Product Documentation

ER/Studio® Data Architect

Quick Start Guide

3rd Edition

Version 9.1
Published October 12, 2011

Embarcadero Technologies, Inc. is a leading provider of award-winning tools for application developers and database professionals so they can design systems right, build them faster and run them better, regardless of their platform or programming language. Ninety of the Fortune 100 and an active community of more than three million users worldwide rely on Embarcadero products to increase productivity, reduce costs, simplify change management and compliance and accelerate innovation. The company's flagship tools include: Embarcadero® Change Manager™, CodeGear™ RAD Studio, DBArtisan®, Delphi®, ER/Studio®, JBuilder® and Rapid SQL®. Founded in 1993, Embarcadero is headquartered in San Francisco, with offices located around the world. Embarcadero is online at www.embarcadero.com.

October 12, 2011

Contents

INTRODUCING ER/STUDIO DATA ARCHITECT

ER/Studio Data Architect (ER/Studio DA) is a visual modeling application used for platform-independent logical data architecture analysis and design, in addition to platform-specific physical database design and construction. Its powerful, multi-level design environment addresses the everyday needs of database administrators, developers, and data architects who build and maintain large, complex database applications and strive to consolidate, report, and re-use metadata across the enterprise.

ER/Studio DA's progressive interface and simplicity has been designed to effectively address the ease-of-use issues which have plagued data modeling and CASE tools for the past decade and more. The application equips you to create, understand, and manage the life-cycle of mission-critical database designs and business metadata within the enterprise.

ER/Studio DA is rich and customizable. ER/Studio DA offers:

Strong logical design capabilities

- The ability to spawn many physical designs from a corporate logical design
- Bi-directional model comparison and information synchronization
- Visual-Basic for Applications API for product customization
- Powerful DDL reverse engineering and generation
- Metadata import and export capabilities
- Data lineage documentation
- Sophisticated XML, HTML, and RTF-based documentation and reporting facilities.

ABOUT THIS DOCUMENT

This document helps you install and use ER/Studio DA and introduces you to the basic functionality of ER/Studio DA.

For more information, see the Embarcadero website at www.embarcadero.com.

NOTICE FOR DEVELOPER EDITION USERS

The Developer Edition of ER/Studio Data Architect provides the user with all the standard functionality of ER/Studio except for the following limitations:

- No access to the ER/Studio Repository and ER/Studio Enterprise Portal.
- Limited import and export functionality; the Import and Export Metadata Model Bridges are not supported.
- Limited database support.

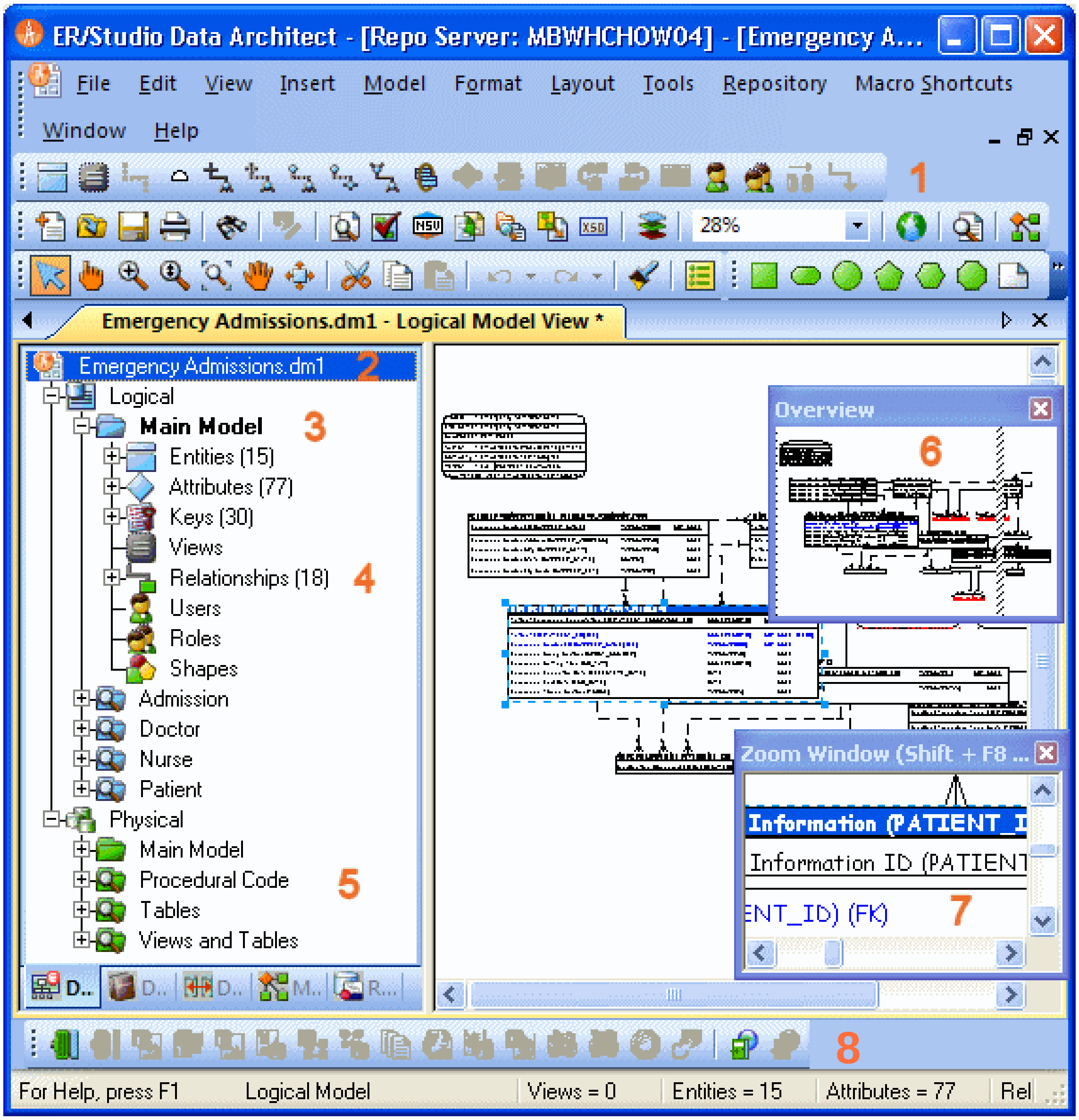
APPLICATION INTERFACE

The application interface is divided into several tabbed windows that let you navigate and customize your workspace. The application includes context-sensitive toolbars, menus, intuitive icons, keyboard commands, and other resources to help you work efficiently. ER/Studio DA's user interface has a standard Windows look and feel.

For a detailed description of the internal data structure of the ER/Studio DA application and the Repository, you can look at Metadata models (DM1 files), usually found in the following directories:

- **Windows XP**: `C:\Documents and Settings\All Users\Application Data\Embarcadero\ERStudioDA_X.X\Sample Models`
- **Windows Vista**: `C:\ProgramData\Embarcadero\ERStudioDA_X.X\Sample Models`

For information on enhanced UI features, click a number in the image below:



1 ER/Studio DA has an enhanced user interface that is similar to Windows XP with intuitive icons.

2 The name of the active model, with or without the full path to the file, can be displayed on the title bar of the main application window. This is an option you can set in **Tools > Options > Application > Application Defaults**.

3 ER/Studio DA displays the active model in bold type, so that you can easily see which model or submodel you are working on.

4 The **Data Model** tab of the Data Model Explorer displays indexes, relationships, and attributes as separate nodes.

5 The **Data Model** tab of the Data Model Explorer displays schema objects like packages, procedures, functions, materialized views, auxiliary tables, and triggers as separate nodes.

6 The **Overview Window** lets you navigate large Data Models.

7 The **Zoom Window** helps you focus on the details of a specific area of a large, reduced diagram.

8 You can dock toolbars anywhere in the ER/Studio DA user interface.

The Data Model Explorer is comprised of five tabs that offer easy access to important functionality and also includes an Explorer tree that lets you efficiently navigate the objects of your data models. The ER/Studio DA interface to the Data Model is divided into two main areas. On the left is the Data Model Explorer and on the right is the Data Model Window. This convention is carried out on other tabs of the user interface, such as the Repository and Data Lineage tabs, where you'll find an explorer on the left and a graphical representation on the right. The Data Model Window also provides a complete workspace for creating physical and logical data models and smaller, movable windows to help you navigate large data models.

NEW FEATURES OF ER/STUDIO DA 9.1

Database Support Improvements

Netezza: **Synonym** database objects are now supported.

SQL Server 2005 and later: Support for disabling referential integrity has been added to relationships. The **NOCHECK** option disables checking the existing data against the FK constraint, when adding it. It also disables the FK constraint so that future data inserts and updates are not validated against it.

SQL Server 2008 and later: The **geometry** and **geography spatial** data types are now supported.

NCR TeraData: The **WITH NOCHECK** option for FK constraints is now supported. This disables constraint data checking when performing SQL operations on the data in the table the FK is created on.

Oracle 9i and later: The **NoValidate** option is available for relationships and indexes. Enable this option if you do not want the constraint to be applied to the data in the table.

Oracle 10g and later: The **SYS.XML Type** and **SDO_GEOMETRY** data types are now supported.

Oracle 11g and later: **Virtual columns** are now supported.

DB2 LUW: **Multidimensional Clustering** (MDC) tables are now supported.

NEW FEATURES OF ER/STUDIO DA 9.0.1

User Interface Improvement

Double-click behavior on the Where Used tab of the Table Editor and Entity Editor can be controlled by the Tools > Options > Where used tab so that double-clicking an object will either bring up the object editor or the user-defined mapping editor.

DB2 z/OS Support Improvement

For DB2 z/OS physical models a new Generated by Default option is available for ROWID columns and columns marked as IDENTITY columns. That option is also available for TIMESTAMP data types to indicate a "row change timestamp" column.

NEW FEATURES OF ER/STUDIO DA 9.0 AND ER/STUDIO REPOSITORY 6.0.

Additional Platform Support

ER/Studio DA now fully supports Netezza 4.6 and 5.0.

User Interface Improvements

Since the denormalization feature has changed in 9.0, any documents containing denormalization mappings will be upgraded to use the new feature of ER/Studio DA 9.0. Denormalization mappings are now retained with the table data and can be viewed and edited on the Where Used tab of the Table Editor. Denormalization history, including Columns, Indexes, and Relationships of pre-map tables, is no longer retained, so the undo and make permanent functions specifically for denormalizations are no longer available. Denormalization mappings have been worked into the normal undo function of the product allowing them to be undone like other operations. However, the names of the pre-map tables are retained. Retaining only essential pre-map information means that the DM1 file with denormalizations is smaller and therefore opens faster.

For more information, see “New Features of ER/Studio DA 9.0 and ER/Studio Repository 6.0” and “Optimizing Query Performance on the Physical Model (Denormalizing the Physical Model)” in the *ER/Studio Data Architect User Guide* and online help.

ER/Studio Repository Performance Improvements

ER/Studio Repository software is now multi-threaded, enabling some processes to occur concurrently. The following describes some important rules that affect multi-threading on the Repository:

- If a Check In transaction is being processed on a given Diagram or Data Dictionary, all other operations on that Diagram or Data Dictionary must wait until the Check In completes.

- If an Add or Check In transaction is being processed, all other Add or Check In transactions must wait until the initial transaction completes. This same rule applies to Set Named Release, Branch and Branch/Merge Check In operations.
- Most other common tasks, including Log Ins, Log Outs, Gets, Check Outs and Security Center operations are processed concurrently by the server, but may be affected by database level-locking mechanisms.

INSTALLATION

This section will help you ensure your environment meets the technical requirements and helps you install ER/Studio Data Architect or ER/Studio Data Architect XE. It is comprised of the following topics:

ER/STUDIO DATA ARCHITECT INSTALLATION CHECKLIST

Use the following checklist to ensure you correctly install and configure ER/Studio Data Architect.

Done	N/A	Requirements
		Download ER/Studio Data Architect You can download a 14-day trial version of ER/Studio Data Architect, from www.embarcadero.com/downloads. Embarcadero will send you a license registration code by email.
		Review ER/Studio Data Architect ReadMe For the most current installation and usage information, see the ReadMe at docs.embarcadero.com.
		Ensure System Compatibility For system requirements, see System Requirements on page 12.
		Install ER/Studio Data Architect For installation details, see Installing ER/Studio Data Architect on page 12.
		License ER/Studio Data Architect For ER/Studio Data Architect licensing details, see Licensing ER/Studio Data Architect on page 14.
		Run and Test ER/Studio Data Architect For step-by-step instructions on using ER/Studio Data Architect, see Tutorials on page 17.

SYSTEM REQUIREMENTS

Review the requirements before you install ER/Studio Data Architect. Adhering to these requirements optimizes ER/Studio Data Architect performance.

ER/Studio Data Architect Requirements		
Hardware	Processor	Pentium III, 600 MHz or higher
	RAM	1 GB
	Disk Space	340 MB
		175 MB in the System TEMP directory for temporary file storage during installation.
	DVD Reader	
Operating System	Any of the following: • Windows 7 (32-bit and 64-bit modes) • Windows Server 2008 • Windows Vista • Windows XP Professional • Windows 2003 (32-bit mode) • Windows 2000 Pro Server • Advanced Server with Service Pack 2	
Software	To use an Open Data Base Connectivity (ODBC) database as a data source, install the ODBC driver on your client computer. To connect to the DBMS using native connectivity, install the native client libraries available from the DBMS vendor.	
Privileges	Local administrator privileges (required for installation only).	

INSTALLING ER/STUDIO DATA ARCHITECT

NOTE: To ensure compatibility, the release dates of ER/Studio Repository and ER/Studio Data Architect must be the same.

1 Log on to Windows with local administrator privileges.

2 Start the ER/Studio Data Architect installation program.

NOTE: To do a silent install, you can run the installation program from the command prompt with the -s parameter.

3 Walk through the installation wizard following the onscreen prompts.

The following provides additional information for some options that are not self-explanatory:

- **Default Notation**: Choose the notation type you want ER/Studio Data Architect to use by default to display the model.
  - **IDEF1X**: Data modeling technique used by many branches of the United States Federal Government.
  - **IE (Martin/Finkelstein)**: Information Engineering (IE) as developed by Martin and later revised by Finkelstein.
  - **IE (Crow's Feet)**: Uses IE notation and represents relationships with connecting lines between entities, and symbols at the ends of those lines to represent the cardinality of the relationship. This is the notation used in Oracle texts and in other applications such as Visio. Foreign keys are displayed in entity boxes.
  - **IE (Hide Foreign Keys)**: Uses standard IE notation, but foreign keys are not displayed in entity boxes.

  NOTE: You can change the notation setting at a later time. To change the notation, choose Tools > Options from the ER/Studio Data Architect Main menu, and then click the Logical or Physical tab where you can choose the notation type.

- **Shared Application Data Location**: By default, ER/Studio Data Architect installs shared directories and application files such as the DatatypeMapping, Macros, and Sample Models directories in the following locations:
  - **Windows XP:** `C:\Documents and Settings\All Users\Application Data\Embarcadero\ERStudio_X.X`
  - **Windows Vista:** `C:\ProgramData\Embarcadero\ERStudio_X.X`

  If you prefer to store these files in another directory, follow the procedure in Changing the Location of ER/Studio Data Architect Shared Files on page 13.

- For step-by-step instructions on how to use ER/Studio Data Architect, see the *ER/Studio Data Architect Evaluation Guide* at docs.embarcadero.com.

Changing the Location of ER/Studio Data Architect Shared Files

NOTE: You must have local administrator privileges to change the location where ER/Studio Data Architect stores and searches for shared application data.

1 Choose **Help > About ER/Studio Data Architect**, and then click **File Path**.

2 In the **File Path** dialog, browse to an existing directory where you want to store your shared files.

NOTE: The directory path you choose must be the location where you have created an Embarcadero\ERStudioDA_X.X directory or where you want ER/Studio Data Architect to create this directory.

ER/Studio Data Architect creates `C:\Documents and Settings\All Users\Application Data\Embarcadero\ERStudioDA_X.X` directory, if it does not already exist.

3 In the **File Path** dialog, click **OK**, and then in the **About ERStudio** dialog, click **OK** again.

ER/Studio Data Architect searches for the datatype mappings in this directory.

4 Choose **Tools > Options > Directories**, specify where you want to store your macros, and then click **OK**.

5 Copy the contents of the `DatatypeMapping`, `Macros`, and `Sample Models` directories to the new location.

6 To effect the changes, restart ER/Studio Data Architect.

INSTALLING ER/STUDIO XE2

For information on installing ER/Studio with Embarcadero AppWave, please see the AppWave Quick Start and AppWave Administrator Guide.

LICENSING ER/STUDIO DATA ARCHITECT

ER/Studio Data Architect requires one or more licenses in order to run. A baseline license provides basic feature support for the product. In addition, incremental licenses may be required to support specific DBMS platforms, product add-ons or other functions.

ER/Studio Data Architect supports three types of licenses:

- **Trial licenses**: A license for a 14-day, full-featured trial version of the product. The trial license must be registered before you can use the product. For more information, see Registering a Trial or Workstation License at Application Startup on page 15.
- **Workstation licenses**: A license or set of licenses that is tied to a particular workstation. The product can only be used on that workstation. For more information, see Registering a Workstation License after Application Startup on page 15.
- **Networked licenses**: Networked licenses are administered and distributed by a central License Server (Embarcadero License Center or Acresso FLEXnet Publisher). There are two types of networked licenses:
  - **Concurrent licensing**: Users on different machines take turns using licenses from a shared pool.
  - **Networked Named User licensing**: Licenses are pre-assigned to specific users that are setup on the license server's user list. Those users are guaranteed to have licenses available any time.

  For more information, see Registering a Networked License on page 16.

  If you have problems registering, see Registering by Phone on page 16.

Registering a Trial or Workstation License at Application Startup

Shortly after downloading a trial version of an Embarcadero product, you should receive an e-mail from Embarcadero with a serial number, which must be registered as follows. These instructions also apply if you have purchased a product and received a serial number prior to running the application for the first time, or if the trial period has expired.

1 Start the application.

The Embarcadero License Registration dialog appears.

2 Copy the serial number from the e-mail, and then paste it into the **Serial Number** field.

3 Enter your Embarcadero Developer Network (EDN) account credentials in the **Login or Email** and **Password** fields.

If you have not previously created an EDN account, or have forgotten your password, click **I need to create ...** or **I've lost my password**.

4 Click **Register**.

The activation file should be downloaded and installed automatically. If this does not happen, click the **Trouble Connecting? Try Web Registration** link in the license registration dialog and follow the prompts. If you still have problems, see Registering by Phone on page 16.

Registering a Workstation License after Application Startup

The following instructions assume that you have received a workstation license by e-mail and that you currently have a valid trial license. If you did not install a trial version or if the trial period has expired, follow the instructions in Registering a Trial or Workstation License at Application Startup on page 15 instead.

1 Choose **Help > Launch License Manager**.

2 In the **License Manager** dialog, select the version of ER/Studio Data Architect the license applies to: **Standard**, **Developer**, or **Enterprise**.

3 Choose **Serial > Add**.

4 Copy the serial number from the e-mail, paste it into the **Add Serial Number** dialog, and then click **OK**.

5 Expand **Unregistered Serial Numbers**, right-click the serial number just added, and then click **Register**.

A registration dialog appears.

NOTE: The Registration Code box shows a machine-specific identifier required with other registration methods.

6 Ensure that **Register using Online Registration** is selected.

7 Provide Developer Network credentials in the **Login name or Email** and **Password** fields.

8 Click **Register**.

The activation file should download and install automatically. If this does not happen, click the **Trouble Connecting? Try Web Registration** link and follow the prompts. If you still have problems, see Registering by Phone on page 16.

Registering by Phone

If you have problems with either of the above procedures, you can register licenses by phone. You must provide the following information:

- Developer Network credentials
- The registration code displayed in the Embarcadero License Registration dialog that appears when you start an unlicensed application
- The product base license serial number
- The license serial numbers for any additional features you have purchased.

For North America, Latin America, and Asia Pacific: call (415) 834-3131 option 2, and then follow the prompts. The hours are Monday to Friday, 6:00 A.M. to 6:00 P.M. Pacific time.

For Europe, Africa, and the Middle East: Call +44 (0)1628-684 494. The hours are Monday to Friday, 9 A.M. to 5:30 P.M. U.K. time.

Shortly after phoning in, you will receive an e-mail containing an activation file. Then do the following:

1 Save the activation file to the desktop or to a scratch directory such as `c:\temp`.

2 From within the application, choose **Help > Launch License Manager**.

3 Choose **License > Import**.

4 Navigate to the directory where you saved the activation file, and then double-click the activation file.

5 Click the **Import** button to import the activation file, and then click **Finish**.

Registering a Networked License

If your organization uses networked licensing, an administrator, department head, or someone providing a similar function, will provide you with a. slip activation file. Point to that license file in the Product Registration Window that appears when you launch the product.

If you are upgrading your license, or purchased a license after using a trial license, you can import your license through the License Manager. Click **Help > License Manager** and then select **License > Import** and point to the .slip activation file.

TUTORIALS

The tutorials are intended to help you get started using Embarcadero's data modeling and database design solution, ER/Studio Data Architect (ER/Studio DA).

After completing these tutorials, you'll have the foundation you need to explore the many features and benefits of ER/Studio DA. You will have learned how to do the following:

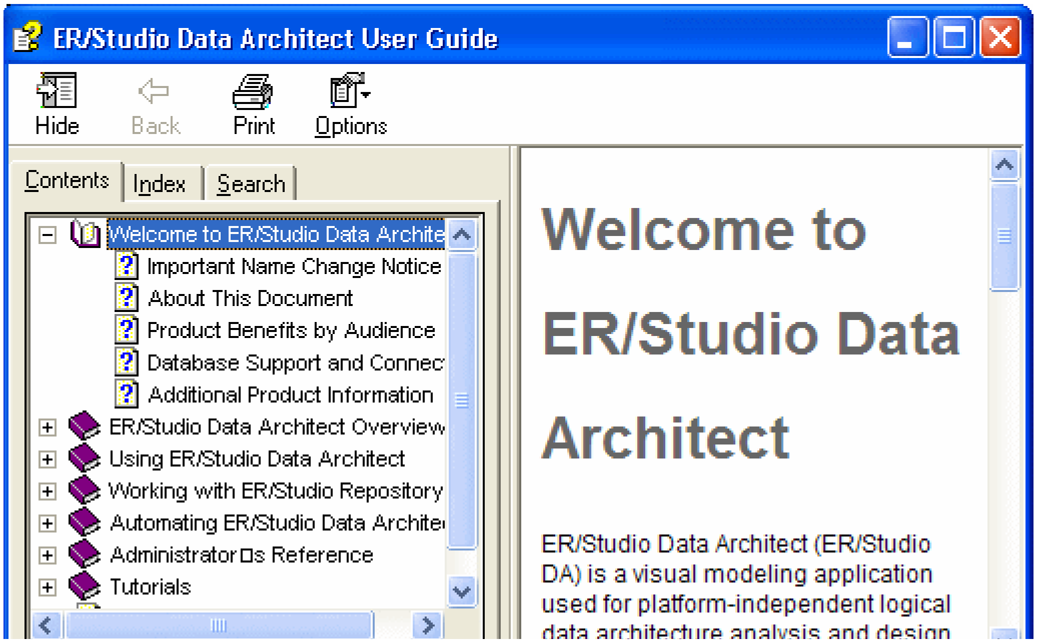
- Create a new data model.
- Work with logical and physical diagrams.
- Leverage productivity-focused features such as powerful reporting engines.
- Use common tasks and commands to make you more productive.

The tutorials are divided into nine sessions. Do them all at once or complete them individually as your time permits.

- Getting Started with ER/Studio DA on page 18
- Logical and Physical Modeling on page 20
- Documenting an Existing Database on page 36
- Documenting Data Lineage on page 47
- Diagram Navigation and Aesthetics on page 56

You can use this basic tutorial as a road map of product highlights, but also to help you find your own path in exploring ER/Studio DA.

Once you have started, from the Main menu you can click Help to find many additional resources that complement and build on many of the activities shown in this brief guide.

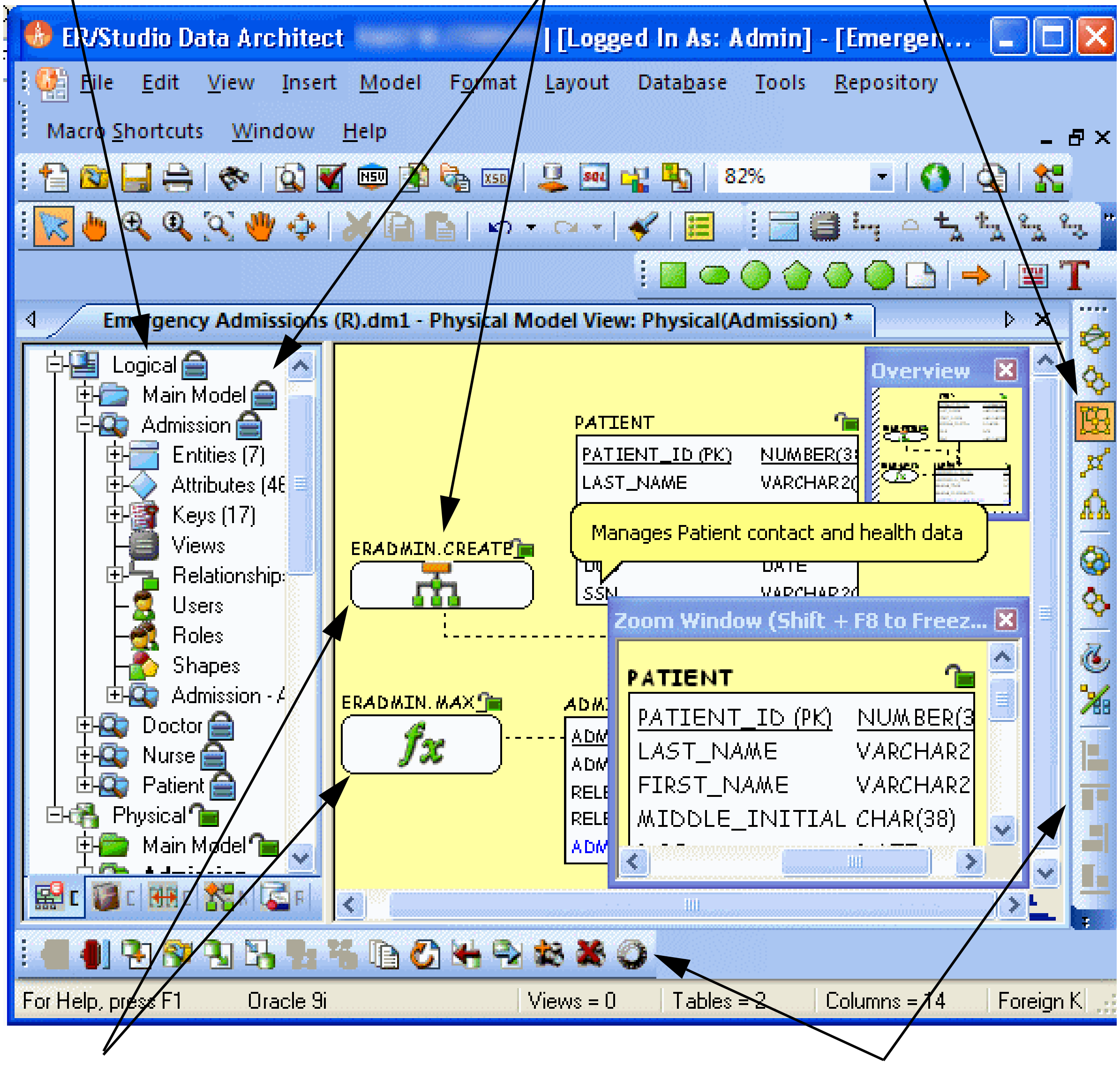

ER/Studio Data Architect User Guide
Hide
Back
Print
Options
Contents
Index
Search
Welcome to ER/Studio Data Archite
Important Name Change Notice
About This Document
Product Benefits by Audience
Database Support and Connec
Additional Product Information
ER/Studio Data Architect Overview
Using ER/Studio Data Architect
Working with ER/Studio Repository
Automating ER/Studio Data Archite
Administrator□s Reference
Tutorials

Welcome to ER/Studio Data Architect

ER/Studio Data Architect (ER/Studio DA) is a visual modeling application used for platform-independent logical data architecture analysis and design

GETTING STARTED WITH ER/STUDIO DA

The graphic below names and describes the functionality of some key elements of the ER/Studio DA user interface.



Data Model Explorer displays information about logical and physical models, submodels and nested submodels.

Repository object status icons display real-time user access information.

Sophisticated diagram auto-layout tools provide single-click clean up of diagram objects.

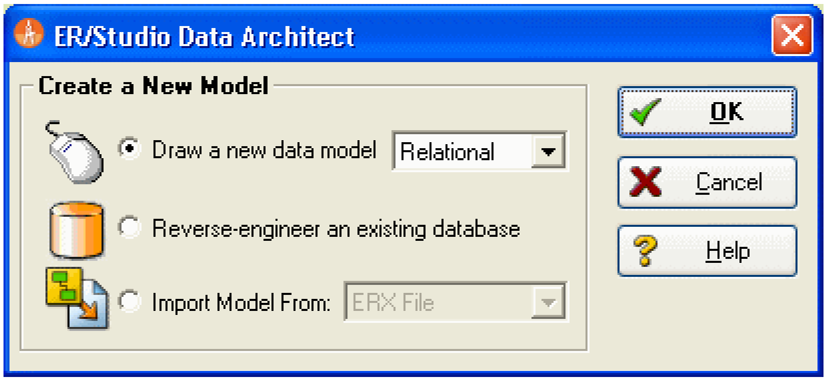
Complex schema objects like functions can be displayed to illustrate dependencies.

Toolbars are dockable anywhere in the ER/Studio DA application window.

Starting to Data Model with ER/Studio DA

1 On the Windows **Start > Programs** menu, choose **Embarcadero > ERStudio Data Architect.**

2 Choose **File > New > Draw a new data model**.



ER/Studio Data Architect

Create a New Model

● Draw a new data model: Relational

○ Reverse-engineer an existing database

○ Import Model From: ERX File

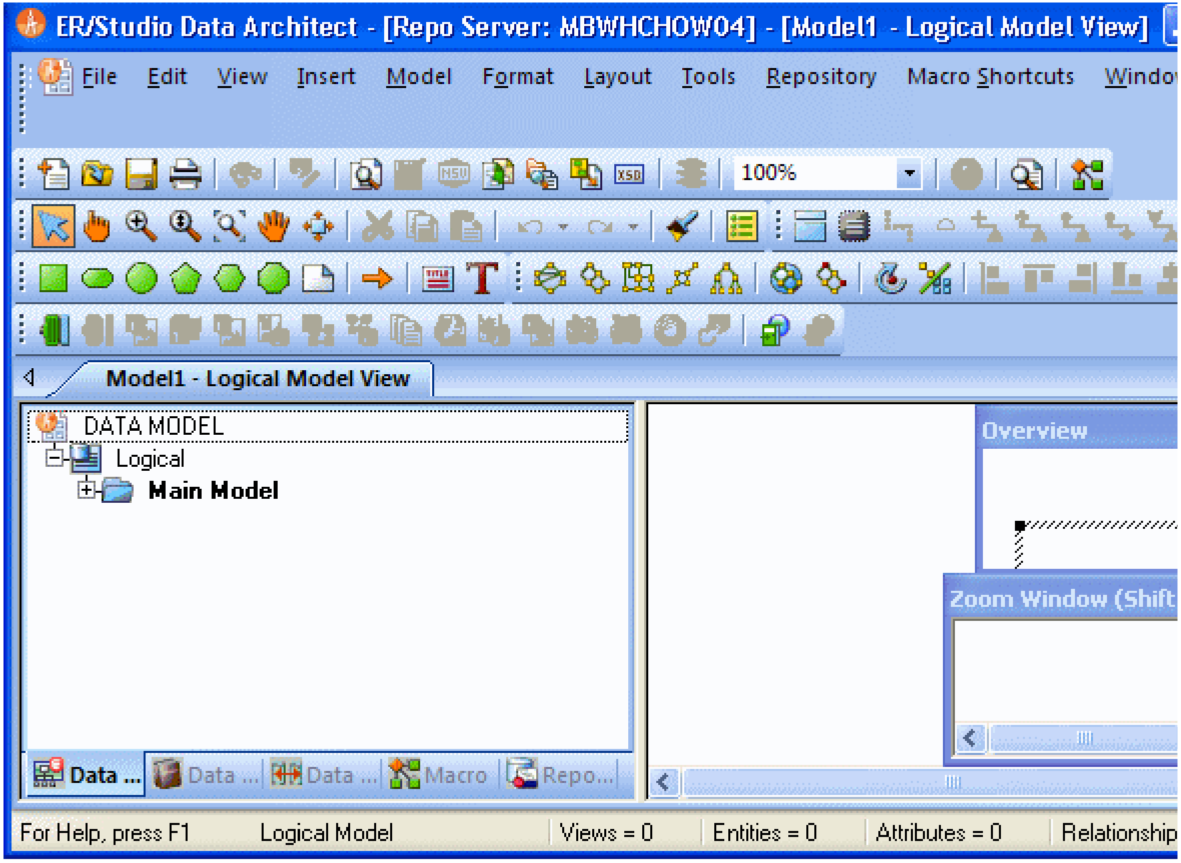
As you can see in the **Create a New Model** dialog, there are a number of ways to begin modeling with ER/Studio DA:

- Build a new design from the ground up by drawing a new data model.
- Build a data model from an existing database through live reverse engineering.
- Import designs from other modeling products such as Computer Associate's ERwin or SQL files.

TIP: You can select an initial layout style for your model before the SQL import takes place.

3 Ensure **Relational** is selected for the new model type and then click **OK**.

After selecting **Draw a new data model** and clicking **OK**, ER/Studio DA will resemble the image below:



Logical and Physical Modeling

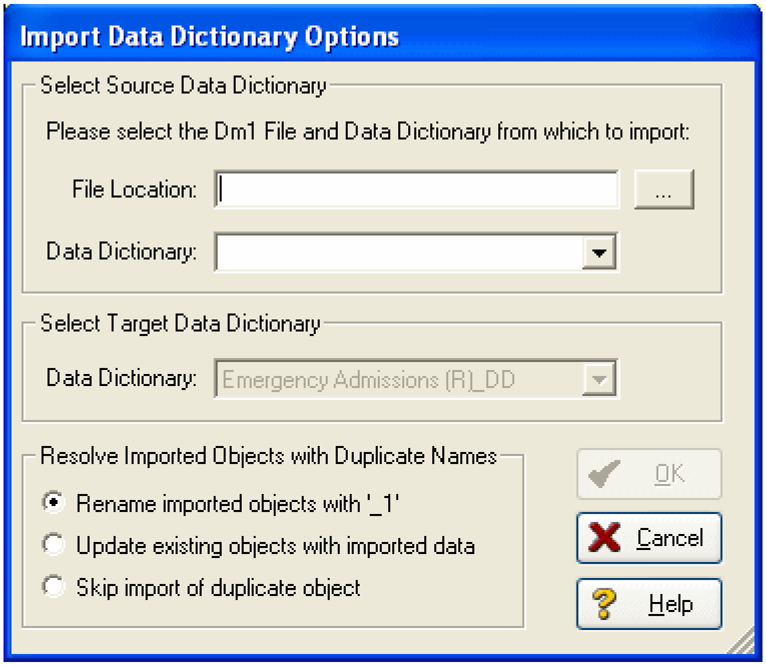
ER/Studio DA supports both logical (non-DBMS or technology-specific) modeling and physical (DBMS-specific) modeling. ER/Studio DA is designed to allow organizations the flexibility to analyze and design a business problem or application logically and generate as many different physical interpretations from the logical model as required. Multiple physical models can be generated from the logical model for the same DBMS (for example, Oracle) or other DBMSs (such as Oracle, SQL Server and DB2). Generating logical and physical models will be discussed in detail in the following sessions.

- Using Data Dictionary Domains to Populate New Entity on page 21
- Establishing Relationships Between Entities on page 24
- Creating and Working with Submodels in ER/Studio DA on page 25
- Generating Physical Models from a Logical Model on page 28
- Denormalizing the Physical Model on page 31
- Finding out How an Entity Maps to the Physical Model on page 34

USING DATA DICTIONARY DOMAINS TO POPULATE NEW ENTITY

As instructed in Getting Started with ER/Studio DA on page 18, you have chosen to draw a new data model to begin a logical model from the ground up. Before we begin to add entities, we will populate ER/Studio DA with some sample domains (Domains are re-usable attributes).

1 Choose **File > Import Data Dictionary**.



Import Data Dictionary Options

Select Source Data Dictionary

Please select the Dm1 File and Data Dictionary from which to import:

File Location:

Data Dictionary:

Select Target Data Dictionary

Data Dictionary: Emergency Admissions (R)_DD

Resolve Imported Objects with Duplicate Names

● Rename imported objects with '_1'

○ Update existing objects with imported data

○ Skip import of duplicate object

2 Next to the **File Location** box, click the ellipsis and browse to the `Sample Models` folder, which is located at:

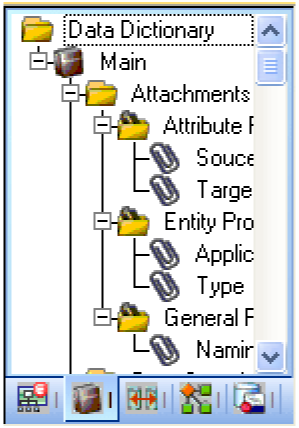
- **For Windows XP**:
  `C:\Documents and Settings\All Users\Application Data\Embarcadero\ERStudioDA_X.X\Sample Models`
- **For Windows Vista**:
  `C:\ProgramData\Embarcadero\ERStudioDA_X.X\Sample Models`

3 Double-click the `Orders.dm1` sample model and then click **OK**.

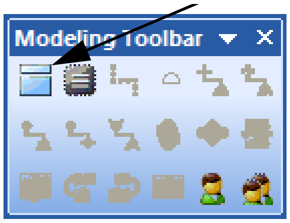
This model contains a pre-populated, sample data dictionary.

NOTE: Under Resolve Imported Objects with Duplicate Names, you can choose between a couple of options to determine how the dictionary objects are imported. This is important when importing into a diagram that already has dictionary objects in it.

Once opened, you will see that the ER/Studio DA Model Explorer has automatically switched to the **Data Dictionary** tab to allow immediate drag and drop access to domains.

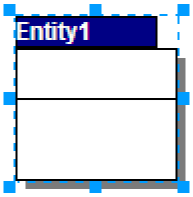


4 Now, to add an entity to the Data Model Window, click the **Entity** tool on the **Modeling Toolbar**,



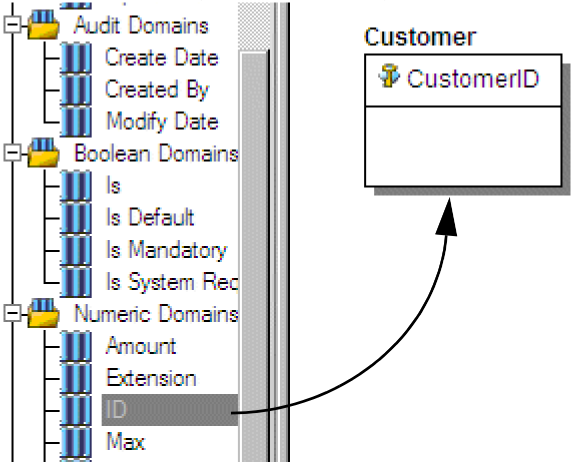
and then click in the Data Model Window to drop the entity.

TIP: The cursor will change to an entity symbol once the **Entity** tool is clicked, so you can click to drop as many entities on the Data Model Window as you want.



5 Right-click to return your mouse to the selection cursor.

6 In the entity name field, type `Customer`, replacing the default entity name, Entity1.



7 In the **Domains** folder of the **Data Dictionary** tab, locate the **ID** domain in the **Numeric Domains** folder.

8 Click the **ID** domain (do not release your mouse), drag it onto the **Customer** entity, and then release it just below the entity's name, which is the entity's Primary Key field.

> **TIP:** You can edit or rename an entity and insert, edit or rename attributes by holding down the **Shift** key and then clicking the name or attribute. Pressing the **Tab** key cycles between the entity's name, and primary key and non-primary key fields. After naming the field, press **Return** to insert a new field.

> **TIP:** If you need to zoom in on the entity to read the entity name and attributes, press **F8** to view the **Zoom Window**, then use the scroll bars of the **Zoom Window** to center the entity on the Data Model Window. Press **Shift** while rolling the mouse wheel forward to increase the view magnification. You can use the **Pan** tool to reposition the view to better see the entity.

9 In the entity, click **ID**, the name of the attribute we just created from the ID domain, and change its name to `CustomerID`, as seen in the illustration above.

10 Repeat the process in step 7 and step 8 to populate the Customer entity with the following domains:

- **Name** and **Phone** from the **Property Domains** folder
- **Address**, **City**, **State**, **Zip Code** from the **Address Domains** folder.

11 Drop another entity on the **Data Model Window** and call it `Order`.

12 Drag the ID domain onto the Order entity's Primary Key field, change the ID domain name to `OrderID`.

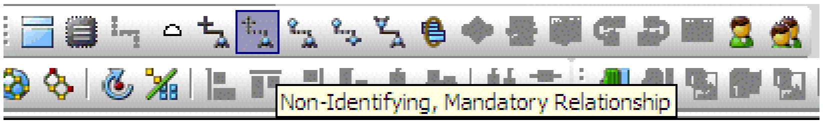
13 Right-click the Order entity and select **Comments**.

14 Enter some text in the **Comments** editor, click **Add** and then click **OK**. You can add comments to any database object in the same manner. This can be very useful when you want to provide feedback on a model to other users.

15 Save your data model. We will use it in the next session of this tutorial.

Establishing Relationships Between Entities

1 On the **Modeling toolbar**, click the **Non-Identifying, Mandatory Relationship** tool.



NOTE: The screen shots of this tutorial were taken using a model where the notation was set to IE (Crow's Feet). Depending upon the notation your model is currently set to, the icons for each relationship will be slightly different.

TIP: You can change the model notation by choosing **Model > Model Options**, and then selecting another notation option in the Notation area.

2 To establish a relationship between Customer and Order, click the parent entity, **Customer** and then click the child entity, **Order**.

NOTE: ER/Studio DA supports sound design practices by automatically propagating the primary key, from parent to child entities. If there are candidate alternate keys that should also be propagated to the child, in the **Relationship Editor** you can choose all available parent entity keys in the Parent Key list. Deleting a relationship will remove a non-native propagated attribute. However, if you want to keep the child columns of the relationship or foreign constraint, when you delete the relationship you can check the Make Foreign Keys Native option. In this case, if the relationship between Customer and Order is deleted, the CustomerID will be left in the Order entity.

What is a Domain? A Domain is a valuable tool in establishing standard, re-usable Attributes/ Columns. They allow data modelers to create a data element once (such as an ID field you require all of your entities to leverage as its primary key) which has the same data type, definition, rule, and constraint no matter where the data element is distributed and bound. Read more about Domains in ER/Studio DA Help.

CREATING AND WORKING WITH SUBMODELS IN ER/STUDIO DA

Now that you have a general understanding of how to build logical models from the ground up in ER/Studio DA, it is important to understand how to work with an important navigation feature of ER/Studio DA, called Submodels. Submodels and nested submodels are designed to break down large, complicated views of a data model in order to focus on a specific area. An important aspect of Submodels to understand is that any changes made in the submodel…*other than layout, color, display settings, notation, etc. which can be unique to the submodel*…will occur automatically in the Main Model view. In other words, change or add an attribute to an object in a Submodel and the change is automatically propagated to its Main Model counterpart.

Close the current sample model, and open a more mature model. Use this exercise to learn more about submodeling.

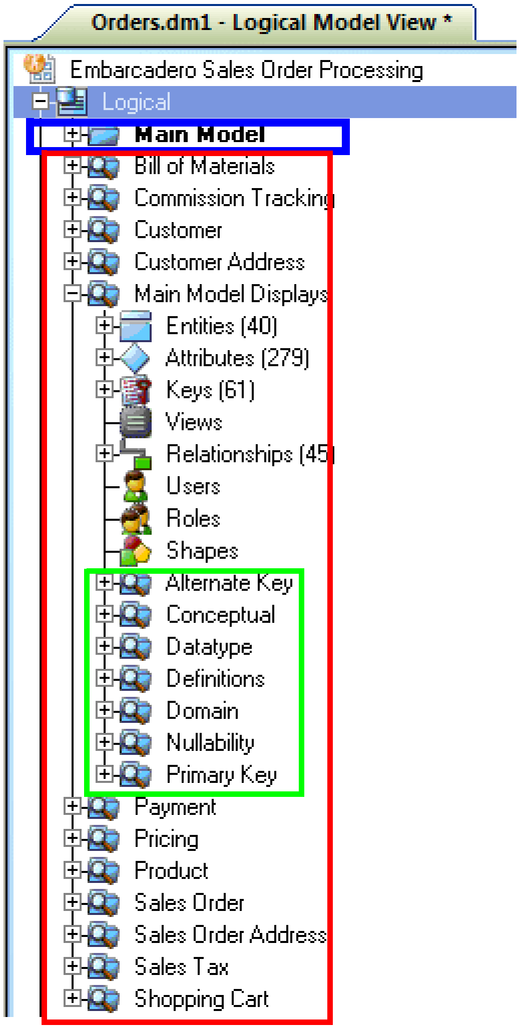
1 Choose **File > Open**.

2 Select **Orders.dm1** and then click **Open**.

3 To preserve this sample model for future use, choose **File > Save As** and then save the `Orders.dm1` file with a new name.

In this exercise, we will be modifying this model.

4 Collapse the folders in the **Data Model** tab of the Data Model Explorer to look like the image below:



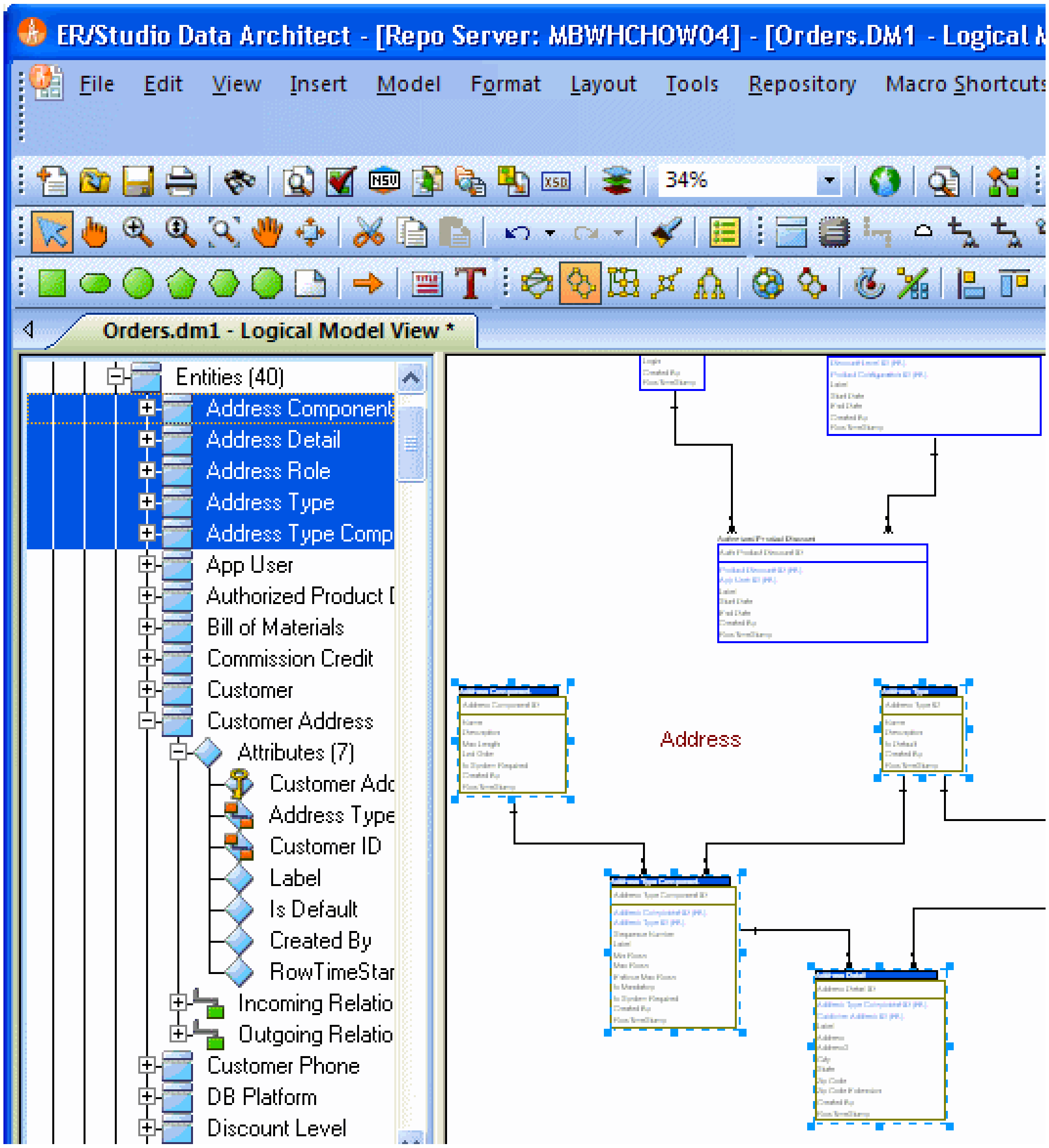
In the `Orders.DM1` sample model, there are no physical models. This model includes several submodel folders that help to describe the logical model:

- Main Model – This is the entire collection of all logical objects in the Orders.DM1 file. Note the absence of the magnifying glass on the folder icon which designates it as the main model.
- Bill of Materials through Shopping Cart – These are submodels, which are smaller collections of entities derived from the Main Model that help to describe specific areas of the Main Model free from other entities.
- Alternate Key through Primary Key – These are nested submodels, which can go ‘n’ levels deep and are literally submodels of submodels.

Feel free to explore. Click the plus sign (+) to expand these folders.

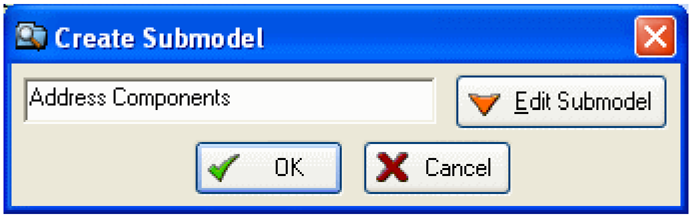
Let’s create a submodel with all the objects related to the `Orders.DM1` Address components.

5 To make a new submodel, navigate to Logical Main model and with the **CTRL** key depressed, click the objects in the Data Model Explorer, as seen in the image below.



NOTE: Any objects selected in the Data Model Explorer will also be selected on the Data Model Window as well (also seen here). You can also select entities by lassoing them on the Data Model Window.

6 With the entities selected, choose **Model > Create Submodel**.



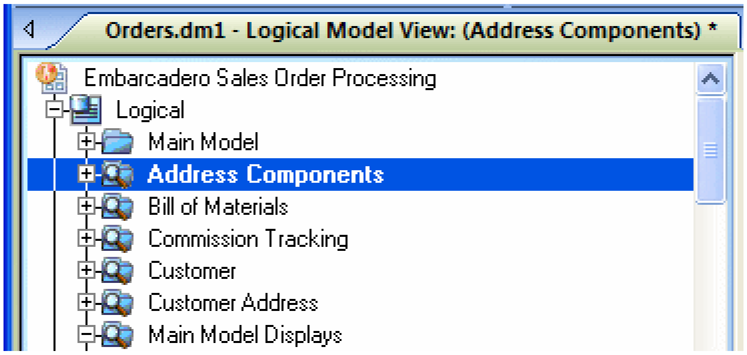
Create Submodel

Address Components

7 Below **Create Submodel**, enter `Address Components` as the name for the submodel.

8 Click **OK**.

ER/Studio DA creates the **Address Components** submodel.



What do the results look like and how do I navigate to the submodel? Once created, you will see the new submodel listed in the Data Model Explorer, denoted as a submodel by the magnifying glass over its folder, as in the case with Bill of Materials and the other submodels.

GENERATING PHYSICAL MODELS FROM A LOGICAL MODEL

ER/Studio DA can generate as many physical models from a single logical model as desired. There are many ways to leverage multiple physical models in ER/Studio DA to help the design process. Examples of how multiple physical models are used are:

- Managing change in an existing application: Maintain independent development, test, and production physical model diagrams that represent specific databases.
- Migrating database applications: Use ER/Studio DA as an analysis and design hub for migrating database applications. Manage a physical model of the legacy source database application in addition to its new target physical model, which can be for an entirely new DBMS than originally maintained in the legacy database.

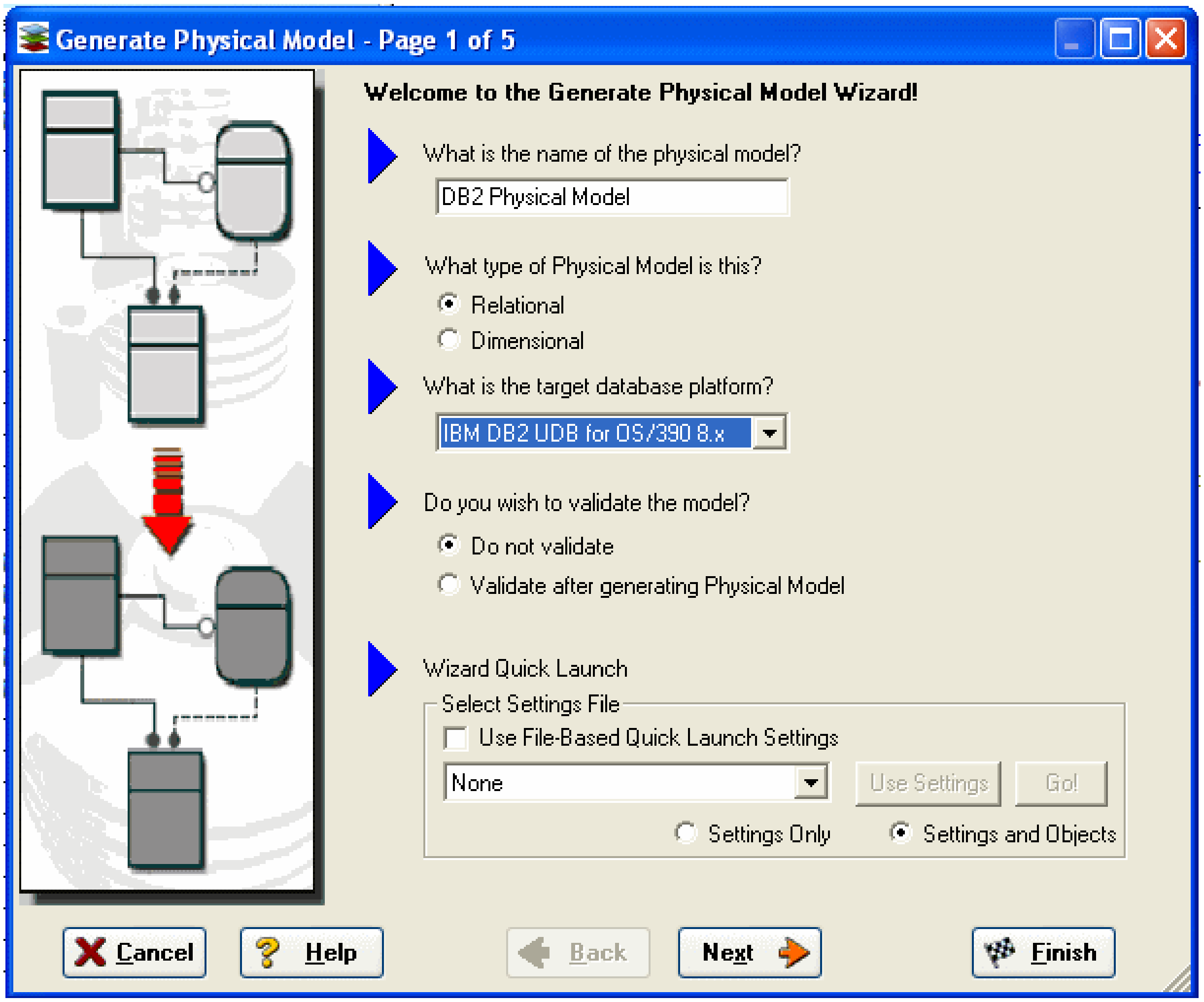
Let's generate a new physical model from a logical model in order to build a database. We will use the `Orders.DM1` sample model.

1 Open your version of the `Orders.DM1` sample model.

TIP: Use the steps shown in the last session to do so.

2 Select the **Main Model** and then click **Model > Generate Physical Model**.

ER/Studio DA invokes a step-by-step wizard to walk you through the process of generating a DBMS-specific physical model.



Generate Physical Model - Page 1 of 5

Welcome to the Generate Physical Model Wizard!

What is the name of the physical model?
DB2 Physical Model

What type of Physical Model is this?
● Relational
○ Dimensional

What is the target database platform?
IBM DB2 UDB for OS/390 8.x

Do you wish to validate the model?
● Do not validate
○ Validate after generating Physical Model

Wizard Quick Launch
Select Settings File
□ Use File-Based Quick Launch Settings
None
○ Settings Only ● Settings and Objects

3 Name the new physical model, `DB2 Physical Model` and then select **DB2 UDB for OS/390 8.x** as the target DBMS to generate.

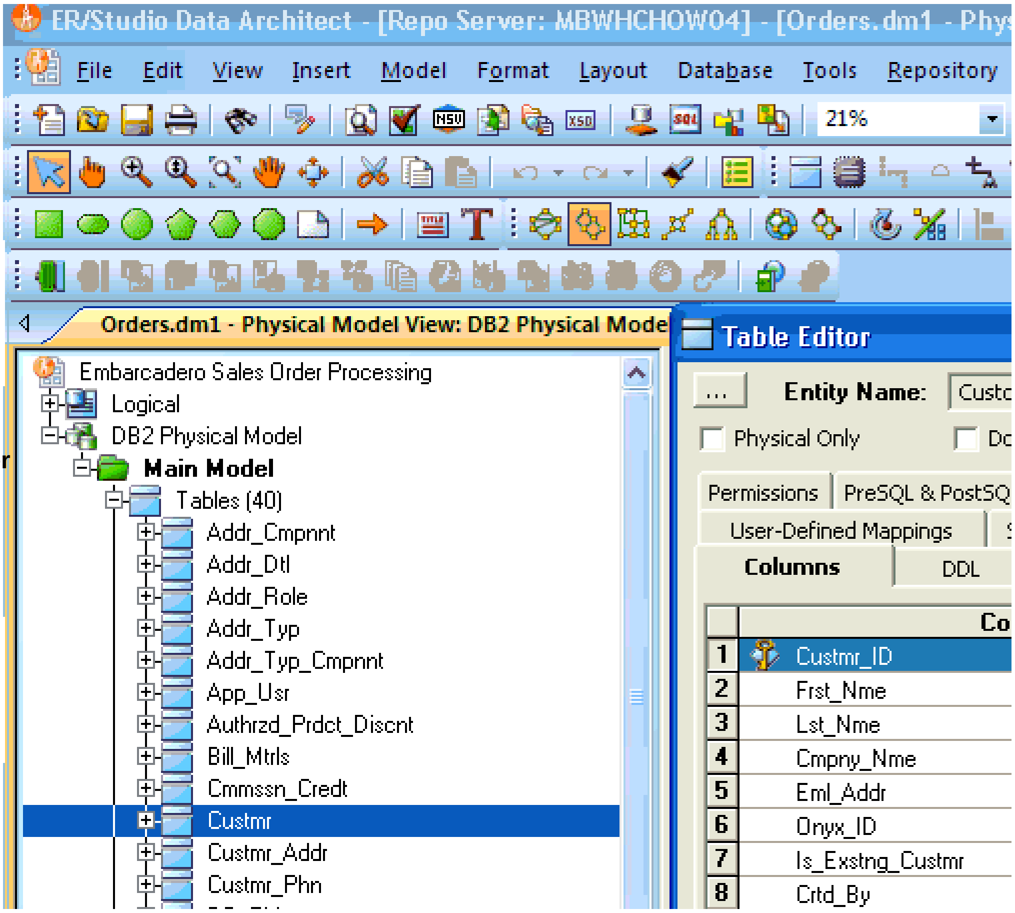
4 Continue through the **Generate Physical Model** wizard, which prompts very clear and concise questions about how you want your physical model to be generated.

NOTE: The wizard prompts you to customize items such as individual object selection, index assignment, default storage parameters, resolution of many-to-many relationships that may be in the logical model, and naming conventions. A DBMS-specific validation check is also provided in this wizard.

TIP: The **Quick Launch** can store common settings so that an operation can be reused on this model or on any other models. You can reuse the settings on another model by choosing the **Use File-Based Quick Launch Setting** option when saving the Quick Launch information on the last page of the wizard.

5 To generate the new Physical Model, on the last page of the wizard, click **Finish**.

Now that a physical model has been generated from the logical model, feel free to navigate to specific objects via the Data Model Explorer, such as the CUSTMR table selected here. Double-click and view the physical details of the object such as DDL, Indexes, Partitions, and Storage.



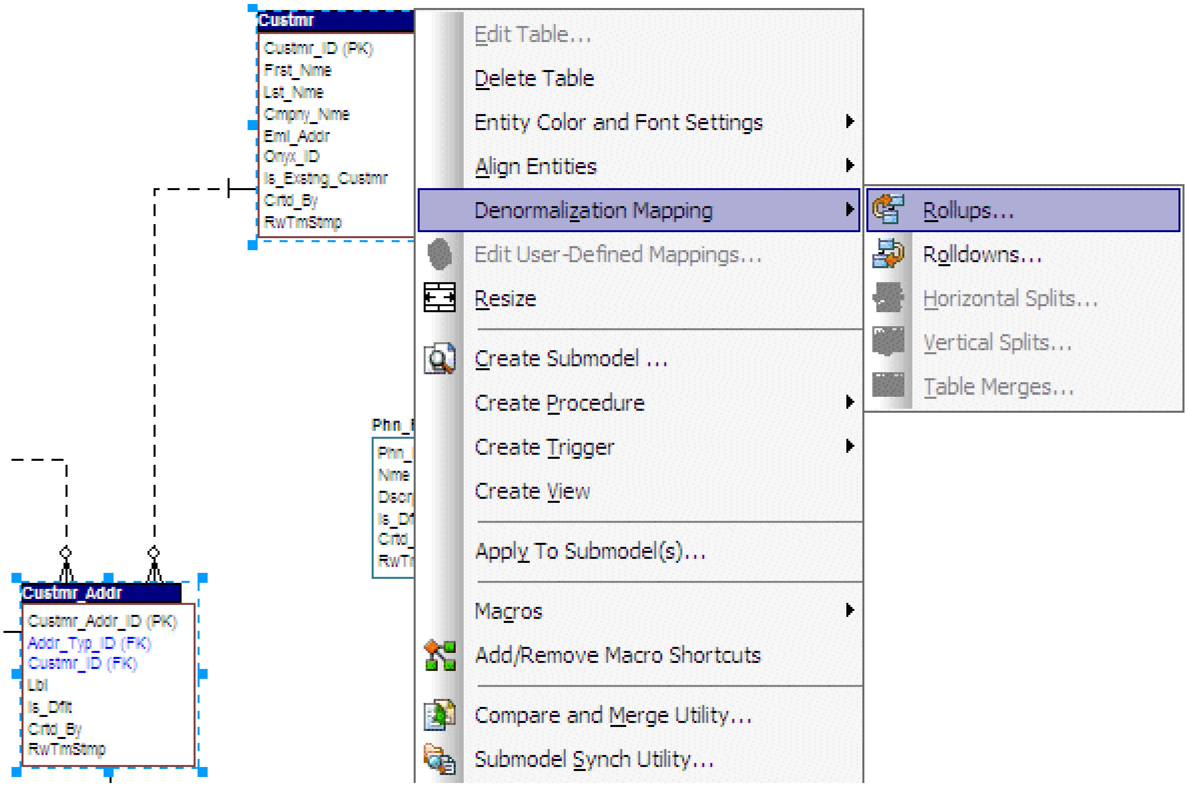
New Physical Model in the Data Model Explorer

Double-click CUSTMR, to invoke the Table Editor and view design details, such as the DDL script.

DENORMALIZING THE PHYSICAL MODEL

ER/Studio DA comes equipped with denormalization wizards to help you optimize the physical design once the physical model is generated. The wizards help automate the process and keep the ties between the physical tables and the logical entities.

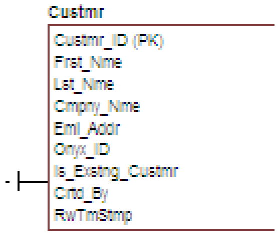
The active, denormalization wizards available depend on which tables are selected in the physical model when you select Denormalization Mapping. For example, if two tables that are related to each other are selected, the valid operations would be Rollups or Rolldowns.



When only one table is selected, the option to split the tables becomes available. The Table Merge option is available when two unrelated tables are selected.

Let's walk through an example of a denormalization operation using the generated physical model in a previous session of this tutorial. We may want to reduce the overhead on the Custmr table by splitting it into two physical tables, Custmr_East and Custmr_West. Splitting the table can reduce the query time and provide opportunities to store the tables at different locations which could further reduce lookup time.

Before the operation, the Custmr table should look like:



Custmr

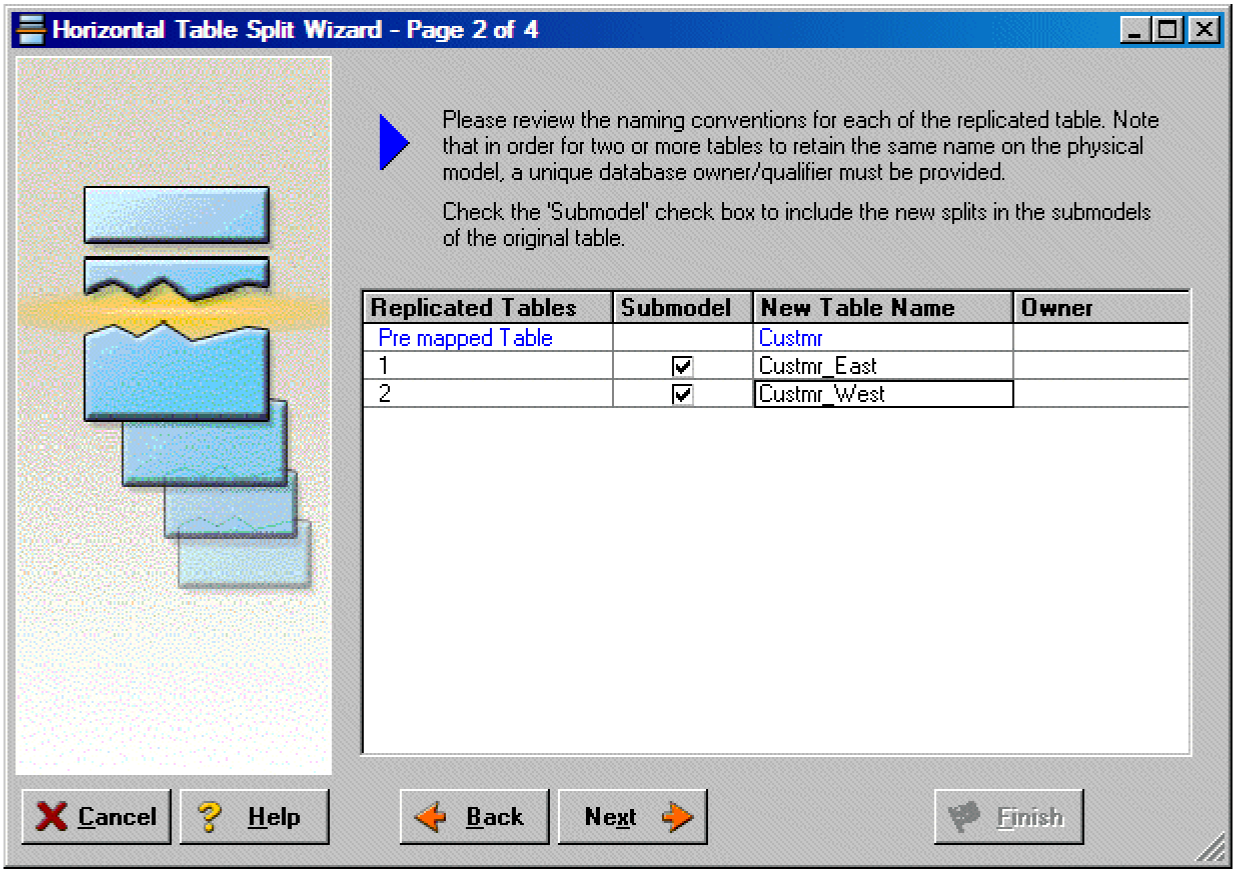
Custmr_ID (PK)
Frst_Nme
Lst_Nme
Cmpny_Nme
Eml_Addr
Onyx_ID
Is_Exstng_Custmr
Crtd_By
RwTmStmp

1. Open the `Orders1.dm1` model you modified and saved in the last session.
2. In the Data Model Explorer, right-click the **Custmr** table in the Physical Model.
3. Choose **Denormalization Mapping > Horizontal Splits**.

   Notice that since only Custmr is selected, the only possible mappings are vertical and horizontal splits.

   The **Horizontal Table Split Wizard** launches.
4. On Page 1, type `2` for the number of splits.

5 On Page 2, rename splits1 and 2 to `Custmr_East` and `Custmr_West` respectively.



Please review the naming conventions for each of the replicated table. Note that in order for two or more tables to retain the same name on the physical model, a unique database owner/qualifier must be provided.

Check the 'Submodel' check box to include the new splits in the submodels of the original table.

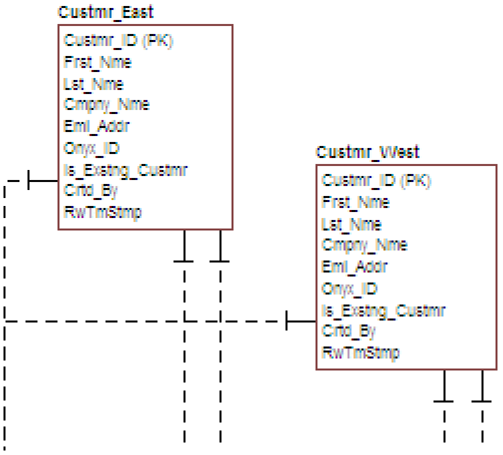
Replicated Tables	Submodel	New Table Name	Owner
Pre mapped Table		Custmr	
1	☑	Custmr_East	
2	☑	Custmr_West	

6 On Page 3, click **Next**.

We will keep all the relationships.

7 On Page 4, type a name and definition for the denormalization operation, and then click **Finish**.

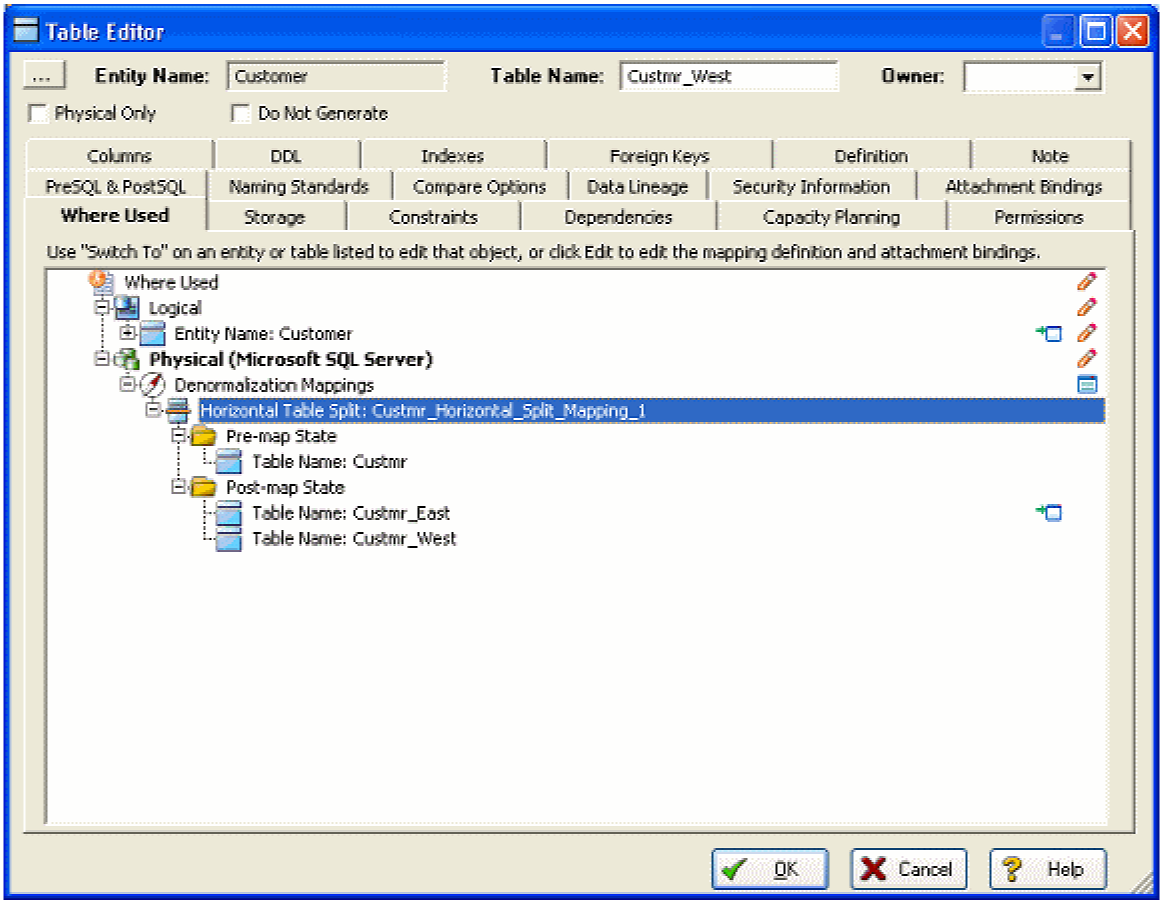
Finished! *After* the split the Custmr table will be two physical tables that look like this:



The two tables are identical except for the name.

You can selectively choose which attributes are included in the resultant tables by using a vertical split.

The denormalization mapping is stored with each of the tables.



Entity Name: Customer
Table Name: Custmr_West
Owner:

□ Physical Only
□ Do Not Generate

Use "Switch To" on an entity or table listed to edit that object, or click Edit to edit the mapping definition and attachment bindings.

```
Where Used
  Logical
    Entity Name: Customer
  Physical (Microsoft SQL Server)
    Denormalization Mappings
      Horizontal Table Split: Custmr_Horizontal_Split_Mapping_1
        Pre-map State
          Table Name: Custmr
        Post-map State
          Table Name: Custmr_East
          Table Name: Custmr_West
```

You can use the denormalization information to undo the operation or see the history of what happened. ER/Studio DA tracks the before and after states of these operations. This comes in handy in the next section where we discuss the **Where Used** analysis that can be performed between the logical and physical models.

Finding out How an Entity Maps to the Physical Model

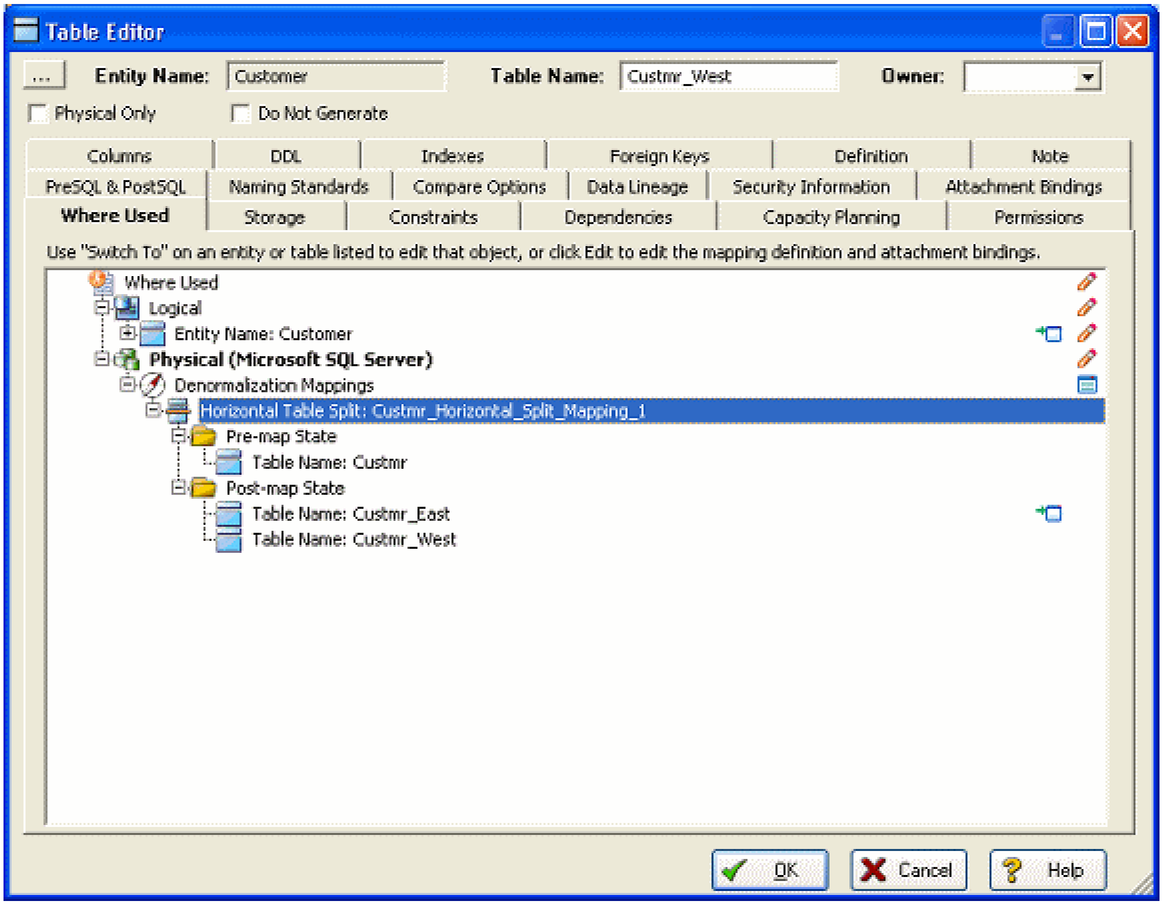
Now that we have performed a denormalization operation, the logical entity, Customer, essentially has become two physical tables, Custmr_East and Custmr_West. The ties between the logical and physical models are not lost. ER/Studio DA allows you to see what Customer in the logical model maps to in the DB2 physical model.

Let's take a look at the Customer entity in the logical model.

1 In the Data Model Explorer, navigate back to the **Customer** entity in the Logical model.

2 To start the Entity Editor, double-click the **Customer** entity.

3 Click the **Where Used** tab.



Entity Name: Customer Table Name: Custmr_West Owner:

□ Physical Only □ Do Not Generate

Use "Switch To" on an entity or table listed to edit that object, or click Edit to edit the mapping definition and attachment bindings.

```
Where Used
  Logical
    Entity Name: Customer
  Physical (Microsoft SQL Server)
    Denormalization Mappings
      Horizontal Table Split: Custmr_Horizontal_Split_Mapping_1
        Pre-map State
          Table Name: Custmr
        Post-map State
          Table Name: Custmr_East
          Table Name: Custmr_West
```

Once the tree is expanded, you can see the lineage of what has happened to the object. Notice that `Custmr_East` and `Custmr_West` are listed as physical implementations of the `Customer` entity. The denormalization mapping object shows how the end result was achieved.

The **Where Used** tab also displays the submodel usage of a particular entity within the logical or physical model. This allows you to see which business areas the entity belongs to.

NOTE: **Where Used** information is also available for attributes and columns.

Conclusion

In this session, you have seen how incredibly quick and easy it is to:

- Build a logical data model from scratch.

- Create a new submodel view to understand how to model on specific parts of a larger Main Model.
- Generate a physical model from a logical database in preparation for building a new database.
- Denormalize objects in the physical model.
- View the mappings between the logical and physical models using the Where Used tab.

For more information, refer to the ER/Studio DA Help and then review the topics "Developing the Logical Model" and Developing the Physical Model," "Synchronizing Physical and Logical Models," and "Generating a Script File or Database."

Documenting an Existing Database

One of ER/Studio DA's most powerful applications is that of a documentation generator to communicate complex databases and associated metadata to the Enterprise. ER/Studio DA is equipped with extensive report generation capabilities:

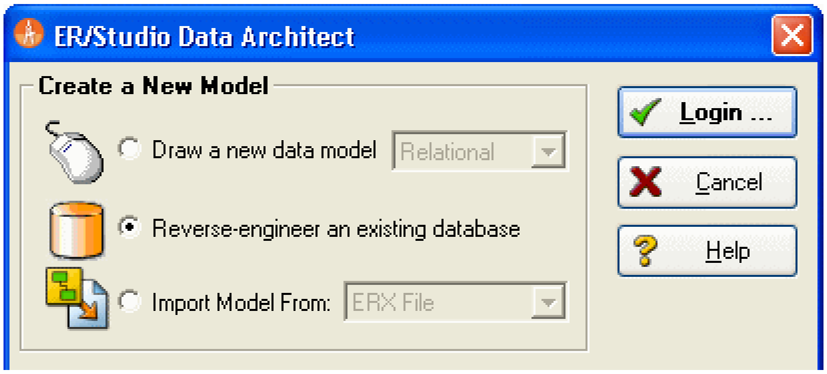
- **HTML Report Generation**: Instantaneous generation of an HTML-based Web site designed to provide simple navigability through data models and model metadata using standard browsers such as Microsoft's Internet Explorer or Mozilla Firefox.
- **RTF Report Generation**: Instantaneous documentation generation compatible with applications like Microsoft Word.

In the exercise below, we will reverse-engineer an existing database and generate an HTML report for distribution and navigation to those who depend upon the information about the data model, but who may not be permitted to connect to the database for security or organizational reasons.

GENERATING AN HTML INTRANET DICTIONARY REPORT

PRE-REQUISITE: This exercise assumes that you can connect to an existing database in order to document it. Please refer to "Reverse-Engineering an Existing Database" in ER/Studio DA Help for explicit setup details if needed. If you cannot connect to an existing database, you can still generate documentation from the installed sample models; skip steps 1 through step 7 below which relate to in reverse-engineering and begin at step 8 after opening a sample model included with ER/Studio DA.

1 Choose **File > New**.

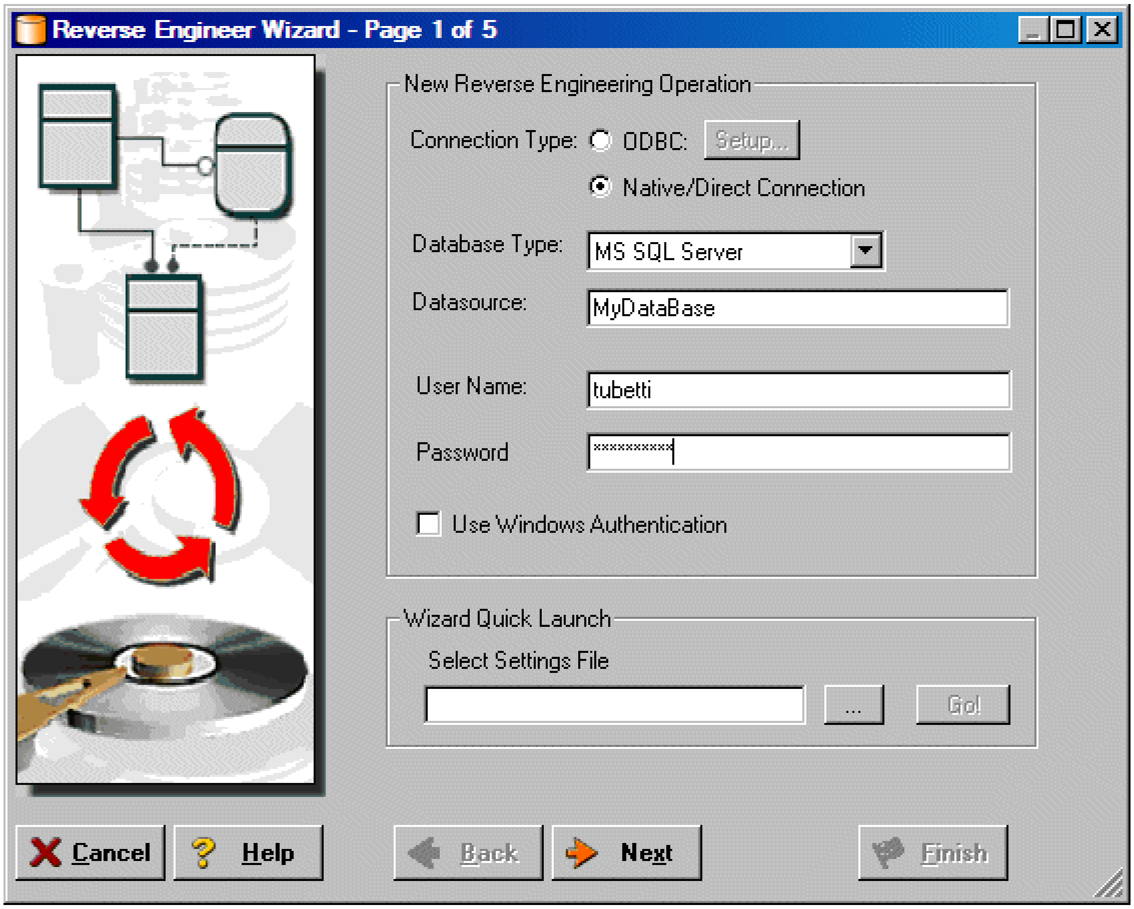


2 Select **Reverse-engineer an existing database**.

3 Click **Login**.

You can reverse engineer the database from either an ODBC datasource or via Native RDBMS client connectivity. In this example, Native Connectivity to Microsoft SQL Server will be demonstrated.

4 Type the relevant connectivity information such as the data source name, and user name and password and then click **Next**.



New Reverse Engineering Operation

Connection Type: ○ ODBC: ● Native/Direct Connection

Database Type: MS SQL Server

Datasource: MyDataBase

User Name: tubetti

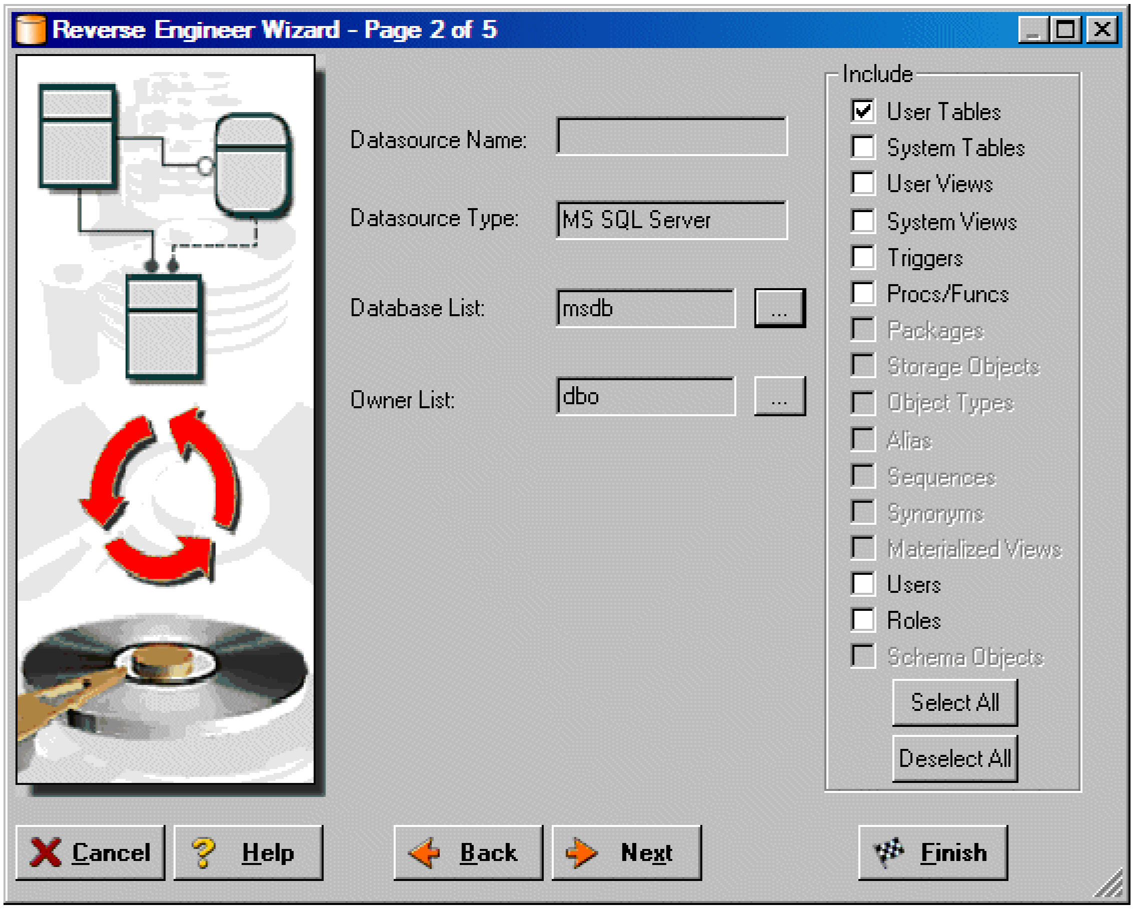
Password: **********

□ Use Windows Authentication

Wizard Quick Launch

Select Settings File

5 Walk through the **Reverse Engineer Wizard** selecting the objects, options, and layout preferences for the model.



Reverse Engineer Wizard - Page 2 of 5

Datasource Name:

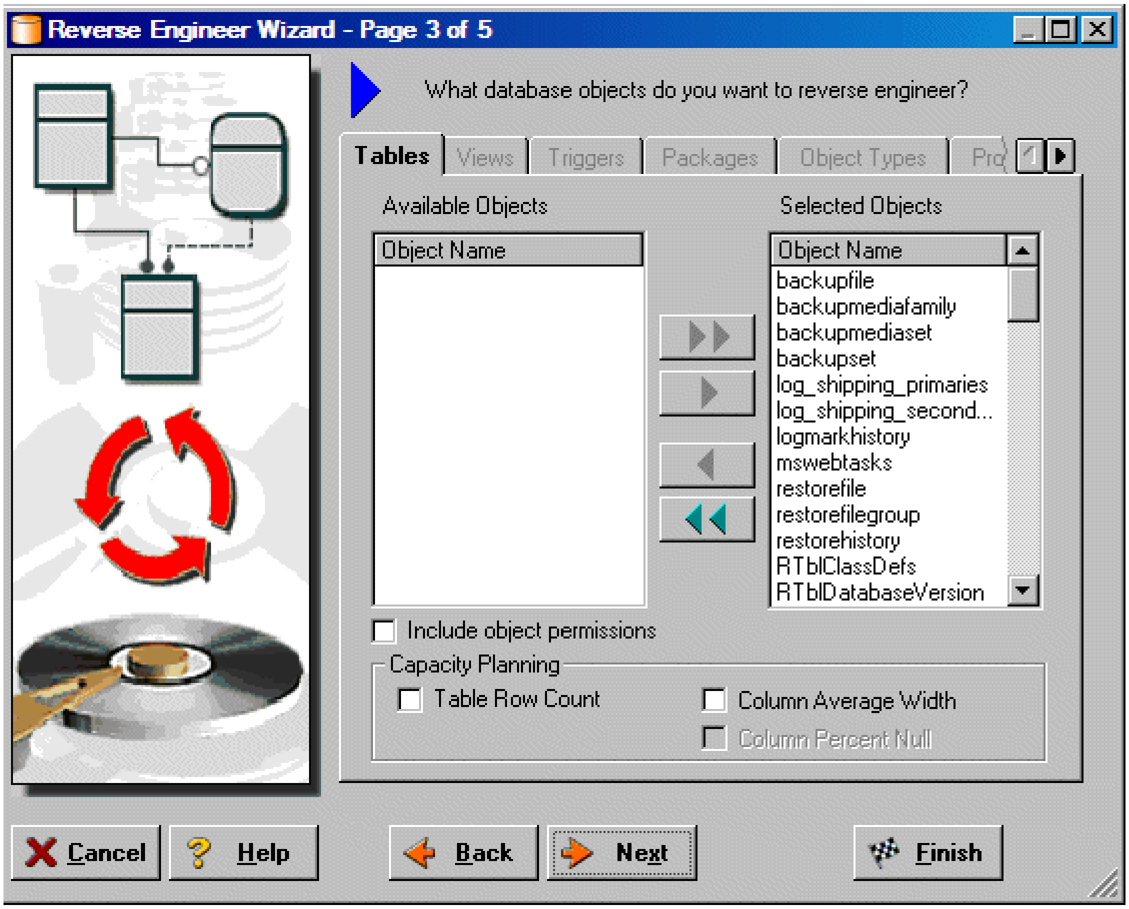
Datasource Type: MS SQL Server

Database List: msdb

Owner List: dbo

Include

☑ User Tables
☐ System Tables
☐ User Views
☐ System Views
☐ Triggers
☐ Procs/Funcs
☐ Packages
☐ Storage Objects
☐ Object Types
☐ Alias
☐ Sequences
☐ Synonyms
☐ Materialized Views
☐ Users
☐ Roles
☐ Schema Objects



What database objects do you want to reverse engineer?

Available Objects

Object Name

Selected Objects

Object Name
backupfile
backupmediafamily
backupmediaset
backupset
log_shipping_primaries
log_shipping_second...
logmarkhistory
mswebtasks
restorefile
restorefilegroup
restorehistory
RTblClassDefs
RTblDatabaseVersion

□ Include object permissions

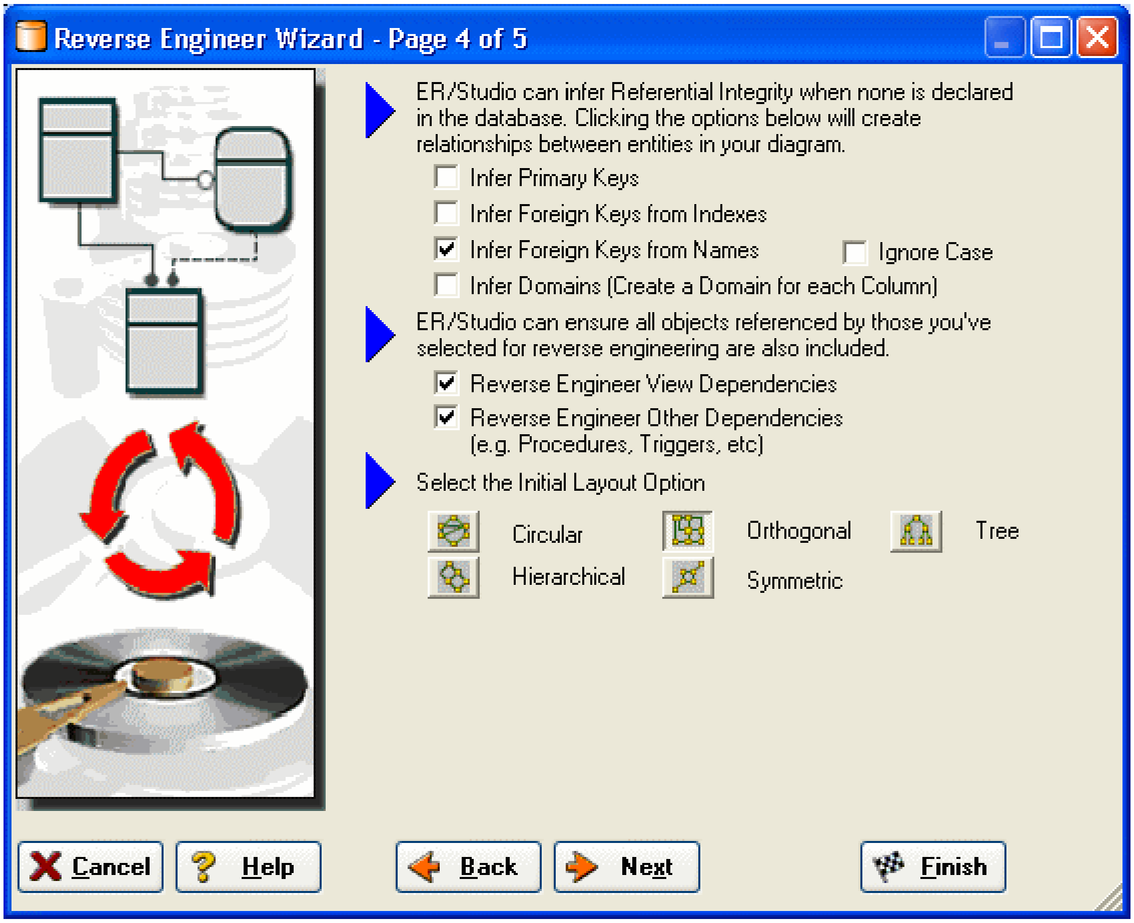
Capacity Planning

□ Table Row Count

□ Column Average Width

□ Column Percent Null

6 Continue through the wizard to select layout styles and other preferences.



Reverse Engineer Wizard - Page 4 of 5

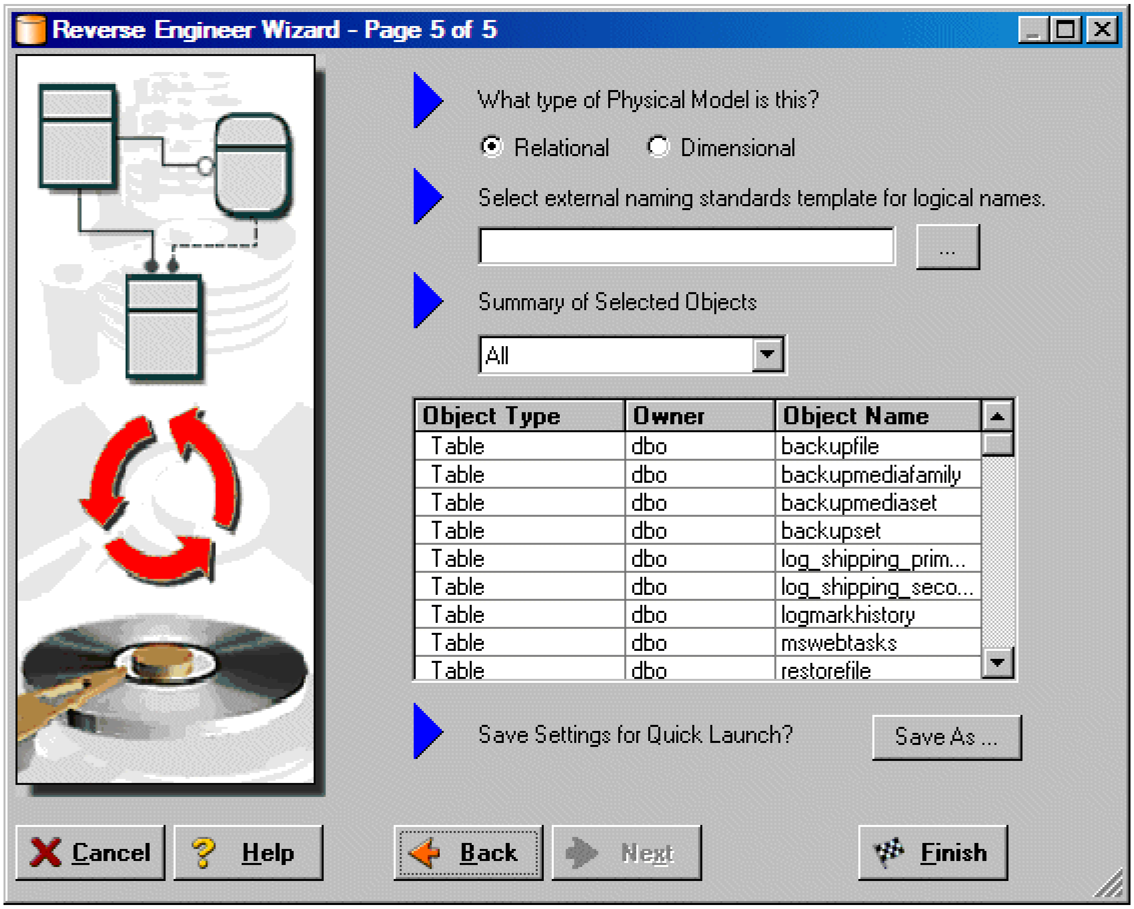
ER/Studio can infer Referential Integrity when none is declared in the database. Clicking the options below will create relationships between entities in your diagram.

- □ Infer Primary Keys
- □ Infer Foreign Keys from Indexes
- ☑ Infer Foreign Keys from Names □ Ignore Case
- □ Infer Domains (Create a Domain for each Column)

ER/Studio can ensure all objects referenced by those you've selected for reverse engineering are also included.

- ☑ Reverse Engineer View Dependencies
- ☑ Reverse Engineer Other Dependencies (e.g. Procedures, Triggers, etc)

Select the Initial Layout Option



Reverse Engineer Wizard - Page 5 of 5

What type of Physical Model is this?

● Relational ○ Dimensional

Select external naming standards template for logical names.

Summary of Selected Objects

All

Object Type	Owner	Object Name
Table	dbo	backupfile
Table	dbo	backupmediafamily
Table	dbo	backupmediaset
Table	dbo	backupset
Table	dbo	log_shipping_prim...
Table	dbo	log_shipping_seco...
Table	dbo	logmarkhistory
Table	dbo	mswebtasks
Table	dbo	restorefile

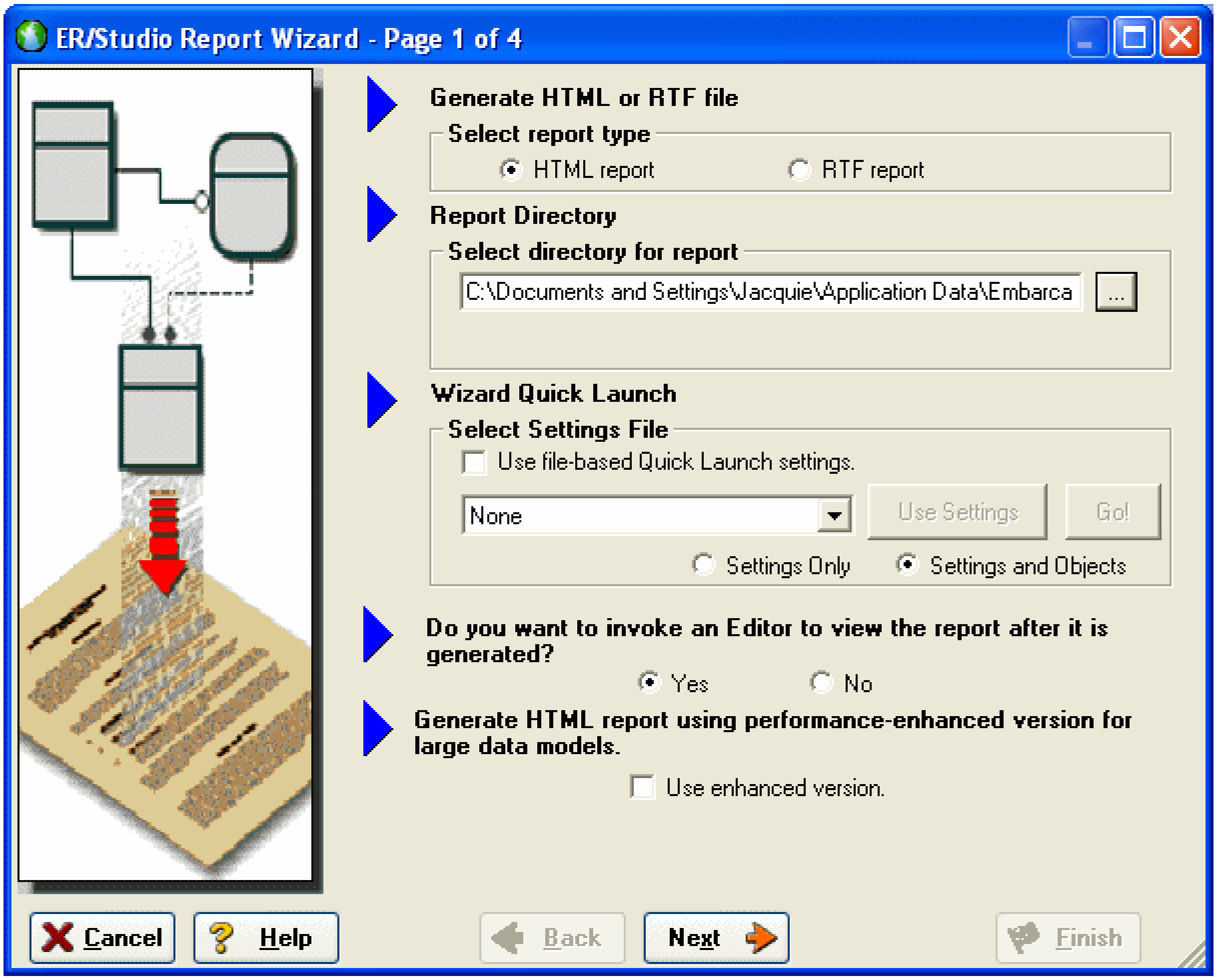
Save Settings for Quick Launch?

7 Click **Finish** and ER/Studio DA reverse engineers your database!

Once reverse engineering of your database is complete, we will generate a complete HTML report of the database for others in your organization to review.

8 In the Data Model Explorer, select the **Physical Main Model**.

9 Choose **Tools > Generate Reports**.



ER/Studio Report Wizard - Page 1 of 4

Generate HTML or RTF file

Select report type

● HTML report ○ RTF report

Report Directory

Select directory for report

C:\Documents and Settings\Jacquie\Application Data\Embarca

Wizard Quick Launch

Select Settings File

□ Use file-based Quick Launch settings.

None

○ Settings Only ● Settings and Objects

Do you want to invoke an Editor to view the report after it is generated?

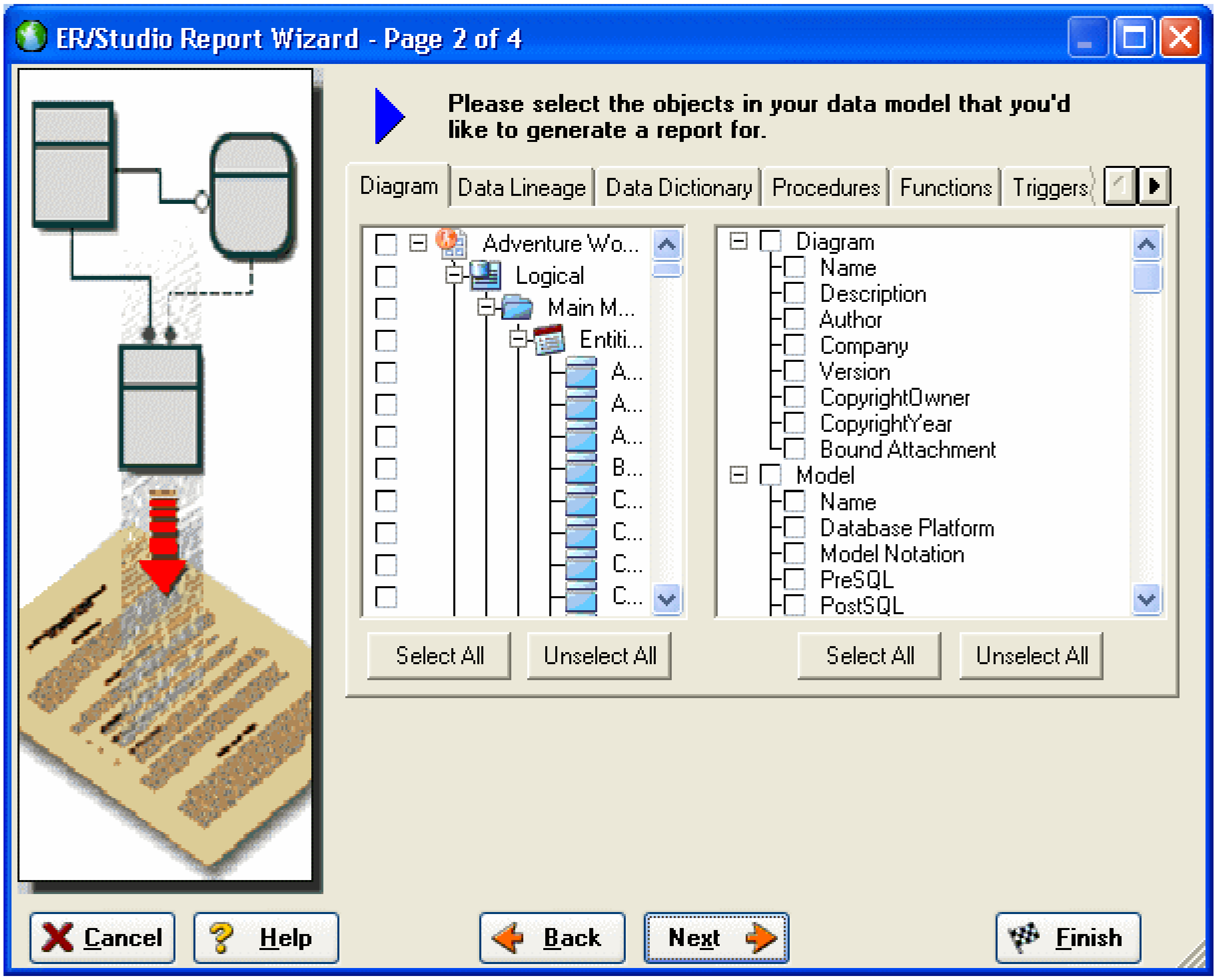
● Yes ○ No

Generate HTML report using performance-enhanced version for large data models.

□ Use enhanced version.

10 On the first page of the wizard, for the report type, select **HTML**.

11 On page 2, click **Select All** in both areas of the **Diagram** tab.

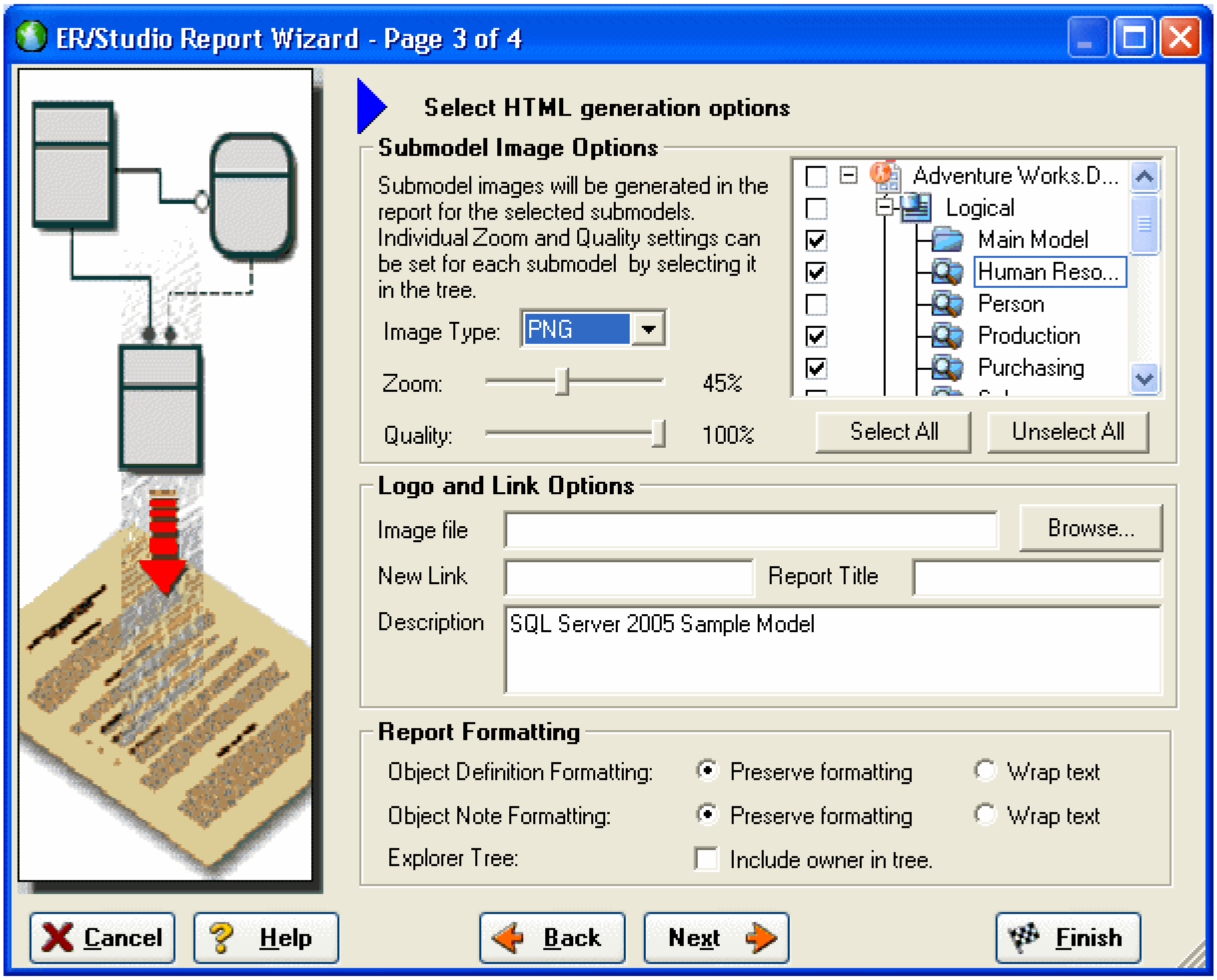


12 Click **Select All** in both areas of the **Data Dictionary** and the **Procedures** tabs, and then click **Next**.

NOTE: The tabs available on page 2 depend on what objects are supported by the database platform of the selected model. Some databases support more database objects than Microsoft SQL Server 2005, such as Oracle11g and IBM DB2 LUW 9.x for which there are also tabs on this page for procedures, functions, triggers, packages, and tablespaces. If the model was previously denormalized, a **Denormalization Mappings** tab would also appear.

13 On page 3, in the **Submodel Image Options** area, click **Select All**.

TIP: In the **Logo and Link Options**, you can choose to replace the ER/Studio DA default Embarcadero Technologies logo in favor of your own corporate logo (and Hyperlink).



ER/Studio Report Wizard - Page 3 of 4

Select HTML generation options

Submodel Image Options

Submodel images will be generated in the report for the selected submodels. Individual Zoom and Quality settings can be set for each submodel by selecting it in the tree.

Image Type: PNG

Zoom: 45%

Quality: 100%

- □ Adventure Works.D...
- □ Logical
- ☑ Main Model
- ☑ Human Reso...
- □ Person
- ☑ Production
- ☑ Purchasing

Logo and Link Options

Image file:

New Link:

Report Title:

Description: SQL Server 2005 Sample Model

Report Formatting

Object Definition Formatting: ● Preserve formatting ○ Wrap text

Object Note Formatting: ● Preserve formatting ○ Wrap text

Explorer Tree: □ Include owner in tree.

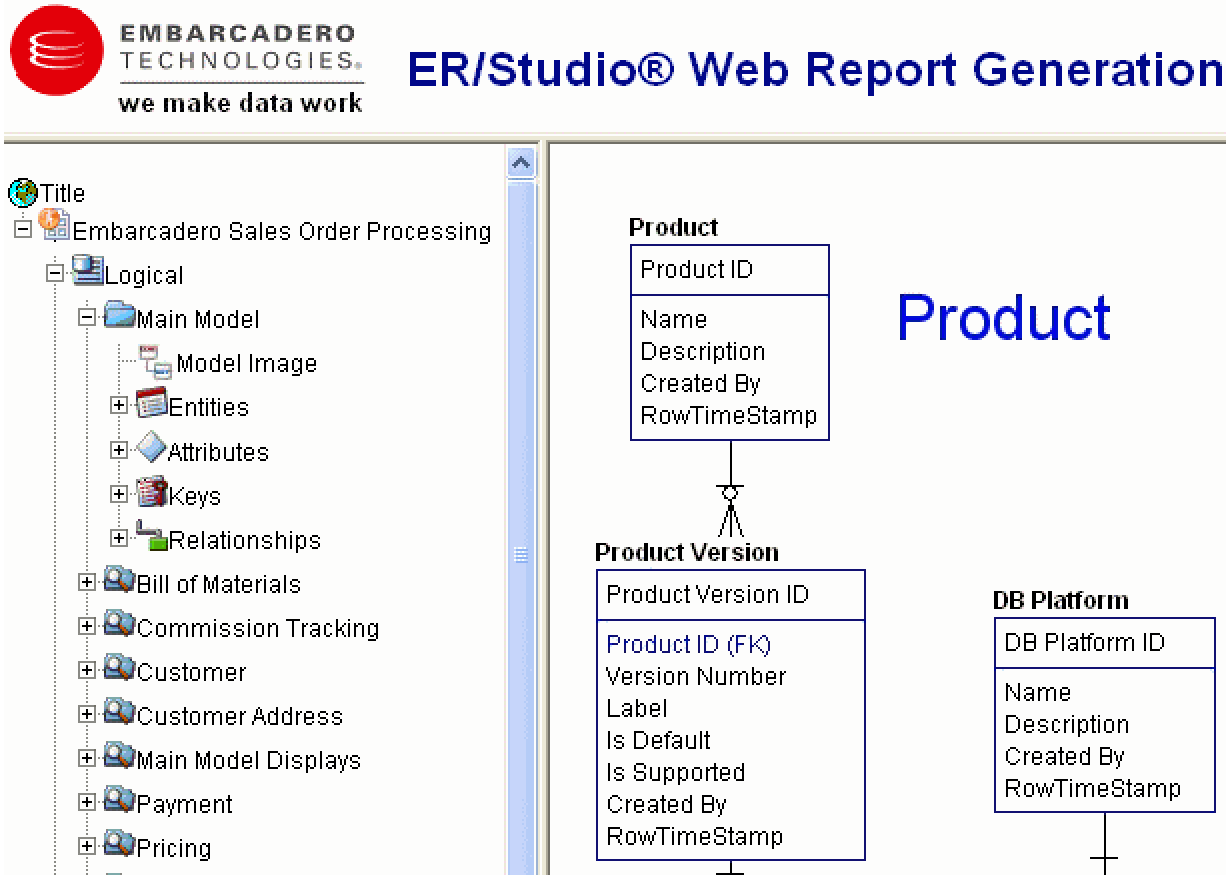
Because HTML formatting can be included in object definitions, you can also choose to preserve the formatting specified on the **Definitions** tab of the various object editors.

14 Click **Next** to advance to Page 4 of 4, and then click **Finish**.

ER/Studio DA then begins the report publication process and launches the default browser so you can review the report.

15 Finished!

Start navigating the report via your browser. Navigation will perform exactly as it does when you are using ER/Studio DA! Expand the tree to find Model Image and click on it (see below). You will see a read-only version of your data model (as seen below). Use the Explorer to navigate to any metadata you want or select the entities and relationships in the model image to jump to their information.



Conclusion

In this session, you learned how to:

- Connect to and reverse-engineer an existing database with ER/Studio DA.
- Document a database in seconds by using ER/Studio DA’s automatic HTML documentation publication facility.

For more information on Reporting, review the “Generating RTF and HTML Model Reports” in the *ER/Studio Data Architect User Guide*.

DOCUMENTING DATA LINEAGE

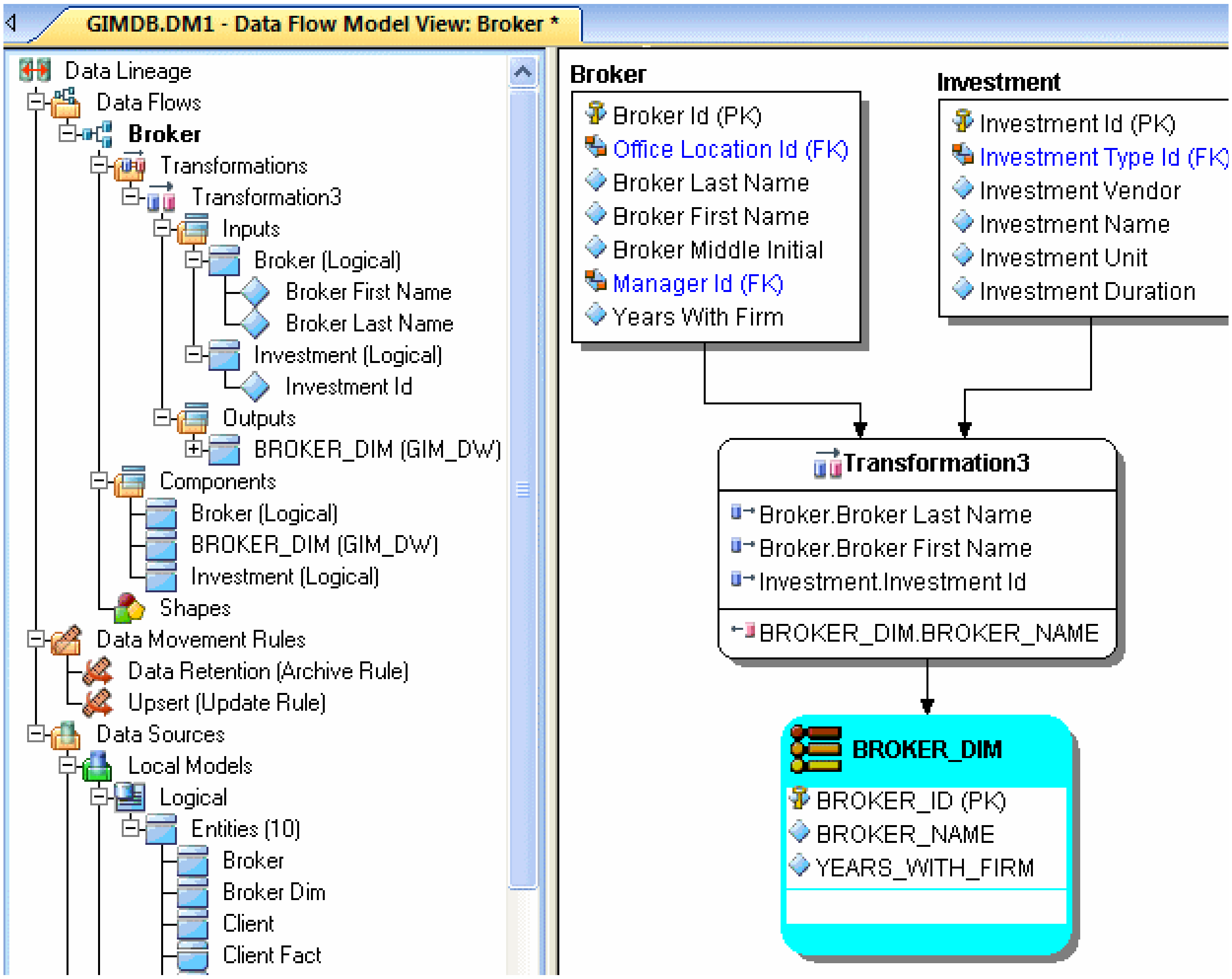
The Data Lineage feature of ER/Studio DA enables you to document the movement of data from point A to point B (and any intermediate steps in between). This movement is sometimes referred to as Extraction, Transformation and Load (ETL). Points A and B can be anything from flat files, high-end databases such as Oracle and DB2, XML, Access databases, and Excel worksheets. This is sometimes referred to as source and target mapping. A model produced in ERStudio can represent any point along the way. Data Architects need the ability to specify the *source* or *target* of data down to the column-level. Along with the metadata that defines the source and target mapping are rules for how the data is manipulated along the way.

This section will help you document the data lineage of your systems. It is comprised of the following tasks which correspond to the general ETL workflow:

- Creating a Data Flow on page 48
- Creating a Data Movement Rule on page 49
- Defining External Source and Target Systems on page 51
- Creating a Data Lineage and Transformation Visualization on page 53

Creating a Data Flow

The Data Flow organizes and encapsulates one data transformation and the source tables and columns used in the transformation to produce the target data. Multi-tiered mappings are possible and there can be multiple transformations involving different columns between two tables as illustrated below.

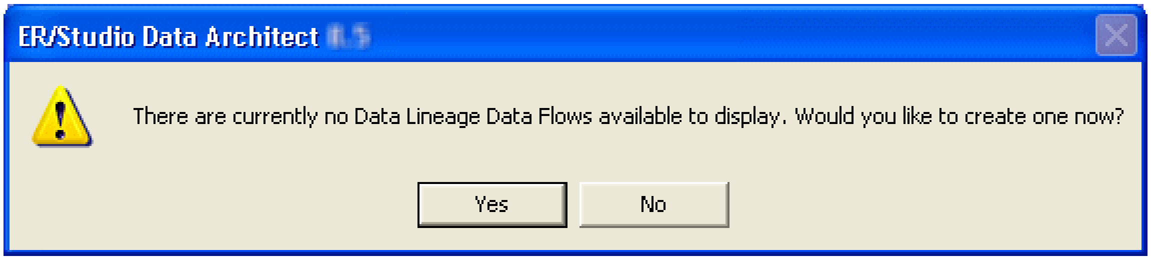


Create a Data Lineage Data Flow

1 Choose **File > Open** and select the `GIMB.DM1` diagram in the `Sample Models` directory.

2 Click the **Data Lineage** tab at the bottom of the application window.

You are prompted to create a **Data Lineage Data Flow**.

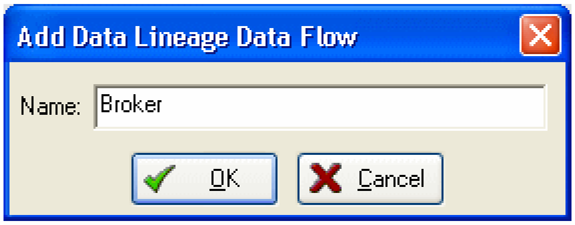


There are currently no Data Lineage Data Flows available to display. Would you like to create one now?

3 Click **Yes**.

If this is not the first time you click the Data Lineage tab after opening a diagram, from the **Data Lineage Explorer,** right-click the **Data Flows** node and then click **Create Data Flow**.

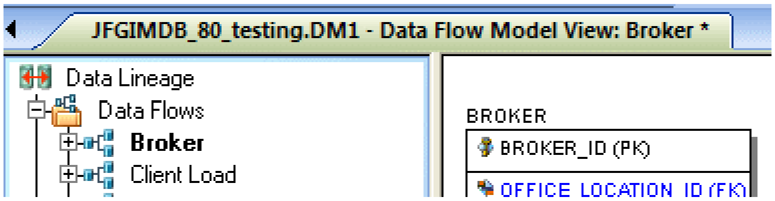
4 Enter a data lineage data flow name and then click **OK**.



Add Data Lineage Data Flow

Name: Broker

NOTE: The name that appears in the diagram title tab at the top of the application window is appended with: data flow name, when you click a task in the **Data Lineage** explorer, such as `GIMDB.DM1 - Data Flow Model View: Broker*`.



JFGIMDB_80_testing.DM1 - Data Flow Model View: Broker *

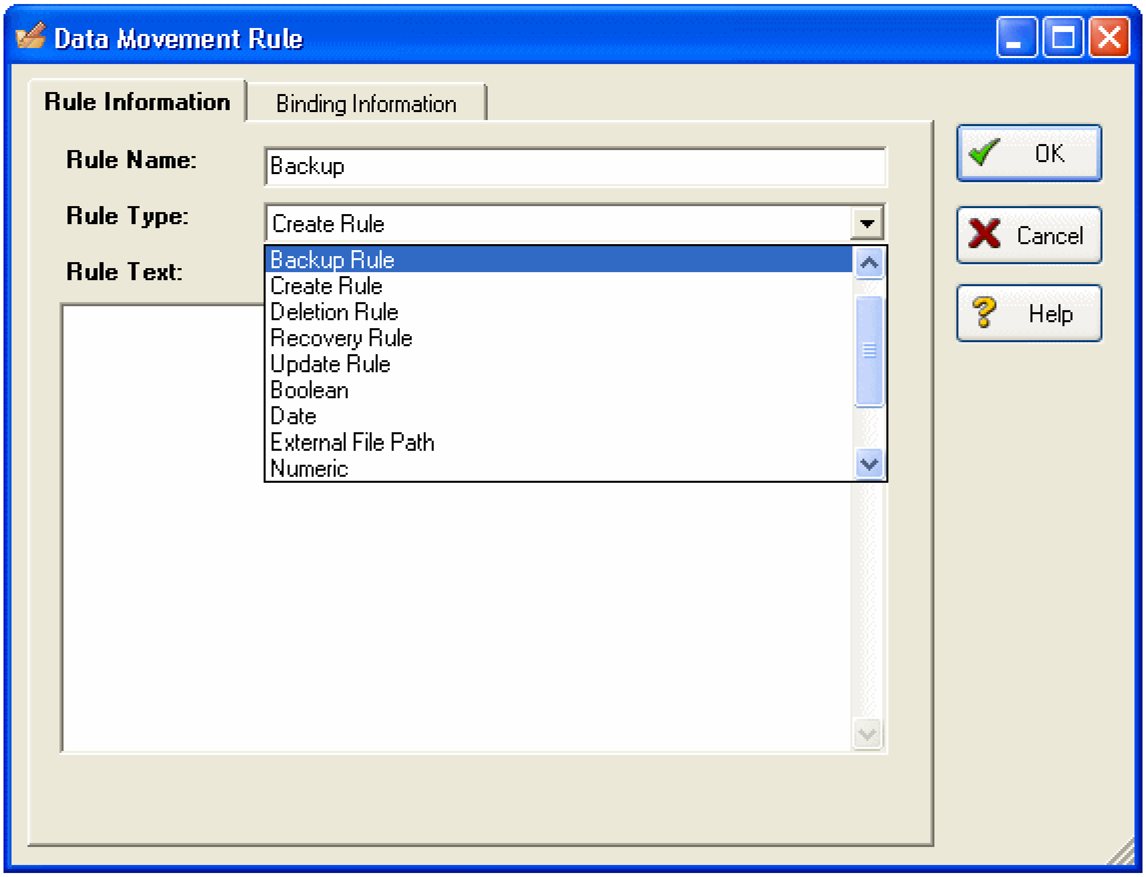
The Data Flow has been created.

Creating a Data Movement Rule

Data Movement rules describe how source and target tables and entities are related. You can relate source data to one or more tables and entities in the same model, the active diagram, or to tables imported from external systems. The rules defined here are used at the table level on the Data Lineage tab of the entity and table editors.

Create a data movement rule

1 On the **Data Lineage** tab, right-click **Data Movement Rules** and choose **New Data Movement Rule**.



Rule Name: Backup

Rule Type: Create Rule

Rule Text:

2 Complete the **Data Movement Rule** editor as required and then click **OK** to exit the editor.

TIP: Once created, you can edit the Data Movement rule by double-clicking it to launch the **Data Movement Rule** editor.

The following describes options that require additional explanation:

Rule Information tab

- **Rule Name**: Enter a name that indicates the operation and objects acted on, depending on the specifics of your binding definition.
- **Rule Type**: Select a generic movement rule type that best describes the data movement.
- **Rule Text**: Document your data movement plan here, perhaps adding instructions or contingency plans.

Binding Information tab

Select the object classes and/or specific objects to which you want to bind this attachment. You can override this setting using the **Data Lineage** tab of the entity or table editor.

Defining External Source and Target Systems

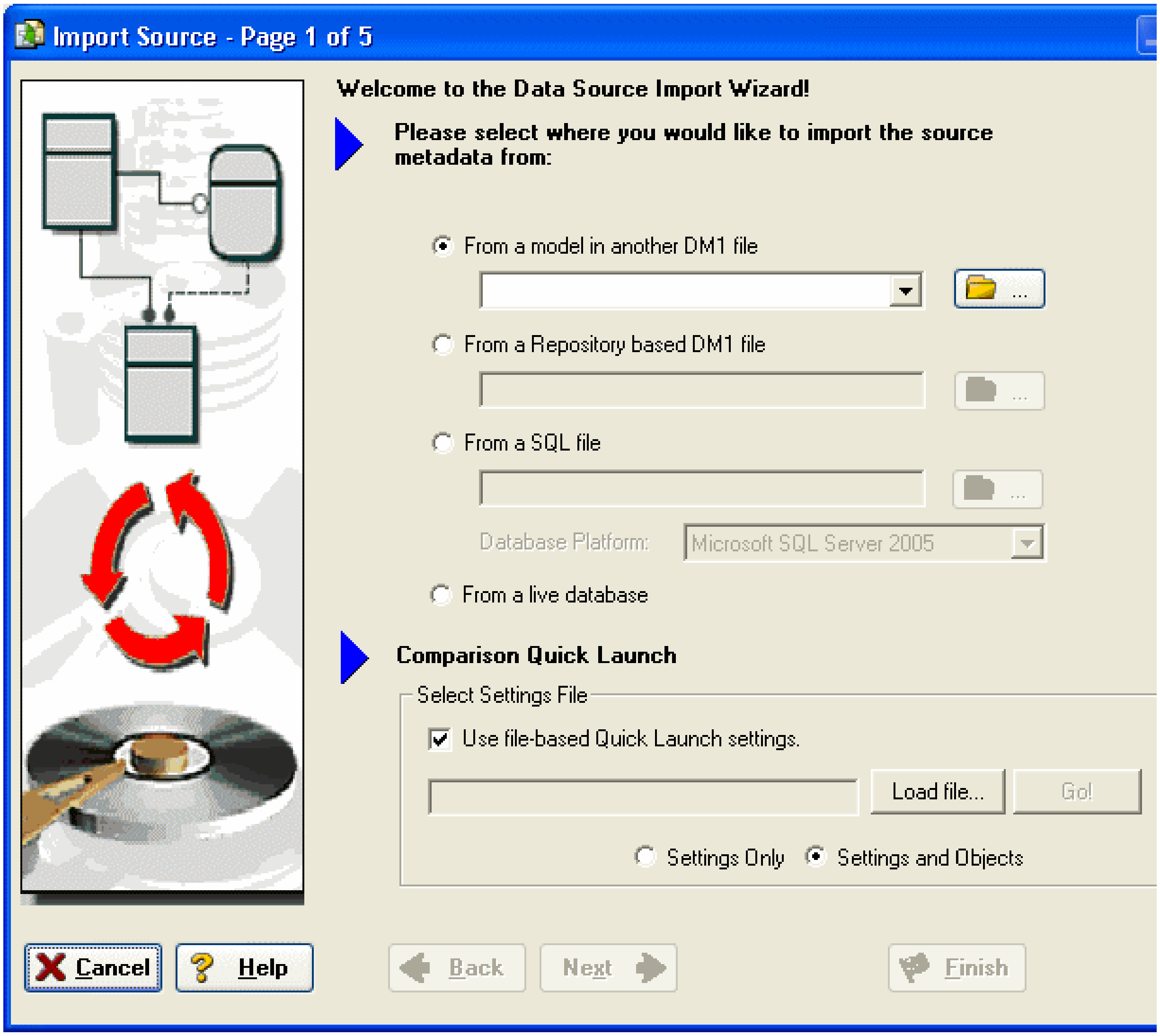
Data sources can originate from models in the active diagram (local models) or from external sources that are either imported into the active diagram or created on the Data Lineage tab. A data source can be imported from `*.dm1` files, `*.dt1` files, database or from SQL files, flat files, and other common application files. The following describes how to import metadata from an external source.

NOTE: Source data imported through the Data Lineage tab only includes information such as table and column name, datatype, nillability, primary key, and column definitions. To obtain more details, reverse engineer the database or import it into ER/Studio DA using the Metadata Wizard.

Import external source or target data

1 From the **Data Lineage** tab, expand the **Data Sources** node.

2 Right-click **Other Sources** and choose **Import New Source**.



Import Source - Page 1 of 5

Welcome to the Data Source Import Wizard!

Please select where you would like to import the source metadata from:

● From a model in another DM1 file

○ From a Repository based DM1 file

○ From a SQL file

Database Platform: Microsoft SQL Server 2005

○ From a live database

Comparison Quick Launch

Select Settings File

☑ Use file-based Quick Launch settings.

○ Settings Only ● Settings and Objects

3 Complete the **Import Source** wizard as required and then click **Finish** to import the source.

The new source will appear under the **Other Sources** node.

The following describe options that require additional explanation:

Page 1 - Please select where you would like to import the source metadata from

- **From a Repository based DM1 file**: Lets you obtain source from data models and Named Releases managed within the ER/Studio Repository. When you select this option, ER/Studio DA opens the Repository Operation Status dialog box and the **Get From Repository** dialog box. This process connects to the current Repository Server defined in the Repository settings. The **Import Source** wizard automatically gets the diagram.
- **From an SQL file** ER/Studio DA imports the SQL file.
- **From a live database:** If you select this option, a page appears where you can select the database and connection type. The connection type can be either ODBC or Native/Direct Connection. For information about connecting to databases, including troubleshooting information, see “Connecting to Database Sources and Targets” in the *ER/Studio Data Architect User Guide*.
- **Comparison Quick Launch**: The Compare Quick Launch data is saved as an *.rvo file. For information on using the **Quick Launch** option in the wizard, see “Saving and Using Quick Launch Settings” in the *ER/Studio Data Architect User Guide*.

Page 5 - Results

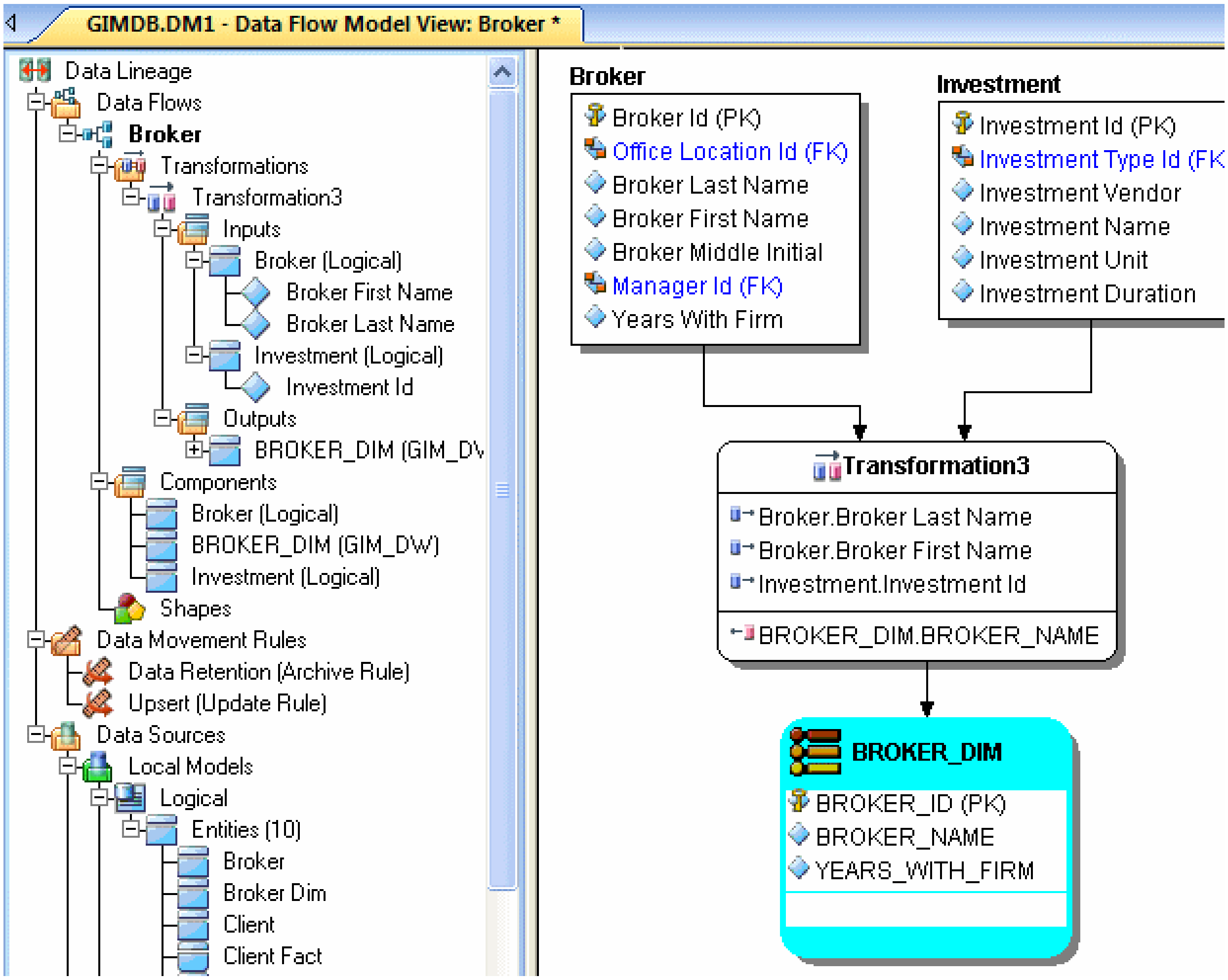
- **Current and Target Model Display Grid**: Between the Source and Target models is a Resolution column. The default merge decision is Merge the data into the new source file. You can click on any item in the **Resolution** column to enable the decision list. If you want to change the decision, click the list and then click the new resolution. When you change the default resolution of an object, the decisions of their dependent properties and objects are automatically updated. You can also click the category folders, like the **Tables Resolution** column to change all the decisions for all the underlying objects in that object category. And, you can use the **CTRL** key to select multiple items, and then right click to enable the decision list.
- **SQL Difference**: To enable the SQL Difference utility, select any difference that is a long text field, such as a Definition, Note, or DDL, and then click SQL Difference to view the differences between the SQL of the models. This utility only allows you to view the differences; difference resolutions are performed on the **Results** page of the **Compare and Merge Utility**.
- **Filter Report on Results:** Create a report of the source content and your chosen resolutions. You can choose to create an HTML or an RTF report.

TIP: You can modify the default display using the options at the bottom of the page.

Creating a Data Lineage and Transformation Visualization

1 To create the data source or transformation input tables, expand **Data Sources > Local Models > Logical > Entities** and then drag and drop the **Broker** and **Investment** tables onto the Data Lineage window.

2 To create the data target or transformation output tables, navigate to **Data Sources > Local Models > GIM_DW** and then drag and drop the **Broker** table onto the Data Lineage window.



3 To obtain the Transformation Insertion tool, right-click an empty space in the **Data Lineage** window and then click **Insert Transformation**.

4 To insert the transformation, click in the **Data Lineage** window between the source and target data sources and then right-click to drop the **Transformation Insertion** tool.

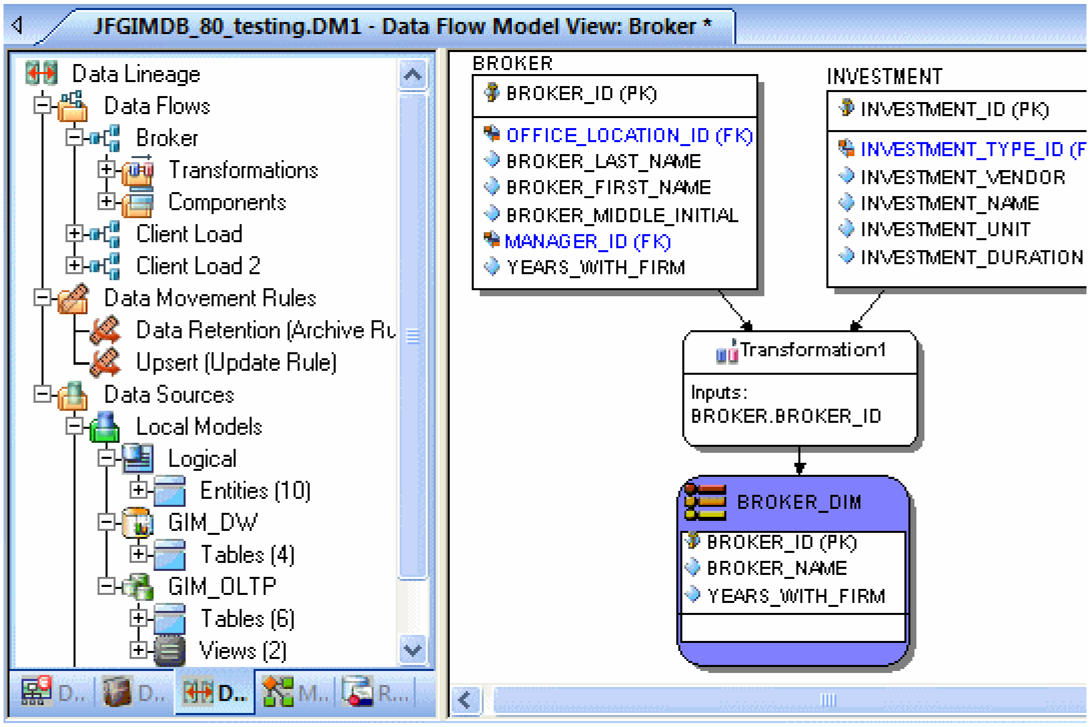
5 Reposition and resize the transformation object to suit your needs.

6 Right-click an empty space of the **Data Lineage** window and then click **Insert Data Stream**.

TIP: Transformation and Data Flow tools are also available on the toolbar. Hover the mouse over the tools to find the tool you need.

7 Click an input and then click the transformation object. Repeat as many times as necessary to link all the inputs to the transformation object.

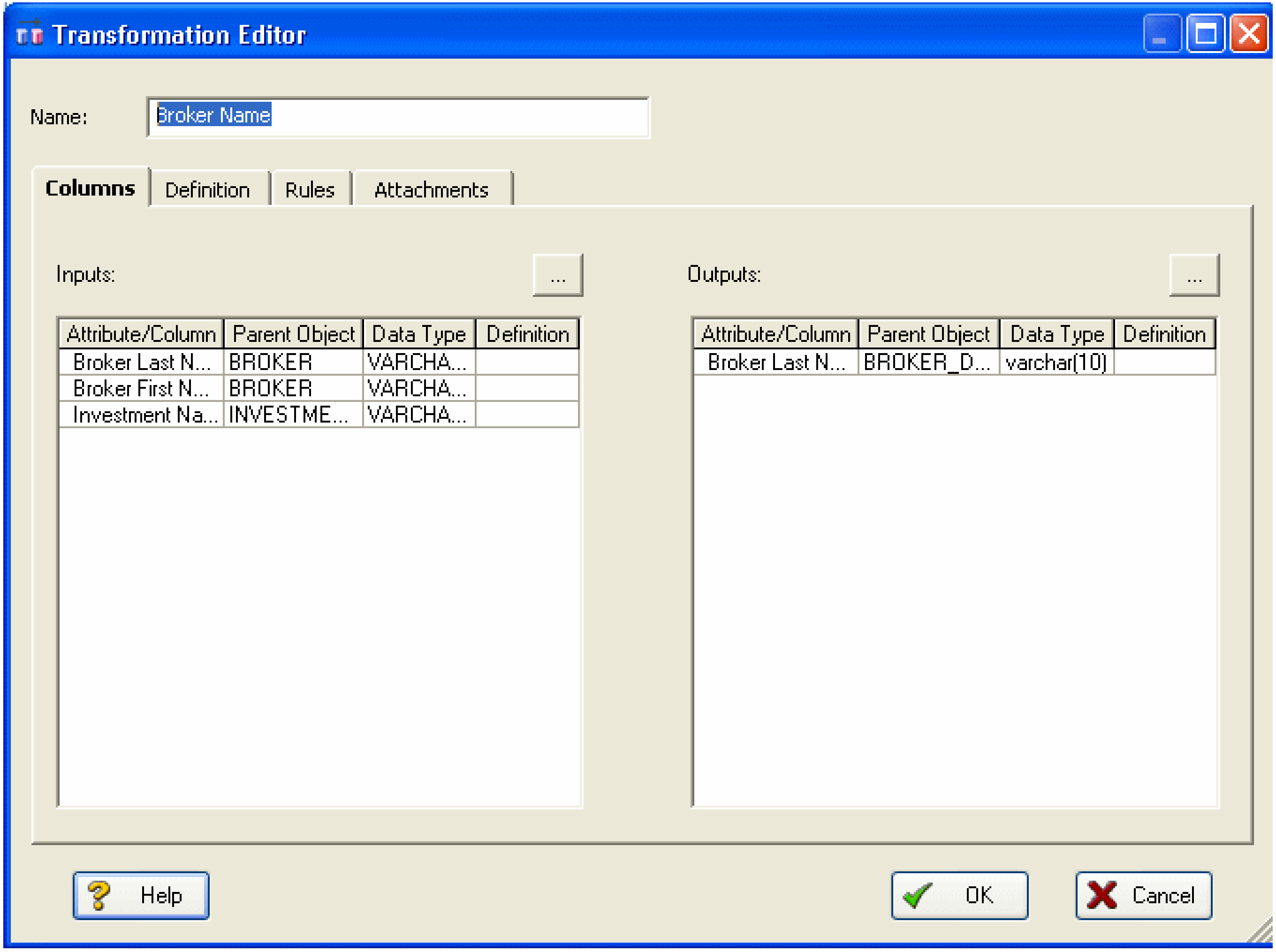
8 Click the transformation object and then click an output.



TIP: If the Inputs and Outputs do not display in the diagram as they do in the illustration above, choose **View > Diagram and Object Display Options > Transformation** and then click **Inputs and Outputs Columns**.

9 To define which columns should be used in the transformation and any transformation rules, double-click the new transformation to open the **Transformation Editor**.

10 Complete the **Transformation Editor** as required and then click **OK** to exit the editor.



Name: Broker Name

Inputs:

Attribute/Column	Parent Object	Data Type	Definition
Broker Last N...	BROKER	VARCHA...	
Broker First N...	BROKER	VARCHA...	
Investment Na...	INVESTME...	VARCHA...	

Outputs:

Attribute/Column	Parent Object	Data Type	Definition
Broker Last N...	BROKER_D...	varchar(10)	

You are done! Now you can more easily share your ideas with your colleagues!

TIP: Once the Data Flow is created, you can double-click it to change its name, or double click a transformation or component to change its properties.

The following describes options in the Transformation Editor that require additional explanation:

Columns tab

- **Inputs**: Click the ellipsis (...) button to choose the inputs to be transformed in this task.
- **Outputs**: Click the ellipsis (...) button to choose the outputs resulting from the transformation.

Definition tab

- **Business**: Describe the transformation for your audience.
- **Code**: Enter the code that will perform the transformation, such as a SELECT statement, or a VBBasic or Java Script function or procedure.

Data Movement Rules tab

These are the rules from the Data Movement Rules node of the Data Lineage Explorer.

NOTE: You can delete or edit an input or output column by double-clicking the transformation in the **Data Lineage** window, clicking the ellipsis in the **Transformation Editor** and then deselecting the column you want to remove.

Attachments tab

Bind an external piece of information or attachment to the transformation. You can also remove an attachment from an object, override the default value of an attachment, or change the position of a bound attachment. To override the value of the attachment you have moved to the **Selected Attachments** grid, double-click the **Value** field of the target attachment. ER/Studio DA opens the **Value Override Editor** or a list, depending on the attachment datatype. Attachments are created in the **Attachments** folder of the **Data Dictionary** and must be applied to the default before they will display on this tab.

Diagram Navigation and Aesthetics

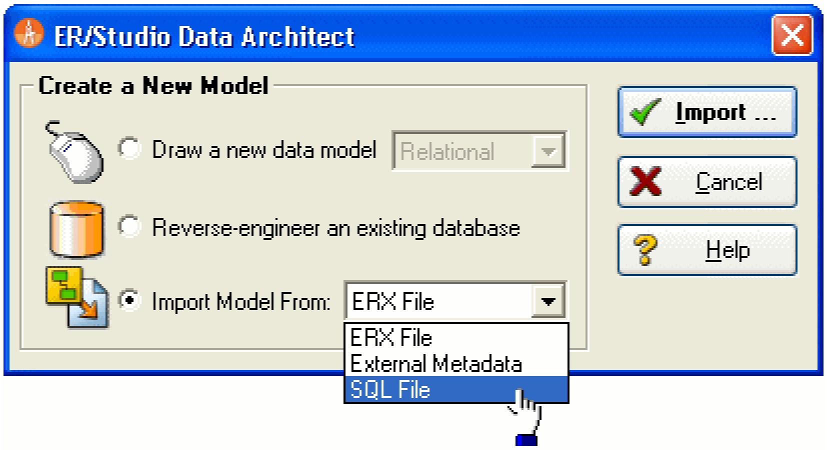
To assist with the creation of presentation-quality diagrams that are easy to navigate and are aesthetically pleasing, ER/Studio DA offers progressive diagram Auto Layout and Navigation utilities that also help you to clean up complex diagrams. Modelers should spend time solving complex database or business data model problems, not forcing boxes and lines to look a certain way.

Navigating the Diagram

To demonstrate some of ER/Studio DA's layout and navigation utilities, we will import a sample SQL script provided with ER/Studio DA.

1 Close any files you have open.

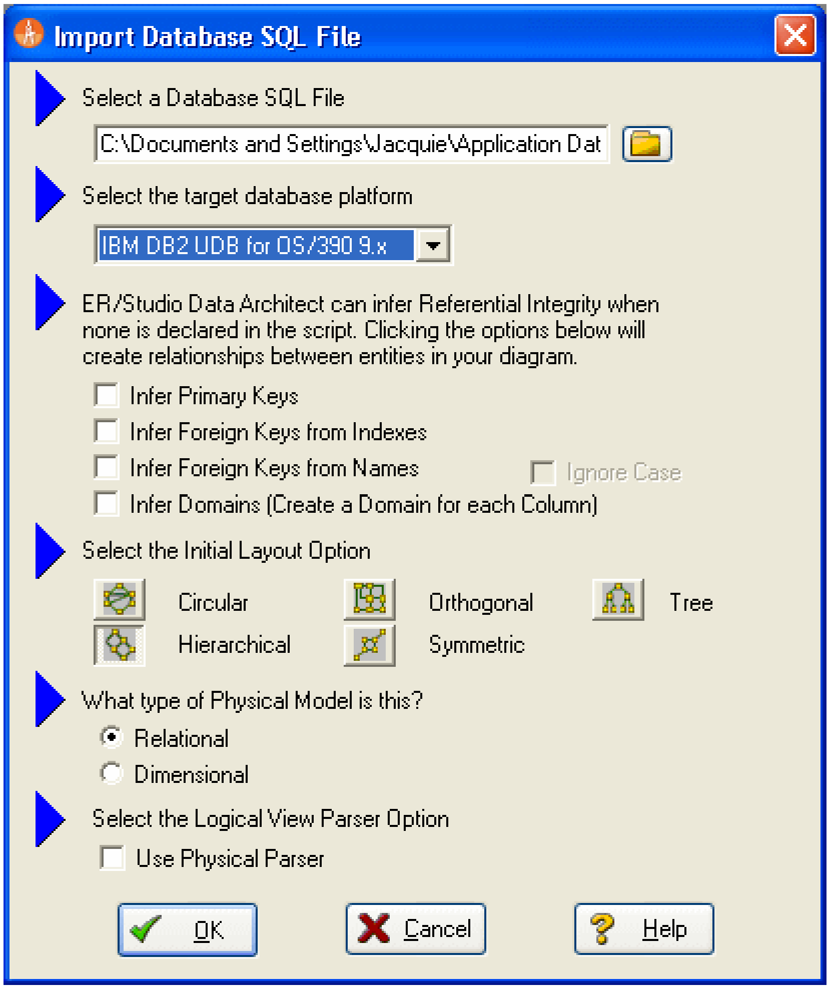
2 Choose **File > New**.



3 Select **Import Model From:** and then in the import list, click **SQL File**.

NOTE: The ERX File choice enables you to import Computer Associates ERwin 3.5.2 ERX files. Selecting **From External Metadata** launches the MetaWizard to import from alternative sources.

The **Import Database SQL File** dialog appears:



Import Database SQL File

Select a Database SQL File

C:\Documents and Settings\Jacquie\Application Dat

Select the target database platform

IBM DB2 UDB for OS/390 9.x

ER/Studio Data Architect can infer Referential Integrity when none is declared in the script. Clicking the options below will create relationships between entities in your diagram.

□ Infer Primary Keys

□ Infer Foreign Keys from Indexes

□ Infer Foreign Keys from Names □ Ignore Case

□ Infer Domains (Create a Domain for each Column)

Select the Initial Layout Option

Circular Orthogonal Tree

Hierarchical Symmetric

What type of Physical Model is this?

● Relational

○ Dimensional

Select the Logical View Parser Option

□ Use Physical Parser

4 To the right of **Select a Database SQL File** click the folder icon, click **IBM DB2 OS390.SQL**, and then click **Open**.

The full path to this file is:

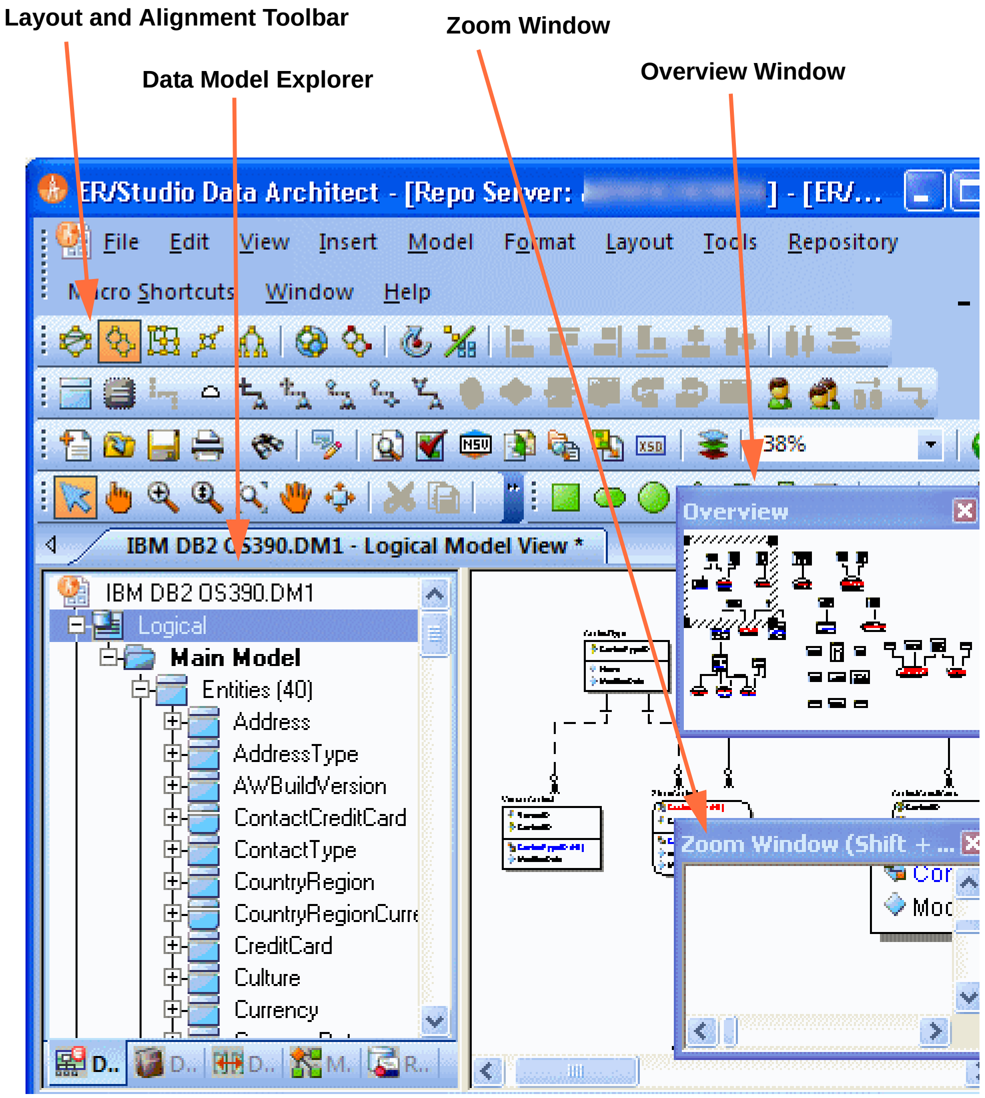
For Windows XP:
`C:\Documents and Settings\`*`<user>`*`\Application Data\Embarcadero\ERStudio\SQLCode`

For Windows Vista:
`C:\Users\`*`<user>`*`\AppData\Roaming\Embarcadero\ERStudio\SQLCodeSample`

5 In the **Select the target database platform** list, click **IBM DB2 UDB for OS /390 9.x**.

6 Click **OK**.

Finished! Once the SQL Script is finished importing (as depicted below) the following items will assist you in leveraging a variety of Auto Layout and Navigation Features.



- **Layout and Alignment Toolbar**: Use any of the four Auto Layout styles to change the layout of the diagram with the click of a button. The auto layout styles are all entirely customizable styles. You can also customize the diagram layout via the Layout Properties pages that can be launched by clicking **Layout > Layout Properties**.
- **Data Model Explorer**: Click on any object in the **Data Model** explorer and it will automatically be selected in the diagram and focused in both the **Zoom Window** and **Overview** window.
- **Overview Window**: Use this window as a thumbnail of your model to pan the entire model or zoom in and out. It can also pan and zoom the diagram if grabbed or sized. If the **Overview** window is not already visible, press **F9** to activate it.

- **Zoom Window**: Use this window as a magnifying glass to enlarge any diagram objects under your mouse cursor. You can also press **SHIFT+F8** to freeze the zoom window to keep a single object frozen while you continue to pan around the diagram. If the **Zoom** window is not already visible, press **F8** to activate it.

Diagram Aesthetics

One of the tremendous benefits of building data models is the wide range of audiences that can realize value from them. Part of this relies on what information is displayed on the diagram. Depending on the audience you may want to limit or expand what is displayed. For example, developers may benefit from looking at a model that displays data type, null option, and unique and non-unique index information, while business analysts may just need the entity name and the definition. ER/Studio DA offers many display properties that can be customized exactly for this purpose.

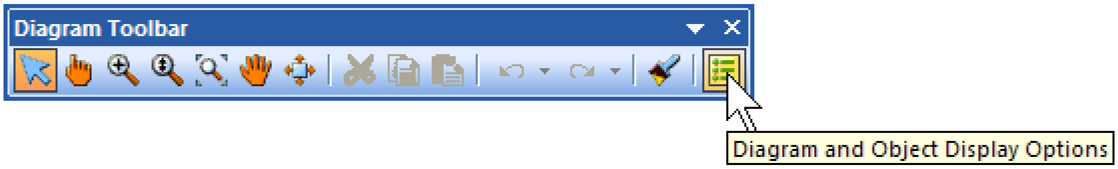
Continuing with the previous section, we will use the DB2 model that was built to demonstrate some of the ways to customize the appearance of the model.

We will use the Diagram and Object Display Options dialog on the Diagram toolbar to further customize the view of the logical and physical models.

NOTE: You can use the Colors & Fonts tool to customize the look and feel further of each model.

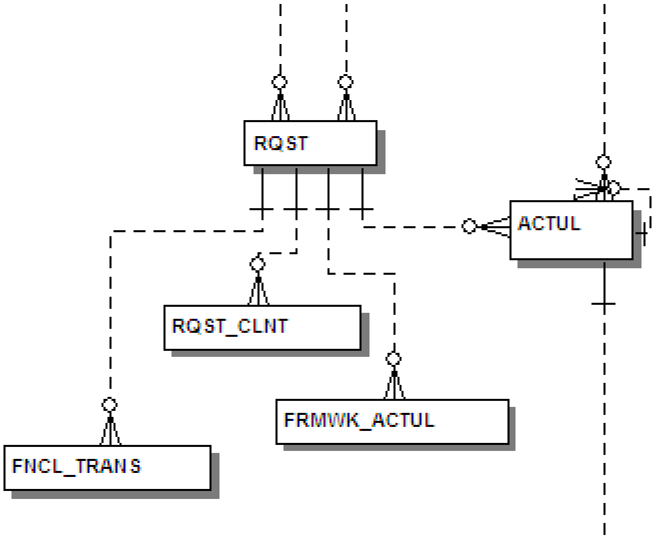
Setting the Logical Model Display

1 Select the logical model and then on the **Diagram Toolbar**, click the **Diagram and Object Display Options** tool.



2 In the **Diagram and Object Display Options** dialog, click the **Entity** tab, and then in the **Display Level** area, select **Entity**.

3 Click **OK**.



NOTE: Only entity names are displayed for each entity. You may also want to re-layout the diagram since the entity sizes have changed.

Setting the Physical Model Display

1 Select the physical model and then choose **View > Diagram and Object Display Options**.

2 In the **Diagram And Object Display Options** dialog, click the **Table** tab.

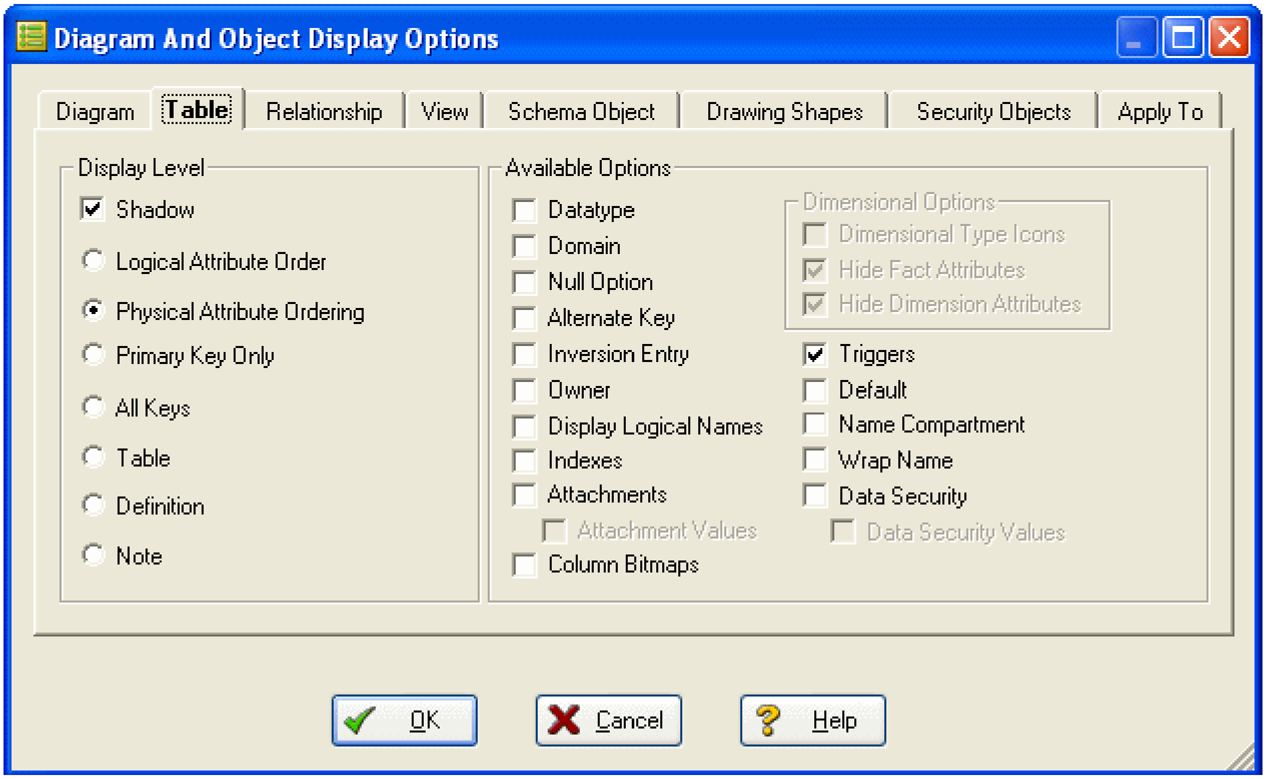


Diagram And Object Display Options

Tabs: Diagram | **Table** | Relationship | View | Schema Object | Drawing Shapes | Security Objects | Apply To

Display Level
- ☑ Shadow
- ○ Logical Attribute Order
- ● Physical Attribute Ordering
- ○ Primary Key Only
- ○ All Keys
- ○ Table
- ○ Definition
- ○ Note

Available Options
- □ Datatype
- □ Domain
- □ Null Option
- □ Alternate Key
- □ Inversion Entry
- □ Owner
- □ Display Logical Names
- □ Indexes
- □ Attachments
  - □ Attachment Values
- □ Column Bitmaps

Dimensional Options
- □ Dimensional Type Icons
- ☑ Hide Fact Attributes
- ☑ Hide Dimension Attributes

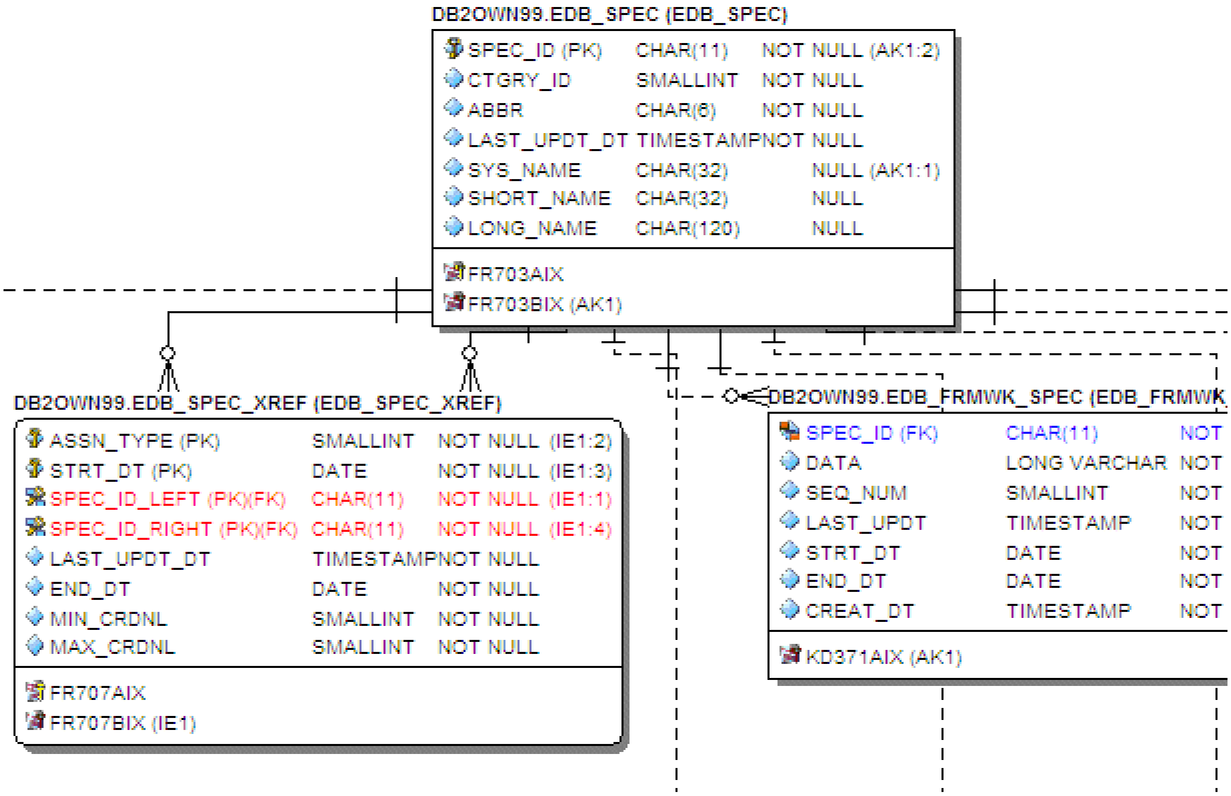
- ☑ Triggers
- □ Default
- □ Name Compartment
- □ Wrap Name
- □ Data Security
  - □ Data Security Values

3 In the **Display Level** area, select **Physical Attribute Ordering**.

4 In the **Available Options** area, select the specific properties you want to display.

5 Click **OK**.

The model should now display more details for the physical model, as seen below.



DB2OWN99.EDB_SPEC (EDB_SPEC)

SPEC_ID (PK)	CHAR(11)	NOT NULL (AK1:2)
CTGRY_ID	SMALLINT	NOT NULL
ABBR	CHAR(6)	NOT NULL
LAST_UPDT_DT	TIMESTAMPNOT NULL	
SYS_NAME	CHAR(32)	NULL (AK1:1)
SHORT_NAME	CHAR(32)	NULL
LONG_NAME	CHAR(120)	NULL
FR703AIX		
FR703BIX (AK1)		

DB2OWN99.EDB_SPEC_XREF (EDB_SPEC_XREF)

ASSN_TYPE (PK)	SMALLINT	NOT NULL (IE1:2)
STRT_DT (PK)	DATE	NOT NULL (IE1:3)
SPEC_ID_LEFT (PK)(FK)	CHAR(11)	NOT NULL (IE1:1)
SPEC_ID_RIGHT (PK)(FK)	CHAR(11)	NOT NULL (IE1:4)
LAST_UPDT_DT	TIMESTAMPNOT NULL	
END_DT	DATE	NOT NULL
MIN_CRDNL	SMALLINT	NOT NULL
MAX_CRDNL	SMALLINT	NOT NULL
FR707AIX		
FR707BIX (IE1)		

DB2OWN99.EDB_FRMWK_SPEC (EDB_FRMWK

SPEC_ID (FK)	CHAR(11)	NOT
DATA	LONG VARCHAR	NOT
SEQ_NUM	SMALLINT	NOT
LAST_UPDT	TIMESTAMP	NOT
STRT_DT	DATE	NOT
END_DT	DATE	NOT
CREAT_DT	TIMESTAMP	NOT
KD371AIX (AK1)		

NOTE: Since the sizes of the objects changed, you may want to change the model layout using one of the ER/Studio DA advanced layout engines. You can also customize the default display properties for new models by choosing **Tools > Options**, and then selecting the desired options on the **Display** tab.

Conclusion

In this session, you have learned how to:

- Import an SQL file and allow ER/Studio DA to automatically create a diagram.
- Use a variety of auto layout and navigation tools to enhance the aesthetic experience of the diagram and to improve the data model navigability.
- Customize the display of both the logical and physical models.