



Database Migration for Siebel Billing Manager and Communications Service Manager

Version 5.1

Date Published 02/28/2006

Copyright © 2005, 2006, Oracle and/or its affiliates
All rights reserved

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

1 Migrating to a New Version of Billing Manager or Service Manager

Overview 5

MSSQL 5

 Migrating 4.5.3 to 5.1.0 5

 Migrating 4.5.2.0 to 4.5.3.0 6

Oracle on Unix 6

 Migrating to 5.1 6

 Migrating from 4.5.3 to 5.1 (CSM Only) 8

Oracle on Windows 9

 Migrating 4.5.3 to 5.1.0 9

 Migrating 4.5.3.0 to 4.5.3.x 10

 Migrating 4.5.2.0 to 4.5.3.0 10

 Migrating from 4.5.3 to 5.1 (CSM Only) 10

2 Migrating To A New Version of eaDirect

Overview 13

Migrating an Existing Oracle Database 13

Checking for Errors and Resetting Permissions 16

 Checking .log files for errors 16

Updating Database Passwords for Your Application Server 16

 UNIX 16

 Windows 17

Migrating J2EE Applications 17

Migrating To a Custom Data Source EJB 17

3 Migration Notes

eaDirect 19

eaPay 22

SDK 22

4 Account Data Migration

Migrating data from Index (estatement) tables 25

1

Migrating to a New Version of Billing Manager or Service Manager

Overview

This chapter provides instructions for migrating to a new version of Billing Manager or Service Manager. This involves:

- Backing up your existing database and checking logins and user objects
- Migrating your OLTP and OLSP databases
- Checking for errors and resetting permissions after database migration
- Un-deploying J2EE web applications and redeploying the new versions
- Migrating to a custom data source as necessary

Always check the release notes for the version you are upgrading to for any last minute information, which may update or supersede information in this document.

The rest of this chapter describes the migration steps for each database platform for Billing Manager and Service Manager.

MSSQL

This section describes how to migrate a Self-Service Microsoft SQL server database from one version to another on the Windows operating system.

Migrating 4.5.3 to 5.1.0

This section describes how to migrate an existing Self-Service database from version 4.5.3 to version 5.1.0.

To migrate the Self-Service database:

- 1 Set the ANT_HOME, JAVA_HOME and PATH environment variables in the command prompt

For example:

```
C:\set ANT_HOME=C:\Siebel\CBM\ant
```

```
C:\set JAVA_HOME=D:\bea81sp4\jdk142_05
```

```
C:\set PATH=%JAVA_HOME%\bin;%ANT_HOME%\bin;%PATH%
```

- 2 Change your working directory to <TBM_HOME>\ db\mssql\migration\453to510 .
- 3 Change the following property values in the cbm510.properties file in the folder for your environment:

DB_NAME
DB_USERNAME
DB_PASSWORD
DB_TMPUSERNAME
DB_TMPPASSWORD
DB_SYSADMIN
DB_ADMINPASSWD
INSTALL_DB_HOME

- 4 Execute the ant build script in the command prompt. For example:

```
C:\>ant -f dbinstall.xml
```

(The above command should be executed in the same machine as the DB server)

Migrating 4.5.2.0 to 4.5.3.0

This section describes how to migrate an existing Self-Service database from version 4.5.2.0 to version 4.5.3.0.

To migrate the Self-Service database:

- 1 Change your working directory to <TBM_HOME>\db\mssql\migration\452to453.
- 2 Log into SQL Query Analyzer as CBM db user and execute these scripts in sequence:

```
drop_tsm_schema.sql  
create_tsm_schema.sql  
create_tsm_const.sql  
create_migrate_data.sql
```

Oracle on Unix

This section describes how to migrate a Self-Service Oracle database from one version to another on a Unix system.

Migrating to 5.1

To migrate the Self-Service database:

- 1 Log into the database server as oracle user.
- 2 Change your working directory to <TBM_HOME>\db\oracle
- 3 Execute the setup_schema.sh script. For example:

```
./setup_schema.sh
```

The following screen displays:

```
-CBM Enhancement Main Menu
-----
[1] Sign in Menu
[2] Apply CBM enhancement

[3] Initial data population
[4] CBM Migration

[Q] Quit
-----
Enter Your Selection:
```

- 4 Select option 1. The login screen displays (example entries are also shown):

```
                SIGN IN MENU
-----

Please enter Oracle SID      --> edx0
(Enter appropriate Oracle SID)

Enter Database Username     --> edx_dba
(Enter user name for the Self-Service database)

Enter Database Password     --> edx
(Enter the Password for the Self-Service database)

Please enter the password for SYS user ... >change_on_install
(Enter the Password for the "sys" user)
```

- 5 Selection option 4 for migration. The following menu displays:

```
Database Version Migration
-----

1. Version 4.5.3.1-4.5.3.8 To 5.1.0.0
2. Version 4.5.2.0 To 4.5.3.1
R. Exit to Main menu

Enter Your Option:
```

- 6 Select option1 (Version 4.5.3.1-4.5.3.8 to 5.1.0.0) to migrate the database from 4.5.3 to 5.1.0. The following is displayed:

```
**Warning**

We strongly advise a full backup of your existing
database before applying the migration.

Do you have a backup (Y/N):Y
```

still continue with migration...(Y/N):Y

The system executes the database scripts. You will be prompted whether to execute each script (Y or N). The default is "Y."

- 7 Select option 2 (4.5.2.0 to 4.5.3.1) to migrate the database from 4.5.2.0 to 4.5.3.1. The following is displayed:

****Warning****

We strongly advise a full backup of your existing database before applying the migration.

Do you have a backup (Y/N):Y

still continue with migration...(Y/N):Y

The system executes the database scripts. You will be prompted whether to execute each script (Y or N). The default is "Y."

Migrating from 4.5.3 to 5.1 (CSM Only)

If you are installing Communications Service Manager, you must also run the following scripts to fully migrate to 5.1:

- 1 Enter the following:

```
cd <CBM_HOME>/db/tsm/oracle/migration/453to453x
sqlplus {user}/{password}@{tns_name}
SQL>@cmscase.sql;
SQL>@create_sequences.sql;
SQL>exit
```

Where {user} is the database user, {password} is the database password, and {tns_name} is the Oracle tns name.

- 2 Enter the following:

```
cd <CBM_HOME>/db/tsm/oracle/migration/453to510
sqlplus {user}/{password}@{tns_name}
SQL>@csm_index.sql;
SQL>exit
```

Where {user} is the database user, {password} is the database password, and {tns_name} is the Oracle tns name.

Oracle on Windows

This section describes how to migrate a Self-Service Oracle database to a newer version.

Migrating 4.5.3 to 5.1.0

To migrate the Self-Service database:

- 1 Change your working directory to <TBM_HOME>\db\oracle\migration\453to510.
- 2 Log into SQLPLUS as Billing Manager user and run the following script:


```
Set ORACLE_SID=<ORACLE_SID>
sqlplus -s /nolog
connect <tbm_user>/<tbm_user_password>
spool migrate_453_510.log
@add_column.sql
```
- 3 Connect into database as the "SYS" db user, and run the following script:


```
connect sys/<sys_password> as sysdba
@temp_user_creation.sql
```
- 4 Connect to the database as the "CBM_SCRPT" user (the password is "cbm"), and run the following script:


```
connect CBM_SCRPT /cbm
@<CBM_HOME>\db\oracle\install_schema_migrate.sql
```

Where CBM_HOME is the absolute path to your CBM installation.
- 5 Connect to the database as the Billing Manager user and run the following script:


```
connect <tbm_user>/<tbm_user_password>
@rename_const_index.sql <tbm_user>
@move_tablespace.sql <tbm_user>

@move_indexes_to_tablespaces.sql <tbm_user>
@rebuild_index.sql <tbm_user>
```
- 6 Connect to the database as the "SYS" user, and run the following script:


```
connect sys/<sys_password> as sysdba
drop user CBM_SCRPT cascade;
spool off
exit;
```
- 7 Check the migrate_453_510.log file for any errors.

Migrating 4.5.3.0 to 4.5.3.x

To migrate the Self-Service database:

- 1 Change your working directory to <TBM_HOME>\db\oracle\migration\453to453x.
- 2 Log into SQLPLUS as Billing Manager user and run the migration script. For example:

```
Set ORACLE_SID=<ORACLE_SID>
sqlplus -s /nolog
connect <tbm_user>/<tbm_user_password>
spool migrate_453_453_x.log
@add_indexes.sql;
spool off
exit;
```
- 3 Check the migrate_453_453_x.log file for any errors.

Migrating 4.5.2.0 to 4.5.3.0

To migrate the Self-Service database:

- 1 Change your working directory to <TBM_HOME> \db\oracle\migration\452to453 .
- 2 Log into SQLPLUS as the Billing Manager user and run the migration script. For example:

```
Set ORACLE_SID=<ORACLE_SID>
sqlplus -s /nolog
connect <tbm_user>/<tbm_user_password>
spool migrate_452_453.log
@drop_tsm_schema.sql;
@create_tsm_schema.sql;
@create_tsm_const.sql;
@create_migrate_data.sql;
spool off
exit;
```
- 3 Check the migrate_452_453.log file for any errors.

Migrating from 4.5.3 to 5.1 (CSM Only)

If you are installing Communications Service Manager, you must also run the following scripts to fully migrate to 5.1:

- 1 Enter the following:

```
cd <CBM_HOME>\db\tsm\oracle\migration\453to453x
sqlplus {user}/{password}@{tns_name}
```

```
SQL>@cmscase.sql;  
SQL>@create_sequences.sql;  
SQL>exit
```

Where {user} is the database user, {password} is the database password, and {tns_name} is the Oracle tns name.

- 2 Enter the following:

```
cd <CBM_HOME>\db\tsm\oracle\migration\453to510  
sqlplus {user}/{password}@{tns_name}  
SQL>@csm_index.sql;  
SQL>exit
```

Where {user} is the database user, {password} is the database password, and {tns_name} is the Oracle tns name.

2 Migrating To A New Version of eaDirect

Overview

This chapter describes how to migrate to a new version of eaDirect. This involves:

- Backing up your existing database and checking logins and user objects
- Backing up your data directories **AppProfiles**, **Data**, **Input** and **Output**
- Migrating your eaDirect database
- Checking for errors and resetting permissions after database migration
- Undeploying J2EE web applications and redeploying the new versions
- Migrating to a custom data source as necessary

CAUTION: The installation and configuration examples shown in this guide use default eaDirect pathnames, privileges, and permissions. If you choose not to accept the default values, make sure your values are consistent on all servers across your installation of eaDirect.

NOTE: When migrating Billing Manager (TBM/eBilling), eaDirect is now called Platform Services, and is located in TBM_HOME/estatement instead of /opt/EDCSbd.

Migrating an Existing Oracle Database

To migrate an existing eaDirect database to a newer version, you must run the database setup shell script **edx_admin.sh**. The script's Main Menu has a Database Version Migration option that lists available migration paths.

To migrate an eaDirect database for Oracle on UNIX:

- 1 Switch user to the **DB Admin** user, in this example **oracle**. Oracle requires the administrative user in order to create files.

```
su - oracle
```

- 2 Make sure that the Oracle listener is running for your eaDirect database.
- 3 Change directory to your eaDirect database home directory. For example:

```
cd /opt/EDCSbd/db/oracle
```

- 4 Start the **edx_admin** shell script with the command:

```
./edx_admin.sh
```

The Server Administration Main Menu appears.

Migrating To A New Version of eaDirect

```
edocs eaDirect Server Administration Main Menu
```

```
-----  
[1] Sign in Menu  
[2] Capture Database File Locations  
[3] Install edocs e-Statement  
[4] Initial Data Population  
[5] Database Version Migration  
[Q] Quit  
-----
```

```
Enter Your Selection: 5
```

5 Select option 5, **Database Version Migration**.

6 Enter the username, password, and database name for the eaDirect database.

CAUTION: When creating an Oracle database, limit its name to eight characters. Defining or entering an Oracle SID with more than eight characters causes the Oracle database configuration to fail.

The following example shows sample values for **Oracle** on **Solaris**. Specify suitable values for your platform.

```
Please enter Oracle SID      --> edx_dba  
Enter Database Username     --> edx  
Enter Database Password     --> edx0
```

Press **Enter** to return to the main menu.

7 From the Database Version Migration menu, select the version you are migrating from and migrating to, for example **40To42**.

CAUTION: Be sure to select the correct version for your existing installation.

8 Indicate whether you have already done a full backup of your database.

We strongly advise taking full backup of your existing database before applying the migration

```
Do you have a backup (Y/N): Y
```

If you have not backed up your database, you are asked whether you want to continue the migration process.

```
Still continue with the migration (Y/N): Y
```

9 Enter your choice to abort the migration process or continue. If **no**, you are returned to the Database Version Migration menu where you can select the option to return to the previous menu.

If **yes**, migration begins. Select 'Q' to end the database migration session.

If you choose to continue with the database migration, the migration script **migrate.sh** runs for your version upgrade.

```
*****  
Migration process started  
*****
```

10 Enter the number of partitions for the Index database table:

Enter the partition count... ([1|4|12] Default is 1)> 1

- 11 The following message appears:

This migration requires the creation of 2 new tablespaces!>
Please enter valid paths and ensure at least 200mb disk space...>

Database files can reside wherever you want them to. If you plan to use only one disk location, you can simply specify:

\$ORACLE_HOME/oradata

Depending on your platform, you may then be prompted to enter multiple locations for your **database control** file. This file allows access to the database, so you should specify backup locations in case a database control file is deleted or corrupted. Consult your Oracle documentation for details.

You may also be prompted to enter suggested **mount points** for your database. Consult your Oracle documentation for details.

The configuration process checks the validity of the specified locations and displays a success message if no problems are encountered.

- 12 A successful database migration displays the following message:

Mi grati on successful

- 13 The script prompts you to check your session log for errors. For example:

```
*****
Please check the log file for any errors...
Log file name: /opt/EDCSbd/db/oracle/migrate0411100446.log
*****
Ending migration process
```

- 14 From the Database Version Migration menu, select **Return to previous menu (R)**.

- 15 From the edocs Server Administration Main Menu, select **Quit (Q)**.

To migrate an eaDirect database for MSSQL on Windows:

Run the **edx_pkg_versioning.sql** and **pwc_runtime.sql** scripts located under <CBM_HOME>\statement\db\mssql:

- 1 Launch a Query Analyzer and login as the database schema owner, for example edx_dba.
- 2 Open the above file within the Query Analyzer.
- 3 Click the green arrow to compile.

Checking for Errors and Resetting Permissions

After you migrate a database from a previous release of eaDirect, you should:

- Check **.log** files for errors
- Reset permissions for edocs directories and files

Checking .log files for errors

After migrating an eaDirect database, you should check the following log files for errors:

- **scan.log**
- **migrate<version>.log**

In the context of migrating an eaDirect database, a **normal error** is one that does not cause the upgrade process to fail, such as trying to drop an object that does not exist. An **abnormal error** is one that can cause the database migration process to fail, which can have a cascading effect. A single abnormal error can lead to many other abnormal errors as the database migration process proceeds.

Although there is no easy way to differentiate between normal and abnormal errors, there is a way to check whether the database upgrade was successful. Typically, if the process does not flag invalid objects or there are no violations of referential constraints, then the operation was successful.

Resetting Permissions

After migrating your database, confirm that all the eaDirect directories, with the exception of the **/db** directory and its subdirectories, reflect the same directory and file ownership as that of the application server owner.

For database files and directories under **\$EDX_HOME**, eaDirect uses the default owner and group permissions.

Updating Database Passwords for Your Application Server

UNIX

If the database password for an existing database has changed, UNIX administrators can update your eaDirect environment with the application server script **edx_config**. Run this script **on your application server** using the instructions in Capturing UNIX Environment Data with `edx_config`. Enter your new password at the prompt.

Once the script has completed, check to see that the following files have been updated with your new password:

- `edx_env`
- `edx_sqllldr.config`
- `ws_config.xml` (WebSphere) or `config.xml` (WebLogic)
- JVM settings for system properties in your application server console

Windows

If the database password for an existing database has changed, Windows administrators can update your eaDirect environment with the application server script `edx_load.config.bat`. Edit this script **on your application server** using the instructions in Editing Windows Environment Data with `edx_env.bat` and `edx_load.config.bat` to have the correct user, password and database alias. This file is then sourced at application server startup.

Migrating J2EE Applications

When migrating to a new version of Siebel Self-Service software, you must first un-deploy J2EE applications running on your application server. Once you have migrated your database, you should then redeploy the new versions of each eaDirect web application, including eaDirect and samples. These sample applications provide examples of how to use the current features of eaDirect. You must:

- See the *Release Notes* for your version for important information about specific migration requirements and settings.
- See *Deploying and Customizing J2EE Applications* for information about the components that make up the J2EE and Web applications, and the procedures for recreating EAR and WAR files.
- Merge your custom code into the sample application, and rename it. For more information, see *Deploying and Customizing J2EE Applications*.
- Redeploy your custom application.

For more information on deployment, see the eaDirect guide to *Deploying and Customizing J2EE Applications*.

CAUTION: If you receive error messages when starting up your application server after migration, check to make sure you have undeployed all old versions of edocs J2EE web applications and redeployed the new versions. edocs does not recommend manually editing the `config.xml` files for deployment.

Migrating To a Custom Data Source EJB

eaDirect 3.4 and later versions allow you to specify a datasource EJB for each eaDirect application (DDN) you create in the Command Center. When creating an eaDirect application in the Command Center, a datasource refers to an EJB in your application (EAR file) that specifies summary information and location of your document data.

CAUTION: Please consult the *Release Notes* for information on migrating from one particular version to another, as migration may require several steps.

Migrating To A New Version of eaDirect

During database migration, the `edx_admin.sh` script maps DDNs to the default datasource **edx/ejb/EdocsDataSource**, packaged in **ear-eStatement.ear** . For more information about datasource mapping, see the eaDirect *Production Guide* and *XML Services Guide*.

To change the datasource value to point to a custom datasource, or to another datasource in a different deployed application, you can edit and run the SQL script **update_data_source.sql** on the database server. This script is located in the `/migration/<version>` subdirectory of the database directory in your eaDirect installation.

TIP: You do not need to run this script if you are using the default datasource.

You need to edit **update_data_source.sql** to replace the placeholder values for datasource name and document definition name (DDN) with your own values. Since each DDN may now have its own datasource, you can copy and paste the update sequence for each DDN. Save and close **update_data_source.sql** and run this SQL script to update your DDNs.

CAUTION: Consult the Release Notes for your version and platform to locate the correct version of `update_data_source.sql`.

3 Migration Notes

These migration notes were originally published in the eaDirect release notes beginning with Version 4.2.1.9.

eaDirect

- If migrating from a version prior to eaDirect 4.0 and using Oracle, please follow the instructions in the document `Index_Table_Migration.txt` located in `$EDX_HOME` in addition to following the normal migration procedures.
- If you are migrating from eaDirect 4.1 and using Oracle, please run the following command after migrating to eaDirect 4.2 (if necessary).

```
./mi grate_4x_patch.sh
```

NOTE: This script prompts for ORACLE_SID, database username and database password.

- It is important to monitor the amount of database space used during the migration. The eaDirect/eaPay migration involves several database tables and their data. The size of the tables varies from small to huge. Depending on the volume of the tables, during the migration the database may use a huge amount of temporary space, undo space (applicable to Oracle), redo space (applicable to Oracle and DB2), and table/index data space. It is recommended that you monitor the space growth and pre-allocate enough space before migration and/or during the migration. It is advisable to allocate a big enough chunk of space just to make sure the migration doesn't fail due to insufficient space.
- For the Oracle database, if you see that any stored object (package, procedure, function) is "invalid" after the migration, it may not be due to the migration. To determine if this is a critical issue, recompile the "invalid" object manually. For example, if DDN_MGR package becomes invalid, log on to the database as the database owner and compile it as follows:

```
SQL> alter package ddn_mgr compile;
```

If the above compile command runs successfully, then this is not an issue and you have resolved it. Otherwise, it may be a critical issue and you should contact edocs Technical Support.

Migration Notes

- eaDirect 4.0 through 4.2 require a separate indexing DDF to be used exclusively with an application's Indexer job. eaDirect indexes all fields in the DDF published for Indexer (and displays all fields in the hit list). The DDF should contain all the fields, group fields, and sub-documents that require indexing. The eaDirect 4.0 through 4.2 database migration migrates the last indexed table structure. The fields defined in the DDF used for the Indexer job must correspond to the migrated index table structure. After you have migrated the database, adding, removing, or renaming fields in the DDF requires that you manually modify your database tables. For a new application, ideally you should define every field you want indexed when you originally create the DDF. Once you run the Indexer job with this DDF, however, adding, removing, or renaming fields in the DDF requires that you manually modify your database tables. For information on how to manually modify the eaDirect database tables to add, remove, or rename existing fields please consult edocs Technical Services.
- As of 4.x, the database migration migrates existing Optional Fields specified in the Index tables. For new Indexer jobs created in 4.x, ten optional fields are created by default. You no longer have the option of specifying how many optional fields to include.
- If you have enabled Charting for Command Center Bill Statistics when using WebLogic prior to 4.0, you need to copy *javachart.jar* included in the *ear-easample.ear* and then edit *edx.config* to add the *javachart.jar* to the CLASSPATH. For specific details, see the WebLogic product notes in the release notes.
- If there is a failed IXLoader task in 3.4.x, then after migrating to 4.2 the instance should be cancelled and re-run, **not** retried from the existing failed instance.
- When you migrate to 4.2 from any prior release, you must delete and recreate any detail extractor jobs (and their schedules). Existing data from an old job can still be seen and is not affected. But the job itself, if you want to run it again, needs to be deleted and recreated to be run successfully. You must use the same detail extractor job name when recreating it, or it reprocesses index files that have already been processed and generate unique constraint errors.
- When you migrate to 4.2, do not remove any of the data files in application data directories. If you do, you will not be able to view previously indexed statements.
- *script.js* has moved. If you are unable to view statements, change the path of this file to *enrollment/scripts/script.js*. More information about *script.js* is provided below. (Note: this only applies for migration from 2.x to 4.2, not 3.x to 4.2.)
- Upon a migration from 2.1.x, both user statistics report and bill statistics report WILL NOT bring any information back. This is due to new tables introduced since 3.0. The reports no longer use the *rpt_activity* table; after migration, the reports include data from 3.x and forward only.
- Migration to eaDirect 4.2 on AIX is from versions 2.2.4.1 and higher. For migration of versions prior to 2.2.4.1 on the AIX platform, please contact Technical Support.
- If eaDirect processed an Email Job on the same day you migrate from a version prior to 3.4, wait at least 1 day before running the Email Job again to set the IVNScanner's "Scan Number of Days" parameter low enough so it doesn't pick up the older, already processed IVNs.
- When migrating a MSSQL database to 4.2 from a version older than 4.0 or 4.1, you must invoke the DBConfigTool more than once. You must run the tool for each step in the process. Database migration is not complete until the final step, "40 or 41 to 4.2" migration, has finished successfully.

- As mentioned in the Installation and Configuration Guides, if you are not using the default datasource you need to run the *update_data_source.sql* script for your database. This only applies for those migrations from a version prior to 3.4. The locations for each file are:
 - Oracle: <eaDirect-home-directory>/db/oracle/migration/32to34/
 - SQL Server: <eaDirect-home-directory>\db\mssql\migration\32to34\
 - DB2: <eaDirect-home-directory>/db/db2/migration/32to343/
- When migrating from versions prior to 3.2, if the user had used restricted characters as part of their password, they will not be able to login in 4.2. They include:


```
!@#%$^&*()_+[]\|}{;'": , . /?><
```
- When migrating your Web applications to eaDirect 4.2, you must upgrade ejb-application.jar to the latest version included in the eaSample sample application provided with eaDirect.
- When migrating to 4.5.2.5 and above, and using the Oracle database, please do the following two tasks: run \$TBM_HOME/estatement/db/oracle/migrate_4x_patch.sh.
 - run the following from the directory \$TBM_HOME/db/oracle/migration/452to4521:


```
sql pl us {user}/password}@{tns_name}
SQL>@upgrade_db.sql
SQL>exi t
```

where {user} is the database user, password is the database password, and {tns_name} is the Oracle database tns name.
- If migrating using an Oracle database, please run:
 - For 4.5.2 or 4.5.2.4 to 4.5.3:


```
sql pl us {user}/{password}@{tns_name}
SQL>@AddTSMSchema.sql
SQL>@MigrateData.sql
SQL>exi t
```

where:

{user} is the database user.
 {password} is the database password.
 {tns_name} is the Oracle tns name.

Note that above sql scripts are found in \$TBM_HOME/db/oracle/migration/452to453
 - For 4.5.2.x or 4.5.3 to 4.5.3.1:


```
Opt/Run estatement/db/oracle/migrate_4x_patch.sh
```

eaPay

- AVS service is an extension to the existing credit card validation process. If the biller has enabled the AVS, they need to specify the AVS enabled Verisign account at the time of Verisign gateway setup. For the migration, AVS is set to true (enabled) by default, so the biller needs to update the created Verisign credit card gateways to specify the AVS enabled Verisign account information. (If the biller does not want it, they can disable the AVS and update the Verisign gateway configuration.) As part of the migration process, the biller needs to send out emails to their customers about this new service and informing them to update their credit card accounts with address information exactly as it appears on the bill, otherwise the transactions will fail.
- The following APIs from the `IPaymentAccountAccessor` interface and `PaymentAccountAccessorImpl` and `SSOPaymentAccountAccessor` classes have been removed and are no longer supported:

```
PaymentAccountQueryResult getPaymentAccount(String uid, String pid,
ExtendedAccountQueryParams extAccQueryParams)
```

```
PaymentAccountQueryResult getPaymentAccounts(String uid,
ExtendedAccountQueryParams extAccQueryParams)
```

```
PaymentAccountQueryResult getPaymentAccounts(IAccount iaccount,
ExtendedAccountQueryParams extAccQueryParams)
```

```
PaymentAccountQueryResult getPaymentAccounts(String biller, PaymentType type,
PaymentAccountState state, int maxCount, ExtendedAccountQueryParams
extAccQueryParams)
```

```
List getPaymentAccountsToBeActivated(String ddn, PaymentType type, int maxCount)
```

```
PaymentAccountQueryResult getPaymentAccounts(String biller, PaymentType type,
String notifyStatus, int maxCount, ExtendedAccountQueryParams extAccQueryParams)
```

```
PaymentAccountQueryResult getExpiredPaymentAccounts(String ddn, PaymentType
pmtType, List status, java.util.Date expireDate, ExtendedAccountQueryParams
extAccQueryParams)
```

SDK

The following are some general migration issues with the APIs for the edocs SDK:

- The `Logout.java` class must implement the method `getUserIdAttribute` of the `com.edocs.app.LoginApp` class. The method should retrieve the user's unique id. It is implemented by the `Login.java` class also.
- Email Notification Job users: The Account Resolver code which is used to return email addresses needs to be rewritten using the `SampleAccountResolver.java` provided in this release. The method `search` has replaced the `getEmailAddressesByAccountNumber`. Also note that every email must resolve at least twice or the system may throw an `ArrayOutOfBoundsException`.

The following are the deprecated APIs from previous releases:

Deprecated Classes

```
com.edocs.jndi.cda.CDADataSourceFactory
```

```
com.edocs.jndi.cda.CDADB2Factory
```

```
com.edocs.jndi.cda.CDAJDBCFactory
```

```
com.edocs.jndi.cda.CDAOracleFactory
```

com.edocs.enrollment.HttpRequestParameters
 com.edocs.enrollment.user.IAdmissionFactory
 com.edocs.enrollment.user.IAuthenticationFactory

Deprecated Interfaces

com.edocs.enrollment.user.IAccountEnumeration
 com.edocs.enrollment.user.IAdmission
 com.edocs.enrollment.user.IAdmissionHome
 com.edocs.enrollment.user.IAuthentication
 com.edocs.enrollment.user.IAuthenticationHome

Deprecated Fields

com.edocs.app.App.METHOD

Deprecated Methods

com.edocs.enrollment.user.IAuthentication.authenticate(NameValue[])
 com.edocs.enrollment.user.IAccount.close()
 com.edocs.enrollment.user.IAdmission.createAccount(NameValue[])
 com.edocs.enrollment.user.IAdmission.deleteAccount(IAccount)
 com.edocs.enrollment.user.IAdmission.fetchAccount(NameValue[])
 com.edocs.app.verify.Verify.getAccountListSize(String)
 com.edocs.enrollment.user.IAccount.getBillers()
 com.edocs.enrollment.user.AccountBean.getBillers()
 com.edocs.app.user.User.getDocument(String, String[], String, String, String, String)
 com.edocs.app.user.User.getDocumentDistriButedInputStream(String, String[], String, String, String, String)
 com.edocs.enrollment.user.IAccountResolver.getEmailAddressesByAccountNumber(String, String)
 com.edocs.enrollment.user.IAccount.getExactValue(String, String)
 com.edocs.enrollment.user.AccountBean.getExactValue(String, String)
 com.edocs.enrollment.user.IAccount.getValue(String, String)
 com.edocs.enrollment.user.AccountBean.getValue(String, String)
 com.edocs.enrollment.user.IAccountResolver.isAccountValid(String, String)
 com.edocs.enrollment.user.IAccount.isValid(NameValue[])
 com.edocs.enrollment.user.AccountBean.isValid(NameValue[])
 com.edocs.enrollment.user.IAuthentication.reAuthenticate(Handle, NameValue[])
 com.edocs.enrollment.user.IAccountResolver.search(Attributes, String[])
 com.edocs.enrollment.user.IAdmission.selectAccounts(NameValue[])
 com.edocs.app.user.User.sendSortedDocument(String, String[], String, String, String, String, SortParams, boolean, boolean, HttpServletRequest, Writer)
 com.edocs.common.web.validation.ValidatorBean.setIllegalParameterValues(String[])
 com.edocs.common.web.validation.ValidatorBean.setLegalParameterNames(String[])
 com.edocs.enrollment.user.IAdmission.updateAccount(IAccount, NameValue[])
 com.edocs.app.user.User.updateSummaryInfo(String, Name, String, String, String)
 com.edocs.common.web.validation.ValidatorBean.validateParameterNames(HttpServletRequest)

com.edocs.common.web.validation.ValidatorBean.validateParameterValues(HttpServletRequest)

4 Account Data Migration

Migrating data from Index (estatement) tables

This document describes a set of migration scripts that can populate specific (defined by PS team) Billing Manager tables from statement account #s stored in index (estatement) tables.

This process requires extracting account #s from index table and then populates necessary Billing Manager tables according to the format described below.

1 (Step 1) Extracting account numbers from index tables

Extract account #s (Z_PRIMARY where Z_CONTEXT == null) from index tables [PS should clarify whether we need to scan all index tables or only the latest index table to get account numbers]

Note: Account numbers can have duplicate entries, so remove duplicate while extracting.

Figure 1 shows sample data and its format in index tables (using simulated data). Siebel Professional Services provides actual index table format and sample data as required, but for this example the Z_PRIMARY and Z_CONTEXT columns are sufficient.

Z_PRIMARY	Z_DOC_DATE	Z_DOC_ID	Z_IVN	Z_CONTEXT	BillDate	PhoneSummary	MobileCount	MobileNum	Group5u
80011008	6/17/2005	ivn-1/po-0/bc-6253	1	<NULL>	05/04/04	600	2	407-113-5451	900
382-473-4937	6/17/2005	ivn-1/po-1003165//	1	06122165	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
382-215-2803	6/17/2005	ivn-1/po-1025259//	1	06122165	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
13875658	6/17/2005	ivn-1/po-1036405//	1	<NULL>	05/04/04	<NULL>	1	784-749-3874	<NULL>
784-749-3874	6/17/2005	ivn-1/po-1038357//	1	13875658	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
14464399	6/17/2005	ivn-1/po-1066157//	1	<NULL>	05/04/04	<NULL>	1	748-373-7646	<NULL>
748-373-7646	6/17/2005	ivn-1/po-1067647//	1	14464399	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
03286008	6/17/2005	ivn-1/po-1124948//	1	<NULL>	05/04/04	600	20	847-349-4100	900
847-349-4100	6/17/2005	ivn-1/po-1134924//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4101	6/17/2005	ivn-1/po-1156027//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4102	6/17/2005	ivn-1/po-1166414//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4103	6/17/2005	ivn-1/po-1175641//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
31569801	6/17/2005	ivn-1/po-121372/b/	1	<NULL>	05/04/04	600	2	494-392-9463	900
847-349-4104	6/17/2005	ivn-1/po-1227056//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4105	6/17/2005	ivn-1/po-1236289//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4117	6/17/2005	ivn-1/po-1248870//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4106	6/17/2005	ivn-1/po-1251577//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
494-392-9463	6/17/2005	ivn-1/po-125282/b/	1	31569801	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4107	6/17/2005	ivn-1/po-1269076//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4118	6/17/2005	ivn-1/po-1334215//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4108	6/17/2005	ivn-1/po-1336922//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4109	6/17/2005	ivn-1/po-1355443//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>
847-349-4110	6/17/2005	ivn-1/po-1368166//	1	03286008	<NULL>	<NULL>	<NULL>	<NULL>	<NULL>

Fig. 1 - Sample data in index table.

2 (Step 2) Insert account numbers (extracted from index tables) into Billing Manager tables.

You must update the following tables for each account number qualified from Step1.

EDX_BSL_AMF_BACCOUNT

EDX_HIER_HNODE

Account Data Migration

Before updating the above two tables with account number information, you must load the static data (pre-defined) for the dependant tables of above two. Listed below are the dependent tables that need to be populated along with their values.

Step 2.1 – Loading static data

EDX_BSL_CMF_COMPANYPROFILE

This table will have only one row.

ID = [sequence]
NAME = __default__

EDX_BSL_CMF_COMPANY

This table will have only one row.

ID = [sequence]
PROFID = [EDX_BSL_CMF_COMPANYPROFILE.ID]

EDX_HIER_HTYPE

This table will have only two rows as described below.

Row1

ID = [sequence]
CODE = "BILLING"
NAME = "Billing Hierarchy"
DESCRIPTION = "Billing hierarchy"

Row2

ID = [sequence]
CODE = "BUSINESS"
NAME = "Business Hierarchy"
DESCRIPTION = "Business hierarchy"

EDX_HIER_HIERARCHY

This table will have only one row.

ID = [sequence]
HTYPE = [EDX_HIER_HTYPE.ID where CODE="BILLING"]
COMPANYID = [EDX_BSL_CMF_COMPANY.ID]
NAME= "Billing Accounts"
DISPLAYNAME = "Billing Accounts"
CREATEDAT = timestamp (I think this is java current time in millisecc from 1/1/1970, 00:00:00. so we need to calculate this dynamically)
CREATEDBY = "siebel admin"
DELETEDAT = 0
PUBLISHEDAT = 0

EDX_HIER_HNODE

NODEID = [sequence]
PARENTID = null
HIERARCHYID = [EDX_HIER_HIERARCHY.ID]
LINKTARGETURI = "edx:hierarchy:folder:31b8b6086207ef7f:1c2fdde:1048bb8e8ac:-

```

8000: root"
LINKTARGETID = "root"
ENTITYSCONTAINER = 1
PATH = 1 [NODEID]
LINKTARGETNAME = "Billing Accounts"
CREATEDAT = timestamp (same as in EDX_HIER_HIERARCHY)
CREATEDBY = "siebel admin"
MODIFIEDAT = timestamp
MODIFIEDBY = "siebel admin"
DELETEDAT = 0
LINKTARGETTYPE = "com.edocs.common.hierarchy.api.HierarchyFolder"

```

Step 2.2 - Updating Account # related tables;

EDX_BSL_AMF_BACCOUNT

Each account number extracted from index tables should be inserted into this table according to the below format.

```

ID = [sequence]
NAME = "<<account #>>" [account # is the value extracted from index table
(Z_PRIMARY)]
TYPE = "ACCOUNT"
CREATEDAT = timestamp (same as in EDX_HIER_HIERARCHY)
CREATEDBY = "siebel admin"

```

For example, after adding two accounts, EDX_BSL_AMF_BACCOUNT looks like;

nodeID	name	description	type	createdAt	createdBy	modifiedAt	modifiedBy
1	80011008		ACCOUNT	1119035648547	siebeladmin	<NULL>	<NULL>
2	29006120		ACCOUNT	1122252381160	siebeladmin	<NULL>	<NULL>
*							

Fig. 2 – sample rows in edx_bsl_amf_baccount table.

EDX_HIER_HNODE

For each row inserted to EDX_BSL_AMF_BACCOUNT table, there should be corresponding entry in this table. The format is as follows:

```

NODEID = [sequence]
PARENTID = [NODEID in the very first row - root node]
HIERARCHYID = [EDX_HIER_HIERARCHY.ID]
LINKTARGETURI = "edx:amf:billingaccount:<<account #>>"
LINKTARGETID = "<<account #>>"
ENTITYSCONTAINER = 1
PATH = [root node's NODEID.current row NODEID]
LINKTARGETNAME = "<<account #>>"
CREATEDAT = timestamp (same as in EDX_HIER_HIERARCHY)
CREATEDBY = "siebel admin"
MODIFIEDAT = timestamp
MODIFIEDBY = "siebel admin"
DELETEDAT = 0
LINKTARGETTYPE = "ACCOUNT"

```

For example, after adding two accounts, EDX_HIER_HNODE looks like:

Account Data Migration

nodeID	parentID	hierarchyID	linkTargetURI	linkTargetID	entit	path	linkTargetName	alias	description
1	<NULL>	1	edx:hierarchy:folder:31b8b6086207	root	1	1	Billing Accounts	<NULL>	<NULL>
2	1	1	edx:amf:billingaccount:80011008	80011008	1	1.2	80011008	<NULL>	<NULL>
3	1	1	edx:amf:billingaccount:29006120	29006120	1	1.3	29006120	<NULL>	<NULL>
*									

continued...

entit	path	linkTargetName	alias	description	createdAt	createdBy	modifiedAt	modifiedBy	deletedAt	linkTargetType
1	1	Billing Accounts	<NULL>	<NULL>	1119035648172	siebeladmin	1119035648172	siebeladmin	0	com.edocs.common.hierarchy.api.IHierarchyFolder
1	1.2	80011008	<NULL>	<NULL>	1119035648813	siebeladmin	1119035648813	siebeladmin	0	ACCOUNT
1	1.3	29006120	<NULL>	<NULL>	1122252381300	siebeladmin	1122252381332	siebeladmin	0	ACCOUNT

Fig. 3 – Sample rows in edx_hier_hnode table.

NOTE:

- All unique keys [*sequence*] should be generated using "*hibernate_sequence*" sequence.
- Need to write errors to a log file. Shows the execution summary at the end of the log file:

How many records inserted successfully.

How many records failed to insert.

ASSUMPTION:

1. PS team agreed to populate all other Billing Manager tables by a PS effort, for example, migrating enrolled users, mapping enrolled user ids to correct nodeid in HNODE table.