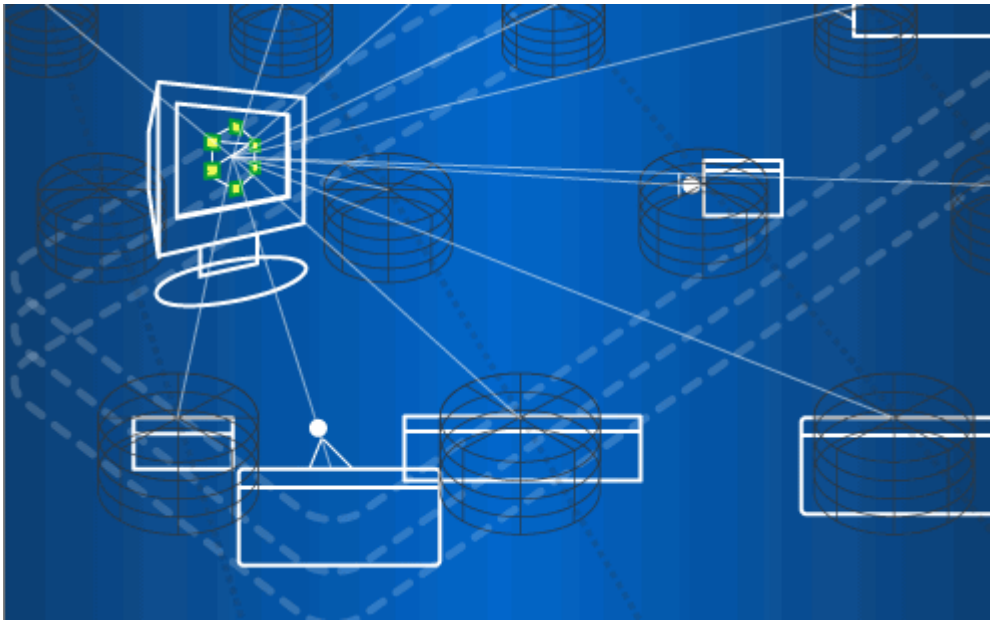
EMBARCADERO TECHNOLOGIES®

ER/Studio® Quick-Start Guide

A guide to getting started with Embarcadero's data modeling solution, ER/Studio, and its team-oriented modeling collaboration server (Enterprise Edition)



Embarcadero Technologies, Inc.
100 California Street, 12th floor

San Francisco, CA 94111
Tel. 415.834.3131 Fax 415.434.1721
www.embarcadero.com

Embarcadero Technologies, Inc.
100 California Street, 12th Floor
San Francisco, CA 94111

Contents

Introduction to Embarcadero's ER/Studio

ER/Studio 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 meta data across the enterprise.

ER/Studio's progressive interface and the simplicity of its understanding out of the box 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 the user to create, understand and manage the lifecycle of mission-critical database designs and business metadata within the enterprise.

The product offers strong logical design capabilities, the ability to spawn many physical designs from a corporate logical design, bi-directional model comparison and synchronization of information, rich and customizable, Visual-Basic for Applications API for product customization, powerful DDL reverse engineering and generation, metadata import and export capabilities and sophisticated HTML and RTF-based documentation and reporting facilities.

Product Benefits by Audience

Data Modelers and Data Architects

ER/Studio is critical for organizations concerned with **eliminating data redundancy**, creating an **enterprise view of data assets** and assisting development with making informed decisions about how best to **reuse elements** pre-defined by the enterprise. It's powerful logical (non-database or technology specific) analysis and design environment helps to normalize and create an enterprise view of the "objects of importance" an organization is concerned with managing data and more importantly can communicate this quickly through powerful reporting mechanisms to the enterprise.

Database Administrators and Database Developers

Managing databases can be incredibly difficult without a blue print or road map to understand important object dependencies. ER/Studio's round-trip engineering capabilities including database **reverse-engineering** provide database administrators (DBAs) or developers with important 'physical' data models in seconds. These models can be used as powerful and efficient **change management** platforms, allowing users to update a model for the purpose of required changes which need to be implemented at the database and **automatically generate** DBMS-specific, syntactically correct alteration or database DDL.

Business and IT Management

ER/Studio's robust reporting facilities allow delivery of critical information about designs to the enterprise in seconds. This heavily leveraged and beneficial capability of ER/Studio allows users to provide, in literally seconds, **clear, easily navigable and safe-to-distribute documentation** about a database or enterprise data model to those who need to review it.

About this Evaluation Guide

This evaluation guide is intended to help you get started using Embarcadero's data modeling and database design solution, ER/Studio and its modeling collaboration server repository.

After completion of this evaluation guide, you'll have the foundation you need to explore the many features and benefits of ER/Studio. You'll have learned how to create a new data model, work with logical and physical diagrams, leverage the productivity-focused features such as its powerful reporting engines and learn the basics of the modeling collaboration server repository. You'll also have become familiar with some frequently used tasks and commands to make you more productive.

This guide is divided into 7 easy to learn sessions. Do them all at once, or complete them individually as you have time.

Session 1: Getting Started with ER/Studio

Session 2: Logical and Physical Modeling in ER/Studio

Session 3: Documenting an Existing Database

Session 4: Diagram Aesthetics and Navigation

Session 5: Metadata Importation and Exportation

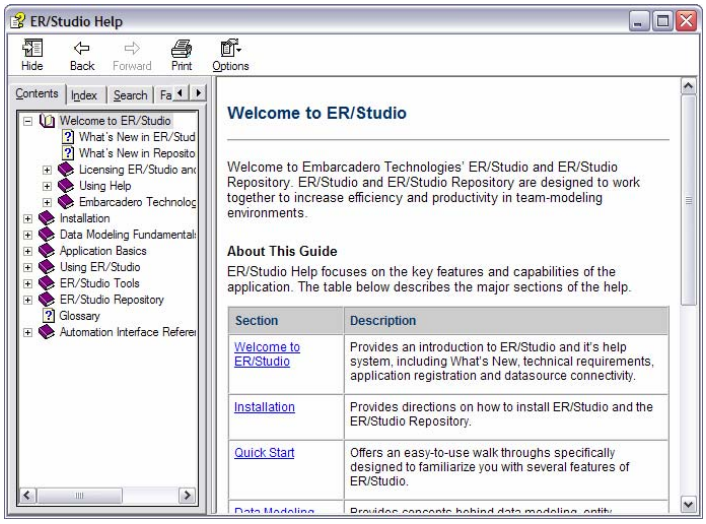
Session 6: Dimensional Modeling

Session 7: Automating Tasks

Session 8: Using the Modeling Collaboration Server (Enterprise Edition)

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

Once you've started, you can select **Help** from the toolbar to find many additional resources that complement and build on many of the activities shown in this brief guide.



Session 1: Getting Started with ER/Studio

Download and Install

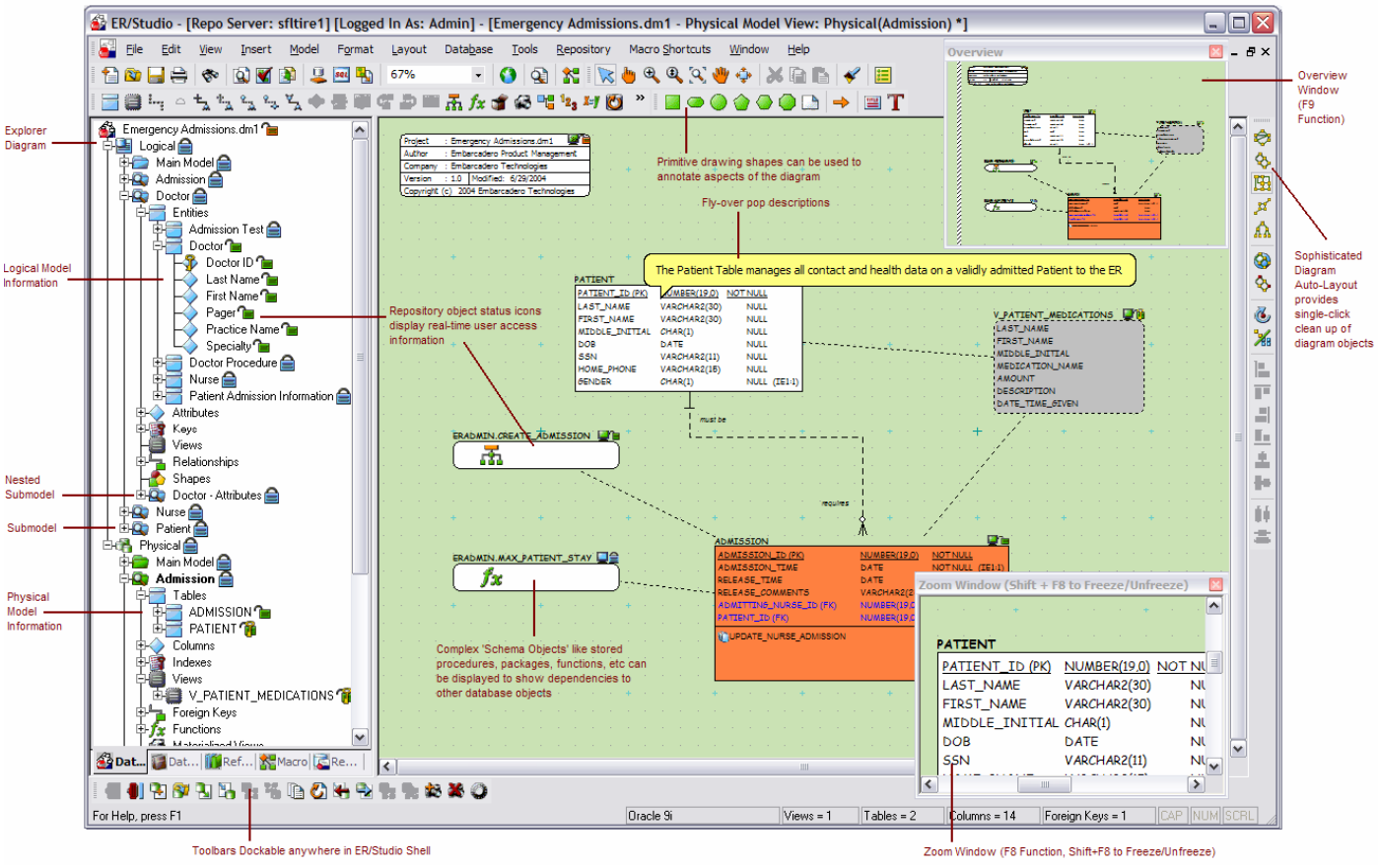
You can obtain the latest version of the ER/Studio Enterprise software from the Embarcadero website at http://www.embarcadero.com/downloads/downloaderstudio.jsp

Click "**Download**", and follow the steps indicated. Save the file on your computer and then double-click to launch the self-extracting file that will guide you through the installation process.

When you first install an evaluation copy of ER/Studio, you can use the tool for 14 days. After that time, a permanent license is needed and can be acquired by contacting Sales@Embarcadero.com or calling (415) 834 3131.

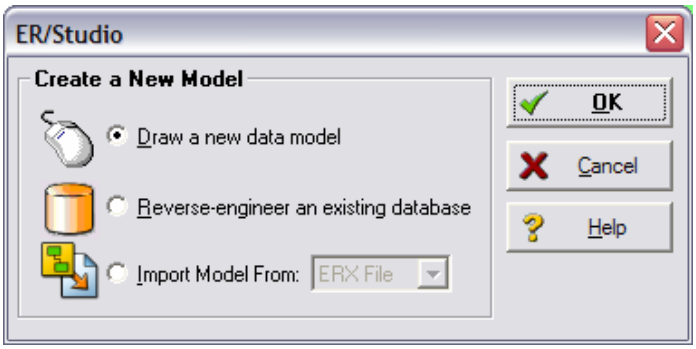
Overview

The graphic below illustrates all the elements of the ER/Studio Application User Interface



Starting to data model with ER/Studio

1. Choose Embarcadero ER/Studio from the **Start > Program** menu.
2. Select **File > New** (or CTRL + N)
3. Opt to select **Draw a new data model**
4. Click **OK**



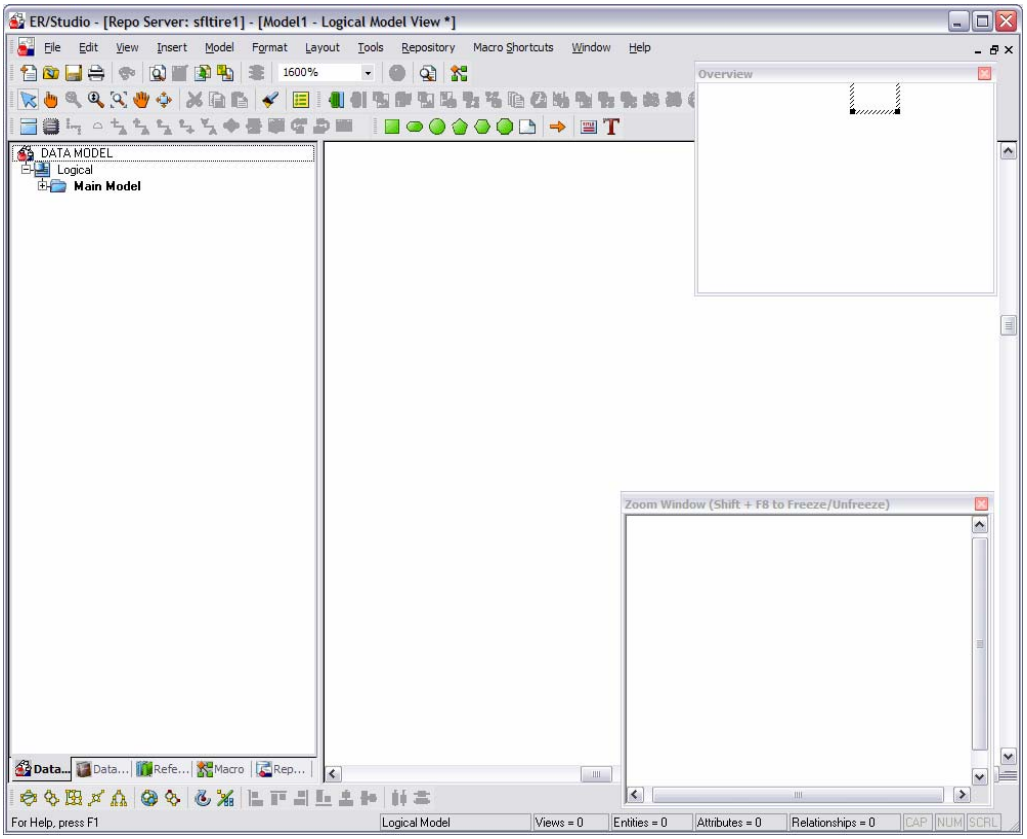
As you can see through the "Create a New Model" dialog, there are a number of ways to begin modeling with ER/Studio:

1) Building a new design "from the ground up" by "drawing a new data model", or…

2) Building a data model from an existing database through live reverse engineering as well as…

3) Importing designs from other modeling products such as Computer Associate's ERwin or SQL files.

As instructed above, begin with "Draw a new data model" to set us up for Session 2: Logical and Physical Modeling in ER/Studio. After selecting 'Draw a new data model" and clicking OK, ER/Studio will look like the image below:



Session 2: Logical and Physical Modeling with ER/Studio

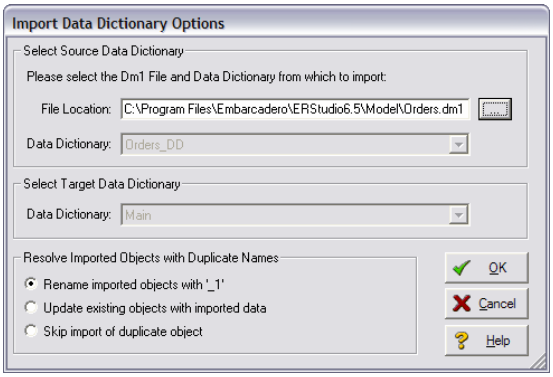
ER/Studio supports both Logical (non-DBMS or technology-specific) modeling with Physical (DBMS-specific) modeling. ER/Studio is designed to allow organizations the flexibility to analyze and design a business problem or application logically and generate as many different physical 'interpretations' from the logical model as the user wishes. Multiple physical models can be generated from the Logical for the same DBMS (say Oracle) or varied DBMSs (say Oracle, SQL Server and DB2). This will be discussed in detail in the following sessions.

Start a Logical Model from the Ground Up

As instructed in Session 1 above, you have chosen to 'Draw a new Data Model' to begin a Logical Model from the ground up. Before, we begin to add entities, let's populate ER/Studio with some sample re-useable Domains (these are re-usable attributes).

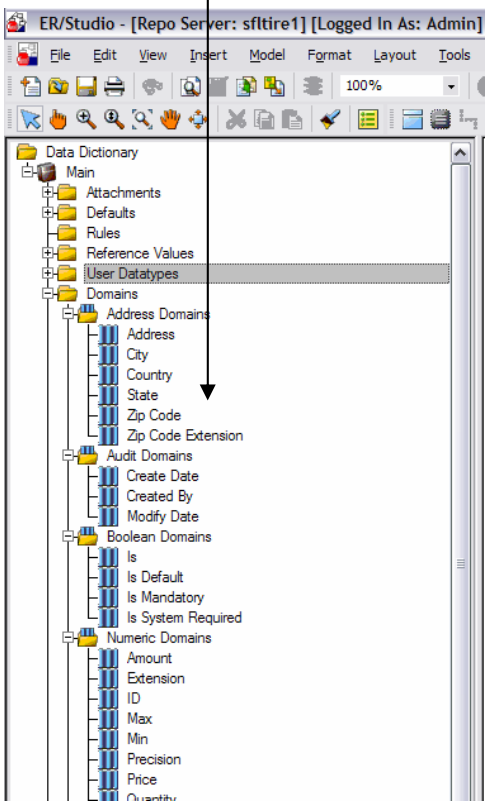
1. Select **File > Import Data Dictionary**.

2. Browse the "Orders.dm1" sample model in the ERStudio\Model folder. This model contains a pre-populated sample data dictionary. ***Note:*** You can choose between a couple of options to determine how the dictionary objects are imported. This is more important when importing into a diagram that already has dictionary objects in it.



3. Click **OK.**

4. Once opened, note that ER/Studio's Explorer Browser pane has automatically switched to its 'Data Dictionary' tab to allow immediate drag and drop access of the Domains:

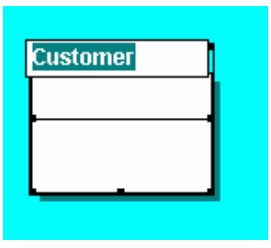


5. Now, add an entity to the Modeling workspace by clicking on the Entity icon on the Diagram Toolbar:

6. Drop the Entity on the workspace. (***TIP****! Your mouse icon will change to an entity symbol once the Entity icon (or any icon) is clicked so you may drop as many on the diagram workspace as you wish. Right mouse click to return your mouse to its original arrow cursor)*

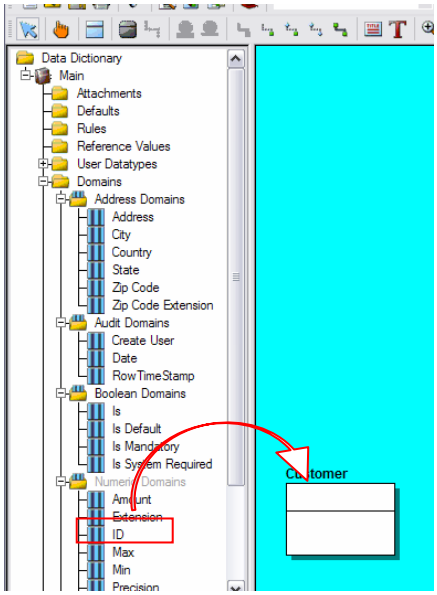
7. Type in an Entity Name ('Customer') to the new entity directly on-screen.

(TIP*! You can insert and edit/re-name an entity's name and attribute directly on screen by holding down the 'Shift' key and clicking on the attribute with your mouse. Do this to the ID Domain and change its name to "CustomerID". Use your 'tab' key to cycle between an entity's name, pk and non-pk fields and 'Return' to*



insert a new field.)

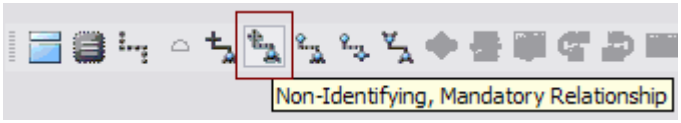
8. From the Data Dictionary, locate the "ID" Domain, click on it (do not release your mouse) and drag out to Customer entity and release it in the entity's Primary Key Field (the field below the entity's name) like you see in this picture:



9. Repeat this process by populating the Customer entity with the Name, Address, City, State Zip Code and Phone Domains from the Data Dictionary tree.

10. Drop another Entity on the Modeling workspace. Call this entity 'Order.' Feel free to populate it's Primary Key with an ID which you can change to "OrderID" via the steps above.

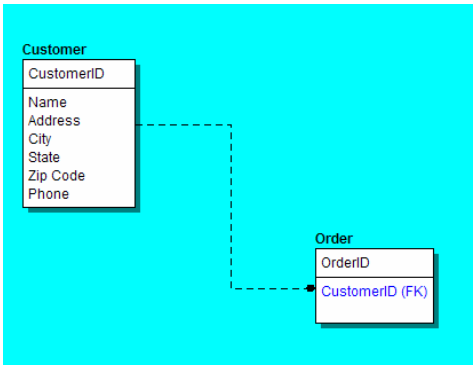
11. Next, select the "Non-Identifying" relationship from the Diagram Toolbar:



(**Note**: depending upon the notation your model is currently set to (IDEF1X in this case). The icons for each relationship will be slightly different)

12. To establish a relationship between Customer and Order, click once over the 'parent' entity (Customer) and then click over the 'child' entity (Order).

***Note** that ER/Studio supports sound design practices by automatically propagating the Primary Key, from parent to child entities. If there are 'candidate' alternate keys that should be propagated to the child, you may choose the drop down box in the relationship Editor with all available parent entity keys. Deletion of the relationship will remove a "non-native" propagated attribute. If you wish to leave the child columns of the relationship or foreign constraint, you can check the "Make Native" option. In this case the CustomerID will be left in Order):*



What are Domains?

Domains are 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 for all of your entities to leverage as its primary key) which has the same data type, definition, rule, constraint, etc no matter where it is distributed and bound. Read more about Domains in ER/Studio's Help system.

Creating and Working with Submodels in ER/Studio

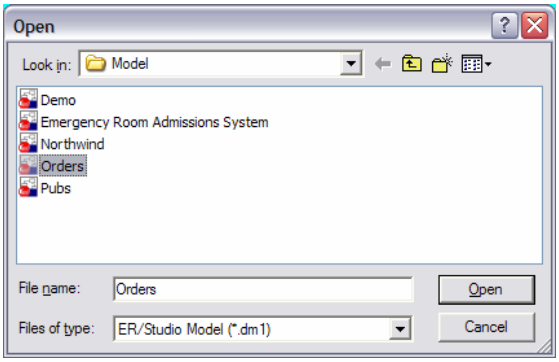
Now that you've achieved a general understanding of how to build logical models from the ground up in ER/Studio, it's important to understand how to work with an incredibly important navigation feature of the product called 'Submodels.' Let's close out of your current sample model, and open a more mature model. Follow these steps to learn more about Submodeling:

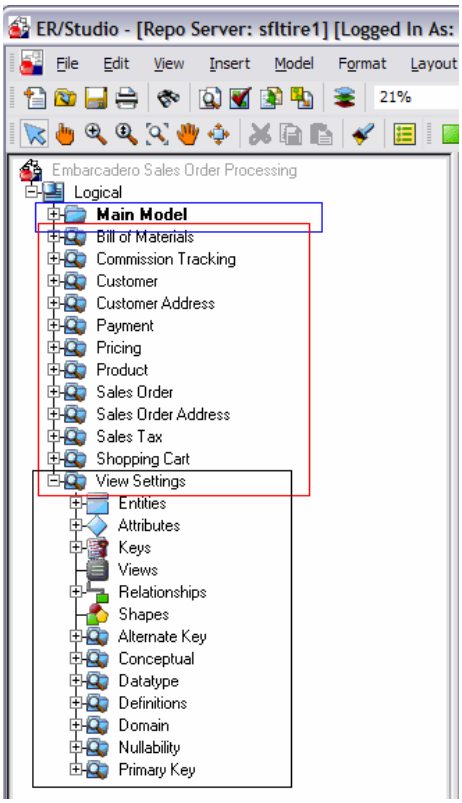
1. **File > Open** (or CTRL + O)

2. Choose the Orders file and click 'Open'.

3. At this point you can 'Save As' the Orders.dm1 file to a new name if you wish to preserve this sample model for future use. We'll be making modifications to it in this exercise.

4. Collapse the Data Model tab of the Explorer Browser to look like the image below:





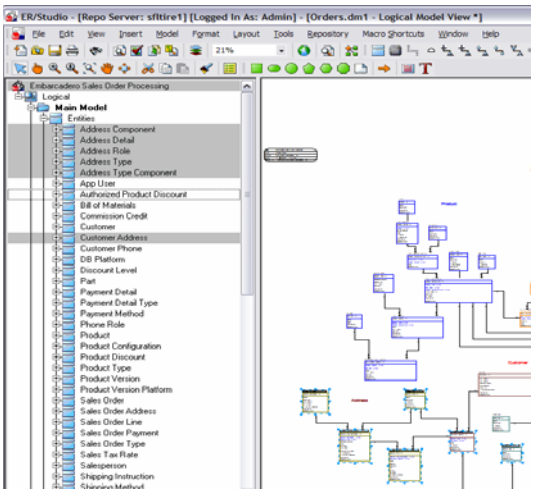
5. Note that in this Orders.DM1 sample model, there are no physical models, but a series of Submodel folders which help to describe the logical model:

"Main Model" is the entire collection of all logical objects for 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 **"View Settings"** are Submodels which are smaller collections of entities derived from the Main Model which help to describe specific areas of the Main Model free from other entities.

"Alternate Key" through **"Nullability"** are Nested Submodels, which can go 'n' levels deep and are literally "Submodels of Submodels."

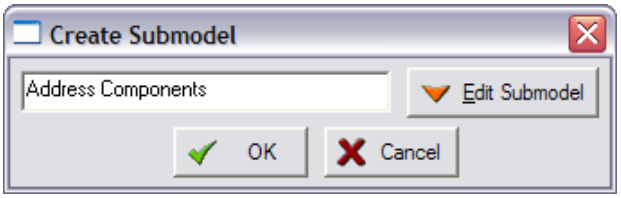
Feel free to click on and expand these folders to navigate them.



6. To make a new Submodel, navigate to “Main Model.” Let’s create a Submodel with all those objects related to Orders.DM1’s “Address” components.

7. With your CTRL key depressed, click on all the following objects in the Explorer Browser as seen in the image on the right. Any objects selected in the browser will also be selected on the diagram as well (also seen here). This can also been done by lassoing entities on the diagram workspace.

8. With the entities selected, select ER/Studio’s **Model > Create Submodel** menu.



9. Provide a name “Address Components” for the Submodel (seen to the right):

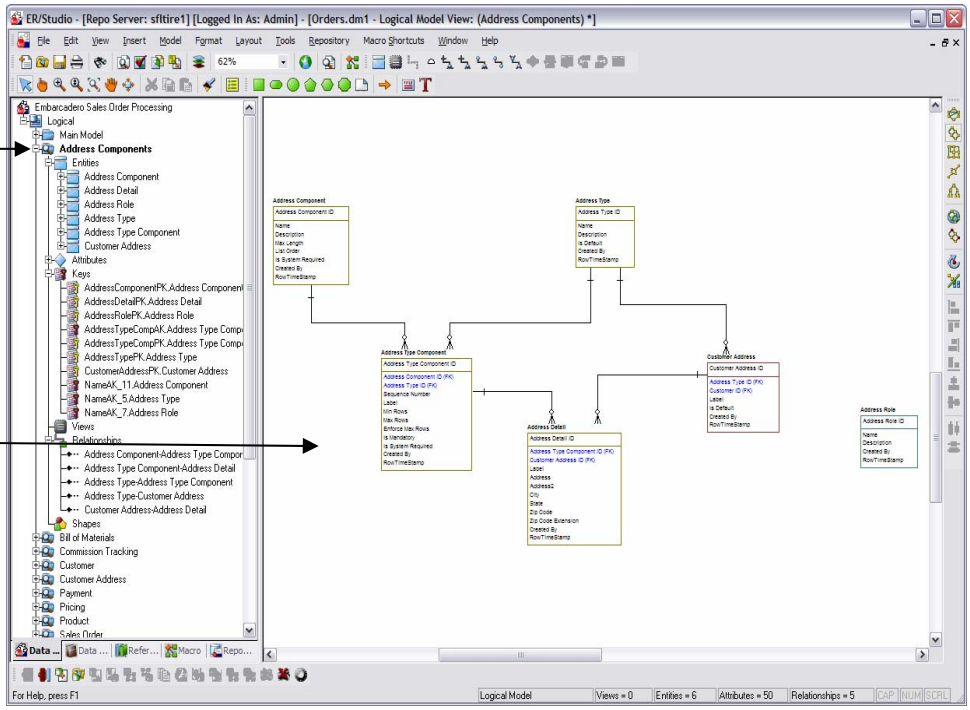
10. After hitting **OK**, ER/Studio will automatically create a Submodel called “Address Components”

What do the results look like and how do I navigate to the Submodel? :

Once created, you’ll see the new Submodel listed in the Explorer Browser and denoted with a magnifying glass over its folder as in the case of Bill of Materials and the other Submodels.

The new “Address Components” Submodel folder:

The chosen “Address Components” Submodel objects



What are Submodels?

Submodels and 'Nested Submodels' (e.g. "Submodels of Submodels") are designed to break down large, complicated Main Model views of a data model in order to focus on a specific area. An important understanding of Submodels is that any changes made here 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 its Main Model counterpart will have the change propagated automatically.

Generating Physical Models from a Logical Model

ER/Studio has the ability to generate as many Physical Models from a single Logical Model as the user wishes. There are many ways users may leverage multiple Physical Models in ER/Studio to help their design process. Examples of this ability in use are:

a. **Managing Change of an Existing Application:** Maintain independent Development, Test and Production Physical model diagrams representative of specific databases.

b. **Migrating Database Applications:** Use ER/Studio as an analysis and design 'hub' for migrating database applications. Manage a physical model of the legacy 'source' database application in conjunction with its new 'target' Physical Model which may be for an entirely new DBMS than originally maintained in the legacy database.

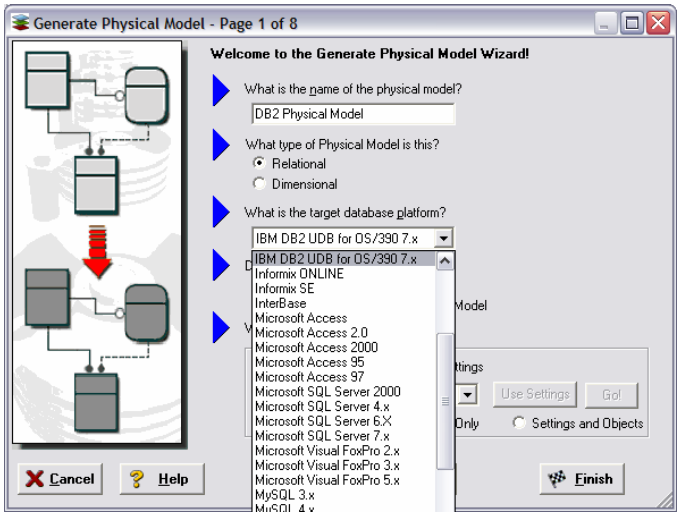
Let's generate a new Physical Model from a Logical in order to build a database. Let's use the Orders.DM1 sample model to demonstrate.

1. Open the "Orders" sample Model. (Use the steps shown above in the last session to do so).

2. From the Main Menu Bar select **Model > Generate Physical Model**.

Note: *ER/Studio will invoke a step-by-step wizard to walk you through the process of generating a DBMS-specific Physical Model.*

3. Name the new Physical Model and select the target DBMS you wish to generate. As in the image to the right, the name of the Physical Model will be "DB2 Physical Model" and DB2 UDB for OS/390 7 has been chosen from the drop down list as the target Physical Model.



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

Note*:* The wizard will prompt questions 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, naming conventions and a DBMS-specific validation check will be provided in this wizard.

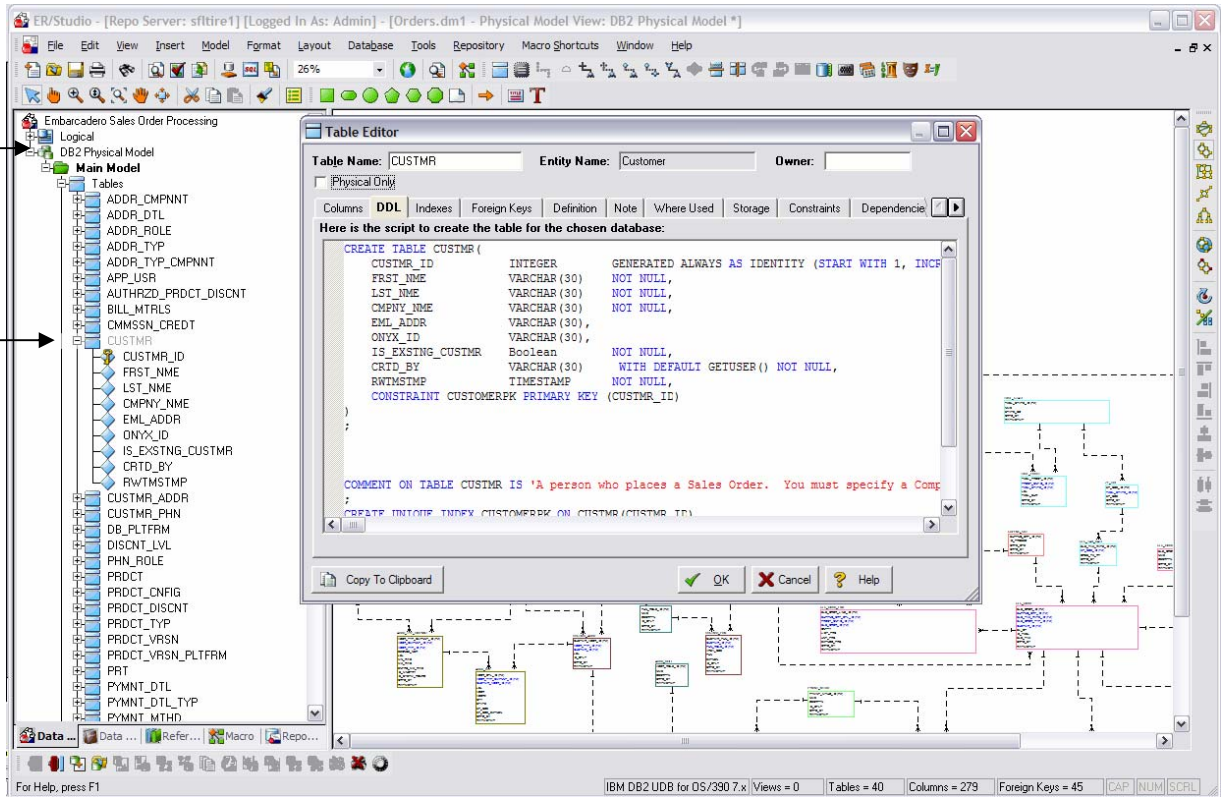
(TIP!** The Quick Launch can be used to store common settings so that the operation can be reused on this model or other any other models. If you wish to reuse the settings on another model, you just need to choose the "External File" option instead of the "In DM1" option when saving the Quick Launch information.**)

5. Select **Finish** on the last page of the wizard to generate the new Physical Model.

Now that a Physical Model is generated from the Logical, feel free to navigate to specific objects via the Explorer Browser (such as the CUSTMR table selected here, double click and view the physical details of the object such as DDL, Indexes, Partitions, Storage etc:

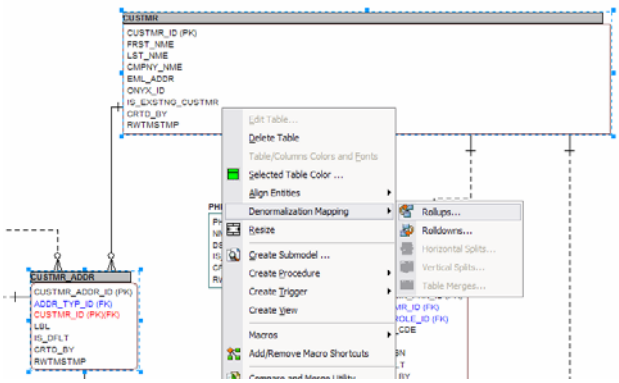
New Physical Model in Explorer Browser

Double Click on CUSTMR in the Explorer Browser to expose the CUSTMR Table Editor and navigate through specific details of the object like DDL in this case.



Denormalizing the Physical Model

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

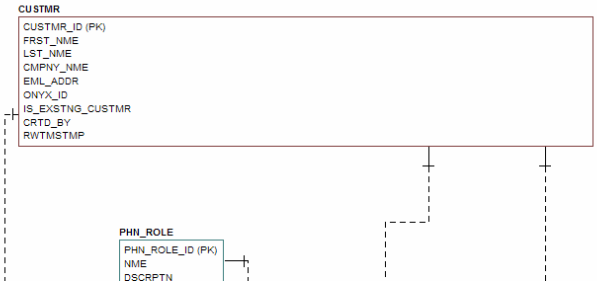


Note: The active, denormalization wizards depend on what objects are selected in the physical model. For example, if two tables have an identifying relationship between them, the valid operations are Roll up or Roll down. Notice that CUSTMR and CUSTMR_ADDR are selected and there is an

identifying relationship between them, so the Rollup and Rolldown operations are available.

Let's walk through an example of a denormalization operation using the generated physical model in previous section. We may want to reduce the overhead on the CUSTMR table by splitting it into two physical tables, CUSTMR_EAST and CUSTMR_WEST.

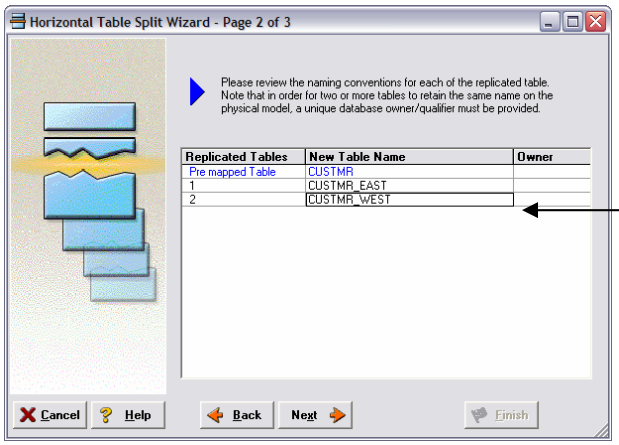
Before the operation, the CUSTMR table should look like:



1. Right click on the CUSTMR table.

2. Go to **Denormalization Mapping > Horizontal Splits…** Notice that since only CUSTMR is selected, the only possible mappings are vertical and horizontal splits.

3. The Horizontal Split Wizard will launch.



4. In step 1, enter **2** for the number of splits

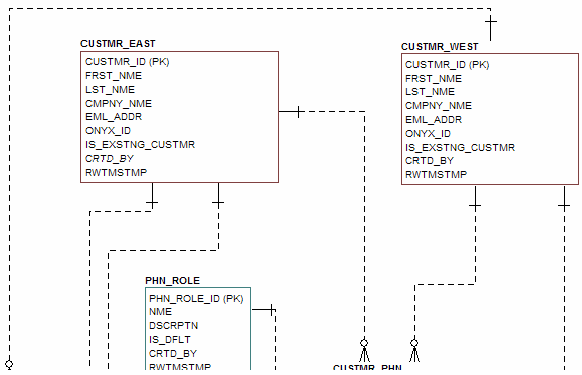
5. In step 2, rename each split to CUSTMR_EAST and CUSTMR_WEST

6. In Step 3, supply a name and definition for the denormalization operation.

7. Hit **OK.**

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

Note: The two tables are identical except for the name. To selectively choose which attributes end up in the resultant tables you can use a vertical split.



The denormalization mapping is stored in the data model tree underneath the submodels. You can use this to undo the operation or see the history of what happened. ER/Studio 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.

"Where Used" Analysis

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 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.

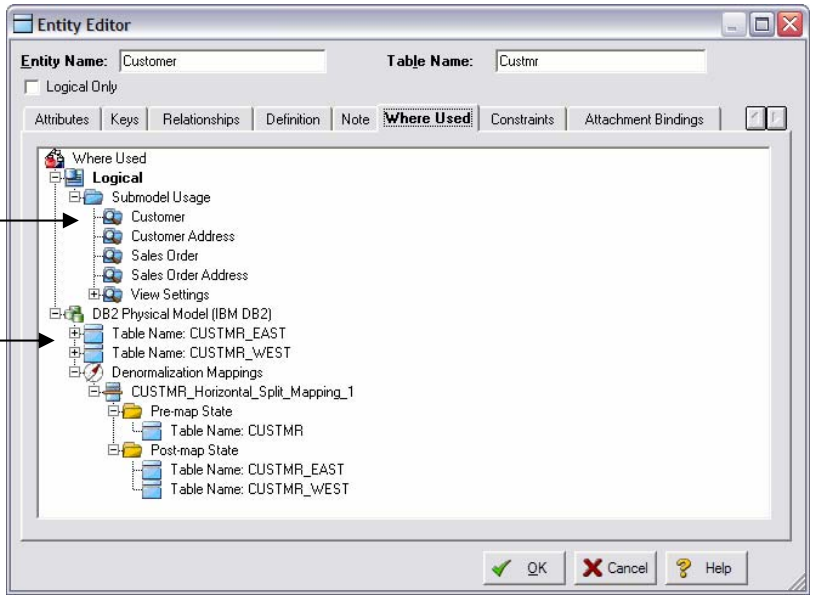
1. Navigate back to the Customer entity in the logical model using the data model explorer
2. Open the Customer entity editor
3. Navigate to the "Where Used" tab

Once the tree is expanded, you can now take a look at the lineage of what has happened to the object. Notice that CUSTMR_EAST and CUSTMR_WEST are listed as the physical implementations of the Customer entity. The denormalization mapping object from the data model explorer tree is visible to see how the end result was achieved.

The "Where Used" tabs will also show you the submodel usage of a particular entity within the logical or physical model. This allows you to see what business "areas" the entity belongs to.

Logical Model Submodel Usage

Physical Implementation of Customer



Note: The "Where Used" information is also available for attributes and columns.

Session 2 Conclusion:

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

a. Build a Logical Data Model from Scratch

b. Create a new Submodel view to understand how to model on specific parts of a larger 'Main Model.'

c. Generate a physical model from a logical in preparation for building a new database.

d. Denormalize objects in the physical model.

e. "See" the mappings between the logical and physical models using the "Where Used" tabs.

For more assistance on these issues, please feel free to refer to ER/Studio's Help system and review sections on "Logical Modeling Features" and "Physical Modeling Features" which will help describe processes like

- **SQL Generation** and…
- **Merging changes between Logical and Physical Models**.

Session 3: Documenting an Existing Database

One of ER/Studio's most powerful applications is that of a "documentation generator" for communication of complex databases and associated meta data to the Enterprise. The product is equipped, out of the box, with extensive report generation capabilities:

- **RTF Report Generation:** Instantaneous 'hard copy' documentation generation to environments like Microsoft Word.
- **HTML Report Generation:** Instantaneous generation of a HTML-based web site designed to provide simple navigability through data models and model meta data through standard Browsers such as Microsoft's Internet Explorer or Netscape Navigator.

In the exercise below, we'll use ER/Studio to reverse-engineer an existing database and generate an HTML report for immediate distribution and navigation by those who need to depend upon the information about the data model, but may not be permitted to have connectivity to the database for security or organization reasons.

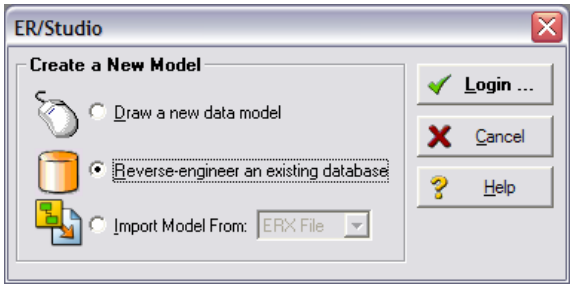
Generating an HTML Intranet Dictionary Report

PRE-REQUISITE: *For this exercise, it will be assumed that the user can connect to an existing database in order to document it. Please refer to "Reverse-engineering an existing database" in ER/Studio's Help Index for explicit set up details if needed. If no database is available, documentation can still be generated from sample models provided upon installation. Simply skip the steps below involved in reverse-engineering and begin at Step 9 after opening any sample model included with ER/Studio.*

1. From the main menu bar, select **File > New**.

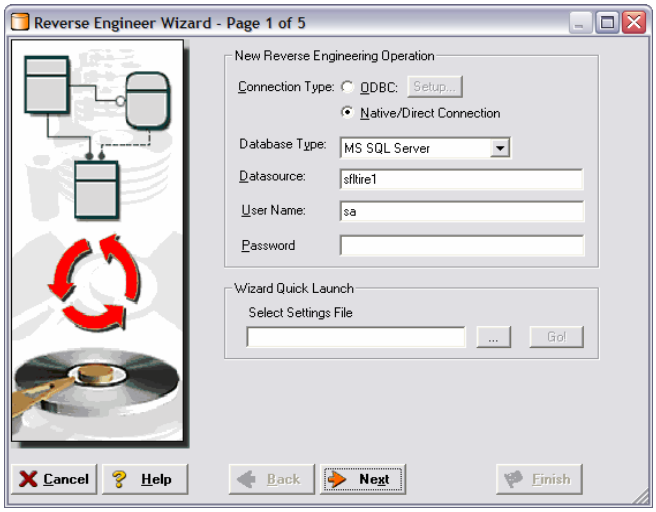
2. Select the middle radio button to Reverse-engineer an existing database:

3. Click **Login**…



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

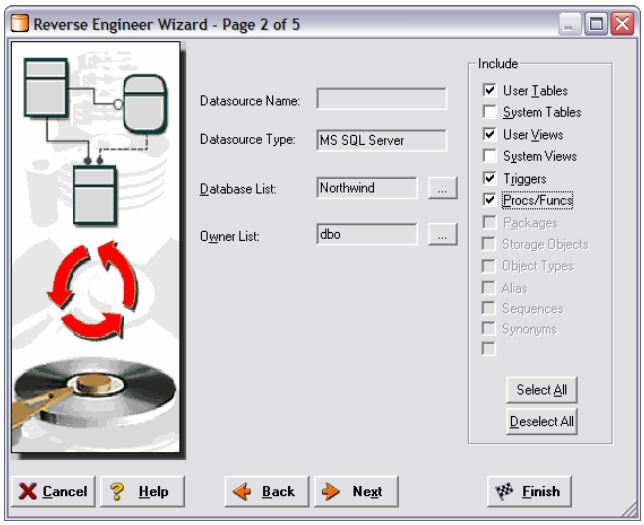
5. Input the correct connectivity information such as the datasource name and user name/password information.



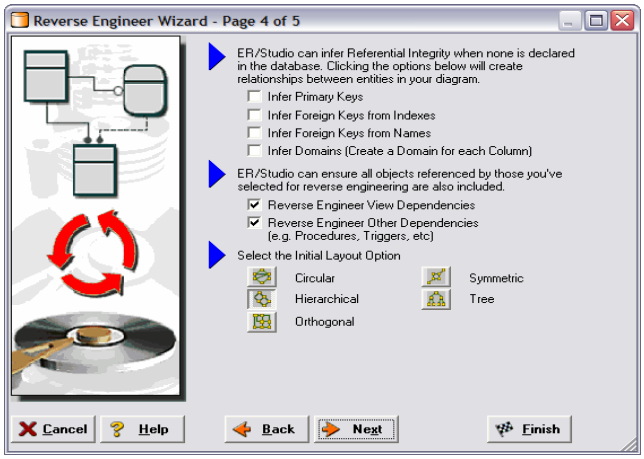
6. Click '**Next'**.

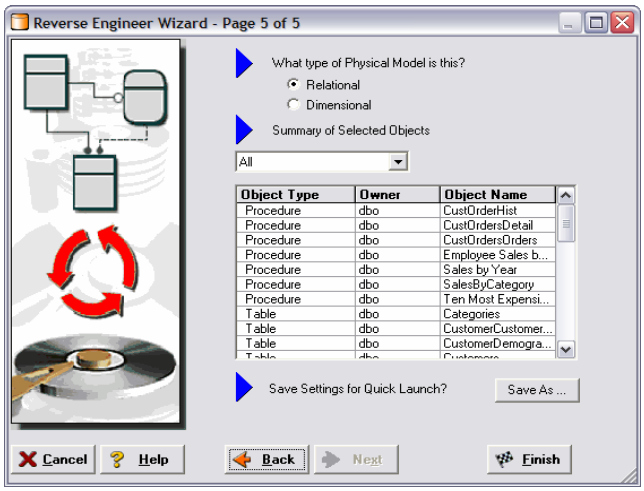
7. Walk through the Reverse Engineering wizard to select the objects, options and layout preferences you wish for your database model:

Choose the SQL Server Database (Northwind in this case) and object types (Tables, Views, Procedures, etc)…



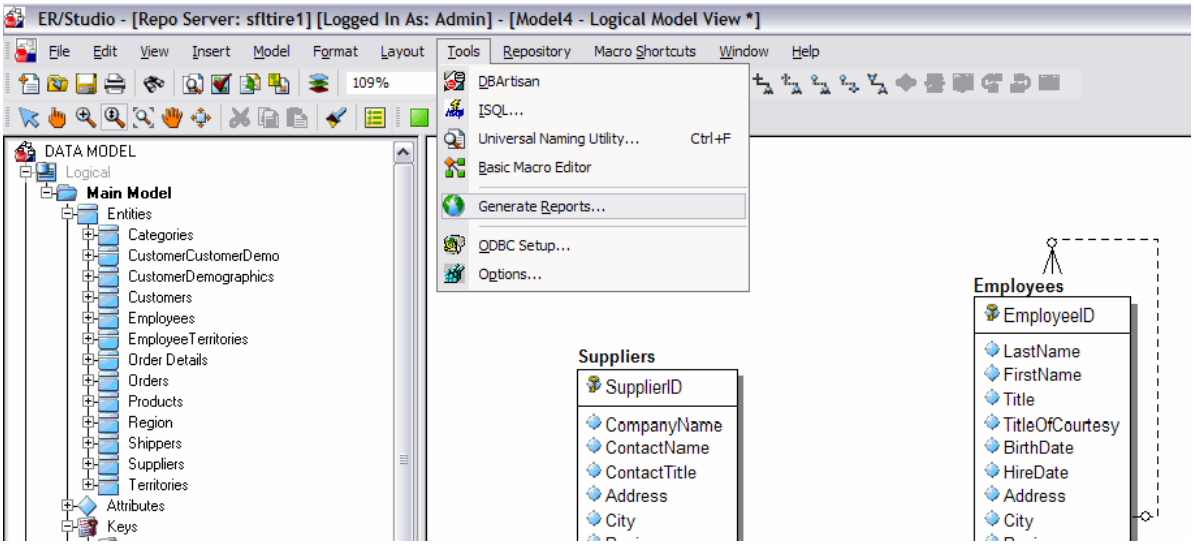
Continue through the wizard to select layout styles and other preferences…





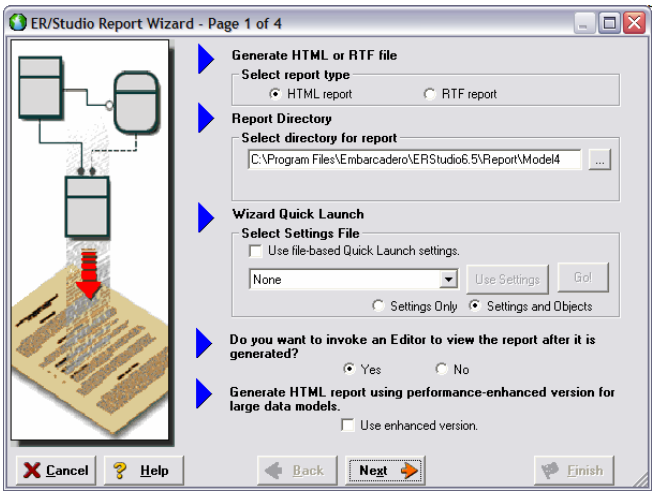
8. Hit "**Finish**" and let ER/Studio reverse engineer your database!

Once reverse engineering of your database is complete, let's now generate a complete HTML report of the database for others in your organization to review….

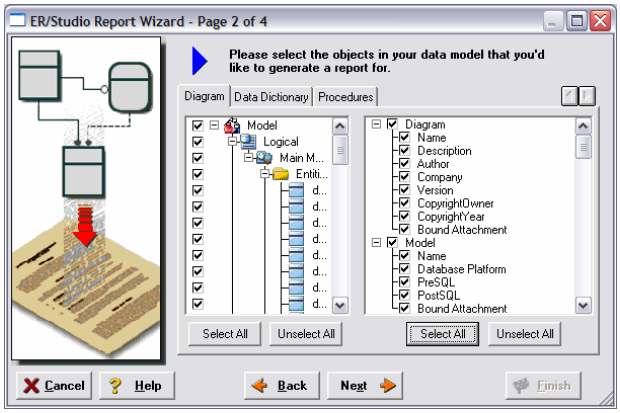


9. As depicted in the picture above, select **Tools > Generate Reports.** The Generate Reports wizard will deploy. Note that the Physical 'Main Model' has been selected in the Explorer before reporting.

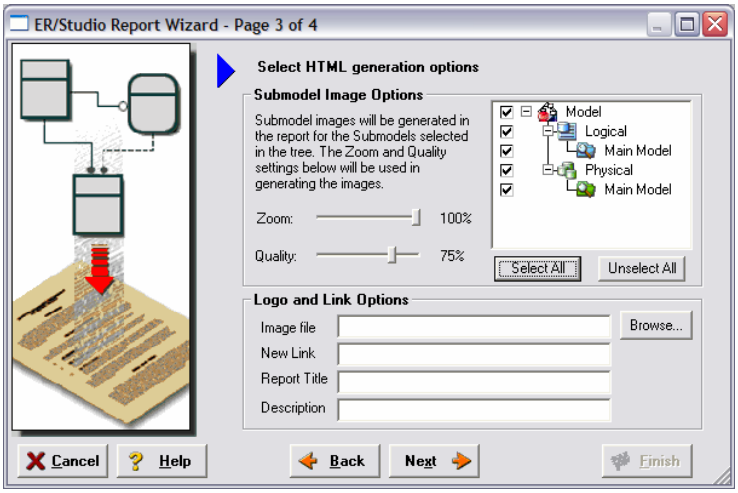
10. On the wizard's first page, select 'HTML' as seen here:



11. On page 2, leverage the wizard's infinite customization and select what you want for your report. Ensure BOTH the desired objects in the left pane AND the criteria selection options in the right pane have been selected. 'Select All' has been chosen in both panes in this screen shot. Do the same for all objects:



12. On page 3, you have the option to include many features such as model images that should be included in the report, a custom corporate Image to replace the default Embarcadero image, report title information and more: Select the model images you see depicted here:

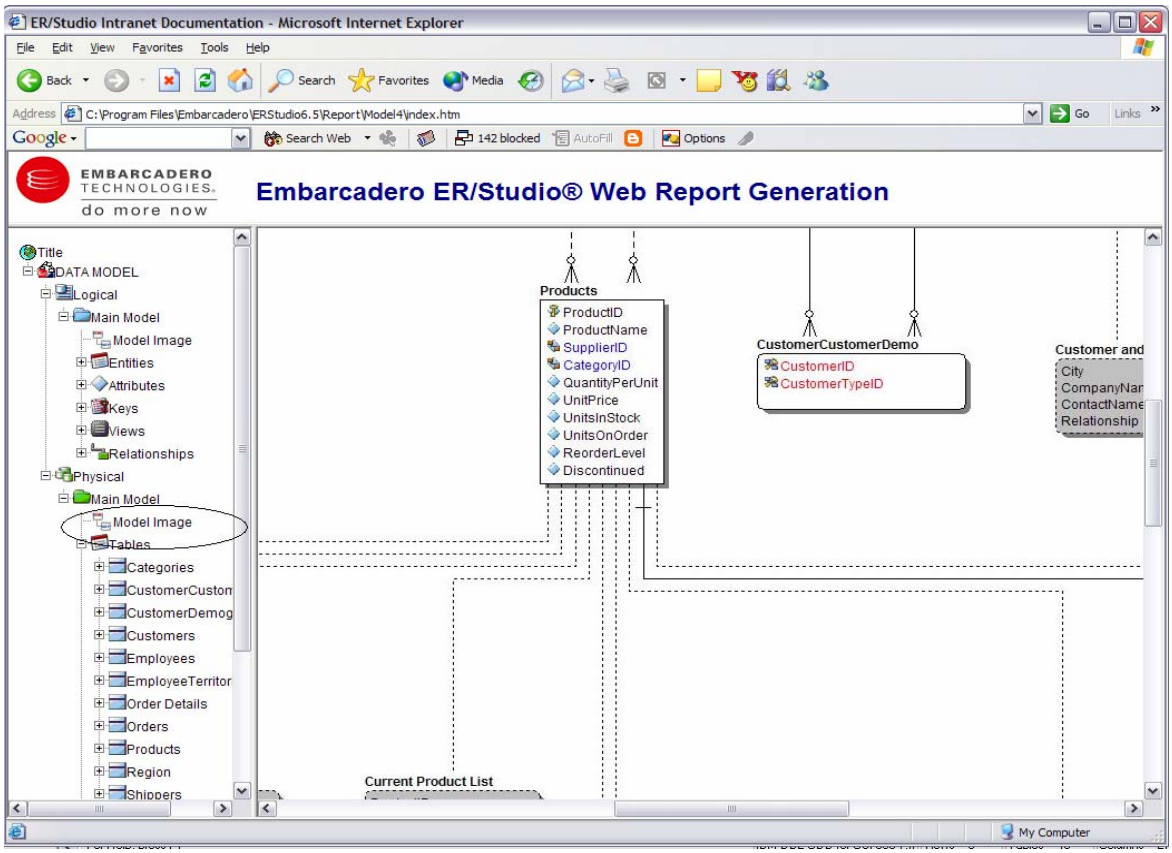


13. Hit next to advance to Page 4 of 4 and hit "Finish". ER/Studio will then begin the Report publication process and call up your default browser to review the report.

Note: In the 'Logo and Link Options, you may select to replace ERStudio's default 'Embarcadero Technologies' logo in favor of your own corporate logo (and underlying Hyperlink).

Finished! Start navigating the report via the Explorer Browser. The navigation will be exactly as it is if you are using ER/Studio! Expand the tree to find 'Model Image' and click on it (see below). You will be presented with a read-only version of your data model (as seen below). The Explorer will navigate you to

ANY metadata you wish or simply select the entities and relationships in the model image to jump to their information.



Session 3 Conclusion:

In this session, you have learned

a. How to connect to and reverse-engineer an existing database with ER/Studio

b. How to document this database in seconds by ER/Studio's automatic HTML documentation publication facility.

For more assistance on Reporting, please feel free to refer to ER/Studio's Help system and review the section on Reports found under the main Contents listing.

Session 4: Diagram Navigation and Aesthetics

The fruit a powerful data modeling product like ER/Studio bears are its diagrams. To assist with presentation-quality diagrams that provide simple navigate, are aesthetically pleasing and most of all offer easy means to 'clean up' complex diagrams, ER/Studio offers progressive diagram Auto Layout and Navigation utilities. Modelers should spend time on solving complex database or business data model problems, not spending time forcing 'boxes and lines' to look a certain way.

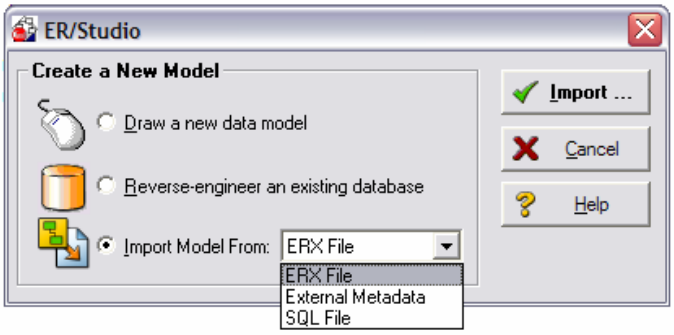
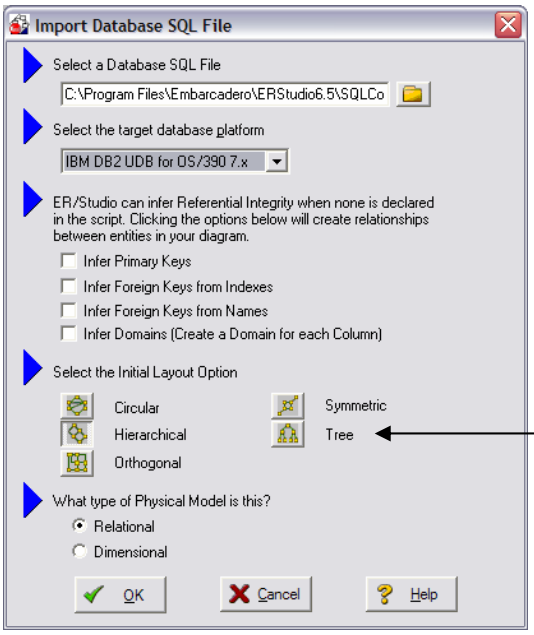


Diagram Navigation

To demonstrate some of ER/Studio's layout and navigation utilities, we'll import a sample SQL script provided with your evaluation copy of ER/Studio.

1. Launch **File>New>Import File**…choose SQL File. (**From ERX file** indicates the products ability to import Computer Associates ERwin 3.5.2 ERX files, **From External Metadata** launches the MetaWizard to import from alternative sources. See below for more info.).

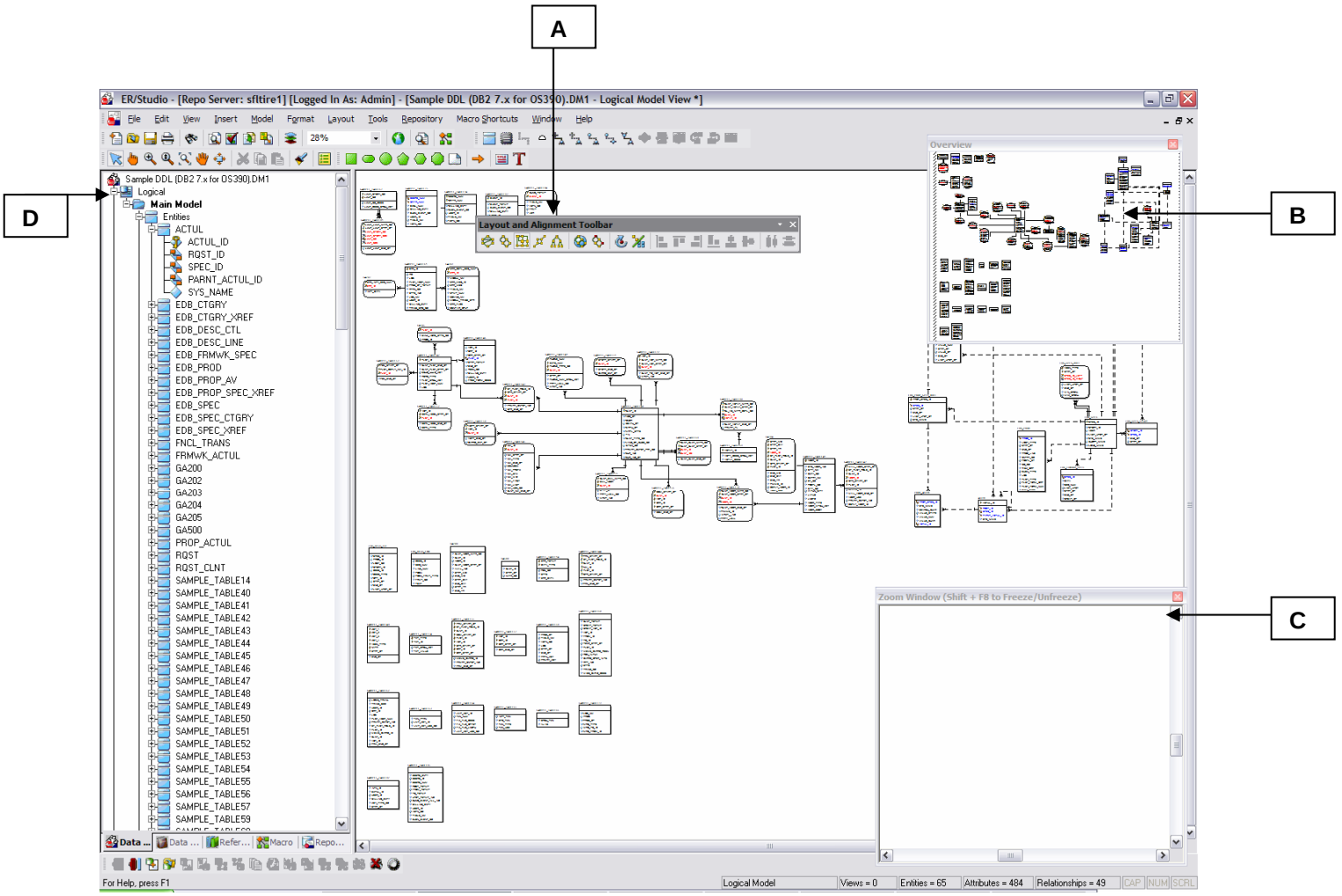
2. Once the Import Database SQL File is launched, navigate to C:\Program Files\Embarcadero\ERStudio6.5\SQLCode\Sample DDL (DB2 7.x for OS390).SQL. This SQL file is sample IBM DB2 OS/390 7.x. Ensure that is selected as the target platform as you see depicted below.



3. Hit "OK"

Note the ability to select an initial layout style for your model BEFORE the SQL import has taken place.

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.



A: **Layout and Alignment Toolbar:** Use any of the 4 Auto Layout styles to change the layout of the diagram with the click of a button. These are all entirely customizable styles as well via the Layout Properties dialog found in the “Layout” Main Menu.

B: **Overview Window:** (If this is not already visible, hit your F-9 key to activate or in ‘View’ drop down menu). Use this 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.

C: **Zoom Window:** (If this is not already visible, hit your F-8 key to activate or in ‘View’ drop down menu). Use this as a ‘magnifying glass’ to enlarge any diagram objects under your mouse cursor. You may also use the SHIFT + F8 key stroke to ‘freeze’ the zoom window to keep a single object frozen while you continue to pan around the diagram.

D: **Dynamic Explorer Browser**: Click on any object in the Explorer Browser and it will automatically be selected in the diagram and focused in both the Zoom and Overview windows.

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 opt to limit or expand what is displayed. For example, developers may benefit from looking at a model that displays data type, null option, unique and non-unique index information, etc while business analyst may just need the entity name and the definition. ER/Studio offers many display properties that can be customized exactly for this purpose.

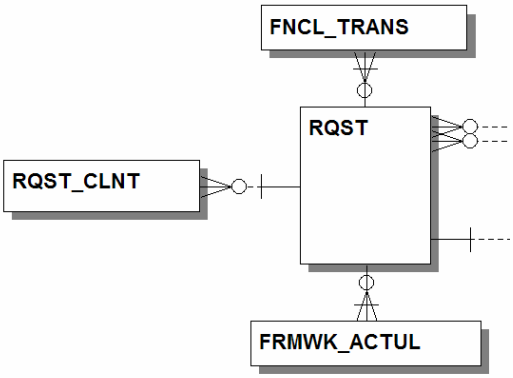
Continuing with the previous section, we'll use the DB2 model that was built to demonstrate some of the ways to customize the appearance of the model.

We'll use the **Diagram and Object Display Options** dialog on the Diagram toolbar to customize the view of the logical and physical models.

Note: You can use the **Colors & Fonts Dialog** to customize the look and feel further of each model.

Setting the Logical Model Display

1. Navigate to the logical model and launch the **Diagram and Objects Display Options** dialog from the diagram toolbar.



2. Under the **Entity > Display Level** select "Entity"

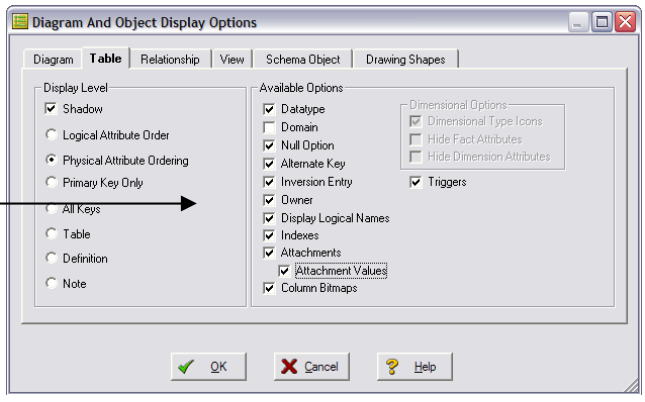
3. Hit **OK.** Notice that only entity names are displayed for each entity. You may also want to re-layout the diagram since the entity sizes have changed.

Setting the Physical Model Display

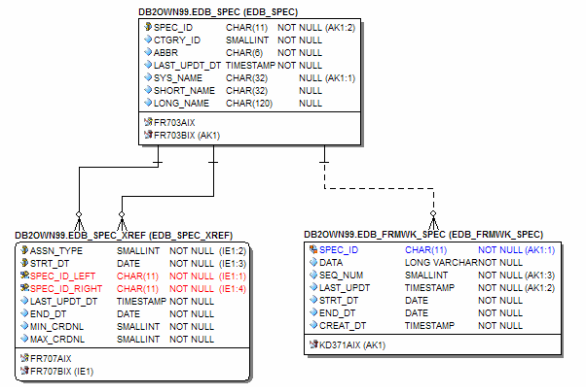
1. Navigate to the physical model and launch the same **Diagram and Objects Display Options** dialog.

2. Under **Table > Display Level** select "Physical Column Ordering"

3. Under **Available Options** select the specific properties you want to display



4. Hit **OK.** Your model should now have a more detailed display for the physical model, seen below.



Note: Since the size of the objects has changed, you may want to re-layout the model using one of ER/Studio's advanced layout engines. You can also customize the default display properties for new models in **Tools > Options > Display.**

Session 4 Conclusion:

In this session, you have learned

a. How to import a *.SQL file into ERStudio and allow it to automatically create a diagram
b. Use a variety of Auto Layout and Navigation tools to enhance the experience of diagram aesthetics and overall navigation of the data model.
c. How to customize the display of both the logical and physical models.

Session 5: Metadata Importation and Exportation

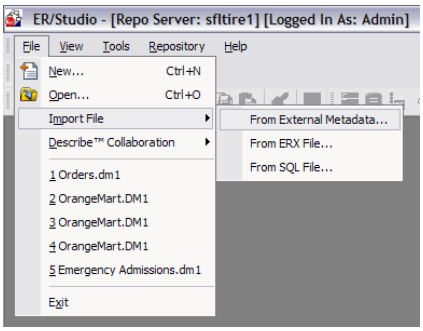
MetaWizard allows you to import and export metadata from a wide spectrum of sources and targets. Various formats of metadata are supported which provide you connectivity to environments like XML Schemas and DTDs, OMG's CWM-XMI, business intelligence repositories like Business Objects, Cognos, DB2 Cube Views, various UML and data modeling tools and more.

Note: The MetaWizard is a separately licensed module. For evaluation purposes the Import Bridge is enabled during the install, but the Export Bridge is not. You will need to contact Sales@embarcadero.com to enable the Export Bridge for evaluation.

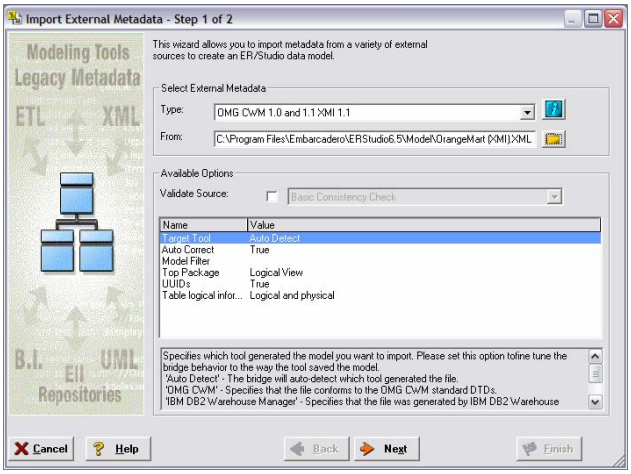
Importing Metadata

Let's walk through an example of building a model from a particular metadata source. In this case we'll use OMG CWM XMI 1.1, one of the popular formats used by various modeling tools.

1. Launch the MetaWizard, by going to **File > Import File > From External Metadata:**



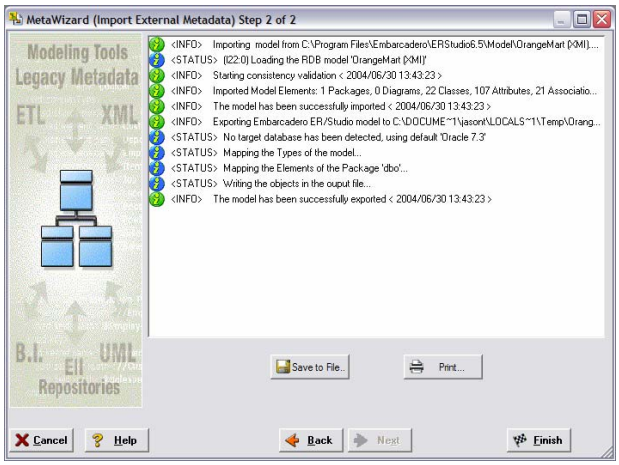
2. Select the appropriate type of metadata from the **Type** list. In this case use OMG CWM 1.0 and 1.1 XMI 1.1.



Note: Each respective environment has specific versions. This determines how the metadata is translated. If you intend to import models or metadata from another source of your own, please make sure you select appropriate platform and version.

3. Open **OrangeMart (XMI).xml** from the **ERStudio\Model** folder by hitting the **Browse** button.

4. Hit **Next.** The MetaWizard will run a validation on the file imported.



5. Hit **Finish.** This will build a logical and physical model based on the source metadata. Please keep this model for later use in the Evaluation Guide.

Note: In some cases the layout will not import from the source metadata. If this happens you can use one of ER/Studio's advanced layout engines explained in an earlier section.

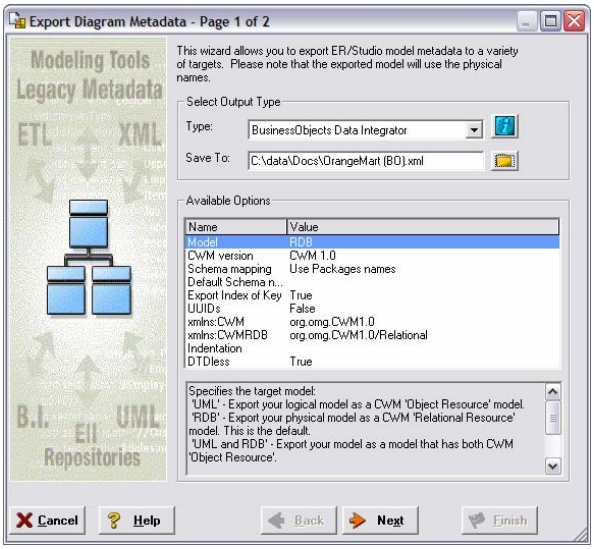
Exporting Metadata

ER/Studio can export metadata in the same formats from which it can import. The difference is that metadata can be exported from the entire diagram using the **Export** on the **File** Menu or any of the submodels including the main model by right clicking on the submodel.

Let's walk through an example of exporting. We'll use the model that was built from the XMI file. In this example we want to export the diagram metadata to Business Objects so the business intelligence folks can use the metadata to generate reports.

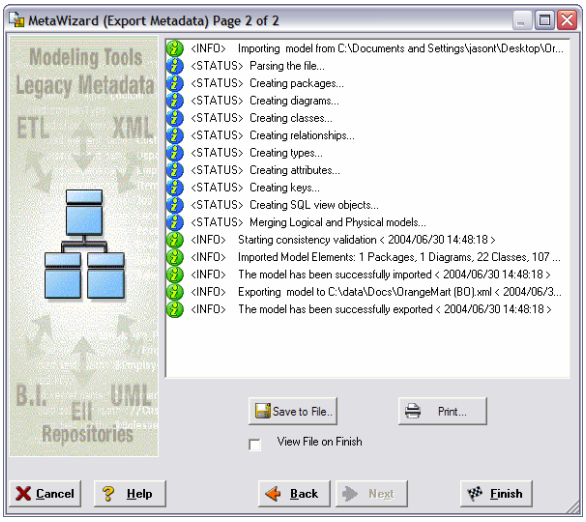
Note: You need the proper license configuration to continue.

1. Go to **File > Export File > Export Diagram Metadata…** This will launch the Export Bridge.



2. Select Business Objects Data Integrator as the type of Metadata.

3. Choose a file location for the XML file.



4. Hit **Next.** This will run a check on the exported metadata.

5. Hit **Finish.** This will save the file to the specified location.

Session 5 Concussion:

In this session we've explored metadata management capabilities of ER/Studio, specifically:

a. How to import metadata from a wide range of sources to produce a logical and physical model.

b. How to export metadata to an equally wide range of sources so that metadata can be shared with other groups with you organization.

Session 6: Dimensional Modeling

ER/Studio allows you to model dimensional structures such as star and snowflake schemas which can be leveraged for data warehouses, data marts and OLAP. ER/Studio's dimensional notation helps you easily visualize and build these complex models by using icons for the various table types and enforcing rules specific to dimensional modeling standards. This session will help you construct a dimensional model and walk you through some of the aspects inherent to the dimensional notation.

Overview of Dimensional Notation

First, let's create a dimensional model. There are a number of ways you can designate a model as "dimensional".

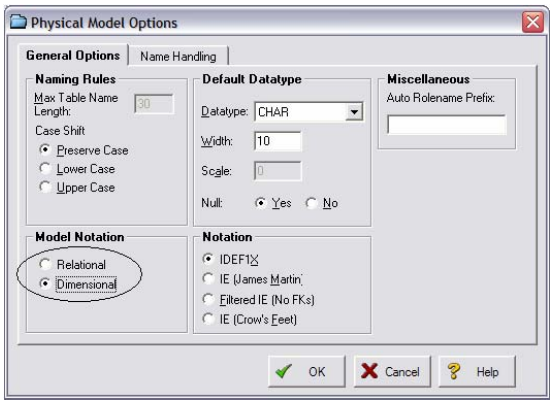
If you are creating a new model you can do one of the following:

1. In Step 5 of the Reverse Engineer Wizard.
2. In the SQL Import Dialog.
3. In Step 1 of the Generate Physical Model Wizard.

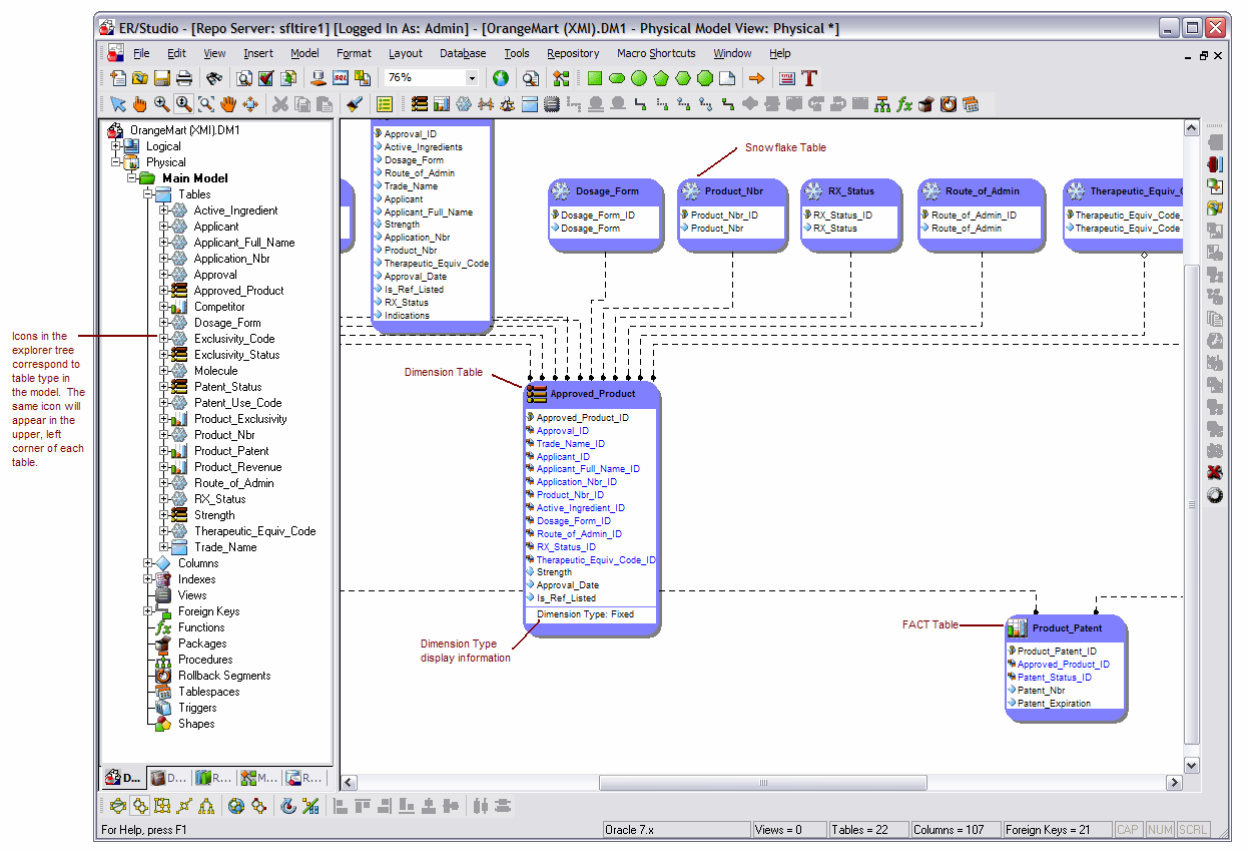
If you have an existing physical model, you can change the type in **Model > Model Options.** For the purpose of this session, we will use the model created from the XMI file in Session 5. If you skipped to this session, please go back to Session 5 and walk through the **Import Metadata** section.

Since we have an existing model, we'll have to change notation.

1. Navigate to the Physical model.
2. Right click on the physical model and go to **Model Options.**
3. Set the Model Notation to Dimensional.
4. Hit **OK.**



Notice that the look and feel of the tables has changed and each table has a specific icon depending on the type of table ER/Studio "thinks" it is. ERStudio will analyze the foreign key chains of the model and use dimensional modeling rules to decipher fact tables from dimension and snowflake tables or other dimensional tables. The picture below gives an overview of the notation.

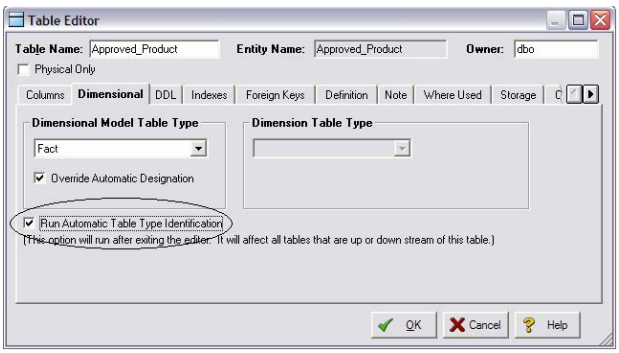


Working in a Dimensional Model

Notice in the picture above that ER/Studio "guessed" that Product_Patent is a fact table, Approved_Product is a dimension table and the parent tables of Approved_Product are snowflakes. This is because Product_Patent has no child tables, Approved_Product is a parent of it and the parent tables of Approved_Product are two relationships away from the perceived fact table. Analyzing this a little further, it looks like Approved_Product is actually the fact table, Product_Patent could be a bridge to another fact table, and the parents of Approved_Product are actually qualifiers of Approved_Product or dimensions. The table type can be changed to override what ER/Studio originally designated the table.

Let's walk through an example.

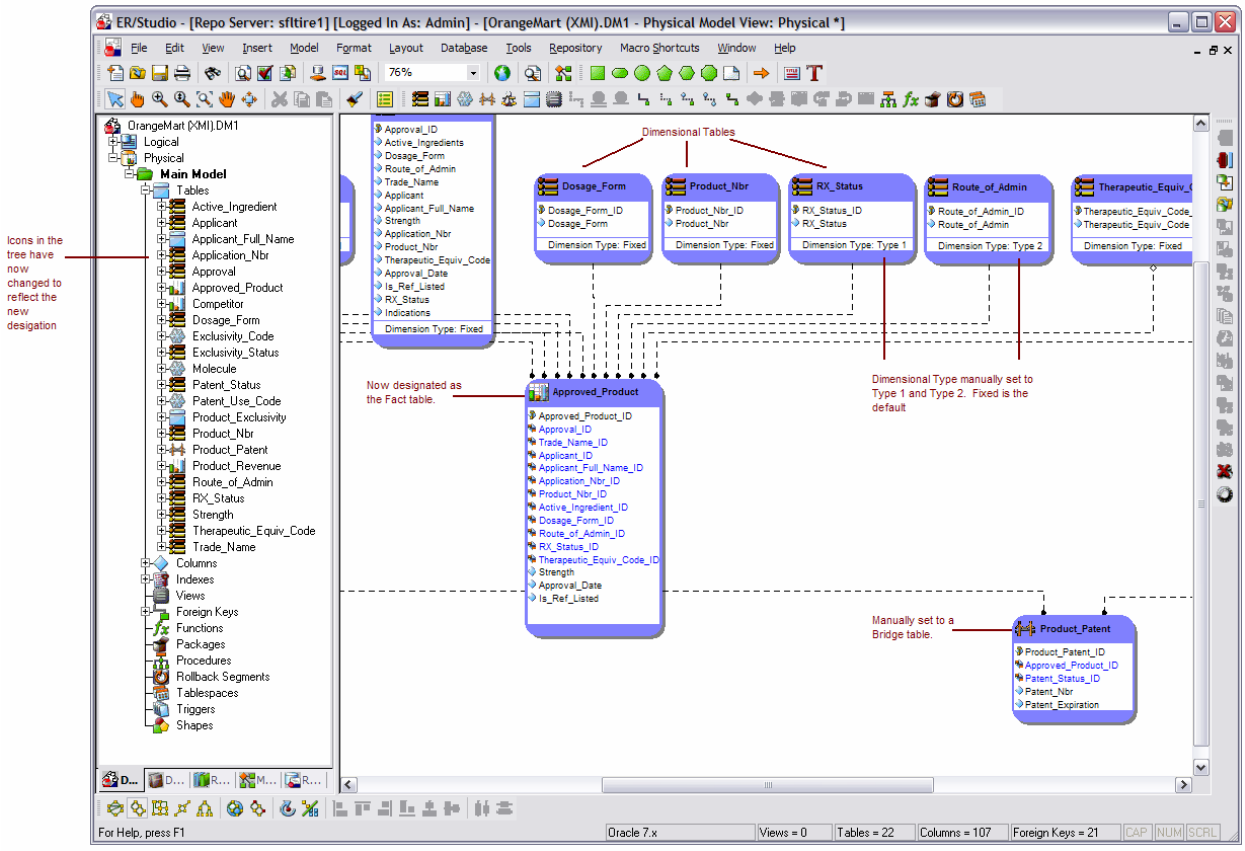
1. Open the editor of the Approved_Product table.
2. Go to the Dimensional tab.
3. In the **Dimensional Model Table Type** drop down, select **Fact.** The **Override Automatic Designation** option will be checked.
4. Next, check the **Run Automatic Table Type Identification** option.
5. Hit **OK.**



***(TIP!** If you want to change designation of a table without affecting related tables, you can turn off the **Run Automatic Table Type** option.)*

Note: There are other dimensional model type tables such as Bridge and Hierarchy Navigation.

The result will now be that Approved_Product is now the fact table. The parent tables if it will all become dimension tables and the Product_Patent table will be designed undefined. As another exercise, you can go into some of the dimension tables and change the type of dimension depending on the desired data refresh rate as seen in the below picture.



Session 7: Automating Tasks

ER/Studio is equipped with a highly documented **Automation Interface**. This automation interface is driven by the Sax Basic language (a derivation of the Visual Basic for Applications language) and serves many purposes, fundamentally for the purpose of allowing API access to ER/Studio for our customers for business and user-specific customization of the product.

There are two main reasons to employ the Automation Interface:

- **Automate Routine Tasks** Use the automation interface to automate tedious, routine modeling tasks or customize ER/Studio to enforce modeling practices in your organization. For example, write a macro that will automatically colorize child tables that contain propagated foreign keys. Or, write a macro to automatically insert a specific name and primary key into new entities as they are created.

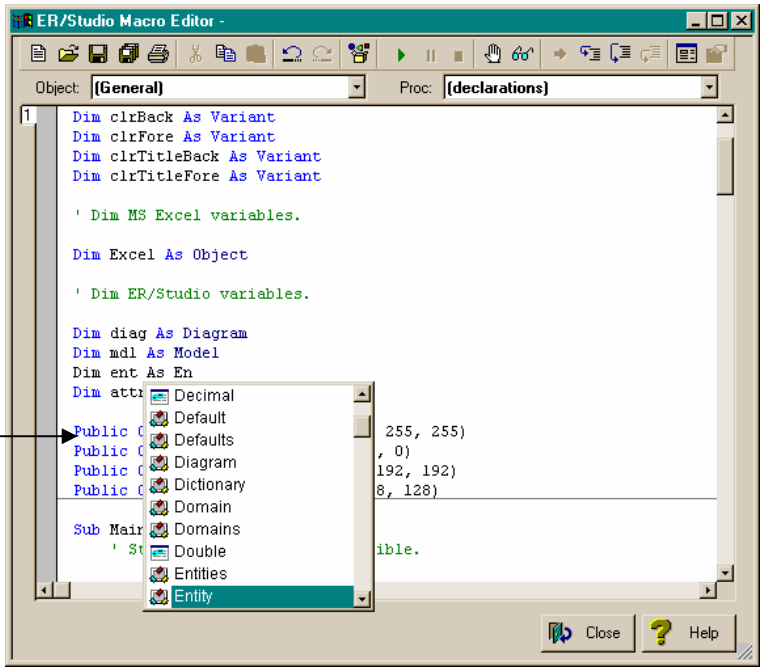
- **Collaborate with other applications** Your ER/Studio models contain valuable meta data that you would like to access from Microsoft Excel, Access, Outlook, etc. Using ER/Studio's new automation interface, you can now collaborate with any external application that has an exposed API or its own automation interface.

In this walk-through, we'll demonstrate an example of ER/Studio leveraging its automation interface to dramatically increase modeler productivity. You'll not be writing any Sax Basic (VBA) code in this walk-through, but simply running a macro that's already been written for you and included with the product. Users may write their own macros through a Sax Basic Integrated Development Environment included in ER/Studio:

The Sax Basic Integrated Development Environment:

To access this editor and create your own macros, pull down ER/Studio's **Tools > Basic Macro Editor** selection.

Note the use of automatic drop downs dialogs to access ER/Studio's referenced Automation Objects as the user is typing:



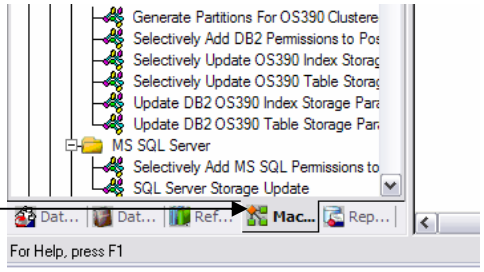
Automating the modeling process:

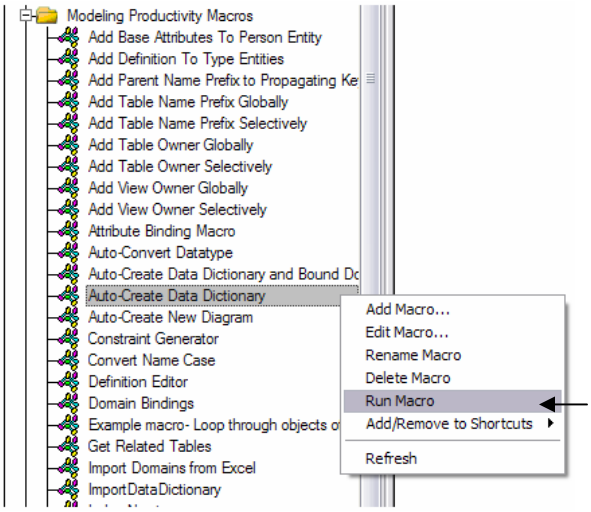
Using Visual Basic macros to speed development and enforce the re-use or meta data.

As indicated above, we'll be leveraging macros provided with the product in this example which will help to demonstrate powerful Automated modeling activities users can benefit from.

1. Close all prior diagrams down. Choose **File > New** and select 'Draw a New Data Model'.

2. Once Step 1 is complete, switch to the 'Macros' tab of ER/Studio's Explorer:

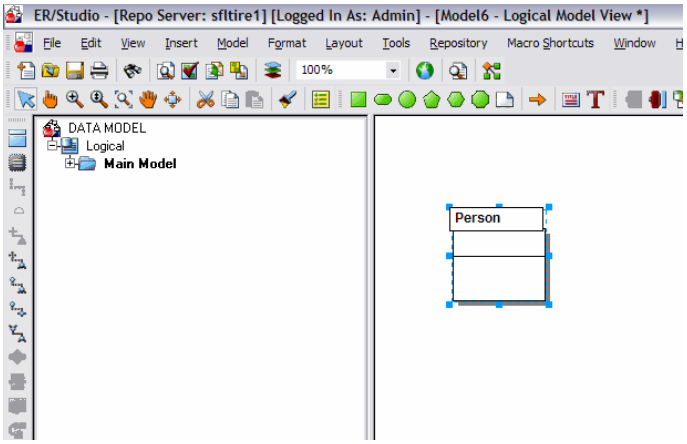




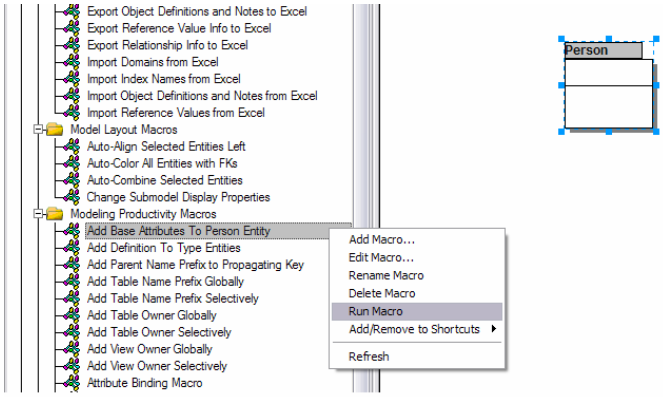
3. Locate the "Auto Create Data Dictionary" macro in the Explorer.

4. Right mouse Click over this macro and choose 'Run Macro'. Running this macro will create a set of Domains in the Data Dictionary which the next macro run will leverage. (See Section 2 above for more information on Domains).

5. With the set of Domains now ready, create an entity on the drawing workspace. Name this entity 'Person'. You do not need to implement ANY attributes for your Person entity at this time:



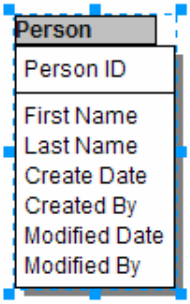
6. With the Person entity selected on the workspace, find the "Add Base Attributes To Person Entity" macro, and like step 4 above, Run Macro:



7. Once complete, you'll note that the Person entity will have ALL of its attributes automatically created for you! You have just:

- Saved the effort of manually typing these 'standard' attributes into selected entities that require them.

- Bound all the new attributes to ‘Domains’ for proper standard enforcement.



You are free to customize these macros in any way you choose! This example merely stresses how to increase the productivity of modelers to automate repetitive tasks such as ensuring entities conform to the same standard set of attributes.

Feel free to explore the other macros we’ve included for you as well to see how they may increase your productivity.

Session 7 Conclusion:

In this session, you have learned

a. How to access the Basic Macro Editor to create your own macros from scratch.

b. How to locate the Macro tab of the Explorer Browser and launch sample macros included with ER/Studio to help increase modeler productivity.

For more assistance on the Automation Interface, please feel free to refer to ER/Studio’s Help system and review the section on ”Automation Interface” found under the main Contents listing.

Session 8: Collaborative Modeling with ER/Studio Enterprise

ER/Studio Enterprise includes a server-side component to ER/Studio designed to distribute work across modeling team members in a safe and controlled way, facilitating a real-time collaborative modeling environment and increasing productivity for teams out of the box. The solution implements utilities and features that enable concurrent modeling, version management for model and model objects, establishment of continually reusable data elements, and more. The secure, scalable environment is fully integrated with the current, natural workflow in ER/Studio.

This portion of the evaluation guide is intended to give a brief overview and walkthrough of ER/Studio Enterprise. It will start with the install and configuration of the repository and continue on to include inserting a diagram into the repository, working with the diagram in the repository, versioning the diagram, sharing and reusing objects across diagrams and finally applying security to your diagrams. It is intended as an introduction of the repository. For more information please refer to the repository section of ER/Studio’s online Help or contact Technical Support at mailto:support@embarcadero.com or call (415) 834 3131 x2.

Getting Started with ER/Studio Enterprise

Download and Install

Should you wish to evaluate the collaborative modeling benefits of ER/Studio Enterprise, you will need to download and install the separate installation executable. You can download the repository installation executable from the Embarcadero website at:

(http://www.embarcadero.com/downloads/download.html).

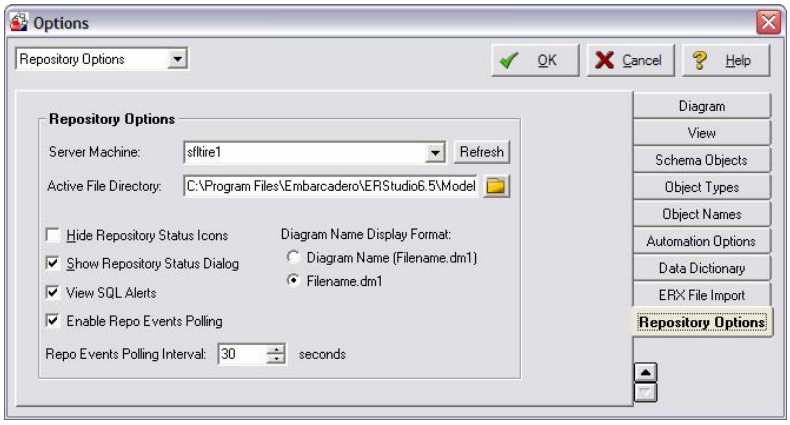
You need to download the ER/Studio Enterprise zip file or the ER/Studio Standard Upgrade to Enterprise executable.

ER/Studio Enterprise requires installation on an RDBMS of your choice: IBM DB2 UDB, Oracle, Sybase ASE or Microsoft SQL Server. Two components will be installed: the 'Server' and the 'Database'. The server machine requires the chosen database's client utilities to be installed in advance so that it can initially build and subsequently connect and communicate with the database thereafter.

For more comprehensive installation instructions, please refer to either:

- ER/Studio's online help under the "Installation" section. For information regarding database sizing, repository server requirements and Architecture, refer to "Repository Care and Maintenance" section under the Repository section of the ER/Studio Online Help.
- Installation Guides are available on Embarcadero.com: http://www.embarcadero.com/resources/documentation.html

Configuring ER/Studio to Connect to Repository



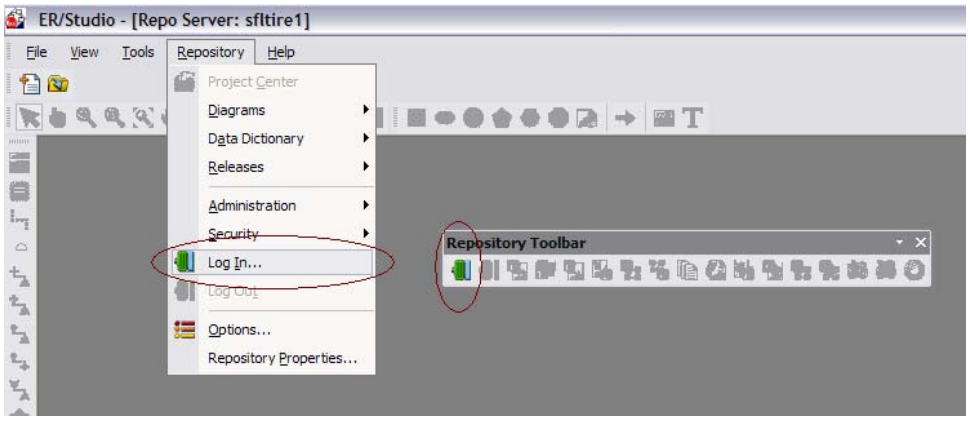
Once the server component has been installed, you can configure ER/Studio to connect to the repository by going to the Repository drop down menu and selecting **Options**. If no Repository menu is visible on ER/Studio's main menu, you can go to the general ER/Studio Tools\Options\Repository Options. ER/Studio can automatically detect Repositories already installed on your network by hitting the "refresh" button. Otherwise you can manually enter the Repository Server machine name in the specified field.

Also choose an Active File Directory for the local ER/Studio DM1 diagram files in the Options dialog. ER/Studio manages a local 'working copy' of the data model and submits changes you've made to this file to the Repository or, conversely, updates changes others have made from the Repository in order to update your locally managed file. All of this you control through the sophisticated "Review Changes" user interface. It is recommended this Active File directory is on your local machine and not a network location. You will need read/write privileges on this path.

***Note**: If the Repository drop down menu on ER/Studio's Main Menu bar is grayed out, check to see if you have a valid license. You can check this by going to **Help > About.** This will tell you what module you have installed. If you do not see "RepoClient" or it is grayed out then you will need an evaluation extension to trial the software for 14 days. After that a permanent license is required.*

Connecting to the Repository

To connect to the Repository, go to **Repository > Log in...** This will prompt you for a user ID and password. The default login after installation is "Admin" and the default password is "Admin". Both are case sensitive. Once you are connected to the repository, you are now ready to add diagrams.



Working with Diagrams

Adding a Diagram into the Repository

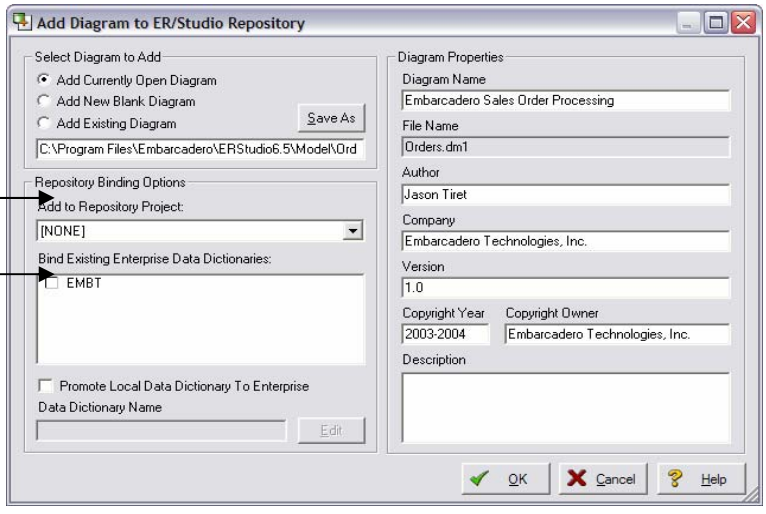
You will need to create a new diagram in ER/Studio before adding it to the Repository. You can do this by going to **File > New,** then select new diagram or you can reverse engineer an existing database. For the purpose of this evaluation guide will open an existing sample diagram in the ERStudio7.0\Models folder.

1. Go to **File > Open**, under the ERStudio7.0\Model folder select "Orders.dm1" and hit **OK**.

2. Once the diagram is opened, go to the **Repository > Diagrams > Add Diagram** selection. You can fill in the appropriate information in the Add Diagram dialog. To configure the diagram you can optionally bind any Enterprise Data Dictionaries and/or assign the diagram to a project. Hit **OK** when done. This will start the process of adding a diagram.

Select Project placement for diagram.

Select Enterprise Data Dictionary to bind to diagram.



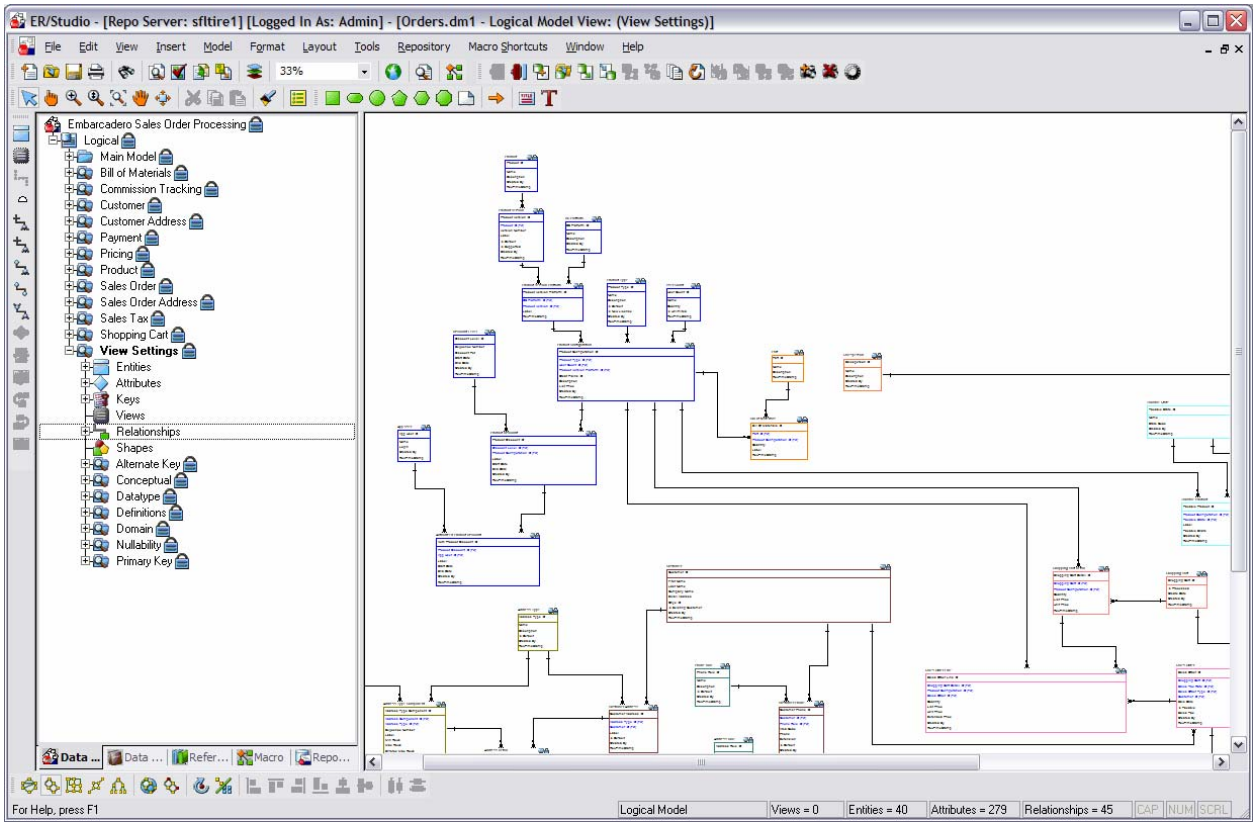
Once the Add Diagram operation is finished, you will see Repository status icons appear on the model objects (explained below). Now added, the diagram is available for any users that can connect to your repository and whom have been provided security.

Repository Status Icons

Once the Orders diagram has been added to the repository, Status Icons will appear on the Explorer Tree and diagram after the diagram is inserted into the repository. These are the 'lock' and 'monitor' icons you see below.

- **Lock icons** indicate real-time status of object meta data in the repository (e.g. attributes, definitions, storage properties, etc)
- **Monitor icons** indicate real-time status of display meta data (e.g. object color, font, etc)

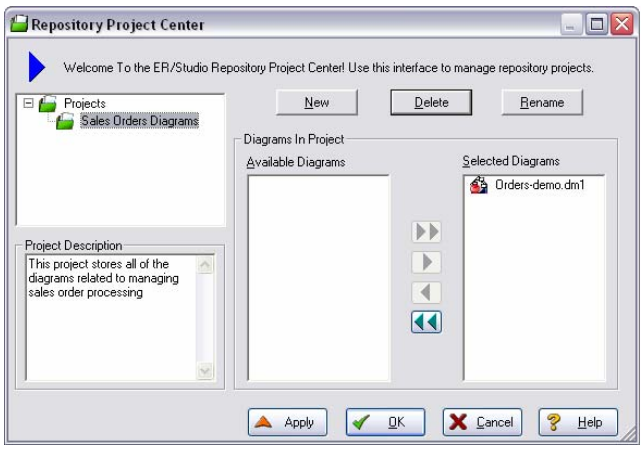
These icons will communicate the check out status of the diagram and the objects in the models to the user. Depending on what type of check out (exclusive vs. non-exclusive) and who has it checked out (you locally vs. others remotely), the status icons will change to provide with a real-time status of exactly who is doing what, and when, to a diagram object. A matrix of these icons is available in Help.



Organizing Diagrams through the Project Center in the Repository

Projects offer a way to organize your diagrams into groups. This organization will be come evident when you or anyone else performs a 'Get Diagram' to view the contents of the Repository while accessing a diagram. Projects are a user-customized way pf partitioning ER/Studio diagrams and Enterprise Data Dictionaries managed in the repository and allows for Security to be supplied at the project level to all Diagrams managed in the Repository such as "No Access". You can organize them by subject matter, e.g., "Sales Diagrams", "HR Diagrams", etc or by the groups who will be working on certain diagrams, e.g., "DBA diagrams", "DA Diagrams", etc.

Projects can be added, edited and deleted by going to **Repository > Project Center**. Let's create a project for the newly added Orders data model.



1. Go to **Repository > Project Center**, hit **New**.

2. Enter a name, "Sales Order Diagrams", and description for the project, hit OK.

3. Move Orders over to the selected diagrams, hit OK.

Note: Nested projects are supported in the repository. You can create a nested project under the "Sales Order Diagrams" by selecting it and clicking **New.** The new project will be under "Sales Order Diagrams."

Checking out diagrams vs. checking out objects

ER/Studio Enterprise's management of diagrams behaves very similarly to Source Code Control systems you may use for document or source code management. The difference is the degree by which ER/Studio can allow for object check out and team collaboration. Each and every element in an ER/Studio diagram can be individually checked out; starting from the entire diagram itself, down to individual elements managed in a Diagram should the user require it (e.g. entities). There are two types of check outs that can be used depending on how you want to work on the diagrams, models and model objects.

1. **Exclusive Checkout** – This is a very restrictive and secure mode and will lock the objects that are checked out within the repository, so that no remote users can work on or more specifically check out the same object at the same time.
2. **(Non-Exclusive) Checkout** – Checking out an object 'normally' will allow multiple team members to work collaboratively on the same elements at the same time. Objects can be simultaneously checked out by two or more users concurrently. Any conflict will be resolved with ER/Studio advanced Review Changes dialog.

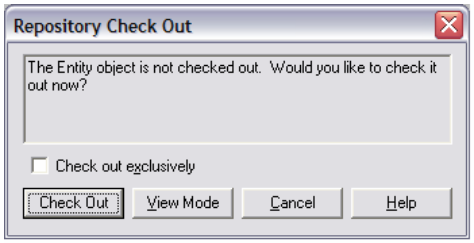
Assume for the remainder of this session that Exclusive Lock Out is not required. Let's look at a selection of Check Out scenarios:

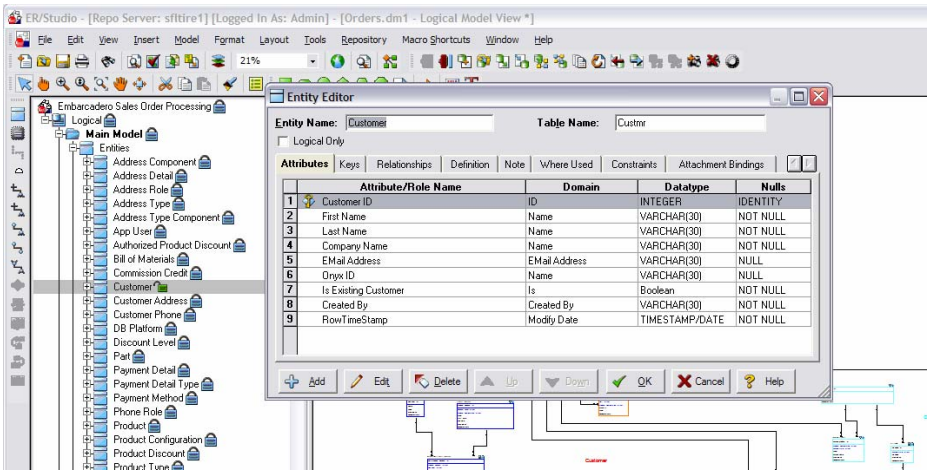
Checking out at the Object Level

You can check out individual objects by right clicking on the object in the explorer tree or double clicking on the object in the diagram.

1. Navigate to the Customer entity in the main model of Orders using the explorer tree.

2. In the diagram, double click on the Customer table. This will prompt you to check out the object. Click 'Check Out'.





3. Once the editor opens, note the change to the status icon in the Explorer browser indicating you have checked out the object locally to begin working.

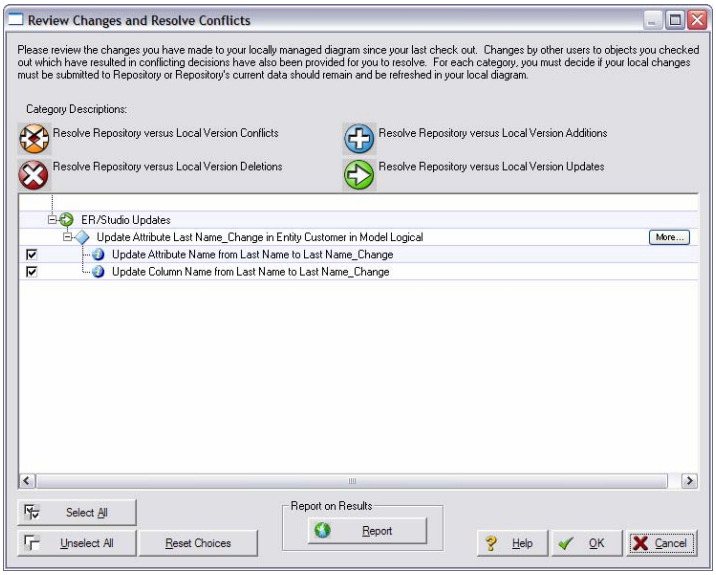
4. Rename the "Last Name" attribute to "Last Name_Change". Hit **OK** to exit the editor.

5. Right Click on Customer and select to “Check in Objects”. This will prompt you for check in comments and notes as well as allow you to review the changes.

6. Select the ‘Review Changes’ checkbox and hit **OK**.

7. You will be presented with a status of what has changed locally to provide you with an opportunity to report, review and possibly reject changes before check in:

8. Leave the selections checked and select **OK**, when finished Reviewing Changes. The information has now been saved to the repository.



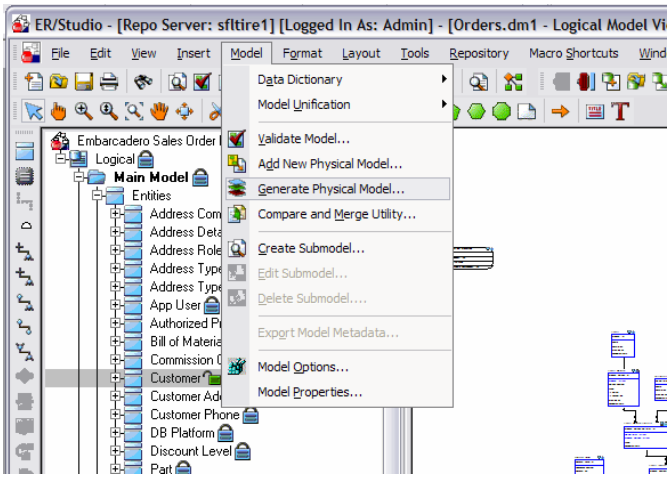
Checking Out at the Diagram Level

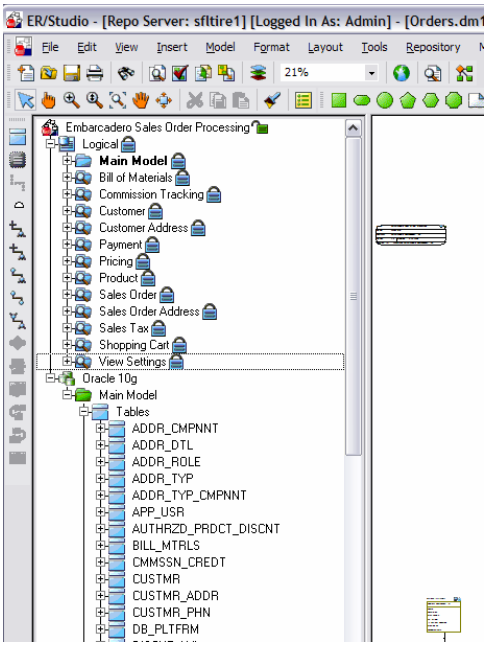
In some cases you may want access to the entire ER/Studio diagram. As an example you may need to derive a new physical model from the logical. In this case you will need to check out the entire diagram. We’ll use this example for our next exercise:

1. Go to **Repository > Diagrams > Check Out Diagram**.

2. Now let’s generate a physical model. Select **Model > Generate Physical Model**:

3. Run through the Generate physical model choosing the options you want. Hit **Finish**. NOTE: You can hit **Finish** at any time through the wizard to take the defaults. Note now an entirely new Physical Model has been generated as seen in the Explorer Browser.





4. Go to **Repository > Diagrams > Check In Diagram.** This will again prompt for any check in comments and if you want to Review Changes.

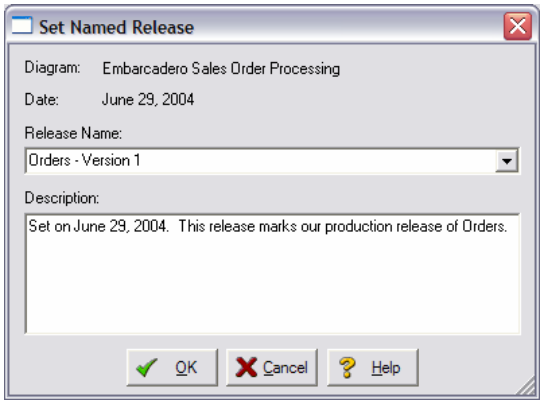
5. Hit **OK**. Your new Physical Model has now been saved to the repository.

Versioning Diagrams

Up until this point we’ve inserted a diagram into the repository and made a few simple changes. Repository will automatically track the

changes to the diagram you and team members are making and apply versions associated with each check in. These versions are located in the version history of each object (accessed through the right click menu). Assume you would like to set 'frozen' baselines of the entire diagram as a mechanism to track milestone releases of the diagram. We can do this by setting a "Named Release." These releases can be used to rollback the diagram back to a previous state if the changes since the last release are not desired.

Setting a Named Release for the "Orders"

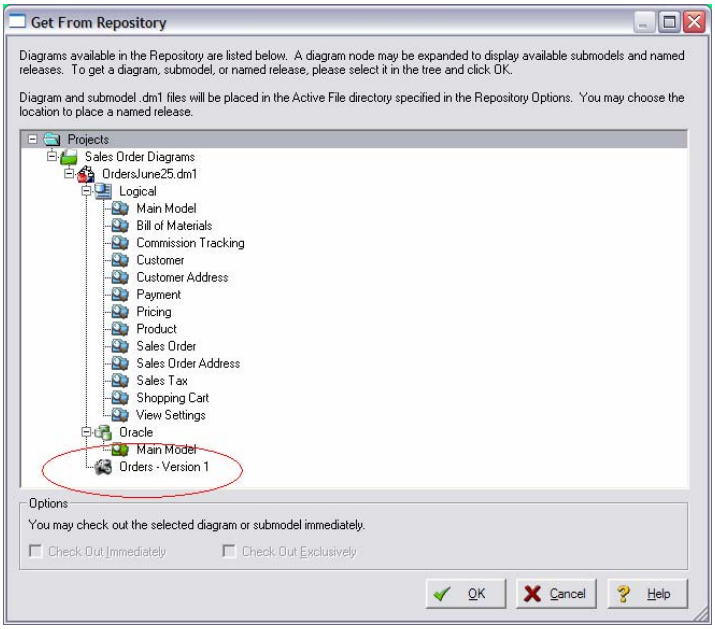


1. Go to **Repository > Releases > Set Named Release**.

2. In the dialog which appears, enter a name, "Orders – Version 1" and any description you choose for the release:

3. Hit **OK**. The release was stored in the repository.

4. At any time we can access the named release from the Get Diagram interface or by going to **Repository > Releases > Get Named Release**. You will see any named releases demarked with a 'camera' icon:



When and if you get a named release, the diagram will appear with 'camera' icons on Diagram objects in lieu of traditional lock icons. This diagram can be used to roll back and replace an existing diagram or compared against the 'active' diagram to individually roll back changes for certain objects you wish.

Please review the "Rollback Diagram" section in help for more information.

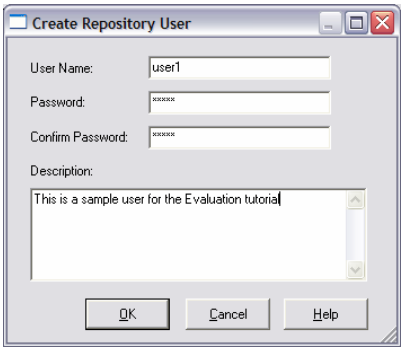
Applying Security to Diagrams through Security Center

Now that diagrams such as Orders and others you'll eventually add to the Repository will be shared across a team, it is a good idea to control who is accessing them. ER/Repository offers a simple to use 'Security Center' for all of these needs. The Security Center will enable you to create users, roles and apply them selectively to the Projects, diagrams, specific models within Diagrams and Data Dictionaries in the Repository. Let's look at each more closely.

Creating a User

Before anyone else can log in and begin using ER/repository, you need to create instances of "Users". To set up individual users, follow these steps:

1. Launch Security Center from the Repository Menu. **Repository > Security > Security Center**.

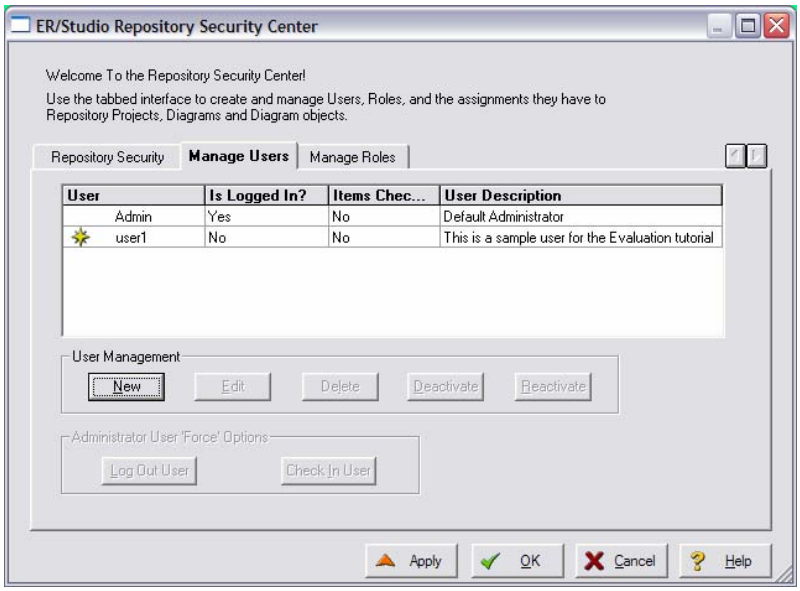


2. Navigate to the **Manage Users** tab.

3. Hit **New**. Supply a name, password and a description for the new user. We'll use "user1" as the name for this sample user.

4. Hit **OK** on the Create Repository User dialog.

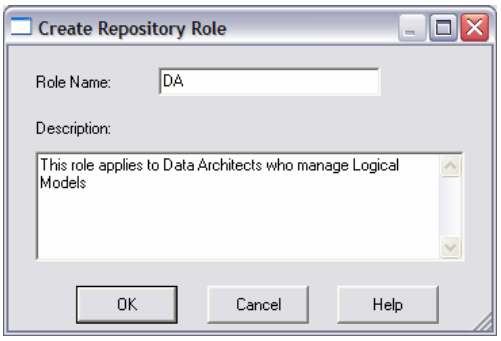
5. Now that you are back in the Security Center, note your "user1" has been created 'locally', but has not yet been submitted back to the ER/Repository. You can see this is the case as there is a star next to the user. Hit "Apply" and this will submit changes to the Repository. The "Apply" feature will allow you send these incremental updates to the repository to continue working in the Security Center.



With "user1" created and submitted to the Repository, let's move on to building a "Role." Keep the Security Center open after step 5 above.

Creating a Role

Now that our User is built, we need to build a "package of permissions" for the user through the Repository's "Roles" to allow/not allow certain activities to be done against things in the Repository. We do this by creating a "Role" under the **Manage Roles** tab. In this example, let's assume we want to create a role for all Data Architects in the organization. Let's assume DA's have permission to create, manage and modify Logical models, but have no rights to modify Physical (DBMS-specific) models.



1. Switch to the **Manage Roles** tab.

2. Hit **New**. Supply a name and description, such as "DA" and provide a description or the role:

3. Hit **OK** and return back to the Security Center. You are in the **Manage Roles** tab. With the initial "DA" Role created, we now need to build a permission set for it.

4. Set permissions for each Repository Object Type…beginning with the Repository itself. This is done under the **Role Permission Details.** As an example, you may not want DAs to access or modify anything in the Security Center, so you may leave these items unchecked as seen here:

5. Continue through the rest of the object types, Project, Diagram, Data Dictionary…etc. As mentioned above, the "DA" will have no rights or privileges to modify Physical models, so leave **Physical Main Model** and **Submodel** entirely unchecked.

6. Hit **Apply** to save the changes to the Repository.

With the "DA" Role created and submitted to the Repository, let's move on to binding the User and Role with specific Diagrams in the Repository. Keep the Security Center open after step 6 above.

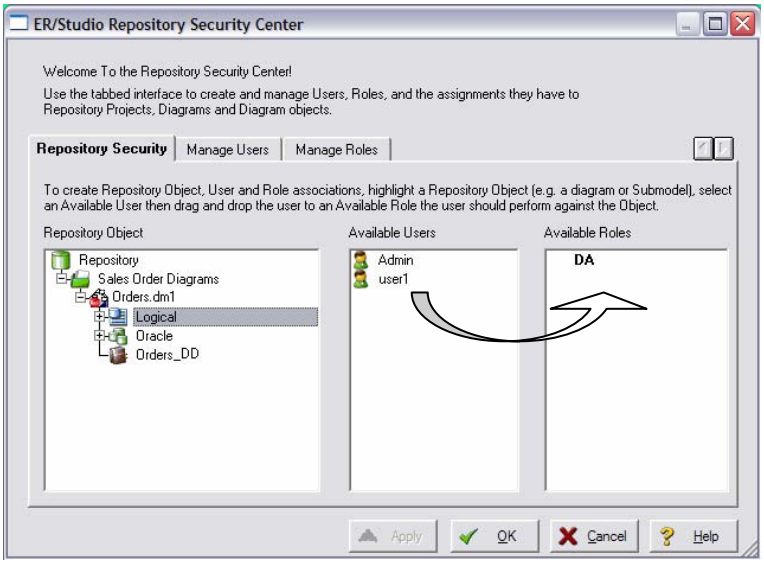
Applying permissions to models, diagrams and dictionaries

Now that we've create a User, "user1", and a Role, "DA", we must now decide what diagrams or parts of diagrams we want to assign (e.g. 'bind') these permissions to our users. This is done under the Security Center's **Repository Security** tab.

1. Switch to the **Repository Security** tab. You will see the Orders diagram within the "Sales Order Diagrams" project you created earlier. Expand this node and select the Logical Model node as you see here:

2. Grab "user1" from **Available Users** and drag him over to "DA" role you created. This will give "user1" the "DA" permissions you set when creating the Role on the Orders logical model and all its Submodels.

3. Further, assuming you want this user bound to the Data Dictionary assigned to Orders, under the Repository Object pane, select the Orders_DD data dictionary.



4. As performed in step 2, grab "user1" again from **Available Users** and drag him over to "DA".

5. Hit **Apply**.

Finished! When "user1" logs into the ER/Repository, she will have only the rights and privileges available that were created in the Role. Do this for other Diagrams you eventually add to the Repository.

Session 8 Conclusion:

At this point we have finished the Repository portion of the evaluation guide. This should have given you a good start to continue working with you diagrams in a collaborative environment. You should now know how to apply security to your diagrams, version your diagrams, check in and out portions of you diagrams and reuse common data elements.

Additional Evaluation Resources

Embarcadero Technologies provides a variety of resources to help support your evaluation and selection of a Data Modeling product for your organization.

Web site

Visit our Web site for current product and company information, educational materials and supporting information. Visit www.embarcadero.com.

To download an evaluation copy of ER/Studio, please visit:
http://www.embarcadero.com/downloads/downloaderstudio.jsp

Electronic Documentation

Detailed reference documentation is available on the ER/Studio Evaluation CD or online at www.embarcadero.com/support

Online FAQ

The ER/Studio online FAQ provides answers to commonly asked questions regarding licensing, installation and other helpful topics.

To review the FAQs for ER/Studio visit:
http://www.embarcadero.com/products/erstudio/erfaq.html

Email Support

You can contact ER/Studio support engineers, consultants and engineers directly by sending inquiries to support@embarcadero.com or log a case through embarcadero.com at:
http://www.embarcadero.com/support/open_case.jsp

Telephone Support

We encourage you to call us anytime you would like help or have questions during your evaluation. Please call 415.834.3131 ext. 2, Monday to Friday, 6:00am - 6:00pm PST, Saturday and Sunday, 8:00am - 5:00 PST.