CA ERwin® Data Modeler

Implementation Guide

Release 9.5.0

ca technologies

This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the “Documentation”) is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

CA Technologies Product References

This document references the following CA Technologies products:

- CA ERwin® Data Modeler Standard Edition
- CA ERwin® Data Modeler Workgroup Edition
- CA ERwin® Data Modeler Navigator Edition

Contact CA Technologies

Understanding your Support

Review support maintenance programs and offerings.

Registering for Support

Access the CA Support online registration site to register for product support.

Accessing Technical Support

For your convenience, CA Technologies provides easy access to "One Stop" support for all editions of CA ERwin Data Modeler, and includes the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

For information about other Home Office, Small Business, and Enterprise CA Technologies products, visit http://ca.com/support.

Provide Feedback

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA Technologies product documentation, complete our short customer survey, which is also available on the CA Support website, found at http://ca.com/docs.

CA ERwin Data Modeler News and Events

Visit www.erwin.com to get up-to-date news, announcements, and events. View video demos and read up on customer success stories and articles by industry experts.

Documentation Changes

The following documentation updates have been made since the last release (r9.0) of this documentation:

- Table of Supported Bridges—Updated with the list of bridges supported for this release.

The following documentation updates have been made since the last release (r8.2) of this documentation:

- Create a Model—Updated this topic per the new design changes in the New Model dialog. It explains different procedures that are available in CA ERwin Data Modeler Version 9 to create a model.
- Apply Naming Standards (see page 38)—Updated this topic per the design changes and application of Naming Standards.
- Apply Data Type Standards (see page 51)—Updated this topic per the design changes and application of Data Type Standards.
- Table of Supported Bridges—Removed MIR XMI format export and import bridges.

Contents

Chapter 5: Naming and Datatype Standards 37

Appendix A: Supported Metadata Integration Bridges 59

Index 65

Chapter 1: Overview

This section contains the following topics:

Database Design Overview

Welcome to CA ERwin Data Modeler (CA ERwin DM), the database design tool that raises the level of data quality in transactional and data warehouse systems. It provides the tools to design and implement databases for transactional business, E-commerce, and data warehousing applications.

You can create and maintain graphical models that represent databases, data warehouses, and enterprise data models. CA ERwin DM provides a modeling platform where corporate data requirements and related database designs can be defined, managed, and implemented across a wide variety of database platforms.

A Windows-based graphical user interface is combined with powerful entity-relationship (ER) diagramming tools, custom editors to define physical database objects, a Model Explorer for a text-based view of model objects, and support for leading SQL and desktop databases.

You can also use CA ERwin DM to streamline the application development process, by allowing different groups (Standards Administrators, Business Analysts, Data Modelers, and so on) to perform work independently, while collaborating and synchronizing. In this way, different groups can simultaneously work on various parts of a model or different model types.

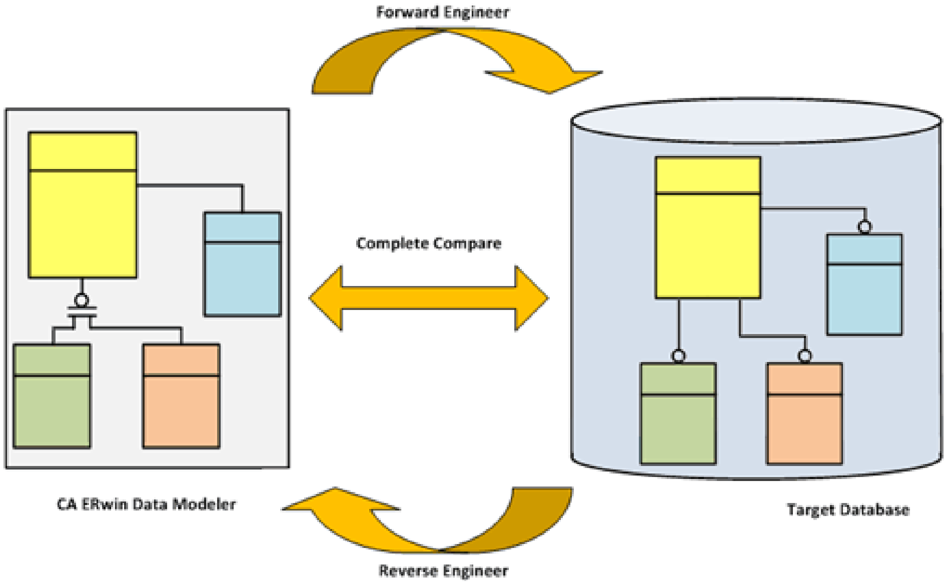
Overview

CA ERwin DM combines a Windows-based graphical user interface with entity-relationship (ER) diagramming tools, and numerous innovative features. These features allow you to easily create and maintain your relational database and the logical and physical models that describe it. CA ERwin DM works by providing a design solution that helps you create a visual blueprint (a data model) for your organization.

CA ERwin DM is much more than a drawing tool. It not only helps you design a logical data model, which captures business rules and requirements, but it also supports the design of a corresponding physical data model for your target server. This enables you to forward engineer this physical data model and automatically generate physical database structures to your system catalog.

CA ERwin DM supports reverse engineering of existing databases and provides both a physical and logical/physical data model so you can maintain an existing database, or migrate from your current target server to a different one.

The Complete Compare technology automates model and database synchronization by letting you compare the model with the database, displaying, and analyzing the differences. This enables you to selectively move the differences into the model or generate them into the database.



The Application Development Cycle

Most data modeling solutions begin with the gathering of business rules and creating logical constructs and continues to the physical design phase followed by the implementation of a database that supports one or more applications. CA ERwin DM supports this iterative process by supporting multiple platforms, reuse of objects, and the ability to synchronize changes between data models across the enterprise.

Chapter 2: Basic Concepts and Features

This section contains the following topics:

Enhance Your Data Model

When you design a data model, you can use many of the drawing and display features to enhance its appearance and make it easier to view and understand.

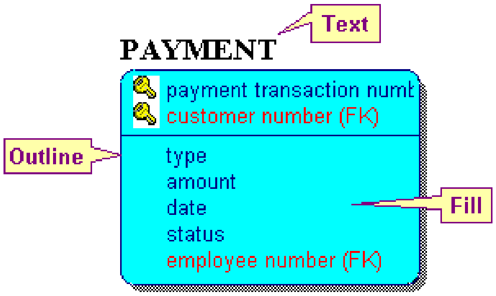
To prepare a diagram for analysis and presentation, you can:

- Enlarge the font size for all entity names so that they are easier to read.
- Change the color of all foreign keys to red, so that they are distinguishable from other attributes or columns.
- Use different fonts and colors to identify new objects from old objects in your model.
- Resize one or more entity boxes
- Align two or more entity boxes

Graphic Features

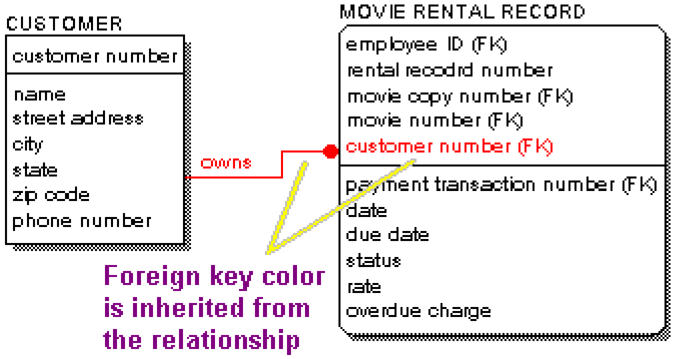
A diagram has several types of graphic elements that you can enhance with color. For example, you can individually specify text, fill, and line sizes and colors for entities, attributes and relationships, or use themes to define the default display of certain objects by diagram, or by model.

In addition to these graphic elements, there are many object types that you can change globally such as Foreign Keys, Entity Names, and Owned Keys.



Objects Can Inherit Color

The color of an object can be related to the color it inherits from other objects in the diagram. For example, you can choose to have a foreign key attribute or column, inherit its font and color from its parent primary key or relationship.



Foreign key color is inherited from the relationship line

You can also change the color of foreign key columns or attributes so that you can easily identify them from owned primary key columns or attributes.

The Theme Editor

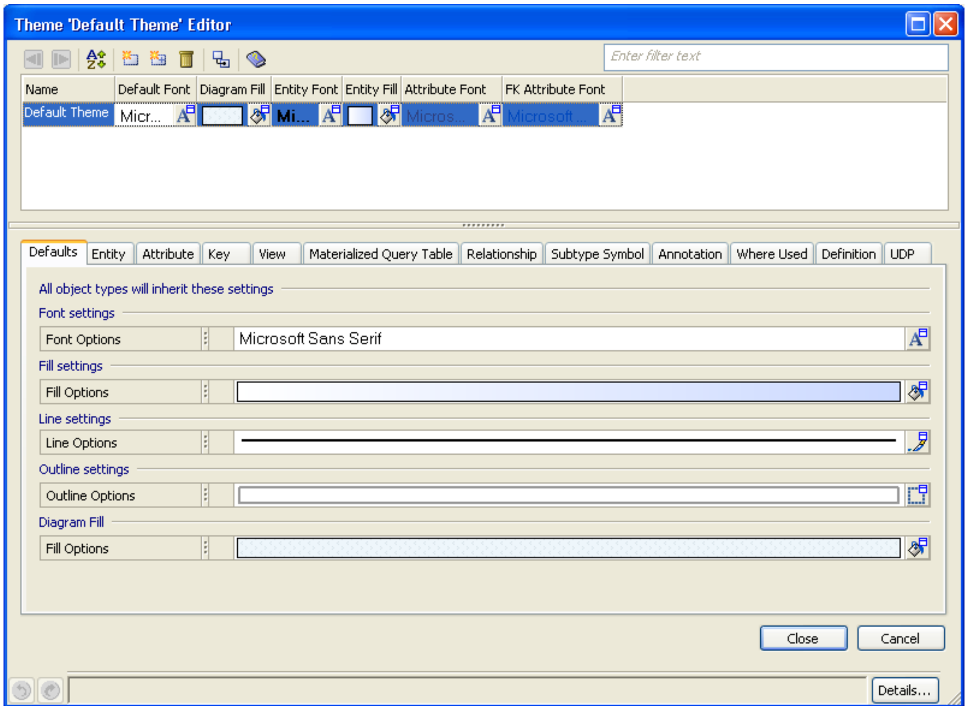
You use options in the Theme Editor to set formatting and display options for objects in your model. You can create a custom Theme and apply it to selected diagrams on your model, or use the Default Theme supplied with the program.

You can open the Theme Editor in any of these ways:

- From the General Tab on the Diagram Editor
- When you click Themes on the View menu
- From the Model Explorer

CA ERwin Data Modeler ships with two themes—Default and Classic. These cannot be edited, but you can create new themes.

You can override settings defined in a Theme, in the Style tab of the individual editors.



Name	Default Font	Diagram Fill	Entity Font	Entity Fill	Attribute Font	FK Attribute Font
Default Theme	Micr...		Mi...		Micros...	Microsoft...

All object types will inherit these settings

Font settings

Font Options: Microsoft Sans Serif

Fill settings

Fill Options

Line settings

Line Options

Outline settings

Outline Options

Diagram Fill

Fill Options

Formatting Options in the Theme Editor

You use the Theme Editor to define the default theme for various ERwin objects. The Theme Editor displays tabs for each object and it depends on the model type that you select. In a logical/physical model, you can switch from the logical model to the physical model to change the settings for each side of the model.

The Theme Editor has the following options:

Name

Specifies the name of the Theme.

Default Font

Specifies the default font for the selected Theme. Click A to change the font style options.

Diagram Fill

Specifies the default fill color for the diagrams in the selected Theme. Click 🪣 to change the fill style options.

Entity Font

Specifies the default font for the entities in the selected Theme in a logical model. Click A to change the font style options.

Table Font

Specifies the default font for the tables in the selected Theme in a physical model. Click A to change the font style options.

Entity Fill

Specifies the default fill color for the entities in the selected Theme in a logical model. Click 🪣 to change the fill style options.

Table Fill

Specifies the default fill color for the tables in the selected Theme in a physical model. Click 🪣 to change the fill style options.

Attribute Font

Specifies the default font for the attributes in the selected Theme in a logical model. Click A to change the font style options.

Column Font

Specifies the default font for the columns in the selected Theme in a physical model. Click A to change the font style options.

FK Attribute Font

Specifies the default font for the foreign key attributes in the selected Theme in a logical model. Click A to change the font style options.

FK Column Font

Specifies the default font for the foreign key columns in the selected Theme in a physical model. Click A to change the font style options.

FK Attribute Color

Specifies the default color for the foreign key attributes in the selected Theme in a logical model. Click to change the color options.

FK Column Color

Specifies the default color for the foreign key columns in the selected Theme in a physical model. Click to change the color options.

The Theme Editor includes the following tabs:

Defaults

Define the font and color for the background window and the overall default for objects on the diagram window.

Entity

Define font, fill, and outline options for entity names and definitions in a logical model.

Attribute

Define fonts and colors options for owned and foreign key attributes in a logical model.

Table

Define font, fill, and outline options for table names and definitions in a physical model.

Column

Define fonts and colors options for owned and foreign key columns in a physical model.

Key

Define the key display preference, the Alternate Key (AK) Attribute font, and the Inversion Entry (IE) Attribute font.

View

Define the font, fill, and outline options for the name and definition for a view. Define display fonts for an owned attribute and expression.

Materialized Query Table (DB2 only)

Define the font, fill, and outline options for the name and definition for a materialized query table. Define display fonts for an owned attribute, expression, Alternate Key (AK) Attribute, and Inversion Entry (IE) Attribute.

Relationship

Define style inheritance options, and the font, color, and line style for relationships.

Subtype Symbol

Define style inheritance options, and the font, color, and line style for subtypes.

Annotation/Shape

Define formatting options such as font, fill, line, and outline for annotations/shapes.

Where Used

Displays what is affected by the selected Theme. The display can include a diagram name, subject areas, individual modeling objects, and drawing objects.

Definition

Lets you enter definition text to associate with the selected Theme.

UDP

Lets you create a UDP to associate with the selected Theme.

Create a New Theme

You use the Theme Editor to create your own theme that sets the preferences for displaying objects in your model. When you create a new theme, you can choose to apply it to any diagram by selecting this theme from the Theme Editor when you are working in that diagram.

You use the Defaults tab to set the preferences for font and fill colors for all objects in your model. When you select a Theme for your model, all objects in your model inherit these values.

Note: The options and tabs change depending on the model type (logical or physical) with which you are working.

To create a new theme

1. Click Themes on the View menu.

   The Theme Editor opens.

2. Click New on the toolbar to create a new theme and work with the following options:

   Name

   Specifies the name of the theme. Enter the theme name in this field.

   Default Font

   Specifies the default font. Click to define the font style.

   Diagram Fill

   Specifies the default fill color for diagrams. Click to define the fill style.

   Entity (or Table) Font

   Specifies the default font for the entities (or tables). Click to define the font style.

   Entity (or Table) Fill

   Specifies the default fill color for the entities (or tables). Click to define the fill style.

   Attribute (or Column) Font

   Specifies the default font for the attributes (or columns). Click to define the font style.

   FK Attribute (or Column) Font

   Specifies the default font for the foreign key attributes or columns. Click to define the font style.

   FK Attribute (or Column) Color

Specifies the default color for the foreign key attributes or columns. Click to define the color.

3. Click the Defaults tab and work with the following options:

   Font Settings

   Specifies the default font. Click to define the general font style.

   Fill Settings

   Specifies the default color for the objects in the diagram. Click to define the general fill style.

   Line Settings

   Specifies the default line style. Click to define the general line style.

   Outline Settings

   Specifies the default outline style for the model. Click to define the general outline style.

   Diagram Fill

   Specifies the default background color for the diagrams in your model. Click to define the general fill style.

4. Work with the options on the other tabs to define display options for specific objects, such as views, relationships, and subtype symbols, or define UDPs or theme definition.

5. Click Close.

   The theme is created and the Theme Editor closes.

Set Font and Fill Preferences for Objects

You use the Defaults tab to set the default preferences for font and fill colors for all objects in your model. When you select a Theme for your model, all objects in your model inherit these values.

To set the default font and fill preferences for objects

1. Click Themes on the View menu.

   The Theme Editor opens.

2. Select the Theme for which you want to set the default values, and click the Defaults tab.

3. Work with the following options:

   Font Settings

   Specifies the default font. Click to change the general font style options.

   Fill Settings

   Specifies the default color for the objects in the diagram. Click to change the general fill style options.

   Line Settings

   Specifies the default line style. Click to change the general line style options.

   Outline Settings

   Specifies the default outline style for the model. Click to change the general outline style options.

   Diagram Fill

   Specifies the default background color for the diagrams in your model. Click to change the general fill style options.

4. Click Close.

   The changes are saved and the Theme Editor closes.

Using a Model to Import Styles

You can maintain and share global formatting templates by creating a Theme in one model and importing it into another model. This process uses the CA ERwin Data Modeler Link Model Source wizard to import the Theme into a second model.

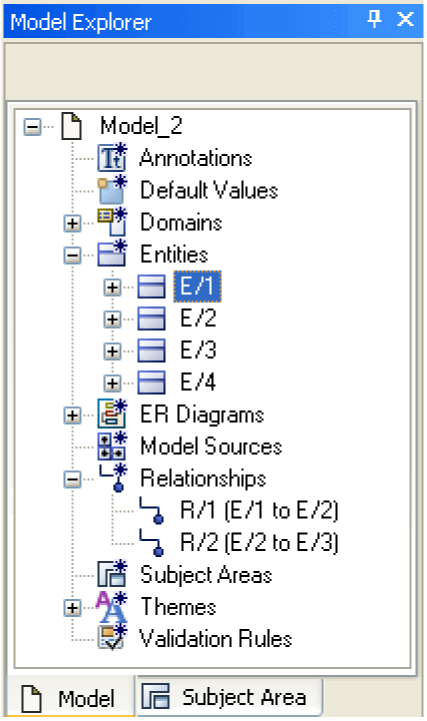
Use this process to maintain a global style template and import a Theme into another model:

- Create a new source model to contain the Theme you want to use as a template for other models.
- Use the Theme editor to create a new Theme and define your formatting styles. Save your source model.
- Open the model in which you want to import the Theme.
- Click Tools, Link Model Source to open the Link Model Source wizard. Select your global style template model as the source model and click Finish. It is not necessary to change any of the other defaults in the wizard.
- The objects from the source model are ready for import into your open model. The Complete Compare Resolve Differences dialog opens, where you can review and import the Theme to your target model.
- Click Finish.

Chapter 3: Model Explorer

The Model Explorer provides an organized, hierarchical, text-based view of your data model and its contents. It provides an easy method for creating objects. The Model Explorer lets you to create, display, navigate, and modify your model using the Model or Subject Area view.

The Model Explorer has two panes, Model and Subject Area. You can switch to a different pane by clicking the tab at the bottom of the Model Explorer:



All of the objects in your model are listed in the Model Explorer, but display differently based on which pane is active.

- The Model pane includes every object in your model including subject areas and domains. This is the default view.
- The Subject Areas pane displays model objects sorted by subject area.

Dynamic Changes

Just as you can work on a graphical view of a model in the Diagram Window, you can perform the same tasks from the Model Explorer using a text-based view of a model. When you make changes to an object in the Model Explorer, the graphical view of the model is immediately updated with the same change.

For example, if you rename a table in the Model Explorer, the new table name replaces the existing table name in the Diagram Window and the relevant editors. Similarly, if you make a change to the diagram or in an editor, you immediately see the change in the Model Explorer.

New Objects

Just as you can create a new entity, table, or relationship using the Toolbox, you can quickly create a new object in the Model Explorer.

To create a new object, right-click on any object and choose New from the shortcut menu. For some objects, such as a domain, you can drag the object from the Model Explorer onto the Diagram Window to create a new attribute or column.

Rename Objects

Every object in your model is represented in the Model Explorer by its name. Just as you can rename an object using on-diagram editing in the Diagram Window, you can rename an object directly in the Model Explorer.

To rename objects

1. Click the object name, and then click again.

   An edit box opens around the object name.

2. Enter the new object name directly in the edit box.

   The new name of that object appears throughout the Model Explorer and the model.

Note: Any unique name rules that you select for your model are enforced when you rename an object in the Model Explorer.

Select Unique Name Rules

Unique name rules are a set of options that define how to respond when you have a duplicate entity or table name in the data model.

To select unique name rules

1. Click Model Naming Options on the Actions menu.

   The Model Naming Options dialog opens.

2. Click the Duplicate Names tab and set preferences for duplicate names.
3. Click OK.

   Your changes are saved and the Model Naming Options dialog closes.

The Go to Diagram Option

When you right-click an entity, table, or view in the Model Explorer and click the Go To Diagram option from the shortcut menu, you can quickly display that object in the Diagram Window.

Object Properties

Each object in a data model has many properties. The easiest way to add or modify object properties is in the object property editor.

Using the Model Explorer, you can open the property editor for that object type. You can do this in one of two ways:

- Right-click on the object and click Properties.
- Select the object and click the Property Editor button from the Model Explorer toolbar.

When the Property Editor opens, you can accept or change the default properties. When you close the editor, the properties are saved with the object.

Note: For bulk editing of object properties, you can use the Bulk Editor.

Regardless of the method you choose to add or edit an object or its properties, the Model Explorer and the data model are always in sync. While you are still in the design phase, the names of objects in your model may change frequently.

Note: Object properties also migrate through relationship lines. In other words, objects can inherit properties from their parent objects.

Object Moving, Copying, and Deleting

Just as you can move, copy, and delete attributes and columns in the Diagram Window, you can also move, copy, and delete most objects in the Model Explorer.

After you move, copy, or delete an object, all of the instances in the model that reference that object are automatically updated.

Add a Domain in the Model Explorer

In the Model Explorer, the Model view lists all of the domains for the current model, which include all of the default domains as well as any that you created.

To add a domain in the Model Explorer

1. Right-click a domain type (blob, string, number, and so on) and click New.

   An edit box opens with a default domain name.

   Note: The selected domain type is assigned as the parent.

2. Enter a name for the domain in the text box and press Enter.

   The new domain appears according to the sort order that you specify (either alphabetically or hierarchically). To switch the sort order of the Domains from hierarchical to alphabetical, right-click the Domains folder in the Model view, and select the sort option that you prefer.

Note: For more information about domains, see Working with Data Models.

Add a Model Source in the Model Explorer

When you work with design layers, it may be important to link a model with its model source so that you can identify the historical parent of a model. After you link a data model to its source, you can synchronize the two models to selectively apply changes to one or the other.

To add a model source in the Model Explorer

1. Right-click the Model Sources Folder and click New on the shortcut menu.

   The Add Model Source wizard opens.

2. Follow the steps in the Add Model Source wizard to finish adding the model source.

3. Click Add to complete the process and close the wizard.

   The source objects are added to your target model, and models are linked. As you continue to work with the models, all the changes you apply to the model and the model source are tracked.

Note: For more information about adding model sources, see Working with Design Layers.

Model View

The primary purpose of the Model view is model exploration and to function as an alternative launching point for object and property editors. It lists all of the object types that appear in the current data model based on the model type and the target server.

For example, if the model type is Logical, the Model Explorer does not include physical objects such as a view. Similarly, if the model type is Physical, the Model Explorer does not include logical objects. If the target server is Oracle, then additional objects specific to Oracle display, such as Clusters.

Other objects that do not display are:

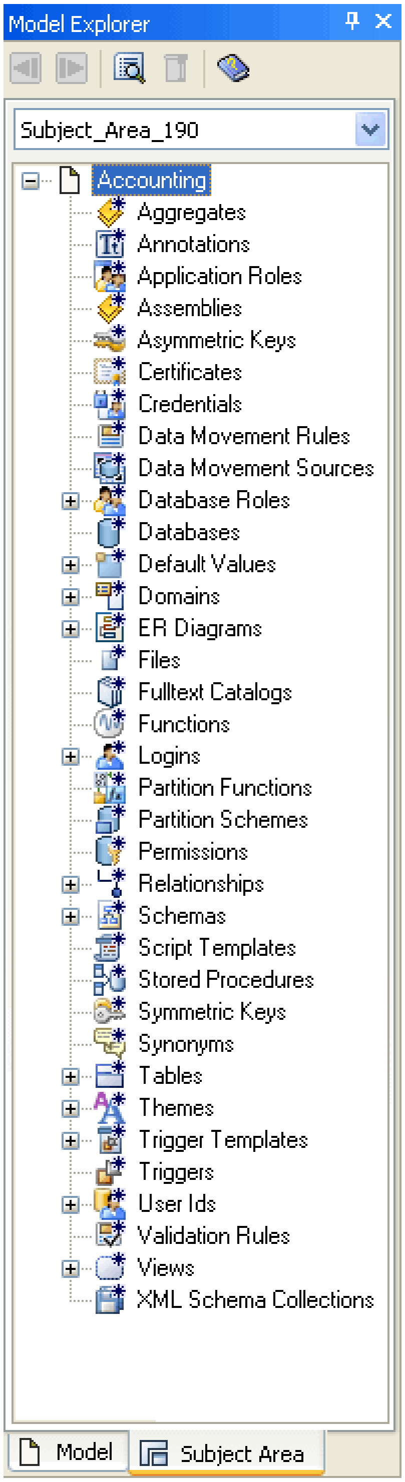
- Minor objects that are handled completely by the editors of the parent object. For example, a Valid Value object is created, deleted, and edited by the Validation Rule Editor.
- An object that represents configuration information that is stored by a CA ERwin Data Modeler process, and is not actual model data, such as a Complete Compare Option Set.

Subject Area View

The Subject Area pane displays model objects sorted by subject area. You can expand each subject area to see a list of the members as well as any stored displays, which appear in folders below the Subject Area to which they belong.

The purpose of the Subject Area view is to provide a filtered view of the model so that unnecessary tree information is minimized. You can create subject areas to divide the model into smaller manageable parts, which is especially helpful when you are working with large models.

At the top of the Model Explorer is a drop-down list where you can select the subject area that you need. When you select the subject area, the contents of the tree control in the Model Explorer filters so that only those objects that are contained within the selected subject area are shown.


Model Explorer
Subject_Area_190
Accounting
Aggregates
Annotations
Application Roles
Assemblies
Asymmetric Keys
Certificates
Credentials
Data Movement Rules
Data Movement Sources
Database Roles
Databases
Default Values
Domains
ER Diagrams
Files
Fulltext Catalogs
Functions
Logins
Partition Functions
Partition Schemes
Permissions
Relationships
Schemas
Script Templates
Stored Procedures
Symmetric Keys
Synonyms
Tables
Themes
Trigger Templates
Triggers
User Ids
Validation Rules
Views
XML Schema Collections
Model
Subject Area

For example, you see that tables are categorized as tables, and views categorized as views, unlike the display of subject area objects in the Model tab view. Relationships are filtered to show only those visible on the diagram if both endpoints are present in the subject area. You can expand the tree for the subject area to see the subject area members.

Note: Global objects that do not relate to the selected subject area are not displayed, such as subject area objects or model source objects.

Create a New Subject Area

You can create a new subject area using the Subject Area Editor.

To create a new subject area

1. Click Subject Areas on the Model menu.

   The Subject Area Editor opens.

2. Click the New button.

   A new subject area with the default name is added to the Navigation Grid. You can change the name directly in the Name cell.

3. Work with the following options:

   Name

   Displays the subject area name. You can change the name directly in this field.

   Auto-populate

   Specifies to automatically populate the selected subject area with all model objects.

4. Click the General tab and specify common properties for the selected subject area.
5. Click the Members tab and specify the members of the selected subject area.
6. (Optional) Click the Definition tab to enter definition text.
7. (Optional) Click the UDP tab to add subject area user-defined properties.
8. Click Close.

   A new subject area is created and the Subject Area Editor closes.

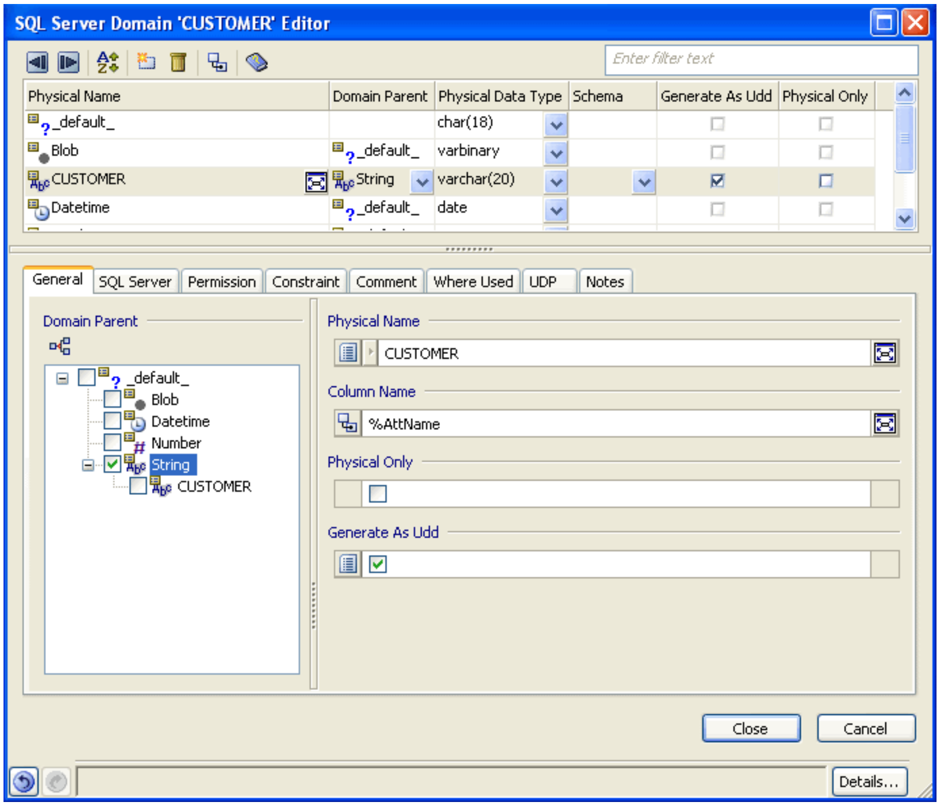
Chapter 4: Domains

A *domain* is a model object that you can use to assign properties quickly to an attribute or column. By using domains, you promote consistency because you can reuse a domain as many times as you like in a single or multiple data models.

Domains also reduce the time spent on development and maintenance. If you change the domain, all attributes or columns associated with the domain also change.

Domain Editor

The Domain Editor lists all the domains for the data model. You can arrange them hierarchically or alphabetically. The Model Explorer has the same list as the Domain Editor and can also be sorted in hierarchical or alphabetic order.



Physical Name	Domain Parent	Physical Data Type	Schema	Generate As Udd	Physical Only
default		char(18)		☐	☐
Blob	_default_	varbinary		☐	☐
CUSTOMER	String	varchar(20)		☑	☐
Datetime	_default_	date		☐	☐

Domain Parent

- ☐ _default_
  - ☐ Blob
  - ☐ Datetime
  - ☐ Number
  - ☑ String
    - ☐ CUSTOMER

Physical Name: CUSTOMER

Column Name: %AttName

Physical Only: ☐

Generate As Udd: ☑

You can create and modify both physical and logical domains using the Domain Editor. Some of the domain properties include:

- Domain name and column name
- Column data type, permission, default value, and valid value
- Domain comment or note and column comment or name
- User defined property
- User notes

Note: The tabs and options in the editor change based on whether the model is logical, physical, or logical/physical. You also cannot change the properties of the standard domains (*<default>*, *Blob*, *Datetime*, *Number*, and *String*).

Inherited and Non-inherited Domain Properties

Domains have two types of properties:

- *Non-inheritable properties* do not migrate to child domains or attributes and columns associated with the domain because they are properties of the domain itself.
- *Inheritable properties* do migrate to child domains and to the attributes and columns associated with the domain.

The inheritable and non-inheritable domain name properties appear on the General tab. They also appear on the target database tab (for example, SQL Server) in a physical model.

Attributes and Columns in Domains

When you drag-and-drop a domain from the Model Explorer onto an entity or table, you can add a new attribute to an entity or a new column to a table. The new attribute or column inherits all the inheritable properties, including its name, from the domain.

Drag a Domain from the Model Explorer to Your Model

Use drag-and-drop to add a domain directly from the Model Explorer to an entity or table in your model.

To drag a domain from the Model Explorer to your model

1. Expand the list of domains in the Model Explorer and click the domain that you want to add to your model.
2. Drag and drop the domain onto the entity or table in your model.

   The domain is added as an attribute or column.

   Note: Depending on the Unique Naming rules that apply to the data model, you may be prompted to rename each attribute or column that you add because the domain's name is automatically assigned to each new attribute or column.

Creating New Domains

You can work with features in the Domain Editor to create a new domain in addition to the default domains supplied with CA ERwin Data Modeler. You can use the Domain Editor to create a domain while you are in a logical model (*logical edit mode*) or a physical model (*physical edit mode*), but the editor's options change depending on the edit mode.

Add a Domain in the Logical Edit Mode

The Domain Editor lets you define properties for columns; you can create a domain and attach it to any number of columns instead of defining properties for each column individually. You can work with features in the Domain Editor to add a domain to the logical model.

To add a domain in the logical edit mode

1. Click Domains on the Model menu.

   The Domain Editor opens.

2. Click the New button on the toolbar.

   A new domain with a default name is added to the Navigation Grid. It is placed in the list in alphabetical order.

3. Work with the following options in the Navigation Grid:

   Name

   Displays the name of the domain in the logical model. When you add a new domain, a default name is automatically assigned. You can change the domain name in this field using in-place editing.

   Note: Duplicate domain names are not allowed.

   Domain Parent

   Displays the current parent domain. You can change the domain using the drop-down list.

   Logical Data Type

   Specifies the logical data type for the domain. You can change the data type using the drop-down list. The drop-down list contains the data types related to the current domain.

   Logical Only

   Specifies whether the domain should only appear in the logical model. Select this check box if you want the selected domain to appear in the logical model only. Clear this check box if you want the selected domain in the physical model. If you are working with a logical-only model and select this check box, when you derive a new physical-only model, the selected domain does not appear in the new model.

   Note: When you select or clear the check box in the Navigation Grid, your selection is automatically reflected in the Logical Only check box in the General tab.

4. Work with the other tabs in the editor to further define the domain properties.

5. Click Close.

   The domain is added to the logical model and the Domain Editor closes.

Add a Domain in the Physical Edit Mode

You can work with features in the Domain Editor to add a domain to the physical model.

To add a domain in the physical edit mode

1. Click Domains on the Model menu.

   The Domain Editor opens.

2. Click the New button on the toolbar.

   A new domain with a default name is added to the Navigation Grid. It is placed in the list in alphabetical order.

3. Work with the following options in the Navigation Grid:

   Physical Name

   > Displays the name of the domain in the physical model. When you add a new domain, a default name is automatically assigned. You can change the domain name in this field using in-place editing.

   Domain Parent

   > Displays the current parent domain. You can change the domain using the drop-down list.

   Physical Data Type

   > Specifies the physical data type for the domain. You can change the data type using the drop-down list. The drop-down list contains the data types related to the current domain.

   Anchored to Type

   > Specifies whether the domain is anchored to a column or a variable. This column is enabled only for a DB2 LUW user defined data type.

   Schema

   > Specifies the schema associated with the domain.

   Distinct Type

   > Specifies whether the domain is of distinct type.

Physical Only

Specifies whether the domain should only appear in the physical model. Select this check box if you want the selected domain to appear in the physical model only. Clear this check box if you want the selected domain in the logical model.

Note: When you select or clear the check box in the Navigation Grid, your selection is automatically reflected in the Physical Only check box in the General tab.

4. Work with the other tabs in the editor to further define the domain properties.
5. Click Close.

   The domain is added to the physical model and the Domain Editor closes.

Assign a Data Type to a Domain

A *data type* is a domain property and defines the physical properties of a column in a database such as length (number of characters), type (alpha or numeric), and precision (decimal). You can also define a logical data type, which usually corresponds to a physical data type.

By default, a new domain is assigned the same data type as its parent domain. You can change it at any time. The available data types for the current model always appear in the drop-down selection in fields where the data type is defined in the Domain Editor.

To assign a data type to a domain

1. Click Domains on the Model menu.

   The Domain Editor opens.

2. Select the domain in the Navigation Grid for which you want to assign a data type. Use the Enter filter text box to filter a large list to quickly locate the domain.
3. Select a new data type for the domain using the drop-down list in the Logical Data Type column (*logical edit mode*) or the Physical Data Type column (*physical edit mode*) in the Navigation Grid.
4. Click Close.

   The data type is assigned to the domain and the Domain Editor closes.

Domain Icons

In the Model Explorer and in the Domain Editor, a unique icon represents each domain. A default icon is used for new domains, but you can choose a different icon to represent the domains you create.

Assign or Change a Domain Icon

You can assign any image (.bmp file) as the icon for the non-inheritable property of the domain itself. You can assign the same image or select a different image that the attributes or columns associated with the domain can inherit.

Note: You can change the domain icon only in the logical model.

To assign or change the domain icon

1. Click Domains on the Model menu.

   The Domain Editor opens.

2. Select the domain in the Navigation Grid that you want to work with and click the General tab.
3. Specify the image to use in the Image drop-down or click the New button to open the Image Editor and create a new one.
4. Click Close.

   The icon is assigned to the selected domain.

Chapter 5: Naming and Datatype Standards

This section contains the following topics:

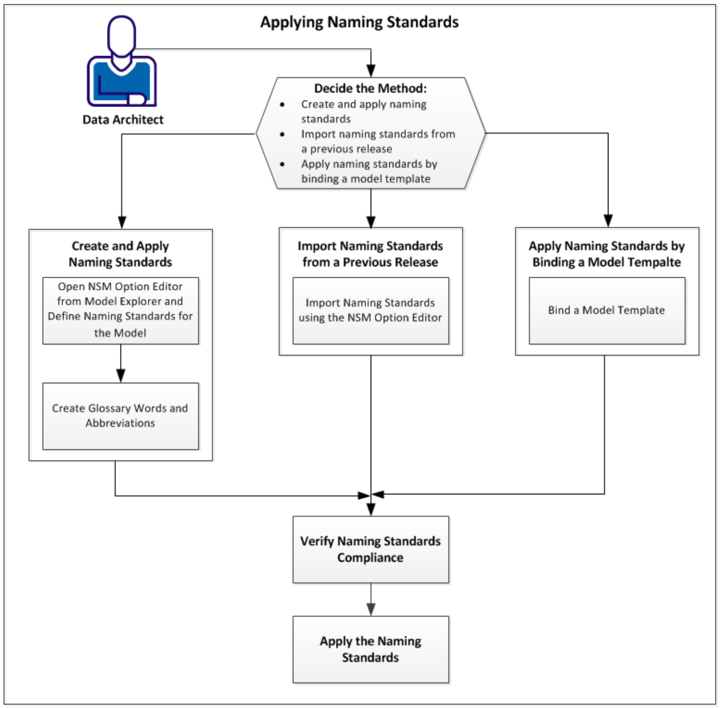
Applying Naming Standards

You create naming standards so that everyone in your organization uses names correctly and consistently. Use NSM Options Editor to create naming standards and develop a glossary that contains glossary words, their abbreviations, and alternate abbreviations.

You can create multiple NSM Option objects and apply one of them to a model. Each NSM Option object includes the naming standards for logical and physical model objects, and the glossary. The components of logical and physical model names are prime, modifier, or class. To define naming standards for entities, tables, attributes, columns, and domains you specify the components and the order in which they are to be presented.

If you do not use naming standards, you may have modeling design problems that produce errors when you run queries to extract sets of data.

You can apply naming standards in three ways, as illustrated in the following diagram:


Applying Naming Standards
Data Architect
Decide the Method:
Create and apply naming standards
Import naming standards from a previous release
Apply naming standards by binding a model template
Create and Apply Naming Standards
Open NSM Option Editor from Model Explorer and Define Naming Standards for the Model
Create Glossary Words and Abbreviations
Import Naming Standards from a Previous Release
Import Naming Standards using the NSM Option Editor
Apply Naming Standards by Binding a Model Tempalte
Bind a Model Template
Verify Naming Standards Compliance
Apply the Naming Standards

Follow these steps apply naming standards:

1. Decide the method. (see page 39)
2. Verify naming standards compliance. (see page 46)
3. Apply the naming standards. (see page 47)

Decide the Method

You can apply naming standards in the following ways:

- Create and apply: Use this option when you are creating the naming standards for the first time.
- Import from a previous version: Use this option if you have created naming standards using a previous version of CA ERwin DM.
- Bind a model template: Use this option if you want to use the naming standards that you have already created and attached to a model or a model template.

Create and Apply Naming Standards

You create naming standards if you are implementing standards for the first time.

Open NSM Option Editor from Model Explorer and Define Naming Standards for a Logical Model

You can specify naming standards for logical names that are different from the naming standards for physical names. Use the Logical tab in NSM Option Editor to apply naming standards to entities, attributes, and domains and to define naming standards by specifying the order in which parts of names are to appear in the logical name.

Follow these steps:

1. Expand the Model Explorer, right-click NSM Options and select New.

   A new NSM Option object is added.

2. Right-click the new object and select Properties.

   NSM Option Editor appears.

3. Go to the Logical tab.
4. Select the part of the name whether prime, modifier1, modifier2, or class that you want to include first in the logical name for the Entity, Attribute, and Domain rows in each of the Part1, Part2, Part3, and Part4 rows.
5. Click Close.

   The data is saved.

Open NSM Option Editor from Model Explorer and Define Naming Standards for a Physical Model

You can specify naming standards for physical names that are different from the naming standards for logical names. Use the Physical tab in NSM Option Editor to apply naming standards to tables, columns, and domains and to define naming standards by specifying the order in which parts of names are to appear in the physical name.

Follow these steps:

1. Expand NSM Options in the Model Explorer and select the NSM Option object for which you want to add physical names.
2. Right-click and select Properties.

   NSM Option Editor appears.
3. Go to the Physical tab.
4. Select the part of the name whether prime, modifier1, modifier2, or class that you want to include first in the physical name for the Table, Column, and Domain rows in each of the Part1, Part2, Part3, and Part4 rows.
5. Click Close.

   The data is saved.

Handle Vowels and Special Characters

If you do not want special characters and vowels in the name of your physical models, you can set actions to remove them.

Follow these steps:

1. Click Actions, Model Naming Options.

   The Model Naming Options dialog opens.

2. Click the Name Mapping tab.

3. Select *one* of the following actions from the Special Characters drop-down list:

   Leave

   Retains the special characters in the physical model object names.

   Remove

   Removes the special characters from the physical model object names.

   Replace

   Replaces the special character in the physical model object names with a specified character. Type the character in the Replace With box.

4. Select the Remove Vowels check box.

5. Click OK.

   The special characters and vowels are set for the Naming Standards in the physical model. The action taken for special characters and vowels are set when you update the physical naming standards.

Create Glossary Words and Abbreviations

You can enter words and their corresponding abbreviations directly into the glossary. You can enter, display, edit, and specify the word type (prime, modifier1, modifier2, or class).

Follow these steps:

1. Expand the Model Explorer, right-click NSM Options and select New.

   A new NSM Option object is added.

2. Right-click the new object and select Properties.

   NSM Option Editor appears.

3. Click the Glossary tab and work with the following options:

   Match whole words only

   > Specifies whether to perform glossary abbreviation substitution on whole words only or to include parts of words or substrings. Select the check box to perform glossary abbreviation substitution on whole words only. Clear the check box to perform glossary abbreviation substitution on complete words, as well as on parts of words or substrings.

   Word Type

   > Displays only the type of glossary words. Valid options are All, Prime, Modifier1, Modifier2, or Class.

   Word/Words

   > Specifies the glossary word or words that you want to abbreviate.

   Abbreviation

   > Specifies the standard abbreviation.

   Alt Abb

   > Specifies an alternate abbreviation.

   P, M1, M2, C

   > Specifies whether you want to use the word as a prime, modifier 1, modifier 2, or class.

4. Click Close.

   The data is saved.

Import Words and Abbreviations into the Naming Standards Glossary

You can import words and abbreviations into the glossary from an existing naming standards CSV file. Be sure to examine the data in the each row and edit or add accordingly, including descriptions for accuracy.

Follow these steps::

1. Expand the Model Explorer, right-click NSM Options and select New.

   A new NSM Option object is added.

2. Right-click the new object and select Properties.

   NSM Option Editor appears.

3. Click the Glossary tab, then click Import.

   The Open dialog appears.

4. Select the path and file name of the CSV file, and click Open.

   The words and corresponding abbreviations are imported into the Naming Standards glossary.

Example

You have two tables, Customer and Accounts. The Customer table has First Name and Last Name columns.

Suppose you want to implement the following standards:

- Prefix the names of the tables with Tbl_.
- Display First Name as Col_FNAME and Last Name as Col_LNAME.

How you do it:

1. Add a new NSM Object and make it active.
2. In the Logical tab, for Entity and Attribute, select Modifier 1 in the Part 1 and Prime in the Part 2 column.
3. In the Physical tab, for the Table and Column object types, select Modifier 1 in the Part 1 and Prime in the Part 2 column.
4. In the Glossary tab, add First Name and Last Name and the corresponding abbreviations. Select the P check box.
5. Open the Model Naming Options dialog and click the Name Mapping tab.
6. Select the Use Glossary check box for Entity to Table and Attribute to Column rows. In the Prefix column, enter Tbl_ in the first row and Col_ in the third row. Click Close.

   The standards you have defined are applied to the model.

Import Naming Standards from a Previous Version

If you have created naming standards in a previous version of CA ERwin DM, you can import them to the current release. When you import naming standards, the definitions for logical and physical models, and glossary words are imported.

Import from a Previous Version

If you have created naming standards for a previous version of CA ERwin DM, you can import them to an NSM Option object. For this, you require a file with .NSM extension.

Follow these steps:

1. Expand the Model Explorer, right-click NSM Options and select New.

   A new NSM Option object is added.

2. Right-click the new object (or an existing object) and select Properties.

   NSM Option Editor appears.

3. Click Import Other.

4. Select the .NSM file and then click Open.

   The NSM file is displayed as a new NSM Option object. The standards for Logical and Physical models are displayed in the respective tabs. The glossary words and corresponding abbreviations are displayed in the Glossary tab.

Apply Naming Standards by Binding a Model Template

When you bind a model template to the current model, the naming standards are also applied. If you run the wizard to select individual objects to bind, select the Naming_Options, NSM_Option, and NSM_Glossary_Word check boxes.

Bind a Template to a Model

Binding a model template to a data model copies the non-built-in objects from the template model to the target model. When you bind a model template, the predefined model objects appear in your model, eliminating the need to enter objects manually or depend on reverse engineering. Changes to the model template are reflected in the models that are bound to it, maintaining consistency of object names. Whenever you open a model with a bound model template, confirm whether to synchronize any model template changes with the objects used in the model.

Binding of model templates can occur at any time necessary, including during model creation, or after the model is created.

Objects brought in to a model from a template fully respect modeling rules. If conflicts occur in the model, the existing data is preserved as much as possible.

Follow these steps:

1. Click Model Templates, Bind Template on the File menu.

   The Bind Model Template dialog opens.

2. Select the model template to bind to the current model using the drop-down control. Use the toolbar to browse for the template file locally on your computer or one that is stored in a mart.

3. Select one of the following options to specify how to bind the template, and click OK.

   Load the entire contents of the template

   Specifies to load the entire contents of the template. This option binds all of the model template objects to the current model without the opportunity to review and select individual template objects. The NSM Option object that is active in the model template is made active in the current model as well.

   Run the Template Wizard to select objects for synchronization

   Specifies to open the Model Template Synchronization Wizard so that you can select the objects in the model template to synchronize with the data model.

   The model opens in the workspace with the template objects copied into it.

Verify Naming Standards Compliance

To know if the naming standards you have created are applied properly, use the Naming Standards Compliance dialog.

Follow these steps:

1. Click Tools, Standards, Check Naming Standards Compliance.

   The Check Name Standards Compliance dialog opens.

2. Select the objects to check and select the Check Word Order and Position check box and click Start.

   All instances of non-compliance are displayed in the Check Name Standards Compliance dialog. You can replace the name, replace all instances of the name, or ignore the non-conforming name.

Apply Naming Standards to a Model

After you define naming standards using the NSM Option Editor, select individual model objects and apply the standards. CA ERwin DM uses the glossary words and abbreviations (or alternate abbreviations) defined in the active NSM Option object to apply naming standards to a model.

Follow these steps:

1. Open a model that includes an active NSM Option object.
2. Click Actions, Model Naming Options.

   The Model Naming Options dialog opens.
3. Click the Name Mapping tab.
4. For each model object type that you want to use the glossary, select the Use Glossary check box and select the type of abbreviation. Click OK.

   The naming standards abbreviations (or alternate abbreviations) defined in the NSM Option object that is active, are applied to the current model.

Note: The translation from logical to physical through the NSM Option Object works only if the physical name is inherited from the logical name. That is, only if the names have not been manually changed in the physical model. If a physical name is modified in a logical/physical model, the inheritance from the logical side to the physical side is overridden and the naming standards no longer work. However, you can reset the override property to inherit from the logical name to restore this inheritance.

NSM File Attached to an Older Version Model

When you open a model that is created in an older version of CA ERwin DM, it is converted so that it is compatible with the current release. If an NSM file was attached to the model, the file is not imported and attached to the model automatically. Import the NSM file to a template and attach the template to the model manually.

Name Hardening Wizard

The Name Hardening Wizard provides a way to override inheritance of names and block name changes or resets within a model. Using the wizard, you can select the object types and object instances for which you must harden logical names or physical names, or both, within the model instead of using the inherited name values.

To run the Name Hardening Wizard, click Manage Name Hardening on the Actions menu.

You can harden names in this wizard for the following object types:

- Attribute/Column
- Default value
- Domain
- Key group/Index
- Relationship
- Entity/Table
- Validation rule

The wizard also includes the flexibility to specify whether to override name inheritance for all object instances of the selected object types, a subset of those object instances, or a specific object instance using the various pages of the wizard. The wizard contains four pages:

Overview

Contains an overview of the Name Hardening Wizard.

Type Selection

Displays the available object types that you can select for which you want to override inheritance such as Attributes, Default Values, and Domains. To make object type selection easier, you can right-click anywhere on this page to display a context menu where you can select either the Select All or Select None option. The options you select on this page are reflected in the next page of the wizard, the Object Selection page.

Note: While you are able to select the Entity check box in the Type Selection page in a logical model, name hardening can only be enabled in the physical model. Logical entity names are not inherited names; they are set on the entity object. Therefore, they do not change and there is no need to harden them.

Object Selection

Specifies the model objects or groups of model objects (collection nodes) for which you want to override inheritance. When you transition to this page from the Type Selection page, all the selections that you made on the Type Selection page are listed and selected on this page.

The collection nodes have check boxes that summarize the state of the objects under that node (subordinate objects). The check boxes for the collection nodes have three states. If none of the subordinate objects are selected, the check box for that collection node is not selected. If all the subordinate objects are selected, then the check box for that collection node is selected. If some of the subordinate objects are selected, then the check box for that collection node is selected, and is also shaded. Clicking the collection node check box allows toggling of the selection of the entire set of subordinate objects. In addition, you can right-click anywhere on this page to display a context menu where you can select either the Select All or Select None option. Any selections you make to the Selection Sets section on this page are immediately reflected in the Selected Objects section.

Action

Reflects the current name hardening settings for the selected objects. The Logical and Physical columns indicate the status of the objects and the Selected Objects column lists their fully qualified names, for example, instead of simply 'a,' an attribute name would be 'OwnerName.E/1.a.' Also, each object has a check box for selection; if you want to override inheritance, or harden, select a check box for the object; clear a check box for an object for which you want to unharden the name. When you unharden an object name, the same name you used in the hardening process remains, however, now you can edit or reset that name (editing or resetting of hardened names is not allowed).

Note: Unhardening an object name does not automatically restore inheritance, it only allows the names to be reset or changed.

Note: If you prefer, you can select all the objects directly on the diagram, launch the wizard, and go directly to the Action page. The objects you select on the diagram populate the Type Selection and Object Selection pages.

Override Name Inheritance

You can select the object types and object instances for which you need to set logical and physical names within the model instead of using the inherited name value, or for which you need to block name changes.

To override name inheritance

1. Click Manage Name Hardening on the Actions menu.

   The Name Hardening Wizard opens.

2. Click Type Selection.

   The Type Selection page opens.

3. Select the object types for which you want to override inheritance, then click Object Selection.

   The Object Selection page opens.

4. Select the model objects for which you want to override inheritance, then click Action.

   The Action page opens.

5. Select the check box for each item for which you want to override inheritance, then click Finish.

   Inheritance is overridden for each item you selected, and the Name Hardening Wizard closes.

Unharden Object Names

If you used the Name Hardening Wizard to set the names of selected object types or object instances (or both) in order to remove inheritance, you can unharden the names of those selected object types and object instances using the Name Hardening Wizard as well. When you unharden an object name, the same name you used in the hardening process remains, however, once the unharden process is complete, you will have the ability to edit or reset that name (editing or resetting of hardened names is not allowed).

Note: Unhardening an object name will not automatically restore inheritance, it only allows the names to be reset or changed.

To unharden object names

1. Click Manage Name Hardening on the Actions menu.

   The Name Hardening Wizard opens.

2. Click Type Selection.

   The Type Selection page opens.

3. Select the object types that you want to unharden, then click Object Selection.

   The Object Selection page opens.

4. Select the model objects that you want to unharden, then click Action.

   The Action page opens.

5. Select the check box for each item for which you want to unharden the names, then click Finish.

   The name for each object that you selected is unhardened, and the Name Hardening Wizard closes.

Applying Data Type Standards

A data type is a predefined set of characteristics for an attribute or column that specifies field length, acceptable characters, and optional and required parameters. For example, the data type Char(18) specifies that the column can store up to 18 alpha and numeric characters.

By default, a data type is applied to every attribute in a logical model and to every column in a physical model. In the logical model, the data type is determined by the domain from which the attribute inherits its properties or from the data type that you assign. In the physical model, the data type is determined by the default value that is specified by the target server or the data type that you assign.

Considering the large quantity of attributes or columns a data model usually has, it is tedious to assign and maintain consistency of data types manually. The DSM Option Editor helps you easily assign and maintain data types in a model.

Create data type standards so that everyone in your organization uses data types consistently.

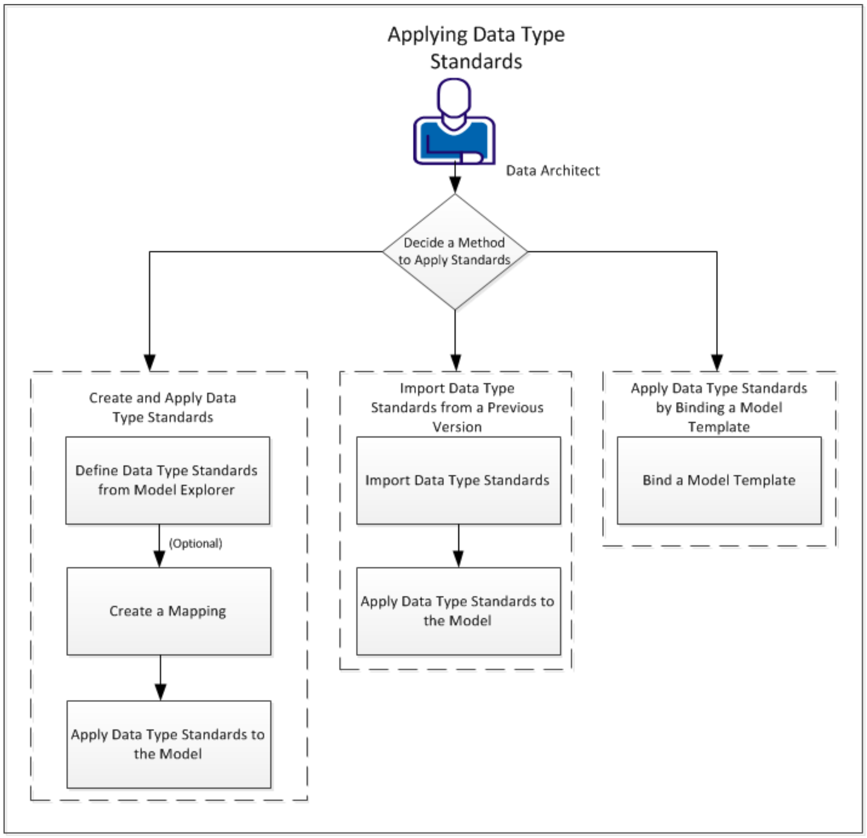
When you create a DSM Option object, it is preloaded with default logical data types. Use the DSM Options Editor to create data types and mappings. You can create multiple DSM Option objects but you can apply only one of them to the model.

You can assign data types in any of the following scenarios:

- Deriving a physical model from a logical model, or switching from a logical model to a physical model. A physical data type is automatically assigned to each column based on the data type assigned to the corresponding attribute.
- Deriving a physical model from a physical model. The corresponding data type for the target database is automatically assigned.

- Migrating from one version of a database to a newer version, or changing the target database. You are prompted you to verify if you want to convert data types. When you convert the data types, the corresponding data type for the target database is automatically assigned.
- Creating a physical model by reverse engineering from a database or script file. Information from the database or script file is extracted, and a physical data type is automatically assigned to each column.

You can apply data type standards in three ways, as illustrated in the following diagram:


Applying Data Type Standards
Data Architect
Decide a Method to Apply Standards
Create and Apply Data Type Standards
Define Data Type Standards from Model Explorer
(Optional)
Create a Mapping
Apply Data Type Standards to the Model
Import Data Type Standards from a Previous Version
Import Data Type Standards
Apply Data Type Standards to the Model
Apply Data Type Standards by Binding a Model Template
Bind a Model Template

Follow these steps to apply data type standards:

Decide a method to apply standards (see page 54).

Follow these steps to create and apply data type standards:

1. Define data type standards from Model Explorer (see page 55).
2. (Optional) Create a mapping (see page 56).
3. Apply data type standards to the model (see page 56).

Follow these steps to import data type standards from a previous version:

1. Import data type standards (see page 57).
2. Apply data type standards to the model (see page 57).

Apply data type standards by binding a model template:

1. Bind a model template (see page 58).

Decide a Method to Apply Standards

You can apply data type standards in three ways:

Create a standard

Create a data type standard and apply it to your model if you are applying standards for the first time. After you create a DSM object and define your standards, you can use the standard in all your data models.

Import from a previous release

Import data type standards from a previous release and apply the standard to your model. It saves your time from creating a new standard and maintains the consistency across all data models in your organization.

Bind a model template

Bind an existing model template having data type standards to your model. You can use the same template for multiple models. You can also sync the template with the current model for any changes in the model. After you sync the template, it will inherit the changes that are made in the current model.

Create and Apply Data Type Standards

Create data type standards if you are implementing standards for the first time.

Define Data Type Standards from Model Explorer

Define data type standards to maintain uniform data type standards across all data models.

Follow these steps:

1. Expand Model Explorer, right-click DSM Options, and click New.

   A data type standards object is added.

2. Right-click the newly created object and click Properties.

   The DSM Option Editor opens.

3. (Optional) Rename the DSM object in the Name field.
4. Click the Logical Definition tab and edit the Domain, Length, Precision, and Default Length values for the required data type.
5. (Optional) To create a data type, click New under the Logical Definition tab.

   A data type is created.

6. (Optional) Rename the new data type and change the properties.

   A data type standard object is defined and saved.

Note: To delete a DSM object or a data type, select the DSM object or the data type row and click Delete.

If you want to create a mapping, you do not have to close the DSM Option Editor window.

(Optional) Create a Mapping

Map data types between different databases to maintain uniform data types in all the data models. Mapping lets you eliminate the differences in the data types when you import a model from one database to another. Also, mapping enables you to use new data types in physical models.

Note: You can create multiple DSM Datatype Mapping objects and DSM Datatype Mapping entries in the DSM Datatype Mapping Editor.

Follow these steps:

1. In the DSM Option Editor, select the Mapping tab and click New.

   A DSM Datatype Mapping object is created.

2. Click Edit.

   The DSM Datatype Mapping Editor opens.

3. (Optional) Edit the name of the DSM Datatype Mapping object.
4. Select the Mapping tab, and select the source database from the From drop-down list and the target database from the To drop-down list.
5. (Optional)Edit the To Datatype field.
6. (Optional) Add a data type for mapping.
7. Click Close.

   The DSM Datatype Mapping Editor closes and you return to the DSM Option Editor dialog.

Apply Data Type Standards to the Model

After you define data type standards, apply the standards by attaching the DSM object to the model. If you have created multiple DSM object, you can attach only one DSM object to the model.

To attach a DSM object to a model, open the DSM Option Editor from Model Explorer, and select the *Is Active* checkbox.

Import Data Type Standards from a Previous Version

If you have created data type standards in a previous version of CA ERwin DM, you can import it to CA ERwin DM version 9. When you import data type standards, mappings also get imported.

Import Data Type Standards

Import data type standards from a previous version so that you can use the same standards in CA ERwin Data Modeler version 9. Importing standards maintains consistency across all data models in your organization and saves time from creating standards.

Note: The file you are importing must have a .DSM extension.

Follow these steps:

1. Expand the Model Explorer, right-click DSM Options and select New.

   A new DSM Option object is added.

2. Right-click the new object (or an existing object) and select Properties.

   The DSM Option Editor appears.

3. Click Import Other.

4. Select the .DSM file from the stored location and click Open.

   The DSM file is displayed as a new DSM Option object.

Apply Data Type Standards to the Model

After you import the data type standards object to CA ERwin Data Modeler version 9, apply the standards by attaching the DSM object to the data model.

To attach a DSM object to a model, open the DSM Option Editor from Model Explorer, and select the *Is Active* checkbox.

Apply Data Type Standards by Binding a Model Template

When you bind a model template having data type standards to the current model, data type standards are also applied.

Bind a Model Template

Binding a model template to a data model copies the non-built-in objects from the template model to the target model. When you bind a model template, the predefined model objects populate your model, eliminating the need to enter objects manually or depend on reverse engineering. Changes to the model template are reflected in the models that use it, maintaining consistency of object names. Whenever you open a model with a bound model template, confirm whether to synchronize any model template changes with the objects used in the model.

Binding of model templates can occur at any time necessary, including during model creation or after the model is created.

Objects brought in to a model from a template fully respect modeling rules. If conflicts occur in the model, the existing data is preserved as much as possible.

Follow these steps:

1. Click File, Model Templates, Bind Template.

   The Bind Model Template dialog opens.

2. Use the toolbar to browse for the template file locally on your computer or in Mart. If you have used the bind template feature before, use the drop-down control to select the model template to bind to the current model.

3. Select one of the following options to specify how to bind the template, and click OK.

   Load the entire contents of the template

   > Specifies to load the entire contents of the template. This option binds all of the model template objects to the current model without the opportunity to review and select individual template objects.

   Run the Template Wizard to select objects for synchronization

   > Specifies to open the Model Template Synchronization Wizard so that you can select the objects in the model template to synchronize with the data model.

   The model opens in the workspace with the template objects copied into it. The data type standard is applied on the model.

Appendix A: Supported Metadata Integration Bridges

Table of Supported Bridges

The following table lists the Meta Integration Technology, Inc. (7.2.1) bridges that are currently available in CA ERwin Data Modeler (as of Release 9.5):

Note: Read the information displayed in the tip text area on the Source and Destination pages of the Import from External Format and Export to External Format dialogs carefully. Complete any requirements as appropriate before proceeding.

Application	Mode
Adaptive Metadata Manager (via CWM XMI)	Import/Export
Altova XMLSpy	Export
Apache Hadoop Hive Native bridges include: Apache Hadoop Hive Apache Hadoop HCatalog Cloudera Enterprise (Hadoop Hive) Cloudera Impala (Hadoop Hive) DataStax Enterprise (Hadoop Hive) Hadapt Hadoop Hive Hortonworks Hadoop Hive MapR Hadoop Hive	Import
Borland Together (via UML 1.x XMI)	Import/Export
Borland Together (via UML 2.x XMI)	Import
CA Component Modeler 3.52 (ParadigmPlus)	Import/Export
CA Component Modeler 4.x (via UML 1.x XMI)	Import/Export
CA COOL:Biz 5.1	Import
CA COOL:BusinessTeam (GroundWorks) 2.2.1	Import
CA COOL:DBA (Terrain for DB2) 5.3.2	Import
CA COOL:Enterprise (ADW) 2.7	Import
CA COOL:Xtras Mapper (TerrainMap for DB2)	Import
CA ERwin 3.x (ERX)	Import

Application	Mode
CA ERwin 4.x Data Modeler	Import
CA ERwin Web Portal	Export
CA Gen	Import/Export
CA Repository for Distributed Systems	Import/Export
CA Repository for z/OS	Import/Export
COBOL Copybook Flat Files	Import
Database (via JDBC)	Import
Database (via ODBC)	Import
Database Data Definition Language (DDL)	Export
Embarcadero Describe (via UML 1.x XMI)	Import/Export
Embarcadero ER/Studio Business Architect (via DM1)	Import/Export
Embarcadero ER/Studio Data Architect	Import/Export
Embarcadero ER/Studio Repository	Import
EMC ProActivity 3.x & 4.0	Import
Gentleware Poseidon (via UML 1.x XMI)	Import/Export
Grandite Silverrun-RDM 2.4.4 to 2.8.6	Import
Google BigQuery	Import
IBM Cognos BI Reporting - Content Manager Packages	Import
IBM Cognos BI Reporting - Content Manager QueryStudio	Import
IBM Cognos BI Reporting - Content Manager ReportStudio	Import
IBM Cognos BI Reporting - Framework Manager (FM)	Import
IBM Cognos BI Reporting - Framework Manager (FM) for ReportNet	Import
IBM Cognos BI Reporting - Framework Manager	Export
IBM Cognos BI Reporting - Framework Manager for ReportNet	Export
IBM Cognos BI Reporting - PowerPlay Transformer	Import
IBM Cognos BI Reporting - QueryStudio	Import
IBM Cognos BI Reporting - ReportStudio	Import
IBM Cognos Series 7 Impromptu	Import
IBM DB2 Cube Views	Import/Export
IBM DB2 Data Server (via JDBC)	Import

Application	Mode
IBM DB2 Warehouse Manager (via CWM XMI)	Import/Export
IBM InfoSphere Data Architect (IDA)	Import/Export
IBM InfoSphere DataStage	Import
IBM InfoSphere Discovery (via CWM XMI)	Import
IBM InfoSphere Federation Server (via JDBC)	Import
IBM InfoSphere Warehouse - InfoSphere Data Architect	Import/Export
IBM Lotus Notes (via JDBC-ODBC)	Import
IBM Rational Rose 4.0	Import/Export
IBM Rational Rose 6.0(98i) to 6.5(2000)	Import/Export
IBM Rational Rose 7.x (2000e and newer)	Import/Export
IBM Rational Rose XDE Developer (via Rose MDL)	Export
IBM Rational Software Architect (RSA) (via Rose MDL)	Export
IBM Rational Software Architect (RSA) (via UML 2.x XMI)	Import
IBM Rational Software Modeler (RSM) (via Rose MDL)	Export
IBM Rational System Architect (SA) 10.4 to 11.x (Encyclopedia)	Import/Export
IBM Rational System Architect (SA) 7.1 to 11.x (File)	Import/Export
IBM Telelogic Tau (via UML 1.x XMI)	Import/Export
Informatica Data Analyzer	Export
Informatica Developer	Import/Export
Informatica PowerCenter	Export
Informatica PowerCenter (File)	Import
Informatica PowerCenter (Repository)	Import
IRI CoSORT RowGen Data Definition File	Export
IRI CoSORT SortCL Data Definition File	Export
Kalido DIW	Import
Meta Integration Repository (MIR) Application Server	Export
Micro Focus (Merant) AppMaster Builder	Import
Microsoft Office Access (via JDBC-ODBC)	Import
Microsoft Office Excel	Import/Export
Microsoft Office Excel (Java Alpha)	Import/Export

Application	Mode
Microsoft Office Visio Database (via ERX)	Import
Microsoft Office Visio UML (via UML 1.x XMI)	Import
Microsoft SQL Server Analysis Services (Repository)	Export
Microsoft SQL Server Analysis Services (via DSV File)	Import/Export
Microsoft SQL Server Analysis Services 2005 to 2008 (File)	Import
Microsoft SQL Server Analysis Services 2005 to 2008 (Repository)	Import
Microsoft SQL Server Analysis Services 7.0 to 2000 (Repository)	Import
Microsoft SQL Server Analysis Services 7.0 to 2000 (via ODBO)	Import
Microsoft SQL Server Data Source View	Import/Export
Microsoft SQL Server Database (via JDBC)	Import
Microsoft SQL Server Integration Services (File)	Export
Microsoft SQL Server Integration Services (via DSV File)	Import/Export
Microsoft SQL Server Reporting Services (File)	Import
Microsoft SQL Server Reporting Services (Repository)	Import
Microsoft SQL Server Repository 2.1b (via XIF)	Import/Export
Microsoft SQL Server Repository 3.x (via MDC)	Import/Export
Microsoft SQL Server XML Data Reduced (XDR) Schema	Import
Microsoft Visual Studio / Modeler 2.0 (via MDL)	Import/Export
MicroStrategy MDX Adapter (via ODBO)	Import
NoMagic MagicDraw (via UML 1.x XMI)	Import/Export
OMG CWM 1.x XMI 1.x	Import/Export
OMG CWM Pre-1.0 XMI 1.1	Import/Export
OMG UML 1.x XMI 1.x	Import/Export
OMG UML 2.x XMI 2.x	Import
Open Text eDOCS Data Integration (Hummingbird) Genio	Import/Export
Oracle Data Integrator (ODI) 11g - Beta bridge	Import/Export
Oracle Database (via JDBC)	Import
Oracle Designer	Import/Export
Oracle Hyperion Application Builder (via CWM XMI)	Import/Export
Oracle Hyperion Essbase - Beta bridge	Import

Application	Mode
Oracle Hyperion Essbase Integration Services (EIS)	Import/Export
Oracle Warehouse Builder (OWB)	Import/Export
Oracle Warehouse Builder (OWB) (via CWM XMI)	Import/Export
SAP BusinessObjects Crystal Reports XI (File)	Import
SAP BusinessObjects Crystal Reports XI (Repository)	Import
SAP BusinessObjects Data Integrator (BODI)	Import/Export
SAP BusinessObjects Data Integrator (BODI)	Import
SAP BusinessObjects Designer (File)	Import/Export
SAP BusinessObjects Designer (Repository)	Import
SAP BusinessObjects Desktop Intelligence (File)	Import
SAP BusinessObjects Desktop Intelligence (Repository)	Import
SAP BusinessObjects Information Design Tool (IDT) (Repository)	Import
SAP BusinessObjects Information Design Tool (IDT) (File)	Import
SAP BusinessObjects Web Intelligence XI	Import
SAP ERP Central Component (ECC) - Beta bridge	Import
SAP NetWeaver Business Warehouse (BW) (via ODBO)	Import
SAP NetWeaver Business Warehouse (BW) (via XML)	Import
SAP NetWeaver Master Data Management (MDM) - Beta bridge	Import/Export
SAS Data Integration Studio (via CWM XMI)	Import/Export
SAS OLAP Server (via ODBO)	Import
Select SE	Import
Sparx Enterprise Architect (EA) (via UML 1.x XMI)	Import/Export
Sparx Enterprise Architect (EA) (via UML 2.x XMI)	Import
Sybase Database (via JDBC)	Import
Sybase PowerDesigner CDM 6.x	Import
Sybase PowerDesigner CDM 7.5 to 16.x	Import
Sybase PowerDesigner CDM 8.0 to 15.x	Export
Sybase PowerDesigner OOM 9.x to 15.x (via UML 1.x XMI)	Import/Export
Sybase PowerDesigner OOM 9.x to 15.x (via UML 2.x XMI)	Import
Sybase PowerDesigner PDM 6.1.x	Import

Application	Mode
Sybase PowerDesigner PDM 7.5 to 16.x	Import
Sybase PowerDesigner PDM 8.x to 15.x	Export
Sybase PowerDesigner LDM 15.x to 16.x	Import
Sybase PowerDesigner XSM 10.x to 15.x	Export
Syncsort DMExpress	Import/Export
Tableau Server - Beta bridge	Import
Talend Data Integration - Beta bridge	Import/Export
Teradata Database (via JDBC)	Import
Tigris ArgoUML (via UML 1.x XMI)	Import/Export
Visible IE:Advantage 6.1	Import
W3C XML DTD 1.0	Import
W3C XML Schema 1.0 (XSD)	Import/Export

Notes:

- For information about how to use the metadata integration wizards, see the CA ERwin Data Modeler online help.
- For Oracle Data Integrator 11g, the new Java API based architecture is supported.
- For Oracle Warehouse Builder 11.2, the new API architecture for 11.2 (different from 11.1) is supported.
- IBM Rational System Architect 7.1 to 11.x (File) and IBM Rational System Architect 10.4 to 11.x (Encyclopedia) must be installed on your local computer before you use these import and export bridges.

Index