*e*Index Global Identifier Product Suite*

e*Index™ Global Identifier Upgrade Guide

Release 5.0.5 for Schema Run-time Environment (SRE)

SEEBEYOND

Version 20050429025612

Table of Contents

Chapter 1

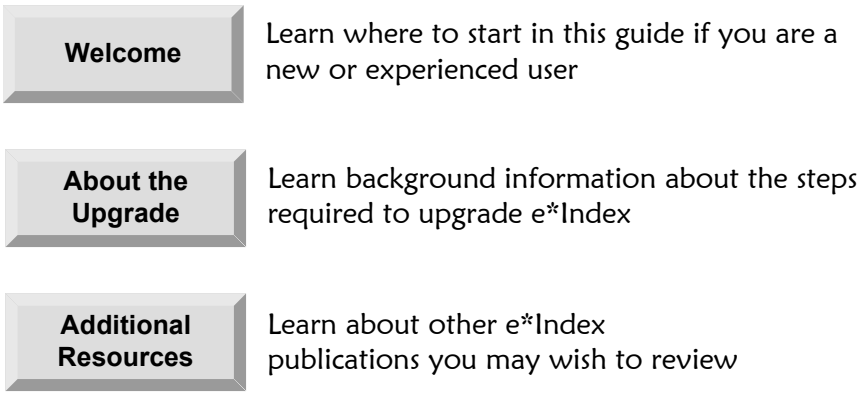
Introduction

About this Chapter

Overview

This introduction provides an overview of the steps you need to follow to upgrade the e*Index GUI, the Schema files, active integration API, and reports from version 4.5.3 to 5.0.5 for SRE. The database structure between versions 4.5.4 and 5.0.5 for SRE are identical, so no changes are required for the database.

The following diagram illustrates the contents of each major topic in this chapter. For the page numbers on which specific topics appear, see "What's Inside" on the following page.



What's Inside

This chapter provides background information and instructions related to the topics listed below.

Introduction

Welcome

Welcome to e*Index, SeeBeyond's enterprise-wide master person index. This document explains how to upgrade e*Index components to version 5.0.5 for SRE on both the client and server machines. This guide only describes how to upgrade from an e*Index 4.5.3 environment. If you currently have a previous version of e*Index, see the *e*Index 4.1.1 to 4.5.3 Upgrade Guide*.

This chapter of the document provides background information you should know before upgrading e*Index. Whether you are a new or established user, you should read through this guide before you begin the upgrade. Please pay particular attention to the overview sections provided at the beginning of each chapter and at the beginning of each section within a chapter. The overview sections are designed to provide background and explanatory information you may need to understand. After reading the overview information, you will be ready to perform specific tasks using the step-by-step instructions provided in each chapter.

What is the Scope of this Guide?

This guide provides step-by-step instructions for upgrading all of the components of e*Index to version 5.0.5 for SRE, including the e*Index Schema files, reports, active integration API, and GUIs. It includes navigational information, functional instructions, and background information where required. This guide also provides a description of the database table.

This guide does not include information or instructions on using any of the e*Index applications. These topics are covered in the appropriate user guide (for more information, see "Additional Resources" on page 1-10).

Who Should Use this Guide?

Any user who upgrades any component of e*Index should read this guide. A thorough knowledge of e*Index is not needed to understand this guide. It is presumed that the reader of this guide is familiar with the e*Gate environment, e*Gate Schemas, and the UNIX environment (if applicable).

How Should this Guide be Used?

For best results, skim through the guide to familiarize yourself with the locations of essential procedures you need to perform. Each chapter begins

with a simple graphic that identifies the information contained in the chapter. The second page of each chapter contains a list of topics and instructions included in the chapter and the associated page numbers.

How is this Guide Organized?

This guide is divided into four chapters that cover the topics shown below.

Chapter	Topics
Chapter 1, Introduction	■ Introduction ■ About the Upgrade Procedure ■ Additional Resources
Chapter 2, Upgrading the e*Index Schema Files	■ Learning About the Upgrade Process ■ Performing the Upgrade
Chapter 3, Upgrading the GUIs and Publications	■ Learning About Upgrade Tasks ■ Performing the Upgrade
Chapter 4, Upgrading the Reports	■ Learning About the Report Upgrades ■ Performing the Upgrade

What Conventions are Used in this Guide?

Before you start using this guide, it is important to understand the icon, special notation, and mouse conventions used.

Icon and Special Notation Conventions

The following conventions are used in this and other e*Index publications to identify special types of information.

Icon or Notation	Type of information
Note	Supplemental information that is helpful to know, but not essential to completing a particular task.
Tip	Information that helps you to apply techniques and procedures described in the text to your specific needs. May also suggest alternative methods.
Important!	Information that is essential to the completion of a task.
Caution!	Advises you to take specific action to avoid loss of data.
▶	Indicates the beginning of a step-by-step instruction.

Icon or Notation	Type of information
✓	Specifies a task to perform before you begin a step-by-step instruction.
	Indicates a cross-reference to other sections of the guide or to other publications.

Mouse Conventions

You can use either a single-button mouse or a multiple-button mouse with e*Index. If you use a multiple-button mouse, the left mouse button is the primary button, unless the mouse is configured differently.

The instructions in this guide may require you to use the mouse in a variety of ways:

- **Point** means to position the mouse pointer until the tip of the pointer rests on whatever you want to point to on the screen.
- **Click** means to press and then immediately release the left mouse button without moving the mouse.
- **Double-click** means to click the left mouse button twice, in rapid succession.
- **Right-click** means to click the right mouse button once.
- **Drag** means to point and then hold down the mouse button as you move the mouse. **Drop** means to let go of the mouse button to place the dragged information where you want it to be moved.
- **Move** means to point to an object on the screen, such as an e*Index Security user group, and drag the mouse to move the object to a screen location of your choice.
- **Highlight** means to select an area of text by dragging the mouse over the desired portion of text that appears on a window.
- **Select** means to point to a list of information on an e*Index window, and then click once to choose the data you want. The information becomes highlighted when selected.
- **Expand** means to double-click a row of information on an expandable list to display more details. The details appear on another row, below the row you double-click.
- **Collapse** means to double-click a row of information on an expandable list to hide the details that appear on the following row.

About the Upgrade Procedure

Overview

This section of the chapter outlines the requirements for e*Index, and summarizes the steps you need to follow to upgrade to e*Index 5.0.5 for SRE, starting with the e*Gate environment and ending with the Java APIs for e*Index Active Integration.

Requirements

The requirements for previous versions of e*Index and e*Index 5.0.5 for SRE are slightly different. You may need to upgrade third-party software in order to complete the upgrade. You must perform the upgrade on a computer running Windows XP, 2000, or 2003. Prior to beginning the upgrade, verify that you have the appropriate software installed.

e*Gate™ Integrator Requirements

If you are not currently running the following versions of e*Gate and the Database e*Way™, you need to upgrade them before performing the e*Index upgrade.

- e*Gate Integrator for SRE
- The Database e*Way for SRE appropriate to the database platform you are using:
  - For an Oracle database, install the Oracle e*Way
  - For a Sybase database, install the Sybase e*Way
  - For a SQL Server database, install the ODBC e*Way
- HL7 Templates for SRE (only if you are transmitting HL7 messages)
- e-Mail e*Way for SRE (if you plan to use the Event Notification capability of e*Index Security)

Database Platform Requirements

Database platforms requirements are identical between e*Index 4.5.3 and e*Index 5.0.5 for SRE. No changes are required to your 4.5.3 database for this upgrade.

Schema Platforms

The e*Index Schema files can be installed on any of the following platforms:

- Windows XP, 2000, or 2003 with required patches
- HP Tru64 V5.1A patch 5, or V5.1B
- HPUX 11 or 11i (PA-RISC), with required patches
- IBM AIX 5.1L and 5.2 with required Maintenance level patches

---

Important! *If you are using Oracle for the database platform and AIX 5.1 for the operating system on the e*Gate server where the e*Index Schema is installed, you need to use the 32-bit version of AIX 5.1.*

- Sun Solaris 8 and 9 with required patches

Java APIs for Active Integration Requirements

For information about requirements for a new installation of the Java APIs, see chapter 7 of the *e*Index Global Identifier Installation Guide*. If you are upgrading a current 4.5.3 version of the API, there are no requirements changes for 5.0.5 for SRE.

Upgrade Overview

Most of the setup for e*Index is performed using a standard InstallShield® Wizard with specific customizations for each component. Each step outlined below describes how to upgrade one component, and is described in detail in its own chapter in this guide. Once you have installed the files for a specific step, some modifications may be required. All required modifications are described in the appropriate chapter.

For clarity, this document describes upgrading one component at a time. To save time, you can install the upgrade files for all components at one time on a client machine, but this will require some extra steps. Figure 1-1 on page 1-9 illustrates the steps you need to follow to install all components at one time.

- Step 1: Upgrade the e*Index Schema Files (described in Chapter 2)
- Step 2: Upgrade the GUI and Publications (described in Chapter 3)
- Step 3: Upgrade the Reports (Chapter 4)
- Step 4: Upgrade the Java APIs for e*Index Active Integration (optional)

Step 1: Upgrade the e*Index Schema Files

Chapter 2, "Upgrading the e*Index Schema Files", outlines the steps required to install the new e*Index Schema files into the e*Gate environment. This chapter discusses file structure and the modifications that may be required to the e*Way configuration file, backend libraries, and Monk files.

Step 2: Upgrade the GUIs and Publications

Chapter 2, "Upgrading the GUIs and Publications," describes the steps you need to follow to upgrade the e*Index GUIs. This chapter includes hardware and software requirements for the upgrade. You can also update the e*Index Electronic Library. This library includes the suite of e*Index documentation and a Welcome document to help you navigate through the files.

Step 3: Upgrade the Reports

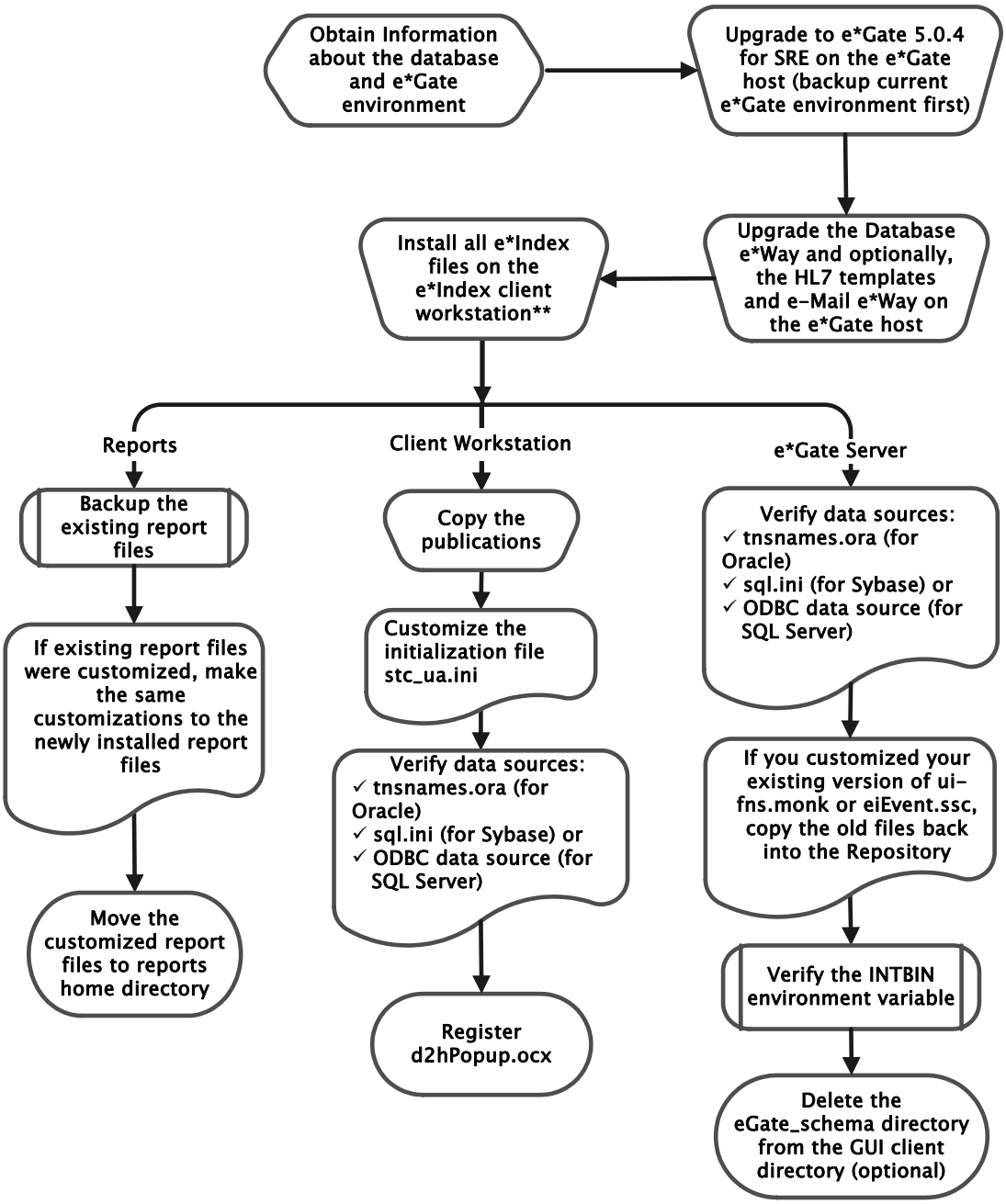
Chapter 3, "Upgrading the Reports," describes the steps you need to follow to upgrade the e*Index reports. The only modification to the reports is the removal of UIDs that have been deactivated from the potential duplicate reports. This step is optional.

Step 4: Install the Java APIs for e*Index Active Integration

The e*Index 5.0.5 for SRE package also includes Java APIs for e*Index Active Integration. If you have installed ESR 49653, which fixed an error that occurred during searches by the API, you do not need to upgrade the API. If you do need to upgrade your API, SeeBeyond recommends installing ESR 49653 rather than installing the files from the 5.0.5 for SRE package. If you do not have a current version of the API and would like to install it, see chapter 7 of the *e*Index Global Identifier Installation Guide*.

Figure 1-1: Upgrade Overview

Upgrading all components at one time. This diagram does not include installing the Java APIs.



** If you customized the file named **ui-fns.monk** in the e*Index schema, make backup copies of the customized file before installing the upgrade files. You also may want to back up the GUI initialization file **stc_ua.ini**.

Additional Resources

SeeBeyond has developed a suite of e*Index user's guides and related publications that are distributed in an electronic library.

- *e*Index Global Identifier User's Guide*
  Helps e*Index Quality Workstation users to perform database maintenance tasks, such as merging and unmerging records, finding and resolving potential duplicates, adding and updating records, and viewing the audit trail.
- *e*Index Administrator User's Guide*
  Helps system administrators configure system parameters, customize e*Index, work with Vality rule set files, and processing codes. This guide also describes how to maintain the information in the database that is used to populate the drop-down lists in the e*Index.
- *e*Index Security User's Guide*
  Helps system administrators add users and user groups to e*Index, to grant security permissions to users and user groups, to maintain user and user group information, and to configure certain system parameters.
- *e*Index Global Identifier Technical Reference*
  Describes message processing for e*Index, as well as database tables and e*Index Monk APIs. This guide also provides a complete listing of e*Index Monk APIs and functions, along with a description, parameters, syntax, return values, and examples for each.
- *e*Index Initial Load User's Guide*
  Provides the background information and instructions that system and database administrators need in order to load legacy data into the e*Index database, including a description of the expected data format and the Schema files included with the load program.
- *Working with Reports for e*Index Global Identifier*
  Provides background information about the GUI and standard reports provided with e*Index, and explains how to modify and run the standard reports (for an Oracle installation only).
- *e*Index Global Identifier Installation Guide*
  Helps system and database administrators install a new e*Index environment for the current release, including e*Index Schema files, the e*Index GUI, and database installation.
- *Java Programmer's Guide for e*Index Active Integration*
  Provides background and implementation information about the Java APIs for e*Index Active Integration. This guide also provides a complete listing of e*Index Java functions, along with a description, parameters, syntax, return values, and examples for each.

Chapter 2

Upgrading the e*Index Schema Files

About this Chapter

Overview

This chapter presents the background information and the step-by-step instructions you need to upgrade the e*Index Schema files for e*Gate to from e*Index 4.5.3 to 5.0.5 for SRE.

The following diagram illustrates the contents of each major topic in this chapter. For the page numbers on which specific topics appear, see the next page of this chapter.

About the e*Index Schema — Learn about the upgrade process, the requirements, and the files that were updated

Upgrade the e*Index Schema — Learn how to upgrade the e*Index Schema files in your e*Gate environment

What's Inside

This chapter provides background information and instructions related to the topics listed below.

Learning About the e*Index Schema

Overview

This section of the chapter provides background information about upgrading the e*Index Schema files for e*Gate.

What Do I Need to Get Started?

Upgrading to e*Index version 5.0.5 for SRE may require some modifications to your e*Gate and Database e*Way environments. We highly recommend that a separate e*Gate and Database e*Way environment be created and tested before any modifications are made to your current working environment. Before you start you, need to have the following software from SeeBeyond for the platform with which you are working.

SeeBeyond Software

- e*Gate for SRE
- The appropriate Database e*Way for SRE
  - For an Oracle database, you need the Oracle e*Way
  - For a Sybase database, you need the Sybase e*Way
  - For a Microsoft SQL Server database, you need the ODBC e*Way
- HL7 Templates for SRE (only if you process HL7 messages)
- e-Mail e*Way for SRE (only if you will use the Event Notification function of e*Index Security)
- e*Index 5.0.5 for SRE

Database Software

One of the following database clients should already be installed on the e*Gate server. For upgrading, you should use the same database vendor as in previous versions.

- Oracle Client version 8.1.7 (8.1.7.2.1 is recommended) or Oracle Client 9*i*
- Sybase Client version 12.0
- Microsoft SQL Server Enterprise Edition 7.0 (only client files are required)

Operating System Software

The e*Gate Schema can be run on any of the following operating systems:

- Windows XP, 2000, or 2003 with required patches

- HP Tru64 V5.1A patch 5, or V5.1B
- HPUX 11 or 11i (PA-RISC), with required patches
- IBM AIX 5.1L and 5.2 with required Maintenance level patches

Important! *If you are using Oracle for the database platform and AIX 5.1 for the operating system on the e*Gate server where the e*Index Schema is installed, you need to use the 32-bit version of AIX 5.1.*

- Sun Solaris 8 and 9 with required patches

Is the Upgrade Process Standard Across Platforms?

The upgrade procedures for Windows and UNIX differ slightly, but both procedures begin on a PC running Windows. The extensions for many of the dynamic link library files that are installed during this process may vary depending on the platform you are running. For Windows the extension is **.dll**. Usually, for HP UNIX, the extension is **.sl**, for Solaris and TRU64 the extension is **.so**, and for AIX the extension is **.a**. However, most e*Gate and e*Index UNIX shared libraries have the extension **.dll**. This will not affect runtime behavior. The variable **<eGate>** is used to specify your e*Gate environment on any platform.

Does the Upgrade Replace Existing e*Index Schemas?

Instead of replacing your existing e*Index Schemas when you perform the upgrade, you only need to upgrade the binary files that have changed for this release, along with some processing files. The files are installed into the **/<eGate>/server/registry/repository/default** directory in the e*Gate home environment instead of into any Schema directories you have created. This ensures that your customizations are not overwritten and that you are getting the most current Schema files for e*Index. Two exceptions are the files **ui-fns.monk** and **eiEvent.ssc**. If you have previously customized these files, you may need to recustomize them after the upgrade.

Performing the Upgrade

Overview

To upgrade the Schema for e*Index, you must complete the following steps:

- Step 1: Back up the e*Gate Environment
- Step 2: Verify the Database Client Software
- Step 3: Upgrade the e*Gate Environment
- Step 4: Upgrade the Database e*Way
- Step 5: Install or Upgrade the e-Mail e*Way (optional)
- Step 6: Install the e*Index Schema Files
- Step 7: Set up the Environment
- Step 8: Update Schema Files

Step 1: Back up the e*Gate Environment

If you are already running e*Gate, it is important to make a FULL backup of the environment for safekeeping before making any changes to your e*Gate environment. In the e*Index Schema components (this includes the files installed in the **default** Schema), the **ui-fns.monk** and **eiEvent.ssc** file will be replaced during the upgrade. If you have customized those existing files, back them up so you can easily re-customize the files.

Step 2: Verify the Database Software

You should already have the correct client database software installed on the e*Gate server. Verify one of the following before continuing.

- For an Oracle database, make sure Oracle Client 8.1.7 or 9*i* is installed. Verify that **tnsnames.ora** includes a stanza for the e*Index database (for more information, see "Step 6: For Oracle Only, Verify **tnsnames.ora**" in Chapter 3 of this guide).
- For a Sybase database, make sure Sybase Client 12.0 is installed. Verify that **sql.ini** has a stanza for the e*Index database server (for more information, see "Step 7: For Sybase Only, Verify **sql.ini**" in Chapter 3 of this guide).
- For a Microsoft SQL Server database, install Microsoft SQL Server Enterprise Edition 7.0 (only the client files are required). Verify that an ODBC data source is defined for the database (see "Step 8: For SQL Server Only, Verify the ODBC Data Source" in chapter 3 of this guide for more information).

For information about installing the database software, refer to the appropriate Oracle, Sybase, or Microsoft SQL Server documentation.

Step 3: Upgrade the e*Gate Environment

Before you perform the upgrade, you must upgrade the e*Gate environment to version 5.0.5 for SRE. For more information, see the *e*Gate Integrator Installation Guide*. If you are not currently using e*Gate 5.0.5 for SRE, install it on a separate environment for testing, away from your production environment. If you are processing HL7 messages, make sure you also upgrade the HL7 template libraries for e*Gate.

Step 4: Upgrade the Database e*Way

Before you perform the upgrade, you must upgrade the Database e*Way you are using (Oracle, Sybase, or ODBC). To upgrade the Database e*Way, refer to refer to chapter 2 of the *e*Way Intelligent Adapter for Oracle User's Guide*, *e*Way Intelligent Adapter for Sybase User's Guide*, or the *e*Way Intelligent Adapter for ODBC User's Guide*.

Step 5: Install or Upgrade the e-Mail e*Way

You need to install or upgrade the e-Mail e*Way only if you will be using the Event Notification function in e*Index Security (for more information, see chapter 3 of the *e*Index Security User's Guide*). For information on installing and implementing the e-Mail e*Way, refer to the *e-Mail e*Way Intelligent Adapter User's Guide*. If you are currently running an earlier version of the e-Mail e*Way than 5.0.5 for SRE, you must upgrade the e*Way.

Step 6: Install the e*Index Schema Files

To begin the installation process, insert the e*Index 5.0.5 installation CD-ROM into the CD-ROM drive on your computer, and make sure no other Windows applications are running.

Important! *If you choose to upgrade all standard components of e*Index (e*Index Schema files, database, reports, GUI, and documentation) at this time:*

- *Install the files on your Quality Workstation in the directory where you want your GUI files to reside.*
- *Review the diagram on page 1-9 in Chapter 1 of this guide.*

To install the e*Index Schema files

Before you begin:

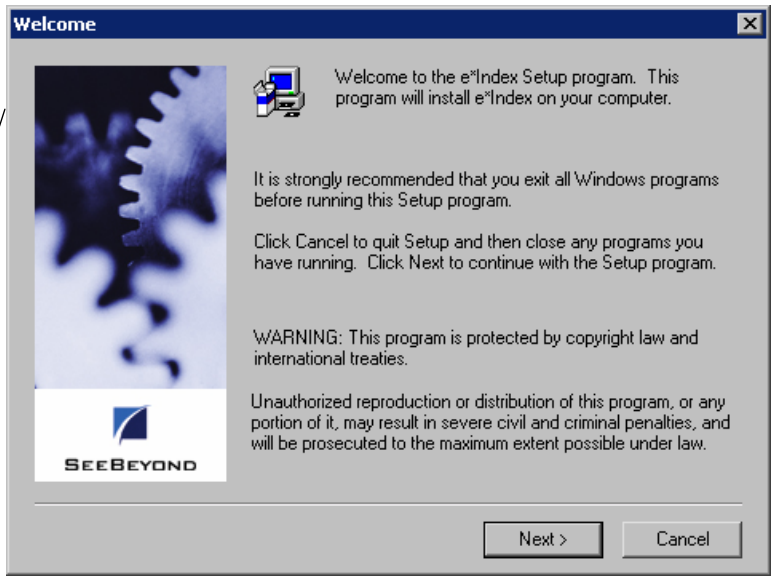
- ✓ Complete "Step 1: Backup your e*Gate Environment" through "Step 5: Install or Upgrade the e-Mail e*Way"
- ✓ Make sure that all Windows applications are closed

1 Insert the e*Index installation CD-ROM into the CD-ROM drive of your computer

2 If Autorun is enabled, the setup program automatically starts. Otherwise:

- On the Windows desktop, double-click the **My Computer** icon and then open the CD-ROM directory.
- Double-click the file name **Setup.exe** to initiate the process that installs the e*Index Schema files. The Welcome window appears, reminding you to close all Windows programs.

e*Index is installed using a standard InstallShield Wizard



3 Do one of the following:

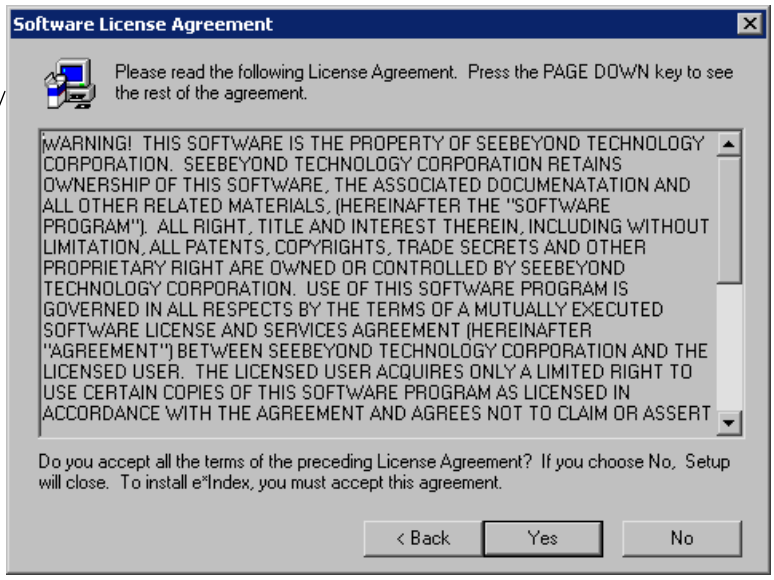
Cancel button

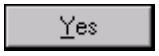
To close any open Windows programs, click **Cancel**, close the programs, and then repeat step 2.

Next button

To continue with the upgrade process without closing any external programs, click **Next**. The Software License Agreement window appears.

You must agree to the terms of the license agreement in order to proceed

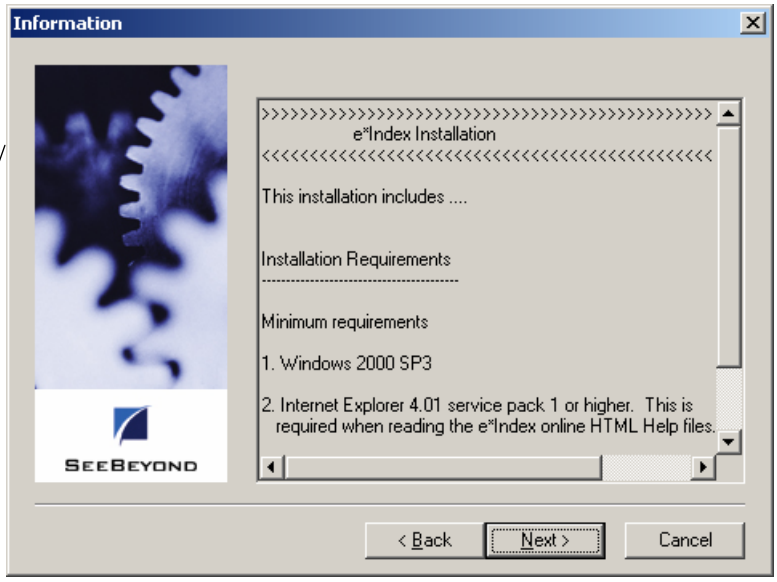




Yes button

4 If you agree to the license agreement, click **Yes**. The Information window appears with a list of requirements to remind you of the applications you need before upgrading e*Index.

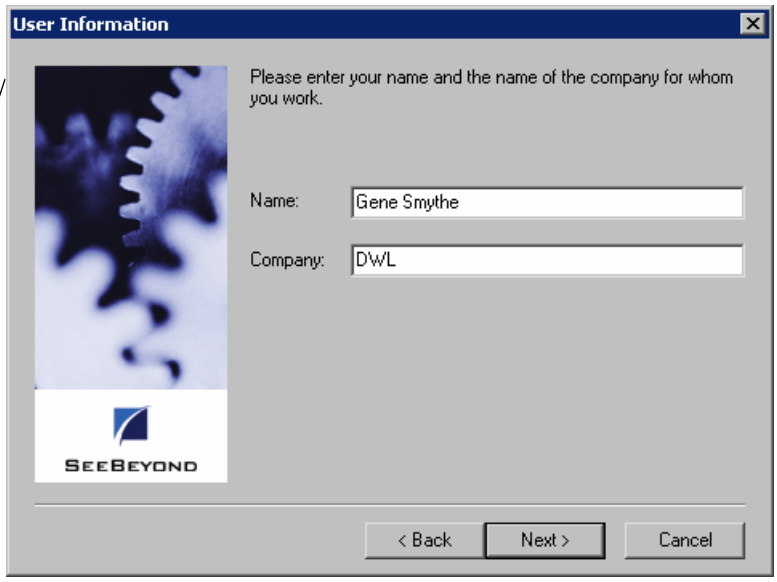
The Information window reminds you to have Windows, Internet Explorer, and database software installed



Next button

5 Click **Next**. The User Information window appears.

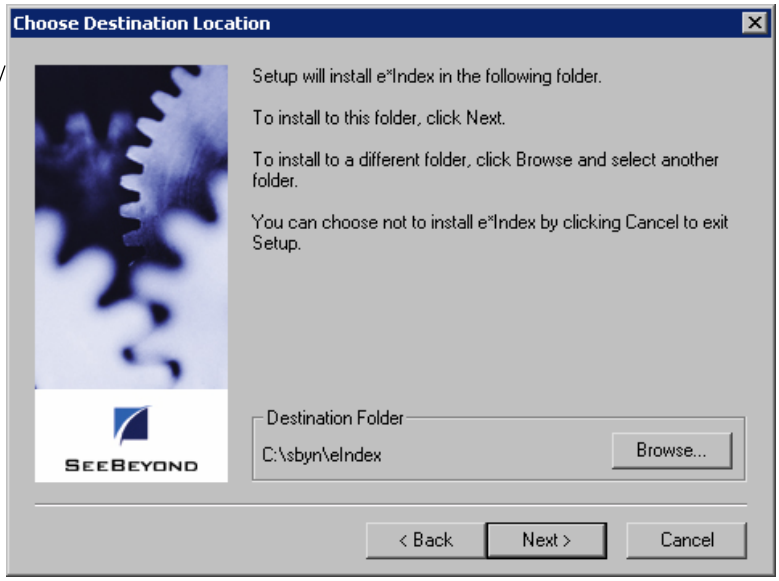
Enter your name and your company name on the User Information window



Next button

6 In the **Name** and **Company** fields, enter your name and your company's name, and then click **Next**. The Choose Destination Location window appears.

On the Choose Destination Location window, specify your e*Gate client path



7 Do one of the following:

Next button

To accept the default folder that appears in the Destination Folder path, click **Next**.

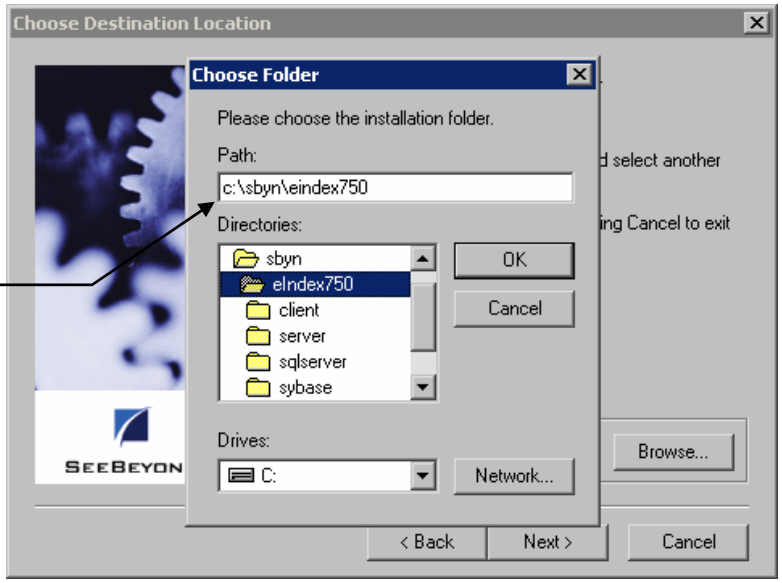
Browse button

To change the location in which the files will be installed:

- Click **Browse**.
- On the Choose Folder dialog, type or select the path where you want to install the files.

- Click **OK**. The new path you specified appears in the Destination Folder path.

Specify a new path on the Choose Folder dialog

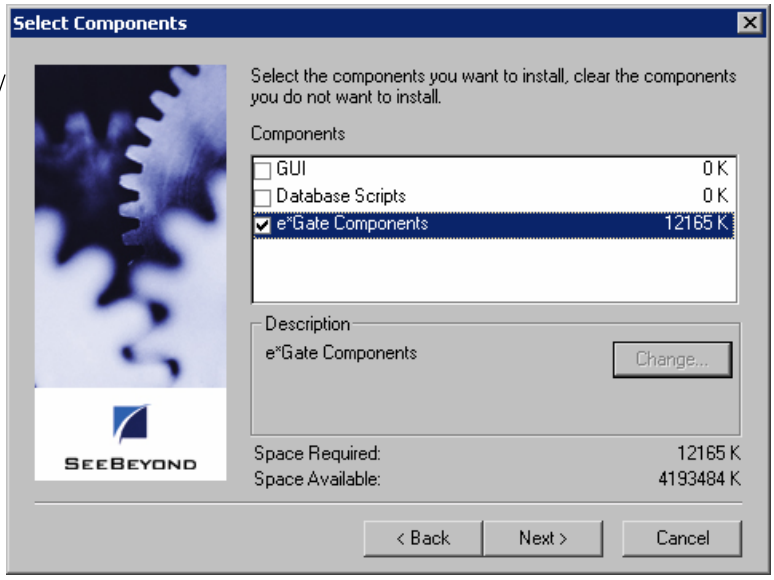


Note: *If the path you specified does not exist, a dialog appears asking if you want to have the folder created. If you select* ***Yes****, Setup creates the specified path for you.*

Next button

8 After you specify the installation path, click **Next**. The Select Components window appears.

Select the components you want to install on the Select Components window



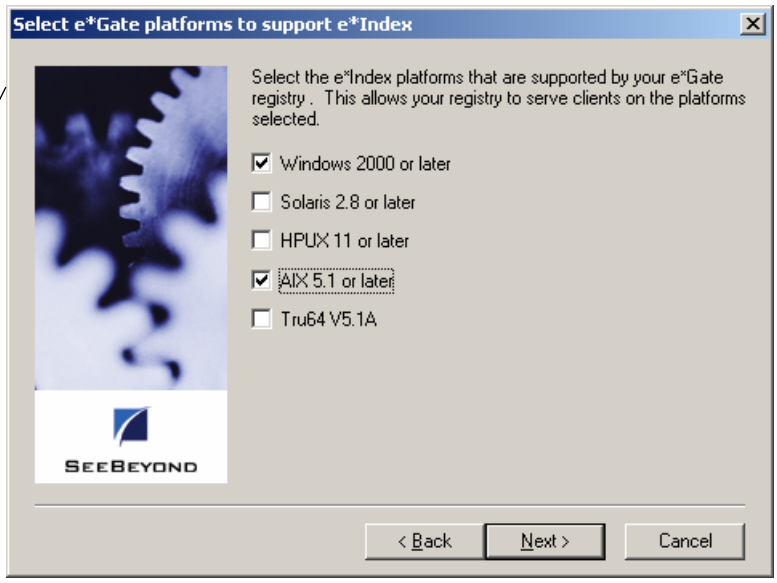
9 In the Components box, select **e*Gate Components**, and make sure that no other components are selected.

Notes:

- *At the bottom of the window, the required space and the available space appears so you can see how much space each component requires compared to the space you have available on your machine.*
- *This step only describes how to install the e*Index Schema upgrade files. To save time, you can install all of the upgrade components of e*Index at one time by selecting every component on the Select Components window. Then, refer to the appropriate chapter to modify the files you installed. For more information, refer to the notes at the beginning of this procedure.*

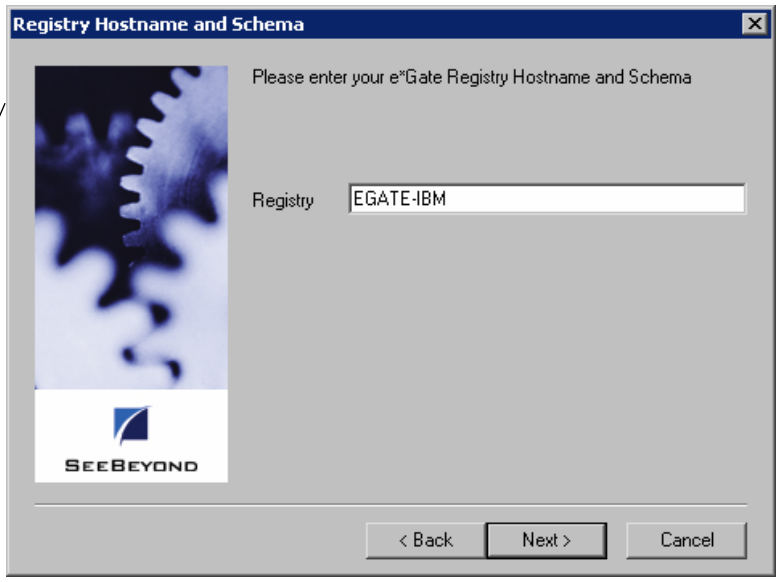
10 On the Select Components window, click **Next**. The Select e*Gate Platforms window appears.

You can specify the platforms on which your e*Gate host servers are running



11 Select the platforms on which you will be running the e*Index Schemas, and then click **Next**. The Registry Hostname and Schema window appears.

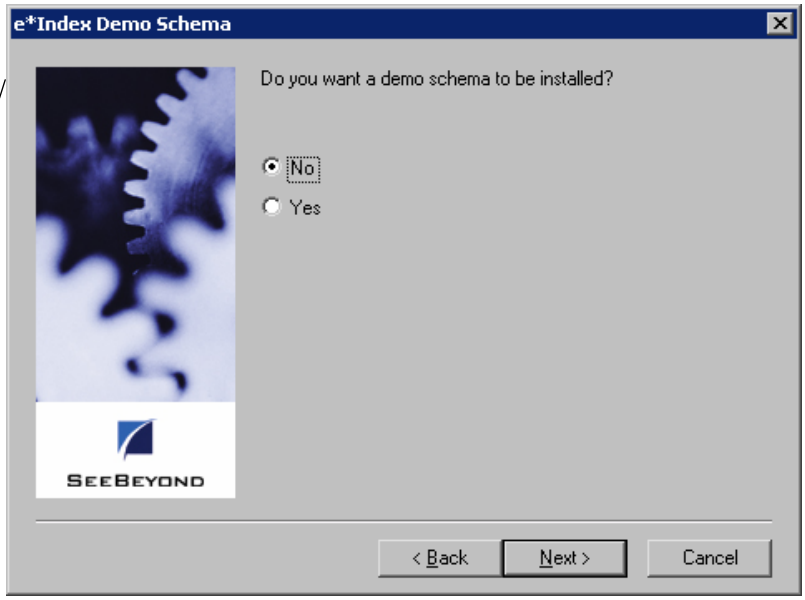
On the Registry Hostname and Schema window, specify information about your e*Gate environment



Next button

12 In the **Registry** field, enter the name of your e*Gate registry host, and then click **Next**. The e*Index Demo Schema window appears

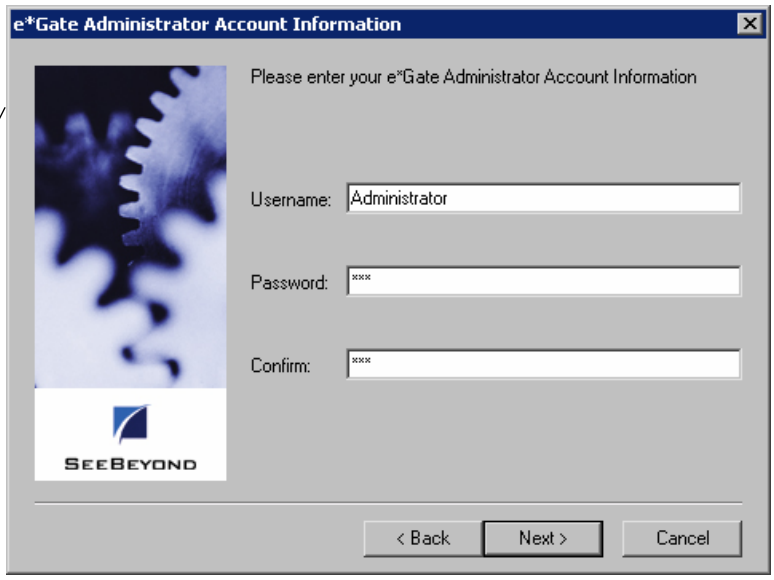
On the e*Index Demo Schema, specify whether you want to install a demo Schema



Next button

13 Select **No** and then click **Next**. The e*Index Administrator Account Information window appears.

Specify information about your e*Gate administrator account on the e*Gate Administrator Account Information window

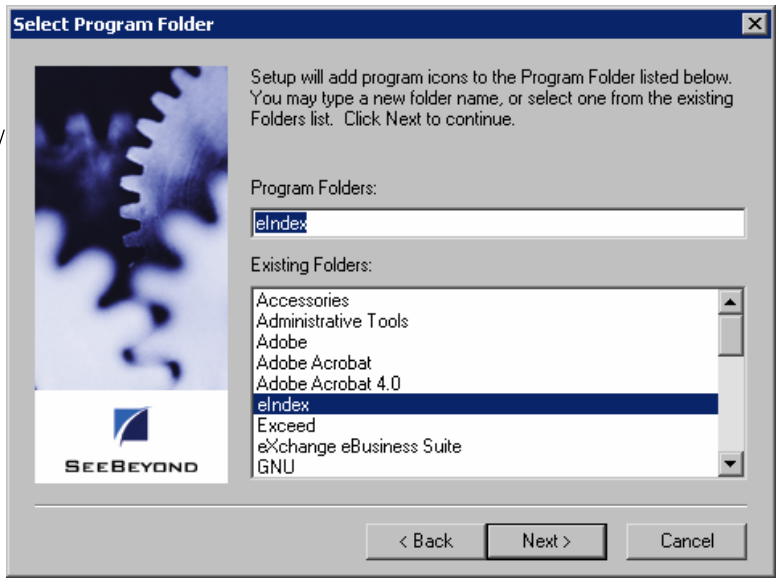


14 In the **Username** and **Password** fields, enter the administrator user ID and password for your e*Gate system. Re-enter the password as confirmation.

Next button

15 Click **Next**. The Select Program Folder window appears.

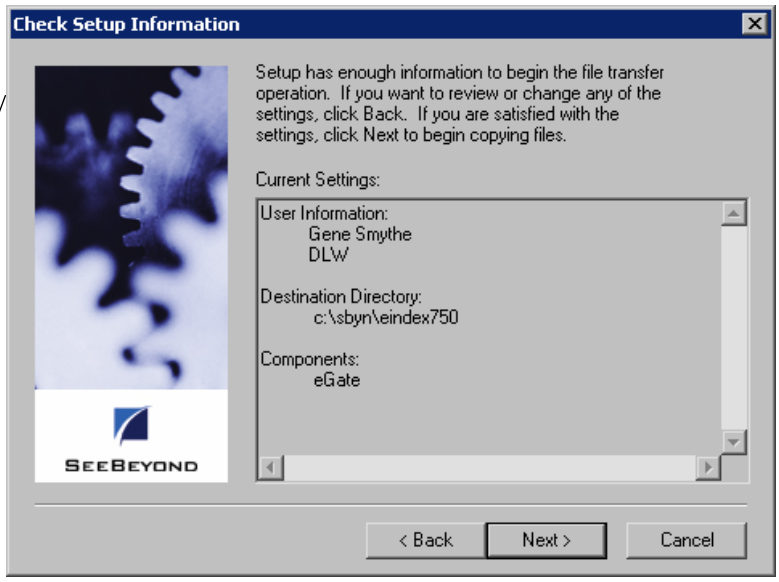
Select the program folder to which you want to add the e*Index icons



Next button

16 Enter the name of the program folder to which you want to add the e*Index icons or accept the default name, and then click **Next**. The Check Setup Information window appears.

Verify the components you are installing on the Check Setup Information window



17 Verify the information you specified, and do one of the following:

< Back
Back button

To change any of the options you selected, click **Back**, and make the necessary changes.

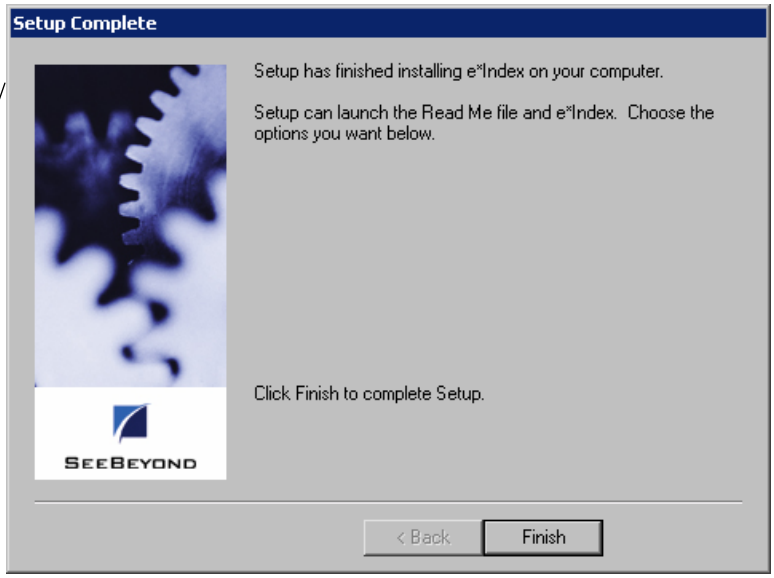
Next >
Next button

To install the files in the specified directory, click **Next**. The Setup Complete window appears after the files are installed and committed to the e*Gate registry.

Finish
Finish button

18 The Setup Complete window indicates that all necessary files are installed. Click **Finish** to conclude the setup process.

The Setup Complete window indicates that the files have been installed



19 Continue to "Step 6: Set up the Environment".

Step 7: Set up the Environment

Make sure the environment variable "INTBIN" points to the location of the code set files used by the Vality matching algorithm. By default, the **\codeset** directory is located in **\<eGate>\client\bin** and in **\<eGate>\Server\registry\repository\default\bin**. You only need to specify the **\<eGate>\client\bin** path for the INTBIN variable. If you plan to move the code set files, be sure to modify INTBIN accordingly.

Step 8: Update Schema Files

If you were using a customized copy of **ui-fns.monk** or **eiEvent.ssc**, copy your backed up version of the files back into the Repository. **ui-fns.monk** is located in **\<eGate>\Server\registry\repository\default\bin\monk_library\ui**. **eiEvent.ssc** is located in **<eGate>\Server\registry\repository\default\bin\monk_scripts\ui**.

Chapter 3

Upgrading an Oracle Database

About this Chapter

Overview

This chapter presents the background information and the step-by-step instructions you need to upgrade an e*Index Oracle database from version 4.5.1 or later to version 5.0.5 for SRE.

The following diagram illustrates the major topics in this chapter. For the page numbers on which specific topics appear, see the next page of this chapter.

About the Upgrade	Learn about upgrading the database and the report files
Upgrade the Database	Learn how to install the database upgrade files, and to modify and run the database scripts
Physical Data Model	View a physical data model of the e*Index database

What's Inside

This chapter provides background information and instructions related to the topics listed below.

Learning About Upgrade Tasks

Overview

This section of the chapter provides background information about the files you install, modify, and execute to upgrade to version 5.0.5 for SRE from an existing e*Index Oracle database.

What do I Need to Know Before I Start?

Make sure you are familiar with your Oracle environment before beginning the upgrade procedure. Know the server name and pathname in which the e*Index database resides, and the Oracle SID name and password of the database you are upgrading. It is helpful to be familiar with your Oracle environment, Oracle networking, and Oracle database administration. Also, determine whether you have region-specific security installed, and, if not, whether you want to install it (for more information, see chapter 3 of the *e*Index Administrator User's Guide* and the *e*Index Security User's Guide*).

e*Index 5.0.5 for SRE is compatible with Oracle 8.1.7 and 9*i*. If you are using a previous version of Oracle, you need to upgrade your database. For Oracle 8.1.7, version 8.1.7.2.1 is recommended.

How is the Database Upgraded?

Installing the database files creates several SQL scripts used to upgrade the existing database; however, you only need to execute one batch file, **upgrade.bat**, to perform the upgrade. The upgrade batch file calls the SQL scripts to make the necessary changes to the database. This file requires some modification prior to execution. It must be run from a machine running a Windows operating system (XP, 2000, or 2003) with Oracle Client installed. You can run the upgrade file from an existing e*Index client workstation.

If you do not have region-specific security currently installed, but plan to use it in the future, you need to modify and run a second file, **install_ssap.bat**. Note that this process can only upgrade a database that is already at version 4.5.1 or later. If you are upgrading from an earlier version, see the appropriate e*Index upgrade guide or your SeeBeyond representative.

Do I Need to Upgrade Report Files?

For Oracle databases only, the monthly e*Index report files **monthly_c** and **monthly_p** were modified for version 4.5.2. They previously contained a date argument that prevented the reports from running correctly. For this release, that date argument was removed and the reports run as expected.

The excerpt below indicates the clause that was modified. The third line of this clause was removed.

```
ui_monthly_stat_proc(to_char(sysdate,'MONTH'),
        to_char(sysdate,'YYYY'));
        to_char(sysdate,'YYYYMM');
```

In addition, changes were made to the **rep_unmerge_t** and **rep_unmerge_y** report files. Previously these reports were printing duplicate entries of unmerge transactions. With the changes, only one instance of each transaction appears on the reports. If you are upgrading your e*Index environment from version 4.5.2 or earlier and are using these report files, you need to upgrade the report files (described on page 3-15).

Performing the Upgrade

Overview

To upgrade the Oracle database, complete the following steps:

- Step 1: Obtain Database Information
- Step 2: Back up the Current Database
- Step 3: Upgrade Oracle Server
- Step 4: Install the Upgrade Files
- Step 5: Verify **tnsnames.ora**
- Step 6: Modify the Upgrade File
- Step 7: Modify **install_ssap.bat** (optional)
- Step 8: Upgrade the Database
- Step 9: Run **install_ssap.bat** (optional)
- Step 10: Move the Report Files

Note: *If you chose to install all components of e*Index when you upgraded the e*Gate schema files, you should have already completed steps 1 through 4. You can begin with "Step 5: Verify* ***tnsnames.ora****".*

Step 1: Obtain Database Information

Before beginning, gather information about your database, such as the database path, the database name, and the system login and password for your database. You should also know the path to your Oracle home directory. If you customized the rule set files for the Vality matching algorithm, you should know the location of the most current rule set files. If you currently do not have region-specific security installed, determine whether you want to install it now. Region-specific security is described in Chapter 3 of the *e*Index Security User's Guide* and the *e*Index Administrator User's Guide*. Because of the complex nature of modifying a database, we recommend that a database administrator perform the following steps.

Step 2: Back up the Current Database

Prior to making any changes to your e*Index database, you should make a complete backup of your current database. For more information on performing Oracle database backups, see the appropriate Oracle documentation.

Step 3: Upgrade Oracle Server

Before you install the e*Index database files, Oracle Server must be installed on the database server. If you are currently running a version of Oracle earlier than 8.1.7, upgrade Oracle before continuing. If your Oracle home directory changes during the upgrade, make a note of the new path. You will need to specify this information later in the database upgrade files. If your database is installed in a Unix environment, you also need to have Oracle client installed on a client workstation in order to run the upgrade script. For more information, see "Step 2: Upgrade the Database Software" in chapter 6 of this guide. For information about upgrading Oracle Server, refer to the appropriate upgrade documentation for Oracle.

Important! *It is crucial that the Oracle Server and Client installation is operational prior to performing the following steps.*

Step 4: Install the Upgrade Files

Installing the database and report files is very similar to the process you followed to install the e*Gate schema files. Install the database upgrade files on a client workstation that has Oracle Client installed and that is running on a Windows XP, 2000, or 2003 operating system.

▶ To install the upgrade files

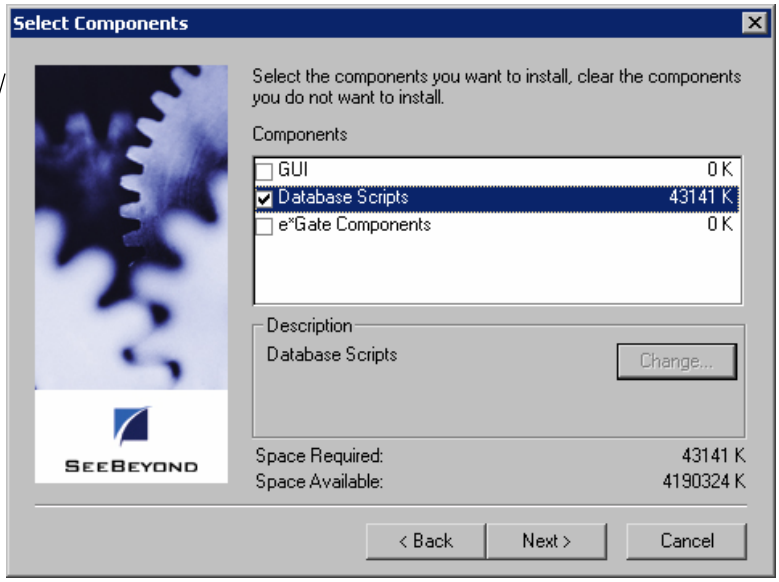
Before you begin:

- ✓ Complete "Step 3: Upgrade Oracle Server"
- ✓ Make sure the e*Index installation CD-ROM is inserted into the CD-ROM drive of a client workstation with Oracle Client installed

1. Follow steps 1 through 7 under "Step 5: Install the e*Index Schema Files" in Chapter 2, "Installing the e*Index Schema Files". The Select Components window should now be visible.

2. On the Select Components window, select the check box next to **Database Scripts**.

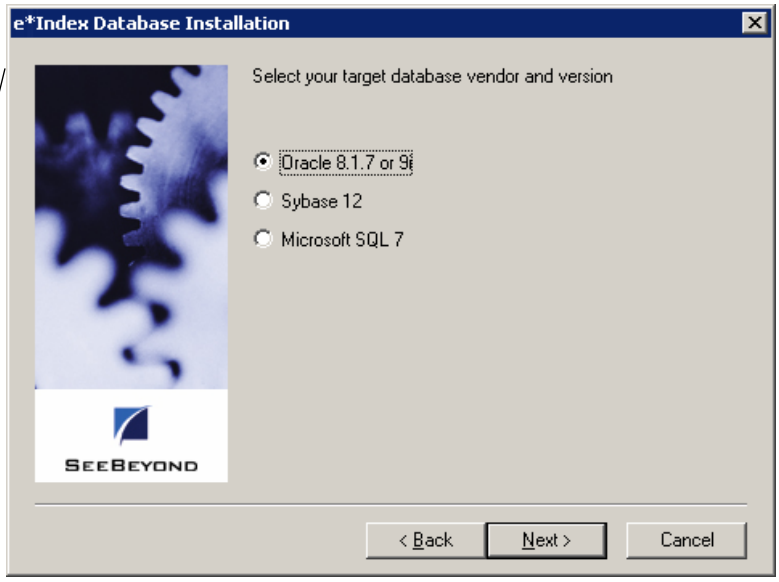
Select the components you want to install on the Select Components window



Next button

3 Click **Next**. The e*Index Database Installation window appears.

On the e*Index Database Installation window, specify your database platform

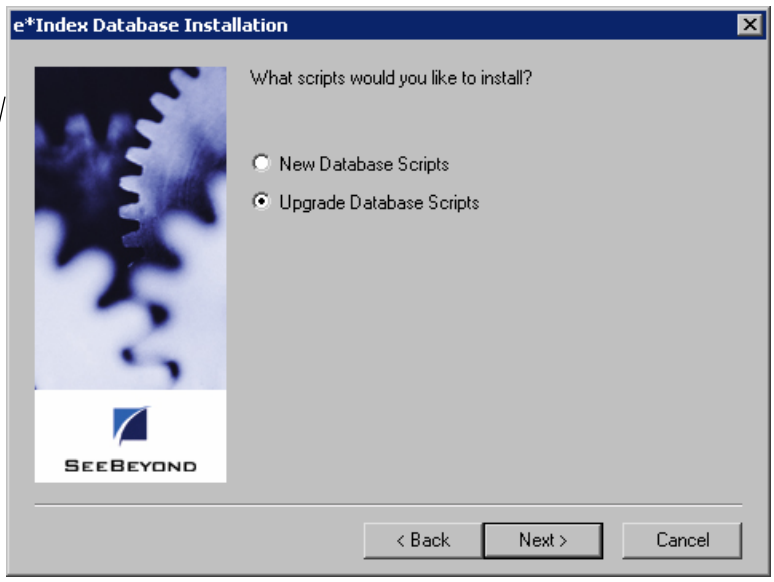


4 Select the name of the database platform you are using (Oracle).

Next button

5 Click **Next**. The second e*Index Database Installation window appears.

On the next e*Index Database Installation window, specify that this is a new installation of e*Index

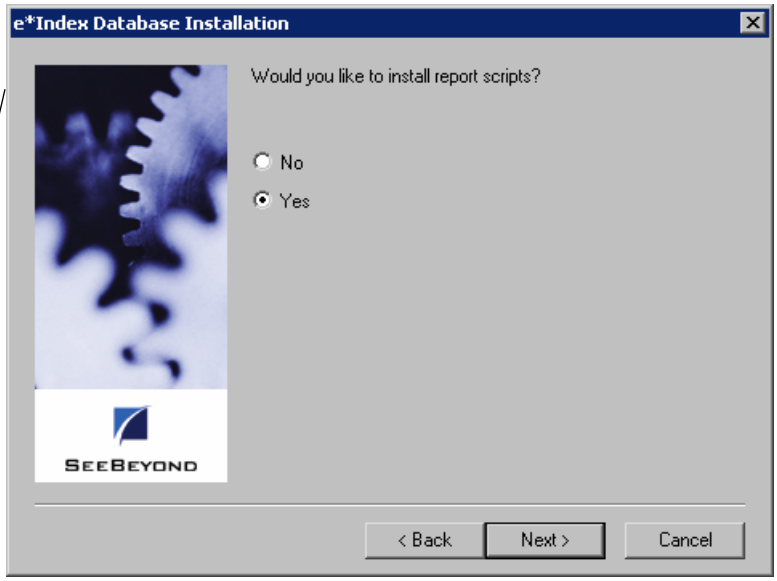


6 Select **Upgrade Database Scripts** to specify that you are upgrading an existing database.

Next button

7 Click **Next**. The final e*Index Database Installation window appears.

On the next e*Index Database Installation window, specify whether you want to install reports

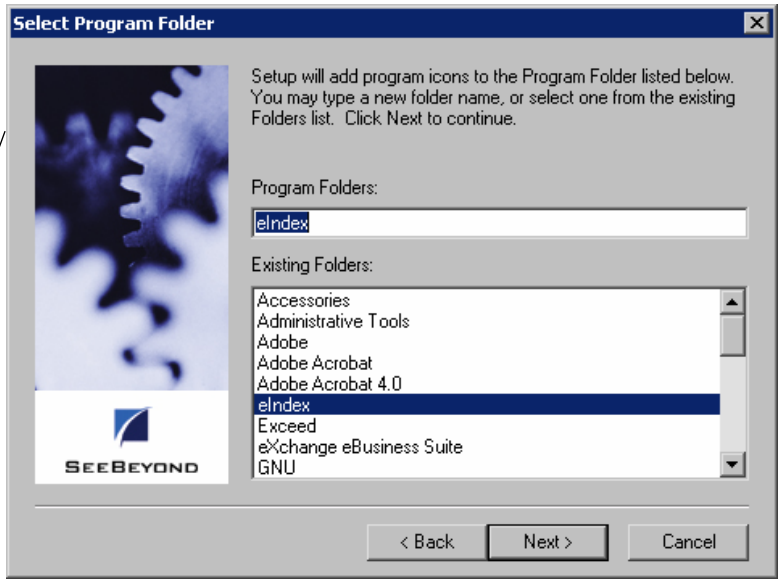


8 Select **Yes** to install the report files, or select **No** if you do not want to install the report files.

Next button

9 Click **Next**. The Select Program Folder window appears.

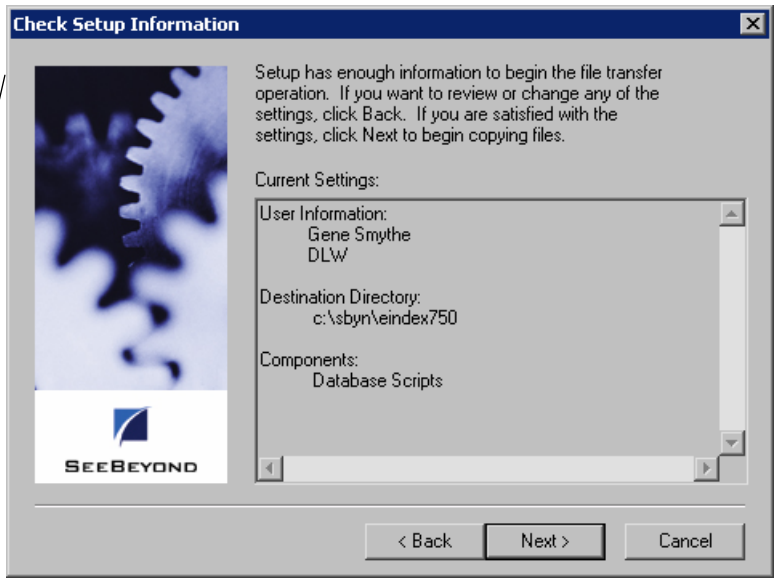
Specify the folder in which to install the program icons on the Select Program Folder window



Next button

10 Enter the name of the folder into which you want to install the program icons or accept the default name, and then click **Next**. The Check Setup Information window appears.

Verify the information you specified on the Check Setup Information window



11 Verify the information you specified, and do one of the following:

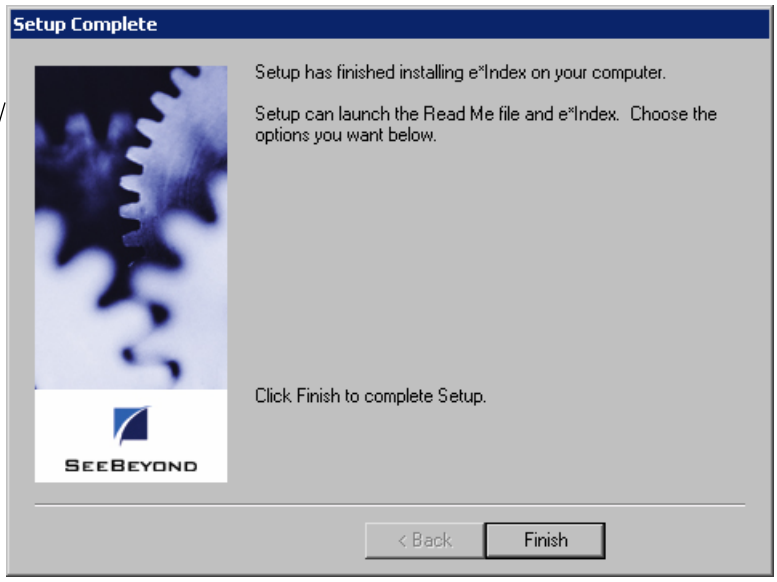
Back button

To change any of the options you selected, click **Back**, and make the necessary changes.

Next button

To continue with the installation, click **Next**. When all files are installed, the Setup Complete window appears.

The Setup Complete window indicates that the database files are installed



Finish button

12 Click **Finish** to complete the setup process.

13 To view the database files that were installed:

- Open Windows Explorer and navigate to the path you specified for the installation files.
- Open the **\server\DBcommon\Client** subdirectory. The Client directory contains the database scripts **upgrade.bat**, **install_ssap.bat**, and **remove_ssap.bat**. The **upgrades** directory contains the SQL files that are called by the upgrade batch file.

14 To view the report files that were installed:

- Open Windows Explorer and navigate to the path you specified for the installation files.
- Open the **\server\UIreports** subdirectory. This directory contains a subdirectory named **Production**, which contains all the report files.

15 Continue to "Step 5: Verify **tnsnames.ora**".

Step 5: Verify tnsnames.ora

Verify that the **tnsnames.ora** file on the computer you are using for the upgrade contains a stanza pointing to the database being upgraded. You should have a stanza for the e*Index database similar to the following example.

```
ei01.world =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (PROTOCOL = TCP)
          (Host = 100.0.0.00)
          (Port = 1000)
        )
    )
    (CONNECT_DATA = (SID = EI01)
    )
  )
```

*If you do not know how to modify the Oracle **tnsnames.ora** file, refer to the appropriate Oracle documentation. Your file may differ from the sample above*

Step 6: Modify the Upgrade File

The file **upgrade.bat** is a batch file that modifies the database to bring it up to version 5.0.5 for SRE. You need to modify certain variables in this file that tell the script how to locate the database instance and installation scripts.

▶ To modify the upgrade file

Before you begin:

- ✓ Complete "Step 5: Verify **tnsnames.ora**"

1. Navigate to the path where the database files are located, and then navigate to the **server\DBcommon\Client** directory.
2. Make a backup copy of the file **upgrade.bat**.
3. Right-click the **upgrade.bat** file and then select **Edit** from the list that appears. Do not double-click this file to open it.
4. Make the following modifications:
   - Enter the Oracle TNS service name of the database in the variable **TNS_NAME** (the TNS service name is defined in **tnsnames.ora**).

- In the variable **ORACLE_SID**, enter the SID name of the database.
- In the variable **INSTALLDRIVE**, enter the drive on which the database installation files are located.
- In the variable **INSTALL_HOME**, enter the path in which the database files are located. Do not include the drive designation in this path. This is the path to the **server** directory (do not include **/server/DBcommon/Server** in the pathname).

A sample of the above variables is illustrated below.

```
SET TNS_NAME=EIO1
SET ORACLE_SID=EIO1
SET INSTALLDRIVE=C:
SET INSTALL_HOME=/TEMP/INSTALL
```

5 If you have modified the system password and the e*Index "UI" user password, modify the values accordingly in the variables **SYSTEMPW** and **UIPW**.

```
SET SYSTEMPW=ORACLE
SET UIPW=eINDEX
```

6 Save the changes to **upgrade.bat** and close the file.

7 Do one of the following:

If you want to install region-specific security capabilities, continue to "Step 7: Modify **install_ssap.bat**".

If you are not installing region-specific security capabilities, skip to "Step 8: Upgrade the Database"

Step 7: Modify install_ssap.bat (optional)

Running **install_ssap.bat** installs the views you need in order to use the region-specific security capabilities of e*Index. You do not need to modify or run this file if you do not want to use this capability or if you already have region-specific security installed.

For more information about region-specific security, see "What is Region-Specific Security?" in Chapter 3 of the *e*Index Security User's Guide* and "About Region-Specific Security" in Chapter 3 of the *e*Index Administrator User's Guide*.

▶ To modify install_ssap.bat

Before you begin:

✓ Complete "Step 6: Modify the Upgrade File"

1 Navigate to the path where the database files are located, and then navigate to the **server\DBcommon\Client** directory.

2 Make a backup copy of the file **install_ssap.bat**.

3 Right-click the **install_ssap.bat** file and then select **Edit** from the list that appears. Do not double-click this file to open it.

4 Enter the Oracle TNS service name of the database in the variable **TNS_NAME** (the TNS service name is defined in **tnsnames.ora**). A sample of the **TNS_NAME** variable is illustrated below.

```
SET TNS_NAME=EIO1
```

5 If you have modified the system password and the e*Index "UI" user password, modify the values accordingly in the variables **SYSTEMPW** and **UIPW**.

```
SET SYSTEMPW=ORACLE
SET UIPW=eINDEX
```

6 Save the changes to **install_ssap.bat** and close the file.

7 Continue to "Step 8: Upgrade the Database".

Step 8: Upgrade the Database

After you have made the necessary modifications to the upgrade files, you can run **upgrade.bat** to create the tables, views, indexes, and so on for your e*Index database.

Important! *Make sure your database is at 4.5.1 or later before continuing. Otherwise, running the following step will create and populate new views, which could take several hours to complete.*

▶ To upgrade the database

Before you begin:

- ✓ Complete "Step 6: Modify the Upgrade Files" and optionally "Step 7: Modify **install_ssap.bat**"

1. Navigate to the path where the database files are located, and then navigate to the **server\DBcommon\Client** directory.
2. Double-click **upgrade.bat** to run the batch file.
3. You can view installation log files in the path installation path in **server\upgrades\Spool**. Review the log file to ensure that there were no errors during the running of the script.
4. Do one of the following:

   If you want to install region-specific security, continue to "Step 9: Run **install_ssap.bat**".

   If you are not installing region-specific security, skip to "Step 10: Move the Report Files".

Step 9: Run install_ssap.bat (optional)

After you have run **upgrade.bat** to modify the e*Index database, you can install system-specific security by running **install_ssap.bat**. If you are already using region-specific security, you can skip this step.

Important! *Do not execute this file if you do not want to use region-specific security. Creating these views requires that you also define the regions associated with each system and assign them to user profiles before e*Index information can be accessed. If you install region-specific security but do not want to use the functionality, you can remove it by running* ***remove_ssap.bat****. You need to modify this file in the same way you modified* ***install_ssap.bat*** *in "Step 7: Modify* ***install_ssap.bat****" earlier in this chapter.*

▶ To run install_ssap.bat

Before you begin:

- ✓ Complete "Step 8: Upgrade the Database"
- ✓ Make sure you completed "Step 8: Modify **install_ssap.bat**"

1. Navigate to the path where the database files are located, and then navigate to the **server\DBcommon\Client** directory.
2. Double-click **install_ssap.bat** to run the batch file.

3 Continue to "Step 10: Move the Report Files".

Step 10: Move the Report Files

If you are upgrading from version 4.5.1, there are changes to the report files. If you have not customized the report files, follow the instructions below. If your report files are customized, change the files as described in "Do I Need to Upgrade Report Files?" on page 3-3x.

If you have customized your existing report files, do not overwrite them when you move the new files. Instead, refer to your customized files to modify the new report files.

▶ To move the report files

Before you begin:

- ✓ Know where the e*Index database home directory is located on the database server

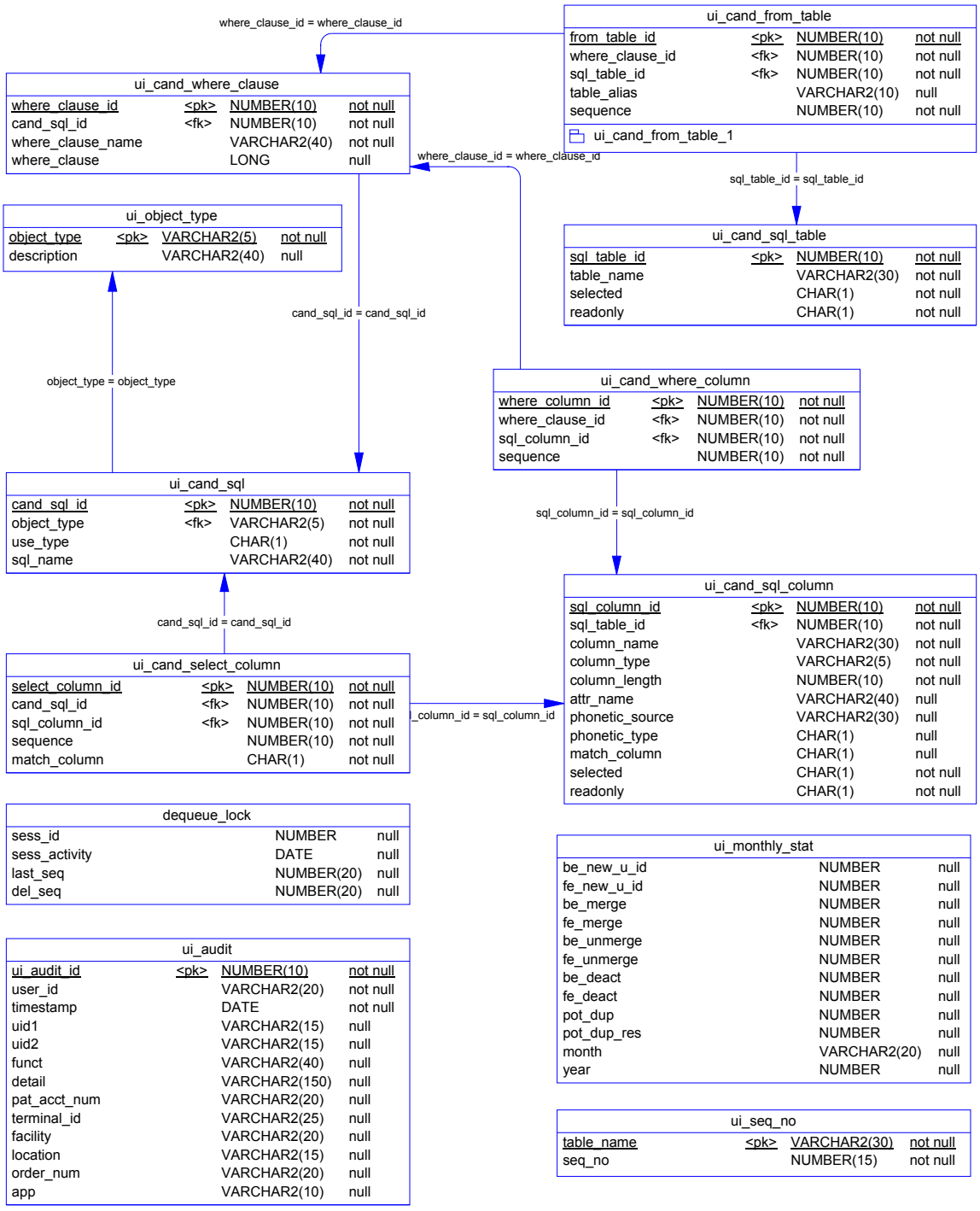
1 Navigate to the path where the database files are located.

2 Open the subdirectory **\server**.

3 Move the folder **UIreports** into the database path on the database server.

e*Index Oracle Database Model

The diagrams on the following pages illustrate the table structure for e*Index on Oracle. Your actual database may vary slightly from this model depending upon the release you have installed. The *ui_person* and *ui_transaction* tables are displayed on two different pages to better illustrate the connections to these two tables.



ui_config			
interface	<pk>	VARCHAR2(255)	not null
code	<pk>	VARCHAR2(255)	not null
value		NUMBER	not null
pk_idx_ui_config			

ui_control			
ctrl_key	<pk>	VARCHAR2(10)	not null
description		VARCHAR2(50)	null
ctrl_value		VARCHAR2(30)	null
create_date		DATE	null
pk_idx_control			

ui_dept			
dept_code	<pk>	VARCHAR2(5)	not null
description		VARCHAR2(20)	null
date_time		DATE	null
pk_idx_dept			

ui_canned_msg			
code	<pk>	VARCHAR2(5)	not null
description		VARCHAR2(80)	not null
create_date		DATE	null
pk_idx_canned			

ui_message			
code	<pk>	VARCHAR2(5)	not null
description		LONG	not null
message_box_header		VARCHAR2(50)	not null
icon		VARCHAR2(15)	null
button		VARCHAR2(20)	null
default_button		NUMBER(1)	null
message_type		VARCHAR2(8)	null
application		VARCHAR2(10)	null
date_time		DATE	null
pk_idx_ui_message			

ui_zip			
zip_code	<pk>	VARCHAR2(8)	not null
zip4		VARCHAR2(4)	null
city	<pk>	VARCHAR2(30)	not null
state	<pk>	VARCHAR2(10)	not null
county		VARCHAR2(3)	null
residence_code		VARCHAR2(4)	null
create_date		DATE	null
pk_idx_zip			

ui_comment			
ui_comment_id	<pk>	NUMBER(10)	not null
u_id		VARCHAR2(15)	not null
type		VARCHAR2(8)	not null
timestamp		DATE	not null
comment_text		LONG	null
ui_org		VARCHAR2(15)	null
ui_id_comment			

ui_msg_header			
ui_msg_header_id	<pk>	NUMBER(20)	not null
queue_id		CHAR(1)	not null
errors		NUMBER(10)	not null
create_date		DATE	not null
create_userid		VARCHAR2(20)	not null
ui_msg_header_1			

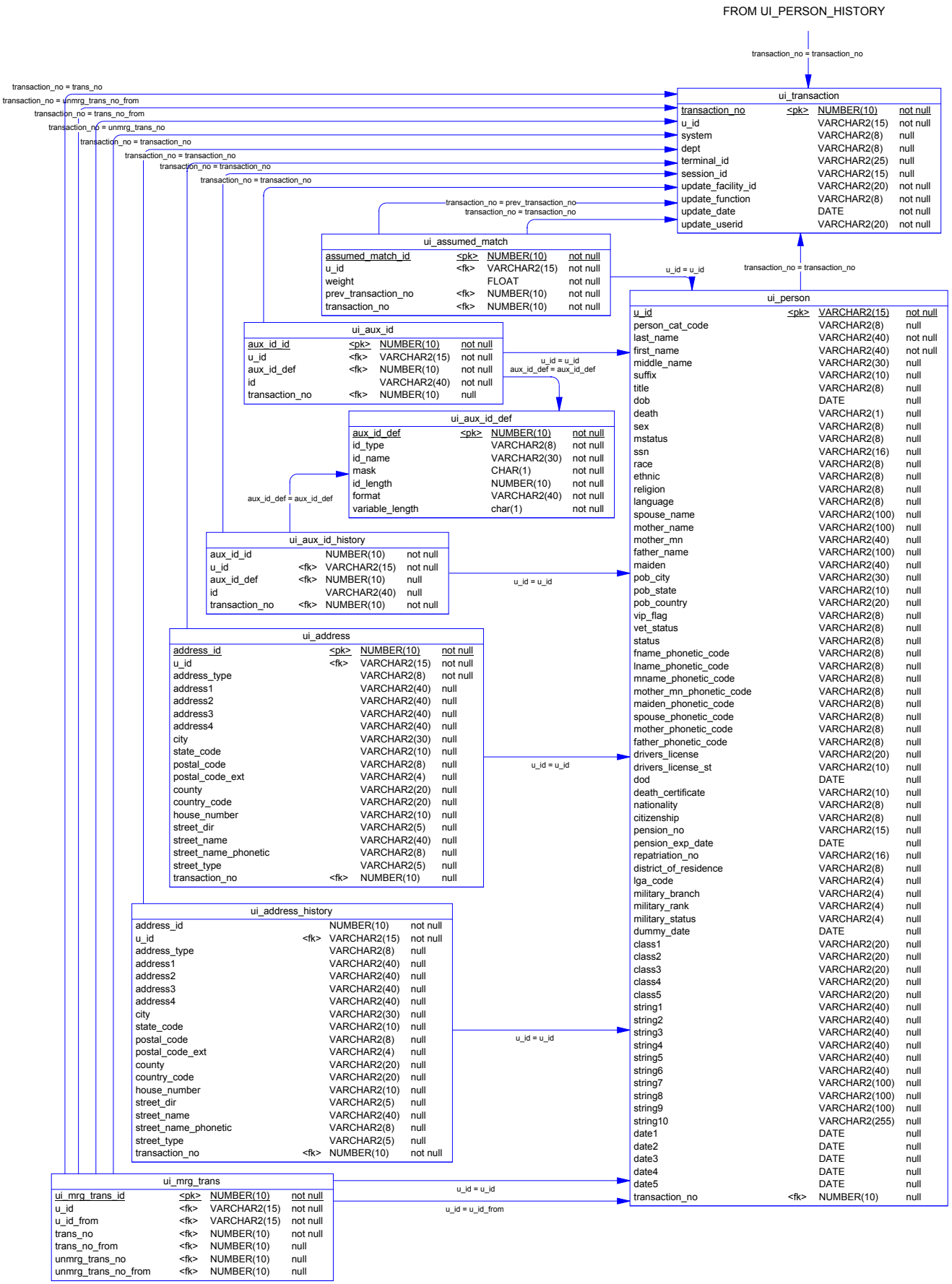
ui_msg_header_id = ui_msg_header_id

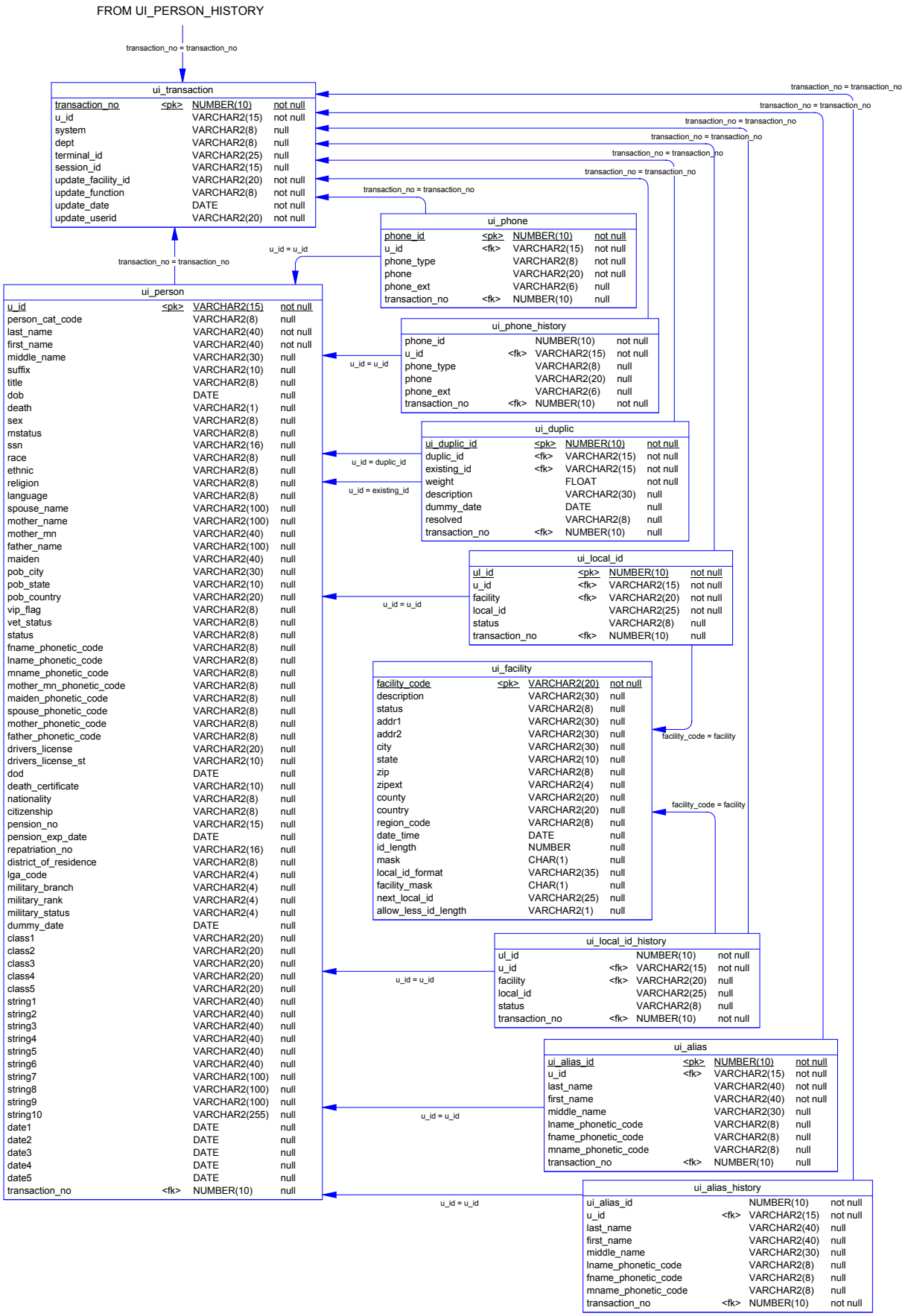
ui_msg_detail			
ui_msg_detail_id	<pk>	NUMBER(20)	not null
ui_msg_header_id	<fk>	NUMBER(20)	null
msg		VARCHAR2(512)	not null
fk_ui_msg_detail			

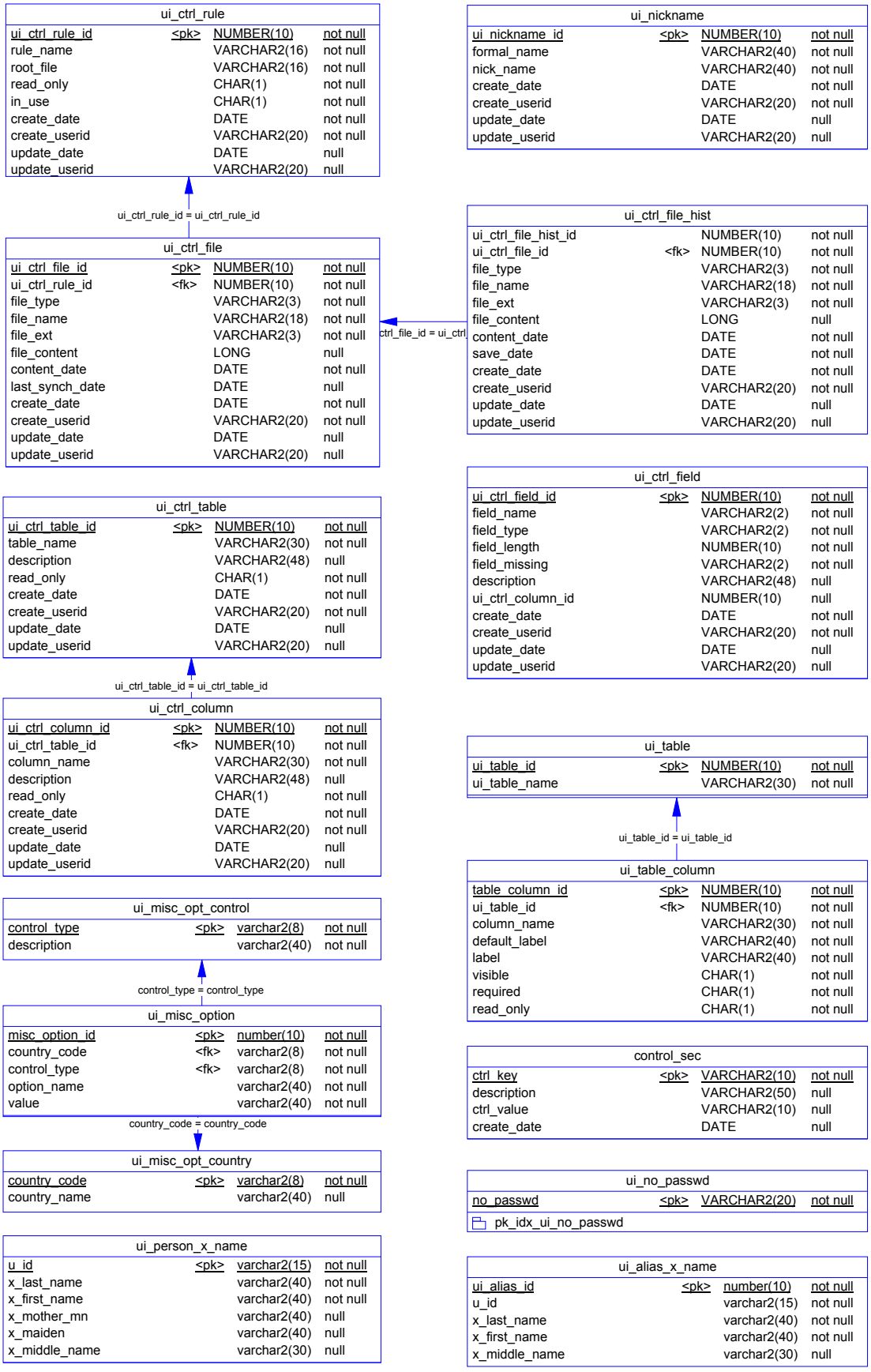
ui_person_history			
ui_person_history_id	<pk>	NUMBER(10)	not null
u_id		VARCHAR2(15)	not null
person_cat_code		VARCHAR2(8)	null
last_name		VARCHAR2(40)	null
first_name		VARCHAR2(40)	null
middle_name		VARCHAR2(30)	null
suffix		VARCHAR2(10)	null
title		VARCHAR2(8)	null
dob		DATE	null
death		VARCHAR2(1)	null
sex		VARCHAR2(8)	null
mstatus		VARCHAR2(8)	null
ssn		VARCHAR2(16)	null
race		VARCHAR2(8)	null
ethnic		VARCHAR2(8)	null
religion		VARCHAR2(8)	null
language		VARCHAR2(8)	null
spouse_name		VARCHAR2(100)	null
mother_name		VARCHAR2(100)	null
mother_mn		VARCHAR2(40)	null
father_name		VARCHAR2(100)	null
maiden		VARCHAR2(40)	null
pob_city		VARCHAR2(30)	null
pob_state		VARCHAR2(10)	null
pob_country		VARCHAR2(20)	null
vip_flag		VARCHAR2(8)	null
vet_status		VARCHAR2(8)	null
status		VARCHAR2(8)	null
fname_phonetic_code		VARCHAR2(8)	null
lname_phonetic_code		VARCHAR2(8)	null
mname_phonetic_code		VARCHAR2(8)	null
mother_mn_phonetic_code		VARCHAR2(8)	null
maiden_phonetic_code		VARCHAR2(8)	null
spouse_phonetic_code		VARCHAR2(8)	null
mother_phonetic_code		VARCHAR2(8)	null
father_phonetic_code		VARCHAR2(8)	null
drivers_license		VARCHAR2(20)	null
drivers_license_st		VARCHAR2(10)	null
dod		DATE	null
death_certificate		VARCHAR2(10)	null
nationality		VARCHAR2(8)	null
citizenship		VARCHAR2(8)	null
pension_no		VARCHAR2(15)	null
pension_exp_date		DATE	null
repatriation_no		VARCHAR2(16)	null
district_of_residence		VARCHAR2(8)	null
lga_code		VARCHAR2(4)	null
military_branch		VARCHAR2(4)	null
military_rank		VARCHAR2(4)	null
military_status		VARCHAR2(4)	null
dummy_date		DATE	null
class1		VARCHAR2(20)	null
class2		VARCHAR2(20)	null
class3		VARCHAR2(20)	null
class4		VARCHAR2(20)	null
class5		VARCHAR2(20)	null
string1		VARCHAR2(40)	null
string2		VARCHAR2(40)	null
string3		VARCHAR2(40)	null
string4		VARCHAR2(40)	null
string5		VARCHAR2(40)	null
string6		VARCHAR2(40)	null
string7		VARCHAR2(100)	null
string8		VARCHAR2(100)	null
string9		VARCHAR2(100)	null
string10		VARCHAR2(255)	null
date1		DATE	null
date2		DATE	null
date3		DATE	null
date4		DATE	null
date5		DATE	null
transaction_no	<fk>	NUMBER(10)	not null

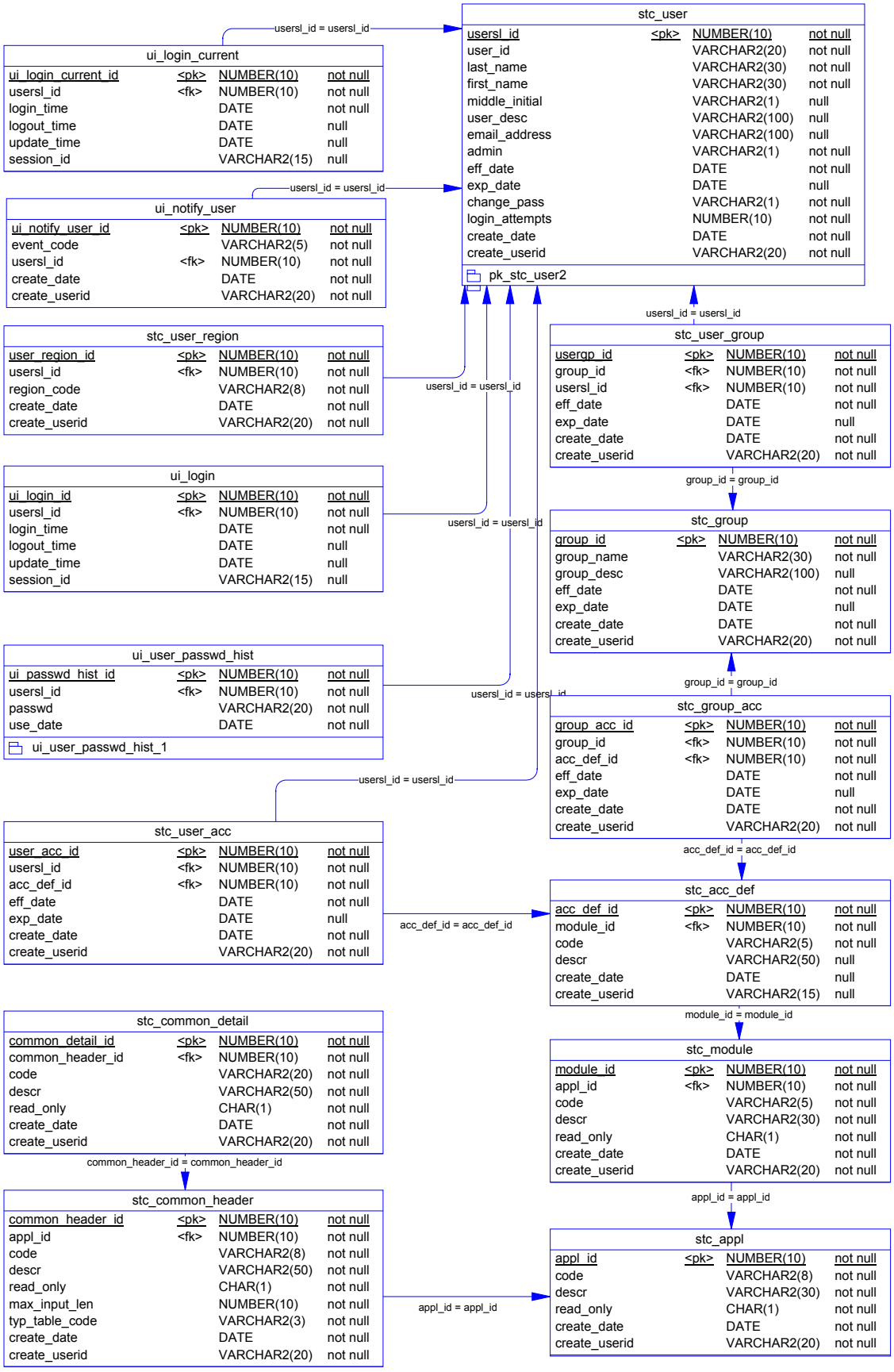
transaction_no = transaction_no

TO UI_TRANSACTION


FROM UI_PERSON_HISTORY
transaction_no = transaction_no
transaction_no = trans_no
transaction_no = unmrg_trans_no_from
transaction_no = trans_no_from
transaction_no = unmrg_trans_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = prev_transaction_no
transaction_no = transaction_no
u_id = u_id
transaction_no = transaction_no
u_id = u_id
aux_id_def = aux_id_def
aux_id_def = aux_id_def
u_id = u_id
u_id = u_id
u_id = u_id
u_id = u_id
u_id = u_id_from
ui_transaction
transaction_no <pk> NUMBER(10) not null
u_id VARCHAR2(15) not null
system VARCHAR2(8) null
dept VARCHAR2(8) null
terminal_id VARCHAR2(25) null
session_id VARCHAR2(15) null
update_facility_id VARCHAR2(20) not null
update_function VARCHAR2(8) not null
update_date DATE not null
update_userid VARCHAR2(20) not null
ui_assumed_match
assumed_match_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
weight FLOAT not null
prev_transaction_no <fk> NUMBER(10) not null
transaction_no <fk> NUMBER(10) not null
ui_aux_id
aux_id_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
aux_id_def <fk> NUMBER(10) not null
id VARCHAR2(40) not null
transaction_no <fk> NUMBER(10) null
ui_aux_id_def
aux_id_def <pk> NUMBER(10) not null
id_type VARCHAR2(8) not null
id_name VARCHAR2(30) not null
mask CHAR(1) not null
id_length NUMBER(10) not null
format VARCHAR2(40) not null
variable_length char(1) not null
ui_aux_id_history
aux_id_id NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
aux_id_def <fk> NUMBER(10) null
id VARCHAR2(40) null
transaction_no <fk> NUMBER(10) not null
ui_address
address_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
address_type VARCHAR2(8) not null
address1 VARCHAR2(40) null
address2 VARCHAR2(40) null
address3 VARCHAR2(40) null
address4 VARCHAR2(40) null
city VARCHAR2(30) null
state_code VARCHAR2(10) null
postal_code VARCHAR2(8) null
postal_code_ext VARCHAR2(4) null
county VARCHAR2(20) null
country_code VARCHAR2(20) null
house_number VARCHAR2(10) null
street_dir VARCHAR2(5) null
street_name VARCHAR2(40) null
street_name_phonetic VARCHAR2(8) null
street_type VARCHAR2(5) null
transaction_no <fk> NUMBER(10) null
ui_address_history
address_id NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
address_type VARCHAR2(8) null
address1 VARCHAR2(40) null
address2 VARCHAR2(40) null
address3 VARCHAR2(40) null
address4 VARCHAR2(40) null
city VARCHAR2(30) null
state_code VARCHAR2(10) null
postal_code VARCHAR2(8) null
postal_code_ext VARCHAR2(4) null
county VARCHAR2(20) null
country_code VARCHAR2(20) null
house_number VARCHAR2(10) null
street_dir VARCHAR2(5) null
street_name VARCHAR2(40) null
street_name_phonetic VARCHAR2(8) null
street_type VARCHAR2(5) null
transaction_no <fk> NUMBER(10) not null
ui_mrg_trans
ui_mrg_trans_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
u_id_from <fk> VARCHAR2(15) not null
trans_no <fk> NUMBER(10) not null
trans_no_from <fk> NUMBER(10) null
unmrg_trans_no <fk> NUMBER(10) null
unmrg_trans_no_from <fk> NUMBER(10) null
ui_person
u_id <pk> VARCHAR2(15) not null
person_cat_code VARCHAR2(8) null
last_name VARCHAR2(40) not null
first_name VARCHAR2(40) not null
middle_name VARCHAR2(30) null
suffix VARCHAR2(10) null
title VARCHAR2(8) null
dob DATE null
death VARCHAR2(1) null
sex VARCHAR2(8) null
mstatus VARCHAR2(8) null
ssn VARCHAR2(16) null
race VARCHAR2(8) null
ethnic VARCHAR2(8) null
religion VARCHAR2(8) null
language VARCHAR2(8) null
spouse_name VARCHAR2(100) null
mother_name VARCHAR2(100) null
mother_mn VARCHAR2(40) null
father_name VARCHAR2(100) null
maiden VARCHAR2(40) null
pob_city VARCHAR2(30) null
pob_state VARCHAR2(10) null
pob_country VARCHAR2(20) null
vip_flag VARCHAR2(8) null
vet_status VARCHAR2(8) null
status VARCHAR2(8) null
fname_phonetic_code VARCHAR2(8) null
lname_phonetic_code VARCHAR2(8) null
mname_phonetic_code VARCHAR2(8) null
mother_mn_phonetic_code VARCHAR2(8) null
maiden_phonetic_code VARCHAR2(8) null
spouse_phonetic_code VARCHAR2(8) null
mother_phonetic_code VARCHAR2(8) null
father_phonetic_code VARCHAR2(8) null
drivers_license VARCHAR2(20) null
drivers_license_st VARCHAR2(10) null
dod DATE null
death_certificate VARCHAR2(10) null
nationality VARCHAR2(8) null
citizenship VARCHAR2(8) null
pension_no VARCHAR2(15) null
pension_exp_date DATE null
repatriation_no VARCHAR2(16) null
district_of_residence VARCHAR2(8) null
lga_code VARCHAR2(4) null
military_branch VARCHAR2(4) null
military_rank VARCHAR2(4) null
military_status VARCHAR2(4) null
dummy_date DATE null
class1 VARCHAR2(20) null
class2 VARCHAR2(20) null
class3 VARCHAR2(20) null
class4 VARCHAR2(20) null
class5 VARCHAR2(20) null
string1 VARCHAR2(40) null
string2 VARCHAR2(40) null
string3 VARCHAR2(40) null
string4 VARCHAR2(40) null
string5 VARCHAR2(40) null
string6 VARCHAR2(40) null
string7 VARCHAR2(100) null
string8 VARCHAR2(100) null
string9 VARCHAR2(100) null
string10 VARCHAR2(255) null
date1 DATE null
date2 DATE null
date3 DATE null
date4 DATE null
date5 DATE null
transaction_no <fk> NUMBER(10) null


FROM UI_PERSON_HISTORY
transaction_no = transaction_no
ui_transaction
transaction_no <pk> NUMBER(10) not null
u_id VARCHAR2(15) not null
system VARCHAR2(8) null
dept VARCHAR2(8) null
terminal_id VARCHAR2(25) null
session_id VARCHAR2(15) null
update_facility_id VARCHAR2(20) not null
update_function VARCHAR2(8) not null
update_date DATE not null
update_userid VARCHAR2(20) not null
ui_phone
phone_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
phone_type VARCHAR2(8) not null
phone VARCHAR2(20) not null
phone_ext VARCHAR2(6) null
transaction_no <fk> NUMBER(10) null
u_id = u_id
ui_person
u_id <pk> VARCHAR2(15) not null
person_cat_code VARCHAR2(8) null
last_name VARCHAR2(40) not null
first_name VARCHAR2(40) not null
middle_name VARCHAR2(30) null
suffix VARCHAR2(10) null
title VARCHAR2(8) null
dob DATE null
death VARCHAR2(1) null
sex VARCHAR2(8) null
mstatus VARCHAR2(8) null
ssn VARCHAR2(16) null
race VARCHAR2(8) null
ethnic VARCHAR2(8) null
religion VARCHAR2(8) null
language VARCHAR2(8) null
spouse_name VARCHAR2(100) null
mother_name VARCHAR2(100) null
mother_mn VARCHAR2(40) null
father_name VARCHAR2(100) null
maiden VARCHAR2(40) null
pob_city VARCHAR2(30) null
pob_state VARCHAR2(10) null
pob_country VARCHAR2(20) null
vip_flag VARCHAR2(8) null
vet_status VARCHAR2(8) null
status VARCHAR2(8) null
fname_phonetic_code VARCHAR2(8) null
lname_phonetic_code VARCHAR2(8) null
mname_phonetic_code VARCHAR2(8) null
mother_mn_phonetic_code VARCHAR2(8) null
maiden_phonetic_code VARCHAR2(8) null
spouse_phonetic_code VARCHAR2(8) null
mother_phonetic_code VARCHAR2(8) null
father_phonetic_code VARCHAR2(8) null
drivers_license VARCHAR2(20) null
drivers_license_st VARCHAR2(10) null
dod DATE null
death_certificate VARCHAR2(10) null
nationality VARCHAR2(8) null
citizenship VARCHAR2(8) null
pension_no VARCHAR2(15) null
pension_exp_date DATE null
repatriation_no VARCHAR2(16) null
district_of_residence VARCHAR2(8) null
lga_code VARCHAR2(4) null
military_branch VARCHAR2(4) null
military_rank VARCHAR2(4) null
military_status VARCHAR2(4) null
dummy_date DATE null
class1 VARCHAR2(20) null
class2 VARCHAR2(20) null
class3 VARCHAR2(20) null
class4 VARCHAR2(20) null
class5 VARCHAR2(20) null
string1 VARCHAR2(40) null
string2 VARCHAR2(40) null
string3 VARCHAR2(40) null
string4 VARCHAR2(40) null
string5 VARCHAR2(40) null
string6 VARCHAR2(40) null
string7 VARCHAR2(100) null
string8 VARCHAR2(100) null
string9 VARCHAR2(100) null
string10 VARCHAR2(255) null
date1 DATE null
date2 DATE null
date3 DATE null
date4 DATE null
date5 DATE null
transaction_no <fk> NUMBER(10) null
ui_phone_history
phone_id NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
phone_type VARCHAR2(8) null
phone VARCHAR2(20) null
phone_ext VARCHAR2(6) null
transaction_no <fk> NUMBER(10) not null
u_id = u_id
ui_duplic
ui_duplic_id <pk> NUMBER(10) not null
duplic_id <fk> VARCHAR2(15) not null
existing_id <fk> VARCHAR2(15) not null
weight FLOAT not null
description VARCHAR2(30) null
dummy_date DATE null
resolved VARCHAR2(8) null
transaction_no <fk> NUMBER(10) null
u_id = duplic_id
u_id = existing_id
ui_local_id
ul_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
facility <fk> VARCHAR2(20) not null
local_id VARCHAR2(25) not null
status VARCHAR2(8) null
transaction_no <fk> NUMBER(10) null
u_id = u_id
ui_facility
facility_code <pk> VARCHAR2(20) not null
description VARCHAR2(30) null
status VARCHAR2(8) null
addr1 VARCHAR2(30) null
addr2 VARCHAR2(30) null
city VARCHAR2(30) null
state VARCHAR2(10) null
zip VARCHAR2(8) null
zipext VARCHAR2(4) null
county VARCHAR2(20) null
country VARCHAR2(20) null
region_code VARCHAR2(8) null
date_time DATE null
id_length NUMBER null
mask CHAR(1) null
local_id_format VARCHAR2(35) null
facility_mask CHAR(1) null
next_local_id VARCHAR2(25) null
allow_less_id_length VARCHAR2(1) null
facility_code = facility
facility_code = facility
ui_local_id_history
ul_id NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
facility <fk> VARCHAR2(20) null
local_id VARCHAR2(25) null
status VARCHAR2(8) null
transaction_no <fk> NUMBER(10) not null
u_id = u_id
ui_alias
ui_alias_id <pk> NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
last_name VARCHAR2(40) not null
first_name VARCHAR2(40) not null
middle_name VARCHAR2(30) null
lname_phonetic_code VARCHAR2(8) null
fname_phonetic_code VARCHAR2(8) null
mname_phonetic_code VARCHAR2(8) null
transaction_no <fk> NUMBER(10) null
u_id = u_id
ui_alias_history
ui_alias_id NUMBER(10) not null
u_id <fk> VARCHAR2(15) not null
last_name VARCHAR2(40) null
first_name VARCHAR2(40) null
middle_name VARCHAR2(30) null
lname_phonetic_code VARCHAR2(8) null
fname_phonetic_code VARCHAR2(8) null
mname_phonetic_code VARCHAR2(8) null
transaction_no <fk> NUMBER(10) not null
u_id = u_id


ui_ctrl_rule
ui_ctrl_rule_id <pk> NUMBER(10) not null
rule_name VARCHAR2(16) not null
root_file VARCHAR2(16) not null
read_only CHAR(1) not null
in_use CHAR(1) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ui_ctrl_rule_id = ui_ctrl_rule_id
ui_ctrl_file
ui_ctrl_file_id <pk> NUMBER(10) not null
ui_ctrl_rule_id <fk> NUMBER(10) not null
file_type VARCHAR2(3) not null
file_name VARCHAR2(18) not null
file_ext VARCHAR2(3) not null
file_content LONG null
content_date DATE not null
last_synch_date DATE null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ctrl_file_id = ui_ctrl
ui_nickname
ui_nickname_id <pk> NUMBER(10) not null
formal_name VARCHAR2(40) not null
nick_name VARCHAR2(40) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ui_ctrl_file_hist
ui_ctrl_file_hist_id NUMBER(10) not null
ui_ctrl_file_id <fk> NUMBER(10) not null
file_type VARCHAR2(3) not null
file_name VARCHAR2(18) not null
file_ext VARCHAR2(3) not null
file_content LONG null
content_date DATE not null
save_date DATE not null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ui_ctrl_table
ui_ctrl_table_id <pk> NUMBER(10) not null
table_name VARCHAR2(30) not null
description VARCHAR2(48) null
read_only CHAR(1) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ui_ctrl_table_id = ui_ctrl_table_id
ui_ctrl_column
ui_ctrl_column_id <pk> NUMBER(10) not null
ui_ctrl_table_id <fk> NUMBER(10) not null
column_name VARCHAR2(30) not null
description VARCHAR2(48) null
read_only CHAR(1) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ui_ctrl_field
ui_ctrl_field_id <pk> NUMBER(10) not null
field_name VARCHAR2(2) not null
field_type VARCHAR2(2) not null
field_length NUMBER(10) not null
field_missing VARCHAR2(2) not null
description VARCHAR2(48) null
ui_ctrl_column_id NUMBER(10) null
create_date DATE not null
create_userid VARCHAR2(20) not null
update_date DATE null
update_userid VARCHAR2(20) null
ui_misc_opt_control
control_type <pk> varchar2(8) not null
description varchar2(40) not null
control_type = control_type
ui_misc_option
misc_option_id <pk> number(10) not null
country_code <fk> varchar2(8) not null
control_type <fk> varchar2(8) not null
option_name varchar2(40) not null
value varchar2(40) not null
country_code = country_code
ui_misc_opt_country
country_code <pk> varchar2(8) not null
country_name varchar2(40) null
ui_person_x_name
u_id <pk> varchar2(15) not null
x_last_name varchar2(40) not null
x_first_name varchar2(40) not null
x_mother_mn varchar2(40) null
x_maiden varchar2(40) null
x_middle_name varchar2(30) null
ui_table
ui_table_id <pk> NUMBER(10) not null
ui_table_name VARCHAR2(30) not null
ui_table_id = ui_table_id
ui_table_column
table_column_id <pk> NUMBER(10) not null
ui_table_id <fk> NUMBER(10) not null
column_name VARCHAR2(30) not null
default_label VARCHAR2(40) not null
label VARCHAR2(40) not null
visible CHAR(1) not null
required CHAR(1) not null
read_only CHAR(1) not null
control_sec
ctrl_key <pk> VARCHAR2(10) not null
description VARCHAR2(50) null
ctrl_value VARCHAR2(10) null
create_date DATE null
ui_no_passwd
no_passwd <pk> VARCHAR2(20) not null
pk_idx_ui_no_passwd
ui_alias_x_name
ui_alias_id <pk> number(10) not null
u_id varchar2(15) not null
x_last_name varchar2(40) not null
x_first_name varchar2(40) not null
x_middle_name varchar2(30) null


stc_user
usersl_id <pk> NUMBER(10) not null
user_id VARCHAR2(20) not null
last_name VARCHAR2(30) not null
first_name VARCHAR2(30) not null
middle_initial VARCHAR2(1) null
user_desc VARCHAR2(100) null
email_address VARCHAR2(100) null
admin VARCHAR2(1) not null
eff_date DATE not null
exp_date DATE null
change_pass VARCHAR2(1) not null
login_attempts NUMBER(10) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
pk_stc_user2
usersl_id = usersl_id
ui_login_current
ui_login_current_id <pk> NUMBER(10) not null
usersl_id <fk> NUMBER(10) not null
login_time DATE not null
logout_time DATE null
update_time DATE null
session_id VARCHAR2(15) null
ui_notify_user
ui_notify_user_id <pk> NUMBER(10) not null
event_code VARCHAR2(5) not null
usersl_id <fk> NUMBER(10) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_user_region
user_region_id <pk> NUMBER(10) not null
usersl_id <fk> NUMBER(10) not null
region_code VARCHAR2(8) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
ui_login
ui_login_id <pk> NUMBER(10) not null
usersl_id <fk> NUMBER(10) not null
login_time DATE not null
logout_time DATE null
update_time DATE null
session_id VARCHAR2(15) null
ui_user_passwd_hist
ui_passwd_hist_id <pk> NUMBER(10) not null
usersl_id <fk> NUMBER(10) not null
passwd VARCHAR2(20) not null
use_date DATE not null
ui_user_passwd_hist_1
stc_user_acc
user_acc_id <pk> NUMBER(10) not null
usersl_id <fk> NUMBER(10) not null
acc_def_id <fk> NUMBER(10) not null
eff_date DATE not null
exp_date DATE null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_common_detail
common_detail_id <pk> NUMBER(10) not null
common_header_id <fk> NUMBER(10) not null
code VARCHAR2(20) not null
descr VARCHAR2(50) not null
read_only CHAR(1) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_common_header
common_header_id <pk> NUMBER(10) not null
appl_id <fk> NUMBER(10) not null
code VARCHAR2(8) not null
descr VARCHAR2(50) not null
read_only CHAR(1) not null
max_input_len NUMBER(10) not null
typ_table_code VARCHAR2(3) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_user_group
usergp_id <pk> NUMBER(10) not null
group_id <fk> NUMBER(10) not null
usersl_id <fk> NUMBER(10) not null
eff_date DATE not null
exp_date DATE null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_group
group_id <pk> NUMBER(10) not null
group_name VARCHAR2(30) not null
group_desc VARCHAR2(100) null
eff_date DATE not null
exp_date DATE null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_group_acc
group_acc_id <pk> NUMBER(10) not null
group_id <fk> NUMBER(10) not null
acc_def_id <fk> NUMBER(10) not null
eff_date DATE not null
exp_date DATE null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_acc_def
acc_def_id <pk> NUMBER(10) not null
module_id <fk> NUMBER(10) not null
code VARCHAR2(5) not null
descr VARCHAR2(50) null
create_date DATE null
create_userid VARCHAR2(15) null
stc_module
module_id <pk> NUMBER(10) not null
appl_id <fk> NUMBER(10) not null
code VARCHAR2(5) not null
descr VARCHAR2(30) not null
read_only CHAR(1) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
stc_appl
appl_id <pk> NUMBER(10) not null
code VARCHAR2(8) not null
descr VARCHAR2(30) not null
read_only CHAR(1) not null
create_date DATE not null
create_userid VARCHAR2(20) not null
usersl_id = usersl_id
usersl_id = usersl_id
usersl_id = usersl_id
usersl_id = usersl_id
usersl_id = usersl_id
usersl_id = usersl_id
group_id = group_id
group_id = group_id
acc_def_id = acc_def_id
acc_def_id = acc_def_id
module_id = module_id
appl_id = appl_id
common_header_id = common_header_id
appl_id = appl_id

Chapter 4

Upgrading a Sybase Database

About this Chapter

Overview

This chapter presents the background information and the step-by-step instructions you need to upgrade an e*Index database for Sybase from version 4.5.1 or later to version 5.0.5 for SRE. The following diagram illustrates the major topics in this chapter. For the page numbers on which specific topics appear, see the next page of this chapter.

About the Upgrade	Learn about installing and modifying the database upgrade files
Upgrade the Database	Learn how to install the database upgrade files, and to modify and run the database scripts
Physical Data Model	View a physical data model of the e*Index database

What's Inside

This chapter provides background information and instructions related to the topics listed below.

Learning About the Database Upgrade

Overview

This section of the chapter provides background information about the files you install, modify, and execute to upgrade an e*Index Sybase database from version 4.5.1 or later to version 5.0.5 for SRE.

What do I Need to Know Before I Start?

Make sure you are familiar with your Sybase environment before beginning the upgrade procedure. Know the server name and pathname in which the e*Index database resides, and the Adaptive Server name and password of the database you are upgrading. It is helpful to be familiar with your Sybase environment, Sybase networking, and Sybase database administration. Also, determine whether you have region-specific security installed, and, if not, whether you want to install it (for more information see chapter 3 of the *e*Index Administrator User's Guide* and the *e*Index Security User's Guide*).

e*Index is only compatible with Sybase 12.0. If you are using a previous version of Sybase, upgrade your database to Sybase 12.0 before beginning the e*Index upgrade.

How is the Database Upgraded?

Installing the database files creates several SQL scripts used to upgrade the existing database; however, you only need to execute one batch file, **upgrade.bat**, to perform the upgrade. The upgrade batch file calls the SQL scripts to make the necessary changes to the database. This file requires some modification prior to execution. It must be run from a machine running a Windows operating system (XP, 2000, or 2003) with Sybase Client installed. You can run the upgrade file from an existing e*Index client workstation.

If you plan to install region-specific security, you need to modify and run a second file, **install_ssap.bat**. Note that this process can only upgrade a database that is already at version 4.5.3 or later.

Performing the Upgrade

Overview

To upgrade an e*Index database for Sybase to version 5.0.5 for SRE, you must complete the following steps:

- Step 1: Obtain Database Information
- Step 2: Back up the Current Database
- Step 3: Upgrade Sybase Server
- Step 4: Install the Database and Report Files
- Step 5: Verify **sql.ini**
- Step 6: Modify the Upgrade File
- Step 7: Modify **install_ssap.bat** (optional)
- Step 8: Upgrade the Database
- Step 9: Run **install_ssap.bat** (optional)

Note: *If you chose to install all components of e*Index when you installed the e*Index Schema files, you should have already completed steps 1 through 4. You can begin with "Step 5: Verify **sql.ini**".*

Step 1: Obtain Database Information

Before beginning, gather information about your database, such as the database path, the database name, and the system login and password for your database. You should also know the path to your Sybase home directory. If you customized the rule set files for the Vality matching algorithm, know the location of the most current rule set files. If you currently do not have region-specific security installed, determine whether you want to install it now. Region-specific security is described in Chapter 3 of the *e*Index Security User's Guide* and the *e*Index Administrator User's Guide*.

Because of the complex nature of modifying a database, we recommend that a database administrator perform the following steps.

Step 2: Back up the Current Database

Prior to making any changes to your e*Index database, you should make a complete backup of your current database. For more information on performing Sybase database backups, see the appropriate Sybase documentation.

Step 3: Upgrade Sybase Server

Before you install the e*Index database files, Sybase 12.0 Server must be installed on the database server. If you are currently running a previous version of Sybase, you need to upgrade Sybase before continuing. If your Sybase home directory changes during the upgrade, make a note of the new path. In order to complete the upgrade, you must also have Sybase 12.0 Client installed on a client workstation (for more information, see "Step 2: Upgrade the Database Software" in chapter 6 of this guide). For information about upgrading Sybase Server, refer to the appropriate upgrade documentation for Sybase.

Important! *It is crucial that the Sybase 12.0 Server and Client installation is operational prior to performing the following steps.*

Step 4: Install the Database and Report Files

Installing the database and report files is very similar to the process you followed to install the e*Index schema files.

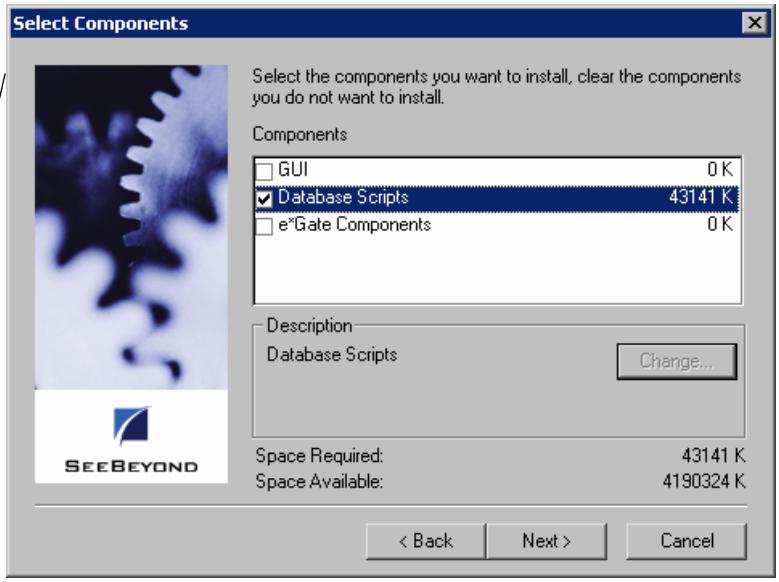
▶ To install the database and report files

Before you begin:

- ✓ Complete "Step 2: Install Sybase"
- ✓ Make sure the e*Index installation CD-ROM is inserted into the CD-ROM drive of a client workstation with Sybase Client 12.0 installed

1. Follow steps 1 through 7 under "Step 5: Install the e*Index Schema Files" in Chapter 2, "Installing the e*Gate Schema Files". The Select Components window should now be visible.
2. On the Select Components window, select the check box next to **Database Scripts**.

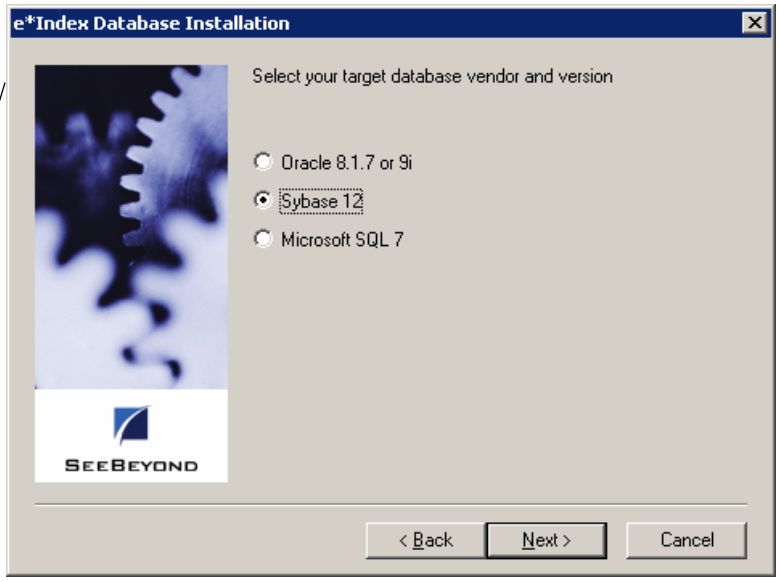
Select the components you want to install on the Select Components window



Next button

3 Click **Next**. The e*Index Database Installation window appears.

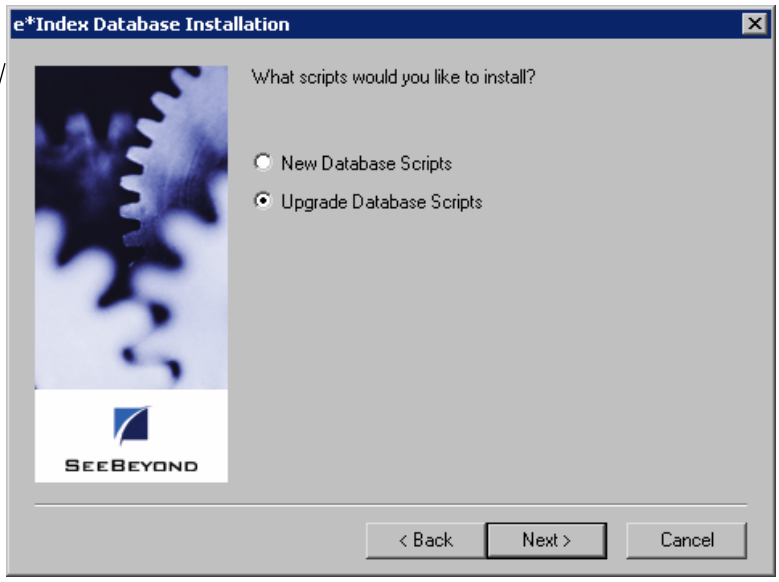
Specify your database platform on the e*Index Database Installation window



Next button

4 Select **Sybase 12**, and then click **Next**. A second e*Index Database Installation window appears.

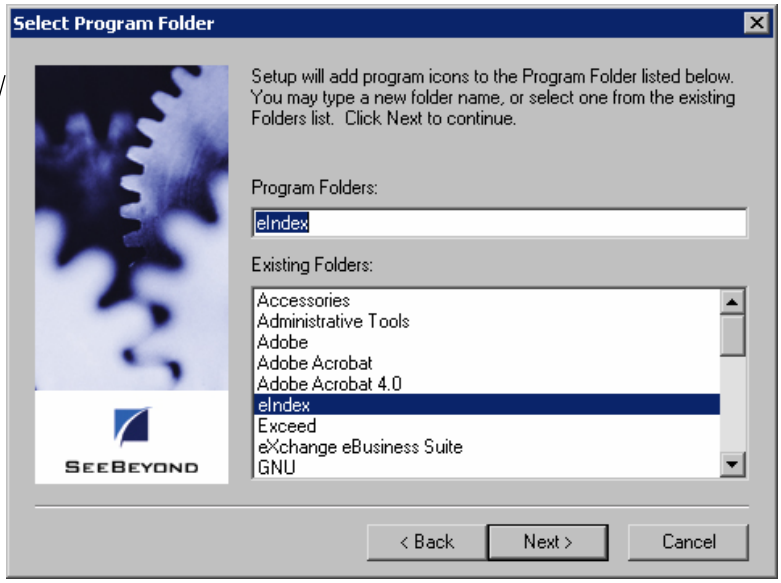
Specify whether to install the scripts to install a new database or the scripts to upgrade an existing database



Next button

5 Select **New Database Scripts**, and then click **Next**. The Select Program Folder window appears.

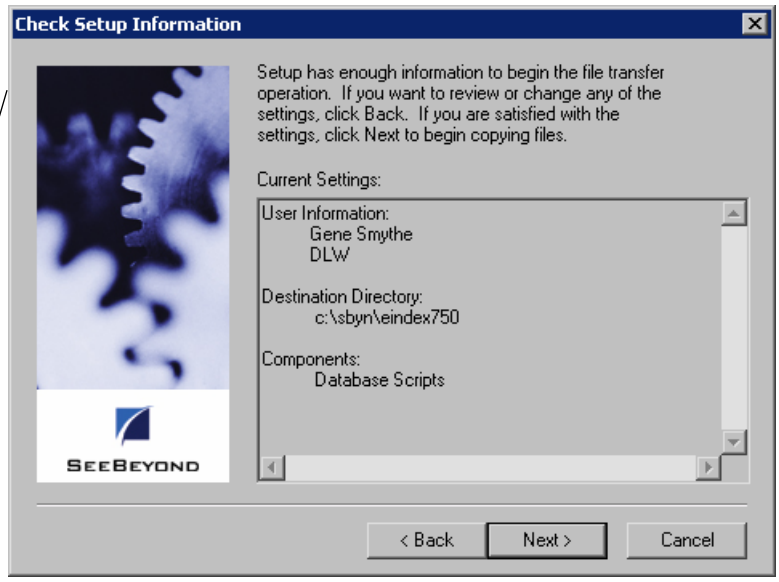
Specify the folder in which to install the program icons on the Select Program Folder window



Next button

6 Enter the name of the folder into which you want to install the program icons or accept the default name, and then click **Next**. The Check Setup Information window appears.

On the Check Setup Information window, verify the information you specified



7 Verify the information you specified, and do one of the following:

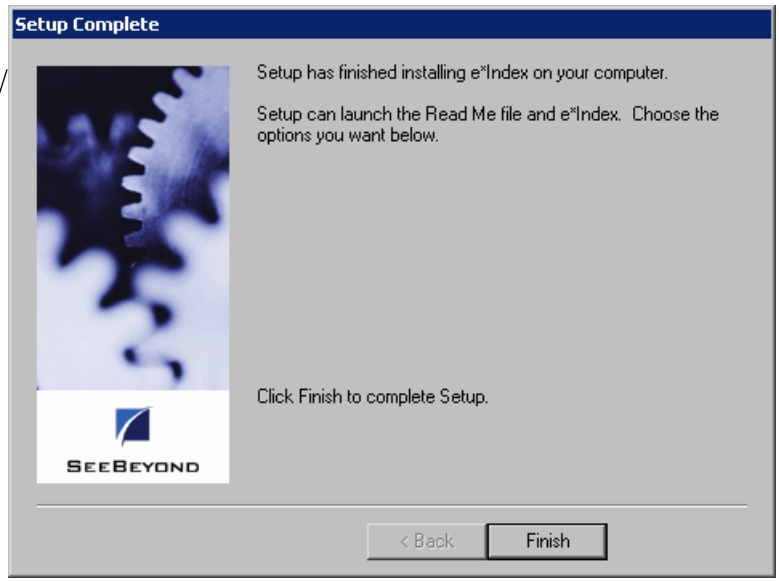
< Back
Back button

To change any of the options you selected, click **Back**, and make the necessary changes.

Next >
Next button

To continue with the installation, click **Next**. When all files are installed, the Setup Complete window appears.

The Setup Complete window indicates that the database files have been installed



Finish
Finish button

8 Click **Finish** to complete the setup process.

9 To view the database files that were installed:

- Open Windows Explorer and navigate to the path you specified for the installation files.
- Open the **\server\DBcommon** subdirectory. You will find one subdirectory named **Client**, and two files, **defs.sql** and **sybenv.bat**. The Client directory contains the database scripts, **upgrade.bat**, **install_ssap.bat**, and **remove_ssap.bat**.

10 Continue to "Step 5: Verify **sql.ini**".

Step 5: Verify sql.ini

Verify that the **sql.ini** file on the computer you are using for the upgrade contains a stanza pointing to the database being upgraded. You should have a stanza for the e*Index database similar to the following example. This sample illustrates how the stanza may appear for an Adaptive Server named "challenger" using a TCP/IP connection. For more information about configuring this file, refer to your Sybase user documentation.

```
[challenger]
master=TCP,challenger,4100
query=TCP,challenger,4100
```

Step 6: Modify the Upgrade File

The file **upgrade.bat** is a batch file that you run to upgrade the database tables, triggers, indexes, and so on. You need to modify certain variables in this file that tell the script how to locate the database instance and installation scripts.

▶ To modify the upgrade file

Before you begin:

✓ Complete "Step 5: Verify **sql.ini**"

1 On the machine from which you will run **upgrade.bat**, navigate to the path where the database files are located, and then navigate to the **\server\DBcommon\Client** directory.

2 Make a backup copy of the file **upgrade.bat**.

3 Right-click the **upgrade.bat** file and then select **Edit** from the list that appears. Do not double-click the file to open it.

4 Make the following modifications:

- In the variable **SERVER**, **e**nter the name of the Sybase Adaptive Server for the database.
- In the variable **DATABASE**, enter the name of the database.
- In the variable **INSTALLDRIVE**, enter the drive on which the database installation files are located.
- In the variable **INSTALL_HOME**, enter the path in which the database installation files are located on the machine from which you will run **upgrade.bat**. Do not include the drive designation in this path. This is the path to the **server** directory (do not include **\server\Dbcommon\Server** in the pathname).
- In the variable **SYSTEMPW**, enter the password for the administrator user ID for Sybase.
- The default value for the variable **UIPW** only needs to be changed if you modified the password for the e*Index **UI** user.

The sample below illustrates the **upgrade.bat** variables.

```
SET SERVER=SYBASE1
SET DATABASE=ei01
SET INSTALLDRIVE=C:
SET INSTALL_HOME=\TEMP\INSTALL
...
SET SYSTEMPW=sybase
SET UIPW=eindex
```

5 Save the changes to **upgrade.bat** and close the file.

6 Do one of the following:

If you want to install region-specific security in the database, continue to "Step 7: Modify **install_ssap.bat** (optional)". If you already have region-specific security installed, you do not need to run this file.

If you do not want to install region-specific security in the database, skip to "Step 8: Upgrade the Database".

Step 7: Modify install_ssap.bat (optional)

Running **install_ssap.bat** installs the views you need in order to use the region-specific security capabilities of e*Index. You do not need to modify or run this file if you do not want to use this capability or if you already have region-specific security installed.

For more information about region-specific security, see "What is Region-Specific Security?" in Chapter 3 of the *e*Index Security User's Guide* and "About Region-Specific Security" in Chapter 3 of the *e*Index Administrator User's Guide*.

▶ To modify install_ssap.bat

Before you begin:

✓ Complete "Step 6: Modify the Upgrade File"

1 On the machine from which you will run **install_ssap.bat**, navigate to the path where the database files are located, and then navigate to the **\server\DBcommon\Client** directory.

2 Make a backup copy of the file **install_ssap.bat**.

3 Right-click the **install_ssap.bat** file and then select **Edit** from the list that appears. Do not double-click this file to open it.

4 Make the following modifications:

- In the variable **SERVER**, **e**nter the name of the Sybase Adaptive Server for the database.
- In the variable **DATABASE**, enter the name of the database.
- The default value for the variable **UIPW** only needs to be changed if you modified the password for the e*Index **UI** user.

The sample below illustrates the **install_ssap.bat** variables.

```
SET SERVER=SYBASE1
SET DATABASE=ei01
...
SET UIPW=eindex
```

5 Save the changes to **install_ssap.bat** and close the file.

6 Continue to "Step 8: Upgrade the Database".

Step 8: Upgrade the Database

Once you modify **upgrade.bat**, you can run the file to update the tables, views, indexes, and so on for your e*Index database.

Important! *Make sure your database is at version 4.5.1 or later before continuing. Otherwise, this step will create and populate new database views, which may take several hours to complete.*

▶ To upgrade the database

Before you begin:

✓ Complete "Step 6: Modify the Upgrade File" and, optionally, "Step 7: Modify **install_ssap.bat**"

1. On the machine from which you will run **upgrade.bat**, navigate to the path where the database files are located, and then navigate to the **\server\DBcommon\Client** directory.
2. Double-click **upgrade.bat** to run the batch file.
3. At the prompt, press any key on the keyboard to complete the installation.
4. Review the log files, located in the path you specified for your installation files in **\server\upgrades\Spool**, to ensure that there were no errors during the running of the script.
5. If you want to install region-specific security, continue to "Step 9: Run **install_ssap.bat** (optional)". Otherwise, your database upgrade is complete.

Step 9: Run install_ssap.bat (optional)

After you have run **upgrade.bat** to update the e*Index database, you can install system-specific security by running **install_ssap.bat**. If you already have region-specific security installed, you do not need to run this script.

Important! *Do not execute this file if you do not want to use region-specific security. Creating these views requires that you also define the regions associated with each system and assign them to user profiles before you can access e*Index information. If you install region-specific security but do not want to use it, run* ***remove_ssap.bat*** *to remove the tables. Modify this file in the same way you modified* ***install_ssap.bat*** *in "Step 7: Modify* ***install_ssap.bat****".*

▶ To run install_ssap.bat

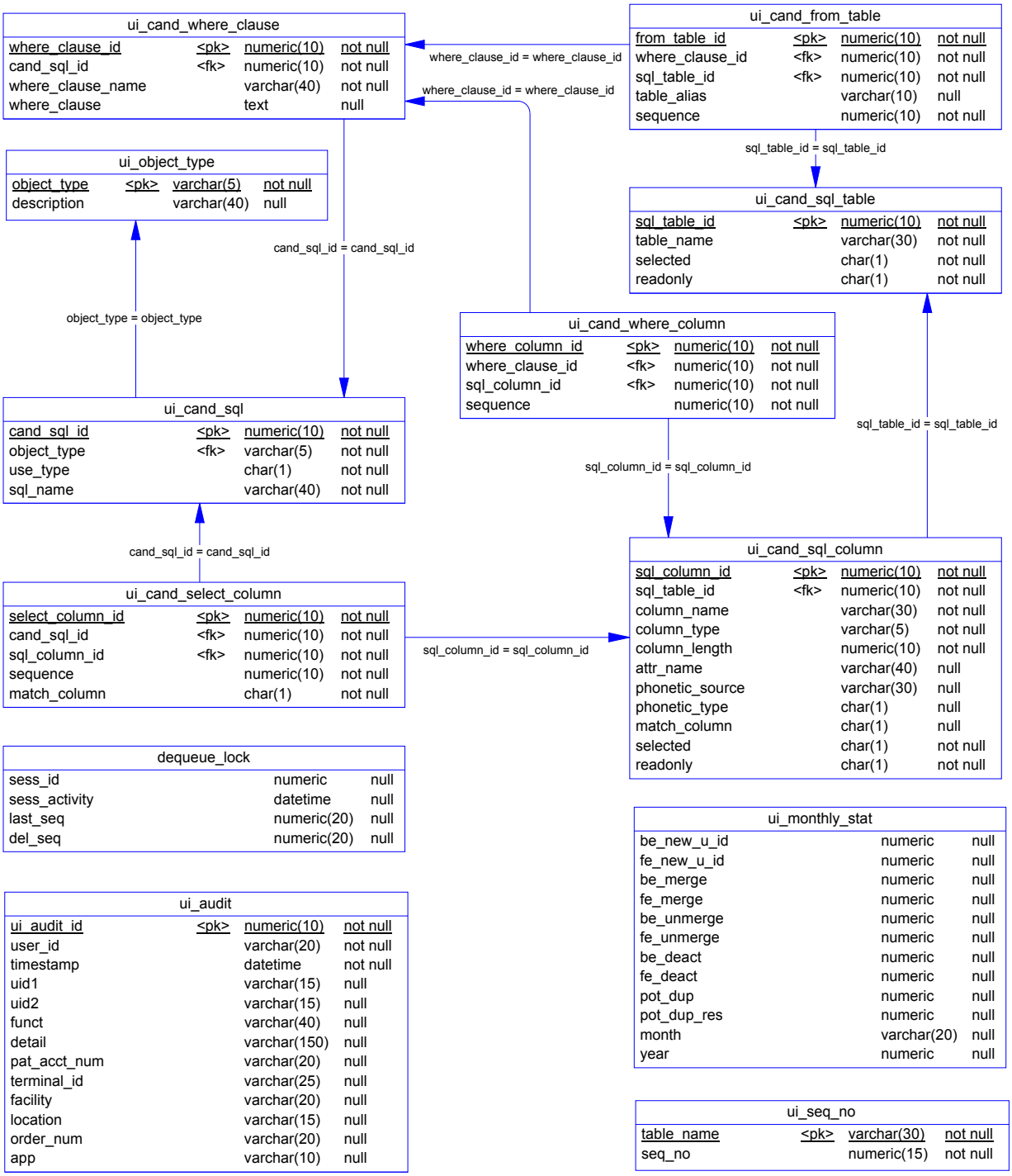
Before you begin:

- ✓ Complete "Step 8: Upgrade the Database"

1. On the machine from which you will run **install_ssap.bat**, navigate to the path where the database files are located, and then navigate to the **\server\DBcommon\Client** directory.
2. Double-click **install_ssap.bat** to run the batch file.
3. At the prompt, press any key on the keyboard to complete the installation.

e*Index Sybase Database Model

The diagrams on the following pages illustrate the table structure for e*Index for Sybase. Your actual database may vary slightly from this model depending upon the release you have installed. The *ui_person* and *ui_transaction* tables are displayed on two different pages to better illustrate the connections to these two tables.



ui_config			
interface	<pk>	varchar(255)	not null
code	<pk>	varchar(255)	not null
value		numeric	not null

ui_control			
ctrl_key	<pk>	varchar(10)	not null
description		varchar(50)	null
ctrl_value		varchar(30)	null
create_date		datetime	null

ui_dept			
dept_code	<pk>	varchar(5)	not null
description		varchar(20)	null
date_time		datetime	null

ui_canned_msg			
code	<pk>	varchar(5)	not null
description		varchar(80)	not null
create_date		datetime	null

ui_message			
code	<pk>	varchar(5)	not null
description		varchar(255)	not null
message_box_header		varchar(50)	not null
icon		varchar(15)	null
button		varchar(20)	null
default_button		numeric(1)	null
message_type		varchar(8)	null
application		varchar(10)	null
date_time		datetime	null

ui_zip			
zip_code	<pk>	varchar(8)	not null
zip4		varchar(4)	null
city	<pk>	varchar(30)	not null
state	<pk>	varchar(10)	not null
county		varchar(3)	null
residence_code		varchar(4)	null
create_date		datetime	null

ui_comment			
ui_comment_id	<pk>	numeric(10)	not null
u_id		varchar(15)	not null
type		varchar(8)	not null
timestamp		datetime	not null
comment_text		text	null
ui_org		varchar(15)	null

ui_msg_header			
ui_msg_header_id	<pk>	numeric(20)	not null
queue_id		char(1)	not null
errors		numeric(10)	not null
create_date		datetime	not null
create_userid		varchar(20)	not null

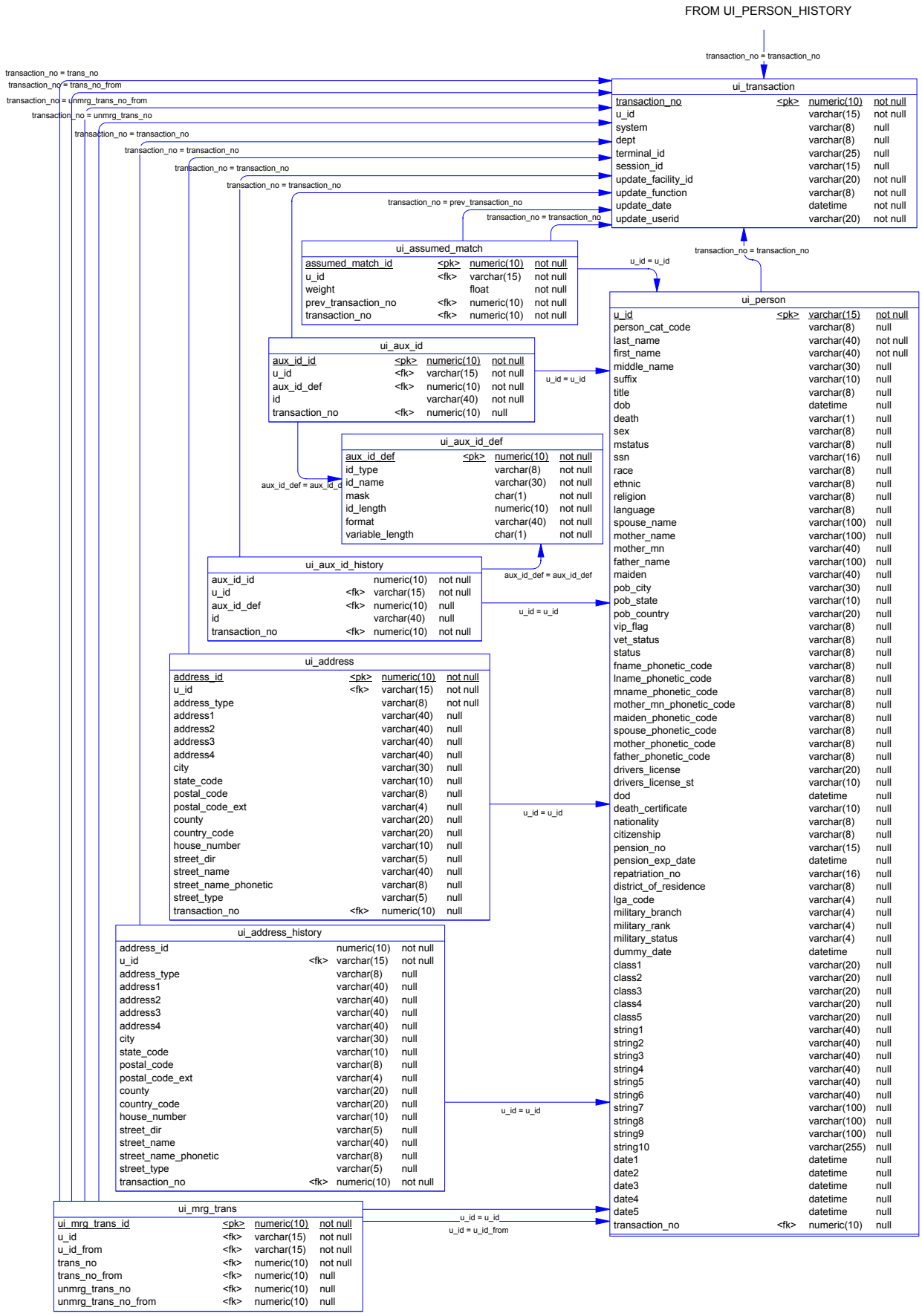
ui_msg_header_id = ui_msg_header_id

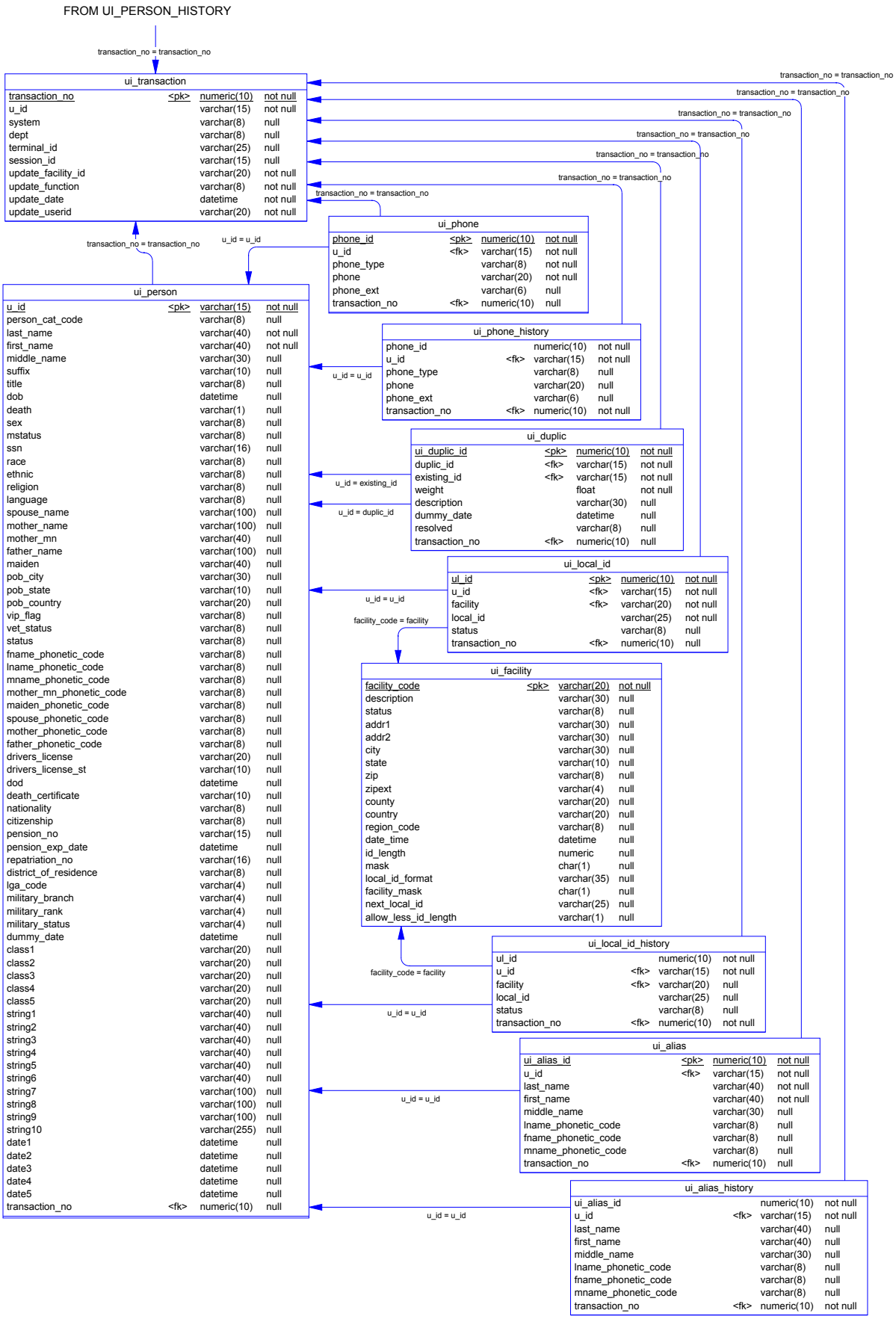
ui_msg_detail			
ui_msg_detail_id	<pk>	numeric(20)	not null
ui_msg_header_id	<fk>	numeric(20)	null
msg		varchar(255)	not null

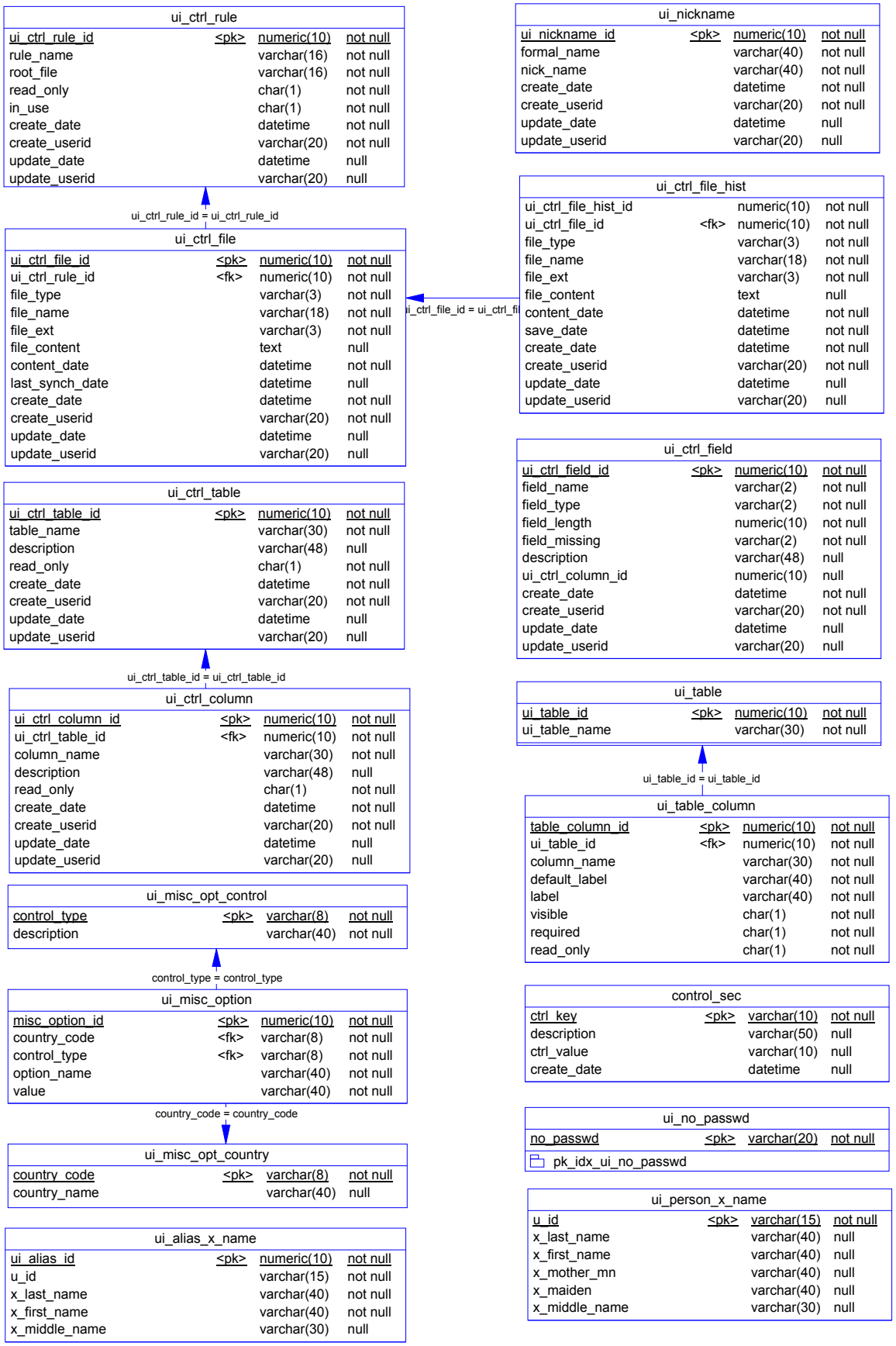
ui_person_history			
ui_person_history_id	<pk>	numeric(10)	not null
u_id		varchar(15)	not null
person_cat_code		varchar(8)	null
last_name		varchar(40)	null
first_name		varchar(40)	null
middle_name		varchar(30)	null
suffix		varchar(10)	null
title		varchar(8)	null
dob		datetime	null
death		varchar(1)	null
sex		varchar(8)	null
mstatus		varchar(8)	null
ssn		varchar(16)	null
race		varchar(8)	null
ethnic		varchar(8)	null
religion		varchar(8)	null
language		varchar(8)	null
spouse_name		varchar(100)	null
mother_name		varchar(100)	null
mother_mn		varchar(40)	null
father_name		varchar(100)	null
maiden		varchar(40)	null
pob_city		varchar(30)	null
pob_state		varchar(10)	null
pob_country		varchar(20)	null
vip_flag		varchar(8)	null
vet_status		varchar(8)	null
status		varchar(8)	null
fname_phonetic_code		varchar(8)	null
lname_phonetic_code		varchar(8)	null
mname_phonetic_code		varchar(8)	null
mother_mn_phonetic_code		varchar(8)	null
maiden_phonetic_code		varchar(8)	null
spouse_phonetic_code		varchar(8)	null
mother_phonetic_code		varchar(8)	null
father_phonetic_code		varchar(8)	null
drivers_license		varchar(20)	null
drivers_license_st		varchar(10)	null
dod		datetime	null
death_certificate		varchar(10)	null
nationality		varchar(8)	null
citizenship		varchar(8)	null
pension_no		varchar(15)	null
pension_exp_date		datetime	null
repatriation_no		varchar(16)	null
district_of_residence		varchar(8)	null
lga_code		varchar(4)	null
military_branch		varchar(4)	null
military_rank		varchar(4)	null
military_status		varchar(4)	null
dummy_date		datetime	null
class1		varchar(20)	null
class2		varchar(20)	null
class3		varchar(20)	null
class4		varchar(20)	null
class5		varchar(20)	null
string1		varchar(40)	null
string2		varchar(40)	null
string3		varchar(40)	null
string4		varchar(40)	null
string5		varchar(40)	null
string6		varchar(40)	null
string7		varchar(100)	null
string8		varchar(100)	null
string9		varchar(100)	null
string10		varchar(255)	null
date1		datetime	null
date2		datetime	null
date3		datetime	null
date4		datetime	null
date5		datetime	null
transaction_no	<fk>	numeric(10)	not null

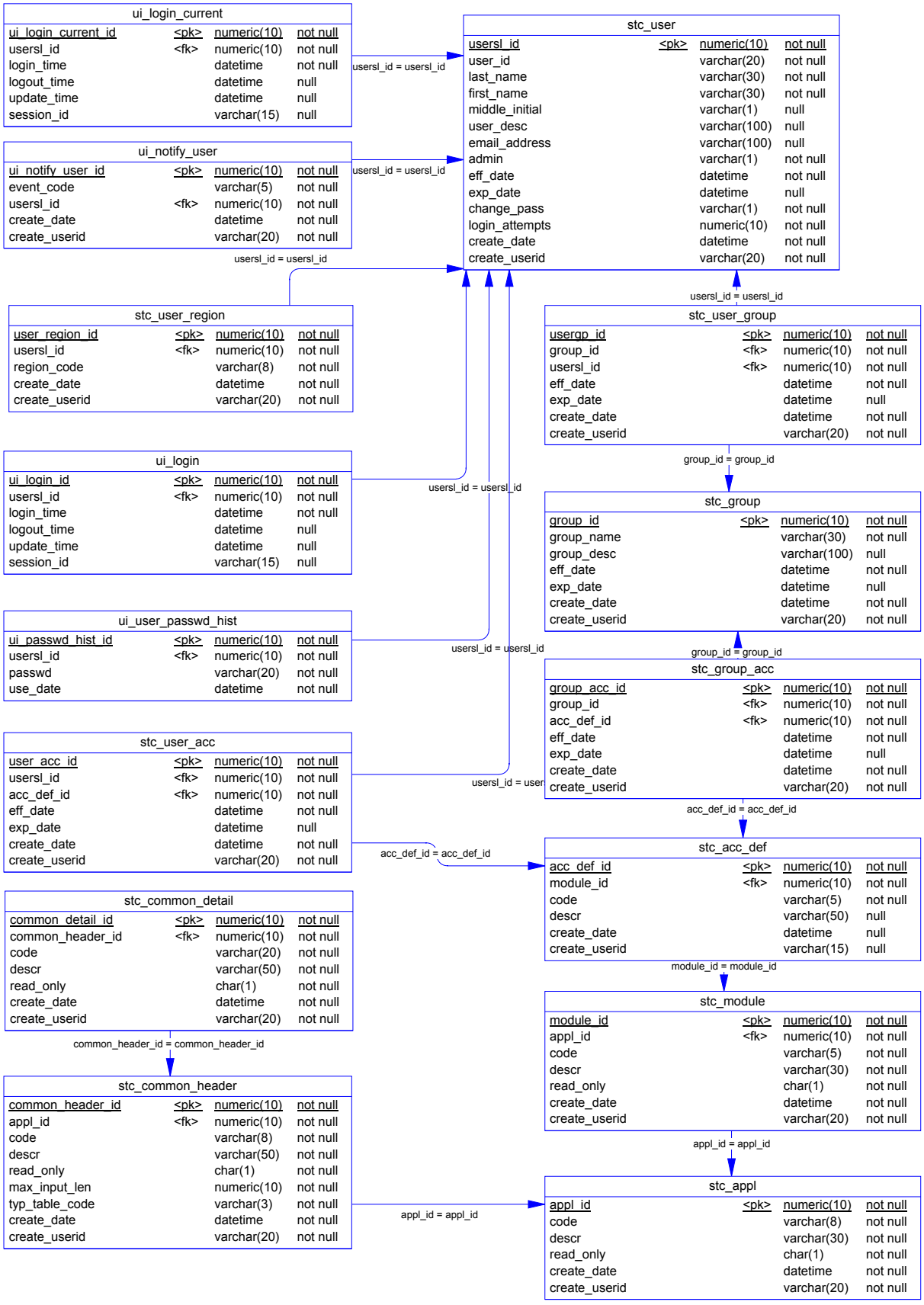
transaction_no = transaction_no

TO UI_TRANSACTION


FROM UI_PERSON_HISTORY
transaction_no = transaction_no
transaction_no = trans_no
transaction_no = trans_no_from
transaction_no = unmrg_trans_no_from
transaction_no = unmrg_trans_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = prev_transaction_no
transaction_no = transaction_no
ui_transaction
transaction_no <pk> numeric(10) not null
u_id varchar(15) not null
system varchar(8) null
dept varchar(8) null
terminal_id varchar(25) null
session_id varchar(15) null
update_facility_id varchar(20) not null
update_function varchar(8) not null
update_date datetime not null
update_userid varchar(20) not null
ui_assumed_match
assumed_match_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
weight float not null
prev_transaction_no <fk> numeric(10) not null
transaction_no <fk> numeric(10) not null
u_id = u_id
transaction_no = transaction_no
ui_person
u_id <pk> varchar(15) not null
person_cat_code varchar(8) null
last_name varchar(40) not null
first_name varchar(40) not null
middle_name varchar(30) null
suffix varchar(10) null
title varchar(8) null
dob datetime null
death varchar(1) null
sex varchar(8) null
mstatus varchar(8) null
ssn varchar(16) null
race varchar(8) null
ethnic varchar(8) null
religion varchar(8) null
language varchar(8) null
spouse_name varchar(100) null
mother_name varchar(100) null
mother_mn varchar(40) null
father_name varchar(100) null
maiden varchar(40) null
pob_city varchar(30) null
pob_state varchar(10) null
pob_country varchar(20) null
vip_flag varchar(8) null
vet_status varchar(8) null
status varchar(8) null
fname_phonetic_code varchar(8) null
lname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
mother_mn_phonetic_code varchar(8) null
maiden_phonetic_code varchar(8) null
spouse_phonetic_code varchar(8) null
mother_phonetic_code varchar(8) null
father_phonetic_code varchar(8) null
drivers_license varchar(20) null
drivers_license_st varchar(10) null
dod datetime null
death_certificate varchar(10) null
nationality varchar(8) null
citizenship varchar(8) null
pension_no varchar(15) null
pension_exp_date datetime null
repatriation_no varchar(16) null
district_of_residence varchar(8) null
lga_code varchar(4) null
military_branch varchar(4) null
military_rank varchar(4) null
military_status varchar(4) null
dummy_date datetime null
class1 varchar(20) null
class2 varchar(20) null
class3 varchar(20) null
class4 varchar(20) null
class5 varchar(20) null
string1 varchar(40) null
string2 varchar(40) null
string3 varchar(40) null
string4 varchar(40) null
string5 varchar(40) null
string6 varchar(40) null
string7 varchar(100) null
string8 varchar(100) null
string9 varchar(100) null
string10 varchar(255) null
date1 datetime null
date2 datetime null
date3 datetime null
date4 datetime null
date5 datetime null
transaction_no <fk> numeric(10) null
ui_aux_id
aux_id_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
aux_id_def <fk> numeric(10) not null
id varchar(40) not null
transaction_no <fk> numeric(10) null
u_id = u_id
ui_aux_id_def
aux_id_def <pk> numeric(10) not null
id_type varchar(8) not null
id_name varchar(30) not null
mask char(1) not null
id_length numeric(10) not null
format varchar(40) not null
variable_length char(1) not null
aux_id_def = aux_id_def
ui_aux_id_history
aux_id_id numeric(10) not null
u_id <fk> varchar(15) not null
aux_id_def <fk> numeric(10) null
id varchar(40) null
transaction_no <fk> numeric(10) not null
aux_id_def = aux_id_def
u_id = u_id
ui_address
address_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
address_type varchar(8) not null
address1 varchar(40) null
address2 varchar(40) null
address3 varchar(40) null
address4 varchar(40) null
city varchar(30) null
state_code varchar(10) null
postal_code varchar(8) null
postal_code_ext varchar(4) null
county varchar(20) null
country_code varchar(20) null
house_number varchar(10) null
street_dir varchar(5) null
street_name varchar(40) null
street_name_phonetic varchar(8) null
street_type varchar(5) null
transaction_no <fk> numeric(10) null
u_id = u_id
ui_address_history
address_id numeric(10) not null
u_id <fk> varchar(15) not null
address_type varchar(8) null
address1 varchar(40) null
address2 varchar(40) null
address3 varchar(40) null
address4 varchar(40) null
city varchar(30) null
state_code varchar(10) null
postal_code varchar(8) null
postal_code_ext varchar(4) null
county varchar(20) null
country_code varchar(20) null
house_number varchar(10) null
street_dir varchar(5) null
street_name varchar(40) null
street_name_phonetic varchar(8) null
street_type varchar(5) null
transaction_no <fk> numeric(10) not null
u_id = u_id
ui_mrg_trans
ui_mrg_trans_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
u_id_from <fk> varchar(15) not null
trans_no <fk> numeric(10) not null
trans_no_from <fk> numeric(10) null
unmrg_trans_no <fk> numeric(10) null
unmrg_trans_no_from <fk> numeric(10) null
u_id = u_id
u_id = u_id_from


FROM UI_PERSON_HISTORY
transaction_no = transaction_no
ui_transaction
transaction_no <pk> numeric(10) not null
u_id varchar(15) not null
system varchar(8) null
dept varchar(8) null
terminal_id varchar(25) null
session_id varchar(15) null
update_facility_id varchar(20) not null
update_function varchar(8) not null
update_date datetime not null
update_userid varchar(20) not null
ui_phone
phone_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
phone_type varchar(8) not null
phone varchar(20) not null
phone_ext varchar(6) null
transaction_no <fk> numeric(10) null
u_id = u_id
ui_person
u_id <pk> varchar(15) not null
person_cat_code varchar(8) null
last_name varchar(40) not null
first_name varchar(40) not null
middle_name varchar(30) null
suffix varchar(10) null
title varchar(8) null
dob datetime null
death varchar(1) null
sex varchar(8) null
mstatus varchar(8) null
ssn varchar(16) null
race varchar(8) null
ethnic varchar(8) null
religion varchar(8) null
language varchar(8) null
spouse_name varchar(100) null
mother_name varchar(100) null
mother_mn varchar(40) null
father_name varchar(100) null
maiden varchar(40) null
pob_city varchar(30) null
pob_state varchar(10) null
pob_country varchar(20) null
vip_flag varchar(8) null
vet_status varchar(8) null
status varchar(8) null
fname_phonetic_code varchar(8) null
lname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
mother_mn_phonetic_code varchar(8) null
maiden_phonetic_code varchar(8) null
spouse_phonetic_code varchar(8) null
mother_phonetic_code varchar(8) null
father_phonetic_code varchar(8) null
drivers_license varchar(20) null
drivers_license_st varchar(10) null
dod datetime null
death_certificate varchar(10) null
nationality varchar(8) null
citizenship varchar(8) null
pension_no varchar(15) null
pension_exp_date datetime null
repatriation_no varchar(16) null
district_of_residence varchar(8) null
lga_code varchar(4) null
military_branch varchar(4) null
military_rank varchar(4) null
military_status varchar(4) null
dummy_date datetime null
class1 varchar(20) null
class2 varchar(20) null
class3 varchar(20) null
class4 varchar(20) null
class5 varchar(20) null
string1 varchar(40) null
string2 varchar(40) null
string3 varchar(40) null
string4 varchar(40) null
string5 varchar(40) null
string6 varchar(40) null
string7 varchar(100) null
string8 varchar(100) null
string9 varchar(100) null
string10 varchar(255) null
date1 datetime null
date2 datetime null
date3 datetime null
date4 datetime null
date5 datetime null
transaction_no <fk> numeric(10) null
ui_phone_history
phone_id numeric(10) not null
u_id <fk> varchar(15) not null
phone_type varchar(8) null
phone varchar(20) null
phone_ext varchar(6) null
transaction_no <fk> numeric(10) not null
ui_duplic
ui_duplic_id <pk> numeric(10) not null
duplic_id <fk> varchar(15) not null
existing_id <fk> varchar(15) not null
weight float not null
description varchar(30) null
dummy_date datetime null
resolved varchar(8) null
transaction_no <fk> numeric(10) null
u_id = existing_id
u_id = duplic_id
ui_local_id
ul_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
facility <fk> varchar(20) not null
local_id varchar(25) not null
status varchar(8) null
transaction_no <fk> numeric(10) null
u_id = u_id
facility_code = facility
ui_facility
facility_code <pk> varchar(20) not null
description varchar(30) null
status varchar(8) null
addr1 varchar(30) null
addr2 varchar(30) null
city varchar(30) null
state varchar(10) null
zip varchar(8) null
zipext varchar(4) null
county varchar(20) null
country varchar(20) null
region_code varchar(8) null
date_time datetime null
id_length numeric null
mask char(1) null
local_id_format varchar(35) null
facility_mask char(1) null
next_local_id varchar(25) null
allow_less_id_length varchar(1) null
ui_local_id_history
ul_id numeric(10) not null
u_id <fk> varchar(15) not null
facility <fk> varchar(20) null
local_id varchar(25) null
status varchar(8) null
transaction_no <fk> numeric(10) not null
ui_alias
ui_alias_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
last_name varchar(40) not null
first_name varchar(40) not null
middle_name varchar(30) null
lname_phonetic_code varchar(8) null
fname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
transaction_no <fk> numeric(10) null
ui_alias_history
ui_alias_id numeric(10) not null
u_id <fk> varchar(15) not null
last_name varchar(40) null
first_name varchar(40) null
middle_name varchar(30) null
lname_phonetic_code varchar(8) null
fname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
transaction_no <fk> numeric(10) not null


ui_ctrl_rule
ui_ctrl_rule_id <pk> numeric(10) not null
rule_name varchar(16) not null
root_file varchar(16) not null
read_only char(1) not null
in_use char(1) not null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_ctrl_rule_id = ui_ctrl_rule_id
ui_ctrl_file
ui_ctrl_file_id <pk> numeric(10) not null
ui_ctrl_rule_id <fk> numeric(10) not null
file_type varchar(3) not null
file_name varchar(18) not null
file_ext varchar(3) not null
file_content text null
content_date datetime not null
last_synch_date datetime null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_ctrl_table
ui_ctrl_table_id <pk> numeric(10) not null
table_name varchar(30) not null
description varchar(48) null
read_only char(1) not null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_ctrl_table_id = ui_ctrl_table_id
ui_ctrl_column
ui_ctrl_column_id <pk> numeric(10) not null
ui_ctrl_table_id <fk> numeric(10) not null
column_name varchar(30) not null
description varchar(48) null
read_only char(1) not null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_misc_opt_control
control_type <pk> varchar(8) not null
description varchar(40) not null
control_type = control_type
ui_misc_option
misc_option_id <pk> numeric(10) not null
country_code <fk> varchar(8) not null
control_type <fk> varchar(8) not null
option_name varchar(40) not null
value varchar(40) not null
country_code = country_code
ui_misc_opt_country
country_code <pk> varchar(8) not null
country_name varchar(40) null
ui_alias_x_name
ui_alias_id <pk> numeric(10) not null
u_id varchar(15) not null
x_last_name varchar(40) not null
x_first_name varchar(40) not null
x_middle_name varchar(30) null
ui_nickname
ui_nickname_id <pk> numeric(10) not null
formal_name varchar(40) not null
nick_name varchar(40) not null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_ctrl_file_hist
ui_ctrl_file_hist_id numeric(10) not null
ui_ctrl_file_id <fk> numeric(10) not null
file_type varchar(3) not null
file_name varchar(18) not null
file_ext varchar(3) not null
file_content text null
content_date datetime not null
save_date datetime not null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_ctrl_field
ui_ctrl_field_id <pk> numeric(10) not null
field_name varchar(2) not null
field_type varchar(2) not null
field_length numeric(10) not null
field_missing varchar(2) not null
description varchar(48) null
ui_ctrl_column_id numeric(10) null
create_date datetime not null
create_userid varchar(20) not null
update_date datetime null
update_userid varchar(20) null
ui_table
ui_table_id <pk> numeric(10) not null
ui_table_name varchar(30) not null
ui_table_id = ui_table_id
ui_table_column
table_column_id <pk> numeric(10) not null
ui_table_id <fk> numeric(10) not null
column_name varchar(30) not null
default_label varchar(40) not null
label varchar(40) not null
visible char(1) not null
required char(1) not null
read_only char(1) not null
control_sec
ctrl_key <pk> varchar(10) not null
description varchar(50) null
ctrl_value varchar(10) null
create_date datetime null
ui_no_passwd
no_passwd <pk> varchar(20) not null
pk_idx_ui_no_passwd
ui_person_x_name
u_id <pk> varchar(15) not null
x_last_name varchar(40) null
x_first_name varchar(40) null
x_mother_mn varchar(40) null
x_maiden varchar(40) null
x_middle_name varchar(30) null


ui_login_current
ui_login_current_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
login_time datetime not null
logout_time datetime null
update_time datetime null
session_id varchar(15) null
usersl_id = usersl_id
stc_user
usersl_id <pk> numeric(10) not null
user_id varchar(20) not null
last_name varchar(30) not null
first_name varchar(30) not null
middle_initial varchar(1) null
user_desc varchar(100) null
email_address varchar(100) null
admin varchar(1) not null
eff_date datetime not null
exp_date datetime null
change_pass varchar(1) not null
login_attempts numeric(10) not null
create_date datetime not null
create_userid varchar(20) not null
ui_notify_user
ui_notify_user_id <pk> numeric(10) not null
event_code varchar(5) not null
usersl_id <fk> numeric(10) not null
create_date datetime not null
create_userid varchar(20) not null
usersl_id = usersl_id
usersl_id = usersl_id
stc_user_region
user_region_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
region_code varchar(8) not null
create_date datetime not null
create_userid varchar(20) not null
usersl_id = usersl_id
stc_user_group
usergp_id <pk> numeric(10) not null
group_id <fk> numeric(10) not null
usersl_id <fk> numeric(10) not null
eff_date datetime not null
exp_date datetime null
create_date datetime not null
create_userid varchar(20) not null
group_id = group_id
ui_login
ui_login_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
login_time datetime not null
logout_time datetime null
update_time datetime null
session_id varchar(15) null
usersl_id = usersl_id
stc_group
group_id <pk> numeric(10) not null
group_name varchar(30) not null
group_desc varchar(100) null
eff_date datetime not null
exp_date datetime null
create_date datetime not null
create_userid varchar(20) not null
group_id = group_id
ui_user_passwd_hist
ui_passwd_hist_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
passwd varchar(20) not null
use_date datetime not null
usersl_id = usersl_id
stc_group_acc
group_acc_id <pk> numeric(10) not null
group_id <fk> numeric(10) not null
acc_def_id <fk> numeric(10) not null
eff_date datetime not null
exp_date datetime null
create_date datetime not null
create_userid varchar(20) not null
acc_def_id = acc_def_id
stc_user_acc
user_acc_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
acc_def_id <fk> numeric(10) not null
eff_date datetime not null
exp_date datetime null
create_date datetime not null
create_userid varchar(20) not null
usersl_id = user
acc_def_id = acc_def_id
stc_acc_def
acc_def_id <pk> numeric(10) not null
module_id <fk> numeric(10) not null
code varchar(5) not null
descr varchar(50) null
create_date datetime null
create_userid varchar(15) null
module_id = module_id
stc_common_detail
common_detail_id <pk> numeric(10) not null
common_header_id <fk> numeric(10) not null
code varchar(20) not null
descr varchar(50) not null
read_only char(1) not null
create_date datetime not null
create_userid varchar(20) not null
common_header_id = common_header_id
stc_module
module_id <pk> numeric(10) not null
appl_id <fk> numeric(10) not null
code varchar(5) not null
descr varchar(30) not null
read_only char(1) not null
create_date datetime not null
create_userid varchar(20) not null
appl_id = appl_id
stc_common_header
common_header_id <pk> numeric(10) not null
appl_id <fk> numeric(10) not null
code varchar(8) not null
descr varchar(50) not null
read_only char(1) not null
max_input_len numeric(10) not null
typ_table_code varchar(3) not null
create_date datetime not null
create_userid varchar(20) not null
appl_id = appl_id
stc_appl
appl_id <pk> numeric(10) not null
code varchar(8) not null
descr varchar(30) not null
read_only char(1) not null
create_date datetime not null
create_userid varchar(20) not null

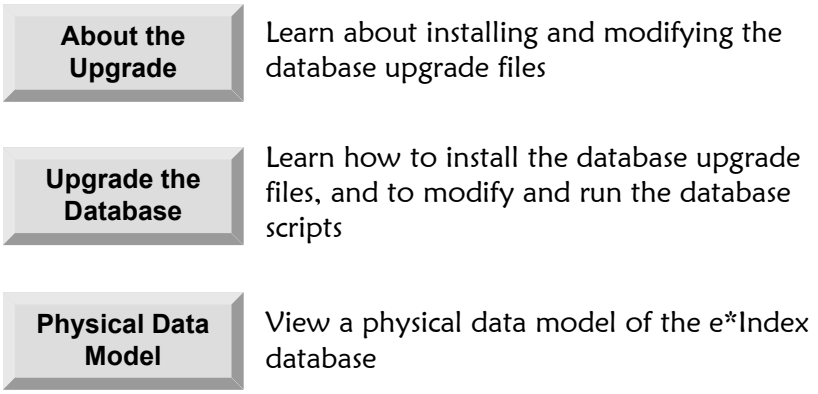
Chapter 5

Upgrading a SQL Server Database

About this Chapter

Overview

This chapter presents the background information and the step-by-step instructions you need to upgrade an e*Index database for Microsoft SQL Server from version 4.5.1 or later to 5.0.5 for SRE. The following diagram illustrates the major topics in this chapter. For the page numbers on which specific topics appear, see the next page of this chapter.



What's Inside

This chapter provides background information and instructions related to the topics listed below.

Learning About the Database Upgrade

Overview

This section of the chapter provides background information about the files you install, modify, and execute to upgrade an e*Index database for Microsoft SQL Server from version 4.5.1 or later to 5.0.5 for SRE.

What do I Need to Know Before I Start?

Make sure you are familiar with your Microsoft SQL Server environment before beginning the upgrade procedure. Know the server name and pathname in which the e*Index database resides, and the name, login ID, and password of the database you are upgrading. It is helpful to be familiar with your SQL Server environment, SQL Server networking, and SQL Server database administration. Also, determine whether you have region-specific security installed, and, if not, whether you want to install it (for more information see chapter 3 of the *e*Index Administrator User's Guide* and the *e*Index Security User's Guide*).

e*Index is only compatible with SQL Server 7.0. If you are using a previous version of SQL Server, upgrade your database to SQL Server 7.0 before upgrading e*Index.

How is the Database Upgraded?

Installing the database files creates several SQL scripts used to upgrade the existing database; however, you only need to execute one batch file, **upgrade.bat**, to perform the upgrade. The upgrade batch file calls the SQL scripts to make the necessary changes to the database. This file requires some modification prior to execution. It must be run from a machine running a Windows operating system (XP, 2000, or 2003) with Microsoft SQL Server client files installed. You can run the upgrade file from an existing e*Index client workstation or from the database server.

If you plan to install region-specific security, you need to modify and run a second file, **install_ssap.bat**. Note that this process can only upgrade a database that is already at version 4.5.1 or higher.

Performing the Upgrade

Overview

To upgrade an e*Index database on Microsoft SQL Server, you must complete the following steps:

- Step 1: Obtain Database Information
- Step 2: Back up the Current Database
- Step 3: Upgrade SQL Server
- Step 4: Install the Database and Report Files
- Step 5: Verify the ODBC Data Source
- Step 6: Modify the Upgrade File
- Step 7: Modify **install_ssap.bat** (optional)
- Step 8: Upgrade the Database
- Step 9: Run **install_ssap.bat** (optional)

Note: *If you chose to install all components of e*Index when you installed the e*Index Schema files, you should have already completed steps 1 through 4. You can begin with "Step 5: Verify the ODBC Data Source".*

Step 1: Obtain Database Information

Before beginning, gather information about your database, such as the database path, the database name, and the system login and password for your database. You should also know the path to your SQL Server home directory. If you customized the rule set files for the Vality matching algorithm, know the location of the most current rule set files. If you currently do not have region-specific security installed, determine whether to install it now. Region-specific security is described in Chapter 3 of the *e*Index Security User's Guide* and the *e*Index Administrator User's Guide*.

Because of the complex nature of modifying a database, we recommend that a database administrator perform the following steps.

Step 2: Back up the Current Database

Prior to making any changes to your e*Index database, make a complete backup of your current database. For more information on performing SQL Server database backups, see the appropriate Microsoft SQL Server documentation.

Step 3: Upgrade Microsoft SQL Server

Before you install the e*Index database files, Microsoft SQL Server 7.0 must be installed on the database server. If you are currently running a previous version of SQL Server, you need to upgrade SQL Server before continuing. If your SQL Server home directory changes during the upgrade, make a note of the new path. For information about upgrading SQL Server, refer to the appropriate upgrade documentation for SQL Server.

Important! *It is crucial that the Microsoft SQL Server installation is operational prior to performing the following steps.*

Step 4: Install the Database and Report Files

Installing the database and report files is very similar to the process you followed to install the e*Index schema files.

▶ To install the database and report files

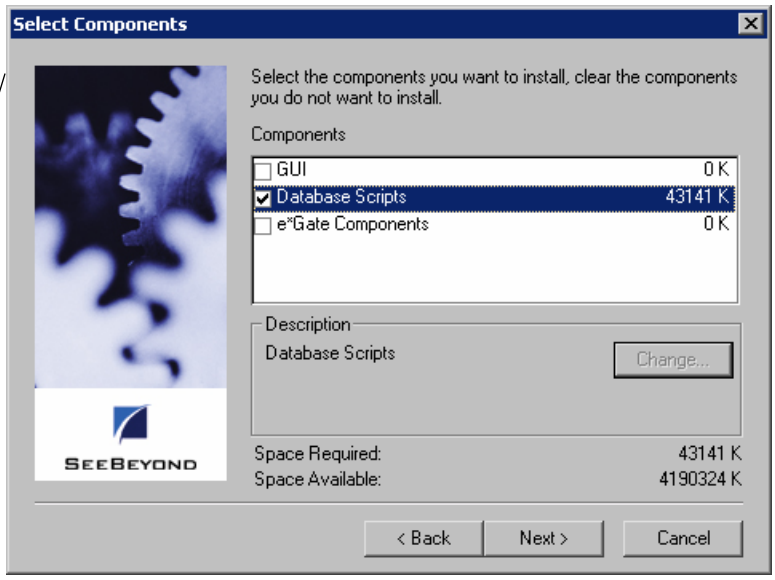
Before you begin:

- ✓ Complete "Step 3: Upgrade Microsoft SQL Server"
- ✓ Make sure the e*Index installation CD-ROM is inserted into the CD-ROM drive of your database server

1. Follow steps 1 through 7 under "Step 5: Install the e*Index Schema Files" in Chapter 2, "Installing the e*Index Schema Files". The Select Components window should now be visible.

2. On the Select Components window, select the check box next to **Database Scripts**.

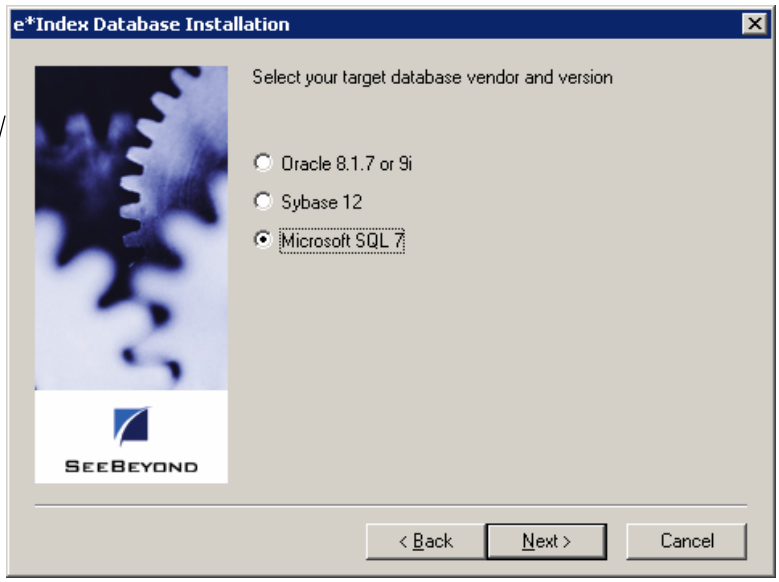
Select the components you want to install on the Select Components window



Next button

3 Click **Next**. The e*Index Database Installation window appears.

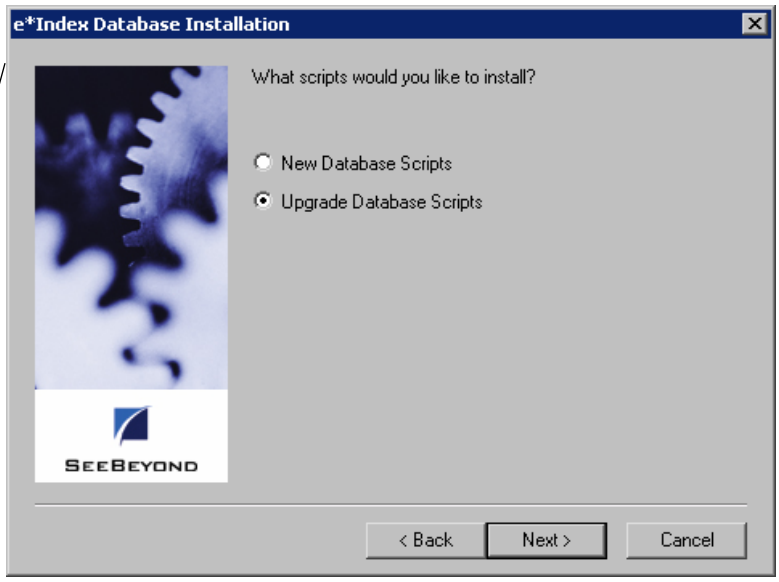
On the e*Index Database Installation window, specify Microsoft SQL Server for the database platform



Next button

4 Select **Microsoft SQL 7**, and then click **Next**. A second e*Index Database Installation window appears.

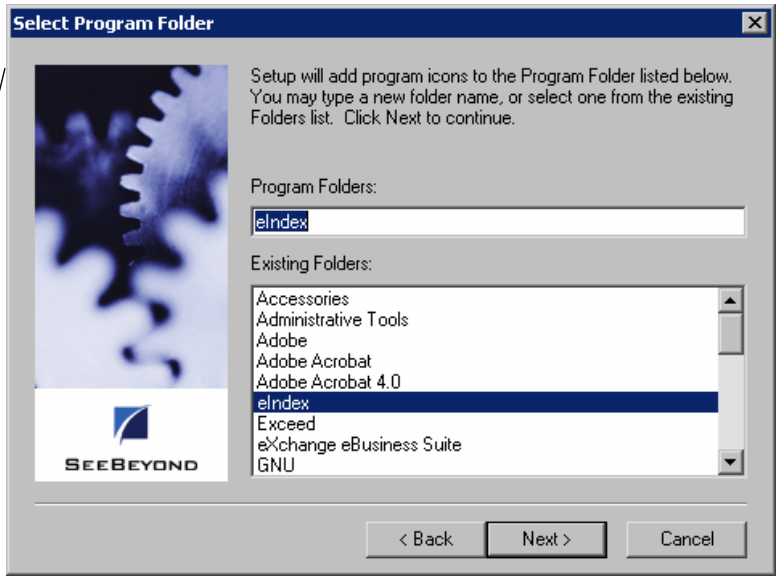
Specify whether to install the scripts to install a new database or the scripts to upgrade an existing database



Next button

5 Select **New Database Scripts**, and then click **Next**. The Select Program Folder window appears.

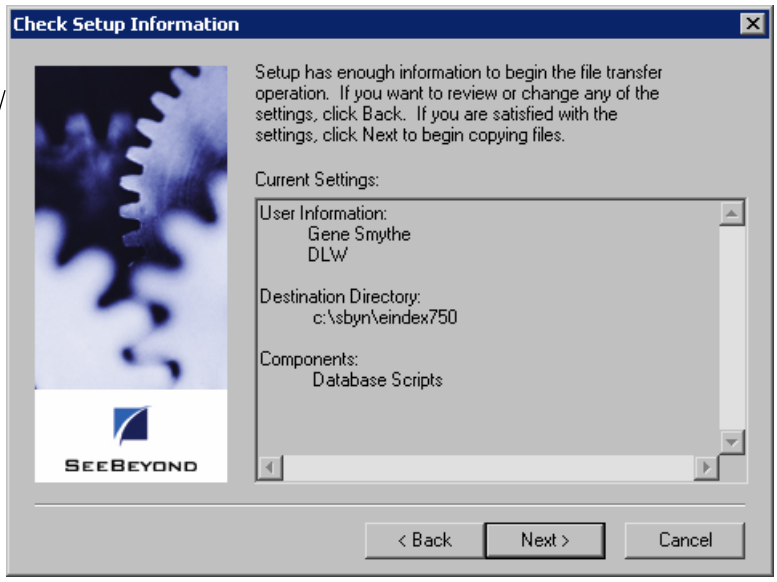
Specify the folder in which to install the program icons on the Select Program Folder window



Next button

6 Enter the name of the folder into which you want to install the program icons or accept the default name, and then click **Next**. The Check Setup Information window appears.

Verify the information you specified on the Check Setup Information window



7 Verify the information you specified, and do one of the following:

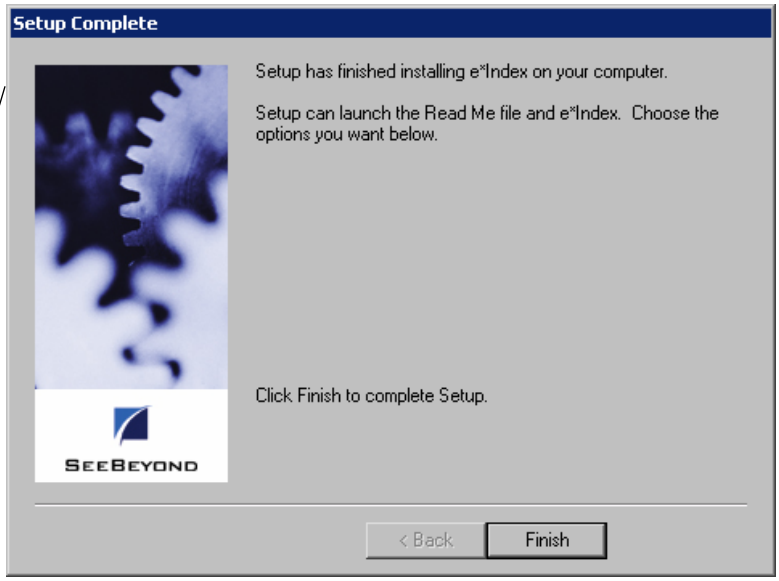
Back button

To change any of the options you selected, click **Back**, and make the necessary changes.

Next button

To continue with the installation, click **Next**. When all files are installed, the Setup Complete window appears.

The Setup Complete window indicates that the database files have been installed



Finish button

8 Click **Finish** to complete the setup process.

9 To view the database files that were installed:

- Open Windows Explorer and navigate to the path you specified for the installation files.
- Open the **\server\DBcommon** subdirectory. You will find two subdirectories, **Server** and **Client**, and one file, **defs.sql**. The Client directory contains the batch files, **upgrade.bat**, **install_ssap.bat**, and **remove_ssap.bat**. The **Server** directory contains the files you need to create the database instance on your server.

10 Continue to "Step 5: Verify the ODBC Data Source".

Step 5: Verify the ODBC Data Source

Verify that an ODBC data source for the e*Index database has been defined on the computer you are using for the upgrade. The data source specifies the driver used to connect to the database and additional information about the database. ODBC data source definitions are accessed through the **Data Sources (ODBC)** option in the Control Panel.

Step 6: Modify the Upgrade File

The file **upgrade.bat** is a batch file that updates the database tables, triggers, indexes, and so on. You need to modify certain variables in this file that tell the script how to locate the database instance and installation scripts.

▶ To modify the upgrade file

Before you begin:

✓ Complete "Step 5: Verify the ODBC Data Source"

1 Navigate to the path where the database files are located on the database server, and then navigate to the **server\DBcommon\Client** directory.

2 Make a backup copy of the file **upgrade.bat**.

3 Right-click the **upgrade.bat** file and then select **Edit** from the list that appears. Do not double-click this file to open it.

4 Make the following modifications:

- In the variable **SERVER**, enter the name of the machine on which the database will reside.
- In the variable **DATABASE**, enter the name of the database.

- In the variable **ODBC_DSN**, enter the name of the ODBC data source you defined in the ODBC Data Source Administrator.
- In the variable **INSTALLDRIVE**, enter the drive on which the database installation files are located.
- In the variable **INSTALL_HOME**, enter the path in which the database files are located. Do not include the drive designation in this path, and do not include the subdirectories **\server\DBcommon\Client**.
- In the variable **SYSTEMPW**, enter the system administrator password for SQL Server.
- Only modify the default value for **UIPW** if you have changed the password for the **UI** e*Index user.

The sample below illustrates the above variables.

```
SET SERVER=SQLSVR
SET DATABASE=ei01
SET ODBC_DSN=EI01
SET INSTALLDRIVE=C:
SET INSTALL_HOME=\TEMP\INSTALL

...
SET SYSTEMPW=eindex
SET UIPW=UI
```

5 Save the changes to **upgrade.bat** and close the file.

6 Do one of the following:

If you want to install region-specific security in the database, Continue to "Step 7: Modify **install_ssap.bat**". If you already have region-specific security installed, you do not need to run this file.

If you do not want to install region-specific security in the database, skip to "Step 8: Upgrade the Database".

Step 7: Modify install_ssap.bat (optional)

Running **install_ssap.bat** installs the views you need in order to use the region-specific security capabilities of e*Index. You do not need to modify or run this file if you do not want to use this capability, or if it was installed when the database was created.

For more information about region-specific security, see "What is Region-Specific Security?" in Chapter 3 of the *e*Index Security User's Guide* and "About Region-Specific Security" in Chapter 3 of the *e*Index Administrator User's Guide*.

▶ To modify install_ssap.bat

Before you begin:

✓ Complete "Step 6: Modify the Upgrade File"

1 Navigate to the path where the database files are located on the database server, and then navigate to the **\server\DBcommon\Client** directory.

2 Make a backup copy of the file **install_ssap.bat**.

3 Right-click the **install_ssap.bat** file and then select **Edit** from the list that appears.

4 Make the following modifications:

- In the variable **SERVER**, **e**nter the name of the machine on which the database will reside.
- In the variable **DATABASE**, enter the name of the database.
- You should not need to modify the default value for the variable **UIPW** unless you changed it after the database was created.

The sample below illustrates the variables you need to modify.

```
SET SERVER=SQLSVR
SET DATABASE=ei01
...
SET UIPW=UI
```

5 Save the changes to **install_ssap.bat** and close the file.

6 Continue to "Step 8: Upgrade the Database".

Step 8: Upgrade the Database

Once you have modified all the necessary files, you can run **upgrade.bat** to update the tables, views, indexes, and so on for your e*Index database.

Important! *Make sure your database is at version 4.5.1 or later before continuing. Otherwise, this step will create and populate new database views, which may take several hours to complete.*

▶ To upgrade the database

Before you begin:

- ✓ Complete "Step 6: Modify the Upgrade File" and, optionally, "Step 7: Modify **install_ssap.bat**"

1. Navigate to the path where the database files are located on the database server, and then navigate to the **server\DBcommon\Client** directory.
2. Double-click **upgrade.bat** to run the batch file.
3. Log files are created in the path you specified for your installation files in **\server\upgrades\Spool**. Review each log file to ensure that there were no errors during the running of the script.
4. At the prompt, press any key on the keyboard to complete the installation.
5. If you want to install region-specific security in the database, continue to "Step 9: Run **install_ssap.bat** (optional)". Otherwise, your database installation is complete.

Step 9: Run install_ssap.bat (optional)

After you have run **upgrade.bat** to update the e*Index database, you can install system-specific security by running **install_ssap.bat**. You do not need to run this file you are not using region-specific security or if it was installed when the database was created.

Important! *Do not execute this file if you do not want to use region-specific security. Creating these views requires that you also define the regions associated with each system and assign them to user profiles before you can access e*Index information. If you install region-specific security but do not want to use the functionality, you can remove it by running* ***remove_ssap.bat****. You need to modify this file in the same way you modified* ***install_ssap.bat*** *in "Step 7: Modify* ***install_ssap.bat****" earlier in this chapter.*

▶ To run install_ssap.bat

Before you begin:

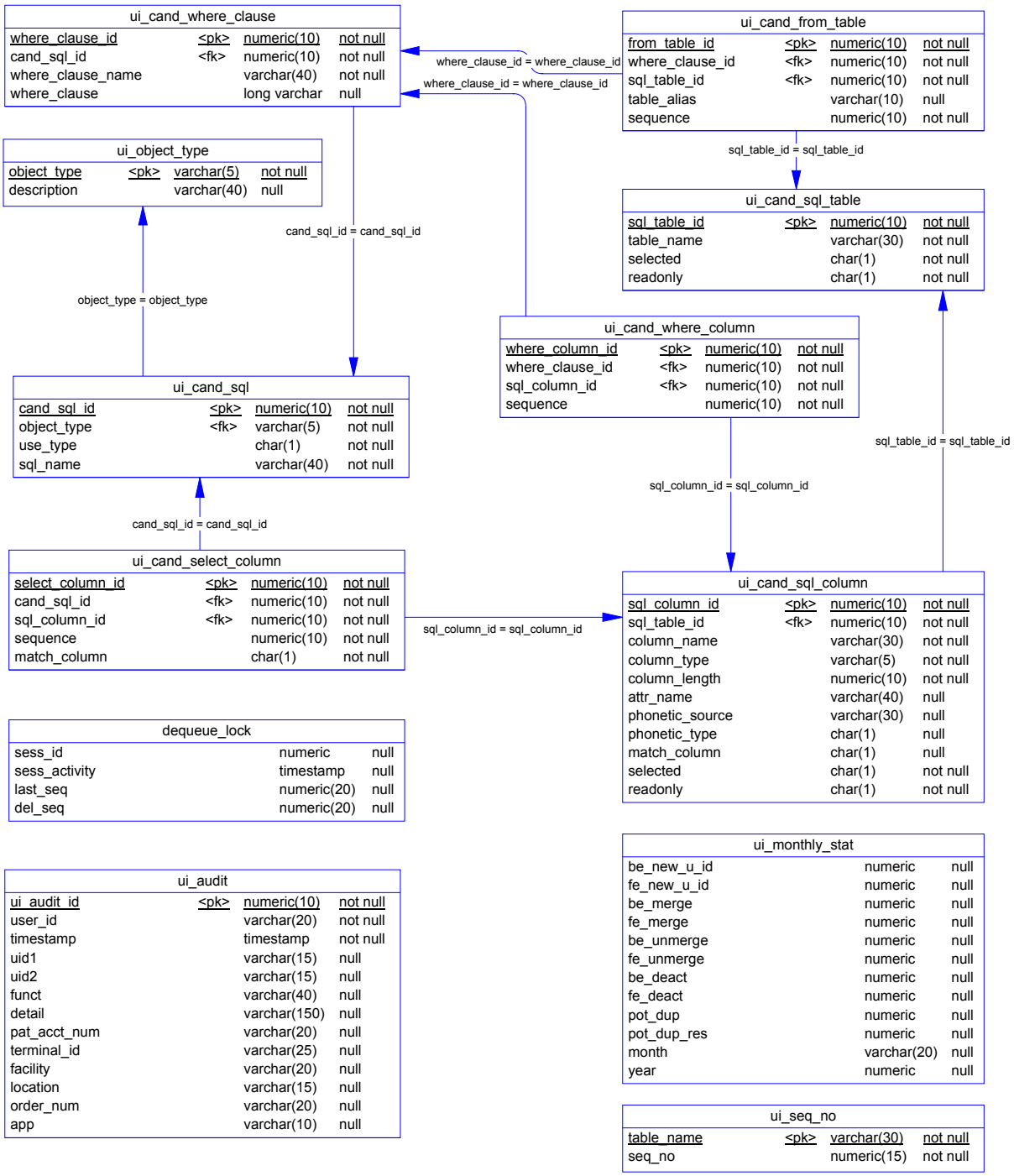
- ✓ Complete "Step 8: Upgrade the Database"

1. Navigate to the path where the database files are located on the database server, and then navigate to the **server\DBcommon\Client** directory.
2. Double-click **install_ssap.bat** to run the batch file.

3 At the prompt, press any key on the keyboard to complete the installation.

e*Index SQL Server Database Model

The diagrams on the following pages illustrate the table structure for e*Index on SQL Server. Your actual database may vary slightly from this model depending on the release you have installed. The *ui_person* and *ui_transaction* tables are each displayed on two different pages to better illustrate the connections to these two tables.



ui_config			
interface	<pk>	varchar(255)	not null
code	<pk>	varchar(255)	not null
value		numeric	not null

ui_control			
ctrl_key	<pk>	varchar(10)	not null
description		varchar(50)	null
ctrl_value		varchar(30)	null
create_date		timestamp	null

ui_dept			
dept_code	<pk>	varchar(5)	not null
description		varchar(20)	null
date_time		timestamp	null

ui_canned_msg			
code	<pk>	varchar(5)	not null
description		varchar(80)	not null
create_date		timestamp	null

ui_message			
code	<pk>	varchar(5)	not null
description		varchar(255)	not null
message_box_header		varchar(50)	not null
icon		varchar(15)	null
button		varchar(20)	null
default_button		numeric(1)	null
message_type		varchar(8)	null
application		varchar(10)	null
date_time		timestamp	null

ui_zip			
zip_code	<pk>	varchar(8)	not null
zip4		varchar(4)	null
city	<pk>	varchar(30)	not null
state	<pk>	varchar(10)	not null
county		varchar(3)	null
residence_code		varchar(4)	null
create_date		timestamp	null

ui_comment			
ui_comment_id	<pk>	numeric(10)	not null
u_id		varchar(15)	not null
type		varchar(8)	not null
timestamp		timestamp	not null
comment_text		long varchar	null
ui_org		varchar(15)	null

ui_msg_header			
ui_msg_header_id	<pk>	numeric(20)	not null
queue_id		char(1)	not null
errors		numeric(10)	not null
create_date		timestamp	not null
create_userid		varchar(20)	not null

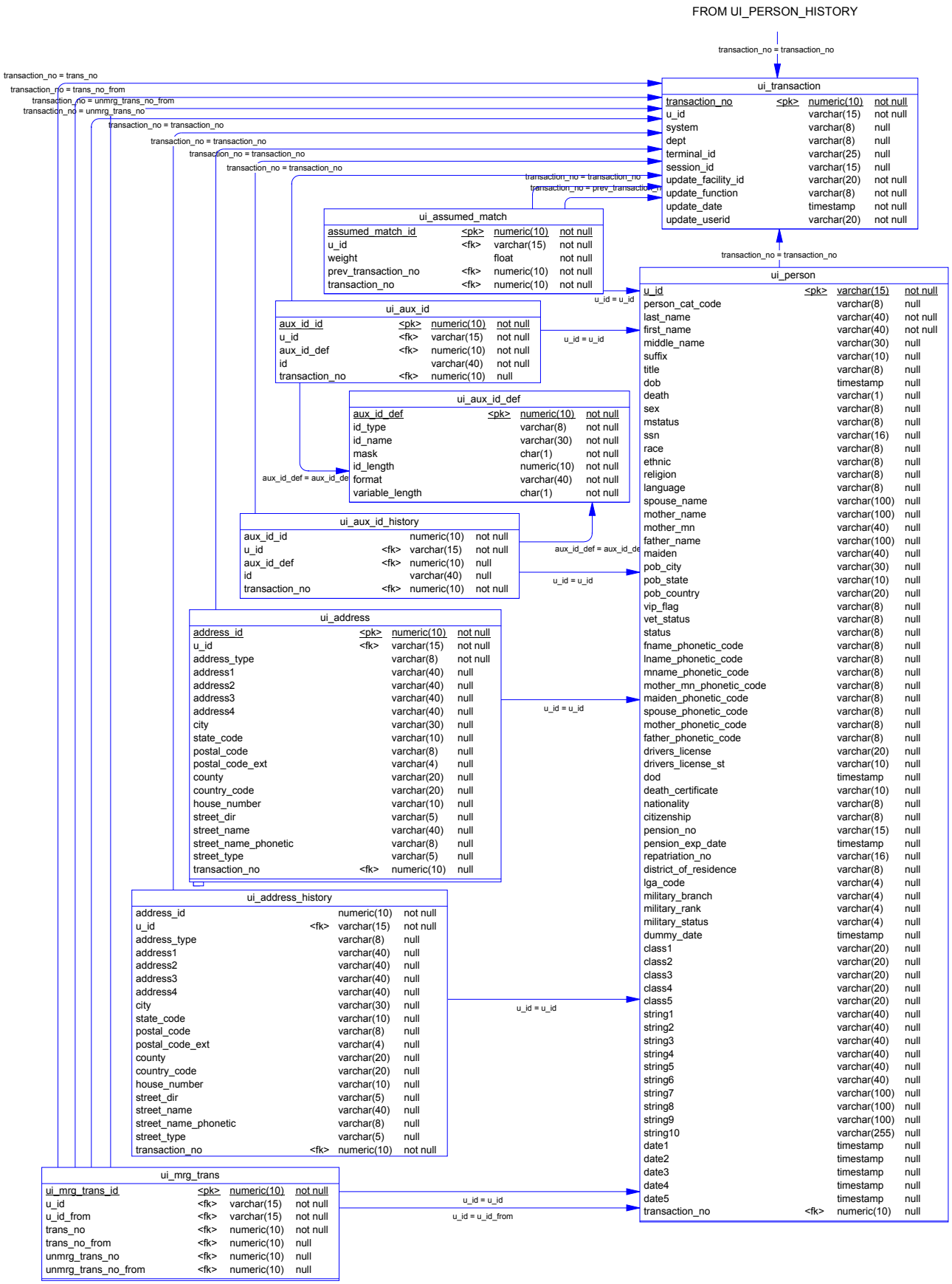
ui_msg_header_id = ui_msg_header_id

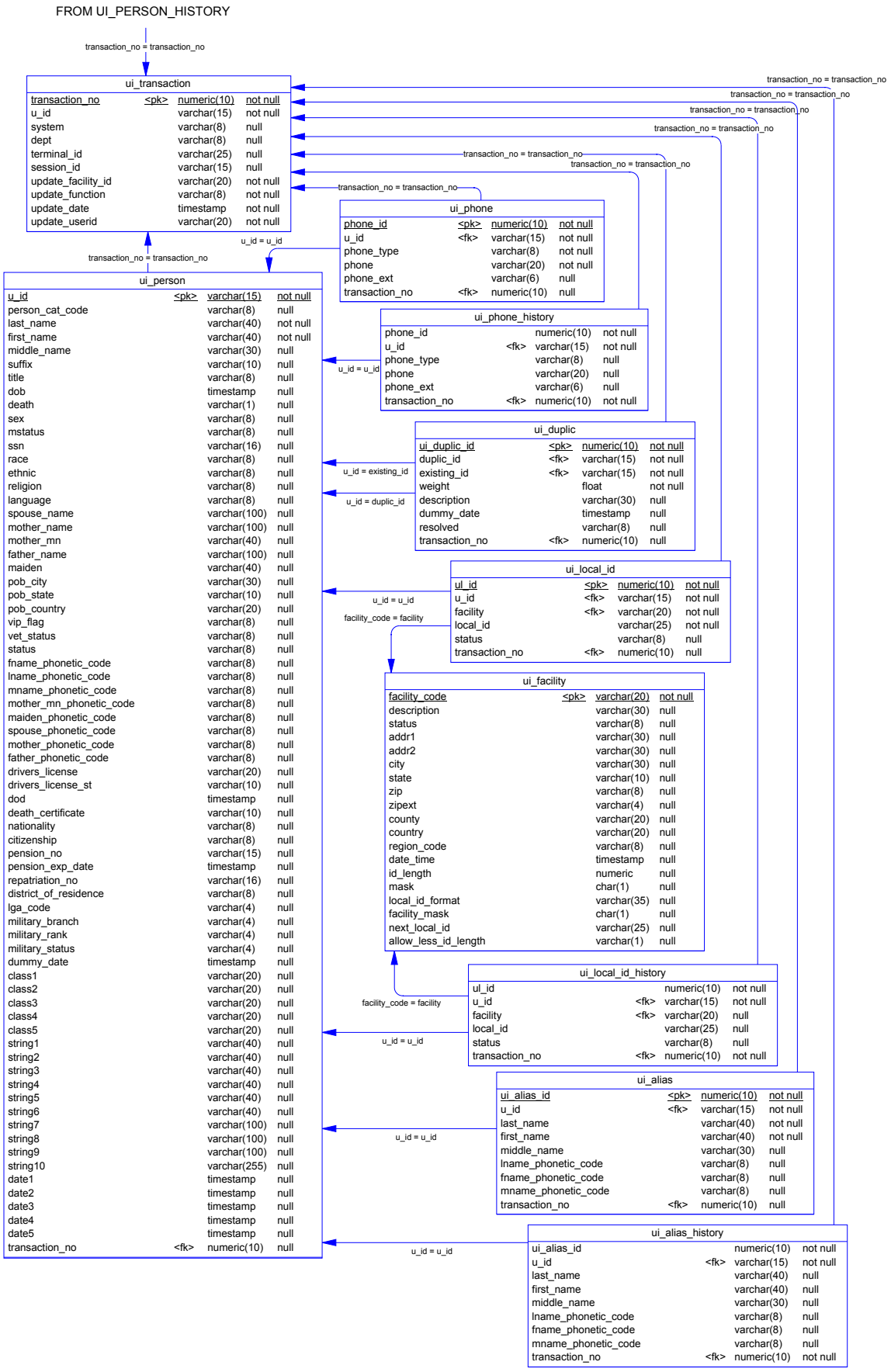
ui_msg_detail			
ui_msg_detail_id	<pk>	numeric(20)	not null
ui_msg_header_id	<fk>	numeric(20)	null
msg		varchar(255)	not null

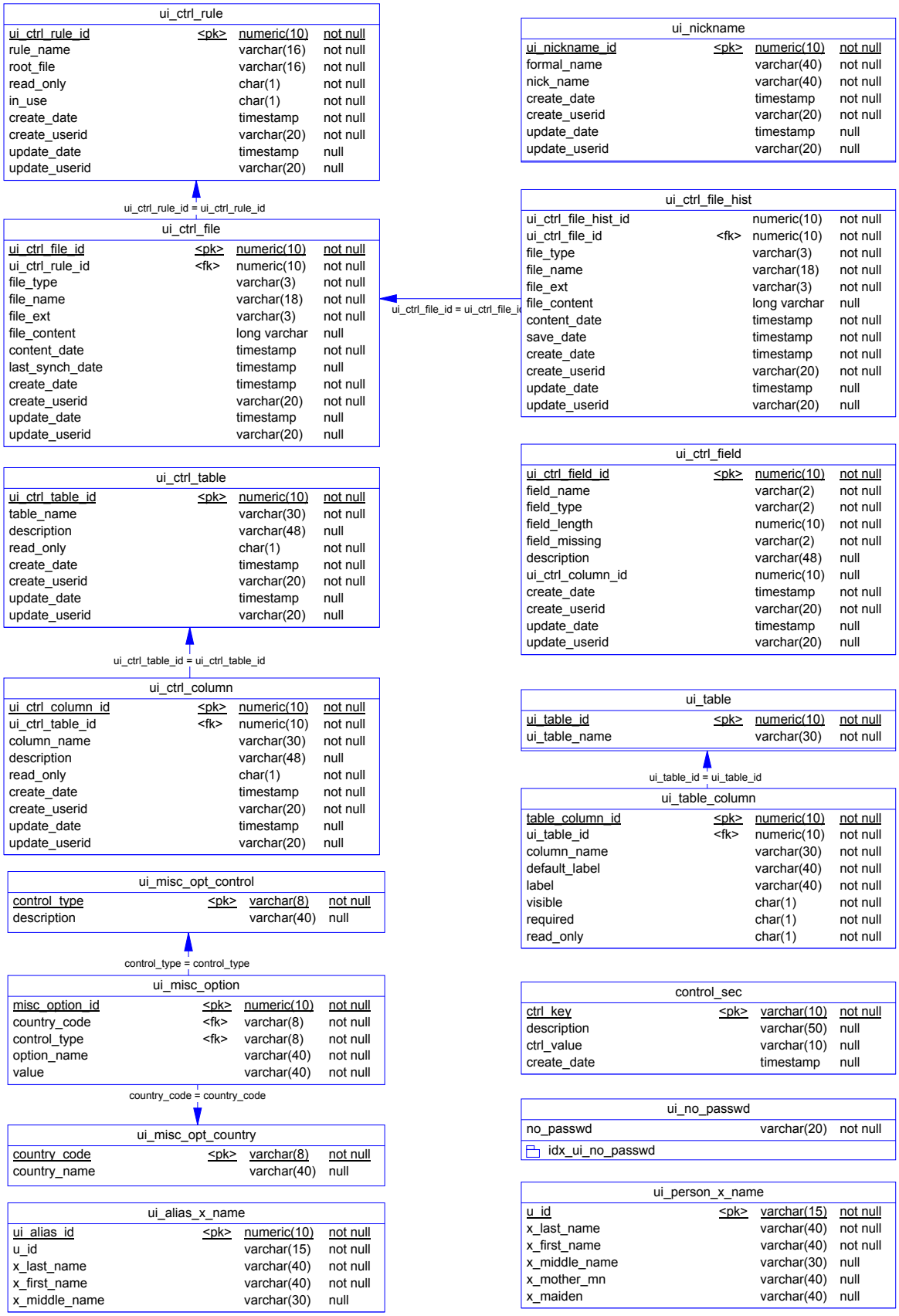
ui_person_history			
ui_person_history_id	<pk>	numeric(10)	not null
u_id		varchar(15)	not null
person_cat_code		varchar(8)	null
last_name		varchar(40)	null
first_name		varchar(40)	null
middle_name		varchar(30)	null
suffix		varchar(10)	null
title		varchar(8)	null
dob		timestamp	null
death		varchar(1)	null
sex		varchar(8)	null
mstatus		varchar(8)	null
ssn		varchar(16)	null
race		varchar(8)	null
ethnic		varchar(8)	null
religion		varchar(8)	null
language		varchar(8)	null
spouse_name		varchar(100)	null
mother_name		varchar(100)	null
mother_mn		varchar(40)	null
father_name		varchar(100)	null
maiden		varchar(40)	null
pob_city		varchar(30)	null
pob_state		varchar(10)	null
pob_country		varchar(20)	null
vip_flag		varchar(8)	null
vet_status		varchar(8)	null
status		varchar(8)	null
fname_phonetic_code		varchar(8)	null
lname_phonetic_code		varchar(8)	null
mname_phonetic_code		varchar(8)	null
mother_mn_phonetic_code		varchar(8)	null
maiden_phonetic_code		varchar(8)	null
spouse_phonetic_code		varchar(8)	null
mother_phonetic_code		varchar(8)	null
father_phonetic_code		varchar(8)	null
drivers_license		varchar(20)	null
drivers_license_st		varchar(10)	null
dod		timestamp	null
death_certificate		varchar(10)	null
nationality		varchar(8)	null
citizenship		varchar(8)	null
pension_no		varchar(15)	null
pension_exp_date		timestamp	null
repatriation_no		varchar(16)	null
district_of_residence		varchar(8)	null
lga_code		varchar(4)	null
military_branch		varchar(4)	null
military_rank		varchar(4)	null
military_status		varchar(4)	null
dummy_date		timestamp	null
class1		varchar(20)	null
class2		varchar(20)	null
class3		varchar(20)	null
class4		varchar(20)	null
class5		varchar(20)	null
string1		varchar(40)	null
string2		varchar(40)	null
string3		varchar(40)	null
string4		varchar(40)	null
string5		varchar(40)	null
string6		varchar(40)	null
string7		varchar(100)	null
string8		varchar(100)	null
string9		varchar(100)	null
string10		varchar(255)	null
date1		timestamp	null
date2		timestamp	null
date3		timestamp	null
date4		timestamp	null
date5		timestamp	null
transaction_no	<fk>	numeric(10)	not null

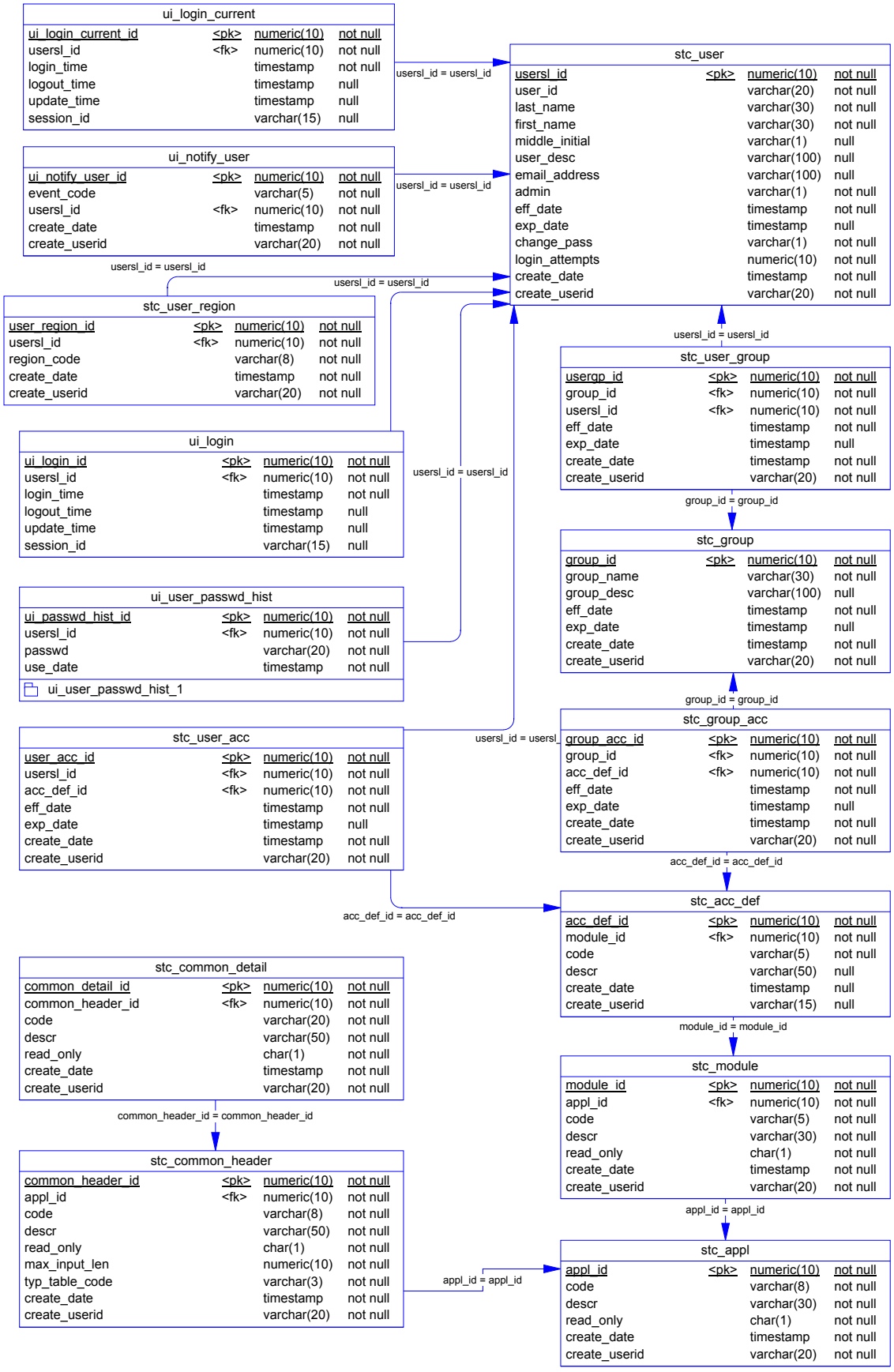
transaction_no = transaction_no

TO UI_TRANSACTION


FROM UI_PERSON_HISTORY
transaction_no = transaction_no
transaction_no = trans_no
transaction_no = trans_no_from
transaction_no = unmrg_trans_no_from
transaction_no = unmrg_trans_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = transaction_no
transaction_no = prev_transaction_no
ui_transaction
transaction_no <pk> numeric(10) not null
u_id varchar(15) not null
system varchar(8) null
dept varchar(8) null
terminal_id varchar(25) null
session_id varchar(15) null
update_facility_id varchar(20) not null
update_function varchar(8) not null
update_date timestamp not null
update_userid varchar(20) not null
ui_assumed_match
assumed_match_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
weight float not null
prev_transaction_no <fk> numeric(10) not null
transaction_no <fk> numeric(10) not null
transaction_no = transaction_no
u_id = u_id
ui_person
u_id <pk> varchar(15) not null
person_cat_code varchar(8) null
last_name varchar(40) not null
first_name varchar(40) not null
middle_name varchar(30) null
suffix varchar(10) null
title varchar(8) null
dob timestamp null
death varchar(1) null
sex varchar(8) null
mstatus varchar(8) null
ssn varchar(16) null
race varchar(8) null
ethnic varchar(8) null
religion varchar(8) null
language varchar(8) null
spouse_name varchar(100) null
mother_name varchar(100) null
mother_mn varchar(40) null
father_name varchar(100) null
maiden varchar(40) null
pob_city varchar(30) null
pob_state varchar(10) null
pob_country varchar(20) null
vip_flag varchar(8) null
vet_status varchar(8) null
status varchar(8) null
fname_phonetic_code varchar(8) null
lname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
mother_mn_phonetic_code varchar(8) null
maiden_phonetic_code varchar(8) null
spouse_phonetic_code varchar(8) null
mother_phonetic_code varchar(8) null
father_phonetic_code varchar(8) null
drivers_license varchar(20) null
drivers_license_st varchar(10) null
dod timestamp null
death_certificate varchar(10) null
nationality varchar(8) null
citizenship varchar(8) null
pension_no varchar(15) null
pension_exp_date timestamp null
repatriation_no varchar(16) null
district_of_residence varchar(8) null
lga_code varchar(4) null
military_branch varchar(4) null
military_rank varchar(4) null
military_status varchar(4) null
dummy_date timestamp null
class1 varchar(20) null
class2 varchar(20) null
class3 varchar(20) null
class4 varchar(20) null
class5 varchar(20) null
string1 varchar(40) null
string2 varchar(40) null
string3 varchar(40) null
string4 varchar(40) null
string5 varchar(40) null
string6 varchar(40) null
string7 varchar(100) null
string8 varchar(100) null
string9 varchar(100) null
string10 varchar(255) null
date1 timestamp null
date2 timestamp null
date3 timestamp null
date4 timestamp null
date5 timestamp null
transaction_no <fk> numeric(10) null
ui_aux_id
aux_id_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
aux_id_def <fk> numeric(10) not null
id varchar(40) not null
transaction_no <fk> numeric(10) null
u_id = u_id
ui_aux_id_def
aux_id_def <pk> numeric(10) not null
id_type varchar(8) not null
id_name varchar(30) not null
mask char(1) not null
id_length numeric(10) not null
format varchar(40) not null
variable_length char(1) not null
aux_id_def = aux_id_def
ui_aux_id_history
aux_id_id numeric(10) not null
u_id <fk> varchar(15) not null
aux_id_def <fk> numeric(10) null
id varchar(40) null
transaction_no <fk> numeric(10) not null
aux_id_def = aux_id_def
u_id = u_id
ui_address
address_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
address_type varchar(8) not null
address1 varchar(40) null
address2 varchar(40) null
address3 varchar(40) null
address4 varchar(40) null
city varchar(30) null
state_code varchar(10) null
postal_code varchar(8) null
postal_code_ext varchar(4) null
county varchar(20) null
country_code varchar(20) null
house_number varchar(10) null
street_dir varchar(5) null
street_name varchar(40) null
street_name_phonetic varchar(8) null
street_type varchar(5) null
transaction_no <fk> numeric(10) null
u_id = u_id
ui_address_history
address_id numeric(10) not null
u_id <fk> varchar(15) not null
address_type varchar(8) null
address1 varchar(40) null
address2 varchar(40) null
address3 varchar(40) null
address4 varchar(40) null
city varchar(30) null
state_code varchar(10) null
postal_code varchar(8) null
postal_code_ext varchar(4) null
county varchar(20) null
country_code varchar(20) null
house_number varchar(10) null
street_dir varchar(5) null
street_name varchar(40) null
street_name_phonetic varchar(8) null
street_type varchar(5) null
transaction_no <fk> numeric(10) not null
u_id = u_id
ui_mrg_trans
ui_mrg_trans_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
u_id_from <fk> varchar(15) not null
trans_no <fk> numeric(10) not null
trans_no_from <fk> numeric(10) null
unmrg_trans_no <fk> numeric(10) null
unmrg_trans_no_from <fk> numeric(10) null
u_id = u_id
u_id = u_id_from


FROM UI_PERSON_HISTORY
transaction_no = transaction_no
ui_transaction
transaction_no <pk> numeric(10) not null
u_id varchar(15) not null
system varchar(8) null
dept varchar(8) null
terminal_id varchar(25) null
session_id varchar(15) null
update_facility_id varchar(20) not null
update_function varchar(8) not null
update_date timestamp not null
update_userid varchar(20) not null
u_id = u_id
ui_person
u_id <pk> varchar(15) not null
person_cat_code varchar(8) null
last_name varchar(40) not null
first_name varchar(40) not null
middle_name varchar(30) null
suffix varchar(10) null
title varchar(8) null
dob timestamp null
death varchar(1) null
sex varchar(8) null
mstatus varchar(8) null
ssn varchar(16) null
race varchar(8) null
ethnic varchar(8) null
religion varchar(8) null
language varchar(8) null
spouse_name varchar(100) null
mother_name varchar(100) null
mother_mn varchar(40) null
father_name varchar(100) null
maiden varchar(40) null
pob_city varchar(30) null
pob_state varchar(10) null
pob_country varchar(20) null
vip_flag varchar(8) null
vet_status varchar(8) null
status varchar(8) null
fname_phonetic_code varchar(8) null
lname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
mother_mn_phonetic_code varchar(8) null
maiden_phonetic_code varchar(8) null
spouse_phonetic_code varchar(8) null
mother_phonetic_code varchar(8) null
father_phonetic_code varchar(8) null
drivers_license varchar(20) null
drivers_license_st varchar(10) null
dod timestamp null
death_certificate varchar(10) null
nationality varchar(8) null
citizenship varchar(8) null
pension_no varchar(15) null
pension_exp_date timestamp null
repatriation_no varchar(16) null
district_of_residence varchar(8) null
lga_code varchar(4) null
military_branch varchar(4) null
military_rank varchar(4) null
military_status varchar(4) null
dummy_date timestamp null
class1 varchar(20) null
class2 varchar(20) null
class3 varchar(20) null
class4 varchar(20) null
class5 varchar(20) null
string1 varchar(40) null
string2 varchar(40) null
string3 varchar(40) null
string4 varchar(40) null
string5 varchar(40) null
string6 varchar(40) null
string7 varchar(100) null
string8 varchar(100) null
string9 varchar(100) null
string10 varchar(255) null
date1 timestamp null
date2 timestamp null
date3 timestamp null
date4 timestamp null
date5 timestamp null
transaction_no <fk> numeric(10) null
ui_phone
phone_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
phone_type varchar(8) not null
phone varchar(20) not null
phone_ext varchar(6) null
transaction_no <fk> numeric(10) null
ui_phone_history
phone_id numeric(10) not null
u_id <fk> varchar(15) not null
phone_type varchar(8) null
phone varchar(20) null
phone_ext varchar(6) null
transaction_no <fk> numeric(10) not null
ui_duplic
ui_duplic_id <pk> numeric(10) not null
duplic_id <fk> varchar(15) not null
existing_id <fk> varchar(15) not null
weight float not null
description varchar(30) null
dummy_date timestamp null
resolved varchar(8) null
transaction_no <fk> numeric(10) null
u_id = existing_id
u_id = duplic_id
ui_local_id
ul_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
facility <fk> varchar(20) not null
local_id varchar(25) not null
status varchar(8) null
transaction_no <fk> numeric(10) null
u_id = u_id
facility_code = facility
ui_facility
facility_code <pk> varchar(20) not null
description varchar(30) null
status varchar(8) null
addr1 varchar(30) null
addr2 varchar(30) null
city varchar(30) null
state varchar(10) null
zip varchar(8) null
zipext varchar(4) null
county varchar(20) null
country varchar(20) null
region_code varchar(8) null
date_time timestamp null
id_length numeric null
mask char(1) null
local_id_format varchar(35) null
facility_mask char(1) null
next_local_id varchar(25) null
allow_less_id_length varchar(1) null
facility_code = facility
ui_local_id_history
ul_id numeric(10) not null
u_id <fk> varchar(15) not null
facility <fk> varchar(20) null
local_id varchar(25) null
status varchar(8) null
transaction_no <fk> numeric(10) not null
u_id = u_id
ui_alias
ui_alias_id <pk> numeric(10) not null
u_id <fk> varchar(15) not null
last_name varchar(40) not null
first_name varchar(40) not null
middle_name varchar(30) null
lname_phonetic_code varchar(8) null
fname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
transaction_no <fk> numeric(10) null
u_id = u_id
ui_alias_history
ui_alias_id numeric(10) not null
u_id <fk> varchar(15) not null
last_name varchar(40) null
first_name varchar(40) null
middle_name varchar(30) null
lname_phonetic_code varchar(8) null
fname_phonetic_code varchar(8) null
mname_phonetic_code varchar(8) null
transaction_no <fk> numeric(10) not null
u_id = u_id


ui_ctrl_rule
ui_ctrl_rule_id <pk> numeric(10) not null
rule_name varchar(16) not null
root_file varchar(16) not null
read_only char(1) not null
in_use char(1) not null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_ctrl_rule_id = ui_ctrl_rule_id
ui_ctrl_file
ui_ctrl_file_id <pk> numeric(10) not null
ui_ctrl_rule_id <fk> numeric(10) not null
file_type varchar(3) not null
file_name varchar(18) not null
file_ext varchar(3) not null
file_content long varchar null
content_date timestamp not null
last_synch_date timestamp null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_ctrl_file_id = ui_ctrl_file_id
ui_ctrl_table
ui_ctrl_table_id <pk> numeric(10) not null
table_name varchar(30) not null
description varchar(48) null
read_only char(1) not null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_ctrl_table_id = ui_ctrl_table_id
ui_ctrl_column
ui_ctrl_column_id <pk> numeric(10) not null
ui_ctrl_table_id <fk> numeric(10) not null
column_name varchar(30) not null
description varchar(48) null
read_only char(1) not null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_misc_opt_control
control_type <pk> varchar(8) not null
description varchar(40) null
control_type = control_type
ui_misc_option
misc_option_id <pk> numeric(10) not null
country_code <fk> varchar(8) not null
control_type <fk> varchar(8) not null
option_name varchar(40) not null
value varchar(40) not null
country_code = country_code
ui_misc_opt_country
country_code <pk> varchar(8) not null
country_name varchar(40) null
ui_alias_x_name
ui_alias_id <pk> numeric(10) not null
u_id varchar(15) not null
x_last_name varchar(40) not null
x_first_name varchar(40) not null
x_middle_name varchar(30) null
ui_nickname
ui_nickname_id <pk> numeric(10) not null
formal_name varchar(40) not null
nick_name varchar(40) not null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_ctrl_file_hist
ui_ctrl_file_hist_id numeric(10) not null
ui_ctrl_file_id <fk> numeric(10) not null
file_type varchar(3) not null
file_name varchar(18) not null
file_ext varchar(3) not null
file_content long varchar null
content_date timestamp not null
save_date timestamp not null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_ctrl_field
ui_ctrl_field_id <pk> numeric(10) not null
field_name varchar(2) not null
field_type varchar(2) not null
field_length numeric(10) not null
field_missing varchar(2) not null
description varchar(48) null
ui_ctrl_column_id numeric(10) null
create_date timestamp not null
create_userid varchar(20) not null
update_date timestamp null
update_userid varchar(20) null
ui_table
ui_table_id <pk> numeric(10) not null
ui_table_name varchar(30) not null
ui_table_id = ui_table_id
ui_table_column
table_column_id <pk> numeric(10) not null
ui_table_id <fk> numeric(10) not null
column_name varchar(30) not null
default_label varchar(40) not null
label varchar(40) not null
visible char(1) not null
required char(1) not null
read_only char(1) not null
control_sec
ctrl_key <pk> varchar(10) not null
description varchar(50) null
ctrl_value varchar(10) null
create_date timestamp null
ui_no_passwd
no_passwd varchar(20) not null
idx_ui_no_passwd
ui_person_x_name
u_id <pk> varchar(15) not null
x_last_name varchar(40) not null
x_first_name varchar(40) not null
x_middle_name varchar(30) null
x_mother_mn varchar(40) null
x_maiden varchar(40) null


ui_login_current
ui_login_current_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
login_time timestamp not null
logout_time timestamp null
update_time timestamp null
session_id varchar(15) null
usersl_id = usersl_id
stc_user
usersl_id <pk> numeric(10) not null
user_id varchar(20) not null
last_name varchar(30) not null
first_name varchar(30) not null
middle_initial varchar(1) null
user_desc varchar(100) null
email_address varchar(100) null
admin varchar(1) not null
eff_date timestamp not null
exp_date timestamp null
change_pass varchar(1) not null
login_attempts numeric(10) not null
create_date timestamp not null
create_userid varchar(20) not null
ui_notify_user
ui_notify_user_id <pk> numeric(10) not null
event_code varchar(5) not null
usersl_id <fk> numeric(10) not null
create_date timestamp not null
create_userid varchar(20) not null
usersl_id = usersl_id
usersl_id = usersl_id
stc_user_region
user_region_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
region_code varchar(8) not null
create_date timestamp not null
create_userid varchar(20) not null
usersl_id = usersl_id
stc_user_group
usergp_id <pk> numeric(10) not null
group_id <fk> numeric(10) not null
usersl_id <fk> numeric(10) not null
eff_date timestamp not null
exp_date timestamp null
create_date timestamp not null
create_userid varchar(20) not null
ui_login
ui_login_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
login_time timestamp not null
logout_time timestamp null
update_time timestamp null
session_id varchar(15) null
usersl_id = usersl_id
group_id = group_id
stc_group
group_id <pk> numeric(10) not null
group_name varchar(30) not null
group_desc varchar(100) null
eff_date timestamp not null
exp_date timestamp null
create_date timestamp not null
create_userid varchar(20) not null
ui_user_passwd_hist
ui_passwd_hist_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
passwd varchar(20) not null
use_date timestamp not null
ui_user_passwd_hist_1
group_id = group_id
stc_group_acc
group_acc_id <pk> numeric(10) not null
group_id <fk> numeric(10) not null
acc_def_id <fk> numeric(10) not null
eff_date timestamp not null
exp_date timestamp null
create_date timestamp not null
create_userid varchar(20) not null
stc_user_acc
user_acc_id <pk> numeric(10) not null
usersl_id <fk> numeric(10) not null
acc_def_id <fk> numeric(10) not null
eff_date timestamp not null
exp_date timestamp null
create_date timestamp not null
create_userid varchar(20) not null
usersl_id = usersl_
acc_def_id = acc_def_id
acc_def_id = acc_def_id
stc_acc_def
acc_def_id <pk> numeric(10) not null
module_id <fk> numeric(10) not null
code varchar(5) not null
descr varchar(50) null
create_date timestamp null
create_userid varchar(15) null
module_id = module_id
stc_common_detail
common_detail_id <pk> numeric(10) not null
common_header_id <fk> numeric(10) not null
code varchar(20) not null
descr varchar(50) not null
read_only char(1) not null
create_date timestamp not null
create_userid varchar(20) not null
stc_module
module_id <pk> numeric(10) not null
appl_id <fk> numeric(10) not null
code varchar(5) not null
descr varchar(30) not null
read_only char(1) not null
create_date timestamp not null
create_userid varchar(20) not null
common_header_id = common_header_id
appl_id = appl_id
stc_common_header
common_header_id <pk> numeric(10) not null
appl_id <fk> numeric(10) not null
code varchar(8) not null
descr varchar(50) not null
read_only char(1) not null
max_input_len numeric(10) not null
typ_table_code varchar(3) not null
create_date timestamp not null
create_userid varchar(20) not null
appl_id = appl_id
stc_appl
appl_id <pk> numeric(10) not null
code varchar(8) not null
descr varchar(30) not null
read_only char(1) not null
create_date timestamp not null
create_userid varchar(20) not null

Chapter 6

Upgrading the GUIs and Publications

About this Chapter

Overview

This chapter presents background information and step-by-step instructions for upgrading the e*Index GUI to version 5.0.5 for SRE. It also includes information about using the electronic documentation for e*Index.

The following diagram illustrates the contents of each major topic in this chapter. For the page numbers on which specific topics appear, see the next page of this chapter.

About the GUI and Publications — Learn about the GUI files and the e*Index Electronic Library

Upgrade the GUI and Publications — Learn how to upgrade the GUI and publications, and to make any required modifications to the GUI files

What's Inside

This chapter provides background information and instructions related to the topics listed below.

Learning About Upgrade Tasks

Overview

This section of the chapter provides background information about upgrading the e*Index GUIs and the online publications.

What are the System Requirements?

In order to successfully install and use the e*Index GUI, the hardware and software items listed below are required. If you are currently running the e*Index GUI, your computer should already meet these requirements.

- Client Hardware
  - Windows XP, 2000, or 2003 with required patches
  - Pentium 90 or higher
  - 32MB Memory
  - 30MB Disk Space
  - VGA or higher
  - NIC Cards
  - Valid TCP/IP Addresses
  - Network Connections
  - CD-ROM Drive
- Client Software
  - Database Client Software
    - Oracle 8.1.7 or 9*i* Client for an Oracle database
    - Sybase 12.0 Client for a Sybase database
    - Microsoft SQL Server 7.0 for a SQL Server database
  - Internet Explorer 4.01 or higher (for online help)
  - e*Index 5.0.5 for SRE

What is the Quality Workstation?

The e*Index Quality Workstation is the client machine on which you install the e*Index GUIs. From this workstation, you can add, delete, modify, and monitor the data in your e*Index database. On the Quality Workstation, you can merge and unmerge member records, search for potential duplicates in the database, compare records, update local ID and alias information, create comments, and so on. You can also print reports and print the current active window for future reference.

What Environment Variables are Created?

When you upgrade the e*Index GUI, a new environment variable, INTBIN, is created. This variable tells the GUI where the **\codeset** directory is located. This folder contains the codeset mapping files used by the Vality matching algorithm. By default, INTBIN is to the current working directory ("."). If you modify your file structure, you may need to modify these two variables.

An existing environment variable, VTICFG, will be reset to the current working directory. If you store your Vality rule set files in a directory other then the e*Index client directory, you need to update the VTICFG variable after the upgrade.

How Do I View the Publications?

Once you copy the e*Index electronic library to your workstation, you can view the documents online using Adobe® Acrobat® Reader. You can also print the files to any postscript printer. If you do not have Acrobat Reader installed on your Workstation, you can install it from the Internet at **http://www.adobe.com**. For more information on the e*Index electronic library, read the **Readme.txt** file included with your publications.

The electronic library includes links between documents, a navigational document named **Welcome.pdf**, an index that allows you to search among all publications, and a feedback form for you to provide us with your comments. The library requires an additional 40MB of disk space.

Performing the Upgrade

Overview

To upgrade the e*Index GUIs, complete the following steps:

- Step 1: Back up **stc_ua.ini**
- Step 2: Uninstall the Current e*Index GUI
- Step 3: Install the new GUI
- Step 4: Copy the Publications
- Step 5: For Oracle Only, Verify **tnsnames.ora**
- Step 6: For Sybase Only, Verify **sql.ini**
- Step 7: For SQL Server Only, Verify the ODBC Data Source
- Step 8: Restore **stc_ua.ini**
- Step 9: Register Online Help Support Files

Note: *These steps assume that you already have the correct version of the database client software installed on the client workstation.*

Step 1: Back up stc_ua.ini

Before installing the upgrade files, back up the e*Index GUI files, and make a backup copy of the initialization file, **stc_ua.ini**. This file is located in the e*Index GUI home directory, and should be copied to a temporary directory. After you install the new GUI, you can copy the database stanzas that you have currently defined into the new **stc_ua.ini** file.

Step 2: Uninstall the Current e*Index GUI

Once you back up **stc_ua.ini**, remove the existing e*Index GUI. To uninstall the e*Index GUI, you can simply delete all files in the e*Index GUI home directory, or, if no other e*Index components are installed on the client machine, you can use the Add/Remove Programs function of the Control Panel to remove e*Index.

Step 3: Install the GUI

Installing the GUI is very similar to the process you followed to install the e*Gate Schema files and the database upgrade files. If you installed all of the upgrade files at the time you installed the e*Index Schema files, you can skip to "Step 4: Copy the Publications".

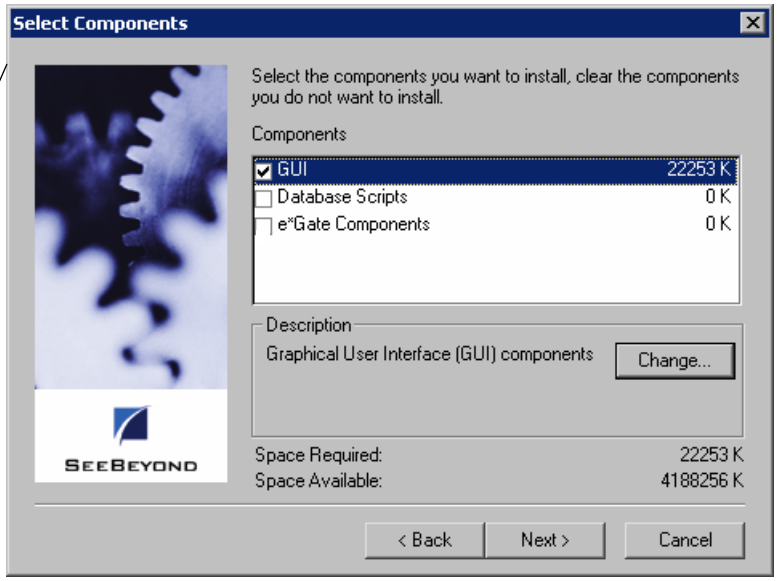
To install the GUI files

Before you begin:

- ✓ Make sure that your workstation meets the requirements listed on page 3-3 of this chapter
- ✓ Complete "Step 1: Back up stc_ua.ini" and "Step 2: Uninstall the Current e*Index GUI"

1 Follow steps 1 through 7 under "Step 5: Install the e*Index Schema Files" in Chapter 2, "Upgrading the e*Index Schema Files." The Select Components window should now be visible.

Use the Select Components window to specify the components you want to install

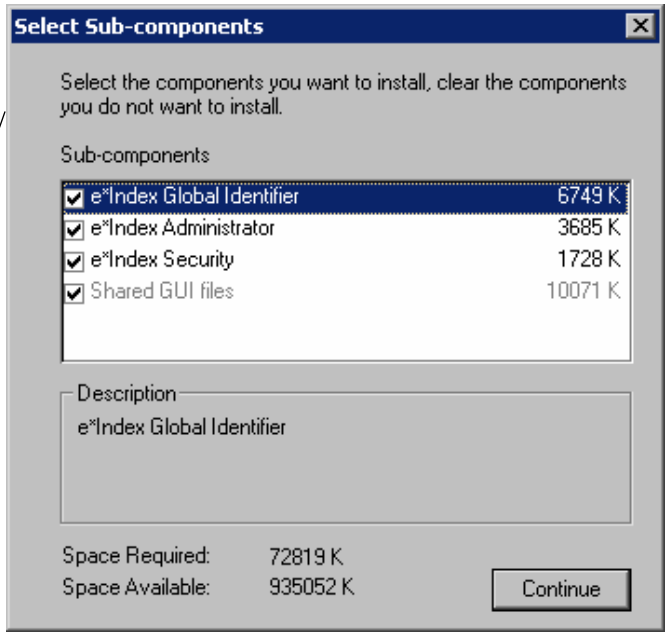


2 On the Select Components window, select **GUI**.

Change button

3 To verify which GUI components are being installed, highlight **GUI**, and then click **Change**. The Select Sub-components window appears.

You can select which GUI files to install from the Select Sub-components window



Continue button

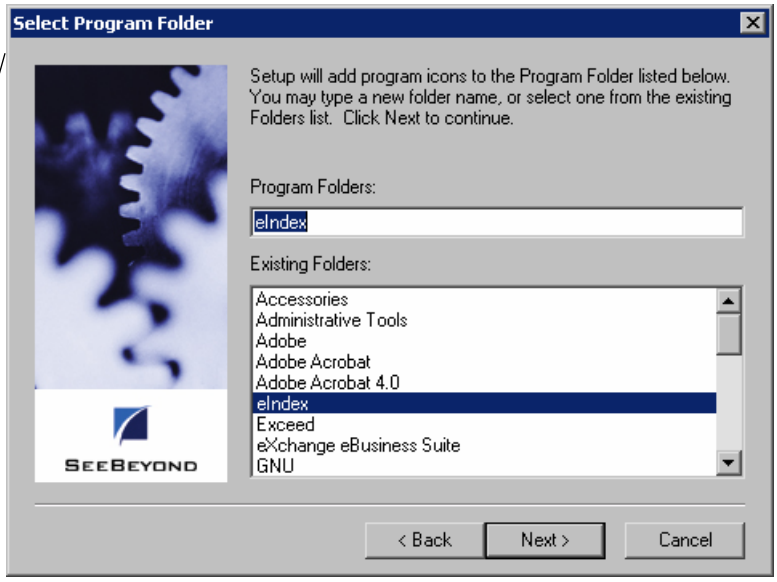
4 Select only the applications you want to install, and then click **Continue**.

Note: *If you are installing e*Index on a non-administrator workstation, you should only select* ***e*Index Global Identifier****.*

Next button

5 On the Select Components window, click **Next**. The Select Program Folder window appears.

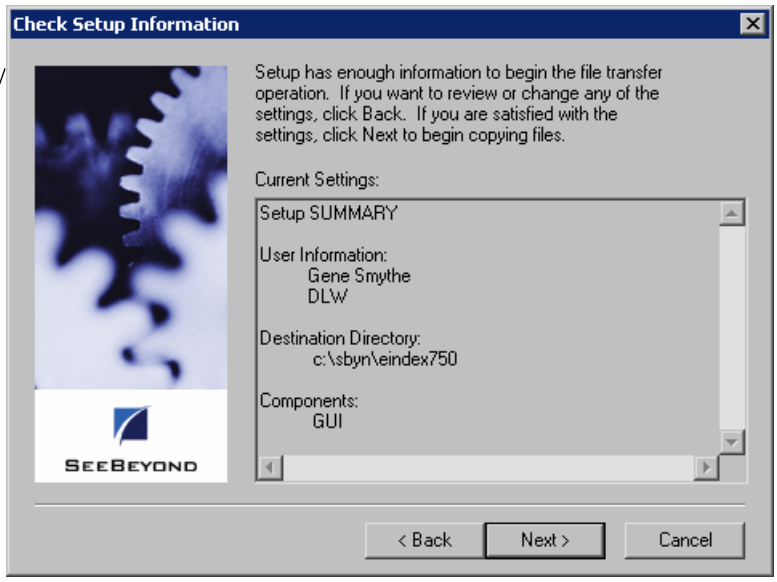
Select a folder for the program icons on the Select Program Folder window



Next button

6 Specify the folder into which you want to install the program icons or accept the default name, and then click **Next**. The Check Setup Information window appears.

Verify the installation information on the Check Setup Information window



7 Verify the information you specified, and do one of the following:

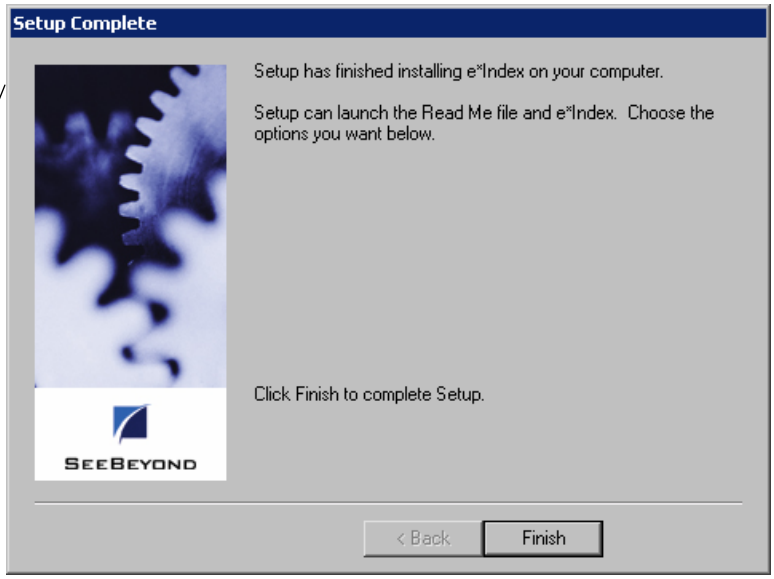
To change any of the options you selected, click **Back**, and make the necessary changes.

Back button

To continue with the installation, click **Next**. When the GUI files are installed, the Setup Complete window appears.

Next button

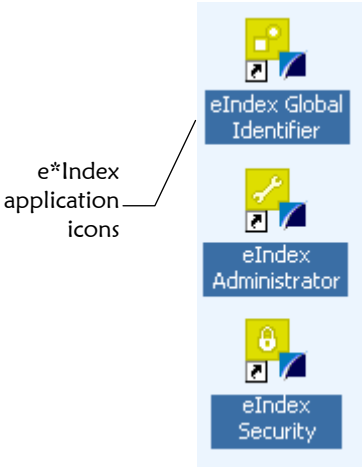
When the Setup Complete window appears, your installation is finished



Finish button

8 Click **Finish** to conclude the installation process and return to the Windows desktop.

9 You can view the e*Index application icons on the Windows desktop.



10 Continue to "Step 5: Copy the Publications."

Step 4: Copy the Publications

The electronic documentation files for e*Index are located on the installation CD-ROM in the folder **\docs**. You can delete the files from your current e*Index publications directory, and copy the new files into the directory. The following is a list of documents included in the electronic library:

- **eI_implementation.pdf**
  The *e*Index Global Identifier Implementation Guide* provides information on how to plan, design, and develop an e*Index implementation.
- **eI_installation.pdf**
  The *e*Index Global Identifier Installation Guide* explains how to install all components of e*Index, including the GUI, database, and e*Gate Schema.
- **eI_java_ref.pdf**
  The *Java Programmer's Reference for e*Index Active Integration* explains how to work with the active integration API, and provides a reference of the available classes and methods.
- **eI_reports.pdf**
  *Working with Reports for e*Index Global Identifier* explains how to modify and run standard reports against the e*Index database.
- **eI_techref.pdf**
  The *e*Index Global Identifier Technical Reference* is designed to assist e*Gate programmers in writing Monk scripts for the e*Ways for e*Index.
- **eI_upgrade.pdf**
  The *e*Index Global Identifier Upgrade Guide* is explains how to upgrade all components of e*Index to version 5.0.5 for SRE.

- **eI_userguide.pdf**
  The *e*Index Global Identifier User's Guide* explains how to use the e*Index GUI.
- **eIA_userguide.pdf**
  The *e*Index Administrator User's Guide* explains how to use the e*Index Administrator GUI.
- **eIS_userguide.pdf**
  The *e*Index Security User's Guide* explains how to set up and maintain security for the e*Index applications.
- **feedback_form.pdf**
  Use this form to provide any comments or suggestions for improving the documentation provided for e*Index.
- **init_load.pdf**
  The *e*Index Initial Load User's Guide* describes how to install, modify, and run the Schemas you use to load legacy data into the e*Index database.
- **Readme.txt**
  This document provides information about the electronic library, such as using the cross-referencing index, search capabilities, and so on.
- **Welcome.pdf**
  The Welcome Document lists all publications available in PDF format, and provides links to each document file. Each file also links back to the Welcome Document.

Step 5: For Oracle Only, Verify tnsnames.ora

If your e*Index database runs on an Oracle platform, verify that the **tnsnames.ora** file contains a stanza pointing to the database you use. If you do not know how to modify the Oracle **tnsnames.ora** file, refer to the appropriate Oracle documentation. Your file may differ from the sample below, depending on how your Oracle networking is configured.

You should have a stanza for the e*Index database similar to the following example.

```
ei01.world =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (PROTOCOL = TCP)
          (Host = 100.0.0.00)
          (Port = 1000)
        )
    )
    (CONNECT_DATA = (SID = EI01)
    )
  )
```

Step 6: For Sybase Only, Verify sql.ini

If your e*Index database runs on a Sybase platform, verify that the **sql.ini** file on the computer you are using contains a stanza pointing to the e*Index database. You should have a stanza for the e*Index database similar to the following example. This sample provides an example of how the stanza may appear for an Adaptive Server named "challenger" using a TCP/IP connection. For more information about configuring this file, refer to your Sybase user documentation.

```
[challenger]
master=TCP,challenger,4100
query=TCP,challenger,4100
```

Step 7: For SQL Server Only, Verify the ODBC Data Source

If your e*Index database runs on a SQL Server platform, verify that an ODBC data source for the e*Index database has been defined on the computer on which you installed the new GUIs. The data source specifies the driver used to connect to the database and additional information about the database. ODBC data source definitions are accessed through the **Data Sources (ODBC)** option in the Control Panel.

Step 8: Restore stc_ua.ini

Once you install the new GUI files, you can copy your customized database stanzas from your original initialization file, **stc_ua.ini**, into the new file (located in the e*Index home directory). After you copy the customized

stanzas, you can delete the old initialization file. You may also want to modify the **FACILITYID** field in the new **stc_ua.ini**.

Step 9: Register the Online Help Support File

If you haven't done so with previous versions, you should register the supporting help file before you use the online help system provided with the e*Index GUIs.

▶ To register d2hPopup.ocx

Before you begin:

- ✓ Open MS-DOS from the Start menu

1. At the command prompt, navigate to your e*Index home directory, and then to the **client** subdirectory.
2. At the prompt, type **regsvr32 d2hPopup.ocx**.

Note: *If you do not register this file, you will be prompted to download the file from a website the first time you open an e*Index online help file. If you choose to download from the website, this file will be registered for you automatically. Once this file is registered, you will not receive the prompt when you open the help files.*
