR_EMPLOYEES

PO_DISTRIBUTIONS_INTERFACE_CAI

PO_FRT_TERMS_EXPORT_CAI

PO_HEADERS_INTERFACE_CAI

PO_INTERFACE_ERRORS_CAI

PO_LINES_INTERFACE_CAI

PO_LOOKUP_CODES

PO_SHIPMENTS_INTERFACE_CAI

PO_VENDORS_EXPORT_CAI

PO_VENDOR_CONTACTS

PO_VENDOR_SITES

PO_VENDOR_SITES_ALL

PO_VENDORS

SAMPLE SYNONYM SCRIPT

There is a script in the <INSTALL_DIR>/rms9000/oapps/orafin_107/rms directory that can be used as an example of how to set up the synonyms above. This script, crsyns.sql (and drsyns.sql) contains syntax appropriate to apply via database links. These synonyms are public synonyms but private synonyms can be used as well. Please note that the names used for the links indicate the different schema owners the link is tied to. Ex.: PO_LINK.WORLD means the object lives in the PO owning schema in the Oracle Financials Database. Please edit these scripts appropriately for the environment.

Financial Interface Installation modification

Before you can upload vendor files from the financials system online, one change must be made to the stand40.pll library. In the library, open the program unit P_UPLOAD. The highlighted sections included below need to be removed/added from the shipped version of the stand40.ppll library. The program unit is shipped as follows:

Shipped version (remove highlighted):

```
PROCEDURE P_UPLOAD IS
   L_financial_ap  VARCHAR2(1); Financial Interface Installation modificaion
   L_error_message VARCHAR2(255);

BEGIN
   imessage('CONSULT_DOCUMENTATION');

 /*  if SYSTEM_OPTIONS_SQL.GET_FINANCIAL_AP(L_error_message,
                                          L_financial_ap) = FALSE then
      emessage(L_error_message);
      raise FORM_TRIGGER_FAILURE;
   end if;
   ---
   if L_financial_ap = 'O' then
      -- call to load Retek staging tables with terms info
      if ORA2RMS.GET_TERMS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if TERMS_VALIDATE_SQL.VALIDATE_TERMS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      -- call to load Retek staging tables with vendor info
      if ORA2RMS.GET_VENDORS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
```

```
      --  with data from Retek staging tables
      if VENDOR_VALIDATE_SQL.VALIDATE_VENDOR(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
   end if;
   ---
   if L_financial_ap = 'P' then
      -- call to load Retek staging tables with terms info
      if PS2RMS.GET_PS_TERMS_STG(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if TERMS_VALIDATE_SQL.VALIDATE_TERMS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      -- call to load Retek staging tables
      if PS2RMS.GET_VENDORS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if VENDOR_VALIDATE_SQL.VALIDATE_VENDOR(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
   end if;
   ---
   -- Commit New Data
   COMMIT;
   -- Tell user upload complete
   imessage('UPLOAD_COMPLETE');*/

EXCEPTION
   when FORM_TRIGGER_FAILURE then
      raise;
   when OTHERS then
      emessage(SQLERRM);
      emessage('UPLOAD_FAILED');
      raise FORM_TRIGGER_FAILURE;
END;
```

For the Oracle Financial Interface, change stand40.pll (P_UPLOAD program unit) to this:

```
PROCEDURE P_UPLOAD IS
   L_financial_ap  VARCHAR2(1);
   L_error_message VARCHAR2(255);

BEGIN
 if SYSTEM_OPTIONS_SQL.GET_FINANCIAL_AP(L_error_message,
                                        L_financial_ap) = FALSE then
      emessage(L_error_message);
      raise FORM_TRIGGER_FAILURE;
   end if;
   ---
   if L_financial_ap = 'O' then
      -- call to load Retek staging tables with terms info
      if ORA2RMS.GET_TERMS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if TERMS_VALIDATE_SQL.VALIDATE_TERMS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      -- call to load Retek staging tables with vendor info
      if ORA2RMS.GET_VENDORS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if VENDOR_VALIDATE_SQL.VALIDATE_VENDOR(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
   end if;
   ---
   /*
   if L_financial_ap = 'P' then
      -- call to load Retek staging tables with terms info
      if PS2RMS.GET_PS_TERMS_STG(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if TERMS_VALIDATE_SQL.VALIDATE_TERMS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      -- call to load Retek staging tables
```

```
      if PS2RMS.GET_VENDORS(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
      --  call to validation program to load Retek base tables
      --  with data from Retek staging tables
      if VENDOR_VALIDATE_SQL.VALIDATE_VENDOR(L_error_message) = FALSE then
         emessage(L_error_message);
         raise FORM_TRIGGER_FAILURE;
      end if;
   end if;
   */---
   -- Commit New Data
   COMMIT;
   -- Tell user upload complete
   imessage('UPLOAD_COMPLETE');

EXCEPTION
   when FORM_TRIGGER_FAILURE then
      raise;
   when OTHERS then
      emessage(SQLERRM);
      emessage('UPLOAD_FAILED');
      raise FORM_TRIGGER_FAILURE;
END;
```

Migrate the new library when complete and then vendor uploads can be completed online.

Appendix J

Setting up multiple Retek Merchandising 9.0.0.0 Environments

Once the initial install of the Retek Merchandising 9.0.0.0 software is complete, additional Database Server environments can be created by applying the following concepts to your set-up.

The Database Server environment consists of a database schema which contains all of the RMS 9.0.0.0 tables, indexes, views, packages, procedures, and functions, as well as the UNIX directory structure which holds the batch libraries and source code/executables. For each environment, the UNIX directory structure is usually owner by a different UNIX user and the RMS 9.0.0.0 schema is owned by an Oracle user that is associated with the UNIX user. Once the initial install is complete, it is relatively easy to create new Database Server environments.

To create a new database schema, invoke the install.rms script found in <INSTALL_DIR>/install. When prompted enter the new environment's schema owner, password, and temp tablespace.

To create the new UNIX directory structure that will be associated with the new database schema, create a new directory to house the environment, and do a recursive copy of the <INSTALL_DIR>/rms9000 directory to the new directory as the UNIX user that will "own" the new environment. Once the new directory structure is in place you will need to set-up the "owning" UNIX user's environment by modifying and using the sample profile found in <INSTALL_DIR>/sample_profiles.

Remember: When setting up the new environment:

The $MMUSER environment variable should be set equal to the "new" RMS 9.0.0.0 schema owner

The $PASSWORD environment variable should be set to the password for the $MMUSER.

The $MMHOME environment variable should be set to the full path of the environment (i.e. /<INSTALL_DIR>/<your_new_environment_directory>)

Appendix K

Installation Instructions Retek Sales Audit 9.0.0.0 - Oracle Site Fuels Management

Database Server Update Instructions

Use the instructions in this addendum to update your installation of Retek Sales Audit (ReSA) 9.0.0.0 (as part of Retek Merchandising System 9.0.0.0) in order to interface Oracle Site Fuels Management (SFM). The instructions describe how to:

1. Add additional Foundation Data values to RMS tables
2. Set up the SFM schema owner prior to installing SFM
3. Update the RMS schema
4. Install RMS database triggers
5. Update the RMS batch file

Note: The instructions in this addendum assume that Retek Merchandising System 9.0.0.0 is installed and correctly configured. Please read these instructions fully before updating your Retek Sales Audit installation.

To Add additional Foundation Data values:

As the RMS 9.0.0.0 schema owner log into Oracle and run the following insert statements:

```
INSERT INTO RTK_ERRORS
(RTK_KEY, RTK_LANG, RTK_TEXT, RTK_USER, RTK_APPROVED)
VALUES
('SFM_CLOSE_FAIL', 1, 'The perform full close procedure for SFM
failed. The store/day is not closed in SFM.',
'<RMS 9.0.0.0 schema owner>', 'N');


INSERT INTO CODE_DETAIL
(CODE_TYPE, CODE, CODE_DESC, REQUIRED_IND, CODE_SEQ)
VALUES
('LSEC', 'LSFM', 'Site Fuels Management', 'Y', 16);
```

To Set up the SFM Schema:

Ensure that you properly set-up the SFM schema owner before installing the SFM schema. The SFM schema owner should be granted the following privileges:

- Developer
- DBA
- Select any table
- Execute any procedure

The SFM schema owner should have private synonyms to all RMS schema owner objects. Use the 'move_user.sql' script (see appendix E in this Installation Guide). The 'move_user.sql' script prompts you for the user you are creating the synonyms for, and the user that owns the objects. You will also need to create public synonyms for all of the SFM schema owner objects in the database.

To Update the RMS Schema:

Update the RMS 9.0.0.0 schema. Change directories to <INSTALL_DIR>/oapps/sfm and log into Oracle as the RMS schema owner, run the following in the listed order:

1. satgasdriveof.pls
2. satgassales.pls
3. satpumptest.pls

This is a list of the custom total packages that are required for the interface between ReSA and SFM. The three packages listed above define total calculation rules, which you must also define using the total calculation wizard. (Note that you can find the total calculation wizard on the ReSA form `total.fmb`.) For example, the total `GASDRIVEOF` corresponds to satgasdriveof.pls from the list. Also, associate each total with the correct usage type. For instance, the total `GASDRIVEOF` requires the usage type `SFMD`.

The table lists each of the three totals, the correct usage type for each total, and the corresponding package name:

Total	Usage Type	Package
GASDRIVEOF	SFMD	satgasdriveof.pls
GASSALES	SFMS	satgassales.pls
PUMPTEST	SFMP	satpumptest.pls

The STORE_DAY_SQL database module will need to be recompiled in order to perform a day close in SFM when it is initiated in Retek Sales Audit (ReSA) 9.0.0.0. Change directories to <INSTALL_DIR/oapps/sfm, log into Oracle as the RMS 9.0.0.0 schema owner and run the following in the listed order:

1. sastdays.pls
2. sastdayb.pls

Install Oracle Site Fuels Management

Before continuing with the instructions, install Oracle Site Fuels Management (SFM) API as directed in the SFM Install Document.

To Install RMS Database Triggers:

1. Change directories to <INSTALL_DIR> /oapps/sfm.
2. Log into Oracle as the RMS schema owner and run each trigger.

   @satrcclo.trg

   @satrcexc.trg

   @satrsdo.trg

   @satrsdat.trg

To Update the RMS Batch File:

The RMS 9.0.0.0 batch install and compile script will prompt you if you have the SFM interface installed and will take the appropriate action for compiling the ReSA batch module that interfaces SFM. You may now continue on with the installation procedures.

Application Sever Update Instructions

Use the instructions in this addendum to update your installation of Retek Sales Audit (ReSA) 9.0.0.0 Application Server Forms(as part of Retek Merchandising System 9.0.0.0) in order to interface Oracle Site Fuels Management (SFM). The instructions describe how to:

1. Compile the Store Day Summary form (sastrday.fmb) to call STORE_DAY_SQL.PERFORM_SFM_FULL_CLOSE if the current store in the system is a fuel store. This version of the sastrday.fmb form will replace the one compiled during the normal install of the Application server.

Compile the Store Day Summary Form

To compile the Store Day Summary form, cd to <INSTALL_DIR>/oapp/sfm directory. Set up the UP environment varible as mentioned in the "Application Server Software Installation Instructions" section of this install guide. Use the fmb2fmx script located in that directory to compile the sastrday.fmb form.

chmod 755 fmb2fmx

./fmb2fmx

This compiles the form and should generate an executable with an extension of .fmx.

Move compiled object to bin directory

Move the forms executable to the bin directory.

mv *.fmx <INSTALL_DIR>/rms9000/forms.bin

You may now continue on with the installation procedures.