Value-Added Reseller (VAR) Language

(i) the software component known as ACUMATE developed and licensed by Lucent Technologies Inc. of Murray Hill, New Jersey, to Oracle and imbedded in the Oracle Retail Predictive Application Server – Enterprise Engine, Oracle Retail Category Management, Oracle Retail Item Planning, Oracle Retail Merchandise Financial Planning, Oracle Retail Advanced Inventory Planning and Oracle Retail Demand Forecasting applications.

(ii) the MicroStrategy Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(iii) the SeeBeyond component developed and licensed by Sun Microsystems, Inc. (Sun) of Santa Clara, California, to Oracle and imbedded in the Oracle Retail Integration Bus application.

(iv) the Wavelink component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Store Inventory Management.

(v) the software component known as Crystal Enterprise Professional and/or Crystal Reports Professional licensed by Business Objects Software Limited (“Business Objects”) and imbedded in Oracle Retail Store Inventory Management.

(vi) the software component known as Access Via™ licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(vii) the software component known as Adobe Flex™ licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

(viii) the software component known as Style Report™ developed and licensed by InetSoft Technology Corp. of Piscataway, New Jersey, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

(ix) the software component known as i-net Crystal-Clear™ developed and licensed by I-NET Software Inc. of Berlin, Germany, to Oracle and imbedded in the Oracle Retail Central Office and Oracle Retail Back Office applications.

(x) the software component known as WebLogic™ developed and licensed by BEA Systems, Inc. of San Jose, California, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.

(xi) the software component known as DataBeacon™ developed and licensed by Cognos Incorporated of Ottawa, Ontario, Canada, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.
## Contents

**Preface** ............................................................................................................................. vii

Audience ...................................................................................................................................... vii

Related Documents .................................................................................................................. vii

Customer Support .................................................................................................................. vii

Review Patch Documentation ................................................................................................. vii

Oracle Retail Documentation on the Oracle Technology Network .......................................... vii

Conventions .............................................................................................................................. viii

1 **Pre-Installation** ............................................................................................................... 1

   Requirements ....................................................................................................................... 1

      Database Server Requirements ...................................................................................... 1

      Application Server Requirements ............................................................................... 1

      Web Browser and Client Requirements ....................................................................... 1

   Supported Products ............................................................................................................. 2

      Supported Oracle Retail Products ................................................................................ 2

      Supported Oracle Retail Integration Technologies ..................................................... 2

      Supported Third-Party Applications ............................................................................ 2

2 **Introduction** ..................................................................................................................... 3

3 **Install SeeBeyond e*Gate Integrator** ............................................................................... 5

4 **RIB eWays in eGate Schema** .......................................................................................... 7

   Import ................................................................................................................................. 7

      Preliminary Steps ......................................................................................................... 7

      Create a New Schema .................................................................................................. 10

      Import RIB Components ............................................................................................. 11

   RIB Schema Configuration .............................................................................................. 14

      Step 1: Modify the Main Participating Host and Control Broker Configuration ... 15

      Step 2: Modify the JMS IQ Manager Configuration .................................................... 17

      Step 3: Modify Connection Point Configurations ....................................................... 19

      Step 4: Delete Unused e*Ways and Unused JMS Queue ......................................... 25

      Step 5: Add/Copy e*Ways for Additional Components ............................................ 30

      Step 6: Edit the rib.properties File to Correspond to the System ......................... 31

      TAFR Specific Entries in rib.properties ..................................................................... 31

      Step 7: Create/Modify Startup Scripts ....................................................................... 32

      Step 8: Update the hibernate.cfg.xml ......................................................................... 33

      Step 9: Copy classes12.zip file to Server and Client Classpath to Utilize jdbc

          10.2.0.1 Driver ........................................................................................................ 33

      Step 10: Configure component.xml .......................................................................... 33

      Step 11: Starting the RIB Components for a New/Modified Schema .................... 34

5 **Database Triggers and Oracle Dependencies** ................................................................ 35

   Database Triggers ............................................................................................................. 35
Oracle Dependencies............................................................................................................35
Verify RIB Error Hospital Database Tables.................................................................35
RIB_DOCTYPE Table and DTD Files..............................................................................38
6 RIB Hospital Administration Tool Installation..........................................................39
   Prerequisites .............................................................................................................39
   Install RIB Administration Tool ...............................................................................39
7 RIB Java EE Integration for OAS ..............................................................................43
   RIB-<App> Installation ............................................................................................43
      Prerequisite ..........................................................................................................43
      Information Gathering about Retail Application (<app>, not rib-<app>) ...............44
      Information Gathering About rib-<app> Application ...........................................44
      Preparing OAS .......................................................................................................45
      Verification of rib-<app>-oc4j-instance Configuration ...........................................46
      Starting up the Dependent Software’s that rib-<app> Requires ..............................47
      Installing rib-<app> on Newly Created rib-<app>-oc4j-instance .............................47
      Configuring the JNDI Information in the Retail Application <app> (Not rib-<app>) ......................................................................................................................54
      Verification of rib-<app> install ..............................................................................54
   RIB-<App> Maintenance ...........................................................................................55
      RIB Deployment Command-line Utility .................................................................55
8 SIM Integration ............................................................................................................57
   Manual Steps .............................................................................................................57
   SIM Reference .........................................................................................................60
      rib.properties File ..................................................................................................60
      Example of a Messaging Component Configuration File ........................................61
A Appendix: RIB Installation Checklist .......................................................................65
   SeeBeyond e*Gate Installation Checklist .................................................................65
   Oracle Retail RIB Installation Checklist (SeeBeyond)...................................................66
   Oracle Retail RIB Configuration Checklist (SeeBeyond) .............................................67
   Oracle Retail RIHA Installation Checklists ...............................................................68
   SeeBeyond Install Notes ..........................................................................................69
   Oracle Retail RIB Installation Checklist (JavaEE – OAS) .............................................73
Oracle Retail Installation Guides contain the requirements and procedures that are necessary for the retailer to install Oracle Retail products.

Audience
This Installation Guide is written for the following audiences:
- Database administrators (DBA)
- System analysts and designers
- Integrators and implementation staff

Related Documents
For more information, see the following documents in the Oracle Retail Integration Bus Release 12.0.1.4 documentation set:
- Oracle Retail Integration Bus Release Notes
- Oracle Retail Integration Bus Operations Guide
- Oracle Retail Integration Bus Integration Guide

Customer Support
https://metalink.oracle.com
When contacting Customer Support, please provide the following:
- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

Review Patch Documentation
For a base release (".0" release, such as 12.0), Oracle Retail strongly recommends that you read all patch documentation before you begin installation procedures. Patch documentation can contain critical information related to the base release, based on new information and code changes that have been made since the base release.

Oracle Retail Documentation on the Oracle Technology Network
In addition to being packaged with each product release (on the base or patch level), all Oracle Retail documentation is available on the following Web site:
http://www.oracle.com/technology/documentation/oracle_retail.html
Documentation should be available on this Web site within a month after a product release. Note that documentation is always available with the packaged code on the release date.
Conventions

**Navigate:** This is a navigate statement. It tells you how to get to the start of the procedure and ends with a screen shot of the starting point and the statement “the Window Name window opens.”

**Note:** This is a note. It is used to call out information that is important, but not necessarily part of the procedure.

This is a code sample
- It is used to display examples of code

A hyperlink appears like this.
### Pre-Installation

**Requirements**

The following hardware and software is required in order to run Oracle Retail Integration Bus (RIB).

#### Database Server Requirements

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle RDBMS</td>
<td>Oracle 10gR2 (10.2.0.2)</td>
</tr>
<tr>
<td>UNIX based OS certified with Oracle Application Server 10g 10.1.3</td>
<td>IBM AIX 5.3,TL3</td>
</tr>
<tr>
<td></td>
<td>SUN Solaris(SPARC)9 09/05</td>
</tr>
<tr>
<td></td>
<td>Oracle Enterprise Linux v4</td>
</tr>
<tr>
<td></td>
<td>HP-UX (PA-RISC) 11.23</td>
</tr>
</tbody>
</table>

#### Application Server Requirements

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIX based OS certified with Oracle Application Server 10g 10.1.3</td>
<td>IBM AIX 5.3,TL3</td>
</tr>
<tr>
<td></td>
<td>SUN Solaris(SPARC)9 09/05</td>
</tr>
<tr>
<td></td>
<td>Oracle Enterprise Linux v4</td>
</tr>
<tr>
<td></td>
<td>HP-UX (PA-RISC) 11.23</td>
</tr>
<tr>
<td>Oracle Application Server</td>
<td>OAS 10.1.3 (+patches)</td>
</tr>
</tbody>
</table>

#### Web Browser and Client Requirements

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Any</td>
</tr>
<tr>
<td>Processor</td>
<td>Any</td>
</tr>
<tr>
<td>Web Browser</td>
<td>Any HTML 2.0 compliant</td>
</tr>
<tr>
<td>Networking</td>
<td>TCP/IP</td>
</tr>
</tbody>
</table>
Supported Products

The following products are supported by RIB.

Supported Oracle Retail Products

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Retail Merchandising System (RMS)</td>
<td>12.0.5</td>
</tr>
<tr>
<td>Oracle Retail Price Management (RPM)</td>
<td>12.0.5</td>
</tr>
<tr>
<td>Oracle Retail Store Inventory Management (SIM)</td>
<td>12.0</td>
</tr>
<tr>
<td>Oracle Retail Allocation</td>
<td>12.0.5</td>
</tr>
<tr>
<td>Oracle Retail Warehouse Management System (RWMS)</td>
<td>12.0.5</td>
</tr>
</tbody>
</table>

Supported Oracle Retail Integration Technologies

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Retail Extract, Transform and Load (RETL)</td>
<td>12.0.1</td>
</tr>
<tr>
<td>Oracle Retail Service Layer (RSL)</td>
<td>12.0.1</td>
</tr>
</tbody>
</table>

Supported Third-Party Applications

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hibernate</td>
<td>2.1.8</td>
</tr>
<tr>
<td>SeeBeyond</td>
<td>EGate 5.0.5 SRE</td>
</tr>
</tbody>
</table>
This manual details the installation of the Retail Integration Bus (RIB). Generally an installation of the RIB contains the following components:

- An installation of the SeeBeyond e*Gate Integrator Product (version 5.0.5). An installation of the SeeBeyond e*Gate Integrator product involves installing the registry host and all participating host software, plus Graphical User Interface hosts for development and system monitoring. See Chapter 2 for details.

- An installation of the RIB eGate schema is imported into the e*Gate Integrator product. This is explained in Chapter 3.

- Configuration points to update the database connection points, JMS queues, and CLASSPATH configuration values. Also, unused adapters are deleted. This is explained in Chapter 3.

- An installation of the Retail Integration Error Hospital administration (RIHA) tool. This is covered in Chapter 5.

- An installation of RIB’s Java EE application on a Java EE 1.3 compliant Server. This is covered in Chapter 6.

- An installation of the RIB Diagnostics and Monitoring tools. This is covered in the RIB document entitled “RIB Diagnostics and Monitoring Guide”.

Note: It is imperative to follow all installation steps of the applications that are being connected to the RIB. Failure to follow these may result in a faulty RIB installation. See the install guides of the relevant Retail applications for more information.

Note: Follow the RIB Installation Checklist in Appendix A while performing the install in order to minimize the chance of a faulty RIB installation.
Below is a high-level picture of the software that needs to be installed in various containers in order to install RIB.

**RIB Deployment Diagram**

![RIB Deployment Diagram]

**Full Install**
A full install is typically performed when a customer has a new release of the RIB and has not had a previous install of the RIB, or if the customer is performing a mandatory upgrade and does not expect to save any previous portions of their RIB install. A full install involves a new and clean installation of the RIB and as such, includes every component in the above picture.

**“Upgrading From A Previous Release” or Delta Install**
An Upgrade/Delta install to assist on upgrading from the previous release of the RIB is provided as an installation option.

**Note:** This release of the RIB always has to be a full, new installation.
Install SeeBeyond e*Gate Integrator

The Retail Integration Bus (RIB) leverages SeeBeyond’s e*Gate Integrator for supplying the needed messaging facilities for integrating applications.

The following steps need to be completed successfully to install the e*Gate product:

1. A Registry host needs to be installed, which will contain the central database of the message formats, as well as publication, subscription and transformation logic.

2. At least one Participating host needs to be installed, which implements the publishers, subscribers and transformations.

3. The required e*Gate add-ons need to be installed.
   - Batch e*Way 5.0.5 add-on
   - Oracle e*Way 5.0.5 add-on (included in this is the JDBC e*Way)

4. The GUI hosts need to be installed, which are used to monitor the operation of the system and to extend or further develop the system’s capabilities.
   - e*Gate GUI 5.0.5

5. There may be a number of required e*Gate ESR (patches) to be installed. A listing of any of these will be included in the RIB Release Notes. ESRs can be found on Retail’s fulfillment center (http://metalink.oracle.com/).

Notes:

All three types of hosts can be present on the same physical machine. However, GUI hosts must execute on a Microsoft Windows platform.

e*Gate requires a Java Runtime Environment (JRE) of at least version 1.3.1. However, the RIB requires a JRE of 1.4.1 or above, as does the RIB installation procedure. eGate provides an ESR to upgrade the JRE from 1.3.1 to 1.4. This ESR must be installed before trying to install RIB eways.

e*Gate Schema Manager and e*Gate Schema Designer applications use the Exceed X-windows application. If a version of Exceed exists on a GUI host, then one must install the e*Gate version into a different directory. The e*Gate version is not a full installation of Exceed.

The instructions for installing the SeeBeyond e*Gate Integrator system are documented in the e*Gate Integrator Installation Guide. This document is found on Disk 2 of the SeeBeyond installation disk set (docs\eGate_Install_Guide.pdf).
**Note**: The following maps the e*gate zip files available at [http://edelivery.oracle.com](http://edelivery.oracle.com) to SeeBeyond’s actual CD Names:

egate505hosts.zip:
SeeBeyond ICAN Suite e*Gate Integrator for SRE 5.0.5 Disc 1

egate505gui.zip:
SeeBeyond ICAN Suite e*Gate Integrator for SRE 5.0.5 Disc 2

egate505addons.zip:
SeeBeyond ICAN Suite Add-ons/Samples for SRE 5.0.5 Disc 4
RIB eWays in eGate Schema

Import

The RIB software is distributed in a single eGate messaging schema. This schema contains all of the RIB’s publishing and subscribing e*Ways (adapters) and Connection Points. It also contains a single JMS Intelligent Queue Manager.

Once the RIB schema has been imported, a system administrator must configure the connection points. Additional configuration modifications may also be needed, such as e*Way CLASSPATH.

The final modifications to the system are due to the site-specific deployment of the system. These changes include distributed components to different hosts, creating failover hosts, developing additional event types, adapters, connection points and collaborations for integrating an enterprise’s non-Oracle Retail applications to the RIB. It also includes creating security roles and privileges. These activities are not considered part of the installation and are not documented in this manual. For more information on these activities, see the SeeBeyond e*Gate Users Guide.

Preliminary Steps

To create and import the RIB schema, take the following preliminary steps:

1. For security reasons, create an “egate” UNIX user that will own the e*Gate files and execute the software.
2. Log onto the UNIX system where e*Gate was installed using this account.
3. Copy the RIB tar files from the RIB installation CD(s) to the location where you are planning to install the RIB software. This location is known as the RETEK_INSTALL_DIR in the remainder of this section. The RIB tar files include ribcommon1201_eng_ga.tar, and ribobjects1201_eng_ga.tar. The RIB tar files for each Retail applications are named ribpak<RIB_version>for<APP><APP_version>_eng_ga.tar (where the application is RWMS, RMS, SIM, AIP, RPM etc.). An example of the <RETEK_INSTALL_DIR> directory name could be called “INSTALL”, located directly under the “egate” user’s home directory. Future releases of the RIB should be installed into this directory, as the directories will have new version numbers in their names. (For example: /files0/egate/INSTALL/)

Note: It is recommended to preserve the RETEK_INSTALL_DIR to serve as future reference for what is installed on a host system. Additionally, auxiliary RIB tools such as the Diagnostics and Monitoring tool may rely on this to poll for information.
4. Once you have copied the RIB tar file(s) to <RETENK_INSTALL_DIR>, extract each file in this directory and change the permissions on the extracted files to make them writable. If you are installing multiple applications with RIB, be sure to copy and extract the tar files of each ribpak for each application you intend to install.

   **Note:** Extract the ribpak for rms tar file after extracting other tar files if you are installing RIB for RMS. (for example, run this command last. tar xvf ribpak1201formrms1201_eng_ga.tar). This dependency is necessary because component.xml will be overwritten if other applications PAKs are installed after the RIB PAK for RMS. As a result, it is necessary to extract the component.xml from the RMS PAK and copy that component.xml to $EHOME/client/classes

   ```
tar xvf ‘filename’

   chmod –R 755 *
   ```

5. Downloading hibernate.

   RIB uses Open Source O-R mapping tool called hibernate (http://www.hibernate.org/). Due to licensing limitation RIB cannot ship hibernate along with its package, so hibernate 2.1.8 has to be download by the person installing rib-<app>. RIB has been certified with only hibernate 2.1.8, do not download any other hibernate version.

   Download hibernate (hibernate-2.1.8.zip) and extract the hibernate2.jar file from inside the zip file. Copy the hibernate2.jar to < RETENK_INSTALL_DIR >/RIB<Version>/external-lib directory.

   ```
cp <your dir>/hibernate2.jar <RETENK_INSTALL_DIR>/RIB<Version>/external-lib
   ```

6. Change directories to <RETENK_INSTALL_DIR>/RIB<version>.

7. Edit the file egate_profile. Make sure the settings for the following variables are correct for your environment.

   - **EHOME** – The directory where SeeBeyond e*Gate was installed.
   - **RETENK_INSTALL_DIR** – The directory created in step 3 above.
   - **EGATE_SERVER_NAME** – The name or IP address of the server e*Gate and the RIB software are installed on.
   - **EGATE_SERVER_PORT** – The port that the e*Gate Registry Host was installed on during the SeeBeyond e*Gate Integrator install.
   - **RIB_SCHEMA** – The name of the eGate Schema into which you wish to import the RIB SeeBeyond components. This value must match the name of the Schema you create in step 3 of Create a new schema below.
   - **Platform specific section (Sun Solaris, IBM AIX, HP-UX)** – Uncomment the section that is applicable to your operating system and ensure that the other two sections are commented out.
8. Edit the “egate” user’s .profile located in the “egate” user’s home directory. Add an entry at the end of this file that “sources” the egate_profile modified in step 6 above.
   - `<RETEK_INSTALL_DIR>/RIB<version>/egate_profile`
     (for example: `./files0/egate/INSTALL/RIB<version>/egate_profile`)
   - Ensure that ‘:’ (dot colon) is at the beginning of the egate user’s PATH variable.
     (for example: `PATH=.:${PATH} ; export PATH`)
   “source” the .profile after making these modifications or start a new Unix session before continuing.

   **Note:** The e*Gate install procedure modifies your profile in a few ways, so make sure your java version is JRE 1.4.1 or above. Type “java --version” at the UNIX command line to verify before you continue.

9. If there was an earlier attempt at installing a previous version of the RIB, it must be inactivated by renaming it:
   a. Make sure that all e*Ways, the control broker, and the registry are shut down.
      On Unix, the following command will show the active processes:
      > `ps -ef | grep stc | grep -v grep`
      If stc processes are still running, be sure to shut down all the stc processes (for example: `kill - 9`).
   b. Rename the `$EHOME/server/registry/<RIB_SCHEMA>.rdb` file to `$EHOME/server/registry/<RIB_SCHEMA>.rdb.bak`.
   c. Rename the `$EHOME/server/registry/repository/<RIB_SCHEMA>` directory to `$EHOME/server/registry/repository/<RIB_SCHEMA>.bak`.
   d. The `<RIB_SCHEMA>.rdb.bak` file and `<RIB_SCHEMA>.bak` directory can be deleted at a later time once the new version has been successfully installed.

10. Start the e*Gate registry by using the “start_egate” script; it is located at:
    `<RETEK_INSTALL_DIR>/RIB<version>/Retek_Sbyn`
Create a New Schema

The RIB schema is imported through a two-step process. The first step involves creating a new schema. This new schema is empty and does not contain any RIB modules.

1. Log in to the e*Gate registry using the e*Gate Schema Designer GUI tool. Log in as Administrator, using the password that was set during the installation of e*Gate Integrator.

**Note:** For the Server: field, enter the EGATE_SERVER_NAME or IP address as specified in the egate_profile modified previously. Also, because the e*Gate Registry Host may not be running on the default port (23001), it is good practice to always specify the port along with the hostname. The format is <hostname>:<port>, (for example, mspdev14:23001).
2. Click **New** to create a new schema. The New Schema dialog box is displayed.

3. In the Enter New Schema Name field, enter a schema name. It is recommended to name this schema RIB<POSTFIX>. A few recommendations would be to name this RIB_PROD (for production), RIB_UAT (for user acceptance test environment), RIB_SYS (for system test environment), etc. This should be named appropriately and uniquely for the intent and use of this schema. (The name entered must match the value set as RIB_SCHEMA in your egate_profile referenced above.)

4. Click Open.

You have now successfully created an empty schema named “RIB<POSTFIX>”.

**Import RIB Components**

The second step of the RIB Schema import process entails the actual importing of the RIB components into the newly created “RIB<POSTFIX>” base schema.

**Load RIB Components - Automated Instructions**

A script to register the new RIB Schema, “RIB<POSTFIX>”, and insert all of the registry modules can be found in the following directory:

```bash
<RETEK_INSTALL_DIR>/RIB<version>/Migration_Scripts
```

**Notes:**

If your “RIB<POSTFIX>” schema has a password for the Administrator user that is different than the default of “STC” you will need to edit the install script and replace the two occurrences of “STC” with your password (case sensitive).

The system must be able to locate the “unzip” and “zip” utilities for the importeways script to work. If not, update the PATH variable in the egate user’s .profile file.

Additionally, a working version of perl must be installed on your machine. Be sure the PATH variable contains an entry to where this file resides.

The DISPLAY variable must be set to the IP address of the machine that the install will run on.

For example, export DISPLAY=10.0.0.5:0.0.

```bash
xterm &
```
When installing on an HP-UX platform, the following lines need to be commented out in $EHOME/client/egateclient.sh file

if [ "LD_PRELOAD" = "" ]; then
  LD_PRELOAD=/u00/egatexi/egate/client/JRE/1.3.1_02/lib/PA_RISC2.0/hotspot/libjvm.sl
else
  LD_PRELOAD=/u00/egatexi/egate/client/JRE/1.3.1_02/lib/PA_RISC2.0/hotspot/libjvm.sl:$LD_PRELOAD
fi

export LD_PRELOAD

egateclient.sh and egatereg.sh eGate scripts add /usr/bin to the PATH environment variable. If your environment has java already installed and available in /usr/bin than eGate picks up that (wrong) version of java instead of the one in $EHOME/client/JRE/1.3.1/bin. To fix this issue, edit the scripts by adding $EHOME/client/JRE/1.3.1/bin to the beginning of the PATH environment variable before /usr/bin.

Even though $EHOME/client/JRE/1.3.1/bin directory says 1.3.1 the actual java version installed in that directory is 1.4.1.

Note: Unless all of these requirements are met, the manual method to import the RIB modules will have to be used.

1. From this directory, run the “install” script (eg ./install).
   a. The ./install script requires input parameters defining which version of the RIB you wish to install (for example, ./install 1201). Executing the install script with no parameters will provide you with a listing of valid input parameters. This listing is compiled by scanning the RETEK_INSTALL_DIR for RIB<version> directories, which are created when the user extracts the tar files into the RETEK_INSTALL_DIR.
   b. After providing “./install” with valid input parameter, the user is prompted to verify the order of install.
   c. “./install” prompts the user to create a backup of the <RIB_SCHEMA> schema. Choosing to do so creates a .zip of the schema that is currently installed before proceeding to import new modules.
   d. Following this, “./install” prompts the user for delta or full install. Choosing “No” proceeds to import ALL components included in the release for full install. Choosing “Yes” proceeds the delta install that only installs the RIB jar files to the existing environment without changing the existing configuration and Eways install. Note this (1201) release does not support delta release.

2. Once the script has completed, use the e*Gate Schema Designer to ensure each module was successfully loaded into the schema.
The “install” script creates soft links in $EHOME for the following 6 scripts:

- start_egate \rightarrow <RETEK_INSTALL_DIR>/RIB<version>/Rib_Support/start_egate
- stop_egate \rightarrow <RETEK_INSTALL_DIR>/RIB<version>/Rib_Support/stop_egate
- start_rib \rightarrow <RETEK_INSTALL_DIR>/RIB<version>/Rib_Support/start_rib
- Egate.txt \rightarrow <RETEK_INSTALL_DIR>/RIB<version>/Egate/Eways/Egate.txt
- start_cb \rightarrow <RETEK_INSTALL_DIR>/RIB<version>/Rib_Support/start_cb
- plist \rightarrow <RETEK_INSTALL_DIR>/RIB<version>/Rib_Support/plist

To continue with the install process jump to RIB schema configuration.

Load RIB Components - Manual Instructions

These instructions are provided in order to individually load modules into the schema. Skip this section if the automated process was used and completed successfully.

Each RIB module can be loaded into the <RIB_SCHEMA> schema manually, if necessary, by running the “Import Definitions from File” feature of the e*Gate Schema Designer GUI.

1. From a Windows PC, which has the e*Gate GUI installed, put the <MODULE>.zip file(s) to be loaded on an accessible drive.
2. Start the e*Gate Schema Designer.
3. Log in to the <RIB_SCHEMA> schema.
4. Select File > Import Definitions from File. The Import Wizard is displayed.
5. Click Next. The Step 1 page is displayed.
6. Select the “Module” radio button. Click Next. The Step 2 page is displayed.
7. Locate/select the <MODULE>.zip file to be imported into the <RIB_SCHEMA> schema. Click Next.
8. Click Finish. The Import Component dialog box is displayed, asking for confirmation as to which Participating Host/Control Broker pair to import into. Click OK.
9. Repeat the above steps for each <MODULE>.zip file that you wish to import.
10. Continue the install process: RIB schema configuration.
RIB Schema Configuration

After the RIB schema has been imported, the schema must be configured for the site-specific environment. This chapter details the minimum changes needed for the RIB to run. It assumes that the RIB is deployed on a single host and that only a single JMS IQ Manager is needed. This deployment configuration is not appropriate for all RIB installations. Production environment deployments may choose to distribute different specific e*Ways and JMS queues among multiple hosts. This type of production deployment is not covered in this manual.

The following steps are required to configure the RIB schema:

1. Modify the main Participating Host and Control Broker configuration.
2. Modify the JMS IQ Manager configuration.
3. Modify Connection Point configurations.
4. Delete unused e*Ways and unused JMS queue.
5. Add/Copy e*Ways for additional components.
6. Edit the rib.properties file to correspond to the system.
7. Create/modify startup scripts.
8. Update the hibernate.cfg.xml.
9. Copy classes12.zip file to server and client classpath to utilize jdbc 10.2.0.1.0 driver.
10. Review component.xml.
11. Starting the RIB components for a new/modified schema.
Step 1: Modify the Main Participating Host and Control Broker Configuration

The first step in the RIB messaging schema configuration is to modify the main participating host and control broker’s configuration. The <RIB_SCHEMA> schema includes a single participating host and control broker that contains all of the messaging e*Ways and associated components. If these are not modified, then the configuration will attempt to resolve host names and ports as specified by the supplied/shipped configuration.

Alternatively, you can change the configuration of the participating host and its supplied control broker.

Changing these configurations is a manual process performed with the e*Gate Schema Designer application. This application must be installed on a Microsoft Windows 2000 or Microsoft Windows NT platform. Specific platform requirements are detailed in the SeeBeyond e*Gate Integrator Installation Guide.

These instructions modify both the names and IP address of the participating host and control broker. The name of the control broker must match any start-up scripts used.

Modify the Configuration

1. Open the e*Gate Schema Designer.
2. Connect the e*Gate Schema Designer to the <RIB_SCHEMA> schema. The following window is displayed:

The main e*Gate Schema Designer Window
3. Right-click on the first active participating host displayed. A command list is displayed.

**Note:** An active participating host is one *without* the string “(inactive)” as part of its name. If there is not a participating host *without* the “(inactive)” string, refer to the SeeBeyond System Administrator’s Guide for instructions on how to activate the correct participating host.

4. Select Properties. The Participating Host Properties dialog box is displayed.

**Participating Host Properties window**

5. In the Network host name or IP address field, enter the e*Gate server name.

6. In the Network domain name field, enter the correct network domain name for your environment.

7. Click OK.
Step 2: Modify the JMS IQ Manager Configuration

The JMS IQ Manager is initially configured to use the TCP/IP port number of 24053 for all e*Gate publishers and subscribers to connect to. If this port is used by other applications, then the JMS IQ Manager will not be able to be started. Complete this step only if port 24053 is NOT available.

1. In the main e*Gate Schema Designer window, right-click on the iqmJMS queue manager. (The iqmJMS queue manager is towards the bottom of the Components frame, below all of the e*Ways.)

2. Select Properties. The IQ Manager Properties dialog box is displayed.

3. Click on the Advanced tab at the top of the window.
4. In the TCP/IP port number field, change the port number to an available port.
5. Click OK. Note the port number for the next step.

**Note:** If there are multiple instances of e*Gate running on a single UNIX server, close attention must be paid to the registry, control broker and JMS ports. Runtime problems will be encountered if instances share the same ports.
Step 3: Modify Connection Point Configurations

The next step is to modify the Connection Point configurations to reflect the JMS IQ Manager and Oracle databases used. This is performed in the e*Gate Schema Designer application.

From the main window, click on the e*Way Connections folder. The window changes to reflect the available connections.

Connection Points

There are two types of connection points supplied with the RIB: SeeBeyond JMS and Oracle.
Change SeeBeyond JMS Connection Points

The SeeBeyond JMS connection points must connect to a known JMS IQ Manager. This requires knowledge of both the port number and host name. The host name is the name of the host used in step 1 “Modify the main Participating Host and Control Broker configuration”. The TCP/IP port number is initially set to 24053. Change the TCP/IP port number only if Step 3 changed the port number of the iqmJMS IQ Manager. Otherwise, leave the port number as 24053.

Ensure that the connection point connects to the correct JMS IQ Manager. Both cpJMS.cfg and cpDSDDetailsNoXA.cfg should be configured by using the following steps. Note that in the default installation, all SeeBeyond JMS Connection points except cpDSDDetailsNoXA share the same configuration file, cpJMS.cfg. Only one of the SeeBeyond JMS connections which use cpJMS.cfg needs to be modified to change all of the connection points that use cpJMS.cfg.

1. Locate any one of the SeeBeyond JMS connection points, right click on it and select Properties. The e*Way Connection Properties dialog box is displayed.

![e*Way Connection Properties window](image)

e*Way Connection Properties window
2. To change the address of the JMS IQ Manager the connection point connects to, edit the configuration file from one of the connection points using it. Multiple connection points may use the same connection point for sending messages to and from the JMS queue. The RIB schema initially uses only a single JMS queue for all messages.

3. Click Edit to change the address of the queue associated with the e*Way Connection Configuration File section of this properties window. The Connection Point configuration file edit dialog box is displayed.
4. In the Goto Section field, select the Message Service section from the drop-down list.

5. In the Host Name field, enter the name of the host on which the JMS IQ Manager resides. If all components are running on the same host, this is the same name as specified in Step 1.

6. If the JMS IQ Manager’s TCP/IP Port number was changed in Step 3, then scroll down to the Port Number field and enter the number used in Step 3.

7. Select File > Save to save the new configuration.

8. Select File > Promote To Runtime to make the configuration change take effect in the schema.

9. Select File > Close to exit the window.
Changing Oracle Database Connection Points

All of the Oracle database Connection points must be altered to reflect the database instance and the user-ID/login for each of the applications.

1. Open the Connection Point Configuration window for the Oracle Connection Points, in the same manner as was done in the previous section for the SeeBeyond JMS Connection points.

Oracle Database Configuration Edit window (DataSource Section)
2. All configuration parameters of interest are found in the DataSource section. The table below lists which parameters should be changed.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>Specifies the name of the Java class in the JDBC driver (Usually oracle.jdbc.xa.client.OracleXADataSource)</td>
</tr>
<tr>
<td>DriverType</td>
<td>This is the JDBC driver type (Usually thin)</td>
</tr>
<tr>
<td>ServerName</td>
<td>Name of server to connect to. Must have a valid Oracle listener.</td>
</tr>
<tr>
<td>PortNumber</td>
<td>Database connection port number. (Usually 1521.)</td>
</tr>
<tr>
<td>DatabaseName</td>
<td>Database System ID (SID).</td>
</tr>
<tr>
<td>user name</td>
<td>Login name to use</td>
</tr>
<tr>
<td>Password</td>
<td>Login password to use. This is stored in an encrypted form and displayed as a series of asterisks.</td>
</tr>
</tbody>
</table>

3. Change all Oracle database connection point settings.
   You can delete the connection points used by publishers and subscribers for applications not installed. However, there is no harm in leaving these connection points as is.

**Note:** Be sure to save and promote to runtime any changes made in the connection point configuration files.
Step 4: Delete Unused e*Ways and Unused JMS Queue

If the entire suite of RIB supported applications are not available or are not used, then delete the e*Ways associated with these applications. For messages that are directed to multiple applications or application instances, the presence of an e*Way will cause the JMS queue to store messages until all subscribers have received them. If a subscriber exists and never starts nor successfully consumes a message, then the JMS queue will never delete its copy of the message. Eventually, the JMS queue will exceed its configured message storage limits and message publication will halt.

1. From the main e*Gate Schema Designer window, click on the Components tab in the lower left corner of the screen.
2. Expand the Participating Hosts folder in the left hand side frame, if not already expanded.
3. Expand the control broker containing all of the RIB e*Ways so that the list of e*Ways is presented.

![Expanded control broker]
4. Highlight the e*Way you wish to delete by clicking on it in the left hand frame. This will give you visibility in the right hand window to the collaboration associated with this e*Way.

Viewing a collaboration

5. Open the collaboration associated with the e*Way you wish to delete by highlighting and double-clicking on it.

Collaboration properties window

6. At this point you must determine if the e*Way you wish to delete is a publishing e*Way or a subscribing e*Way. You can identify a publishing e*Way by determining its source of subscription. This is identified in the Subscriptions section of the Collaboration – properties window (opened in step 5 above). This column indicates which connection settings the e*Way will use. If this connection is to an outside application (i.e. RMS, RDM, SIM, etc) this e*Ways is considered a publishing e*Way. The naming convention employed by Oracle Retail makes these easily identifiable. Examples include cpToAndFromRMS, cpToAndFromRDMWH1, and cpToAndFromSIM.

If you have determined that the e*Way you wish to delete is a publishing e*Way you may now skip ahead to step 8. If the e*Way you wish to delete is a subscribing e*Way, please proceed with the next step.
7. You have determined that your e*Way is a subscribing e*Way. Because this e*Way is a subscribing e*Way, there are additional pieces to remove that if left in a SeeBeyond schema could ultimately cause the performance issues alluded to in the opening paragraph.

When a registry is started in a SeeBeyond schema, it creates a unique subscriber for each subscribing e*Way/collaboration in that schema on the JMSQueue. SeeBeyond then ensures that each subscriber receives a message once. Until all subscribers for a particular message indicate to SeeBeyond that they have handled that message, a copy of the message is kept in the JMSQueue. Because of this “journaling” of messages, we must be sure that if we remove a subscribing e*Way from a SeeBeyond schema, we remove its subscriber as well. If we fail to do this, a copy of the message will be stored in the JMSQueue taking up memory resources.

Removing a Subscriber:

1. To remove a subscriber from a JMSQueue you must first identify the Topic which the e*Way is subscribing to. The Topic can be found by opening the collaboration properties window and looking under the Event Type column in Subscriptions section of the window. Topics will always begin with the two letters “et” (I.E. etASNIn).

Identify the Topic in the Event Type column

2. Be sure the JMS IQ is running. Open the JMS Administrator tool. Instructions on use of the JMS Administrator can be found in the SeeBeyond document – JMS Intelligent Queue User’s Guide – Chapter 5: Managing Events and JMS IQ Managers.

3. From the list of topics given in the JMS Administrator, find the topic that matches the Topic you found in step A above. Once you find your topic, expand the Topic and Subscriber nodes in the GUI. Here you see a list of all collaborations that contain subscribers to this topic. Highlight the collaboration which matches the collaboration whose properties you looked at in step A.
4. Before proceeding to the next step it is essential the user understand the significance of deleting a subscriber. The deletion of a subscriber also deletes any data from the JMSQueue for which that subscriber may be waiting to process. When deleting a subscriber that was not desired as part of a schema, this is of no concern. However, when deleting a subscriber because the functionality of its e*Way has been replaced, special care must be taken to insure no loss of data occurs. Before deleting subscriber which is being replaced by another, be sure that all data in the JMSQueue for the old subscriber has been processed. This can be done by shutting down the publishing e*Ways/process, clicking on the Messages tab in the right hand pane of the JMS Administrator and looking to see that No Message exists for the subscriber you wish to delete. If a message does exist, wait for the subscribing application to finish processing the message before proceeding.

Check to see no messages exist for subscriber
5. Right click on the appropriate subscriber/collaboration and choose “Delete Subscriber” from the dropdown menu (note: the e*Way which corresponds to this subscriber must not be running in order to delete).

Deleting a subscriber

6. Close the JMS Administrator and return to the e*Gate Schema Designer.

7. Right-click on the collaboration. Choose Delete to delete the collaboration. A confirmation dialog box is displayed.

Deleting the collaboration
8. Right-click on the e*Way. Choose Delete to delete the e*Way. A confirmation dialog box is displayed.

9. Log onto the Unix system using the “egate” user that owns the e*Gate files and executes the software.

10. Navigate to the Unix $EHOME directory. The Egate.txt file contains a listing of all of the e*Ways contained the RIB 11.1.0 release. Modify this file to no longer contain a reference to the e*Way you have just removed from your schema.

11. Repeat this process to delete all e*Ways which are not desired.

12. A default JMS server named <hostname>_iqmgr is created after the E*Gate and RIB install. One should right-click the default JMS server in the e*Gate Schema Designer and choose “delete” in the pop-up menu to delete the default JMS server. RIB uses the iqmJMS JMS server only.

**Step 5: Add/Copy e*Ways for Additional Components**

Occasionally, there will be a need to add additional e*Ways to the imported schema. Often times, it is possible to copy an existing e*Way, reconfigure the various pieces that make up the e*Way, and continue from there. If it is necessary to add/copy e*Ways, please refer to the SeeBeyond e*Gate Users Guide for the correct procedures.
Step 6: Edit the rib.properties File to Correspond to the System

Next, edit the rib.properties file to reflect the site-specific mappings and properties. The RIB import process copied a rib.properties file from `<RETEK_INSTALL_DIR>/RIB<version>/Rib_Support/src` to the `$EHOME/client/classes` directory. If a previous version of this file existed, it was renamed to rib.properties.bak. Some of the types of properties in this file are:

- Error Hospital specifics such as the max number of attempts to try for a failing message and the delay between each attempt.
- Multi-channel indication for Purchase Orders.
- Multi-threading settings.
- Facility ID mappings. These must correspond to codes in the RMS application for the correct routing of message to RDM instances.
- Logging specific settings.
- “no event” sleep duration settings.

If the rib.properties file in `$EHOME/client/classes` is not to be used, then append the full path of the directory containing the file that will be used to the beginning of the CLASSPATH variable setting.

TAFR Specific Entries in rib.properties

All of these parameters are TAFR specific and are used in routing the messages to the correct application.

These boolean variables (True/False) are set to indicate which type of applications are active and which combination of to_loc/from_loc each application is designed to handle. If the message being sent contains both a “from location” and a “to location”, then the message could be sent to both the “from” and “to” location applications. However, not all systems can handle this information as a sender and receiver. This property was primarily developed for the SIM application. This application can handle the Transfer message when it is to be sent from a store or if it is going to be shipped to their store. A lot of times the receiving location does not know or care about the transfer prior to receiving the ASN Out or ASN In message. In a pure Oracle Retail environment that contains both RWMS and SIM the setting next to the property below should be used. If message does not meet the conditions of the below properties, the message will be dropped.

StoreSystemActive=True – This states that a store system is active and if a message is being routed it, should be routed to the store topic if the properties below also meet the conditions.

StoreSystemRouteToToLoc=True – This states that if the store is the to_loc of a message (such as a transfer) then the message should be routed to the store application topic.

StoreSystemRouteToFromLoc=True – This states that if the store is the from_loc of a message (such as a transfer) then the message should be routed to the store application topic.

WarehouseSystemActive=True – This states that a store system is active and if a message is being routed it, should be routed to the store topic if the properties below also meet the conditions.

WarehouseSystemRouteToToLoc=False – This states that if the warehouse is the to_loc of a message (such as a transfer) then the message should be routed to the warehouse application topic.
WarehouseSystemRouteToFromLoc=True – This states that if the warehouse is the from_loc of a message (such as a transfer) then the message should be routed to the warehouse application topic.

ExternalSystemActive=False – This states that an external system is active and if a message is being routed it, should be routed to the external topic if the properties below also meet the conditions.

ExternalSystemRouteToToLoc=False – This states that if an external ID is in the to_loc of a message (such as a transfer) then the message should be routed to the external application topic.

ExternalSystemRouteToFromLoc=False – This states that if the external ID is in the from_loc of a message (such as a transfer) then the message should be routed to the external application topic.

Note: See the Oracle Retail Integration Bus Operations Guide for more information on the values for the rib.properties file.

Step 7: Create/Modify Startup Scripts

The final installation step is to create RIB startup scripts for UNIX systems. These scripts start up the SeeBeyond e*Gate registry and the control broker for the <RIB_SCHEMA> Schema. Default scripts are provided for starting the registry service and the control broker in the $EHOME directory. Refer to the “start_egate” and a “start_cb” scripts.

The commands that implement this functionality are stcregd and stccb. When executed, they run as daemons. Depending on logging and other parameters, they may log items to their stderr or stdout files. These commands are detailed further in the following manuals:

- Oracle Retail Integration Bus Operations Guide
- SeeBeyond e*Gate Integrator User’s Guide
- SeeBeyond e*Gate Integrator System Administration and Operations Guide
Step 8: Update the hibernate.cfg.xml

In order for hibernate to validate the RIB error hospital database connection, the hibernate.cfg.xml file located in the $EHOME/client/classes directory must have the following sections updated with valid database settings (database user, database password) for the appropriate error hospital database connection.

```xml
<property name="hibernate.connection.username">seebeyond2</property>
<property name="hibernate.connection.password">rib104</property>
```

One needs remove the following line from existing hibernate.cfg.xml.

```xml
<property name="hibernate.connection.url">jdbc:oracle:thin:@server:port:database</property>
```

One needs change the "show_sql" property from true to false in the existing hibernate.cfg.xml.

We strongly recommend run the UNIX diff between the existing hibernate.cfg.xml and $RETEK_INSTALL_DIR/RIB<version>/Retek_Sbyn/conf/hibernate.cfg.xml to verify and merge exact differences.

Step 9: Copy classes12.zip file to Server and Client Classpath to Utilize jdbc 10.2.0.1 Driver

RIB provides the 10.2.0.1 driver to overwrite SeeBeyond’s default driver.

The file exists in $RETEK_INSTALL_DIR/<RIB_SCHEMA>/ThirdPartyLib/oracle/classes12.zip. It should be copied to $EHOME/server/registry/repository/default/ThirdParty/oracle/classes/ and $EHOME/client/ThirdParty/oracle/classes/

Step 10: Configure component.xml

The component.xml file located in the $EHOME/client/classes directory is for mapping the Eway collaboration implementation to RIB Oracle package for each RIB family. Verify that the component defined in the component.xml matches with your PL/SQL application settings.
Step 11: Starting the RIB Components for a New/Modified Schema

start_rib is intended for use in starting the e*Way components of a new schema or when new e*Way components have been added to an existing schema. The RIB architecture requires that the first time an e*Way is started in a schema, it is started in a particular order (subscriber, TAFR, publisher). The start_rib script uses an input file called Egate.txt to identify an e*Way as a subscriber, a TAFR, or a publisher. Egate.txt contains a listing of all of the e*Way components packaged with the RIB. start_rib is executed with one or several of the parameters listed within its usage. To view the start_rib usage, change UNIX directories to $EHOME and run start_rib with no parameters.

It is necessary that start_rib is run to start the e*Way components immediately after installation and configuration of the RIB components. The following command should be executed while the registry and control broker are running: start_rib JMS SUB TAFR. After these processes have successfully started (check to be certain connection points were accurately set and DB connection was successful), if the client is integrating with SIM, start the SIM services following SIM instruction. Run start_rib once more using the following command: start_rib PUB HOSP. All of the e*Way process listed in the Egate.txt file should now be running. Because all Egate.txt contains a list of all eway components, it is possible that certain eways will show a failure on startup due to the fact they are not part of this particular installation (that is, if RWMS is not installed, then all ewXXXFromRDMWH1 eways will fail on startup). The Schema Manager can be used to verify all eways were started by using their color coding; blue for up, red for down.

If desired, new e*Way components may be started using the start_rib command by adding the new components to the start_rib input file – Egate.txt. This file can be located in the $EHOME directory. Unused e*Way components may also be removed from Egate.txt.
Database Triggers and Oracle Dependencies

Database Triggers

Once the RIB has been installed and configured, the publishing applications need to be told to begin to publish data. There are multiple ways to initiate the publishing process. Each Retail product’s operation guide contains this information.

Oracle Dependencies

In order for the RIB to function correctly, you must install Oracle’s XML Developer’s Kit for PL/SQL on your database server. This can be downloaded from Oracle Technology Network. The version of XML Developer’s Kit for PL/SQL must be dated 3/28/2002 or later. There is a bug in prior version that prevents the RIB from working correctly.

Verify RIB Error Hospital Database Tables

One feature of the RIB is the Error Hospital subsystem used to store and retry messages that have processing problems by a subscribing application. This facility allows for non-dependent messages to continue to be processed by the application until the failure has been resolved and the message successfully consumed.

These tables were created during the database portion of the RWMS, SIM, RPM, AIP, or RMS installations. The DDL to create these tables can be found on these products’ installation CDs.

For each RIB Error Hospital, verify that the four hospital tables exist in each application’s database schema and that they have the correct columns. The four tables necessary are:
**RIB_MESSAGE**

```
DROP TABLE RIB_MESSAGE CASCADE CONSTRAINTS;
CREATE TABLE RIB_MESSAGE (
    MESSAGE_NUM NUMBER(8) NOT NULL,
    LOCATION VARCHAR2(60) NOT NULL,
    FAMILY VARCHAR2(25) NOT NULL,
    TYPE VARCHAR2(30) NOT NULL,
    ID VARCHAR2(255),
    RIB_MESSAGE_ID VARCHAR2(255),
    PUBLISH_TIME DATE,
    IN_QUEUE VARCHAR2(1) NOT NULL,
    MESSAGE_DATA CLOB NOT NULL,
    ATTEMPT_COUNT NUMBER(4) NOT NULL,
    MAX_ATTEMPTS NUMBER(4) NOT NULL,
    NEXT_ATTEMPT_TIME DATE,
    DELETE_PENDING VARCHAR2(1) NOT NULL,
    TOPIC_NAME VARCHAR2(255),
    THREAD_VALUE NUMBER(22,8),
    JMS_QUEUE_ID VARCHAR2(255),
    CUSTOM_FLAG VARCHAR2(1) DEFAULT 'F' NOT NULL,
    CUSTOM_DATA CLOB,
    REASON_CODE VARCHAR2(10) NOT NULL,
    CONSTRAINT PK_RIB_MESSAGE
    PRIMARY KEY (MESSAGE_NUM));
```

**Note:** The field MESSAGE_NUM in the table RIB_MESSAGE has an associated script to create the sequence. It is required to run this script – rib_message_seq.sql. This script can be found in the `<RETEK_INSTALL_DIR>/RIBfor<APP><version>/XML_U tilities` directory.

**RIB_MESSAGE_FAILURE**

```
DROP TABLE RIB_MESSAGE_FAILURE CASCADE CONSTRAINTS;
CREATE TABLE RIB_MESSAGE_FAILURE (
    MESSAGE_NUM NUMBER(8) NOT NULL,
    SEQ_NUMBER NUMBER(6) NOT NULL,
    TIME DATE NOT NULL,
    LOCATION VARCHAR2(60) NOT NULL,
    DESCRIPTION VARCHAR2(4000) NOT NULL,
    ERROR_TYPE VARCHAR2(2) DEFAULT 'SY' NOT NULL,
    ERROR_CODE VARCHAR2(25) NULL,
    CONSTRAINT PK_RIB_MESSAGE_FAILURE
    PRIMARY KEY (MESSAGE_NUM, SEQ_NUMBER));
```
**RIB_MESSAGE_ROUTING_INFO**

```sql
DROP TABLE RIB_MESSAGE_ROUTING_INFO CASCADE CONSTRAINTS;

CREATE TABLE RIB_MESSAGE_ROUTING_INFO (  
    MESSAGE_NUM NUMBER(8) NOT NULL,
    SEQ_NUMBER NUMBER(6) NOT NULL,
    NAME VARCHAR2(25) NOT NULL,
    VALUE VARCHAR2(25) NOT NULL,
    DETAIL1_NAME VARCHAR2(25),
    DETAIL1_VALUE VARCHAR2(300),
    DETAIL2_NAME VARCHAR2(25),
    DETAIL2_VALUE VARCHAR2(300),
    CONSTRAINT PK_RIB_MESSAGE_ROUTING_INFO
    PRIMARY KEY (MESSAGE_NUM, SEQ_NUMBER));
```

**RIB_MESSAGE_HOSPITAL_REF**

```sql
DROP TABLE RIB_MESSAGE_HOSPITAL_REF CASCADE CONSTRAINTS;

CREATE TABLE RIB_MESSAGE_HOSPITAL_REF (  
    MESSAGE_NUM NUMBER(8) NOT NULL,
    SEQ_NUMBER NUMBER(6) NOT NULL,
    HOSPITAL_REF VARCHAR2(8) NOT NULL,
    LOCATION VARCHAR2(60),
    MESSAGE_FAMILY VARCHAR2(25),
    NEW_REASON_CODE VARCHAR2(10),
    OLD_REASON_CODE VARCHAR2(10),
    CONSTRAINT PK_RIB_MESSAGE_HOSPITAL_REF
    PRIMARY KEY (MESSAGE_NUM, SEQ_NUMBER));
```

**Note:** If these tables need to be created outside of a Retail Application’s installation, scripts called ‘all_rib_tables.sql, all_rib_table_values.sql’ can be found in the `<RETEK_INSTALL_DIR>/<RIB_SCHEMA>/XML` directory.

If your database already has these tables in place from a prior RIB installation, the rib_message table may need to be updated with new table changes.
RIB_DOCTYPE Table and DTD Files

The RIB_DOCTYPE table should also have been created and populated by the Retail Application installation(s) (for example, RMS). The integration with Java EE applications now requires that the DTD files themselves be network accessible. To accomplish this, all of the DTD files in `<RETEK_INSTALL_DIR>/RIB<Version>/XML/DTDs/dtd.war` should be deployed to a web server running at the installation site. Once this is done, and the URL to the DTDs is known, the value in the DOC_TYPE_URL column will need to reflect this location.

The rib_doctypes_rms.dat for RMS or rib_doctypes_rdm.dat for RWMS files which are found in the `<RETEK_INSTALL_DIR>/RIB<Version>/XML` directory are the data files that RMS or RDM use to populate the table with DOC_TYPE_URL data by using Oracle SQL Loader with rib_doctypes_rms.ctl or rib_doctypes_rdm.ctl. The URLs can be globally replaced with the new one.

**Note:** If custom RMS, a custom version of this script will exist. It should be applied instead.
RIB Hospital Administration Tool Installation

The RIB Hospital Administration Tool (RIHA) is a java executable/application provided to perform RIB administration functions in the Error Hospital database.

Prerequisites

The minimum and preferred Java Runtime Engine (JRE) version to use with RIHA is 1.4.2_10. This JRE must be installed on the workstation/PC where RIHA will be installed prior to running the configuration script.

Install RIB Administration Tool

These instructions assume RIHA will be installed in a PC running a Windows operating system. To install RIHA in a Unix environment, follow these instructions by using the proper file separator character and script extensions.

1. Copy the riha1201_eng.ga.tar file to the location where RIHA will be installed (e.g.: C:\RIB_Tools.)
2. Decompress the tar file with an archive utility (e.g.: WinZip). This will extract all RIHA files into a directory named RIHA1201 (e.g.: C:\RIB_Tools\RIHA1201)
3. Downloading hibernate.
   RIHA uses Open Source O-R mapping tool called hibernate (http://www.hibernate.org/). Due to licensing limitation RIHA cannot ship hibernate along with its package, so hibernate 2.1.8 has to be download by the person installing RIHA. RIHA has been certified with only hibernate 2.1.8, do not download any other hibernate version.
   Download hibernate (hibernate-2.1.8.zip) and extract the hibernate2.jar file from inside the zip file. Copy the hibernate2.jar to RIHA<Version>/external-lib directory. 
   cp <your download dir>/hibernate2.jar <your riha dir>/RIHA1201/external-lib
4. Go to the RIHA1201\bin directory and execute the riha-config.bat file. This script will drive the rest of the installation and configuration process:

a. Create a user login. RIHA provides users with access to information contained in the applications error hospital database. Users can view and modify this data and control the variables that make possible the feedback of messages into the system. Because of this, limiting the accessibility to this tool is imperative. RIHA supports the creation of user logins to guarantee that only designated users can execute this tool. The following prompts ask the user to enter the credentials to create a new user login for RIHA. More than one user can be created.

Starting RIHA configuration utility...

Create a new user login:

Enter User Id: jdoe

Enter User First Name: John

Enter User Last Name: Doe

Enter password: <password does not show>

Verify password: <password does not show>

User jdoe created.
Do you want to create another user? (y/n): n
b. Next, the user is presented with a list of standard Retail applications to choose from and configure the database connection where the Error Hospital of each application resides. It also prompts the user for the server hosting the DTD files for the RIB messages. Once all this information is entered, the user is returned back to the Retail applications list to either configure a new connection or complete this process.

Creating hibernate configuration files...

You can access multiple Error Hospital databases by setting multiple configuration files (even for the same product). Select [D]one when finished.

Please choose a product for configuring database information:

1) AIP  - Retail Advanced Inventory Planning
2) SIM  - Retail Store Inventory Management
3) RCOM - Retail Customer Order Management
4) RMS  - Retail Merchandising System
5) RPM  - Retail Price Management
6) RWMS - Retail Warehouse Management System

([1], [2], [3], [4], [5], [6], [D]one): 4

Enter database host name (e.g.: mspdev05.oracle.com): mspdev05
Enter database port (e.g.: 1521): 1521
Enter database instance (e.g.: DEV): DEV
Enter user name: schema_owner
Enter password: <password does not show>
Verify password: <password does not show>
Enter name of server hosting dtd files (e.g.: mspdev05.oracle.com): mspdev05
Enter port number of server hosting dtd files (e.g.: 8080): 8080

Please choose a product for configuring database information:

1) AIP  - Retail Advanced Inventory Planning
2) SIM  - Retail Store Inventory Management
3) RCOM - Retail Customer Order Management
4) RMS  - Retail Merchandising System
5) RPM  - Retail Price Management
6) RWMS - Retail Warehouse Management System

([1], [2], [3], [4], [5], [6], [D]one): d
c. Finally the user is prompted to enter the full path for the web browser to use when displaying the online help.

Please enter the full path for your default browser.
This will be used when displaying the online help (C:\Program Files\Internet Explorer\iexplore.exe): <hitting Enter will accept the value in parenthesis>

RIHA configuration completed.

RIHA is configured and ready. Execute the riha.bat script to start.
RIB Java EE Integration for OAS

This chapter describes the process of installing and configuring rib-<app> enterprise application in the Oracle Application Server (OAS). For every Java EE retail application (for example, RPM, AIP, SIM) that uses RIB as enterprise integration tool there is a RIB connector Java EE application that connects the retail application to the messaging system. This connector application is referred in chapter as rib-<app>, where “app” stands for any of the Retail applications (for example, rpm, aip, sim) supported by RIB.

The naming conventions used in the chapter are as follows:
- “Retail application” means one of the retail business Java EE application viz. rpm, aip, sim. It is referred to generically as “<app>”.
- The RIB connector application for a retail application is generically referred to as “rib-<app>”.

RIB-<App> Installation

The following section describes the process of configuring OAS for RIB Java EE application and installing RIB on the configured OAS. The person installing must follow the steps below in the order it appears in this chapter. If for some reason there is a problem in one step do not proceed to the next step without first fixing the problem.

Prerequisite

Before installing rib-<app> make sure that the following software components are properly installed and configured.
- Java 1.4.2.10 JDK (RPM, AIP)
- Java 1.5 JDK is required for SIM
- 10g (10.2.0.1) Database
- RIB tables are installed in the retail application database schema
- Egate 5.0.5
- OAS 10.1.3
- Retail application <app> that this rib-<app> will connect to is already installed in OAS. It is required that <app> and rib-<app> applications do not share an oc4j instance and are installed in its own respective oc4j instance.

Note: For instructions on how to change the JDK for an OC4J instance, see the Specifying the JDK in a Managed Configuration section of the OC4J Runtime Configuration chapter of the Oracle Containers for J2EE Configuration and Administration Guide.
Information Gathering about Retail Application (<app>, not rib-<app>)

rib-<app> Java EE application communicates with the Retail Java EE application (<app>) at runtime to send/receive messages. The rib-<app> needs the following connectivity information about the Retail application <app> it is communicating with.

- Hostname where opmn server is running for the application server where <app> ear file is installed. From here on this host name will be referred as RETAIL_APP_OPMN_HOST_NAME in this chapter.
- opmn “request” port for OAS where <app> ear is installed. The port can be found in $ORACLE_HOME/opmn/conf/opmn.xml file. From here on this port number is referred to as RETAIL_APP_OPMN_REQUEST_PORT in this chapter.
- oc4j instance name where <app> ear is installed. From here on this oc4j instance is referred to as RETAIL_APP_OC4J_INSTANCE in this chapter.
- <app> application names as seen through AS control. From here on the Retail application is referred to as RETAIL_APP_NAME in this chapter.
- RIB Error Hospital Database JDBC connection URL for rib-<app>. Usually the RIB Error Hospital tables are installed in the same database schema as the retail application <app>.
- RIB Error Hospital Database user name for rib-<app>.
- RIB Error Hospital Database password for rib-<app>.

Information Gathering About rib-<app> Application.

- opmn “request” port for OAS where rib-<app> ear will be installed. The port can be found in $ORACLE_HOME/opmn/conf/opmn.xml file. From here on this port number is referred to as RIB_APP_OPMN_REQUEST_PORT in this chapter.
- OAS instance name under which oc4j instance will be created for rib-<app>. The OAS instance name can be found in $ORACLE_HOME/opmn/conf/opmn.xml file. From here on OAS instance name is referred to as RIB_APP_OAS_INSTANCE_NAME in this chapter.
- Egate JMS server host name. From here on it is referred to as JMS_HOST in this chapter.
- Egate JMS server port number. From here on it is referred to as JMS_PORT in this chapter.
Preparing OAS

This section describes the process of preparing OAS to install rib-<app> Java EE application. rib-<app> Java EE application runs in its own oc4j server instance called “rib-<app>-oc4j-instance”. The “rib-<app>-oc4j-instance” must be configured to use only ONE operating system process. Use the following steps to create a new oc4j instance for rib-<app> and configure it to rib’s requirement.

1. Log in to the machine where OAS was installed with the operating system user that was used to install the Oracle Application Server (OAS).
2. Make sure the OAS required environment variables are set. Read the OAS documentation for environment variables information.
3. Create rib-<app>-oc4j-instance by executing the following command.

   $ORACLE_HOME/bin/createinstance -instanceName rib-<app>-oc4j-instance

   replace <app> with the actual value of the retail application name. The only acceptable values for the <app> are either “rpm”, “aip” or “sim” (lower case).

   Type in “oc4jadmin” for the oc4j instance password when prompted for. This password can be anything you like as long as it follows OAS standards.

4. Configure rib-rpm-oc4j-instance
   a. Edit $ORACLE_HOME/opmn/conf/opmn.xml file and add the following properties to the rib-<app>-oc4j-instance definition.

   **Note:** All the properties below must be applied only to rib-<app>-oc4j-instance definition.

   b. Change the default (1.5) JVM to a 1.4.2 JVM.
      - Add the following element under the start-parameters category element.

      <data id="java-bin" value="<your path to java home>/bin/java"/>

      Put the full path to the location where your java 1.4.2 executable is located.

   c. Update the java-options section to add the following JVM properties.
      - Specify the JVM’s min and max heap size.

      -Xms500M -Xmx900M

      - Switch off JMX security.

      -Doc4j.jmx.security.proxy.off=true

      - Add the following JVM property for OAS on AIX only.

      -Dibm.cl.eagerresolution
d. Add `<oc4j-options>` to add the following `oc4j` properties.
   
   - Specify `-userThreads` to `oc4j-options` element.

   ```xml
   <data id="oc4j-options" value="-userThreads" />
   ```

   For E.g. after updating the `opmn.xml` for `rib-<app>-oc4j-instance` definition the xml section should look something like the following.

   ```xml
   <process-type id="rib-rpm-oc4j-instance" module-id="OC4J" status="enabled">
      <module-data>
         <category id="start-parameters">
            <data id="java-bin" value="/vol.rtk/java/solaris/java1.4.2_10/bin/java"/>
            <data id="java-options" value="-Doc4j.jmx.security.proxy.off=true -Xms500M -Xmx900M -Dibm.cl.eagerresolution -Djava.security.policy=\$ORACLE_HOME/j2ee/rib-rpm-oc4j-instance/config/java2.policy -Djava.awt.headless=true -Dhttp.webdir.enable=false"/>
            <data id="oc4j-options" value="-userThreads"/>
         </category>
      </module-data>
   </process-type>
   ```

Verification of `rib-<app>-oc4j-instance` Configuration

1. Make sure that `numprocs` attribute for the `process-set` element for `rib-<app>-oc4j-instance` is set to 1.

   For example, `<process-set id="default_group" numprocs="1"/>`

2. Reload the new configuration by executing the following command.

   ```bash
   $ORACLE_HOME/opmn/bin/opmnctl reload
   ```

   If there is any error during reloading, verification has failed, check your configuration and retry.

3. Start the new `rib-<app>-oc4j-instance` by executing the following command.

   ```bash
   $ORACLE_HOME/opmn/bin/opmnctl startproc process-type="rib-<app>-oc4j-instance"
   ```

   replace `<app>` with the actual value of the retail application name. The only acceptable values for the `<app>` are either “rpm”, “aip” or “sim”.

   If there is any error during startup, verification has failed, check your configuration and retry.
Starting up the Dependent Software’s that rib-<app> Requires

Make sure that the following software is properly configured and is running before proceeding to install rib-<app> section.

- Database
- Egate JMS
- Newly created rib-<app>-oc4j-instance

Installing rib-<app> on Newly Created rib-<app>-oc4j-instance

The following steps describe the process of installing rib-<app> in the pre configured rib-<app>-oc4j-instance.

1. Create a “rib_install_stage” directory on the server where OAS is already installed and where rib-<app> will be installed. From here on this directory will be referred as $RIB_INSTALL_HOME. This directory will contain the rib-<app> code and installation/management scripts.

   mkdir <your dir>/rib_install_stage

2. FTP/COPY the ribpak<version>for<app_version>_eng_ga.tar file to the $RIB_INSTALL_HOME directory.

   cp ribpak<version>for<app_version>_eng_ga.tar /<your dir>/rib_install_stage

3. Untar the contents of this file into this directory.

   - Change directory to $RIB_INSTALL_HOME
     cd <your dir>/rib_install_stage

   - Extract the tar ball.
     tar xvf ribpak<version>for<app_version>_eng_ga.tar

The following procedure creates a directory structure that will be used during the installation and maintenance process.

Download hibernate.

RIB uses Open Source O-R mapping tool called hibernate (http://www.hibernate.org/). Due to licensing limitation RIB cannot ship hibernate along with its package, so hibernate 2.1.8 has to be download by the person installing rib-<app>. RIB has been certified with only hibernate 2.1.8, do not download any other hibernate version.

1. Download hibernate (hibernate-2.1.8.zip) and extract the hibernate2.jar file from inside the zip file. Copy the hibernate2.jar to $RIB_INSTALL_HOME/RIBfor<App><Version>/rib/external-lib directory.

   cp <your dir>/hibernate2.jar $RIB_INSTALL_HOME/RIBfor<App><Version>/rib/external-lib

   Each property in rib-config.properties is documented inline in the file. There are two ways to run oc4j, either standalone or in OAS managed state. RIB is only supported in the OAS “manage” setting, so use the “In Managed OC4J” specific values while editing rib-config.properties. Edit each property to match your environment settings.

Following are the properties from the rib-config.properties file that need modification.

```properties
# Oc4j instance home directory. The home directory of the Oc4j AppServer instance where the rib-<app> app will be installed. In standalone oc4j the directory will always be $ORACLE_HOME/j2ee/home. In OAS it must be $ORACLE_HOME/j2ee/<oc4j instance name>. Following out naming convention <oc4j instance name> will always follow the format rib-<app>-oc4j-instance.
#
# e.g.
#
# In Standalone OC4J: OC4J_HOME=/files0/wsperf/Oc4j/j2ee/home
# In Managed OC4J : OC4J_HOME=/files0/wsperf/Oc4j/j2ee/<oc4j instance name>
#
#OC4J_HOME=/home/prantor/development/oc4j_10.1.3/j2ee/rib-rpm-oc4j-instance
#
# Java Home. The directory where java is installed.
#
# e.g.
#
# In Standalone OC4J: JAVA_HOME=/files0/wsperf/javal.4
# In Managed OC4J(OAS): JAVA_HOME=/files0/wsperf/javal.4
#
#JAVA_HOME=/usr/java/jdk1.4.2
#
# Deploying to Managed OC4J(OAS) or standalone OC4J?
# This property must always be true in a production install.
#
# e.g.
#
# In Standalone OC4J: is_managed_oc4j=false
# In Managed OC4J(OAS): is_managed_oc4j=true
#
#is_managed_oc4j=true
#
# Host name. The host name where the rib-<app> app will be installed.
# In Managed OC4J(OAS) host_name is the name of the machine where OAS cluster is running.
#
# e.g.
#
# In Standalone OC4J: host_name=mspdev05
# In Managed OC4J(OAS): host_name=mspdev05
#
#host_name=localhost
```

---

48 Oracle Retail Integration Bus
# Management port number of the server where rib-<app> app will be installed.
# In managed OC4J(OAS) the port number is the
# opmn request port number. Use the value of $RIB_APP_OPMN_REQUEST_PORT
# identified above in the information gathering section. The opmn_request_port is
# not
# applicable for standalone OC4J.
#
# e.g.
#
# In Standalone OC4J: opmn_request_port=
# In Managed OC4J(OAS): opmn_request_port=6004
#
# opmn_request_port=6004

# Oc4j ormi port. The ormi port of the Oc4j instance where the
# rib-<app> app will be installed. In Managed OC4J(OAS)
# oc4j_ormi_port is not applicable as it is dynamically
# assigned in the cluster.
#
# e.g.
#
# In Standard OC4J: oc4j_ormi_port=23791
# In Managed OC4J(OAS): oc4j_ormi_port=
#
# oc4j_ormi_port=23791

# OAS insatnce name. The OAS instance name.
# In standalone OC4J mode oas_instance_name is not applicable.
# Use the value of $RIB_APP_OAS_INSTANCE_NAME
# identified above in the information gathering section.
#
# e.g.
#
# In Standalone OC4J: oas_instance_name=
# In Managed OC4J(OAS): oas_instance_name=OAS.mspdev05
#
# oas_instance_name=OAS.mspdev05

# O4J instance name where the rib-<app> app will be installed. The value should be
# the new oc4j instance we created for rib-<app> application. The oc4j instance
# will be of the format rib-<app>-oc4j-instance.
#
# e.g.
#
# In Standalone OC4J: oc4j_instance_name=home
# In Managed OC4J(OAS): oc4j_instance_name=<rib app oc4j instance name>
#
# oc4j_instance_name=rib-rpm-oc4j-instance
RIB-<App> Installation

# Oc4j admin user. The oc4j admin user of the Oc4j instance where the rib-<app> app will be installed.
# e.g.
# In Standalone OC4J: oc4j_admin_user=oc4jadmin
# In Managed OC4J(OAS): oc4j_admin_user=oc4jadmin
# oc4j_admin_user=oc4jadmin

# Oc4j admin password. The oc4j admin password of the Oc4j instance where the rib-<app> app will be installed. This should be the same password used during creating rib-<app>-oc4j-instance.
# e.g.
# In Standalone OC4J: oc4j_admin_password=oc4jadmin
# In Managed OC4J(OAS): oc4j_admin_password=oc4jadmin
# oc4j_admin_password=oc4jadmin

# The JMS Host name for the SeeBeyond Generic JMS Provider.
# e.g.
# In Standalone OC4J: JMS_HOST=mspdev05
# In Managed OC4J(OAS): JMS_HOST=mspdev05
# JMS_HOST=msppc004137

# The JMS Port number for the SeeBeyond Generic JMS Provider.
# e.g.
# In Standalone OC4J: JMS_PORT=29053
# In Managed OC4J(OAS): JMS_PORT=29053
# JMS_PORT=24053

# JDBC URL to the rib-<app> database. This should point to the database schema where the RIB Error Hospital tables are installed.
# e.g.
# In Standalone OC4J: ribDbJdbcUrl=jdbc:oracle:thin:@mspdev15:1524:dste_9i
# In Managed OC4J(OAS): ribDbJdbcUrl=jdbc:oracle:thin:@mspdev15:1524:dste_9i
#
### ribDbJdbcUrl
```
jdbc:oracle:thin:@mspdev02:1522:ribdt01
```

### ribDbUserId
```
ribDbUserId=ribrgr10
```

### ribDbPassword
```
ribDbPassword=retek
```
As the above comments specify DO NOT edit anything below the line in the rib-config.properties file.

3. Edit

$RIB_INSTALL_HOME/RIBfor<App><Version>/rib/properties/rib.properties

Edit the following properties to match your environment.

log.default.file_path

The value of this property must be the full path to where rib-<app> will write logs incase of a fatal error during message processing. This must always point to $ORACLE_HOME/j2ee/rib-<app>-oc4j-instance/log/rib-<app>

For example:

<before>
log.default.file_path=log/rib-rpm
</before>

<after>
log.default.file_path=/u00/OAS/j2ee/rib-rpm-oc4j-instance/log/rib-rpm
</after>

- dtd_url.default

  The value of this property must point to the location URL where the dtd.war is deployed.

  **Note:** Refer to the dtd installation instruction for more information.

  dtd_url.default=http://mspdev14:8100/dtd/

  E.g.

  <before>
dtd_url.default=http://mspdev14:8100/dtd/
</before>

  <after>
dtd_url.default=http://<your host name>:<your web port name>/dtd/
</after>

- rib-<app> can send email alert messages when there is a fatal error in the system. Configure the following mail.smtp.* properties to enable the email alerting system.

  mail.smtp.host – Your SMTP server host.
  Eg. mail.retek.int

  mail.smtp.port – Your SMTP server port.
  Eg. 25

  mail.smtp.from – The email address that will be the source of the alert.
  Eg. name@company.com

  **Note:** Some SMTP server require this to be a valid email address known to the SMTP server.

  mail.smtp.to.list – The list of email address separated by commas (,) to who alert emails will be sent.
For example, x@company.com, y@company.com

- **war.http.port**
  This property should be set to the value of the HTTP port for the rib-<app>-oc4j-instance.

- **rib.jms.hostname**
  This is the hostname where the eGate JMS server is running.

- **rib.jms.port**
  This is the JMS port number of the eGate JMS server.


4. Edit $RIB_INSTALL_HOME/RIBfor<App><Version>/rib/properties/retek/jndi_providers.xml file to update the connectivity information. Identify the provider section where app="<app>" under the OAS section and update the JNDI url. Remember the only acceptable values for the <app> are either “rpm”, “aip” or “sim”.

   The url is of the following format
   
   url="opmn:ormi://<RETAIL_APP_OPMN_HOST_NAME>:<RETAIL_APP_OPMN_REQUEST_PORT>:<RETAIL_APP_OC4J_INSTANCE>/<RETAIL_APP_NAME>"

   Replace the variables with your environment specific values that were collected in the information gathering section (above).

   For example, the updated section should look like.

   ```xml
   <provider app="rpm" url="opmn:ormi://mspdev40.us.oracle.com:6007:rpm-oc4j-instance/rpm" factory="oracle.j2ee.rmi.RMIInitialContextFactory" />
   ```

   a. Start the rib-<app> installation by executing the following command.

   ```bash
   cd $RIB_INSTALL_HOME/RIBfor<App><Version>/rib
   ./ribinstall.sh
   ```

   **Note:** The script does not redirect standard out and standard error to a file. If you want to save a history of the install redirect it using the shells redirection mechanism.
Configuring the JNDI Information in the Retail Application <app> (Not rib-<app>)

The retail application (<app>) communicates with rib-<app>’s PublisherEjb to publish messages to JMS. The retail application needs to know how to connect/lookup the PublisherEjb through JNDI. Validate that the jndi_providers.xml in the retail applications side (<app>, not rib-<app>) has the correct rib-<app> ear’s JNDI information.

The location of the jndi_providers.xml in the retail application (<app>) side is not managed/controlled by rib-<app> so check the retail application’s documentation about the location of this file.

Locate the jndi_providers.xml file in the retail application and update JNDI url for the provider where app="rib-<app>". The url must be of the following format.

url="opmn:ormi://<RIB_APP_OPMN_HOST_NAME>
</RIB_APP_OPMN_REQUEST_PORT>:<RIB_APP_OC4J_INSTANCE>/<RIB_APP_NAME>"

Restart the retail application.

Verification of rib-<app> install.

- Login through the AS Control and check the rib-<app> application status. It should say that the application (rib-<app>) is up.
- Check the error hospital retry log files in rib-<app>-oc4j-instance/log/rib-<app>/*.log . There should not be any error messages.
- Login to the Egate JMS Administrator. The topic names that rib-<app> subscribes to should get created and should be visible through the JMS Administrator. The subscriber names should also show up as rib_<app>_<interface name>.
**RIB-<App> Maintenance**

**RIB Deployment Command-line Utility**

ribadmin
The command-line utility can be used to perform basic rib-<app> maintenance within the application server context.

**Commands**
The following commands are available for this utility:

- **stop**: Stops the oc4j instance where rib-<app> is installed.
- **start**: Starts the oc4j instance where rib-<app> is installed.
- **dellogs**: Deletes *.log for the rib-<app> application.
- **bounce**: Stops, delete logs and starts the oc4j instance where rib-<app> is installed.
- **genlog4j**: Generates the log4j.xml for rib-<app> application logs.
- **install**: Installs the rib-<app>.ear application on the rib-<app>-oc4j-instance.
- **uninstall**: Uninstalls the rib-<app>.ear application on the rib-<app>-oc4j-instance.
- **reinstall**: Reinstalls the rib-<app>.ear application on the rib-<app>-oc4j-instance.
- **binding**: Generates a new JMS JNDI .bindings file for eGate JMS provider.
  This command can be used to point an already installed rib-<app>.ear to a new eGate JMS server. To point to a new eGate JMS server use the following procedure.
  Edit the JMS_HOST and JMS_PORT property in the rib-config.properties file.
  Run ./ribadmin.sh binding
  Run ./ribadmin.sh bounce
- **props**: Displays the values for all the properties in the script.

**Usage**
Run this utility by using the following command:
ribadmin.sh <command>

**Configuration**
The ribadmin.sh script uses the configuration setting that is defined in rib-config.properties file.
The manual steps outlined in this chapter are not necessary if you are installing RIB with a new install of SIM 11.1 or later. SIM 11.1 includes an installer that installs and configures SIM and RIBforSIM at the same time. See the SIM installation documentation for details. An exception to this is step 8, which still needs to be run manually in the SIM database schema in all situations.

If you are installing RIBforSIM to a pre-existing SIM server installation you must follow the instructions in this section.

This chapter will review the steps required for integrating the RIB with SIM.

**Manual Steps**

Log into the SIM application server as the user who performed the SIM install.

1. Create a temporary install directory for integrating RIB with SIM. This temporary directory will be known as `<SIM_TMPINSTALL_DIR>` for the remainder of this section:
   
   ```
   > mkdir INSTALL
   ```

2. Copy the file `ribpak<version>forsim<sim_version>.tar` to `<SIM_TMPINSTALL_DIR>`.

3. Once you have copied the file, extract the contents.
   
   ```
   > tar xvf 'ribpak<version>forsim<sim_version>.tar'
   ```

4. Ensure all files located in `<SIM_TMPINSTALL_DIR>` have the correct permissions by issuing the following command in `<SIM_TMPINSTALL_DIR>`:
   
   ```
   > chmod -R 755 *
   ```

5. Edit the file `sim_profile`. This file is located at `<SIM_TMPINSTALL_DIR>/RIB<version>`. Make sure the settings for the following variables are correct for your environment.
   
   - SIM_TMPINSTALL_DIR - Same directory created in step 1 above
   - SIM_INSTALL_DIR - The directory where SIM was installed.

   **Note:** The install script expects SIM_TMPINSTALL_DIR to include the temporary install directory created in step 1 above (eg: /files0/jadmin/INSTALL) and SIM_INSTALL_DIR to include the full directory path leading up to server<platform> (for example: /files0/jadmin/sim)

6. Shut down the SIM application server. In `<SIM_TMPINSTALL_DIR>/RIB<version>` run `sim_profile`:
   
   ```
   > . sim_profile
   ```

   
   ```
   > ./installisoconfig (When asked if you wish to proceed, type YES and hit enter)
   ```
8. Steps for the RIB Error Hospital:
   a. Check the SIM schema for the existence of the following RIB error hospital tables:
      RIB_MESSAGE, RIB_MESSAGE_FAILURE, and RIB_MESSAGE_ROUTING_INFO.
   b. If the RIB error hospital tables do not exist, ensure that the following tablespaces
      exist: LOB_DATA, RETEK_DATA, and INDEX_DATA

      **Note:** Do not run the following sql scripts if the rib error
      hospital tables above already exist. These scripts will drop
      and re-create the tables, thus deleting all data in the tables.

   c. Run all_rib_tables.sql as the SIM schema to create the required RIB tables. This
      script is located at <SIM_TMPINSTALL_DIR>/RIB<version>/XML.

   d. Run rib_message_seq.sql as the SIM schema to create the sequence number for
      the table RIB_MESSAGE. This script is located
      at:<SIM_TMPINSTALL_DIR>/RIB<version>/XML.

      **Note:** Since the database scripts drop and re-create the
      tables, ignore any errors resulting from trying to drop
      objects that don’t exist. These errors are expected.

9. Edit the file ribmessaging.cfg. This file is located at
   <SIM_INSTALL_DIR>/server<platform>/retek/sim/files/prod/config. Make sure
   the setting for the following variable is correct for your environment. This value
   must match exactly values listed in the SeeBeyond configuration file cpJMS.cfg of
   HostName+Port Number (IE cpJMS.cfg-HostName: mspdev05.retek.int, cpJMS.cfg-
   Port Number: 37053, ribmessaging.cfg-BROKER: mspdev05.retek.int:37053).
   
   - BROKER=<EGATE_SERVER_NAME>:<JMS_PORT>

   (eg: BROKER= mspdev05.retek.int:37053)

10. Edit the file log4j.xml. This file is located at
    <SIM_INSTALL_DIR>/server<platform>/retek/sim/files/prod/config. Replace all
    occurrences of <SIM_INSTALL_DIR> in this file with the full directory path leading
    up to the /server<platform> directory.

    - Example: <param name="File"
      value="/files0/jadmin/sim11.0/server<platform>/retek/sim/log/itemsmessagingcomponent.log"/>

    (<param name="File"
      value="/files0/jadmin/sim11.0/server<platform>/retek/sim/log/itemsmessagingcomponent.log"/>)

    Replace all occurrences of <SERVER> and <PORT> in this file with the
    <EGATE_SERVER_NAME> and <JMS_PORT> values configured for the
    BROKER property of the ribmessaging.cfg file.
11. Edit the file rib.properties. This file is located at
<SIM_INSTALL_DIR>/server<platform>/retek/sim/files/prod/config. Replace
the occurrence of <SIM_INSTALL_DIR> in this file with the full directory path
leading up to the /server<platform> directory.

- Example:
  log.default.file_path=<SIM_INSTALL_DIR>/server<platform>/retek/sim/log/

  (log.default.file_path
   =/files0/jadmin/sim11.0/server<platform>/retek/sim/log/)

12. Edit Hibernate Properties file

Hibernate is used for Object Relational Mapping and Database interaction within the
RIB’s hospital code. Edit
<SIM_INSTALL_DIR>/server<platform>/retek/sim/files/prod/config
/hibernate.cfg.xml file to define your database information. The following
properties in this file will need to be edited before the RIBforSIM application will be
able to run:

“connection.username” – the username to connect to the SIM schema.
“connection.password” – the password to connect to the SIM schema.

---

**Note:** Make sure connection.url is not defined in
hibernate.cfg.xml file.

---
SIM Reference

The following sections are noted here for reference.

rib.properties File

The rib.properties file contains the RIB specific properties used by the RIB subscribing messaging components under SIM. These messaging components will be deployed in a SIM container, one for each subscribing API. Some of the important sections of this file are illustrated below:

```
# These are the RIB hospital properties.
hospital.attempt.max=5
hospital.attempt.delay=10
hospital.attempt.delayIncrement=10

# These are properties that are also used in the process
# of putting a message in the hospital. The difference here
# is that these properties control some of the concrete classes
# that are used in this process.
failureImpl=com.retek.rib.sbyn.FailureWrapper
routingInfoImpl=com.retek.rib.sbyn.RoutingInfoWrapper
routingInfoDetailImpl=com.retek.rib.sbyn.RoutingInfoDetailWrapper
ribMessageImpl=com.retek.rib.sbyn.RibMessageWrapper
ribMessagesImpl=com.retek.rib.sbyn.RibMessagesWrapper

# The RibBinding class that the RIB should use for XML Binding of Payload objects
# Use com.retek.rib.binding.impl.castor.CastorBindingImpl for Castor (default)
ribBindingImpl=com.retek.rib.binding.impl.castor.CastorBindingImpl

# Path where RIB and Timings log files will be written. It must end with
# a directory separator / or \
log.default.file_path=/<SIM_INSTALL_DIR>/serverUnix/retek/sim/log/

# This property is used to test Rib interaction and
# skip actual call to inject
# To skip call to inject set to value = false
subscriber.call_injector=true
```
Example of a Messaging Component Configuration File

The following is an example of the ASNIn API configuration file. These configuration files can be found in the
<SIM_INSTALL_DIR>/server<platform>/retek/sim/files/prod/config directory.

# The queue or topic name from which to accept messages.
#QUEUE_NAME
TOPIC_NAME=etASNInFromRIBToSIM

# Makes the subscription durable (see JMS specification).
DURABLE_SUBSCRIBER=true

# JMS_COMPONENT_TYPE=Subscriber

# The messaging group for which to listen.
MESSAGING_GROUP=

# Module name to be used for the Rib's context object.
MODULE_NAME=ISOSSubscriberMessagingComponent

# Sub-module name to be used for the Rib's context object.
SUB_MODULE_NAME=ASNIn

# If TRUE, only a single thread will be used to call the
# processMessages(ArrayList) method. If FALSE, multiple
# threads may call this method. Default is TRUE.
SINGLE_THREADED=TRUE

# Disconnect from the server for the specified number of minutes between checks
# for messages. Note that this does not make sense for a non-durable topic-based
# subscription. Therefore, a value of 0 will cause the component to stay
# connected. The default is 0.
CONNECTION_INTERVAL=0

# The config file to use for setting up messaging. The default is
# "messaging.cfg".
MESSAGING_CONFIG=ribmessaging.cfg

#####
# STANDARD COMPONENT PARAMETERS
#####

# Remote Object Lookup Name
REMOTE_NAME=ASNInMessagingComponent

# Collect Performance Statistics
PERFORMANCE=true
# Logging
LOGGING_IMPL=com.retek.iso.cr.logging.LoggingFileServices
LOGGING_FILE_NAME=../log/asnirmessagingcomponent.log
LOGGING_LEVEL=4
LOGGING_PAUSE=5000
LOGGING_SYSTEM_OUT=true
LOGGING_SYSTEM_ERR=true

ribmessaging.cfg File
This configuration file is used to configure the JMS messaging parameters across all of the RIB’s APIs, publishing and subscribing.

# The client impl is the class that implements the MessagingServices contract.
CLIENT_IMPL=com.retek.rib.redsky.RibSeeBeyondJmsServices

# The time-to-live for messages sent to the server.
MESSAGE_LIFETIME=1800000
USE_SESSION_TRANSACTION=true

# This identifies a specific instance of SIM on the JMS topic (used for retryLocation)
UNIQUE_LOCATION_ID=Store1

#---------------------
# Grouping Information
#---------------------

# Default class uses other properties
GROUPING_UTIL=com.retek.iso.cr.messaging.grouping.MessagingGroupService

# POS device-specific class uses Store and Register in global repository
GROUPING_UTIL=com.retek.iso.cr.messaging.grouping.POSMessagingGroupService

# Default group name
DEFAULT_GROUP=RTK

# Default device info
DEVICE_ID=NONE

# Default device type (Unknown = -1, RNS = 0, Application Server = 1, POS terminal = 2, Web server for PDA = 3)
DEVICE_TYPE=-1
# JMS-SPECIFIC

# The Broker is the JMS server address and port. The default port is 1099
# for OpenJMS. We change it to 9099 to avoid a conflict with Cloudscape.

# Note: The broker host name must match the SeeBeyond Connect Point or
# the hospital retry will not find messages in RIB_MESSAGE
BROKER=mspdev14.retek.int:24444

# Username and password are set via administration of the JMS server
USERNAME=
PASSWORD=

# Number of times to try getting a connection to JMS server
MAX_CONNECTION_TRIES=2

# Number of seconds to pause between connection attempts
PAUSE_BETWEEN_TRIES=2

# Number of seconds to pause between "ping" attempts. The system
# will ping the JMS queue to see if it's still there, and then
# wait this number of seconds before pinging again. If this entry
# does not exists, the number of seconds defaults to 5.
PAUSE_BETWEEN_PINGS=5

# Whether to skip publication to the Jms queue manager, and insert
# published messages directly into the Rib hospital tables.
SKIP_JMS=false
Appendix: RIB Installation Checklist

These notes are intended as an aid in the installation of RIB. They are not intended to replace the detailed description of each of the process steps and prerequisites, but to act as a companion to those steps. For a successful installation, a methodical reading and understanding of each step of the Install Guide is a must.

### SeeBeyond e*Gate Installation Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Installation</td>
<td></td>
</tr>
<tr>
<td>UNIX server – OS requirements met</td>
<td>Each OS (AIX, HPUX, Solaris, etc) require specific OS levels and</td>
</tr>
<tr>
<td></td>
<td>patches. See the SeeBeyond Installation Manual for specifics</td>
</tr>
<tr>
<td>eGate user created</td>
<td></td>
</tr>
<tr>
<td>File systems and directories configured</td>
<td></td>
</tr>
<tr>
<td>Installed JRE</td>
<td>See the RIB Release Notes for required version.</td>
</tr>
<tr>
<td>Install SeeBeyond’s e*Gate – Registry and</td>
<td>Configurations will vary based on site architectures. For the RIB 11.x</td>
</tr>
<tr>
<td>Participating Hosts</td>
<td>releases, SeeBeyond eGate 5.0.4 is required. Please see the SeeBeyond</td>
</tr>
<tr>
<td></td>
<td>eGate 5.0.4 release documentation for details</td>
</tr>
<tr>
<td>Install SeeBeyond’s e*Gate Add-ons</td>
<td>See RIB Release Notes for required list</td>
</tr>
<tr>
<td>Install ESRs</td>
<td>See RIB Release Notes for required list</td>
</tr>
<tr>
<td>Install SeeBeyond’s GUI Tools</td>
<td></td>
</tr>
<tr>
<td>Verify GUI tools connection to Default Schema</td>
<td></td>
</tr>
</tbody>
</table>
## Oracle Retail RIB Installation Checklist (SeeBeyond)

<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Preliminary steps</strong></td>
<td></td>
</tr>
<tr>
<td>Copy the RIB tar files to INSTALL directory.</td>
<td></td>
</tr>
<tr>
<td>Un-tar files and set permissions</td>
<td>Observe sequence requirements. RMS must be un-tar’d last.</td>
</tr>
<tr>
<td>Download hibernate2 jar file and put it in external-lib directory.</td>
<td>download version 2.1.8 from <a href="http://prdownloads.sourceforge.net/hibernate">http://prdownloads.sourceforge.net/hibernate</a></td>
</tr>
<tr>
<td>Edit the file egate_profile (located in &lt;RETEK_INSTALL_DIR&gt;/RIB&lt;version&gt;)</td>
<td>Add site specific info. Changes may be need to point to different JRE than SB default</td>
</tr>
<tr>
<td>Edit egateclient.sh</td>
<td>May not be needed. Changes may be need to point to different JRE than SB default</td>
</tr>
<tr>
<td>Edit the “egate” user’s .profile</td>
<td>Add sourcing of egate_profile</td>
</tr>
<tr>
<td>Source the egate_profile</td>
<td>Either logout and back in, or</td>
</tr>
<tr>
<td>Verify Environment variables set</td>
<td>&gt; set</td>
</tr>
<tr>
<td>Verify xterm capability</td>
<td>Set DISPLAY = your PC’s IP address:0.0</td>
</tr>
<tr>
<td>Create new RIB schema</td>
<td>Using GUI Admin tool.</td>
</tr>
<tr>
<td>Open xterm window</td>
<td>For example, RIB1110</td>
</tr>
<tr>
<td>Run the “install” script</td>
<td>This imports all of the appropriate e*ways.</td>
</tr>
<tr>
<td>Verify no errors.</td>
<td></td>
</tr>
<tr>
<td>Verify Install with GUI Admin Tools</td>
<td></td>
</tr>
</tbody>
</table>
### Oracle Retail RIB Configuration Checklist (SeeBeyond)

<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify the main Participating Host and Control Broker configuration.</td>
<td>These must be configured to the correct host name and network.</td>
</tr>
<tr>
<td>Modify the JMS IQ Manager configuration</td>
<td>Set port number</td>
</tr>
<tr>
<td>Modify Connection Point configurations</td>
<td>Modify to reflect the JMS IQ Manager host and port and Oracle databases used.</td>
</tr>
<tr>
<td>Change SeeBeyond JMS Connection Points</td>
<td>In the default installation, all SeeBeyond JMS Connection points share the same configuration file. Modify to host and port.</td>
</tr>
<tr>
<td>Change Oracle Database Connection points</td>
<td>Alter to reflect the database instance and the user-ID/login for each of the applications</td>
</tr>
<tr>
<td>Delete unused e*Ways</td>
<td>Be careful.</td>
</tr>
<tr>
<td>Add/Copy e*Ways for additional components</td>
<td>Add custom e*ways here.</td>
</tr>
<tr>
<td>Edit the rib.properties file. (as needed)</td>
<td>Edit to reflect the site-specific mappings and properties</td>
</tr>
<tr>
<td>Create/modify startup scripts and Egate.txt</td>
<td>Default scripts are provided.</td>
</tr>
<tr>
<td>Update the hibernate.cfg.xml.</td>
<td>Update with valid database settings (database user, database password). Make sure connection.url property is not defined in this file.</td>
</tr>
<tr>
<td>Copy classes12.zip file to server and client classpath from:</td>
<td>To:</td>
</tr>
<tr>
<td><code>&lt;RETEK_INSTALL_DIR&gt;/RIB_SCHEMA&gt;/ThirdPartyLib/oracle/classes12.zip</code></td>
<td><code>$EHOME/client/ThirdParty/oracle/classes/classes12.zip</code></td>
</tr>
<tr>
<td>Configure component.xml (as needed)</td>
<td>This step is required for the delta install only</td>
</tr>
</tbody>
</table>
## Oracle Retail RIHA Installation Checklists

<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pre-Installation</strong></td>
<td></td>
</tr>
<tr>
<td>Verify the JRE Installed on workstation/PC where RIHA will be installed</td>
<td>The minimum and preferred Java Runtime Engine (JRE) version to use with RIHA is 1.4.2_10</td>
</tr>
<tr>
<td>The RIB DTDs must be made network-accessible in order for RIHA to properly display RIB messages</td>
<td>All of the DTD files in <code>&lt;RETEK_INSTALL_DIR&gt;/RIB&lt;Version&gt;/XML/DTDs/dtd.war</code> should be deployed to a web server running at the client.</td>
</tr>
<tr>
<td>Verify RIB Error Hospital database tables are compatible.</td>
<td>There are changes to the Hospital tables between 11.x and 12.x. If your database already has these tables in place from a prior RIB installation, the tables may need to be updated with new table changes. There are scripts to do this (see RIB Installation Guide.)</td>
</tr>
<tr>
<td>Copy the <code>riha1201_eng.ga.tar</code> file to the location where RIHA will be installed</td>
<td>For example, <code>C:\RIB_Tools</code></td>
</tr>
<tr>
<td>Decompress the tar file with an archive utility</td>
<td>For example, WinZip</td>
</tr>
<tr>
<td>Download hibernate jar file and put it in external-lib directory.</td>
<td></td>
</tr>
<tr>
<td>Execute the <code>riha-config.bat</code> file. Follow instructions</td>
<td>This script will drive the rest of the installation and configuration process.</td>
</tr>
<tr>
<td>Verify RIHA is configured and ready.</td>
<td>Execute the <code>riha.bat</code> script to start.</td>
</tr>
</tbody>
</table>
SeeBeyond Install Notes

These notes are intended as an aid in the installation of SeeBeyond 5.0.5. See the e*Gate Integrator Installation Guide Release 5.0.5 Readme.txt for a detailed description of each of the process steps and prerequisites. Substitute site-specific parameters as appropriate. These notes were compiled during installations on a Sun Solaris platform.

Assumptions:

- All OS level patches installed and verified.
- mount the SeeBeyond e*gate 1st CD or untar or unzip into a temporary directory
- and execute the install script
  - cd /vol.cd/egate/egate.505.hosts/setup
  - chmod -R 755 *
  - ./setup.sh

* The script will begin execution -- it is pure text output and expects the
  * [enter] key to take the default.

* prompt for accept license [no]
  > yes

* prompt for Installation type -- Install the Registry 1st
  > 2

* prompt for install path -- Make sure this is where you want it.
  Installation path [/files0/egate/egate/server]:
  > [enter]

* Prompt for ACL
  Do you wish to enable ACL security now? [no]:
  > [enter]

* Choose replication option
  * 1. None
  > 1

* It will begin unpacking files and prompt for [enter] to continue
  * continue till the next real prompt

* Root password prompt -- Do not need it
  > [enter]

* Prompt for host -- pick appropriate platform
  > 2

* Will unpack files then return to same menu -- pick 0 to finish
  > 0

* Prompt for Administrator password -- use STC (uppercase)
  > STC

  Verify Administrator password:
  > STC

* Registry Port -- default is 23001
  Please specify registry port number [23001]:
  > [enter]
* It will unpack and start the registry and prompt to continue
* several times to install the default schema and services
*
* Prompt for Installation Type -- Pick Participating Host
> 1

> Installation path [/files0/egate/egate/client]:  
> [enter]

* Prompt for In-schema debugging
The Java 2 SDK is required for the runtime In-Schema debugging.  
Do you wish to enable In-schema debugging on this parthost? [yes]:  
> [enter]

Please enter the Java 2 SDK Installation path (e.g. /opt/java1.4):  
* It is looking for some files in jre/lib path.
> /vol.rtk/java/solaris/java1.4.2_10

Do you wish to enable ACL security now? [no]:  
> [enter]

NOTE: You must specify an e*Gate registry server that is currently running
Hostname of Registry Server [msptrn01]:  
> [enter]

Please specify registry port number [23001]:  
> [enter]

Please specify the schema for which your control broker will be started
Schema name [MySchema]:  
> [enter]

Administration Login [Administrator]:
[enter]
Enter password for Administrator:
> STC [enter]
Verify Administrator password:
> STC [enter]

Logical Name for this host's
Control Broker [msptrn01_cb]:  
> [enter]

* several prompts to continue as it displays progress

The root password is now required, in order to enable autostart of the
e*Gate participating host services after system reboots.  
If you do not know the root password, or otherwise do not want to install
the startup scripts at this time, just press ENTER at the password prompt.
Password:
> [enter]         *no password

Please specify where you want to store your e*Gate password file.
Password file path [/files0/egate/egate/client]:  
> [enter]

* several prompts to continue as it displays progress

e*Gate Participating Host (Client) installation completed.
Installation type (choose one):
0. Finished with installation. Quit.
1. e*Gate Participating Host (Client)
2. e*Gate Registry Server
> 0

***********
* mount the addons cd
*
> cd /vol.cd/egate/egate.505.addons.disk1/setup
> chmod -R 755 *
> ./setup.sh

Installation type (choose one):
0. Finished with installation. Quit.
1. e*Gate Add-on Applications
> 1

Installation path [/files0/egate/egate/client]:
> [enter]

Updating /files0/egate/egate.ini

The installation directory /files0/egate/egate/client already exists.
Type U to update the contents of /files0/egate/egate/client
Type M to move the contents of
/files0/egate/egate/client to
/files0/egate/egate/client.old

U)pdate or M)ove? [U]:
> [enter]

Hostname of Registry Server [msptrn01]:
> [enter]
Please specify registry port number [23001]:
> [enter]
Registry port number: 23001

Administration Login [Administrator]:
Enter password for Administrator:
> STC
Verify Administrator password:
> STC

Choose Add-on categories to install.
0. Finished choosing add-on categories; continue.
1. eWays
> 1

Select one of the following to install.
2. Batch e*Way 5.0.5
> 2
Support hpux11 for Batch_eWay add-on [No]: > [enter]
Support aix43 for Batch_eWay add-on [No]: > [enter]
Support win32 for Batch_eWay add-on [No]: > [enter]
Support ctru64_4 for Batch_eWay add-on [No]: > [enter]
Support linux6x86 for Batch_eWay add-on [No]: > [enter]

Select one of the following to install.
6. Oracle e*Way 5.0.5
> 6

Support hpux11 for Oracle_eWay add-on [No]: > [enter]
Support aix43 for Oracle_eWay add-on [No]: > [enter]
Support win32 for Oracle_eWay add-on [No]: > [enter]
Support ctru64_4 for Oracle_eWay add-on [No]: > [enter]

Choose Add-on categories to install.
0. Finished choosing add-on categories; continue.
1. eWays
> 0

> 0

e*Gate Add-on Applications installation completed.

Installation type (choose one):
0. Finished with installation. Quit.
1. e*Gate Add-on Applications
> 0

STC e*Gate Installation complete.
## Oracle Retail RIB Installation Checklist (JavaEE – OAS)

<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pre-Installation</strong></td>
<td>Java 1.4.2.10 JDK is already installed. 10g (10.2.0.1) Database is already installed. RIB tables are already installed in the retail application database schema. Egate 5.0.5 is already installed. OAS 10.1.3 is already installed. Retail application &lt;app&gt; that this rib&lt;-app&gt; will connect to is already installed in OAS.</td>
</tr>
<tr>
<td><strong>Gather Info for Environment Parameters</strong></td>
<td></td>
</tr>
<tr>
<td>Info about the application (i.e RPM)</td>
<td></td>
</tr>
<tr>
<td>RETAIL_APP_OPMN_HOST_NAME</td>
<td>Hostname where the opmn server is running for the application server where &lt;app&gt; ear file is installed.</td>
</tr>
<tr>
<td>RETAIL_APP_OPMN_REQUEST_PORT</td>
<td>opmn “request” port for OAS where &lt;app&gt; ear is installed. The port can be found in $ORACLE_HOME/opmn/conf/opmn.xml file.</td>
</tr>
<tr>
<td>RETAIL_APP_OC4J_INSTANCE</td>
<td>oc4j instance name where &lt;app&gt; ear is installed.</td>
</tr>
<tr>
<td>RETAIL_APP_NAME</td>
<td>&lt;app&gt; application names as seen through AS control.</td>
</tr>
<tr>
<td>RIB Error Hospital Database JDBC connection URL for rib&lt;-app&gt;</td>
<td>Usually the RIB Error Hospital tables are installed in the same database schema as the retail application &lt;app&gt;.</td>
</tr>
<tr>
<td>RIB Error Hospital Database username /password for rib&lt;-app&gt;</td>
<td></td>
</tr>
</tbody>
</table>
| Info about the rib application(i.e ribforrpm)                      | opmn “request” port for OAS where rib<-app> ear will be installed. The port can be found in $ORACLE_HOME/opmn/conf/opmn.xml file. For example, <notification-server interface="ipv4">
| RIB_APP_OPMN_REQUEST_PORT                                         | <port local="6110" remote="6210" request="6013"/>                                                                                  |
| RIB_APP_OAS_INSTANCE_NAME                                          | OAS instance name under which oc4j instance will be created for rib<-app>. The OAS instance name can be found in $ORACLE_HOME/opmn/conf/opmn.xml file |
| JMS_HOST                                                          | Egate JMS server host name.                                                                                                          |
| JMS_PORT                                                          | Egate JMS server port number.                                                                                                        |
| Prepare the OAS to install rib<-app> Java EE application. rib<-app> |                                                                                                                                 |
### Task
Create as needed a separate OC4J instance for this install.

<table>
<thead>
<tr>
<th>Notes</th>
</tr>
</thead>
</table>
| It is required that <app> and rib-<app> applications do not share an oc4j instance and are installed in its own respective oc4j instance. 

```
$ORACLE_HOME/bin/createinstance -instanceName rib-<app>-oc4j-instance
```

replace <app> with the actual value of the retail application name. The only acceptable values for the <app> are either “rpm”, “aip” or “rcom” (lower case).

### Configure rib<app>oc4j-instance

Edit 
$ORACLE_HOME/opmn/conf/opmn.xml file and add the following properties to the rib-<app>-oc4j-instance definition.

**Note:** All the properties below must be applied only to rib-<app>-oc4j-instance definition.

1. Change the default (1.5) JVM to a 1.4.2 JVM.
   
   Add the following element under the start-parameters category element.
   
   ```
   <data id="java-bin" value="<your path to java home>/bin/java"/>
   ```
   
   Put the full path to the location where your java 1.4.2 executable is located.
   
   ```
   <data id="java-bin" value="/vol.rtk/java/solaris/java1.4.2_10/bin/java"/>
   ```

2. Update the java-options section to add the following JVM properties.
   
   Specify the JVM’s min and max heap size.
   
   ```
   -Xms500M -Xmx900M
   ```
   
   Switch off JMX security.
   
   ```
   -Doc4j.jmx.security.proxy.off=true
   ```
   
   For example, `<data id="java-options" value="-Doc4j.jmx.security.proxy.off=true -Xms500M -Xmx900M -Dibm.cl.eagerresolution 

   Djava.security.policy=$ORACLE_HOME/j2ee/rib-rpm-oc4j-instance/config/java2.policy -Djava.awt.headless=true -Dhttp.webdir.enable=false"/>
   ```
   
   Add the following JVM property for OAS on AIX only.

   ```
   -Dibm.cl.eagerresolution
   ```

2. Add oc4j-options to add the following oc4j properties.

   Specify -userThreads to oc4j-options element.

   ```
   <data id="oc4j-options" value="-userThreads"/>
   ```

Make sure that numprocs attribute for the process-set element for rib-<app>-oc4j-instance is set to 1.

For example, `<process-set id="default_group" numprocs="1"/>`
<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reload the new configuration</td>
<td>$ORACLE_HOME/opmn/bin/opmnctl reload</td>
</tr>
<tr>
<td>Start the new rib-&lt;app&gt;-oc4j-instance</td>
<td>$ORACLE_HOME/opmn/bin/opmnctl startproc process-type=&quot;rib-&lt;app&gt;-oc4j-instance&quot; replace &lt;app&gt; with the actual value of the retail application name. The only acceptable values for the &lt;app&gt; are either “rpm”, “aip” or “sim”.</td>
</tr>
<tr>
<td>Make sure that the pre-req software components are properly configured and are up before proceeding.</td>
<td>Database is up. Egate JMS is up Newly created rib-&lt;app&gt;-oc4j-instance is up.</td>
</tr>
<tr>
<td>Installing rib-&lt;app&gt; on newly created rib-&lt;app&gt;-oc4j-instance</td>
<td>This directory is referred to as $RIB_INSTALL_HOME</td>
</tr>
<tr>
<td>Create a “rib_install_stage” directory on the server where OAS is already installed and where rib-&lt;app&gt; will be installed</td>
<td>Untar the file</td>
</tr>
<tr>
<td>Untar the file</td>
<td>tar xvf ribpak&lt;version&gt;-for&lt;app_version&gt;-eng_ga.tar</td>
</tr>
<tr>
<td>Warning: The RIB has been certified with only hibernate 2.1.8, do not download any other hibernate version</td>
<td></td>
</tr>
<tr>
<td>Tip: Make sure that there are no extra spaces before/after in any of the rib-config.properties files for installation.</td>
<td>Tip: Make sure that there are no extra spaces before/after in any of the rib-config.properties files for installation.</td>
</tr>
<tr>
<td>OC4J_HOME</td>
<td>The home directory of the Oc4j AppServer instance where the rib-&lt;app&gt; app will be installed. In OAS it must be $ORACLE_HOME/j2ee/&lt;oc4j instance name&gt;. OC4J_HOME=/home/prantor/development/oc4j_10.1.3/j2ee/rib-rpm-oc4j-instance</td>
</tr>
<tr>
<td>JAVA_HOME</td>
<td>JAVA_HOME=/usr/java/jdk1.4.2</td>
</tr>
<tr>
<td>is_managed_oc4j</td>
<td>This property must always be true. is_managed_oc4j=true</td>
</tr>
<tr>
<td>host_name</td>
<td>The host name where the rib-&lt;app&gt; app will be installed. In Managed OC4J(OAS) host_name is the name of the machine where OAS cluster is running. host_name=localhost</td>
</tr>
<tr>
<td>Task</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>opmn_request_port</td>
<td>Management port number of the server where rib-&lt;app&gt; app will be installed. In managed OC4j(OAS) the port number is the opmn request port number. Use the value of $RIB_APP_OPMN_REQUEST_PORT identified in the information gathering section. opmn_request_port=6004</td>
</tr>
<tr>
<td>oc4j_ormi_port</td>
<td>The ormi port of the Oc4j instance where the rib-&lt;app&gt; app will be installed. In Managed OC4j(OAS) oc4j_ormi_port is not applicable as it is dynamically assigned in the cluster. oc4j_ormi_port=</td>
</tr>
<tr>
<td>oas_instance_name</td>
<td>The OAS instance name. Use the value of $RIB_APP_OAS_INSTANCE_NAME identified in the information gathering section. oas_instance_name=OAS.mspdev05</td>
</tr>
<tr>
<td>oc4j_instance_name</td>
<td>Oc4j instance name where the rib-&lt;app&gt; app will be installed. The value should be the new oc4j instance created for rib-&lt;app&gt; application. oc4j_instance_name=rib-rpm-oc4j-instance</td>
</tr>
<tr>
<td>oc4j_admin_user</td>
<td>Oc4j admin user/password. The oc4j admin user of the Oc4j instance where the rib-&lt;app&gt; app will be installed. oc4j_admin_user=oc4jadmin oc4j_admin_password=oc4jadmin</td>
</tr>
<tr>
<td>JMS_HOST</td>
<td>The JMS Host name for the SeeBeyond Generic JMS Provider. JMS_HOST=msppc004137</td>
</tr>
<tr>
<td>JMS_PORT</td>
<td>The JMS Port number for the SeeBeyond Generic JMS Provider. JMS_PORT=24053</td>
</tr>
<tr>
<td>ribDbJdbcUrl</td>
<td>JDBC URL to the rib-&lt;app&gt; database. This should point to the database schema where the RIB Error Hospital tables are installed. ribDbJdbcUrl=jdbc:oracle:thin:@mspdev02:1522:ribdt01</td>
</tr>
<tr>
<td>ribDbUserId</td>
<td>JDBC user/password for rib-&lt;app&gt; database. ribDbUserId=ribdev3 ribDbPassword=retek</td>
</tr>
<tr>
<td>Edit rib.properties</td>
<td>$RIB_INSTALL_HOME/RIBfor&lt;App&gt;&lt;Version&gt;/rib/properties/rib.properties</td>
</tr>
<tr>
<td>log.default.file_path</td>
<td>The value of this property must be the full path to where rib-&lt;app&gt; will write logs incase of a fatal error during message processing. This must always point to $ORACLE_HOME/j2ee/rib-&lt;app&gt;-oc4j-instance/log/rib-rpm log.default.file_path=/u00/OAS/j2ee/rib-rpm-oc4j-instance/log/rib-rpm</td>
</tr>
<tr>
<td>dtd_url.default</td>
<td>The value of this property must point to the location URL where the dtd.war is deployed. dtd_url.default=http://&lt;your host name&gt;:&lt;your web port name&gt;/dtd/</td>
</tr>
<tr>
<td>Task</td>
<td>Notes</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>mail.smtp.host</td>
<td>rib-&lt;app&gt; can send email alert messages when there is a fatal error in the system. mail.smtp.host – Your SMTP server host. eg. mail.retek.int</td>
</tr>
<tr>
<td>mail.smtp.port</td>
<td>mail.smtp.port – Your SMTP server port. eg. 25</td>
</tr>
<tr>
<td>mail.smtp.from</td>
<td>mail.smtp.from – The email address that will be the source of the alert. eg. <a href="mailto:name@company.com">name@company.com</a></td>
</tr>
<tr>
<td>war.http.port</td>
<td>This property should be set to the value of the HTTP port for the rib-&lt;app&gt;-oc4j-instance</td>
</tr>
<tr>
<td>rib.jms.hostname</td>
<td>This is the hostname where the eGate JMS server is running</td>
</tr>
<tr>
<td>rib.jms.port</td>
<td>This is the JMS port number of the eGate JMS server.</td>
</tr>
</tbody>
</table>
| Edit jndi_providers.xml | $RIB_INSTALL_HOME/RIBfor<App><Version>/rib/retek/properties/jndi_providers.xml. Identify the provider section where app="<app>" under the OAS section and update the JNDI url. Remember the only acceptable values for the <app> are either “rpm”, “aip” or “sim”. The url is of the following format url="opmn:ormi://<RETAIL_APP_OPMN_HOST_NAME>:\<RETAIL_APP_OPMN_REQUEST_PORT>:\<RETAIL_APP_OC4J_INSTANCE>\<RETAIL_APP_NAME>" For example, <provider app="rpm" url="opmn:ormi://mspdevel40.us.oracle.com:6007:rpm-oc4j-instance/rpm" factory="oracle.j2ee.rmi.RMIInitialContextFactory" />
| Start the rib-<app> installation | cd $RIB_INSTALL_HOME/RIBfor<App><Version>/rib ./ribinstall.sh **Note:** The script does not redirect standard out and standard error to a file. If you want to save a history of the install redirect it using the shells redirection mechanism. |
| Configuring the JNDI information in the retail application <app> | The retail application needs to know how to connect/lookup the PublisherEjb through JNDI. The jndi_providers.xml in the retail applications side (<app>, not rib-<app>) has to be updated with rib-<app> ear’s JNDI information. **Note:** The location of the jndi_providers.xml in the retail application (<app>) side is not managed/controlled by rib-<app> so check the retail application’s documentation about the location of this file. |
| update JNDI url for the provider where app="rib-<app>" | The url must be of the following format. url="opmn:ormi://<RIB_APP_OPMN_HOST_NAME>:<RIB_APP_OPMN_REQUEST_PORT>:<RIB_APP_OC4J_INSTANCE>/<RIB_APP_NAME>"
<table>
<thead>
<tr>
<th>Task</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restart the retail application</td>
<td>Login through the AS Control and check the rib-&lt;app&gt; application status. It should say that the application (rib-&lt;app&gt;) is up.</td>
</tr>
<tr>
<td>Verification of rib-&lt;app&gt; install check the rib-&lt;app&gt; application status.</td>
<td>Check the error hospital retry log files in rib-&lt;app&gt;-oc4j-instance/log/rib-&lt;app&gt;/*.log. There should not be any error messages.</td>
</tr>
<tr>
<td>Check the error hospital retry log files.</td>
<td>Login to the Egate JMS Administrator. The subscriber names on these topics should how up as rib_&lt;app&gt;_&lt;interface name&gt;.</td>
</tr>
<tr>
<td>The topic names that rib-&lt;app&gt; subscribes to should get created (if not already there) and should be visible through the JMS Administrator.</td>
<td></td>
</tr>
</tbody>
</table>