

# **Toad Data Modeler 3**

## **User Guide**

---

# Table of Contents

<b>Toad Data Modeler 3.....</b>	<b>1</b>
<b>Introduction to Toad Data Modeler.....</b>	<b>11</b>
About Toad Data Modeler.....	11
With Toad Data Modeler you can:.....	11
Benefits.....	11
Key Features.....	12
Other Useful Features.....	13
Technical Requirements.....	13
Supported Databases.....	14
<b>Interface Layout.....</b>	<b>15</b>
Layout Description.....	15
Main Menu and Toolbars.....	16
Application Window.....	18
Workspace versus Designer.....	19
Navigation on Workspace.....	21
Workspace Right-Click Options.....	21
Application View.....	22
Explorers.....	23
Model Explorer.....	23
Message Explorer.....	23
Shortcuts of Objects.....	24
About Shortcuts of Objects.....	24
Use Shortcuts.....	26
Example 1:.....	27
Example 2:.....	29
Tips and Tricks.....	31

Command Line Parameters.....	31
Hot Keys.....	33
Dock Panes.....	36
Modeless Dialogs/Forms.....	38
Instances of Forms.....	40
Inplace Editor.....	41
Object Navigator Box.....	42
Object Viewer.....	43
Search Objects.....	44
Search Items in Trees.....	46
Other Quick Tips.....	46
<b>Physical Data Model.....</b>	<b>48</b>
About Physical Model.....	48
Create Entity Relationship Diagram.....	48
Select Target Database.....	48
Create Entities.....	50
Create Attributes.....	52
Create Primary Key.....	54
Tips and Tricks for Keys.....	58
Other Notes.....	59
Create Relationship.....	59
Identifying Relationship.....	61
Non-Identifying Relationship.....	62
Self-Relationship for Non-identifying Relationship.....	62
M:N Relationship.....	63
Define Cardinality.....	64
Synchronization of NotNull and Mandatory Parent.....	65
Display Partiality.....	66

Set up Referential Integrity Rules .....	67
Connect Parent and Child Entities .....	68
Understanding of Foreign Keys .....	70
PFK .....	71
FK .....	71
Parent Attributes (Rolenames) .....	73
Advanced Database Design .....	75
About Advanced Options .....	75
Dictionary Items .....	75
User Data Types .....	75
Dictionary Types .....	75
Domains .....	76
Export/Import of the Dictionary .....	76
Alternate Keys .....	84
Indexes .....	86
Check Constraints and Triggers .....	89
Views .....	93
Materialized Views .....	96
Procedures, Functions, Schemas, Synonyms, Defaults .....	98
Users and User Groups .....	100
Permissions .....	105
Naming Conventions .....	106
About Naming Conventions .....	106
Basic Operations .....	108
Default Synchronization of Caption and Name .....	109
Scenario 1 .....	111
Scenario 2 .....	119
Scenario 3 .....	122



Scenario 4.....	125
Use Cases - Different Naming Convention Settings.....	127
A: Automatic Synchronization On, Apply Naming Convention Rules On.....	128
B: Automatic Synchronization Off, Apply Naming Convention Rules Off.....	128
C: Automatic Synchronization On, Apply Naming Convention Rules Off.....	128
D: Automatic Synchronization Off, Apply Naming Convention Rules On.....	129
Glossaries, Import of CSV Files.....	130
Verification.....	133
Model Verification.....	133
Documentation.....	135
Documentation to Physical Model.....	135
Report Generation.....	136
XSL Transformation.....	146
Advanced Users.....	147
Where to Find the Simplified XML File?.....	149
Predefined XSL Templates.....	150
Path to XSLT files.....	150
Sample XSLT File.....	151
Special instructions, parameters.....	151
Edit Existing XSLT Templates.....	151
XSD Generation.....	152
Script Generation.....	153
Order of Generated Objects.....	153
Select Encoding.....	155
Select Items.....	155
Detailed Settings.....	157
Referential Integrity.....	158
Select Entities/Workspaces and Save Settings.....	159

Preview SQL Script.....	160
Example of Generated Script.....	162
<b>Reverse Engineering.....</b>	<b>169</b>
About Reverse Engineering.....	169
Select Stored Alias.....	170
Select Data Source.....	172
Select Data Provider.....	172
Connect to Database.....	173
Select Data Migrator.....	174
Select Objects to Reverse.....	174
Select Options.....	175
Save Alias.....	177
Select Schemas and Tables.....	178
Reverse Engineering.....	178
LIVE Reverse Engineering.....	180
How does it work?.....	180
Load Models from DDL/SQL Script.....	182
<b>Model Compare.....</b>	<b>186</b>
Model Compare.....	186
<b>Model Update/Model Merge.....</b>	<b>186</b>
Model Update.....	186
Model Merge.....	189
<b>Alter Scripts.....</b>	<b>192</b>
Alter Scripts.....	192
<b>Model Conversion.....</b>	<b>196</b>
Model Conversion - PER - PER.....	196
Model Conversion - PER - LER.....	199
<b>Import Toad Data Modeler 2.x Models.....</b>	<b>201</b>

Import Toad Data Modeler 2.x Model.....	201
<b>Logical Data Model.....</b>	<b>202</b>
About Logical Model.....	202
Create Logical ER Diagram.....	202
Create Logical Model.....	202
Add Entities.....	204
Add Attributes.....	205
Create Unique Identifier.....	206
Create Relationship.....	208
Specify Relationship Properties.....	208
Create Inheritance.....	210
Specify Inheritance Properties.....	211
Verification.....	213
Model Verification.....	213
Report Generation.....	214
Report Generation.....	214
Model Compare.....	216
LER Model Compare.....	216
Model Update/Model Merge.....	217
LER Model Merge.....	217
Model Conversion.....	218
Information on LER - PER Conversion.....	218
Conversion LER - PER.....	219
<b>Change the Look and Save Models.....</b>	<b>222</b>
Rename.....	222
Display Modes.....	222
Change Notation.....	224

Format Workspaces and Objects.....	225
Display Level for Entities.....	226
Select Colors for Attributes.....	226
Show Grid and Grid Size.....	227
Align Objects.....	228
Arrange Relationship Lines.....	229
Straight Mode.....	229
Right Angled Mode.....	230
Move Relationship Anchor Points.....	230
Move Relationship Lines.....	230
Move, Hide, Find Relationship Names.....	230
Autolayout.....	231
Zoom, Overview, Loupe, Move.....	231
Tips.....	232
Categories.....	233
Add Notes and Note Lines.....	235
Add Stamp.....	236
Save Model.....	238
<b>Maintaining Models.....</b>	<b>240</b>
Model Explorer.....	240
Examples of Use in Model Explorer.....	242
Message Explorer.....	243
To Do List.....	243
Undo/Redo Functions.....	245
<b>Toad for Oracle Integration.....</b>	<b>247</b>
Integration Options.....	247
Toad for Oracle Aliases.....	247

Toad for Oracle as a Default Editor.....	247
Toad for Oracle Icons in Toad Data Modeler.....	248
Open Toad for Oracle Projects.....	248
Import Toad for Oracle ER Diagrams.....	250
<b>Synchronization.....</b>	<b>254</b>
Synchronization.....	254
Database and Model Synchronization.....	254
Physical Model and Logical Model Synchronization.....	254
<b>Version Manager.....</b>	<b>256</b>
Create New Project and Add Files to Projects.....	256
Work with Versions/Revisions.....	261
Version Check Out.....	261
Version Check In.....	262
Places to Manage Versions.....	264
Tips on Version Right-click Options.....	265
View and Sort VM Items.....	269
<b>Configuration.....</b>	<b>271</b>
Options.....	271
<b>Customization - Sample.....</b>	<b>273</b>
Expert Mode.....	273
Script and Package Explorers.....	274
Our Goal.....	274
Our Goal Definition:.....	275
Create Package.....	277
Create a New Package.....	277
Add New Properties in Metamodel.....	281
How To Add New Properties.....	281

Modify Form.....	287
Set Default Values.....	295
How to Define Default Values.....	295
Add Events.....	297
Access Property Values via Scripting Window.....	304
Code:.....	306
Where to find information about objects and their properties and methods?.....	308
Executing the Script.....	310
Create Script.....	311
Code.....	312
Explanation.....	313
Call Existing Script from Model Properties Form.....	313
Code.....	316
Explanation.....	317
Modify HTML Reports.....	317
Code:.....	323
Explanation.....	324
<b>Printing Models.....</b>	<b>327</b>
Page Format Setup.....	327
Preview and Print.....	328
Export to Graphic File.....	330
<b>Appendix: Contact Quest.....</b>	<b>332</b>
Contact Quest Support.....	332
Contact Quest Software.....	332
About Quest Software.....	332
<b>Index.....</b>	<b>334</b>

# Introduction to Toad Data Modeler

## About Toad Data Modeler

Quest Software's Toad Data Modeler helps organizations create, maintain and document their database systems with an easy-to-use graphical interface into new or existing database structures. Toad Data Modeler enables users to reverse engineer database structures, update models, design and generate SQL, compare models and generate alter scripts, create HTML or RTF reports, in addition to versioning of database designs and creation of to-do lists.

## With Toad Data Modeler you can:

- Visually create database structures (Logical and Physical Entity Relationship Diagrams - ERD).
- Create ERD for various database systems (Oracle, MS SQL, MySQL etc.).
- Reverse engineer already existing database structures and see the existing database structure in form of a diagram.
- Add logical data to your diagrams and describe existing database structures better.
- Verify ERD (model) and get a list of Errors, Warnings and Hints.
- Automatically generate SQL code for selected target database.
- Generate detailed documentation in HTML, RTF or PDF format.
- Synchronize your model with physically existing database (using Alter Script Generation and Model Merge features).
- Keep track of changes using internal Version Manager.
- Create and maintain a list of tasks related to their model or particular part of a model using To-Do list feature.
- Perform their daily tasks quickly and comfortably, thanks to GUI improvements.
- Customize the product according to your requirements (including customization of forms) and more.
- Affect objects in ERD via internal scripting (user-defined batch rename scripts etc.).

## Benefits

- Reduce faults in development
- Significantly improve your productivity

- Visualize your database structures and get better understanding of your existing databases
- Verify your work automatically
- Generate very detailed HTML or RTF reports
- Work with existing database structures
- Keep existing database structures up-to-date
- Create databases faster

## Key Features

- Physical Model
- **Logical Model**
- Support for Various Databases
- Reverse Engineering
- SQL/DDDL Script Generation
- **Alter Scripts Generation**
- HTML/RTF Reports
- Version Manager
- Model Explorer
- **Model Merge**
- Model Compare
- Model Verification
- Package Explorer
- Script Explorer
- Autolayout
- Workspaces and Designers
- **Editable Forms**
- Dockable Panes
- Modeless Dialogs/Forms
- Categories
- Message Explorer
- **Undo/Redo**
- To-Do List
- Zoom, Loupe, Model Overview features and many more...



## Other Useful Features

- Object Navigator box
- Possibility to define and change default values in forms/dialogs/frames
- Possibility to open and view different/same parts of your large model at the same time in a different zoom
- Message Explorer
- Multiple graphical representation of one object in a Workspace or more Workspaces (Object Shortcut)
- Colorful distinguishing of objects (parts) of your large models (Categories)
- Fast moving and shifting in your model and among models
- Zooming an object or group of objects
- Model Merge
- Scripting Features - customize the application via Script and Package Explorers
- Creating projects in Version Manager
- Model Update
- Model Conversion
- and more...

## Technical Requirements

	Minimal	Optimal
<b>Platform</b>	Pentium IV	Pentium dual core
<b>Memory</b>	256 MB	1 GB
<b>Hard Disk Space</b>	100 MB	200 MB
<b>Operating System</b>	Microsoft Windows 2000 Professional Edition (Service Pack 4) Microsoft Windows XP 32-bit Microsoft Windows XP 64-bit Microsoft Windows Vista 32-bit Microsoft Windows Vista 64-bit	

## Supported Databases

- DB2 9.5 (LUW)
- DB2 v. 9 (LUW)
- DB2 UDB v. 8 (LUW)
- MS Access 2000/2002/2003
- MS SQL Server 2008
- MS SQL Server 2005
- MS SQLServer 2000
- MySQL 5.1
- MySQL 5.0
- Oracle 11g
- Oracle 10g
- Oracle 9
- PostgreSQL 8.3
- PostgreSQL 8.2
- PostgreSQL 8.1
- Sybase ASE 15
- Sybase ASE 12.5

Support for latest versions of other database systems will be added to the product gradually.

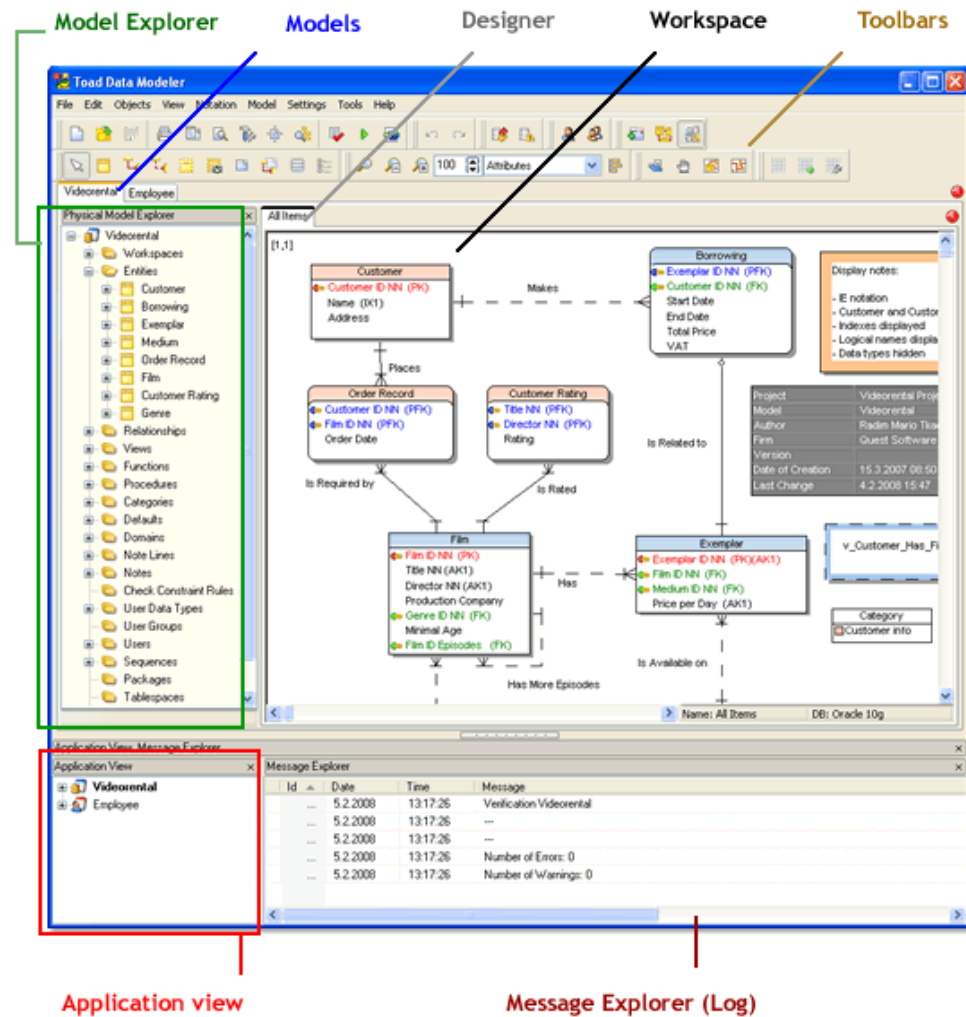
# Interface Layout

## Layout Description

Toad Data Modeler interface is intuitive, clear and flexible thanks to the ability to dock panes. This function allows you to change the layout to suit your needs and preferences.

The default look of Toad Data Modeler consists of the following components:

- Main Menu and Toolbars
- Application Window (with Designers and Workspaces)
- Application View
- Model Explorer
- Message Explorer



## Related Topics

Main Menu and Toolbars (page 16)

Application Window (page 18)

Application View (page 22)

Explorers (page 23)

## Main Menu and Toolbars

Main menu options and toolbars differ by type of model (logical, physical etc.), and by what is active in the Application Window (Designers, Scripting Window etc.). E.g. To enable the



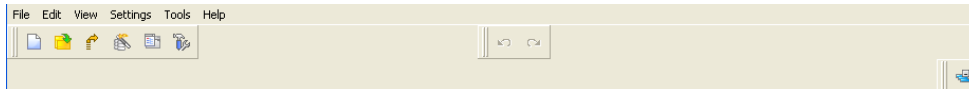
### *To turn on and off particular toolbar*

- » Select **View | Toolbars**.

**Tip:** Right-click the toolbar to modify the toolbar view.

## Main Menu and Toolbars before a model is opened:

(Expert Mode is disabled.)



## Main Menu and Toolbars after the Designer for PER model has been opened:

(Expert Mode is enabled.)



In the **View** menu | **Icons Theme**, you can select Toad Data Modeler icons or Toad for Oracle icons. See the toolbar with Toad for Oracle icon theme:



**Note:** Detailed toolbar description + hot keys used in Toad Data Modeler can be found in the Help file, "Interface Layout" chapter.

## Related Topics

Expert Mode (page 273)

## Application Window

Application Window is a space where Workspaces (WS) and their Designers are placed. It's an area where you design your models, write scripts etc.

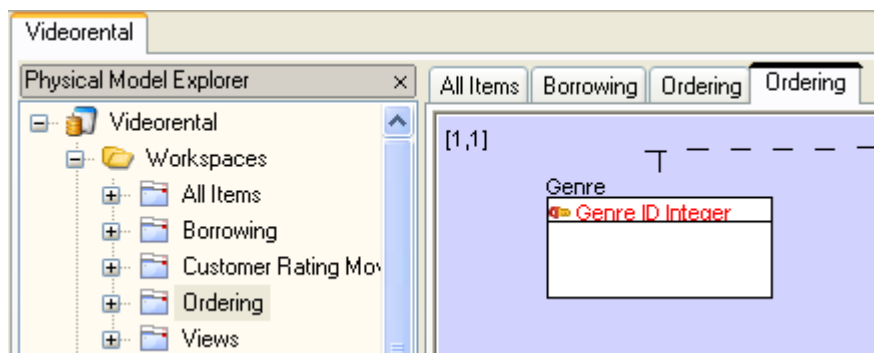
- Toad Data Modeler 3.x allows you to work with several models of same or different databases simultaneously. - The models and their Workspaces and Designers are organized in the Application Window on tabs by models.
- To close all model WS and Designers at one jump, click x (upper x) in the right-hand corner of the model tab in AW.
- To close a tab with particular Designer, click x (lower x) or select **File | Close Designer**.

**Tip:** You can move tabs in the Application Window easily. - Press CTRL, select a tab and use drag&drop techniques to move it.

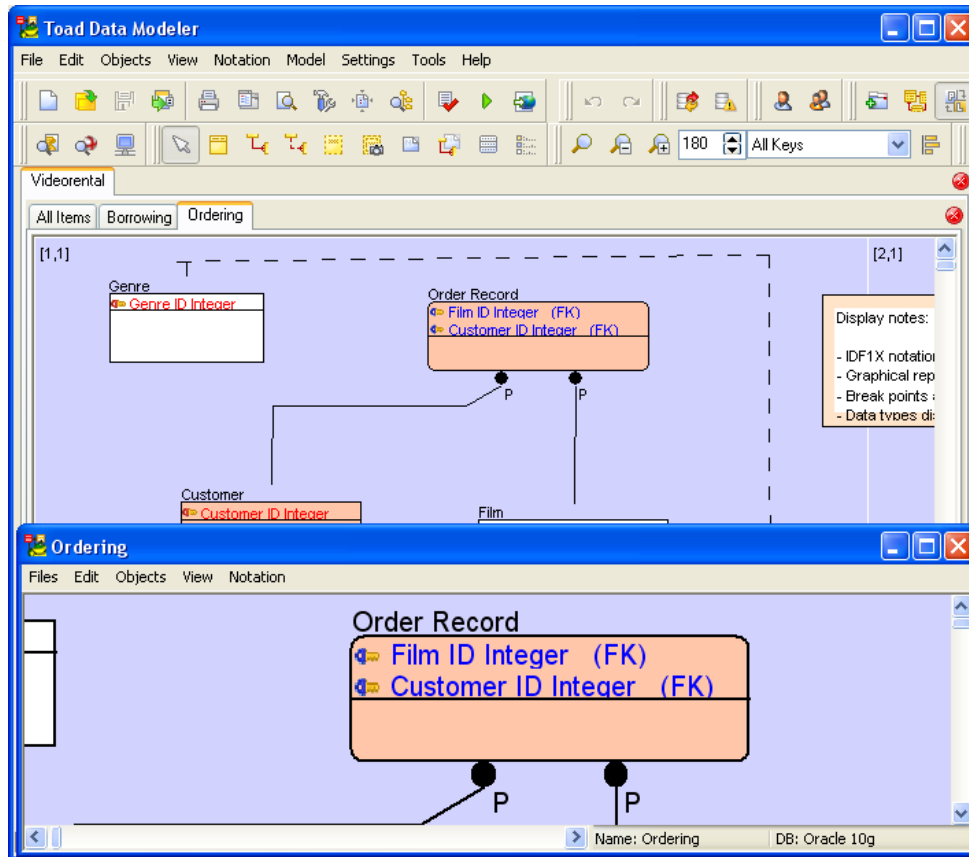
## Workspace versus Designer

- Designer displays a Workspace (WS). It is an area where WS is displayed (tab).
- One Designer represents one WS.
- If you create a new Workspace, its Designer will open automatically in the Application Window.
- A model can have several Workspaces.
- For one WS, you can open several Designers.

**Example:** Videorental model has five Workspaces - see them listed in the Model Explorer. Designers of three of them are opened in the Application View - *All Items*, *Borrowing* and *Ordering*. For the *Ordering* WS, two Designers are opened.

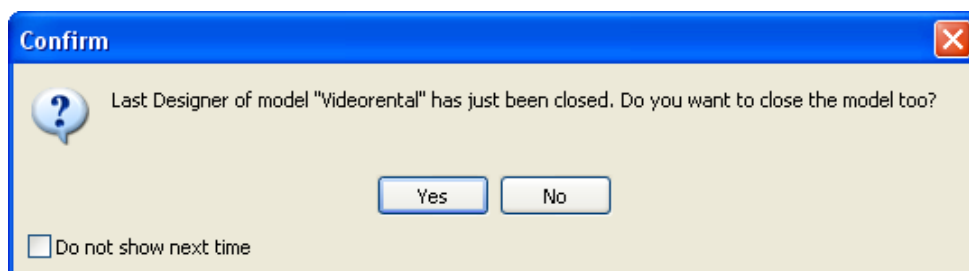


- Workspaces are similar to submodels. They allow you to display different/same objects of your model. The first WS is called *All Items* and is meant as main model.
- Designers opened for the same Workspace display the same objects. They allow you to view different/same parts of the WS, and e.g. in a different zoom.
- You can undock designers or dock them to a particular model in the Application Window.



## Close Model when Closing its Last Designer

When you close the last Designer of your model, the following dialog will display:



This message should draw your attention to the fact that although you have closed your model from the Application Window, it is still open in Toad Data Modeler (see the model listed in the Application View). This message should prevent you from the situation when you open the same model again and rewrite the changes made in first instance of the model.

Select **Yes** to close your model. If you made any changes, you will be asked to save them before the model close.



## Navigation on Workspace

- CTRL + scroll mouse to zoom in/zoom out
- CTRL +, CTRL + Page Up to zoom in
- CTRL -, CTRL + Page Down to zoom out
- Scroll mouse to move up/down on the Workspace
- SHIFT + scroll mouse to move to the right/left on the Workspace
- Holding down the middle button to move on entire page/Workspace

Other tips:

- Possibility to move shapes on the Workspaces using the keyboard arrows
- Possibility to resize shapes of objects on the Workspace: SHIFT + keyboard arrows

## Workspace Right-Click Options

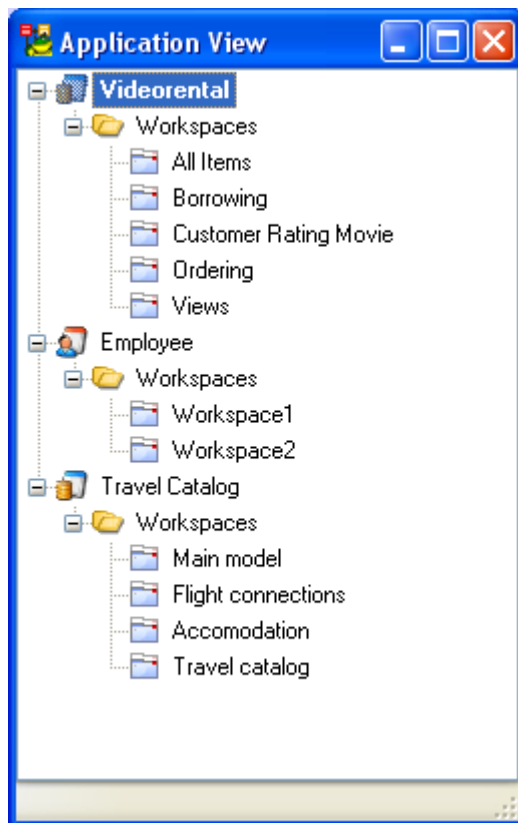
*Right-click the work area in the Application Window to see the following options:*

Option	Description
Edit	Opens the <b>Workspace Properties</b> dialog.
Workspace Format	Opens the <b>Workspace Format</b> dialog.
Add All Model Objects to Workspace	Adds shortcuts of all objects of the model to the selected WS.
Add Selected Objects to Workspace	Opens a dialog where you can select shortcuts of objects that you want to add to the WS.
Straight All Lines	Straights all relationship lines in ERD on particular Workspace. (This option is disabled in the Right Angled mode.)
Change Lines to Right Angled	Changes all relationship lines in ERD on particular Workspace from straight to right-angled.
Select Objects	Selects objects by category or schema/owner on Workspace.

## Application View

- All currently opened models and their Workspaces are listed here.
- Via the Application View, you can manage your models (activate model, save, close, open the **Model Properties** dialog and enter description etc.), edit their names (F2 or Rename), and work with Workspaces (add WS, open Designer, edit WS format, add shortcuts to selected WS etc.)
- In the Application View, also versions and revisions saved in Version Manager (VM) are listed provided that you selected **Check Out** for particular version/revision in VM. So, you can manage them from the Application View or Version Manager. (Note: Version Manager is available after Expert Mode is enabled.)
- You can open several instances of the Application View, e.g. for each model.
- By default, it's docked next to the Message Explorer on the left side, however, you can dock it also in other places.
- To hide/show the Application View, click the splitter in the middle of the pane "Application View, Message Explorer".
- You can close the Application View (x) and open later (select **Tools | Application View**).

### Example:



## Explorers

In the default look of the application, you can see the following Explorers:

- Model Explorer
- Message Explorer

### Model Explorer

In Model Explorer, Workspaces, Designers and all objects of your model, including dictionary types, domains, procedures etc. are listed. The content of Model Explorer depends on model type (physical/logical) and database. You can manage items in Model Explorer very easily.

Model Explorer is docked by default on the left side of the application. You can change its position, however it can be docked only to appropriate model on the Application Window.

See "Model Explorer" (page 240) for more information.

### Message Explorer

In Message Explorer, all hints, errors, warnings and other messages that appear during your work with Toad Data Modeler are displayed.

In Toad Data Modeler, frames e.g. for reverse engineering or script generation, have their own log area too. However, their log area displays only the information on the particular operation proceeding in the dialog/wizard (**Show/Hide Log** options are available.) Message Explorer displays all information on all operations in the application.

See "Message Explorer" (page 243) for more information.

In Toad Data Modeler, there are also other two Explorers - **Script Explorer** and **Package Explorer**. They are available only in Expert Mode. These Explorers are determined for customization of the application and shall be used by advanced users. More details can be found in the "Customization - Sample" chapter.

## Shortcuts of Objects

### About Shortcuts of Objects

In Toad Data Modeler, you can create multiple representatives of an object on one Workspace or more Workspaces - **shortcuts of objects**.

You can create shortcuts of the following objects:

- Entity
- Relationship
- View
- Materialized View
- Inheritance
- Note
- Stamp
- Category (caption of Category)

You can use shortcuts:

- For large models
- While working with Workspaces

### What Is A Shortcut?

- A shortcut is another representative of an object. It is **not** a copy of an object.
- Objects that you can see on Workspaces of your model are shortcuts in fact.

**Tip:** As soon as you create a new entity on the WS, you automatically create a shortcut of this entity on particular WS. -> You can check it out easily: Find the entity in Model Explorer and open its **Shortcuts** folder.

- A shortcut has neither higher, nor lower priority than its object and vice versa.
- A shortcut has the same name as its object. Shortcuts are numbered.
- A shortcut has the same properties as its object. You can edit ANY shortcut - all changes will be applied to all other shortcuts automatically and so will in the original object.
- An object can have none or many shortcuts. Their number is not limited. Also, there is no need to have any shortcut of an object displayed on any WS.
- You can set individual format for each single shortcut of the same object, even o in one WS.

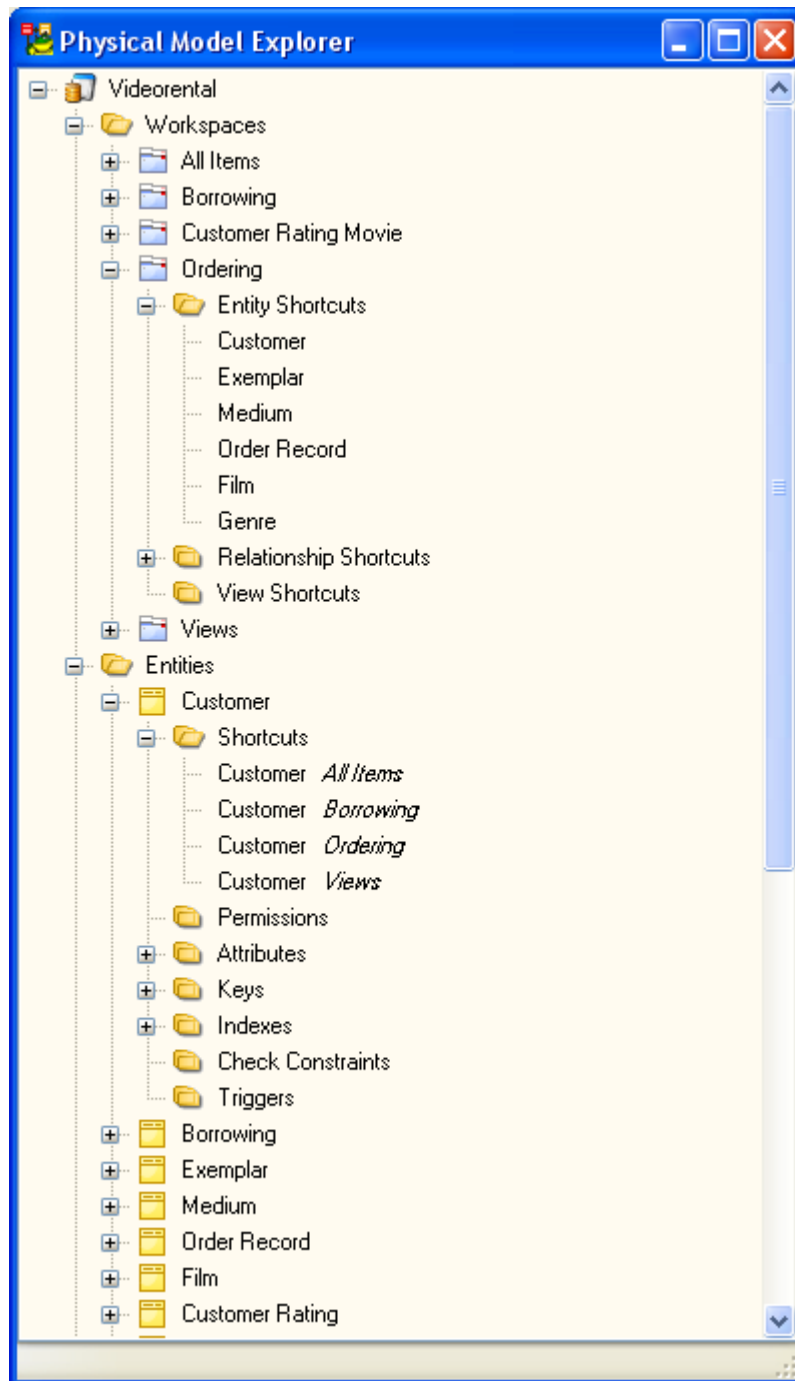
- You can remove shortcuts from Workspace easily - press Delete. If you need to delete particular object from your model completely, including all its shortcuts, press SHIFT + Delete (also **Objects** | **Delete Object**).

**Note:** To learn about the Delete options in Toad Data Modeler and differences between remove shortcut and delete object, please see the Help file, "Modeling", "Model Objects", "Deleting Objects".

## List of Shortcuts

You can find shortcuts of entities, relationships, views and inheritances in **Model Explorer** | particular Object folder (e.g. **Entities**) | **Shortcuts** folder.

List of shortcuts that are contained on particular Workspace can be found in **Model Explorer** | **Workspaces** folder | particular workspace folder (e.g. **Ordering**) | **Entity Shortcuts, Relationship Shortcuts, View Shortcuts**.



## Use Shortcuts

There are several ways how to create shortcuts of objects. Some of them are described in the following examples.

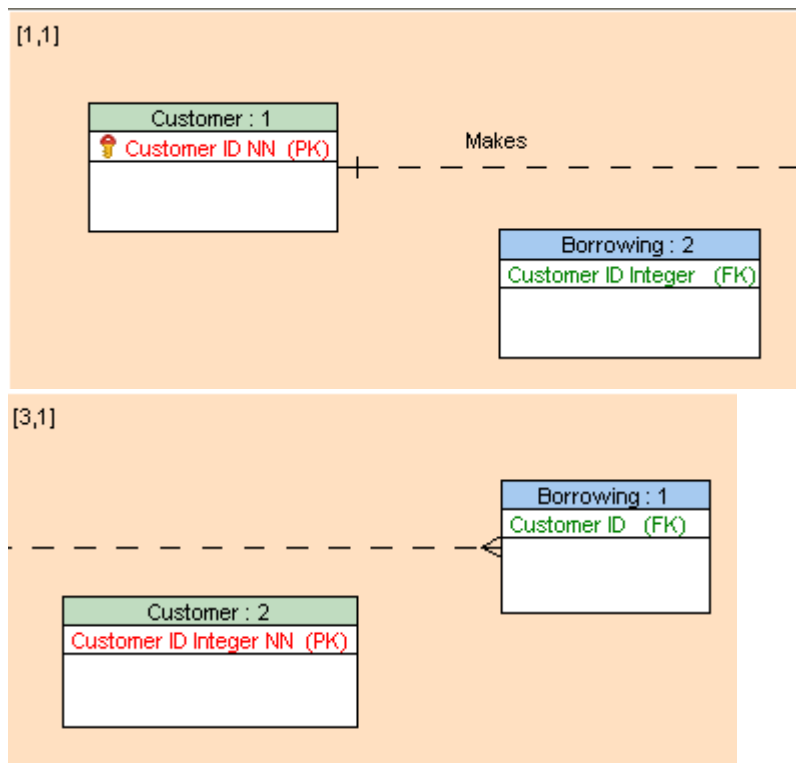
### Example 1:

#### Scenario

Entity *Customer* (placed on page [1,1]) is related to entity *Borrowing* that is placed far from it (placed on page [3,1]). Both entities are displayed on the same Workspace. You want to see the two entities closer to each other.

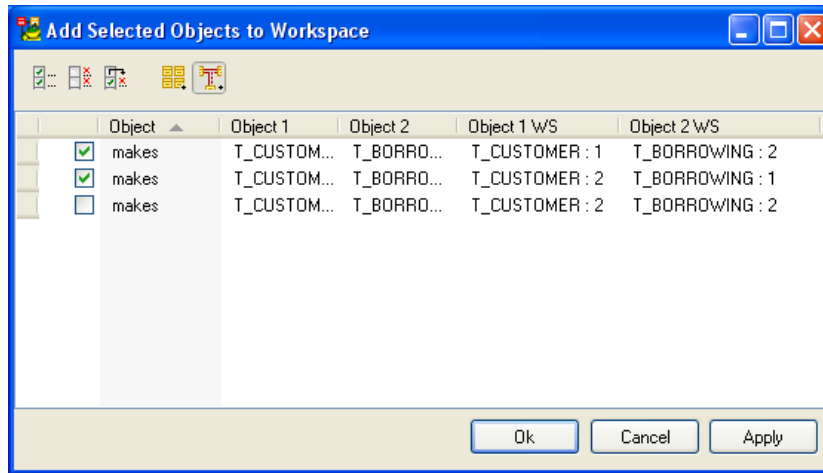
**Scenario:** Create shortcuts for both entities and place them closer to each other.


1. Click the *Customer* entity on the Workspace and hold the mouse key down.
2. Press CTRL + SHIFT keys and hold them down.
3. Drag the *Customer* entity next to the original entity (the mouse cursor will change its appearance).
4. Release the mouse button and then the keys. -> *Customer : 2* shortcut has been created.
5. Take the same steps for entity *Borrowing*.
6. Move the entity shortcuts where you need.



**Scenario:** You think it would be fine to connect the shortcuts with the relationship *makes* and see it on both pages. -> Create shortcuts of the relationship.

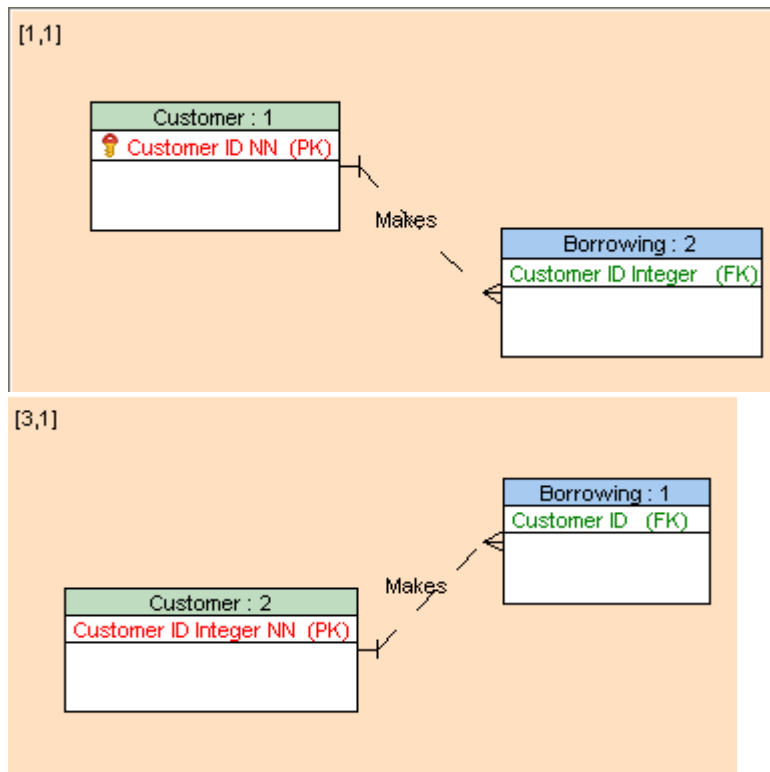
7. Right-click the Workspace and select **Add Selected Objects to Workspace**.



Click **Links**  to display only relationships of which shortcuts can be added to the selected WS.

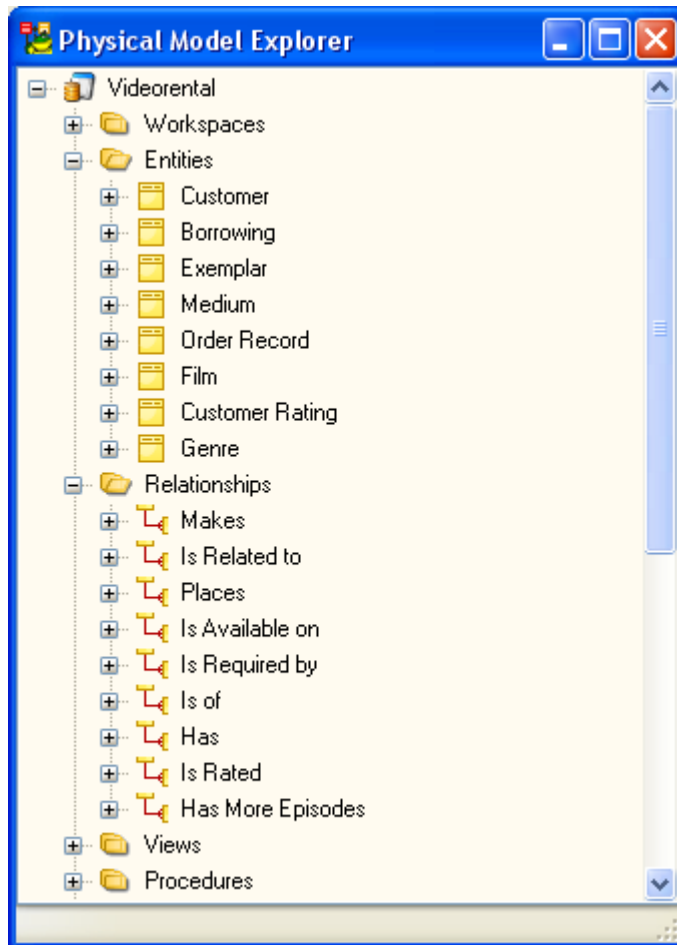
8. See the columns Object 1 WS and Object 2 WS for the *makes* relationship, and select the appropriate items.
9. Confirm **OK** to add two other shortcuts of the *makes* relationship to the WS.
10. Now the original shortcut of the *makes* relationship is useless. Let's remove it. Simply select it on the WS and press **Delete**.

**Result:**





**Note:** Still there is only one entity *Customer*, one entity *Borrowing* and one relationship *Makes* in your model. Feel free to check it out in **Model Explorer** or select **Model | Entities/Relationships**.



#### Example 2:

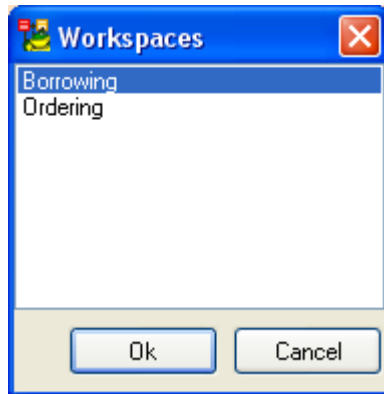
##### Scenario

You have divided your large model into several Workspaces - *All Items* WS, *Borrowing* WS and *Ordering* WS. You need to have entity *Customer* on each of these Workspaces.

You can create shortcuts of this entity on each WS. There are several ways how to do it. Feel free to select the most acceptable method for you.

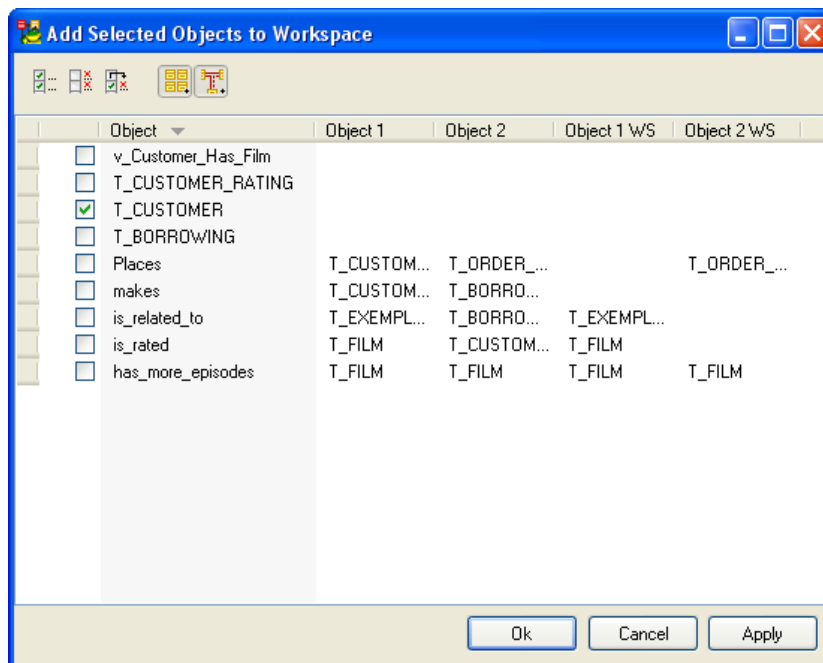
**Scenario A:** Starting point: The *Customer* entity shortcut is displayed only on the *All Items* WS. You need to add it to other two WS.

1. Right-click the *Customer* entity on the *All Items* WS and select **Add into Workspace**.
2. In the **Workspaces** dialog, select the *Borrowing* WS and confirm **OK**.



**Scenario B:** Of course, you can take the same steps for adding the entity to the *Ordering* WS. However, let's check out another way:

3. Activate the *Ordering* WS in the Application Window.
4. Right-click the WS and select **Add Selected Objects to Workspace**.
5. In the dialog, all entity and relationship shortcuts that you can add to this WS are listed. Select table *Customer* and confirm **OK**.



**Scenario C:** Starting point: You have Model Explorer opened, and WS where you want to add the entity is active in the Application Window.

6. Find the *Customer* entity in **Model Explorer** | **Entities** folder.
7. Drag the entity to the Workspace.

**Tips:**

- This way you can create shortcuts of more entities at one jump. Use SHIFT or CTRL for multiple selection of entities and then take the same steps as above.
- Have a look at the **Shortcuts** folder of the *Customer* entity in Model Explorer. See its shortcuts and names of Workspaces where they occur.

## Remove Shortcuts from Workspace

### *To remove a shortcut from Workspace*

- » Select the shortcut and press **Delete** (also **Objects** menu | **Remove from WS**).

### *To delete an object with all its shortcuts from your model*

- » Select any shortcut of the object on the WS and press SHIFT + **Delete** (also **Objects** menu | **Delete Object**).

## Options for Entity Shortcuts

*Right-click an entity shortcut in Model Explorer or on the Workspace to see the following options:*

Option	Description
Fill Parent Objects	Displays shortcuts of parent objects of the selected entity on the Workspace.
Fill Child Objects	Displays shortcuts of child objects of the selected entity on the Workspace.
Fill Parent and Child Objects	Displays shortcuts of parent and child objects of the selected entity on the Workspace.

## Tips and Tricks

### Command Line Parameters

You can use the following command line parameters:

- Open-File
- New-Model
- Execute-Script
- Start-Log
- Import Toad for Oracle ER diagram and Toad for Oracle Project

## Open-File

Opens file defined in command line parameters.

### Definition:

Open-File<space>-File<Delimiter>Name=<string>[,Name=<string>]\*

### Example:

TDM.exe Open-File -File:Name="My logical model.txl", Name="My physical model.txp"

## New-Model

Creates a new model

### Definition:

New-Model<space>-ModelType<Delimiter><string>[<space>-ModelName<delimiter><string>]

### Examples:

TDM.exe New-Model -ModelType "Logical Model"

TDM.exe New-Model -ModelType "Logical Model" -ModelName:"My Database Model"

## Execute-Script

Executes script stored in Toad Data Modeler. First parameter must contain ScriptName, second must contain ScriptMethodName, other parameters are parameters of the ScriptMethod.

### Definition:

Execute-Script<space>[Parameter<Delimiter><HashValues>+]<Switches>

<Switches> := -<InputScript><space>[-<OtherSwitch>]

<InputSwitch> := -TDM

<OtherSwitch> := -Silent

<HashValues> := <ScriptName>=<string>,<MethodName>=<string>[,<Name>=<string>]\*

### Examples:

```
TDM.exe Execute-Script -Parameter  
ScriptName=MyScript,MethodName=MyMethodInsideMyScript,Par1Name=Par1 Value,Par2Name=Par2 Value  
-TDM
```

```
TDM.exe Execute-Script -Parameter ScriptName=MujScript,MethodName=MojeMetoda -TDM  
-Silent
```

### Switches:

-TDM. Explanation: Script is stored in Toad Data Modeler application (Other switches will be defined later and users will be able to execute scripts outside Toad Data Modeler.)

-Silent. Explanation: Script will be executed in Silent mode and you will be able to work with the application regardless the state of the executed script.

### Start-Log

Writes messages to log. May be useful for sending reports to Toad Data Modeler developers.

### Definition:

Start-Log

### Import Toad for Oracle ER diagram and Toad for Oracle Project

```
TDM.exe Import-ToadForOracleERD -file:"NopathMovERD.erd"
```

```
TDM.exe Import-ToadForOracleERD -file:NopatMovERD.erd -  
Connection:"SERVER=OstDbServer:1521/ORCL,  
USER=movies,PROTOCOL=TNS,CONNECTAS=NORMAL,SAVEPASSWORD=1,ORACLEHOME=c:\oracle\product\1  
1,HOST=OstDbServer,SERVICENAME=ORCL, PORT=1521,LDAP=,METHOD=1"
```

### Related Topics

Integration Options (page 247)

### Hot Keys

Shortcut	Description
CTRL+N	Opens the <b>New Model</b> dialog.
CTRL+O	Opens already existing model.

Shortcut	Description
CTRL+S	Saves a model.
CTRL+W	Creates a new Workspace (WS).
F9	Opens the <b>DDL Script Generation</b> dialog.
CTRL+F9	Opens the <b>Model Verification</b> dialog.
CTRL+E	Creates a new entity.
CTRL+R	Creates an identifying relationship.
CTRL+C	Copies selected object(s).
CTRL+X	Cuts selected objects(s).
CTRL+V	Pastes copied or cut object(s).
CTRL+Z	Undo step.
SHIFT+CTRL+Z	Redo step.
CTRL+A	Selects all.
CTRL+F	Opens the <b>Find</b> dialog.
CTRL+ALT+F	Opens the <b>Find in Scripts</b> dialog.
Del	Removes selected object(s) from Workspace.
SHIFT+Del	Deletes selected object(s) from model.
ALT+O	The <b>Delete Confirmation</b> dialog - <b>OK</b> button.
ALT+C	The <b>Delete Confirmation</b> dialog - <b>Cancel</b> button.
CTRL+M	Minimizes Forms.
CTRL+ALT+S	Synchronizes metamodel.
CTRL+I	Creates Inheritance (Logical Model).
F1	Opens the Help file.
F2	Renames model/object.

Shortcut	Description
CTRL+Up	Moves selected object(s) up. (Order of Generated Objects)
CTRL+Down	Moves selected object(s) down. (Order of Generated Objects)
Keyboard arrows	Moves shapes in Workspace (select a shape first).
SHIFT+keyboard arrows	Changes size of entity box.
CTRL+scroll mouse	Zooms in/out.
CTRL+, CTRL+Page Up	Zooms in.
CTRL-, CTRL+Page Down	Zooms out.
SHIFT+scroll mouse	Moves to the right/left in Workspace.
Scroll mouse	Moves up/down in Workspace.
Holding down the middle mouse button	Moves in entire page/Workspace.
SHIFT	Multiple selection of objects (in WS, dialogs, forms, Model Explorer etc.)
Drag&Drop techniques+CTRL+SHIFT	Creates a shortcut of object in WS.
Drag&Drop techniques from Model Explorer	Creates a shortcut of object in WS.
Drag&drop techniques+CTRL	Copies objects in WS/Object dialog (e.g. Entities dialog)/Model Explorer.
CTRL+Drag&Drop techniques	Copies attributes between entities in WS.
CTRL+click an attribute	Selects an attribute to move it to another entity box in WS. (Click the attribute and release CTRL key to move the selected attribute.)
CTRL+click attributes	Multiple selection of attributes in entity box in WS.

Shortcut	Description
CTRL+double-click an object name	Renames the object name in WS (including attributes).
CTRL+click a relationship line	Adds a handle point.
CTRL+click a handle point	Deletes the selected handle point.
ALT+click a relationship line	Selects a relationship line segment.

## Dock Panes

Feel free to customize layout of Toad Data Modeler at your convenience.

You can dock panes to one or more places, or left undocked. After you re-arrange the interface, the layout will be saved automatically after you close the application.

There are two kinds of forms/panes:

- Big dockable forms (Designer, Script Editor etc.)
- Small dockable forms (Message Explorer, Application View etc.)

You can dock forms/panes:

- One on another
- One by another
- One below another
- On tabs
- On the Application Window
- Combination of these options

If you dock forms one on another, you can dock only small forms on big ones or big forms on big ones.

### *To dock a form/pane*

1. Press SHIFT and hold it down.
2. Click the top of the form (pane) and drag it to the required position. All available positions will be shown in a frame.
3. Release the mouse button and then SHIFT to dock the form.

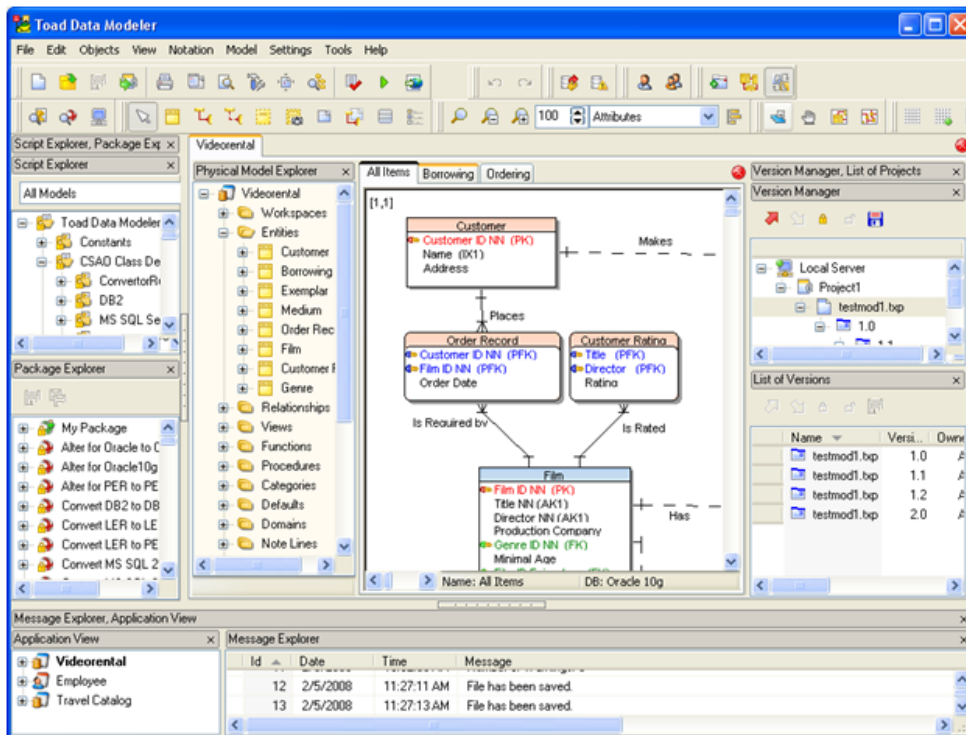


### *To undock a form/pane*

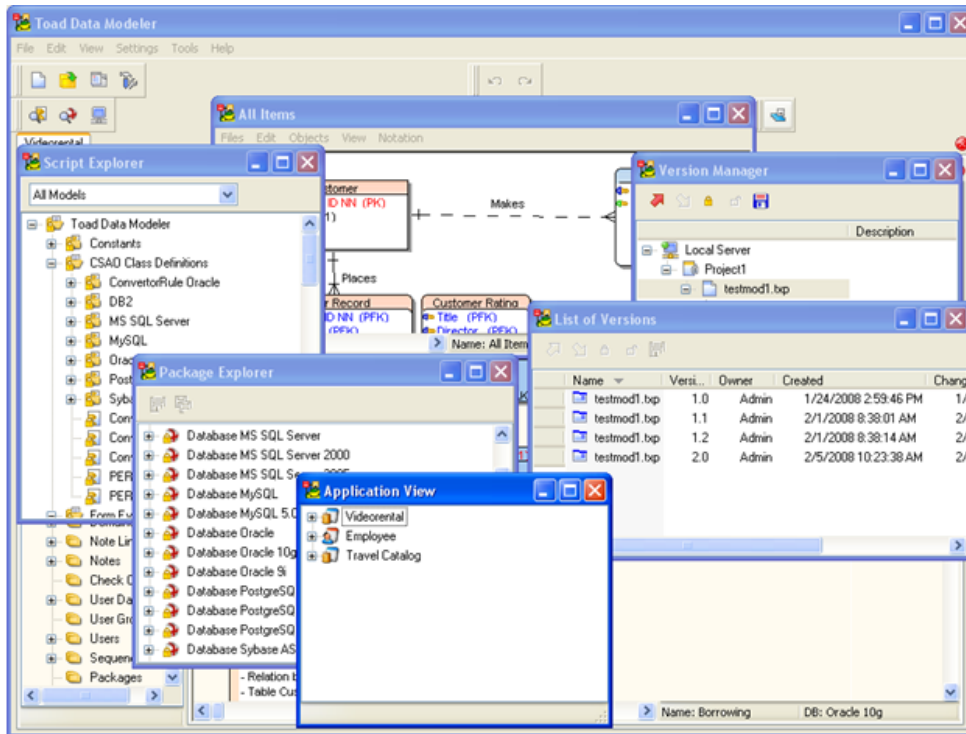
- » Press SHIFT and double-click the top of the form (pane).

### **Example:**

Docked panes



Undocked panes



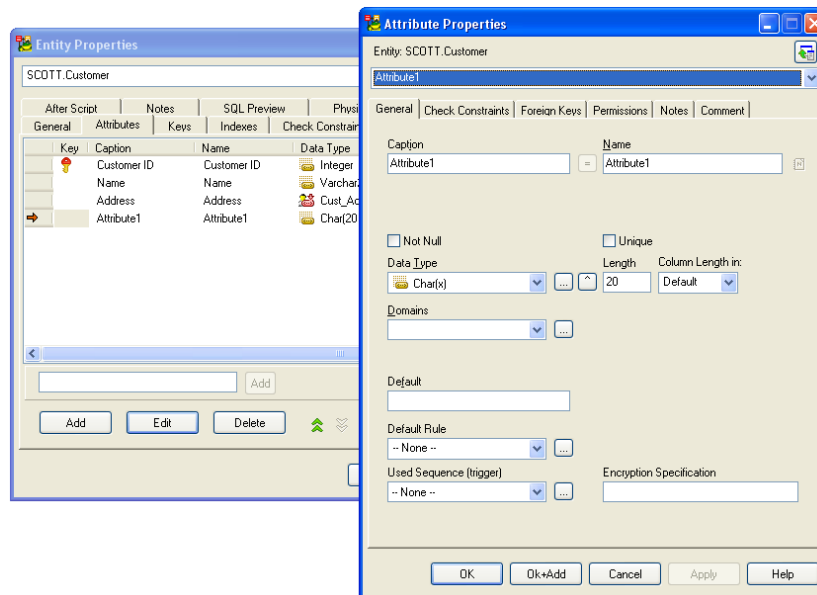
## Modeless Dialogs/Forms

In Toad Data Modeler 3.x, you can open several dialogs/forms at one time, switch among them and work with them concurrently without the necessity to close any of them. This allows you to work with your model fast and comfortably.

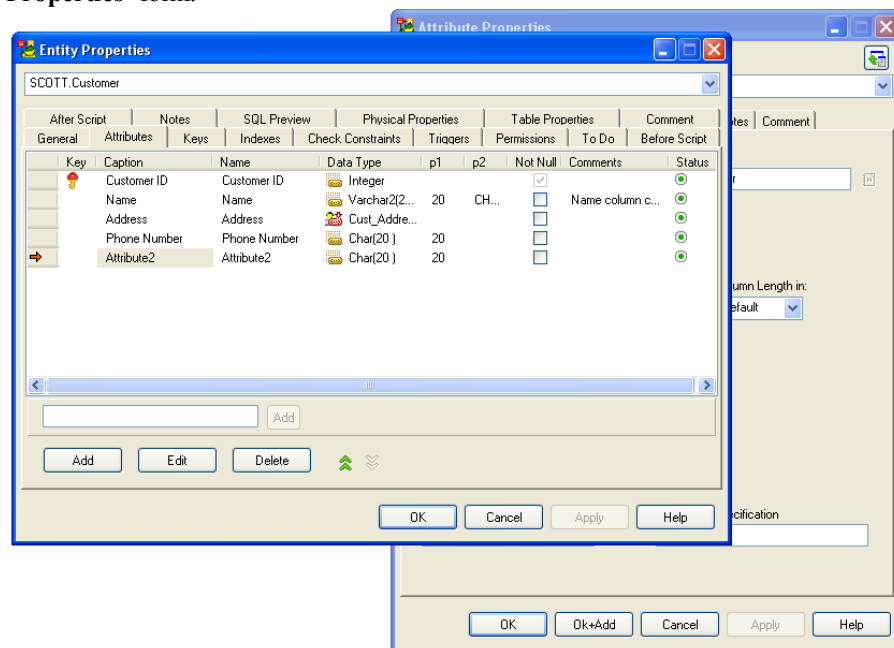
### Scenario

You've created a new *Attribute1* for an entity and started to define its properties. It is probable that you will need to add another attribute (or other attributes) later.

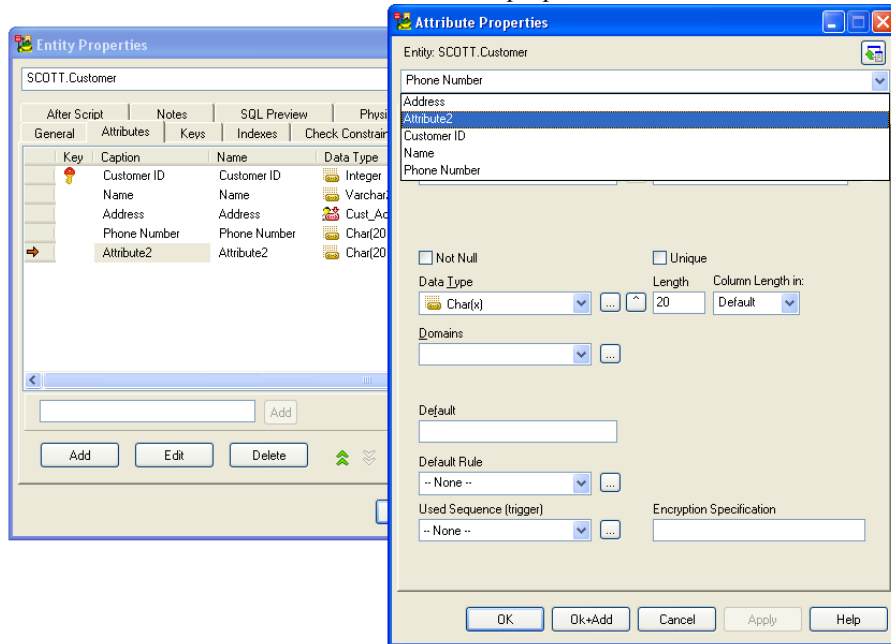
1. Select the newly created *Attribute1* in the the **Entity Properties** form and click **Edit**. The **Entity Properties** form remains opened and the **Attribute Properties** dialog opens.
2. Define properties of the *Attribute1*.



- Now you find that you need to create another attribute. You don't have to close the **Attribute Properties** dialog but can simply click the **Entity Properties** form to activate it and add a new attribute there - click **Add** in the **Entity Properties** form.



- Click the **Attribute Properties** dialog again. From the Object Navigator box, select the new item *Attribute2* to define properties of this attribute.



#### Tips:

- For even quicker adding new attributes, you can use the **OK+Add** button in the **Attribute Properties** dialog.
- While defining properties of an attribute, you can close the **Entity Properties** form. Later, if you find you need to create another attribute, you can:
  - Use **Model Explorer**: right-click the **Attributes** folder of appropriate entity | **Add Attribute**. The newly created item will display in the Object Navigator box in the still opened **Attribute Properties** dialog immediately. Feel free to select it and define its properties.
  - Open the **Entity Properties** form directly from the **Attribute Properties** dialog.

Just click .

**Note:** Another great example of using modeless dialogs is e. g. defining Users and User Groups. ( See "Users and User Groups" (page 100) for more information.

#### Instances of Forms

Toad Data Modeler also allows you to open several instances of the same dialog/form. To enable this option, select **Settings | Options | Application** section | **General** and clear the **Allow One Instance of Form** option. (The change will be applied also for currently opened model.)

What are the advantages of this feature? - For each instance of form/dialog, you can set a different view, dock them where you want, and undock or close/open when necessary.

#### Tips:

- **Application View** - You work with two models that have a lot of Workspaces. You need to see all WS for each model. -> You can open AV for each model.
- **Version Manager** - Open Version Manager for each project for more comfortable work with them.
- **Message Explorer** - Set up different view filter on messages in each Message Explorer.
- **VM, ME, Script and Package Explorer** - Dock the forms to the models where you need them or use them. Set particular view on items in the forms for each model to meet your needs.

**Example 1:** You work with scripts and packages only in *Model X*. Feel free to dock Package and Script Explorers to the model. -> It saves place while working with other models. The packages will be available only for *Model X* - just where you need them.

**Example 2:** Open several instances of Message Explorer - for each model. Dock it on particular models and set a different message view filter for each.

## Inplace Editor

Inplace Editor is another of the great features that can speed up your modeling or simply make it more comfortable.

This feature allows you to edit items directly in a certain place (Workspace, frame), without the necessity to open appropriate **Properties** dialog. See the examples below.

### *To change an entity name directly on the Workspace*

1. Select the entity on the Workspace.
2. Press CTRL and double-click the entity name.



3. Change the name and then click the work area.

**Note:** This way, you can change names of entities, attributes, indexes, relationships, views, categories, stamp items etc.

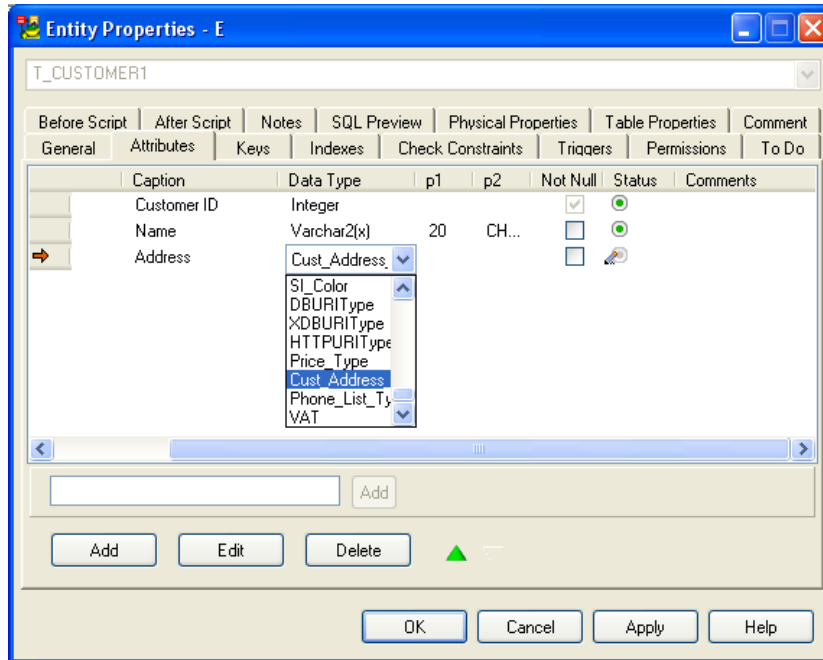
### *To edit attribute properties directly in the entity frame (Entity Properties form)*

1. Select the item that you want to edit and press **F2**.

**Note:** You can also click the item that you want to change, wait a second and

click it again.

2. Change the value and click somewhere in the grid.
3. Confirm **Apply**.



## Object Navigator Box

Object Navigator box, together with the modeless dialogs, can ease and speed up your work in Toad Data Modelersignificantly.

Object Navigator box:

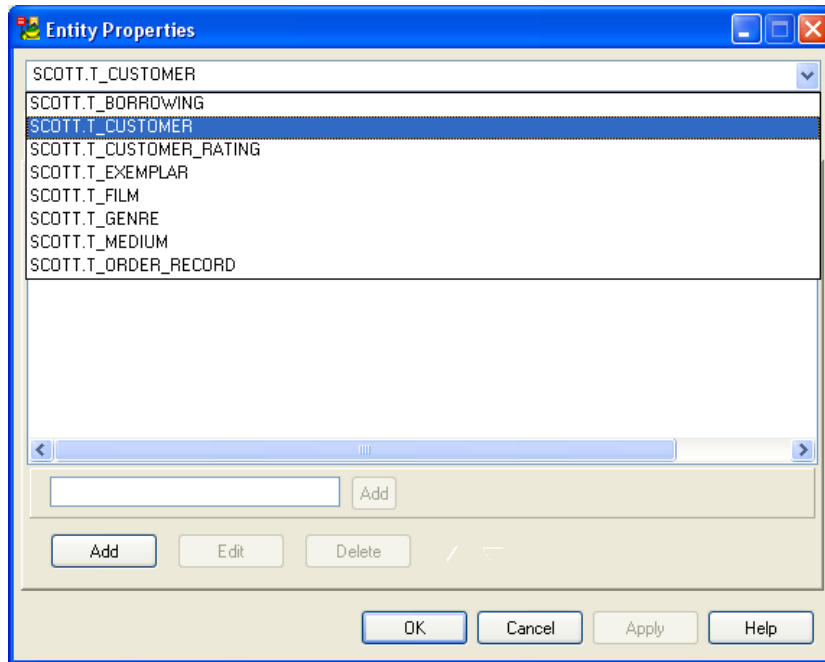
- Is available in object **Properties** dialogs and forms (entity, attribute, check constraint, view, function, users, user groups etc.).
- Allows you to select items for edit from **one (the same) place**.

### Scenario

You need to edit several entities of your model.

1. Double-click ANY entity on the Workspace (WS) to edit it.
2. From the Object Navigator box, select the entity you need to edit (*Borrowing*).
3. Change its properties and confirm **Apply**. -> The changes will be saved and the **Entity Properties** form remains opened.

4. Again, from the Object Navigator box, select another entity you need to edit (*Customer*). Confirm **Apply** to save the changes.
5. Take the same steps until you edit all entities you need.
6. For the last entity, you can confirm the changes **OK** to close the **Entity Properties** form.



Feel free to take advantage of this box when editing relationships, procedures, views, functions, users etc.

**Tip:** Quick search in the Object Navigator box - To find e.g. particular entity in ERD to edit it, double-click any entity on the Workspace, expand the Object Navigator box in the **Entity Properties** form and simply start typing name of the searched entity. The searched item will be found in the list. Click it to open its Properties form.

## Object Viewer

Toad Data Modeler allows you to display a complete list of objects, e.g. attributes, triggers, check constraints, views, etc. of your model with a possibility to open the **Properties** dialog of the selected object.

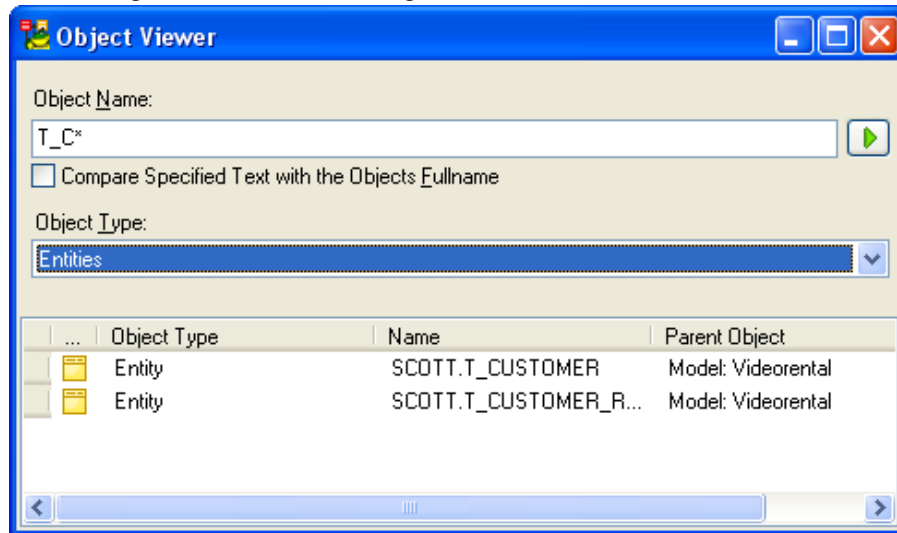
### *To open Object Viewer*

- » Select **Objects | Object Viewer**

**Tip:** You can also open the Object Viewer from Model Explorer or the Application View - just right-click the model name.

### *To view a particular object of your model*

1. In the **Object Viewer** dialog, write an object name (case sensitive). If you want to view all objects of certain type, leave \*.
2. If you want to compare the text defined in the **Object Name** box with fullname of objects (which is e.g. in Oracle database Schema name + Object Name), select the **Compare Specified Text with the Objects Fullname** checkbox. Then only objects where the text defined in the **Object Name** box matches the fullname will be listed.
3. Select an object type.
4. Press the green button on the right.



5. Double-click the selected object to open its **Properties** dialog.

## Search Objects

While working with your model, it can happen that you need to find an object on the Workspace quickly (primarily when your model is large and has several Workspaces). Toad Data Modeler offers you several options how to do it.

### Find Function

**Find** function allows you to find an object on particular Workspace. So, to be able to use this function, you need to know in which Workspace (WS) the object occurs.

1. Activate the WS in the Application Window.
2. Select **Edit | Find** (also CTRL + F).
3. Write name of the searched object to the **Object Name** box. If you don't remember the whole name, you can use some of the following search strings: \*film or only f\* etc.

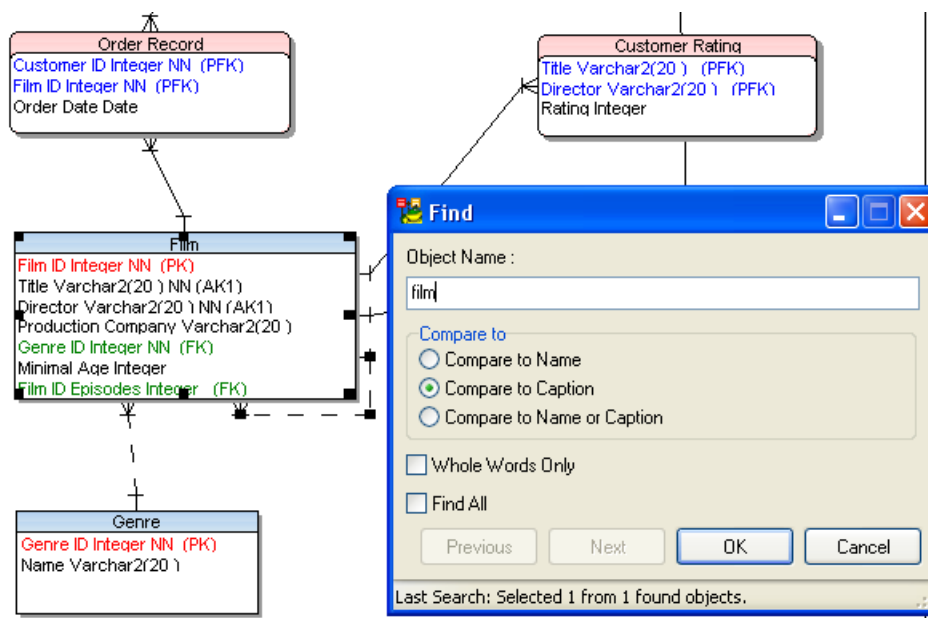


\* - for unlimited number of characters,

? - for one character.

**Note:** Remember that **Compare to Name** refers to a physical view (searches among physical names on the WS) and **Compare to Caption** to a logical view (searches among logical names on the WS). See what view is displayed on the WS or select the third option for sure - **Compare to Name or Caption**.

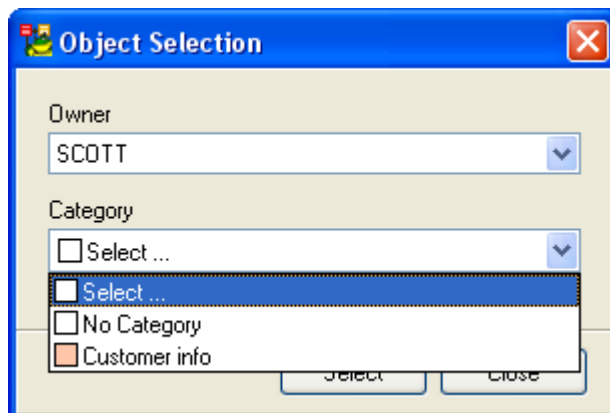
Result: The searched object will be highlighted on the Workspace.



4. Click **Cancel** or x to close the **Find** dialog.

## Select Objects in ERD by Schema or Category

1. Right-click the active Workspace and select **Select Objects**.



2. Define a category or owner or category and owner and click **Select**. All objects that match your setting will be selected on the Workspace.

## Model Explorer

In Model Explorer, shortcuts of entities, views and relationships are listed clearly. You have two options where to find them:

- *In the branch of an object:* e.g. **Entities** | 'Entity Name' | **Shortcuts**. -> Here, all shortcuts of the selected entity with a name of Workspaces where they occur are listed.

**Note:** Double-click the entity name opens the **Entity Properties** form. So, if you want to find the entity on the WS, you need to unfold the **Shortcuts** folder. Here you can double-click the shortcut name or right-click and select **Find Object**.

or

- *In the branch of a Workspace:* **Workspaces** | 'Workspace Name' | **Entity Shortcuts/Relationship Shortcuts/View Shortcuts**. -> Here, all entity/relationship/view shortcuts that occur on the selected WS are listed.

**Note:** Double-click the shortcut name (or right-click | **Find Object**) finds and highlights the shortcut on the WS.

## Search Items in Trees

The quick search function is available:

- In all trees in Toad Data Modeler - e.g. in the: Application View, Model Explorer, Version Manager, Script Explorer, Package Explorer etc. in **all expanded** folders of the tree (the search does not go through folders that are collapsed).
- In the Object Navigator box in **Properties** dialogs.

This function allows you to find e.g. particular entity in Model Explorer, particular project or file in Version Manager, model or Workspace in the Application View, script in Package or Script Explorer etc.

Simply start typing the item name and it will be found in the tree/list at once.

## Other Quick Tips

### Objects on the Workspace and Keyboard Arrows

- Move entities on the Workspace via keyboard arrows.

**Tip:** To set up the size of a step to move, select **Settings** | **Options** | **Graphics** |

### Move Objects by (mm/10) (in tenths of milimeters).

- Select an entity, press SHIFT, hold it down and use the keyboard arrows to change size of the entity box.

## Navigation on Workspace

- CTRL + scroll mouse to zoom in/zoom out
- CTRL+, CTRL + Page Up to zoom in
- CTRL-, CTRL + Page Down to zoom out
- Scroll mouse to move up/down on the Workspace
- SHIFT + scroll mouse to move to the right/left on the Workspace
- Holding down the middle button to move on entire page/Workspace

## Make a Copy of Multiple Objects on the Workspace (CTRL+A, CTRL+C, CTRL+V)

- Before you press CTRL + V to paste the objects, close the **Model Explorer** dialog to accomplish the operation much faster.

(The larger your model is, the more significant difference in speed you will notice.)

## Print Models

- In **Settings | Options | General**, clear the **Print Gradients** checkbox for much faster print performance. (It is disabled by default.)

## HTML Report Layout

- For large models, **Frames - top menu** or **Frames - left menu** options are recommended.

(Frameless report layout is not recommended as it uses Java script that goes through all objects, which takes too much time if your model is large.)

# Physical Data Model

## About Physical Model

Toad Data Modeler allows you to design Entity Relationship Diagrams of particular database platform, convert logical data model to physical model, load ER Diagram directly from your database (Reverse Engineering feature, loading of DDL scripts), update physical models, generate DDL/SQL scripts and also alter scripts, create Dictionary Types, Views, Triggers, Functions, generate detailed documentation to your model (HTML, RTF, PDF, XSLT) etc.

All this and much more relating to Physical Data Modeling is contained in this chapter.

**Note:** In the installation package, you can find a sample physical model *Videorental* (Oracle 10g db). Default location: C:\Program Files\Quest Software\Toad Data Modeler 3\Samples.

Let's create a new physical ER diagram in Toad Data Modeler from scratch.

### Related Topics

Select Target Database (page 48)

About Reverse Engineering (page 169)

Load Models from DDL/SQL Script (page 182)

Import Toad Data Modeler 2.x Model (page 201)

Import Toad for Oracle ER Diagrams (page 250)


## Create Entity Relationship Diagram

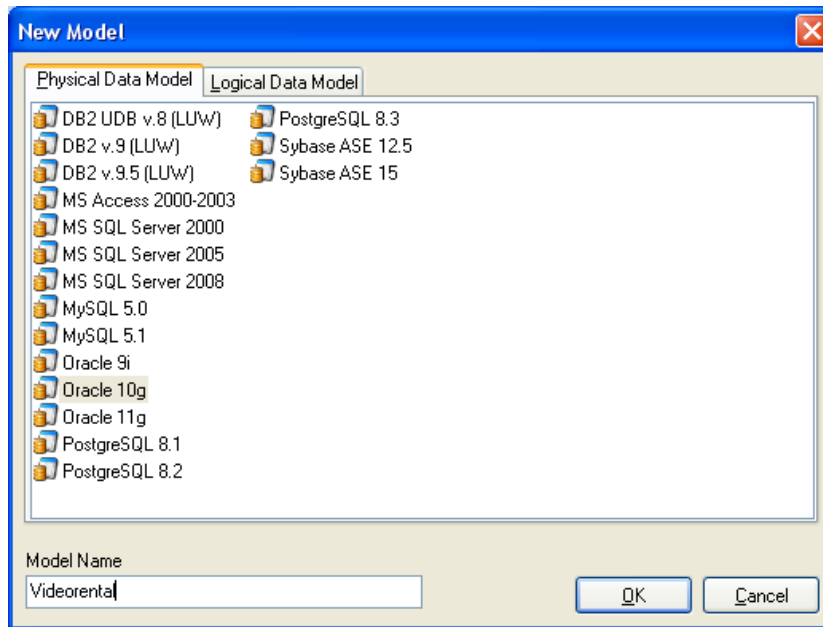
### Select Target Database

Toad Data Modeler supports several database systems and also specifics of the particular database system. Therefore, when you want to create a new ERD from scratch, you have to select a target database first.

#### Scenario

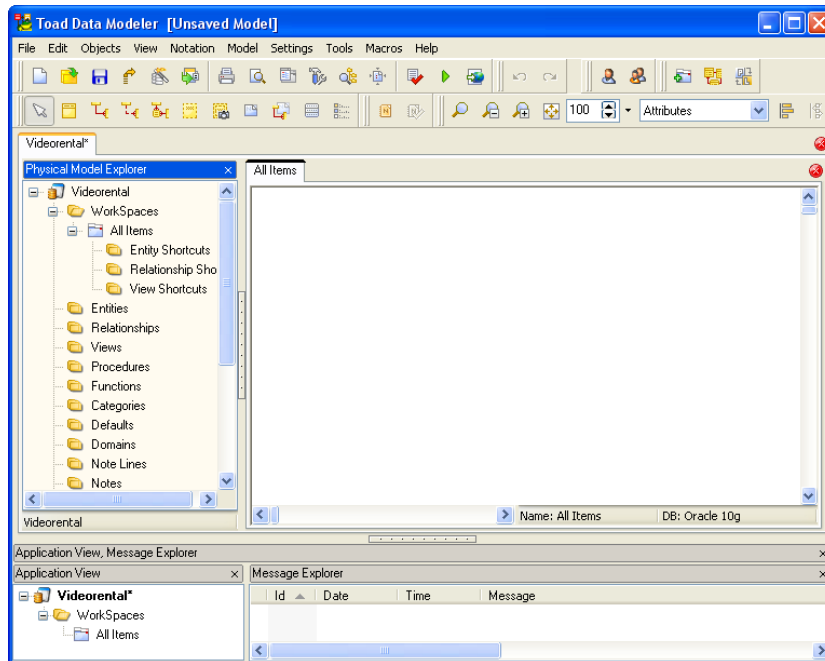
Create Oracle 10g physical model *Videorental*.

1. Click  on the toolbar (also CTRL+N).
2. Click the **Physical Data Model** tab and select a target database - *Oracle 10g*.
3. To the **Model Name** box, write the model name *Videorental*.
4. Confirm **OK**.



Result:

- The model will be listed in the **Application View**.
- In the **Application Window**, designer for the Workspace *All Items* will open automatically.
- The status bar in the Application Window displays the database name. (*DB: Oracle 10g* in our example.)
- Main menu and the Designer toolbar for physical data model will activate.



#### Notes:

- **Database Name** - A database for which the model is created (e.g. Oracle 10g). The database name information can be found at the bottom of the Application Window, or is displayed in the pop-up hint when you point your mouse cursor at the Model Name in the Application View.
- **Model Name** - A model can be saved to more files. In Toad Data Modeler, Model Name should be understood as a title of a document that can be saved to several files of a different name.
- **File Name** - A name of file where the model is saved. File Name is defined after you select **Save Model** or **Save Model as**.
- See "Rename" (page 222) for more information.

#### Related Topics


Create Entities (page 50)

## Create Entities

In Toad Data Modeler, there are several ways how to create entities - on the Workspace, via **Model Explorer** and in the **Entities** dialog (**Model | Entities**). All the options are described in the Help file. Here, you will create entities directly on the Workspace.

### Scenario

Create entity *Customer* on the Workspace in your *Videorental* model.

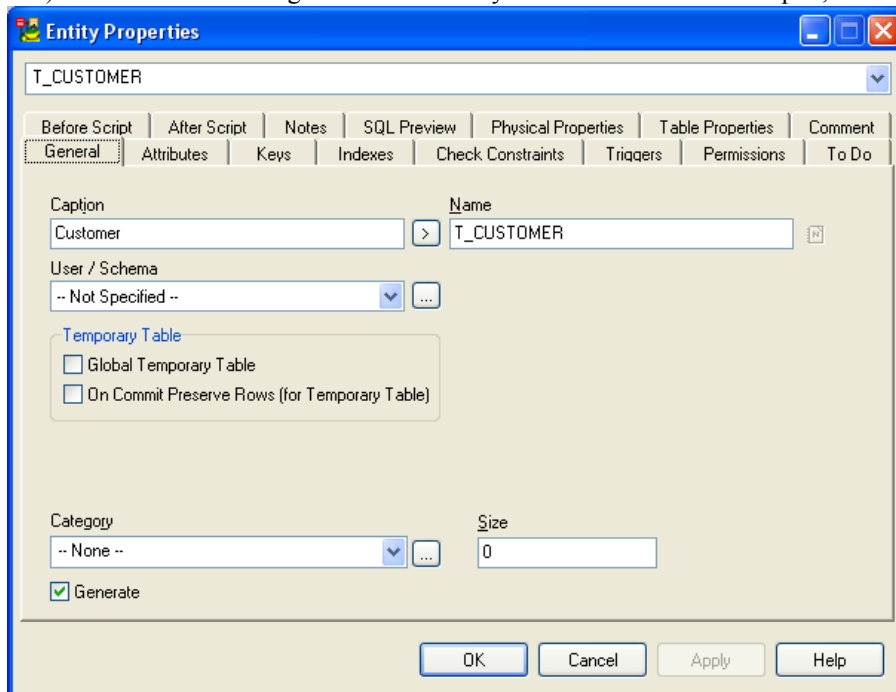
1. Click  on the toolbar (also CTRL+E) and then click anywhere on the work area.
2. Double-click the entity to edit it.
3. On tab **General**, define the entity name.

Caption	Logical entity name - <i>Customer</i> .
Name	Physical entity name - <i>T_CUSTOMER</i> .

**Tip:** See "About Naming Conventions" (page 106) for more information.

**Note:** Appropriate entity name will be displayed on the Workspace according to the view you select (**View** menu | **Show Logical Names**).

4. Define other properties on tab **General** and other tabs (e.g. **Notes**, **Comments** etc.). To save the changes simultaneously and leave the form open, click **Apply**.



The image shows the 'Entity Properties' dialog box for the entity 'T\_CUSTOMER'. The 'General' tab is selected. The 'Caption' field contains 'Customer' and the 'Name' field contains 'T\_CUSTOMER'. The 'User / Schema' dropdown is set to '-- Not Specified --'. Under the 'Temporary Table' section, both 'Global Temporary Table' and 'On Commit Preserve Rows (for Temporary Table)' are unchecked. The 'Category' dropdown is set to '-- None --' and the 'Size' field is set to '0'. The 'Generate' checkbox is checked. At the bottom, there are buttons for 'OK', 'Cancel', 'Apply', and 'Help'.

## Related Topics

Create Attributes (page 52)

## Create Attributes

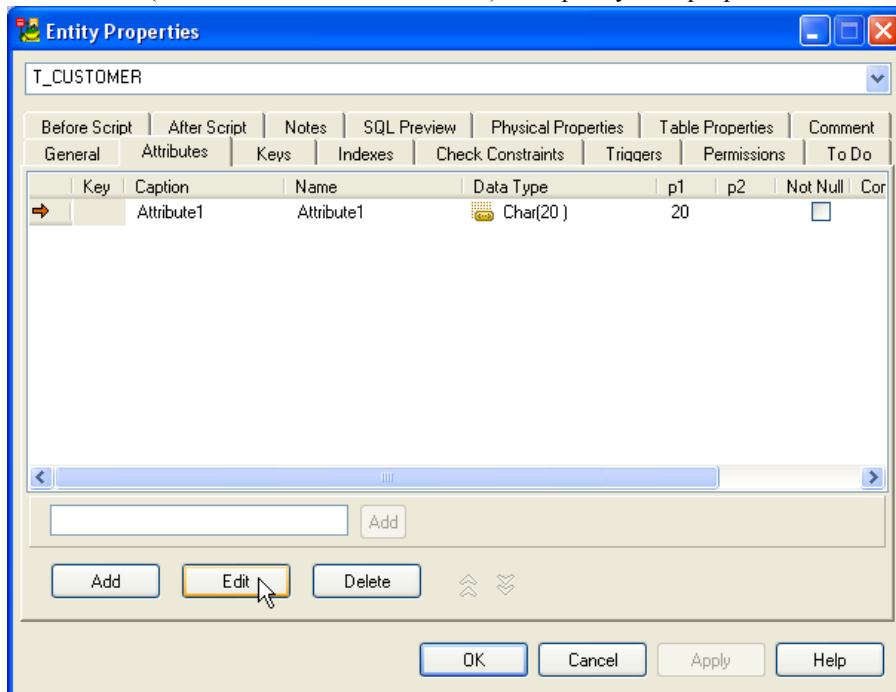
### Scenario

Create attribute *Customer ID* in the *Customer* entity.

1. In the **Entity Properties** form of the *Customer* entity, click the **Attributes** tab.
2. Click **Add** to create *Attribute1*.

**Tip:** You can also use the quick **Add** button. - Write the attribute name and click the small **Add** button then.

3. Confirm **Apply** to be able to edit the attribute.
4. Click **Edit** (or double-click the attribute) to specify its properties.





5. Define properties of the attribute on tab **General**.

Entity: T\_CUSTOMER

customer\_id

Notes | Not Null Constraint | Comment

General | Check Constraints | Foreign Keys | Permissions

Caption: Customer ID

Name: customer\_id

☒ Not Null ☐ Unique

Data Type: Integer

Domains:

Default:

Default Rule: -- None --

Used Sequence (trigger): -- None --

Encryption Specification:

OK Ok+Add Cancel Apply Help

Caption	Logical entity name - <i>Customer ID</i>
Name	Physical entity name - <i>customer_id</i> <b>Tip:</b> See "About Naming Conventions" (page 106) for more information.
Not Null / Unique	Select <b>Not Null</b> checkbox to define the attribute as not null (must have a value). <b>Note:</b> You can select the <b>Unique</b> box to define the attribute as a unique item. If you create a unique attribute, an alternate key for this

	attribute will be created automatically on the <b>Keys</b> tab of the <b>Entity Properties</b> form.
Data Type	<p>Select <b>Data Type</b> from the drop-down menu - <i>Integer</i>.</p> <p><b>Note:</b> If you selected e.g. Char(x) data type, you could define Length and Column Length in values.</p> <p><b>Tip:</b> Before you create new attributes, you can set up a default data type, including parameters, for particular database. See the <b>Settings</b> menu   <b>Options</b>   <b>Physical Model</b>   particular database version   <b>Default Data Type</b> combo-box. (The selected data type will be applied also for Dictionary Types and Domains.)</p>

6. Feel free to define other database dependent options on tab **General** and also on other tabs of the **Attribute Properties** dialog.
7. Confirm **OK**.

## Related Topics

Create Primary Key (page 54)

## Create Primary Key

There is a primary key by default in each entity - see the **Entity Properties** form | **Keys** tab. The key is empty.

You can assign a key to attribute(s) in the **Entity Properties** form on tab **Attributes** or tab **Keys**. Assigning on tab **Attributes** is quicker.

### To assign a key on the Attributes tab

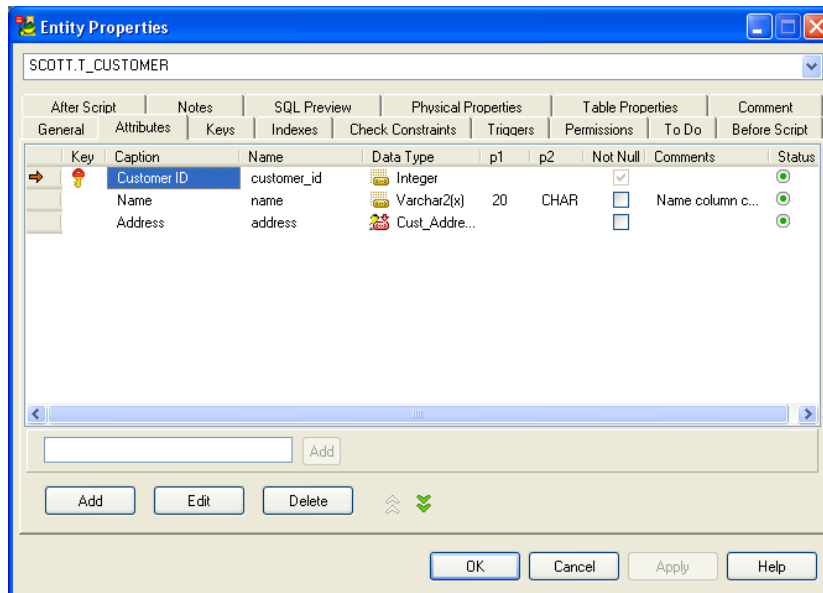
#### Scenario

You want to assign the primary key to attribute *Customer ID*.

1. Double-click the space in the **Key** column next to the *Customer ID* attribute.
2. Confirm **Apply**.

**Note:** If you need to define properties of the key now, click the **Keys** tab and

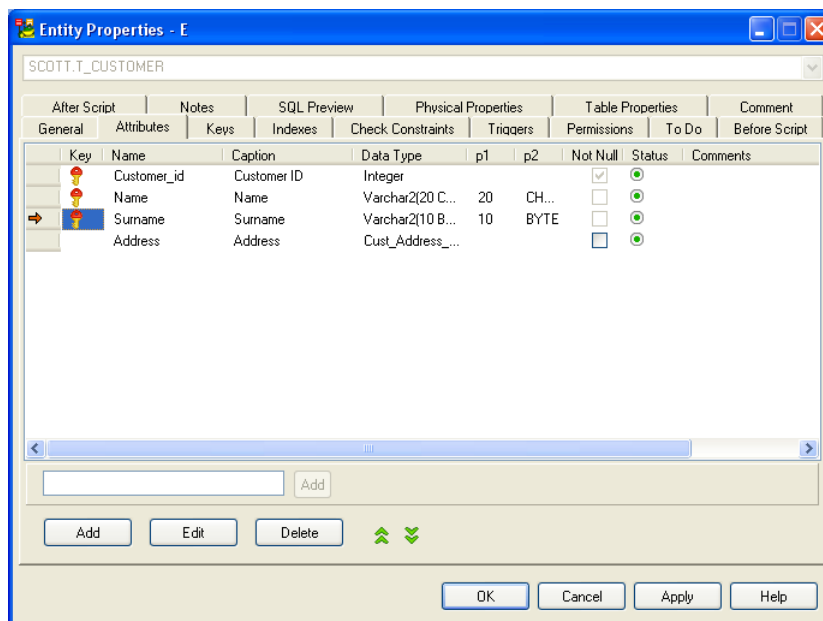
double-click the selected key.



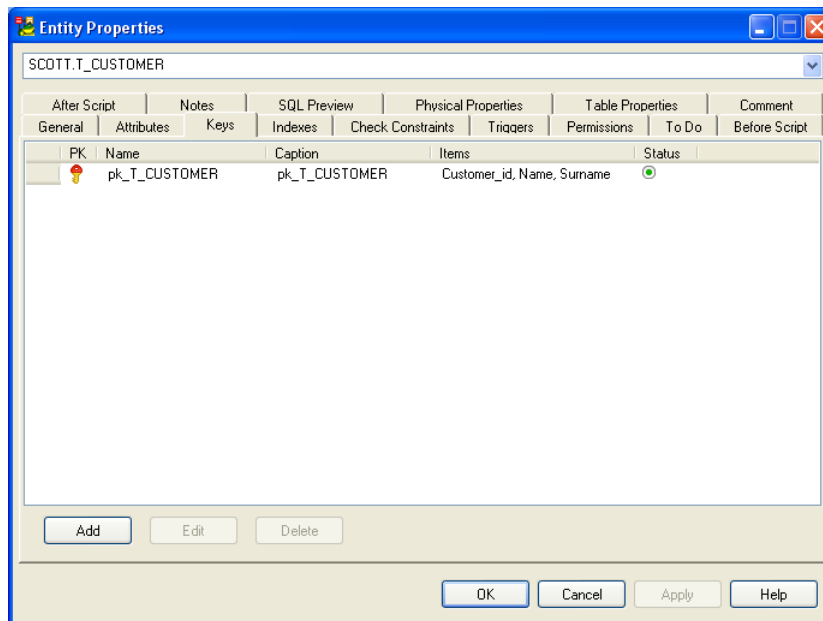
## Scenario - Assign the key to more attributes

You want to add a primary key to the following attributes (and in this order): *Customer ID*, *Name*, *Surname*.

1. Double-click the space in the **Key** column next to the selected attributes in the following order: *Customer ID*, *Name*, *Surname*.

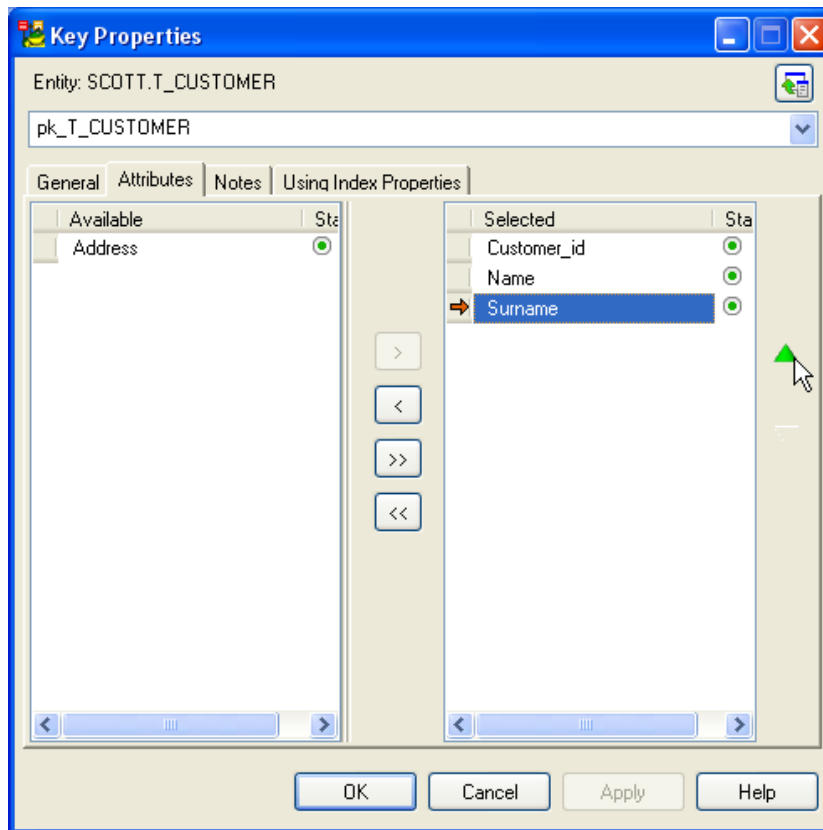


2. See the **Keys** tab | **Items** column now:



Result: The key is not empty any more. The attributes added to the key are listed in the order you added them to the primary key.

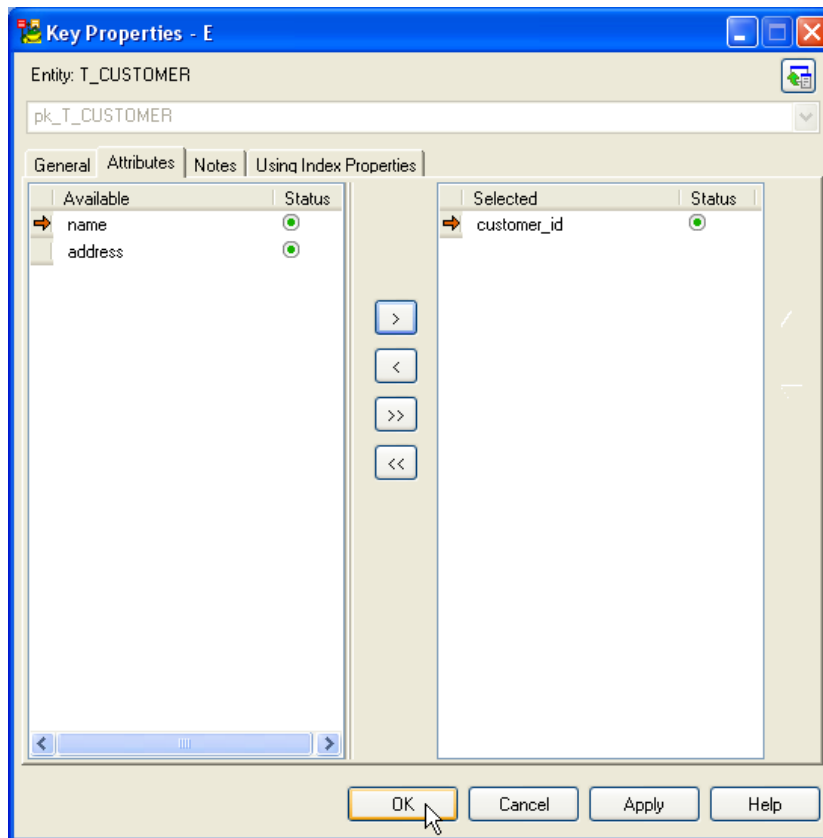
If you wanted to change the order of attributes in the key, you would need to edit the key (double-click the key on tab **Keys**). It is not possible to change the order on tab **Attributes**.



Select the attribute and use the green arrows on the right to move it up/down.

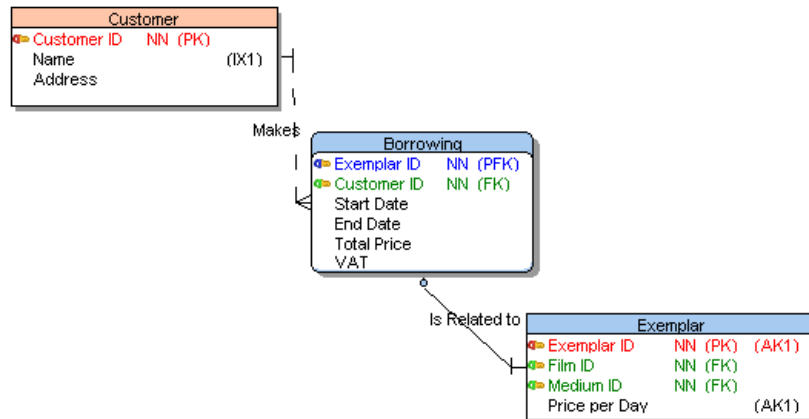
*To assign a key on the Keys tab*

1. In the **Entity Properties** form, click the **Keys** tab.
2. Double-click the empty default *Key1*.
3. Define the key properties on tab **General**.
4. Click the **Attributes** tab.
5. Select the attribute(s) and click the **Add** arrow button to move the attributes from the window **Available** to window **Selected**.
6. Confirm **OK**.



### Tips and Tricks for Keys

- Display keys in entity box in your ER Diagram: From the **Display Level** box in the toolbar, select the appropriate key item.
- Set the display level and graphical display of keys for new models *by default* in the **Setting** menu | **Options** | **Physical Model**.
- Display keys graphically in your ER Diagram:
  1. Right-click the Workspace and select **Workspace Format**.
  2. Click the **Entity** tab and select the **Display Keys Graphically** checkbox.



## Other Notes

- In the **Entity Properties** dialog | **Keys** tab, you can also create alternate keys. Simply click **Add** to create another key.
- Toad Data Modeler creates new alternate keys for unique attributes automatically on the **Keys** tab.
- It's not possible to delete a primary key. However, it can remain empty.
- It's not possible to delete an alternate key of unique attribute (the attribute would have to be deleted).
- More information on keys can be found in the Help file, "Keys" chapter.


## Related Topics

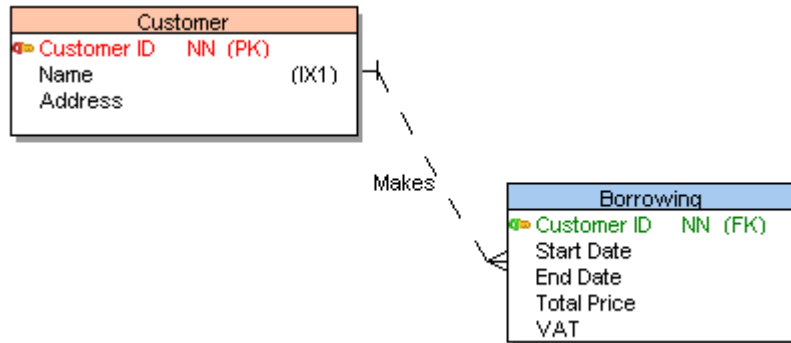
Create Relationship (page 59)

## Create Relationship

### Scenario

Create a non-identifying relationship between entities *Customer* and *Borrowing*.

1. Click  on the toolbar for non-identifying relationship.
2. On the Workspace, click the parent entity *Customer* and then click the child entity *Borrowing*.
3. Double-click the relationship to edit it.
4. Define properties of the relationship.




Specify the relationship name (*Makes*), including inverse name (*Is Made*), define cardinality, referential integrity, linking method etc.

5. Confirm **OK**. Now you know how to create a very simple entity relationship diagram in Toad Data Modeler.

When you create a relationship in your physical model, Toad Data Modeler automatically adds a new FK attribute to the child entity. Such behavior is logical as a foreign key cannot exist without relationship. Such FK attributes cannot be deleted without deleting the appropriate relationship.

See "Understanding of Foreign Keys" (page 70) for more information.

## How to Create Self Relationship

1. Click the Non-identifying relationship icon on the toolbar .
2. Double-click the selected entity.

If a copy of identifying attribute is created after you add the self-relationship- See "Parent Attributes (Rolenames)" (page 73) for more information.

**Tip:** Right-click the entity and select **Align Self Relationship**.

## Tips and Tricks for Relationships

### *To create several relationships at one jump*

1. Press SHIFT and click the **Relationship** or **Non-identifying** icon. A blue frame will appear in the icon.
2. Create as many relationships in you model as you need.
3. Right-click the work area (or click the icon on the toolbar again) to turn this function off.



*To select a mode in which you want to create relationships - Straight Mode or Right Angled mode*

- » Select **Settings** | **Options** | **Physical/Logical Model** | select or clear the **Right Angled Lines** checkbox.

See "Arrange Relationship Lines" (page 229) for more information.

*To add handle points while creating a relationship*

1. Click the parent entity and then click the work area as many times as many handle points you want to create.
2. Finally click the target entity.

*To make only horizontal and vertical relationship lines*

1. Select a relationship type and click the parent entity.
2. Press SHIFT and hold the key down. - Now only vertical and horizontal lines are offered to you wherever you move your mouse cursor. (Click the work area to create some handle points.)
3. Finally, click the target entity.

*To move the selected entity with its relationship line, including its handle points*

- » Select the entity and the appropriate relationship (use SHIFT) and drag and drop the entity.

*To hide relationship names*

1. Right-click the Workspace and select **Workspace Format**.
2. In the **Workspace Format** dialog | **General** tab, clear the **Display Line Names** checkbox.

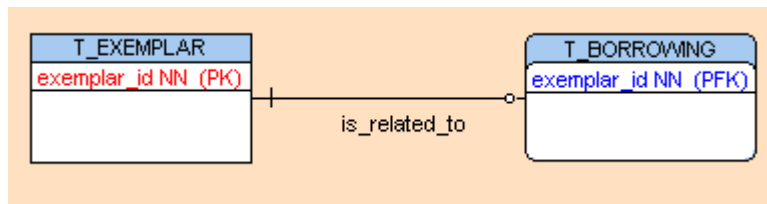
## Relationship Types

Toad Data Modeler supports the following relationship types in physical model:

- Identifying
- Non-identifying
- Self-relationship for non-identifying relationship
- M:N relationship

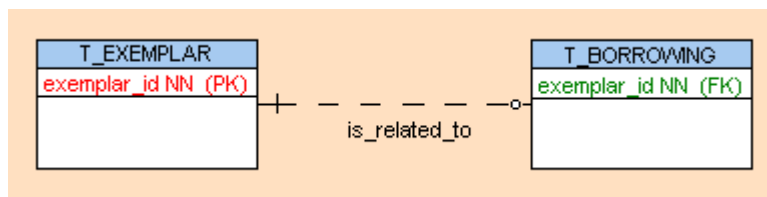
### Identifying Relationship

Primary key migrates from parent entity to child entity and becomes a part of the primary key there. It is used when the primary key of the child entity is unable to provide definite identification. An entity, connected with a parent entity through an identifying relationship, is called "dependent" entity and is shown in a model with rounded corners.



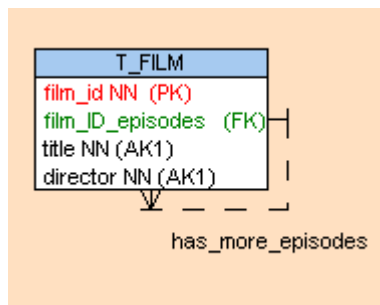
### Non-Identifying Relationship

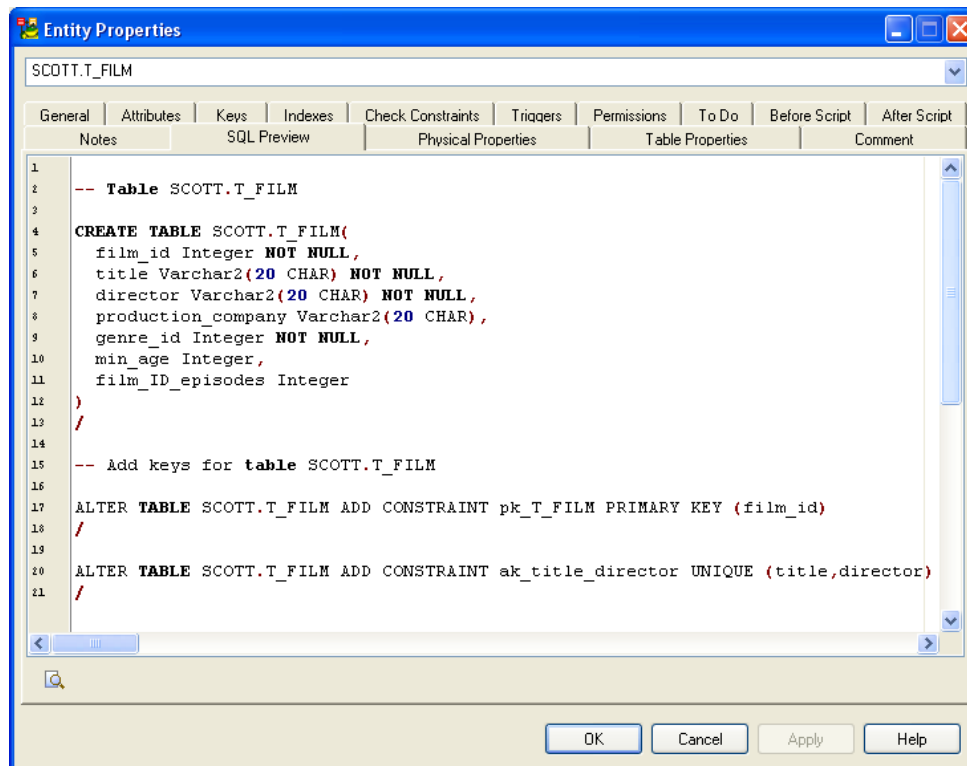
Primary key migrates from parent entity to child entity and does not become a part of the primary key. Non-identifying relationships are represented by dashed lines. In the dependent table, the attribute is referred to as a foreign key.



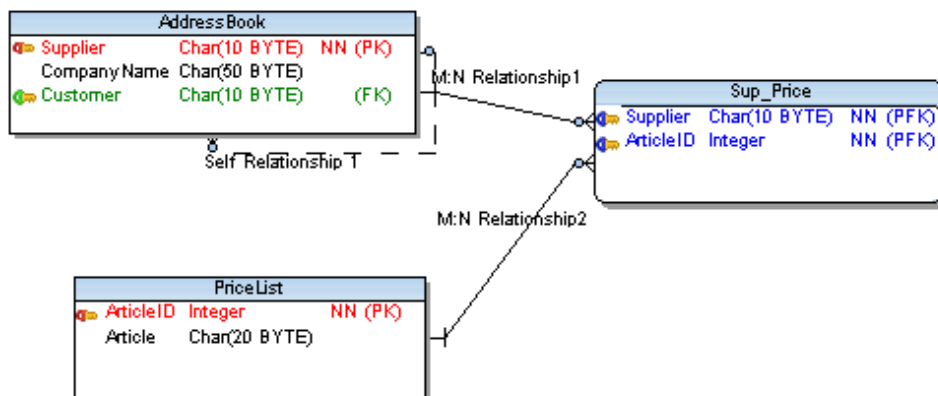
### Self-Relationship for Non-identifying Relationship

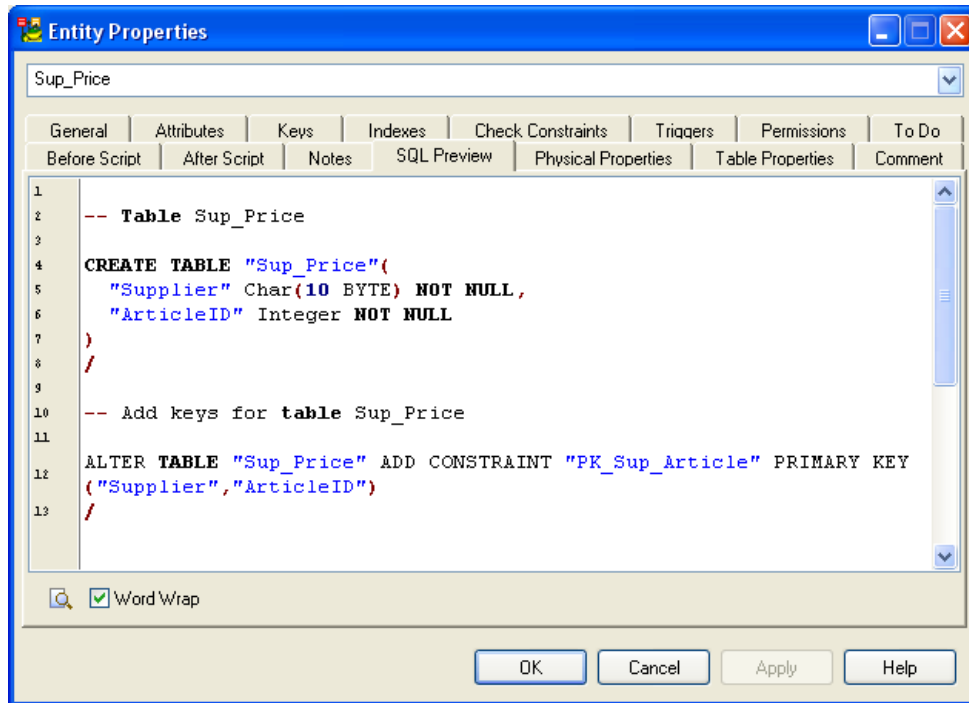
E.g. A person from table Person may be married to another person from the same table. In the example below, we track movies with more episodes.





## M:N Relationship





### *To change type of relationship*

- » Edit the relationship and on the **General** tab select or clear the checkbox **Identifying**.

## Related Topics

Define Cardinality (page 64)

## Define Cardinality

### Scenario

You need to set up cardinality 1:5 for the relationship *Makes*.

1. Double-click the *Makes* relationship on the Workspace.
2. Click the **General** tab | **Cardinality** area.
3. Type 5 to the **Cardinality** box.

**Relationship Properties**

makes

General | Referential Integrity | Foreign Keys | To Do | Notes | SQL Preview | Index to Foreign Key

Caption: makes

Name: makes

Inverse Caption: Is Made

Inverse Name: Is\_Made

Parent Entity Name: SCOTT.T\_CUSTOMER

Child Entity Name: SCOTT.T\_BORROWING

☐ Identifying

**Cardinality**

☒ Mandatory Parent

☒ Mandatory Child

Cardinality: 5

1..1 to 1..5

**Constraint State**

☐ Defferable

☐ Deferred

☐ Disable

☐ No Validate

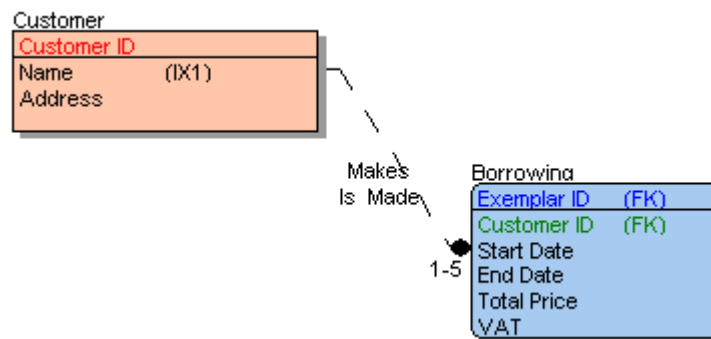
☐ Rely

☒ Create Index to Foreign Key

☒ Generate

OK Cancel Apply Help

- To display the cardinality in your ER diagram, select **Notation | IDEF1X**.



### Synchronization of NotNull and Mandatory Parent.

Select **Settings | Options | Physical Model | General tab | Synchronize Not Null with Mandatory Parent**.

By default, this checkbox is selected. You can change the setting during your modeling, the new setting will be used for new items you will create in your model straight away.

Synchronization enabled: If cardinality is set Mandatory Parent, the FK attribute will be automatically Not Null. If you clear the Not Null checkbox of the FK attribute, Mandatory Parent changes to optional.

Synchronization disabled: If you clear the Not Null checkbox of the FK attribute, the cardinality of Mandatory Parent will remain unchanged.

## Related Topics

Display Partiality (page 66)

## Display Partiality

*To define partiality for Parent and Child tables*

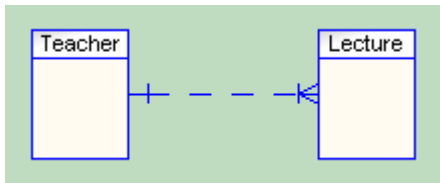
1. Edit the selected relationship on the Workspace.
2. Click the **General** tab in the **Relationship Properties** dialog and select or clear the **Mandatory Parent** and **Mandatory Child** checkboxes.

See how the relationship will look like for different partiality (to display cardinality in your ER diagram, select **Notation** | **IDEF1X**):

Parent: **Mandatory**

Child: **Mandatory**

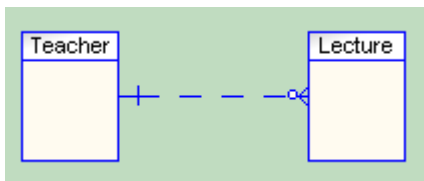
A teacher **MUST** have a lecture (the record related to lecture is mandatory), a lecture **MUST** be attached to a teacher (the record related to teacher is also mandatory. Teacher is Mandatory.)



Parent: **Mandatory**

Child: **Optional**

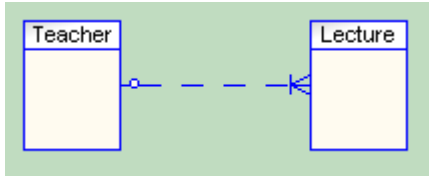
A teacher **MAY** have a lecture (the record related to lecture is Optional), a lecture **MUST** be attached to a teacher (Teacher is Mandatory.)



Parent: **Optional**

Child: **Mandatory**

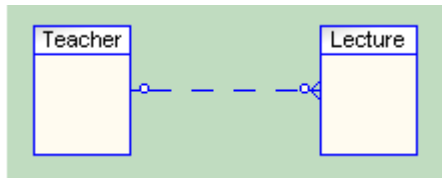
A teacher **MUST** have a lecture (lecture is Mandatory), a lecture **MAY** be attached to a teacher (teacher is Optional).



Parent: **Optional**

Child: **Optional**

A teacher **MAY** have a lecture (lecture is Optional), a lecture **MAY** be attached to a teacher (teacher is Optional).



## Related Topics

Set up Referential Integrity Rules (page 67)

## Set up Referential Integrity Rules

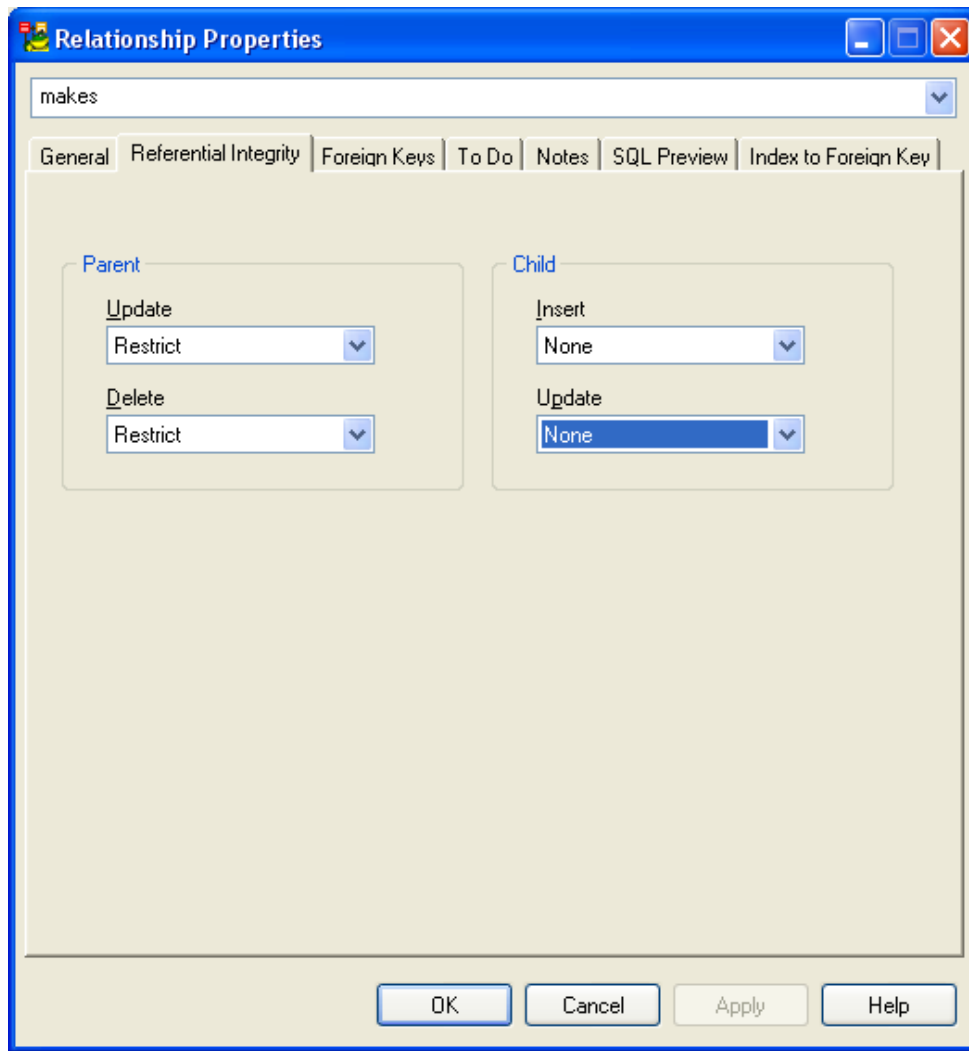
*To set up referential integrity rules for Parent and Child entities*

- » Open the **Relationship Properties** dialog | **Referential Integrity** tab.

Here, you can change the None rule to:

- Restrict
- Cascade
- Set Null
- Set Default

The default rule can be re-configured.



**Note:** Some databases do not support Referential Integrity (RI) rules.

### Related Topics

Connect Parent and Child Entities (page 68)

## Connect Parent and Child Entities

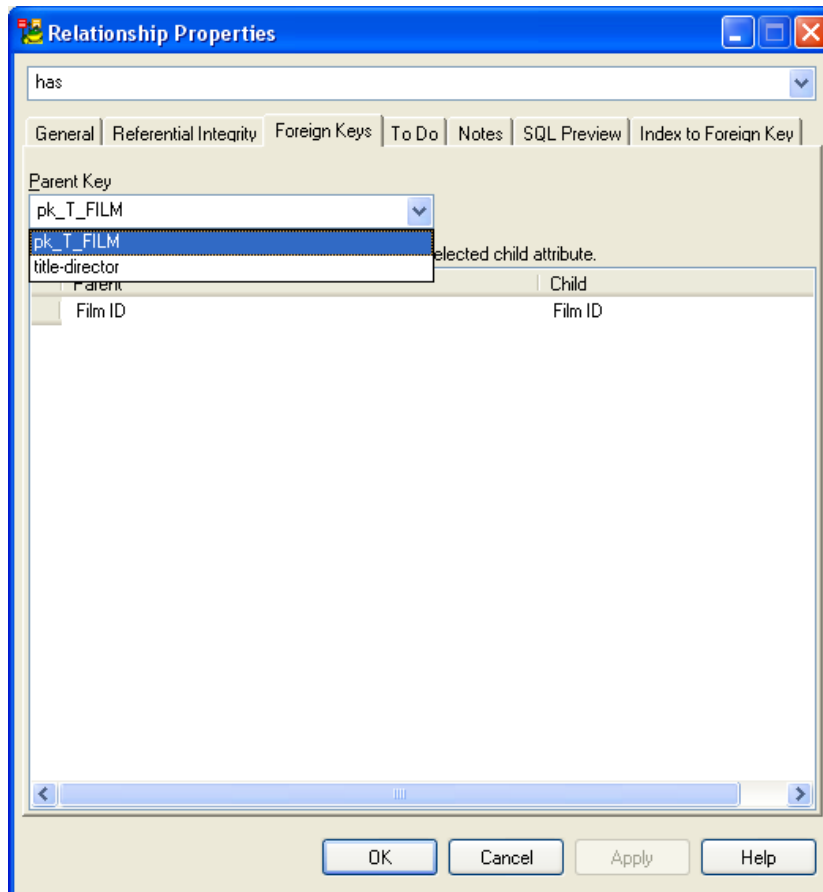
Toad Data Modeler allows you to connect Parent and Child entities through the following unique values:

- Key (Primary or Alternate Key) of parent entity
- Unique Index of parent entity



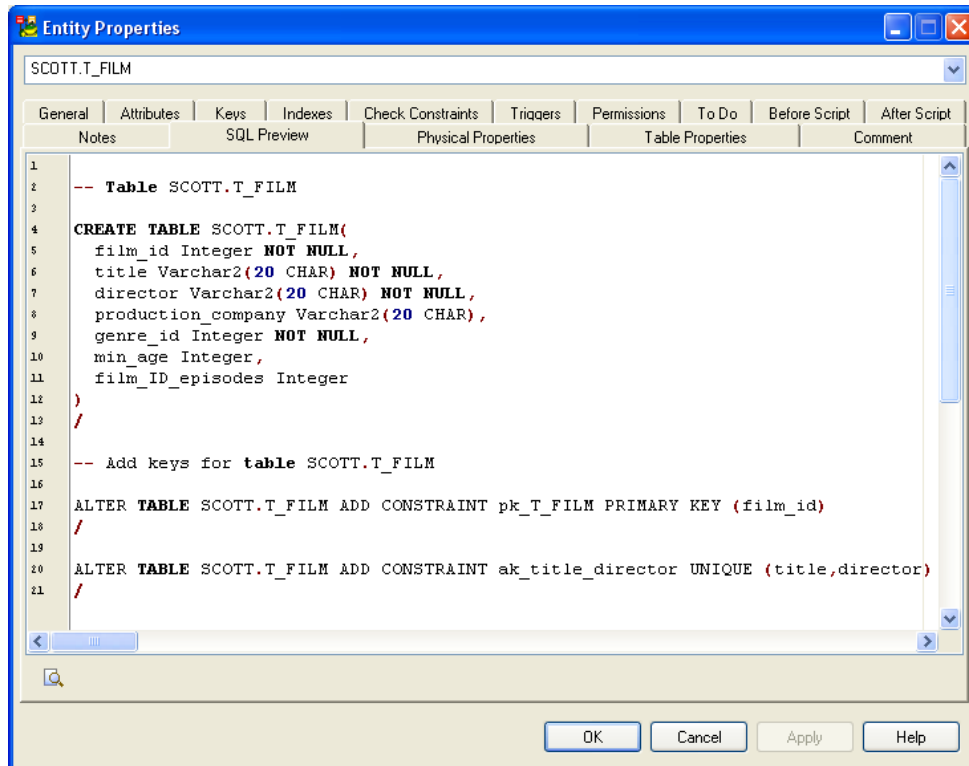
*To change the linking method*

1. Open the **Relationship Properties** dialog | **Foreign Keys** tab.
2. From the **Parent Key** box, select the method. If neither Unique item, nor Alternate Key exists, only the Primary key is available.



**Note:** Some databases do not support Alternate Keys.

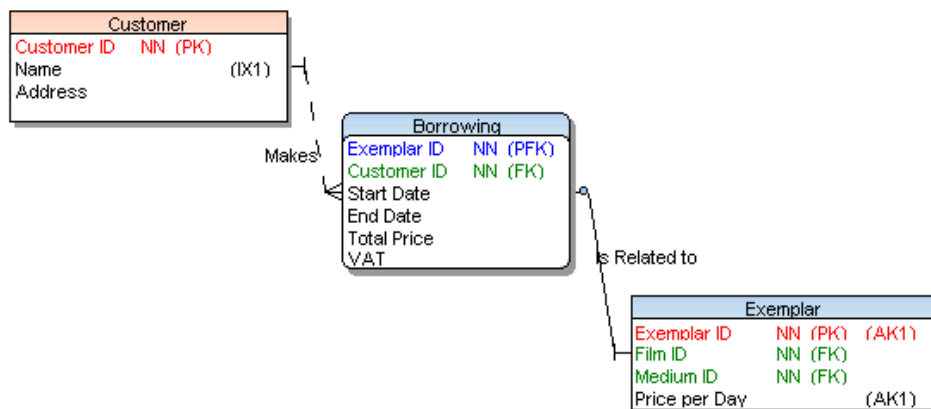
**Example** of generated SQL (Oracle 10g):



## Understanding of Foreign Keys

There are two types of foreign keys in Toad Data Modeler:

- Primary Foreign keys (PFK, graphically marked by blue key).
- Foreign keys (FK, graphically marked by green key).



## PFK

As soon as you create an identifying relationship, Toad Data Modeler automatically adds a copy of the primary key of the Parent table to the Child table - **Primary Foreign Key** (PFK). (The foreign key in Child table is a part of the primary key.)

## FK

As soon as you create a non-identifying relationship, Toad Data Modeler automatically adds a copy of the primary key of the Parent table to the Child table - **Foreign key** (FK). (This foreign key in Child table is not a part of the primary key.)

In other words, foreign keys (PFKs and FKs) are imported from parent entities to child entities automatically when you create relationship. Foreign keys cannot exist without relationships. Therefore, you are not allowed to delete FK or PFK from Child tables either. To delete them, you have to delete appropriate relationship.

In Toad Data Modeler, information on foreign keys can be found in:

- The **Attribute Properties** dialog | **Foreign Keys** tab where you can see details on particular FK.
- The **Relationship Properties** dialog | **Foreign Keys** tab where you set up linking method between parent and child entities.

## Foreign Keys Mapping

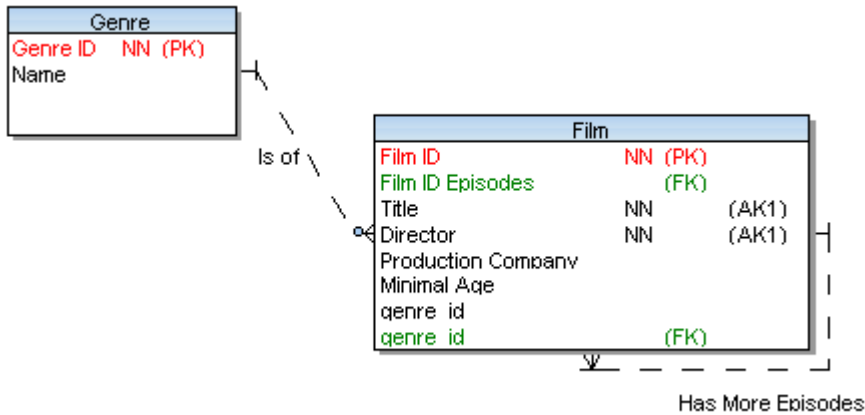
In Toad Data Modeler, you can control your foreign keys. Foreign keys mapping feature allows you to create Compound Keys, use existing keys etc.

### Example:

You have reverse engineered a model without relationships. In Toad Data Modeler, a new model has been created. However, information about foreign keys has been lost as in Toad Data Modeler stands that foreign keys cannot exist without relationships. Now you need to create relationships in Toad Data Modeler.

As soon as you create a new relationship, a new foreign key is added to the child entity automatically.

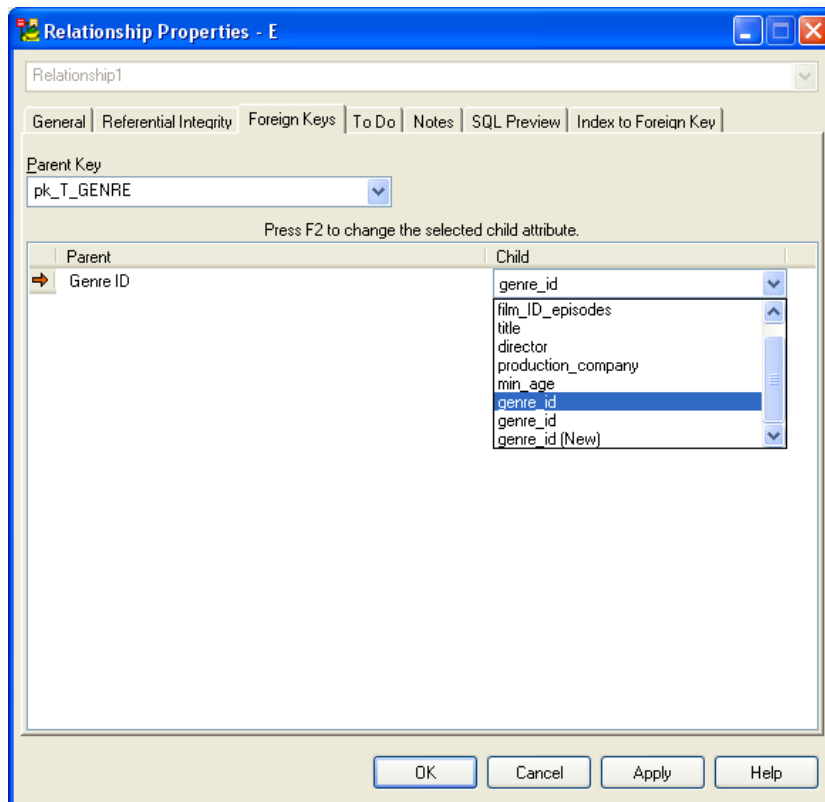
Now the child entity has two attributes of the same name (the original attribute and the newly added foreign key). - See the two *Genre ID* attributes in the *Film* entity in the screenshot below.



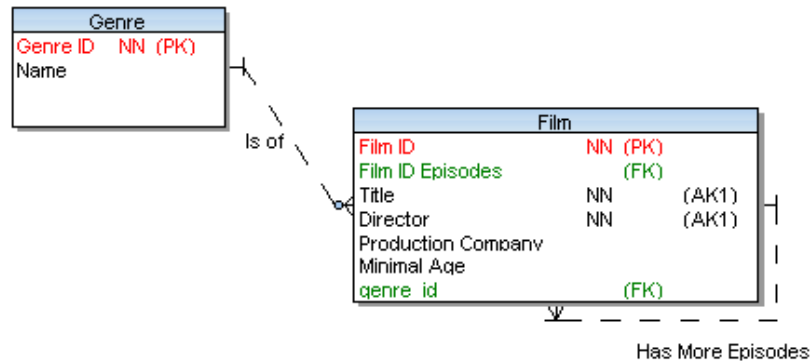
### Solution:

You don't have to erase the original attribute. Instead, you can remap the foreign key:

1. Open the **Relationship Properties** dialog | **Foreign Keys** tab.
2. Click the *Genre ID* item in the **Child** column and press F2 to activate the inplace editor.
3. From the combo-box, select the first *Genre ID* item and confirm **OK**.



**Result:** There is only one *Genre ID* FK in the *Film* entity now.



**Tip:** To restore the original foreign key later, edit the relationship again | **Foreign Keys** tab and select the item *FK attribute (New)*.

## Parent Attributes (Rolenames)

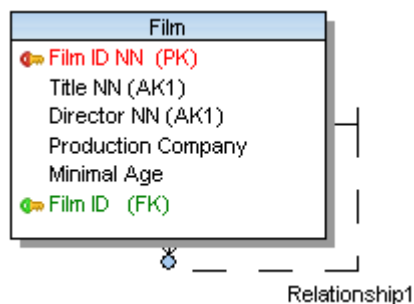
Use this feature when you:

- Create a self-relationship
- Want to change the Foreign Key attribute name

### Scenario

You have just added a self-relationship to the *Film* entity.

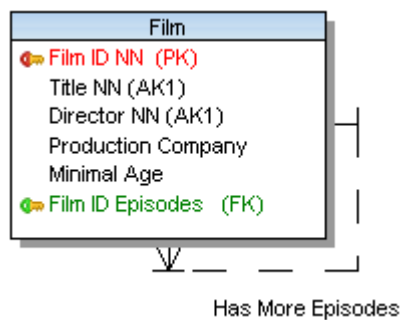
1. A copy of your identifying key attribute has been created in the entity - *Film ID*. Now you have two *Film ID* attributes in the *Film* entity.



2. Edit the newly created Film ID attribute, define a new **Name** and change the **Caption**. Item in the **Parent Attribute** box is not editable. Point your mouse cursor at it to display a pop up dialog with information on the entity and the relationship of the attribute.

The image shows the 'Attribute Properties' dialog box in Toad Data Modeler. The 'Entity' is 'SCOTT.T\_FILM' and the selected attribute is 'film\_ID\_episodes'. The 'General' tab is active, showing fields for 'Caption' (Film ID Episodes), 'Name' (film\_ID\_episodes), 'Parent Attribute' (film\_id), 'Data Type' (Integer), 'Domains', 'Default', 'Default Rule' (None), 'Used Sequence (trigger)' (None), and 'Encryption Specification'. The 'Not Null' and 'Unique' checkboxes are unchecked. The 'OK' button is highlighted.

3. Confirm **OK** and see the change on the Workspace.



## Advanced Database Design

### About Advanced Options

Toad Data Modeler supports several database systems. As there are differences among databases, there are also some specific options that stand only for the particular database.

For example:

- In MS SQL 2000 db, there are no user defined types but only Dictionary types. Dictionary types are called user-defined data types in MS SQL 2000 database.
- In PostgreSQL database, Users are not supported (only User Groups). To User Groups, roles should be written.
- and others.

Please read the Help file, "Databases" topic and select your database to see some screenshots and options specific for your database.

### Dictionary Items

#### Use Dictionary

The Dictionary in Toad Data Modeler includes the following items:

- User Data Types
- Dictionary Types
- Domains

#### User Data Types

They are data types defined by users. They can be generated in final DDL script.

User data types are not derived from data types.

If the database you're using supports user data types, you can freely define them in Toad Data Modeler.

**Example:** Phone List Type (Varray type) in a model for Oracle 10g.

#### Dictionary Types

Dictionary type is an alias of data type. Dictionary types can be generated in final DDL script. They are data types that are derived from other data types. Again, they must be supported by your database so as you can create dictionary types in Toad Data Modeler.

**Example:** Price dictionary type (Float data type) in model for SQL Server 2005 database.

## Domains

Domains represent just logical items. All items stored among Domains will be converted to particular data type while generating DDL script.

## Export/Import of the Dictionary

Toad Data Modeler allows you to use dictionary items also in other models. You can simply export all of them to the .TXI file, and then import them to any model at any time. You can save the .TXI file where you want, no default path is defined.

You cannot select what dictionary items you want to export. Always, all the items are exported at one jump.

### Scenario

You want to use the dictionary items of *Model A* in *Model B*.

1. Open *Model A* in the Application Window.
2. Select **Model | Export Dictionary**.
3. Save the .txi file.
4. Open *Model B* in the Application Window.
5. Select **Model | Import Dictionary**.
6. Select the .txi file and click **Open**.

## User Data Types

For more comfortable work with data types in Toad Data Modeler, you can define your own data types - provided that the database you use supports them.

Advantages of using user data types:

- You create a data type that meet your requirements.
- You can assign the data type to attributes comfortably from the Data Type box in the **Attribute Properties** dialog | **General** tab.
- If you need to edit the data type, you don't have to go through all the attributes of this data type. Just edit the data type in the Dictionary itself (**Model** menu | **User Data Types**), and the change will apply in all attributes of this data type automatically.

### Scenario

Create a user data type *Price Type* (Object type) for your *Videorental* Oracle 10g model. Assign the user data type to attribute *Total Price*.



*To create a user data type*

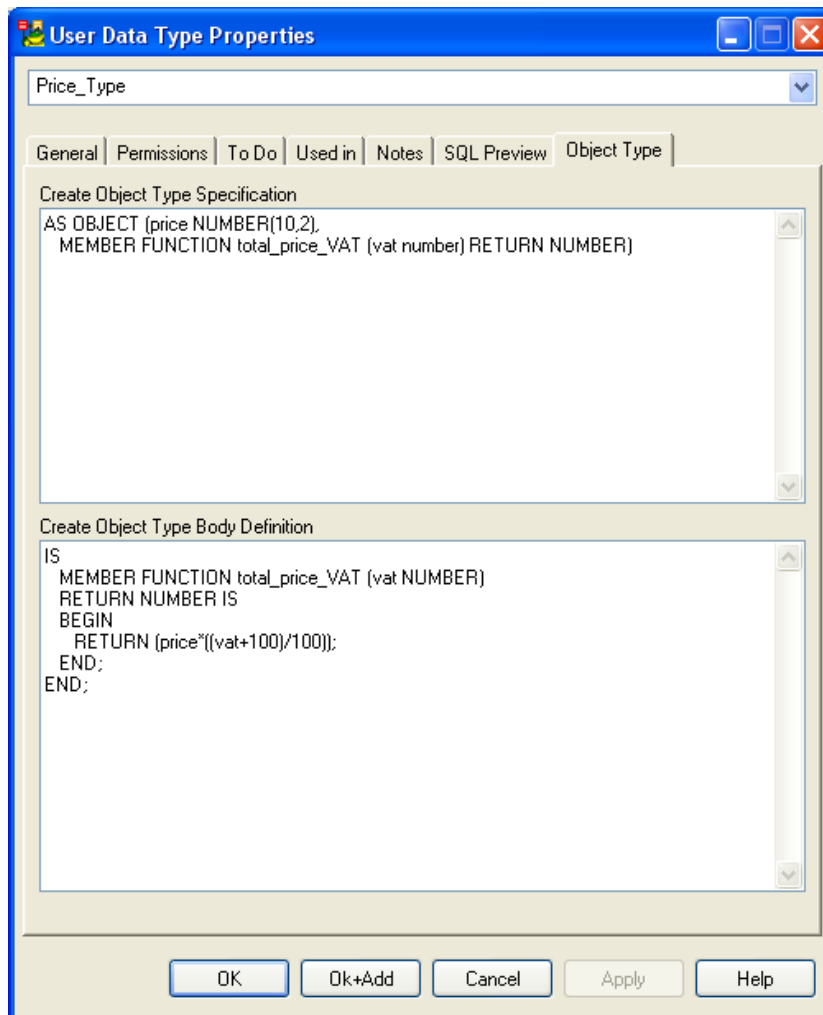
1. Select **Model | User Data Type**.
2. Click **Add** | confirm **Apply** | click **Edit**.

**Tip:** To create a user data type in Model Explorer, r Right-click the **User Data Types** | **Add User Data Type**. Double-click the newly created item to edit it.

3. Define basic properties on tab **General**: Caption, Name, Schema, select type.

The screenshot shows the 'User Data Type Properties' dialog box with the 'General' tab active. The 'Caption' and 'Name' fields both contain 'Price\_Type'. The 'User / Schema' dropdown menu is set to 'SCOTT'. The 'Type' dropdown menu is set to 'Object Type'. There are two unchecked checkboxes: 'Specification Is Wrapped' and 'Body Definition Is Wrapped'. At the bottom, the 'Generate' checkbox is checked, and 'Generate SQL Only' is unchecked. The dialog box has standard Windows window controls at the top right and a row of buttons (OK, Ok+Add, Cancel, Apply, Help) at the bottom.

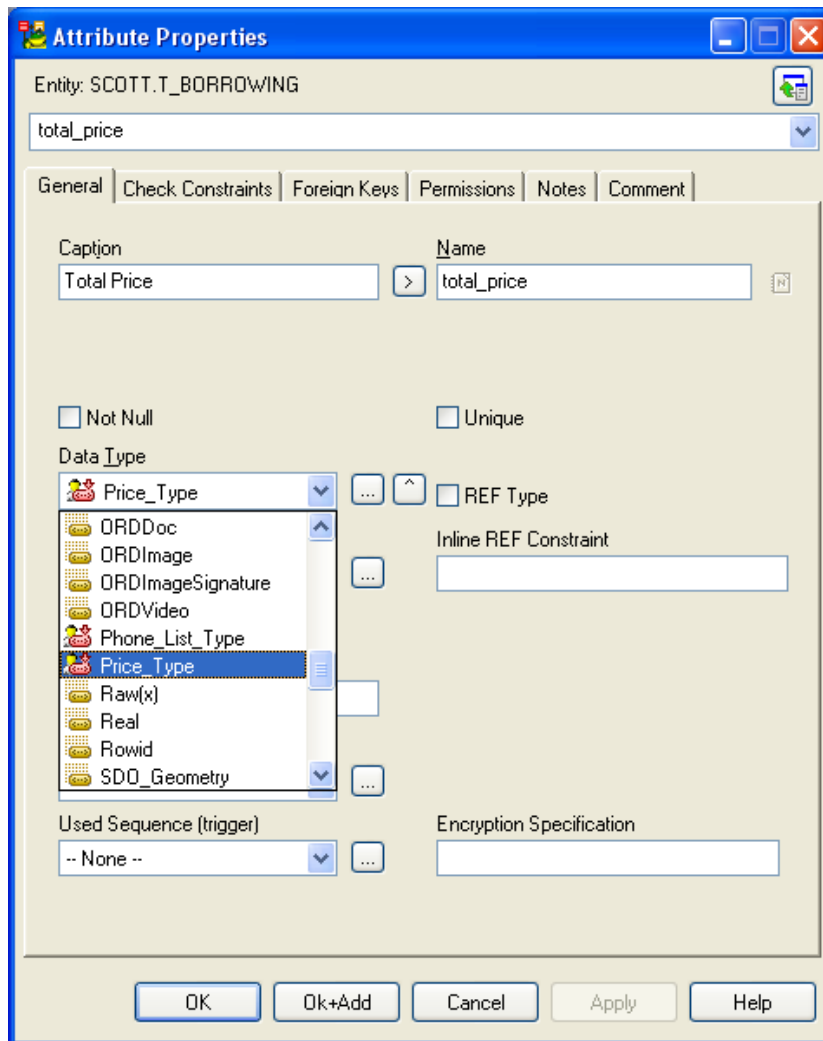
4. If you want, define properties on other tabs, and then click the **Object Type** tab.
5. Here, define the object type specification.



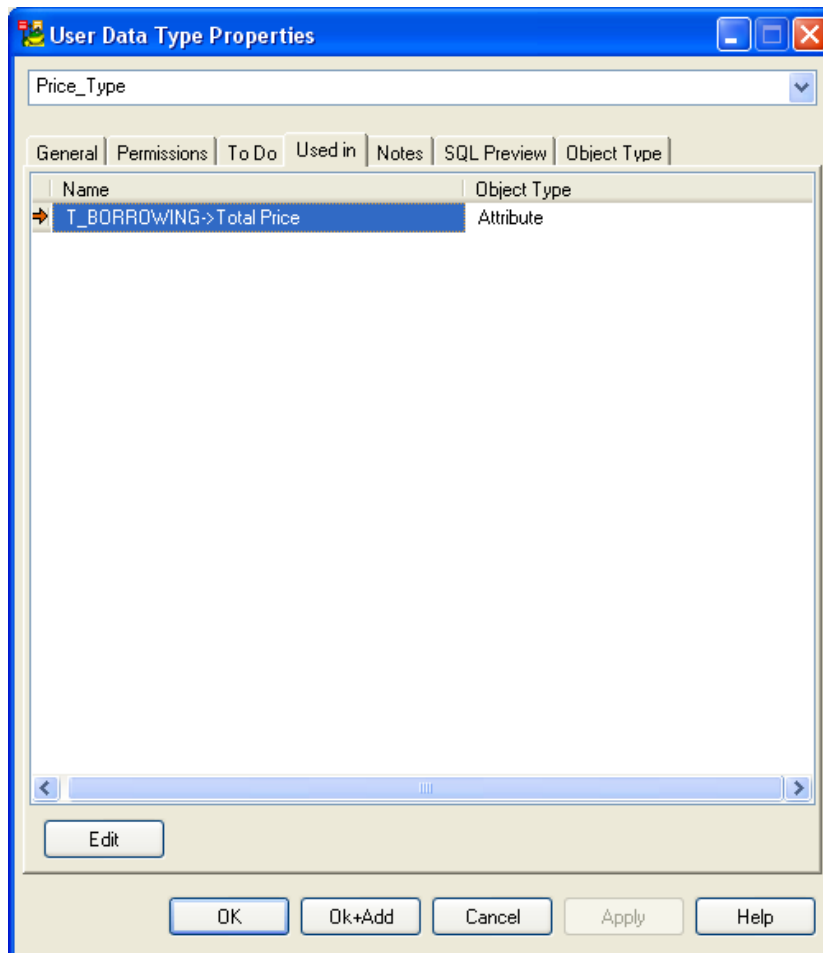
6. Confirm **OK**.

*To assign the new data type to attribute*

1. Open the **Attribute Properties** dialog, tab **General** of the attribute *Total Price*.
2. From the **Data Type** box, select the user data type *Price\_Type*.



**Result:** A list of all attributes that have been assigned appropriate user data type can be found in the **User Data Type Properties** dialog | **Used in** tab. In the **Name** column, there are names of the entity and attribute with this user data type. Double-click it to open the **Attribute Properties** dialog.

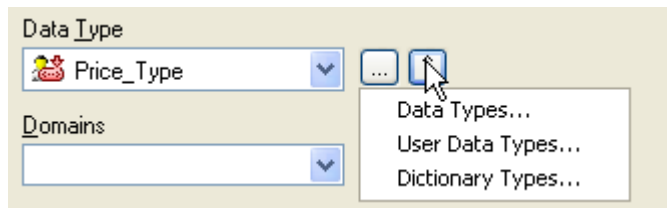


**Example:** Part of the generated script of the *Borrowing* entity (Oracle 10g model).  
See the *Total Price* attribute and its user data type:

```
-- Table SCOTT.T_BORROWING

CREATE TABLE SCOTT.T_BORROWING(
  exemplar_id Integer NOT NULL,
  customer_id Integer NOT NULL,
  start_date Date DEFAULT sysdate,
  end_date Date,
  total_price Price_Type,
  VAT Number(4,2) DEFAULT 19,
  CONSTRAINT check_end_after_start CHECK ((end_date>start_date) or (end_date is null))
);
```

**Tip:** Toad Data Modeler allows you to create and manage user data types and dictionary types from the **Attribute Properties** dialog directly. - See the small two icons next to the **Data Type** box.



## Dictionary Types

In Toad Data Modeler, you can also define dictionary types. Dictionary type is an alias of data type, and must be supported by the database you use.

**Note:** E.g. in Oracle db, only user data types are used. Nevertheless, in MS SQL db, user data types and also dictionary types are used.

The advantages, including how and where to manage dictionary types, are similar to user data types.

Advantages of creating dictionary types:

- You can create a dictionary type that will perfectly meet your requirements.
- You can assign the dictionary type to attributes comfortably from the Data Type box in the **Attribute Properties** dialog | **General** tab.
- If you need to edit the dictionary type, you don't have to go through all the attributes of this dictionary type. Just edit the type in the Dictionary itself (**Model** | **Dictionary Types**), and the change will apply in all attributes of this dictionary type automatically.

### Scenario

Create a dictionary type for SQL Server 2005 model and select the new dictionary type for an attribute.

#### *To create a dictionary type*

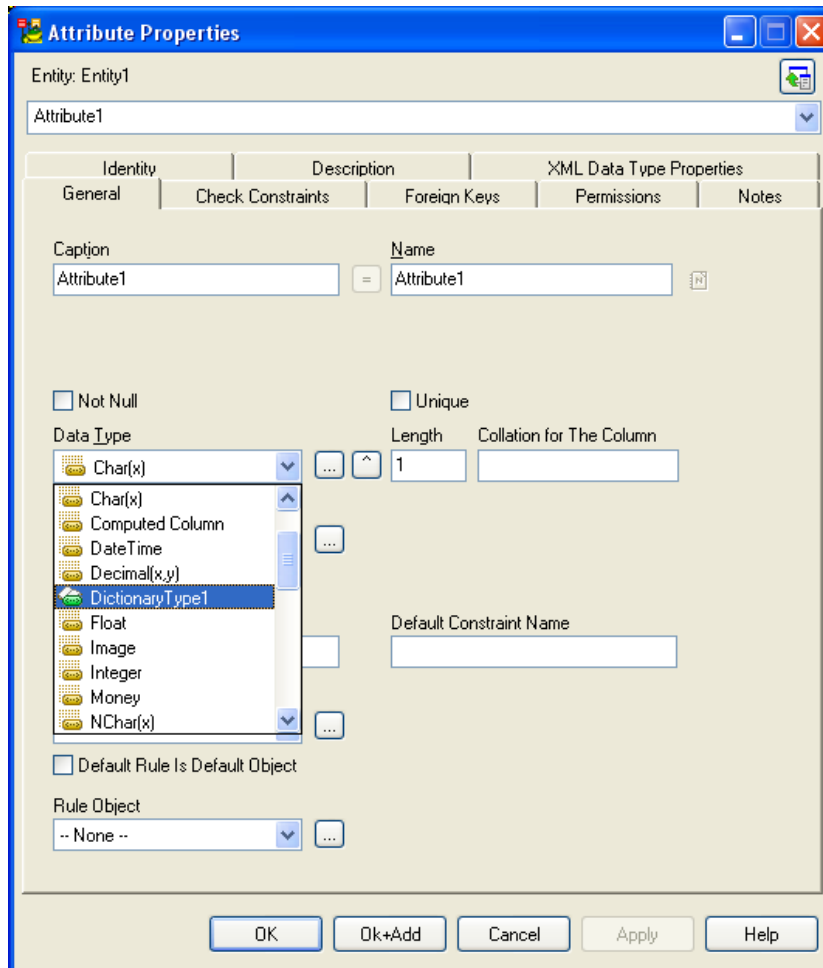
1. Select **Model** | **Dictionary Type**.
2. Click **Add** | confirm **Apply** | click **Edit**.

**Tip:** Create a dictionary type in Model Explorer: Right-click the **Dictionary Types** | **Add Dictionary Type**. Double-click the newly created item to edit it.

3. Define basic properties on tab **General**: Caption, Name, Schema, select data type etc.
4. After you define other properties, confirm **OK**.

*To assign the new dictionary type to attribute*

1. Open the **Attribute Properties** dialog, tab **General** of the attribute.
2. From the **Data Type** box, select the Dictionary Type.



## Domains

The last Dictionary item is a domain. Domains have only a logical meaning and are not generated. If a domain is used in attribute, only values of the domain are transferred to attribute during the DDL script generation process.

### Scenario

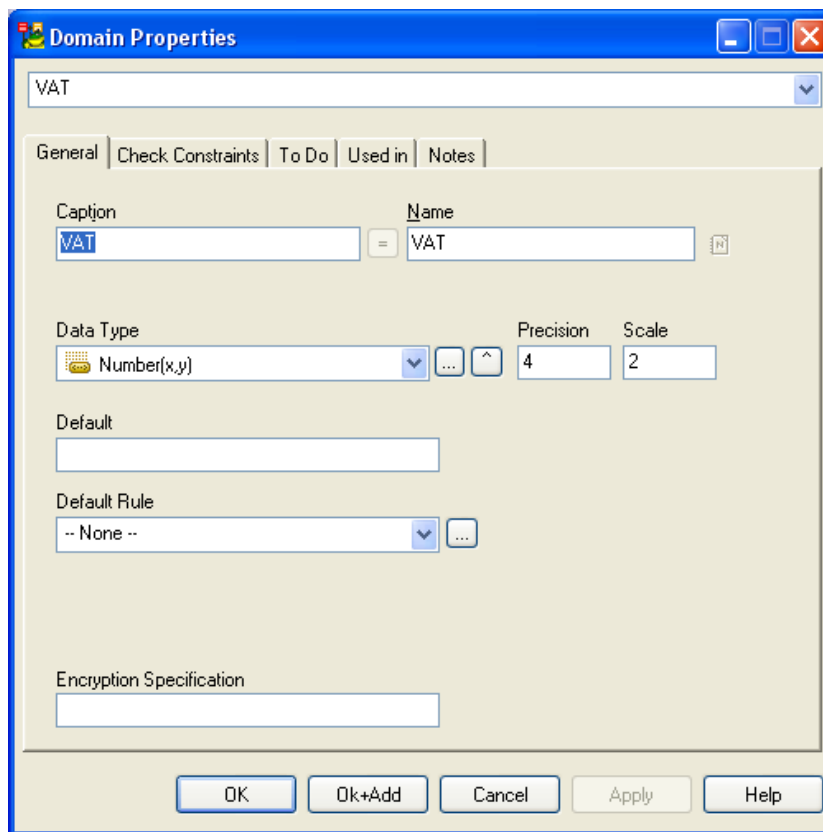
Create a domain *VAT* in your *Videorental* Oracle 10g model and select it for attribute *VAT*.

*To create a domain*

1. Select **Model | Domains**.
2. Click **Add** | confirm **Apply** | click **Edit**.

**Tip:** Create a domain in Model Explorer: Right-click the **Domains** | **Add Domain**.  
Double-click the newly created item to edit it.

3. Define basic properties on tab **General**: Caption, Name, Schema, select Data Type and specify Precision and Scale values.
4. After you define other properties, confirm **OK**.



The image shows the 'Domain Properties' dialog box for a domain named 'VAT'. The dialog has a title bar with standard window controls. Below the title bar is a dropdown menu showing 'VAT'. There are five tabs: 'General', 'Check Constraints', 'To Do', 'Used in', and 'Notes'. The 'General' tab is selected. It contains several fields: 'Caption' with the value 'VAT', 'Name' with the value 'VAT', 'Data Type' set to 'Number(x,y)', 'Precision' set to '4', and 'Scale' set to '2'. There are also fields for 'Default' and 'Default Rule' (set to '-- None --'). At the bottom, there is an 'Encryption Specification' field. The dialog has buttons for 'OK', 'Ok+Add', 'Cancel', 'Apply', and 'Help'.

*To assign the domain to attribute*

1. Open the **Attribute Properties** dialog, tab **General** of the attribute *VAT*.
2. From the **Domains** box, select the newly created domain.

The image shows the 'Attribute Properties' dialog box in Toad Data Modeler. The title bar says 'Attribute Properties'. Inside, the 'Entity' is 'SCOTT.T\_BORROWING' and the 'Attribute' is 'VAT'. There are six tabs: 'General', 'Check Constraints', 'Foreign Keys', 'Permissions', 'Notes', and 'Comment'. The 'General' tab is selected. It contains several fields and checkboxes: 'Caption' is 'VAT', 'Name' is 'VAT', 'Not Null' is unchecked, 'Unique' is unchecked, 'Data Type' is empty, 'Precision' is '4', 'Scale' is '2', 'Domains' is 'VAT', 'Default' is '19', 'Default Rule' is '-- None --', 'Used Sequence (trigger)' is '-- None --', 'Override Encryption Property' is unchecked, and 'Encryption Specification' is empty. At the bottom are buttons for 'OK', 'Ok+Add', 'Cancel', 'Apply', and 'Help'.

**Tip:** You can create and manage domains from the **Attribute Properties** dialog directly. - See the small icon next to the **Domains** box.

## Alternate Keys

As you already know, you can connect Parent and Child entities via:

- Key (Primary or Alternate Key) of parent entity
- Unique Index of parent entity

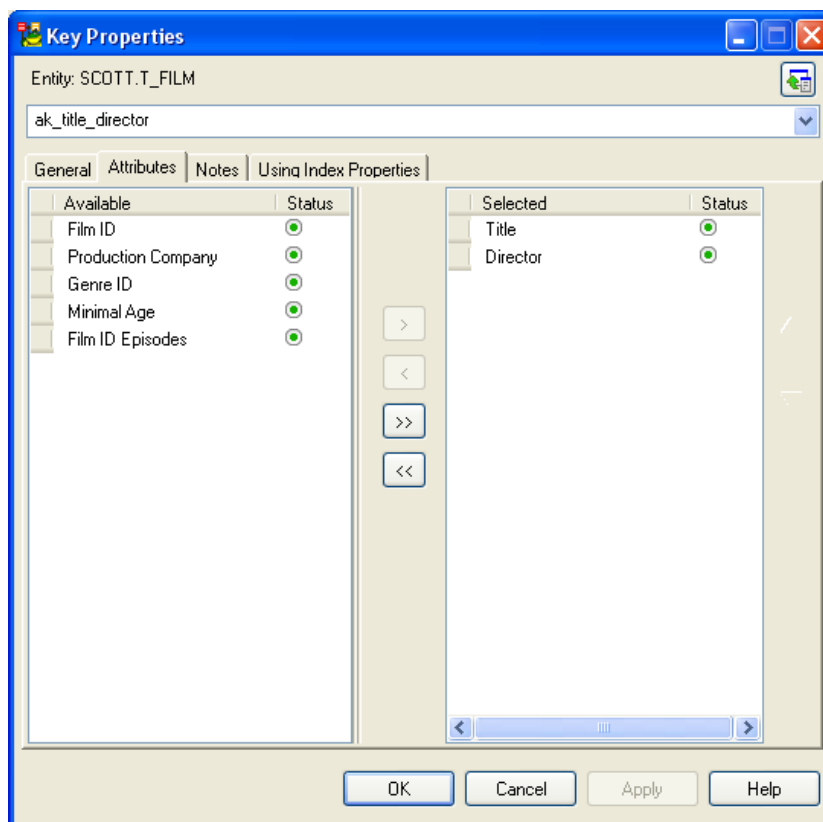
Alternate keys are used when you want to link two entities using two attributes. These two attributes make one unique item.

### Scenario



For entity *Film*, you need to create an alternate key with attributes *Title* and *Director*.

1. Edit the *Film* entity.
2. In the **Entity Properties** form click the **Keys** tab. Here, primary keys and alternate keys are stored, and you can manage them easily - using the Add, Edit and Delete buttons. So now click **Add** to create a new key and confirm **Apply**.
3. Edit the new key.
4. Define its properties on tab **General** and then click the **Attributes** tab.

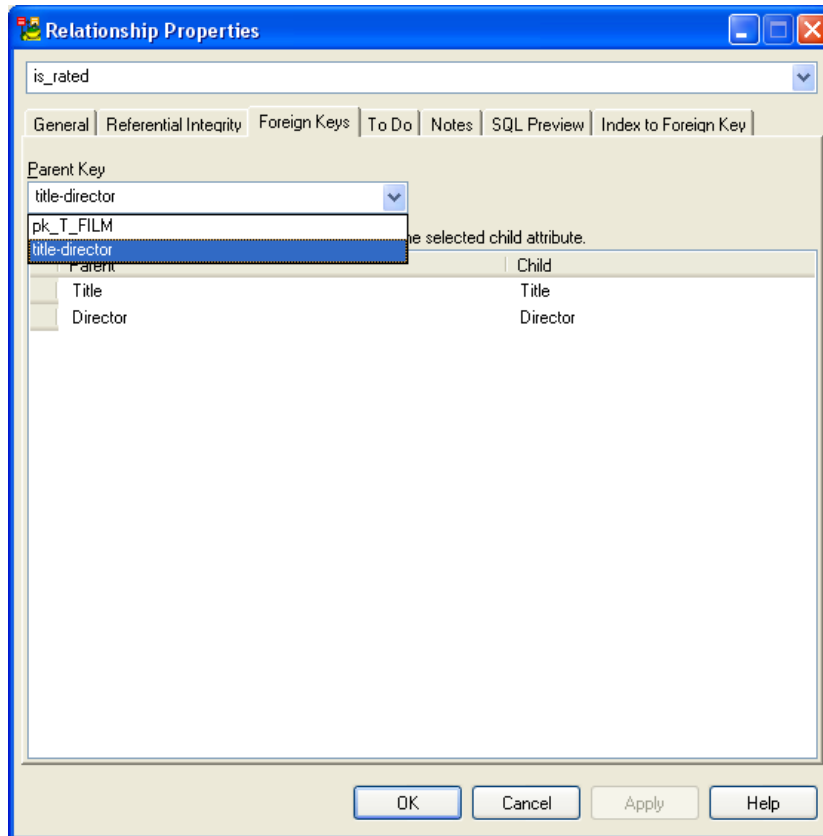


5. From the window **Available**, select attributes *Title* and *Director* and click the **Add** arrow button to move them to window **Selected**.
6. Define properties on other tabs at your convenience (e.g. Notes) and confirm **OK**.

**Scenario:** Now you want to select this alternate key for the linking method.

*To select the alternate key for relationship*

7. Double-click the relationship and click the **Foreign Keys** tab.
8. From the **Parent Key** box, select the alternate key *title-director*.



## Indexes

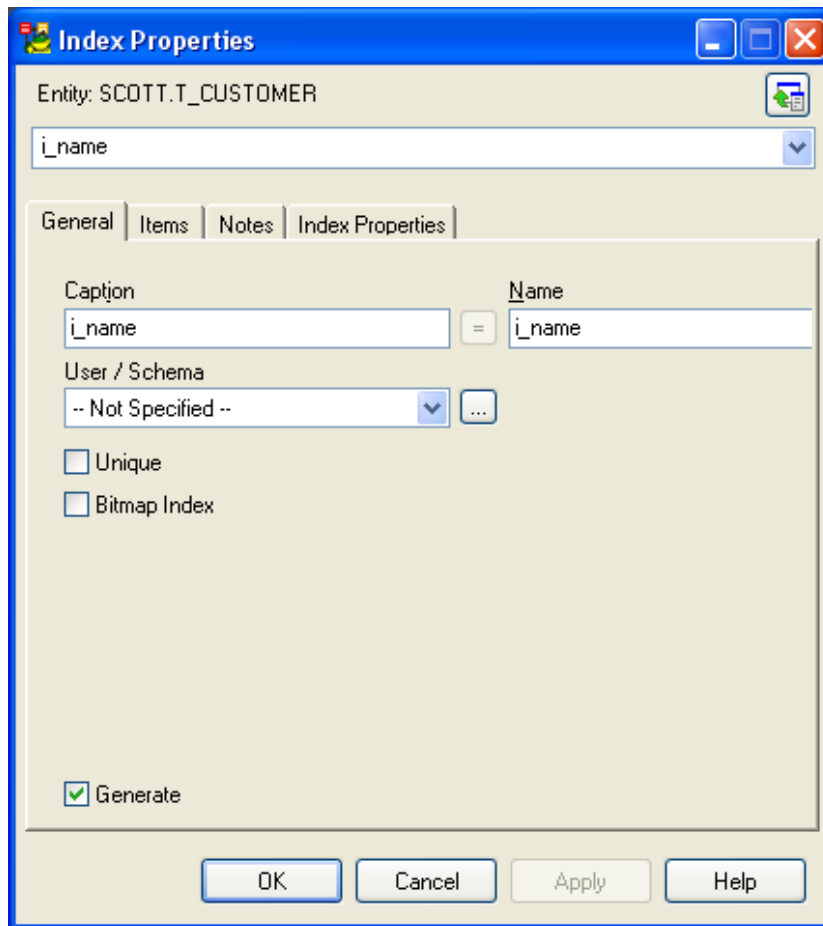
In Toad Data Modeler, you can create indexes on tab **Indexes** of the **Entity Properties** form. Also, here you can edit and delete them. - See the buttons **Add**, **Edit**, **Delete**.

Properties and options for indexes differ according to the database you use.

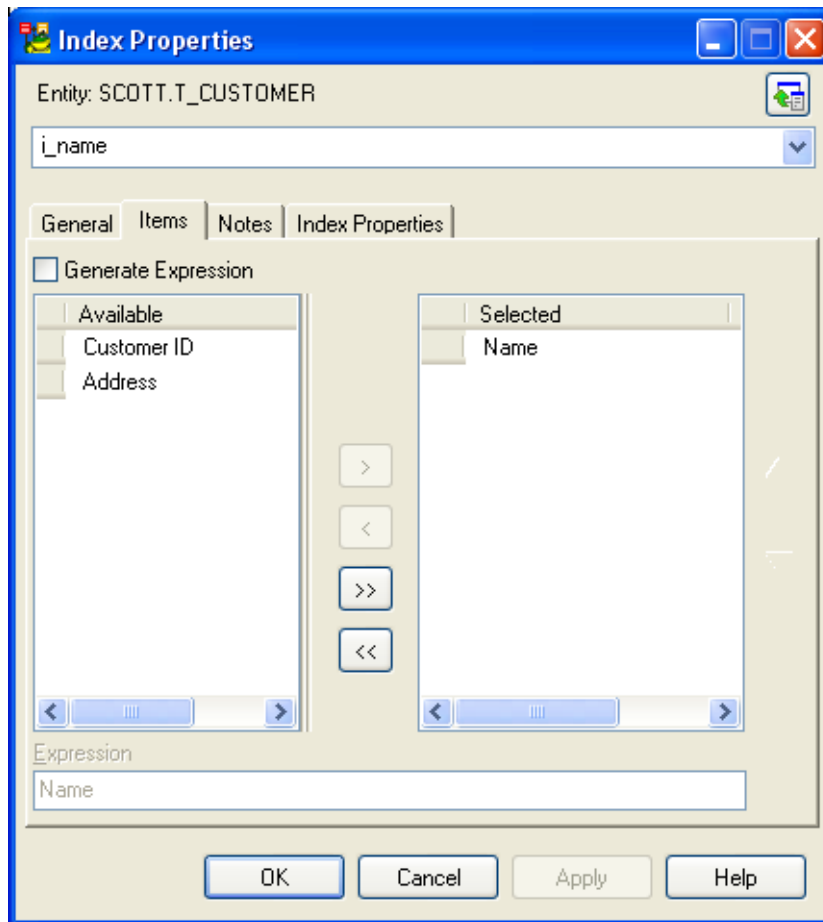
### Scenario

In your *Videorental* Oracle 10g model, create index *i\_name* for the *Customer* entity and display it on the Workspace.

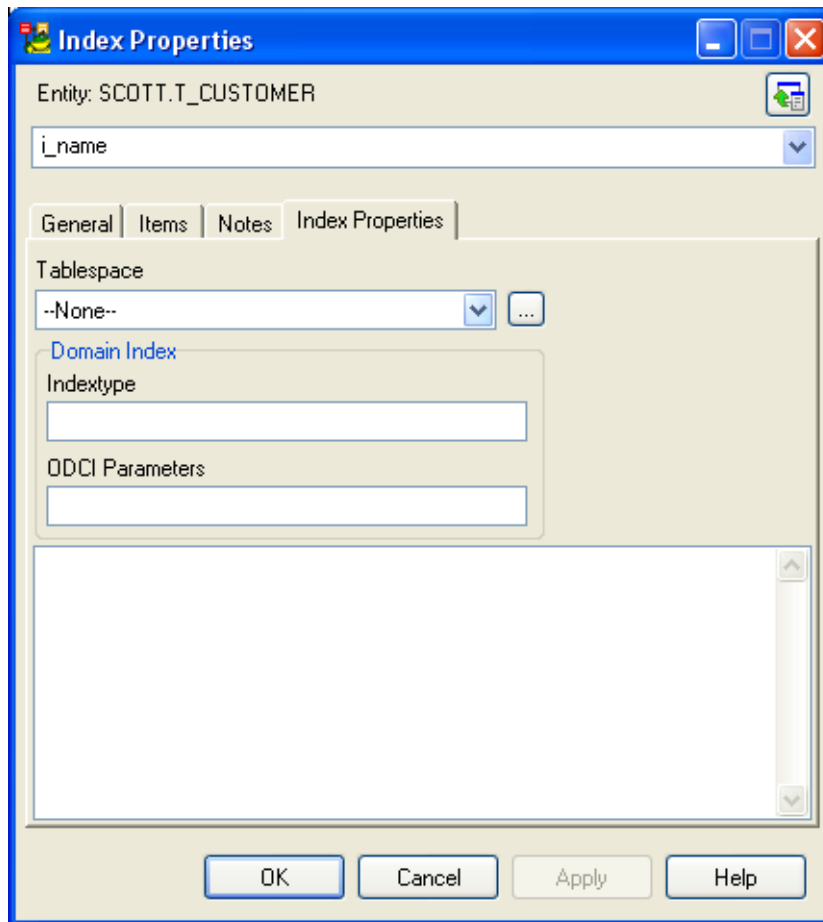
1. Edit the *Customer* entity.
2. In the **Entity Properties** form, **Index** tab click **Add**, confirm **Apply**, **Edit**.
3. Define properties of the new index *i\_name*.



4. On tab **Items**, select the attribute(s) to which you want to assign the index: simply select the attribute *Name* from the window **Available** and click the **Add** arrow button to move the selected item to the window **Selected**.



5. Set options on other tabs and confirm **OK**.



**Scenario:** To display indexes on the Workspace

- Right-click the WS, select **Workspace Format** | **Entity** tab and select the **Display Indexes** checkbox. See how the indexes are displayed:

T_CUSTOMER			
customer id	NN (PK)		
name	NN	(IX1,IX2)	
address	NN	(IX2)	
name (IX1)			
name address (IX2)			

**Note:** No matter if indexes are displayed or not, you can see which attribute belongs to which index (see e.g. attribute *Name* belonging to indexes *i\_name* and *i\_name\_address* - *name (IX1,IX2)*).

## Check Constraints and Triggers

Toad Data Modeler supports check constraints and triggers.

You can create them on appropriate tabs in the **Entity Properties** form. From here, you can also manage them easily (Add, Edit, Delete).

**Tip:** You can also add, edit and delete check constraints and triggers in Model Explorer. Just find the appropriate folders under the selected entity. E.g. *Borrowing* entity/**Triggers** /tri\_BORROWING.

**Example:** Check Constraint (Oracle 10g model)

**Check Constraint Properties**

Entity: SCOTT.T\_BORROWING

check\_end\_after\_start

General | SQL | Notes

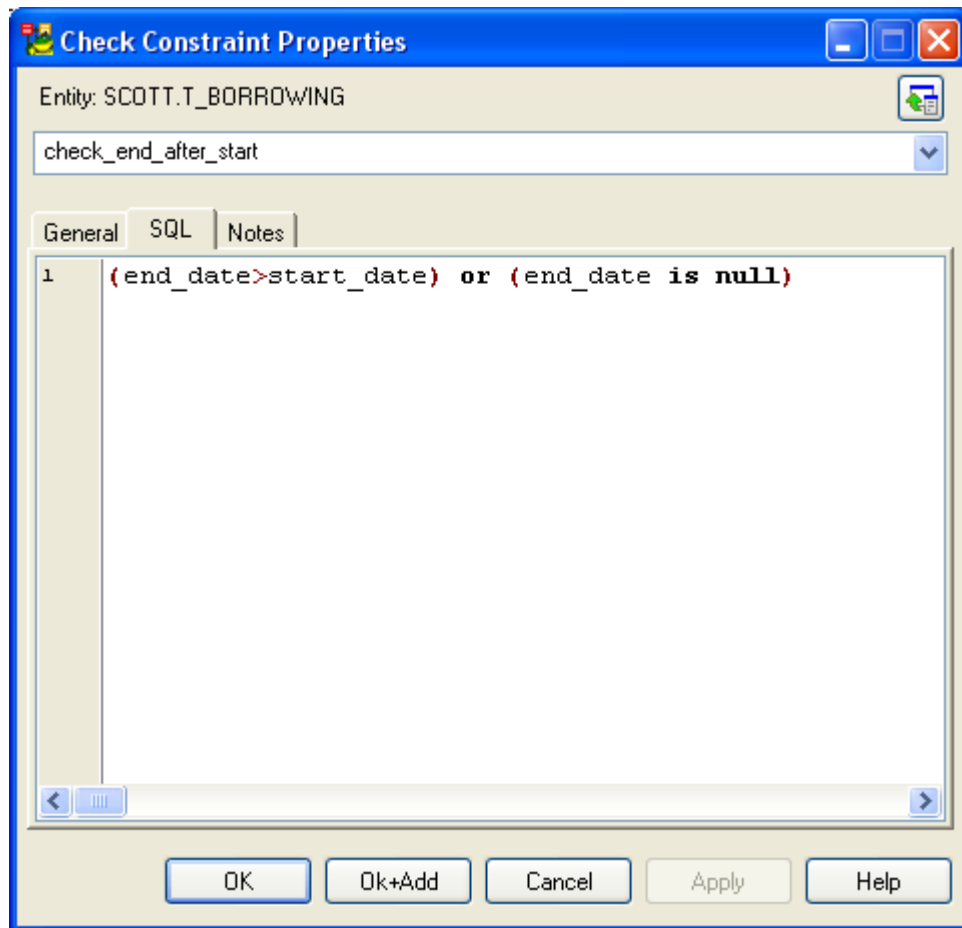
Caption: check\_end\_after\_start    Name: check\_end\_after\_start

Check Constraint Rule: --

☐ Deferrable  
☐ Deferred  
☐ Disable  
☐ No Validate  
☐ Rely

☒ Generate

OK    Ok+Add    Cancel    Apply    Help



**Example:** Trigger (Oracle 10g model)

**Trigger Property**

Entity: SCOTT.T\_BORROWING

SCOTT.tri\_BORROWING

General | SQL | Notes

Caption: tri\_BORROWING = Name: tri\_BORROWING

User / Schema: SCOTT

Trigger Fire: BEFORE

Trigger Events

- ☐ Delete
- ☒ Insert
- ☐ Update

Correlation Names (REFERENCING)

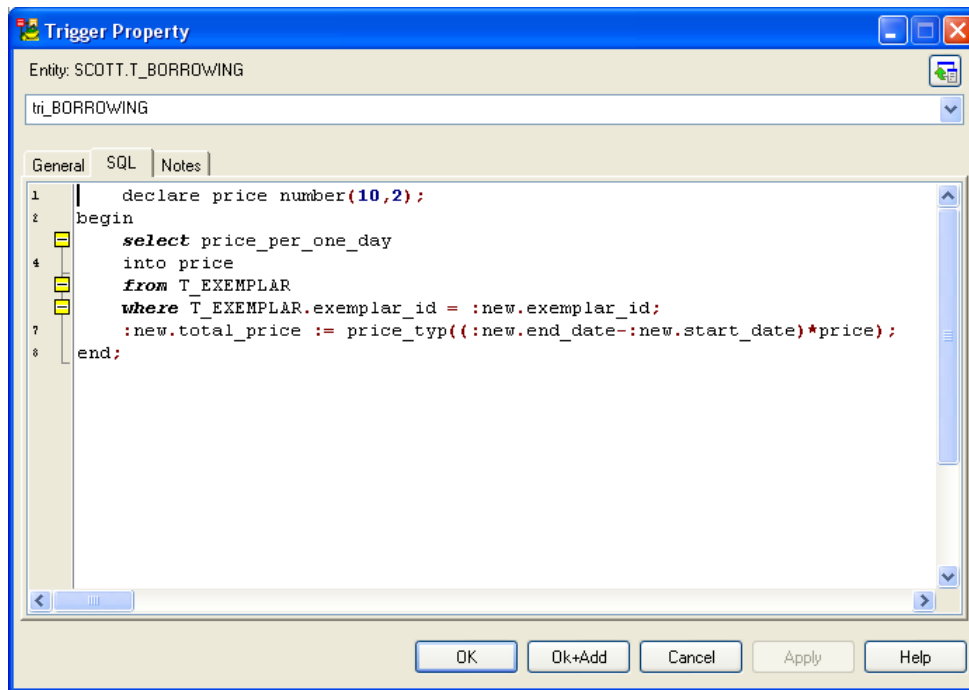
Old:  New:  When Condition:

☒ For Each Row

☒ Generate ☐ Generate SQL Only

OK Ok+Add Cancel Apply Help





## Views

Toad Data Modeler allows you to display views in your ER diagram graphically.


You can manage Views via:

- **Model menu | Views**
- **Model Explorer | Views folder**

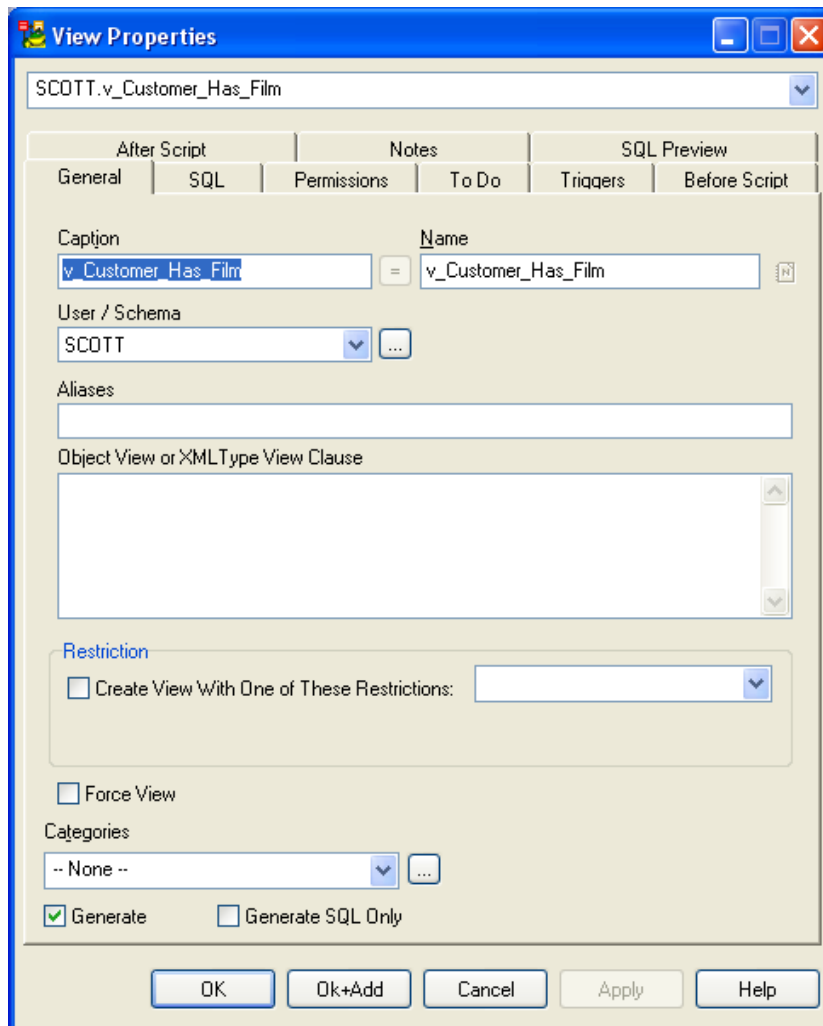
View properties and options are database dependent.

### Scenario

Create a view *Customer Has Film* in your *Videorental* Oracle 10g model.

1. Click  on the toolbar and then click the work area.
2. Double-click the view to edit it.
3. Define properties on tab **General** and other tabs.

**Tip:** Select **Generate SQL Only** if you want to generate only the SQL code written on tab **SQL** and to ignore the rest of items. (The same option works for Functions, Procedures and Triggers.)



The image shows the 'View Properties' dialog box in Toad Data Modeler. The title bar is blue with standard window controls. The main area has a light beige background. At the top, a dropdown menu shows 'SCOTT.v\_Customer\_Has\_Film'. Below this is a tabbed interface with three tabs: 'General' (selected), 'SQL', and 'Permissions'. The 'General' tab contains several fields: 'Caption' (v\_Customer\_Has\_Film), 'Name' (v\_Customer\_Has\_Film), 'User / Schema' (SCOTT), 'Aliases' (empty), 'Object View or XMLType View Clause' (empty), 'Restriction' (unchecked), 'Force View' (unchecked), 'Categories' (-- None --), and 'Generate' (checked). At the bottom are buttons for 'OK', 'Ok+Add', 'Cancel', 'Apply', and 'Help'.

**View Properties**

SCOTT.v\_Customer\_Has\_Film

After Script      Notes      SQL Preview

General    SQL    Permissions    To Do    Trippers    Before Script

Caption      Name

v\_Customer\_Has\_Film    =    v\_Customer\_Has\_Film

User / Schema

SCOTT

Aliases

Object View or XMLType View Clause

**Restriction**

☐ Create View With One of These Restrictions:

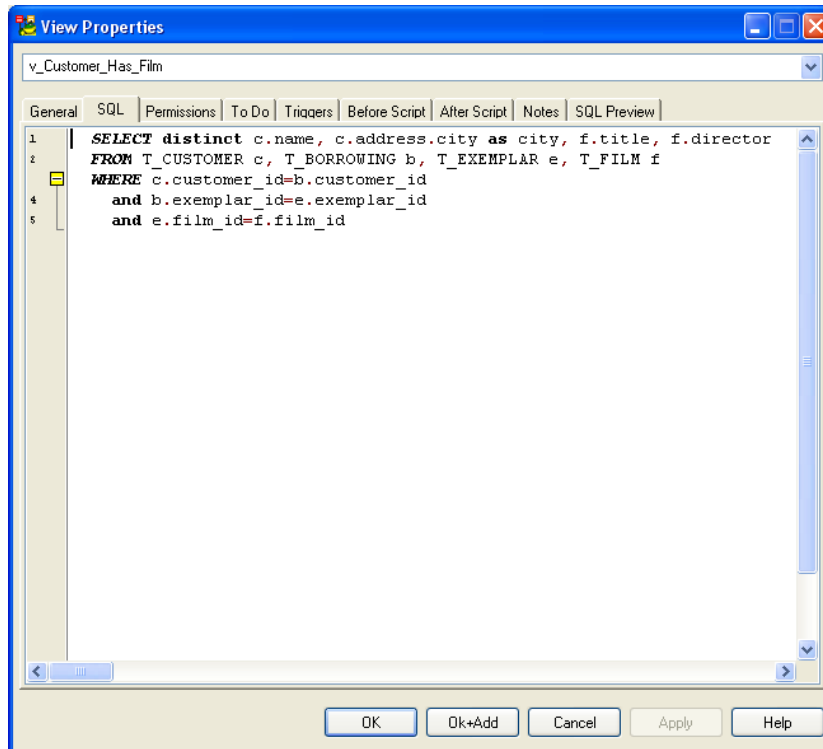
☐ Force View

**Categories**

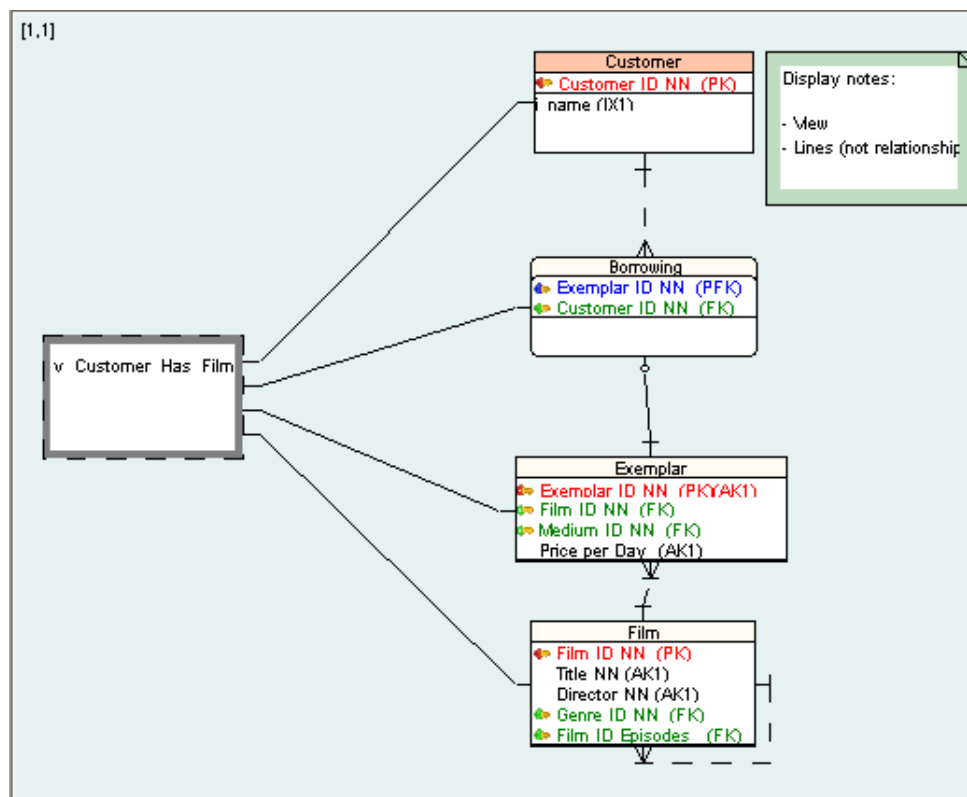
-- None --

☒ Generate    ☐ Generate SQL Only

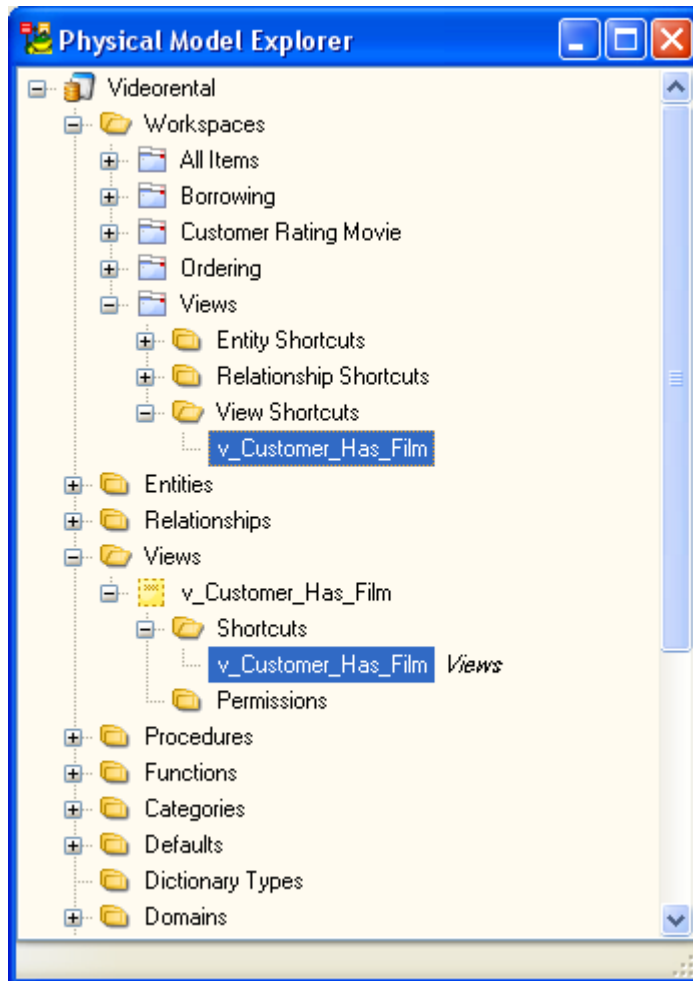
OK    Ok+Add    Cancel    Apply    Help



Graphical display of view on the Workspace (Oracle 10g model):



**Note:** As well as for entities and relationships, you can create shortcuts of views on the Workspaces of your model. List of view shortcuts can be found in **Model Explorer** | in particular Workspace folder and in the **Views** folder | **Shortcuts**.



## Materialized Views


Toad Data Modeler also allows you to display materialized views in your ER diagram graphically for the following databases: Oracle 11g, Oracle 10g and 9i, DB 2 v. 9.5, v. 9 (LUW), DB2 UDB v. 8 (LUW).

You can manage materialized views via:

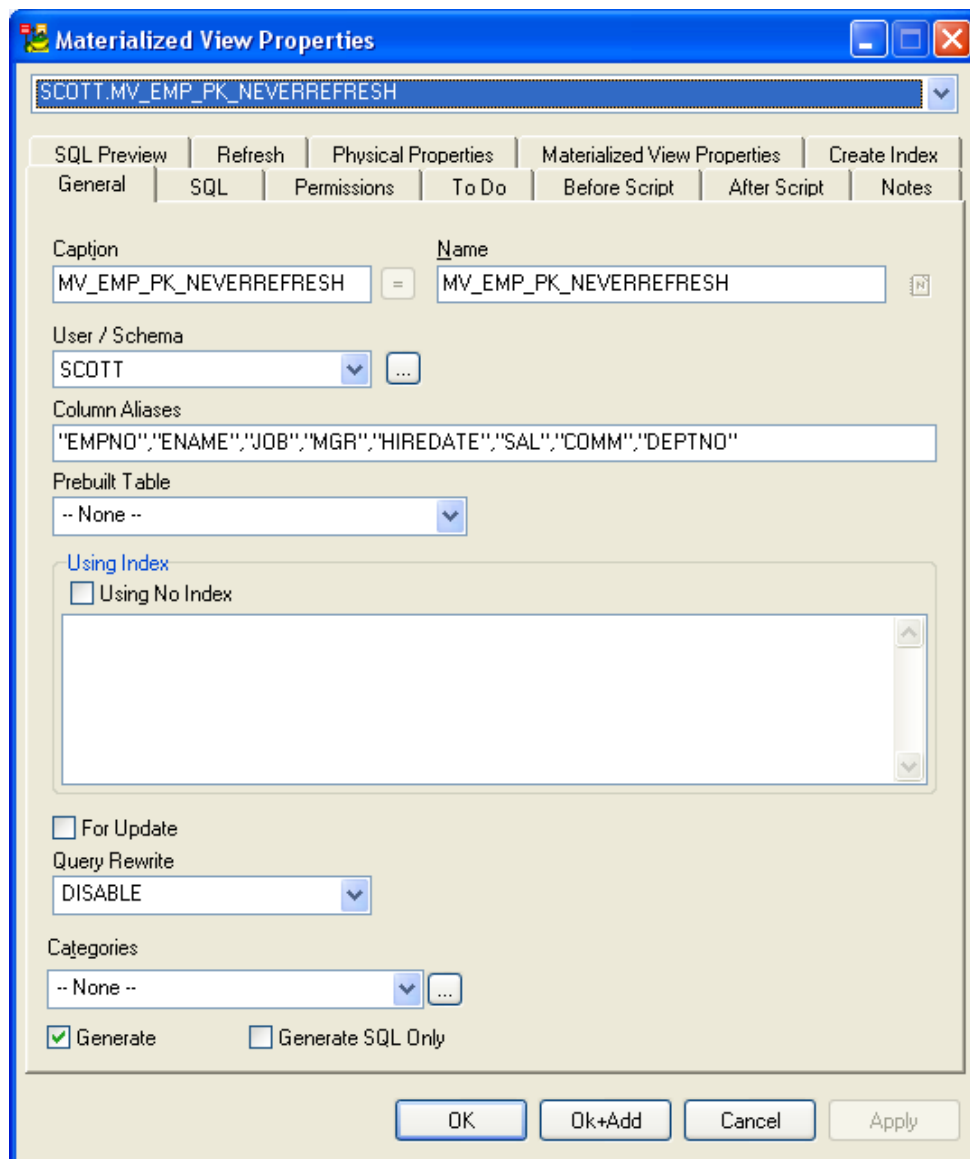
- **Model** menu | **Materialized Views**
- **Model Explorer** | **Materialized Views** folder

Materialized View properties and options are database dependent.

*To create a materialized view*

1. Click  on the toolbar and then click the work area.
2. Double-click the materialized view to define its properties.

**Example** of properties of materialized view (Oracle 10g model):



The image shows the 'Materialized View Properties' dialog box in Toad Data Modeler. The title bar is blue with standard window controls. The main area has a tabbed interface with tabs for 'SQL Preview', 'Refresh', 'Physical Properties', 'Materialized View Properties' (which is selected), and 'Create Index'. Below the tabs are sub-tabs: 'General', 'SQL', 'Permissions', 'To Do', 'Before Script', 'After Script', and 'Notes'. The 'General' sub-tab is active. It contains several fields: 'Caption' and 'Name' both set to 'MV\_EMP\_PK\_NEVERREFRESH'; 'User / Schema' set to 'SCOTT'; 'Column Aliases' containing '"EMPNO","ENAME","JOB","MGR","HIREDATE","SAL","COMM","DEPTNO"'; 'Prebuilt Table' set to '-- None --'; 'Using Index' section with 'Using No Index' selected; 'For Update' checkbox unchecked; 'Query Rewrite' set to 'DISABLE'; 'Categories' set to '-- None --'; and 'Generate' checkbox checked. At the bottom are buttons for 'OK', 'Ok+Add', 'Cancel', and 'Apply'.

**Materialized View Properties**

SCOTT.MV\_EMP\_PK\_NEVERREFRESH

SQL Preview | Refresh | Physical Properties | **Materialized View Properties** | Create Index

General | SQL | Permissions | To Do | Before Script | After Script | Notes

Caption: MV\_EMP\_PK\_NEVERREFRESH = Name: MV\_EMP\_PK\_NEVERREFRESH

User / Schema: SCOTT

Column Aliases: "EMPNO","ENAME","JOB","MGR","HIREDATE","SAL","COMM","DEPTNO"

Prebuilt Table: -- None --

Using Index:  
☐ Using No Index

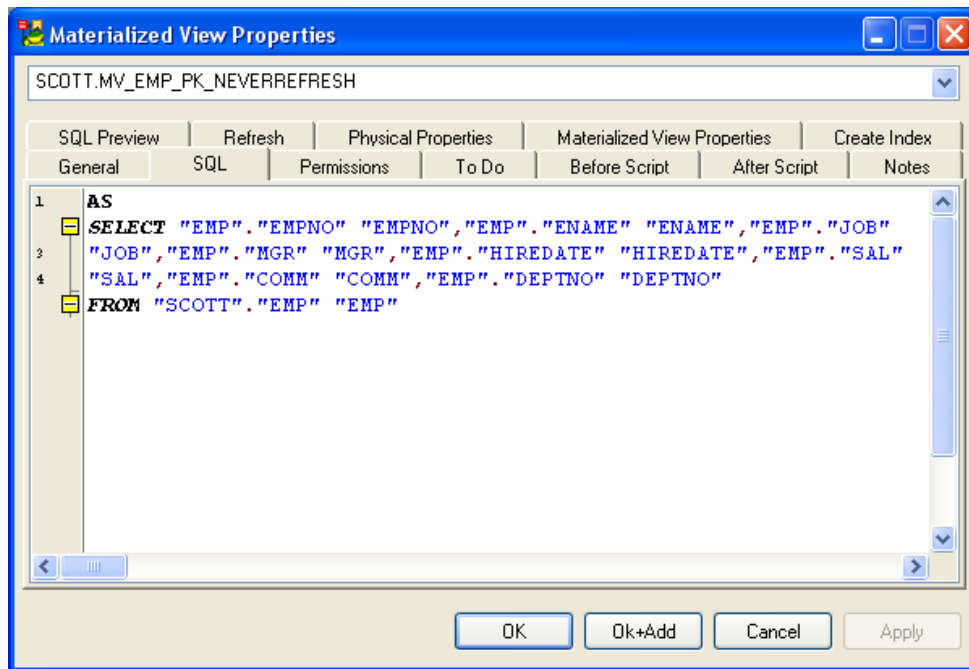
☐ For Update

Query Rewrite: DISABLE

Categories: -- None --

☒ Generate ☐ Generate SQL Only

OK Ok+Add Cancel Apply



## Procedures, Functions, Schemas, Synonyms, Defaults...

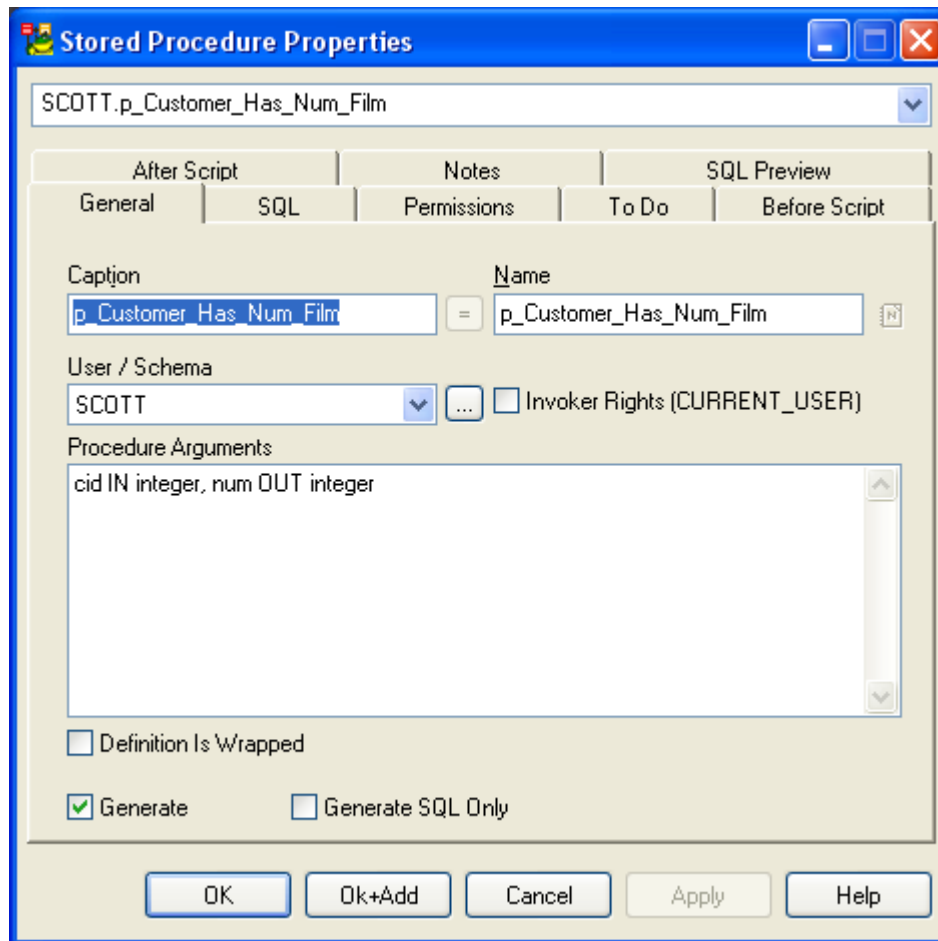
Toad Data Modeler supports procedures, functions, schemas etc., plus other database objects that are supported by particular database system, e.g. Assemblies in SQL Server 2005 database, Sequences in Oracle database etc. (To see the list of these objects for particular database, please see the Help file, "Databases", "Extra Objects" topic.)

The database objects that are supported or used in (almost) all database systems are accessible via the **Model** menu and are listed also in **Model Explorer**. (See **Model | Functions, Procedures, Schemas...**)

Database objects that are specific for particular database system are available only in Model Explorer on appropriate folders. (See Model Explorer of SQL Server 2005 model | **Assemblies** folder.)

From both places, you can add, edit and delete them. - See the appropriate buttons e.g. in the **Schemas** dialog (**Model | Schemas**), or see the pop-up menu of the selected folder or item in Model Explorer (right-click the **Sequences** folder, right-click the selected sequence).

**Example:** Procedure for Oracle 10g model



The image shows a 'Stored Procedure Properties' dialog box. At the top, there is a title bar with a Toad icon and the text 'Stored Procedure Properties'. Below the title bar is a dropdown menu showing 'SCOTT.p\_Customer\_Has\_Num\_Film'. The main area of the dialog is divided into several tabs: 'General', 'SQL', 'Permissions', 'Notes', 'SQL Preview', 'To Do', and 'Before Script'. The 'General' tab is currently selected. Inside the 'General' tab, there are two text boxes: 'Caption' and 'Name'. The 'Caption' box contains 'p\_Customer\_Has\_Num\_Film' and the 'Name' box contains 'p\_Customer\_Has\_Num\_Film'. Below these is a 'User / Schema' dropdown menu showing 'SCOTT'. To the right of the dropdown is a checkbox labeled 'Invoker Rights (CURRENT\_USER)'. Below the 'User / Schema' dropdown is a text area labeled 'Procedure Arguments' containing the text 'cid IN integer, num OUT integer'. At the bottom of the dialog, there are five buttons: 'OK', 'Ok+Add', 'Cancel', 'Apply', and 'Help'. There are also two checkboxes: 'Definition Is Wrapped' and 'Generate' (which is checked).

Stored Procedure Properties

SCOTT.p\_Customer\_Has\_Num\_Film

After Script    Notes    SQL Preview  
General    SQL    Permissions    To Do    Before Script

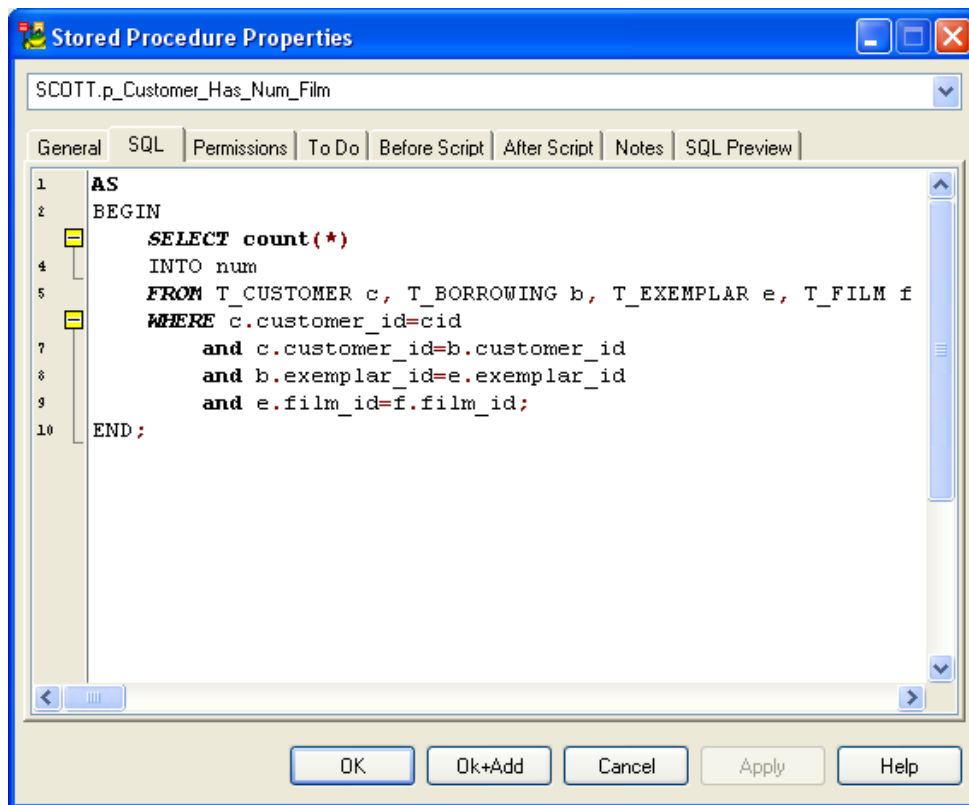
Caption    Name  
p\_Customer\_Has\_Num\_Film    =    p\_Customer\_Has\_Num\_Film

User / Schema  
SCOTT    ...    ☐ Invoker Rights (CURRENT\_USER)

Procedure Arguments  
cid IN integer, num OUT integer

☐ Definition Is Wrapped  
☒ Generate    ☐ Generate SQL Only

OK    Ok+Add    Cancel    Apply    Help




## Users and User Groups

In Toad Data Modeler, you can define users, user groups and assign permissions to them.

### Scenario

Create users in your model - two administrators and a manager, and user groups *Administrators* and *Managers*.

#### *To create users*

1. Click  on the toolbar.
2. In the **Users** dialog, click **Add** three times and confirm **Apply**.
3. Select *User1* and click **Edit**.
4. Define properties on tab **General** for this user *Administrator1* and confirm **Apply**.

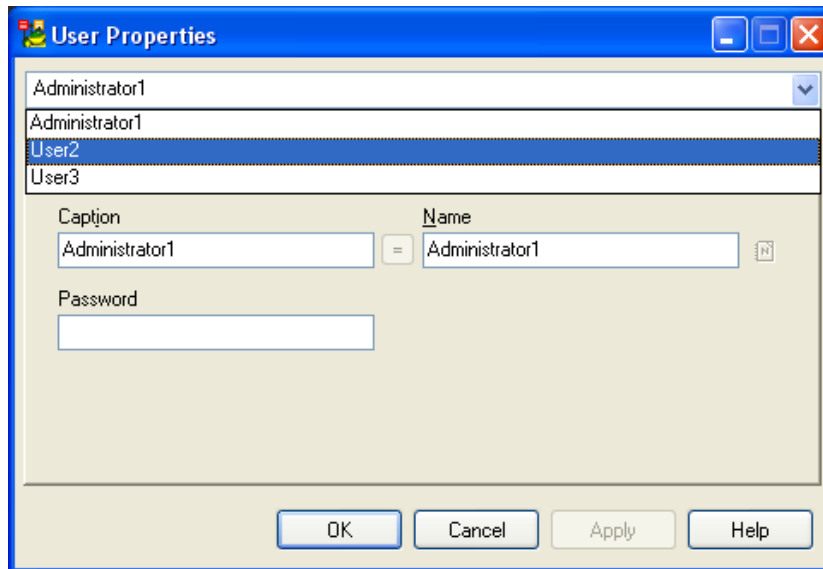
**Note:** As you haven't defined user groups yet, there is no group available on tab **Membership**. Nevertheless, Toad Data Modeler allows you to concurrently create users and user groups - thanks to the modeless dialogs. You can leave the *User1*



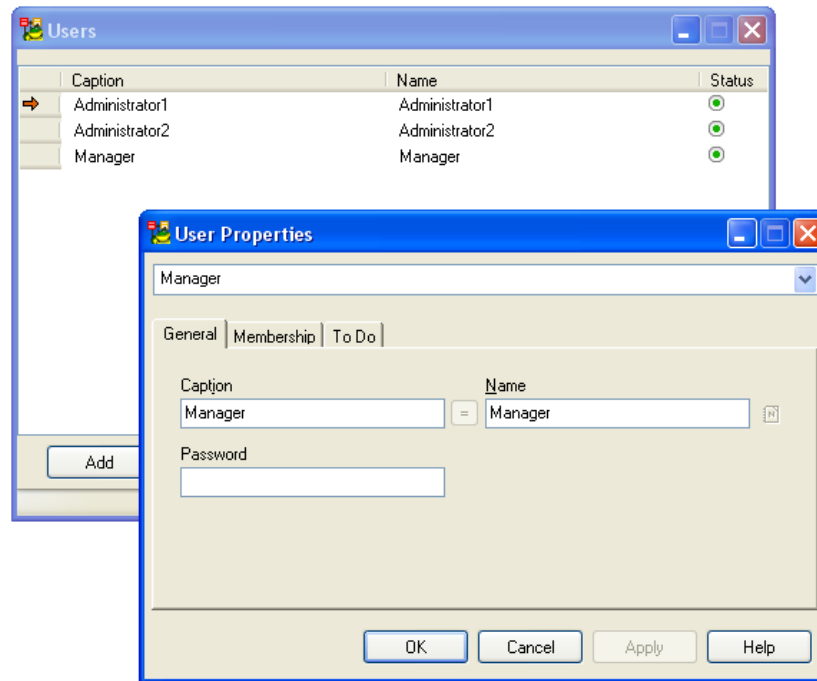
Properties dialog opened and at the same time click the **User Groups** icon and define a new user group.

**Tip:** Generally, you can assign users to user groups either in the **User Properties** dialog or **User Group Properties** dialog. You can select the way that is most comfortable for you.


5. From the Object Navigator box, select another item - *User2*.



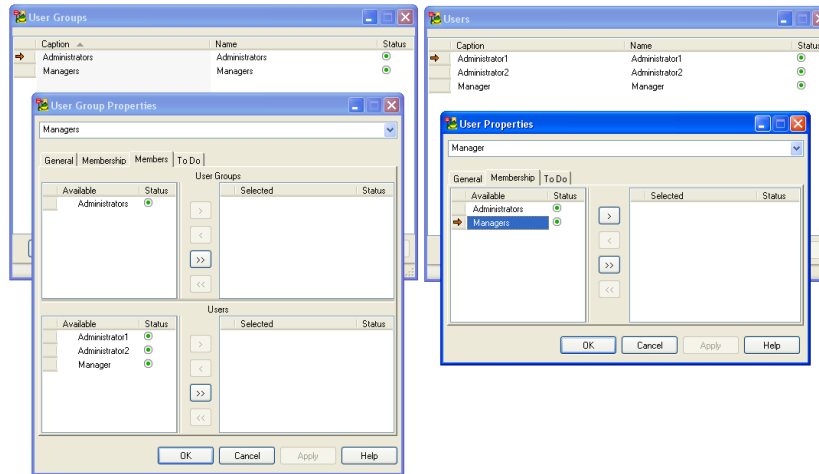
6. Again, define properties and confirm **Apply**. Do the same for the last item.
7. Finally, feel free to confirm the changes **Apply** and leave the dialogs **Users** and **User Properties** opened.



#### *To create user groups*

1. Click  on the toolbar.
2. In the **User Groups** dialog, click **Add** twice and confirm **Apply**.
3. Double-click *UserGroup1*.
4. Define name of this group - *Administrators* and confirm **Apply**.
5. From the Object Navigator box, select the other item.
6. Write its name - *Managers* and confirm **Apply**. -> The dialog remains opened.
7. Click the **Members** tabs of the **Managers User Group Properties** dialog. Here, available user groups and users are listed, and you can add a user group and users to the *Managers* user group.
8. Switch to the **Manager User Properties** dialog and click the **Membership** tab. -> Two available user groups are there. You can assign the *Manager* user to the selected group.

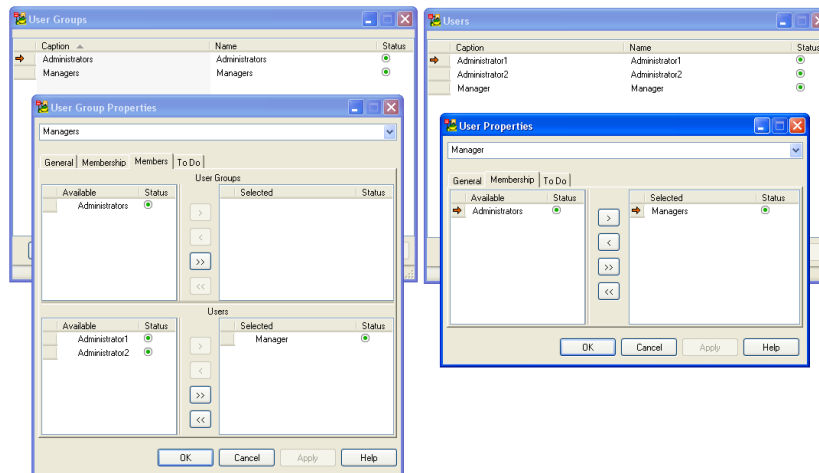
Compare the dialogs in the screenshots below:



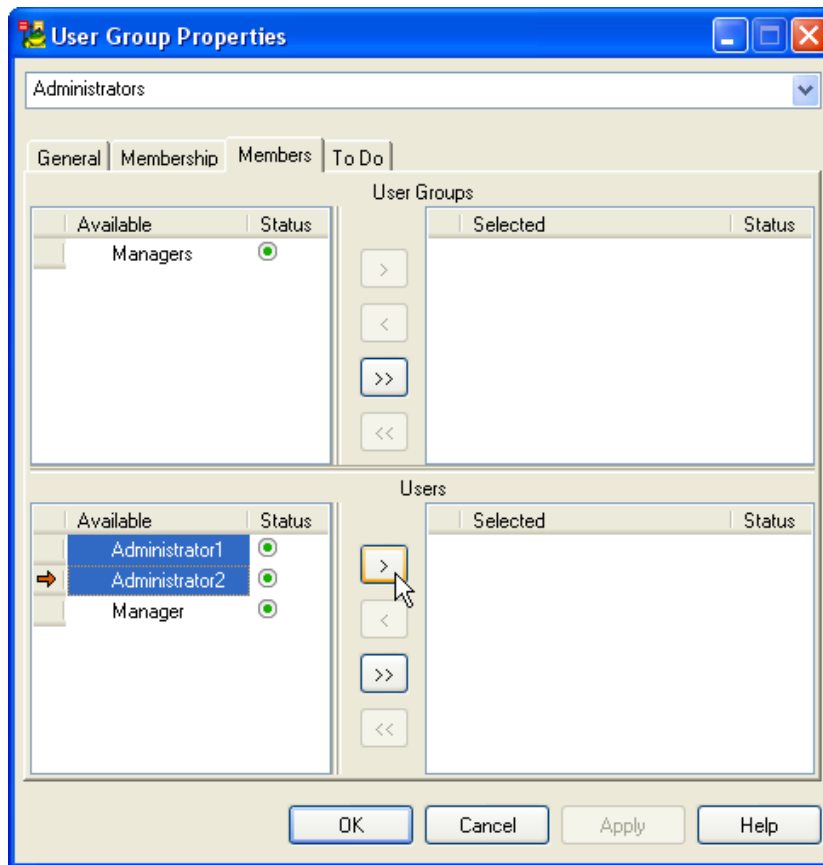
As you can see, you can concurrently work with users and user groups. Select groups for users and assign users to user group in both dialogs.

9. In the **Manager User Properties** dialog, select the *Managers* group and confirm **Apply**. -> Managers item has moved to the window **Selected**. Also, see the **Managers User Group Properties** dialog - user *Manager* has been assigned to this group automatically too.

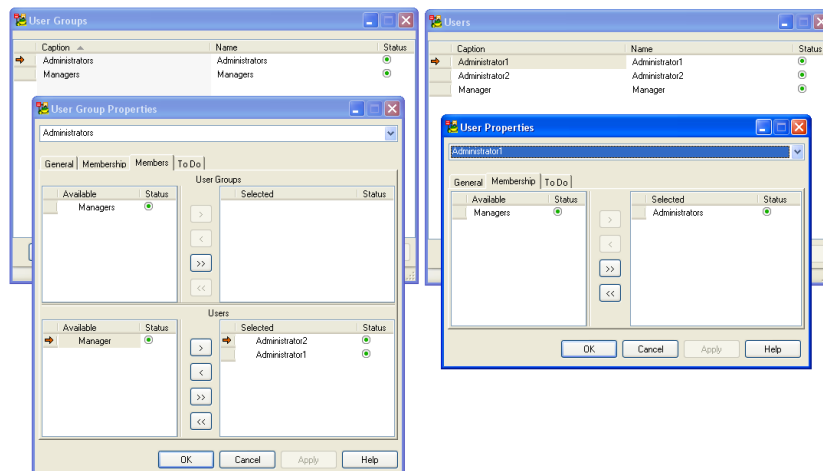
See the change in dialogs:



10. Let's see how it works in the **User Groups** dialog now: From the Object Navigator box, select the *Administrators* user group.
11. Select the two administrator users (use SHIFT key) and click the **Add** arrow button.



12. Confirm **Apply**. (You can check out e.g. the **Admin1 User Properties** dialog to see the change also there - use Object Navigator box again.)



13. Close the dialogs **OK**.

**Note:** Users and User Groups are accessible in the **Model** menu. You can also manage them via **Model Explorer**.

## Permissions

You can assign permissions for the following objects: Entity, Attribute, User Data Type, View, Procedure, Schema, Users and User Groups.

A list of permitted objects varies according to the database you use. E.g. in some databases, it's not possible to assign a permission for Users - e.g. in PostgreSQL database.

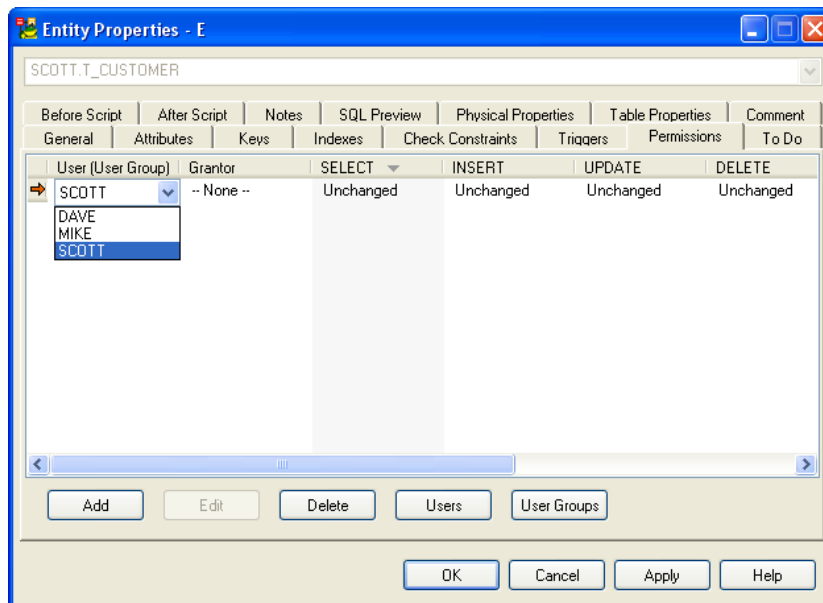
For every object, you can set different permissions (SELECT, INSERT, UPDATE etc.). Again, it depends on the selected database.

Permissions are set on the **Permissions** tab of particular object **Properties** dialog.

To be able to add a permission, it's necessary to define a User or User Group. If you forget to define them first, never mind. Toad Data Modeler allows you to define them directly on tab **Permissions** of appropriate object - see the **Users** and **User Groups** buttons in the screenshot below.

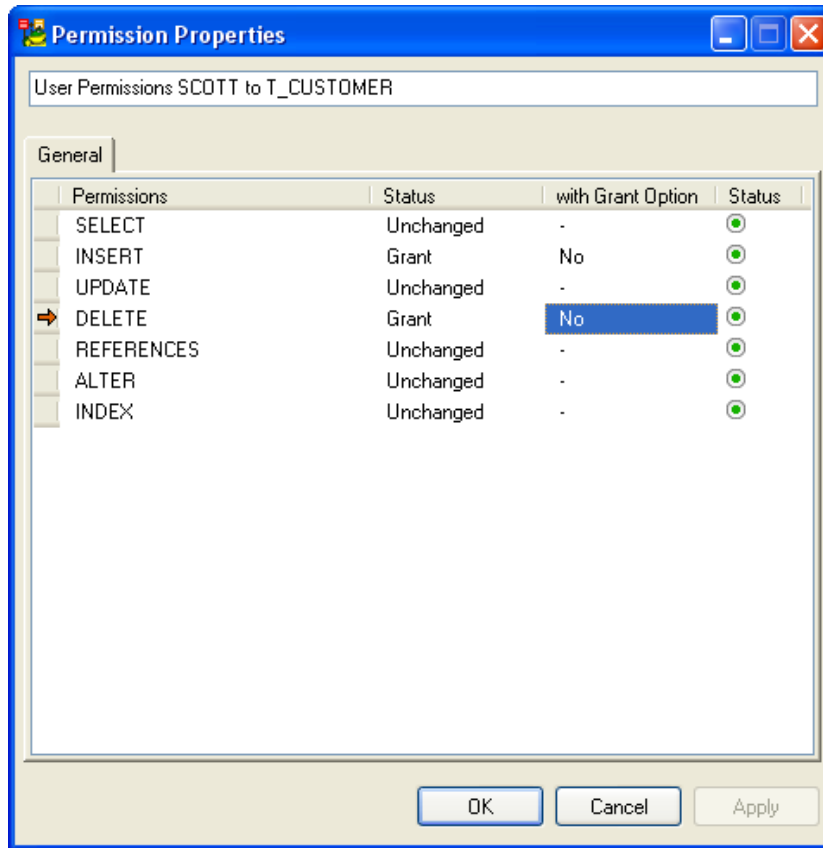
### Example: Entity Permission:

1. Edit an entity and click the **Permissions** tab.
2. Click **Add** and confirm **Apply**.
3. The newly created permission has been assigned to first available user automatically. If you want to change it, click the user item, wait a while and click it again (or press F2).



4. Select a user or user group.
5. Change **Grantor** the same way if you need.
6. Confirm **Apply**.

7. Via the Inplace Editor, you can stay in the **Entity Properties** form and go through all the commands. However, for much quicker settings, edit the permission.
8. In the **Permission Properties** tab, set the permitted actions. Finally confirm **OK**.



## Naming Conventions

### About Naming Conventions

Toad Data Modeler allows you to use naming conventions in your physical models. What are naming conventions? They allow you to:

- Define your own rules for physical names of objects – e.g. add prefix or suffix to the names, use only lower case etc.
- Synchronize caption (logical name) and name (physical name) of objects of your model.
- Verify whether the physical names of objects match the naming convention rules. If they don't, you can decide whether to apply them or not.

**Note:** Naming conventions always influence physical names, NOT captions. Captions (logical names) can be in various forms as they provide logical information only.

## Naming Conventions in Toad Data Modeler – Basic Information


- Naming conventions are supported for most objects of physical model for which you can define a name (physical name) and a caption (logical name).
- A naming convention can be used in various models across different database systems. You can use them for new models, reversed models or models that already exist in Toad Data Modeler.
- Every defined naming convention is stored in external .TXN file stored by default in your Documents folder, directory Toad Data Modeler\ Installation name\Naming Conventions.
- You can define only one naming convention for one model.
- Toad Data Modeler allows you to:
  - Automatically create physical names based on logical names and naming conventions rules. (Example: Customer ID => T\_CUSTOMER\_ID). This feature makes you more productive as you need significantly less time to define both logical and physical names.
  - Use glossaries for automatic word replacement during new objects creation. Glossaries can be made of multiple CSV files and you can easily turn on or turn off the selected glossaries.
  - Verify naming conventions and rewrite physical names in a batch mode. Check your model and replace physical names of objects that don't match naming conventions rules.
  - Synchronize captions and names (without specific naming convention definitions).
  - Import/export of glossaries (\*.CSV files) also from/to other tools.  
See the **Naming Convention Properties** dialog | **Glossary** tab | **Import** button. You can find some CSV files with diacritical marks at: C:\Program Files\Quest Software\Toad Data Modeler 3\Naming Conventions\CSV.

## Related Topics


Basic Operations (page 108)

## Basic Operations

### *To define a new naming convention for your model*

1. Click  on the toolbar (or select **Settings | Naming Convention**).
2. Click **Add New Naming Convention**.
3. In the **Adding Naming Convention** dialog, define the name and confirm **OK**.
4. Select the naming convention for the model and confirm **OK+Edit**.
5. Define the naming convention rules in the **Naming Convention Properties** dialog.

### *To link existing naming convention to your model*


1. Click  on the toolbar (or select **Settings | Naming Convention**).
2. Select the naming convention.

**Note:** You can also select a naming convention to your model in the **Model Properties** dialog | **Naming Convention** tab.

3. Click **OK+Edit**.

### *To verify whether the naming convention rules match physical names in your model*

(and also to decide whether you want to apply them for particular item or not)

1. Click  on the toolbar (or select **Model | Verify Naming Conventions**).
2. In the **Naming Convention Verification & Synchronization** dialog, you can see a list of items that do not match either the naming convention rules (**NC** column) or the synchronization rules (**Sync** column).  
Compare the items in the **Physical Name** column (current physical name) and **Expected Physical Name** column (physical name after the naming convention rules are applied).

**Note:** If more variants of the physical name exists, there is a combo box in the **Expected Physical Name** column. From this combo box, you can choose the preferred name.

**Tip:** Point the mouse cursor at the icons or click the selected row to learn more details.

3. Select the items for which you want to apply the rules.
4. Click **Update** and **Close** then.

### *To edit a naming convention linked to your model*

- » Click  on the toolbar.

### *To manage naming conventions (add, edit, delete, rename)*

- » Select **Tools | Naming Conventions**.



### *To unlink the naming convention from your model*

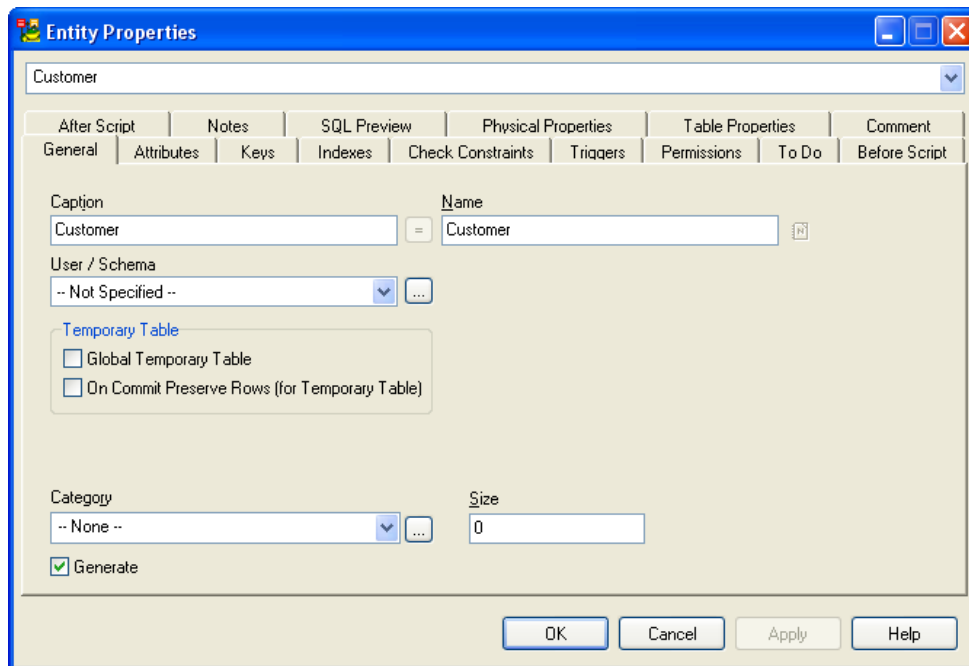
1. Select **Model | Model Properties | Naming Convention** tab.
2. From the **Name** box, select ---.
3. Confirm **OK**.

### Related Topics

Default Synchronization of Caption and Name (page 109)

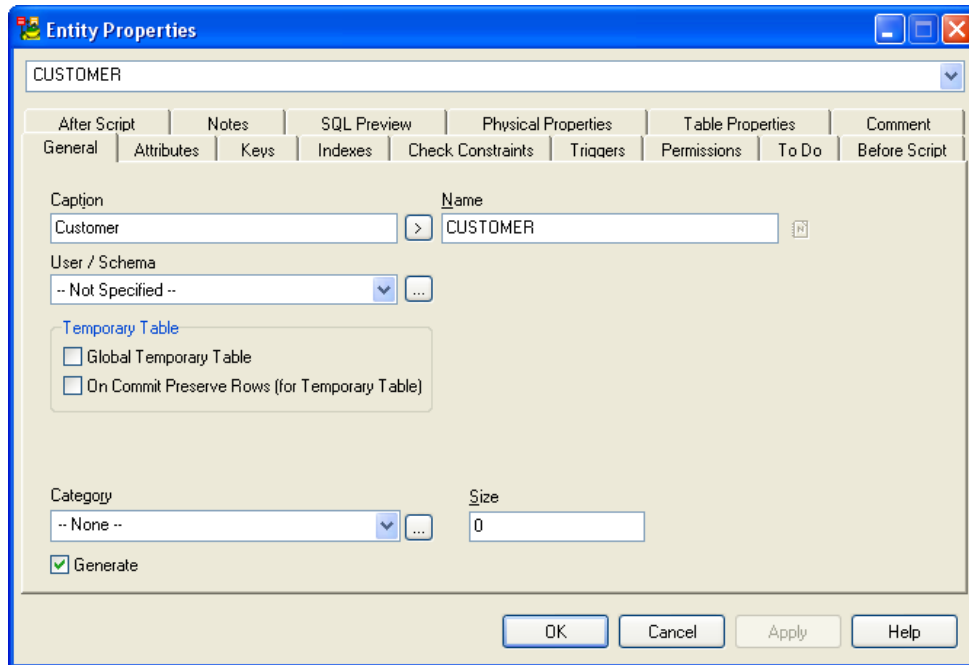
## Default Synchronization of Caption and Name

In new models, automatic synchronization is turned on by default. It means that when you write a logical name, physical name will be automatically the same. See the example below and take notice of the '=' icon between **Caption** and **Name**.

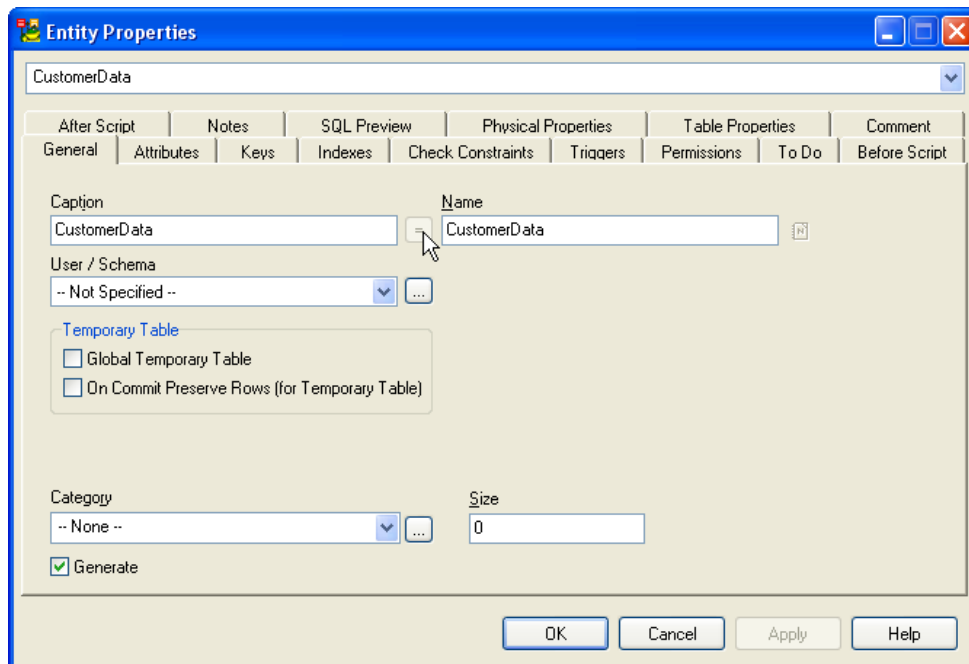


Now, when you change the caption, the name will change too.

Once you change the physical name, you turn off the automatic synchronization automatically. See the example below and take notice of the changed icon '>' between the **Caption** and **Name** boxes.



Now, when you change the caption, the name will not change automatically. However, if you do want to change the name too, simply click '>' and it will change to '=' again. Caption and Name will be identical again and synchronized again.



This is how the default synchronization works in new models by default. Now read the "Scenarios" topics showing how you can use the Naming Conventions feature.

## Related Topics

Scenario 1 (page 111)

Scenario 2 (page 119)

Scenario 3 (page 122)

Scenario 4 (page 125)


## Scenario 1

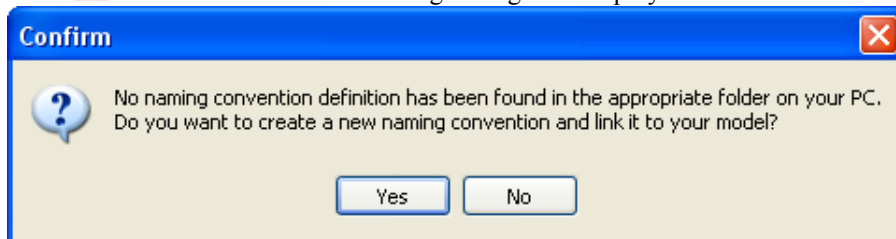
### Scenario

You are going to create a new Oracle 10g model and want to define a naming convention for this model (first in Toad Data Modeler). Physical names of entities should look like this: *T\_UPPERCASE* (prefix\_upper case).

You expect the following behavior: When you enter a caption of a new entity, the defined naming convention rules will be automatically applied in the name of the entity: *Customer* => *T\_CUSTOMER*.

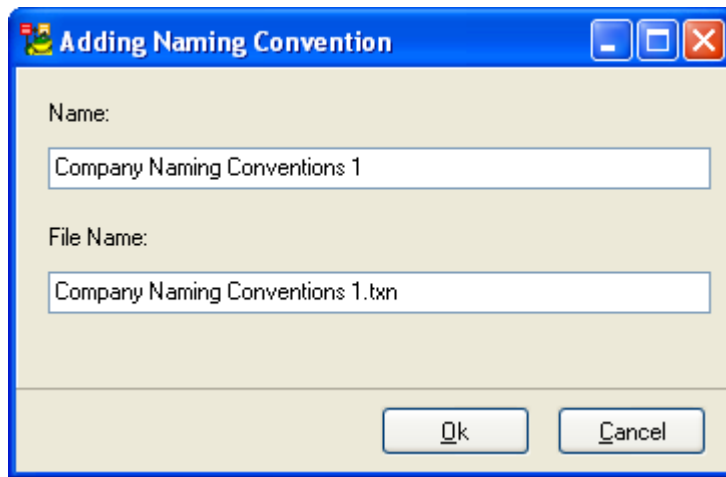
The new model is open and active in the Application Window.

1. Click  on the toolbar. The following dialog will display:



Toad Data Modeler hasn't found any .TXN file with a naming convention and therefore asks you to create it.

2. Click **Yes**.
3. Define name of the new naming convention and confirm **OK**.



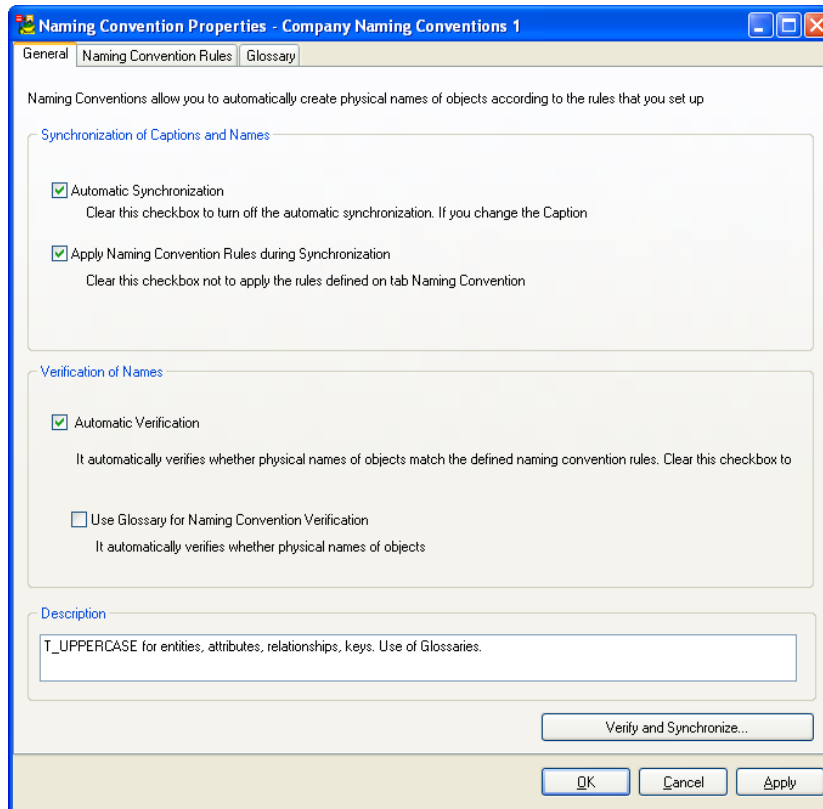
New *Company Naming Conventions 1.txn* file has been created in the default location – your Documents folder, Toad Data Modeler\Installation name\Naming Conventions.

The new naming convention **Properties** dialog opens automatically.

4. Define properties of the naming convention on tab **General**.

**Automatic Synchronization + Apply Naming Convention Rules during Synchronization** checkboxes - Leave these checkboxes selected. It means that when you write a caption, the name will automatically synchronize plus at the same time the naming convention rules will be applied.

To the **Description** box, you can write a brief description on the naming convention.



5. Click the **Naming Convention Rules** tab to define the rules.

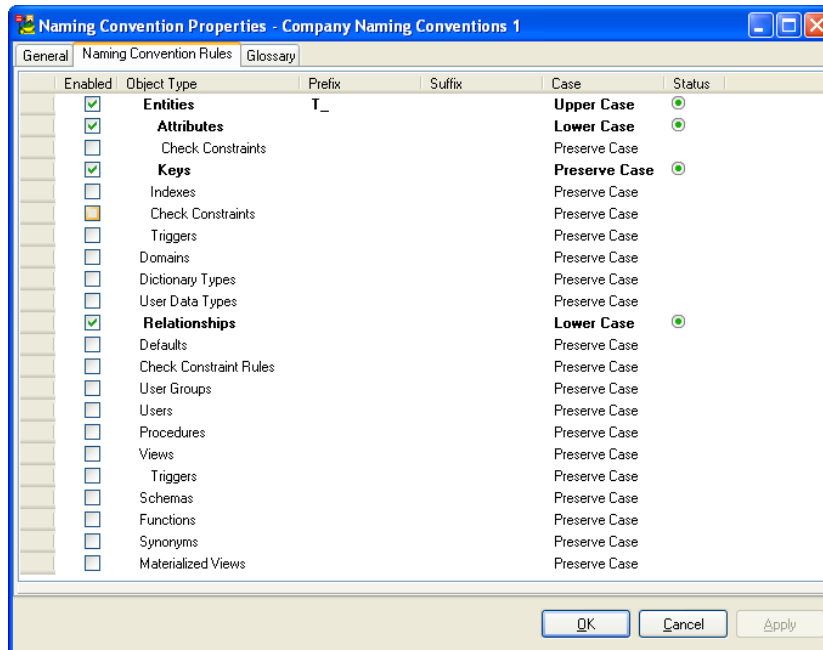
(To define or edit the values, press F2.)

Select the **Entities** checkbox, define the prefix *T\_* (press F2) in the **Prefix** column and from the **Case** column, select *Upper Case*.

**Note:** Press **Enter** to confirm each defined rule.

Define other settings for attributes, keys and relationships.

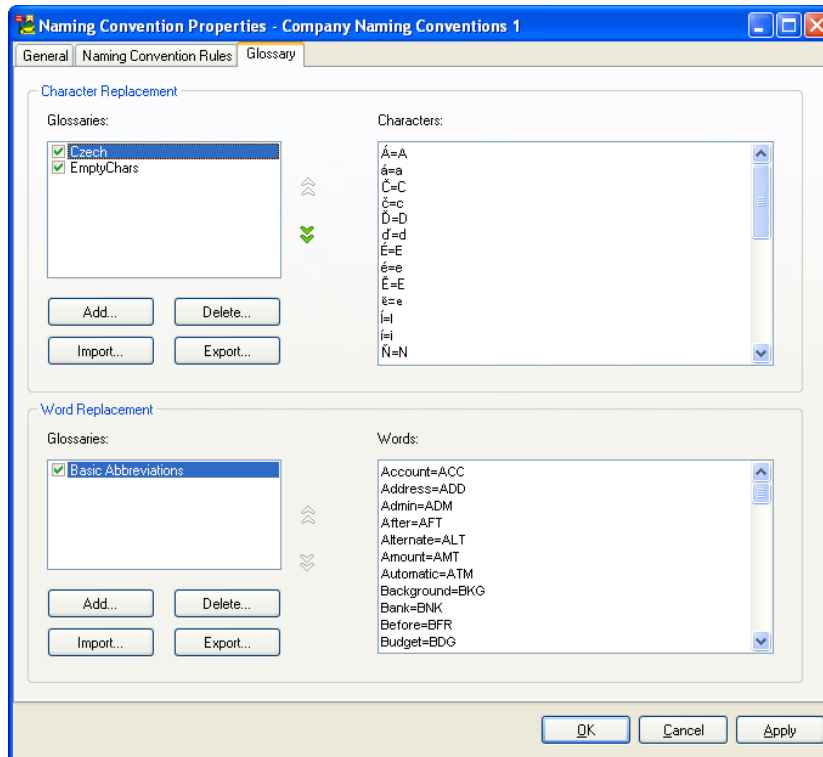
**Tip:** As soon as you confirm the defined rule, the **Enabled** checkbox for particular item will select automatically.



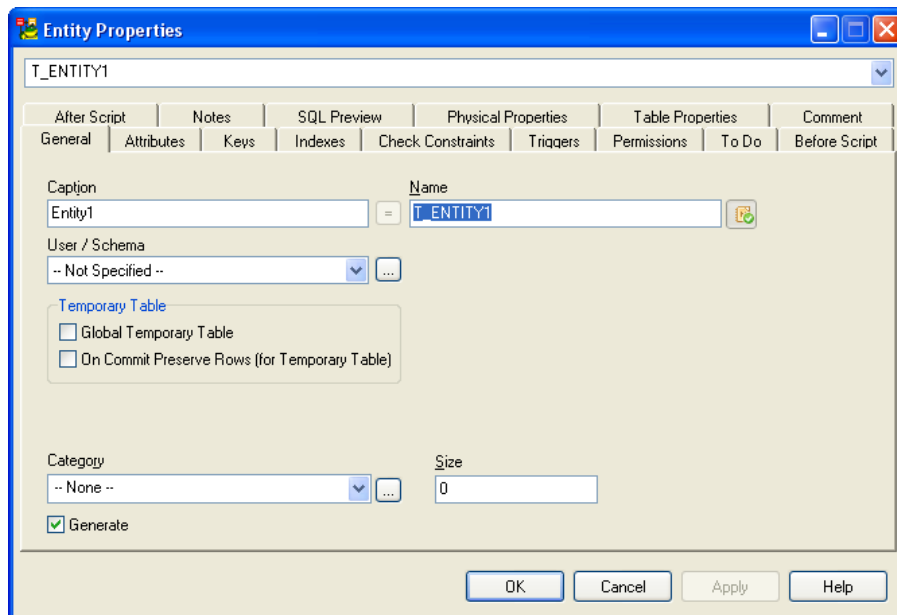
6. Click the **Glossary** tab.

On this tab, you can define what characters and words should be replaced. Here you can also import and export the .CSV files.

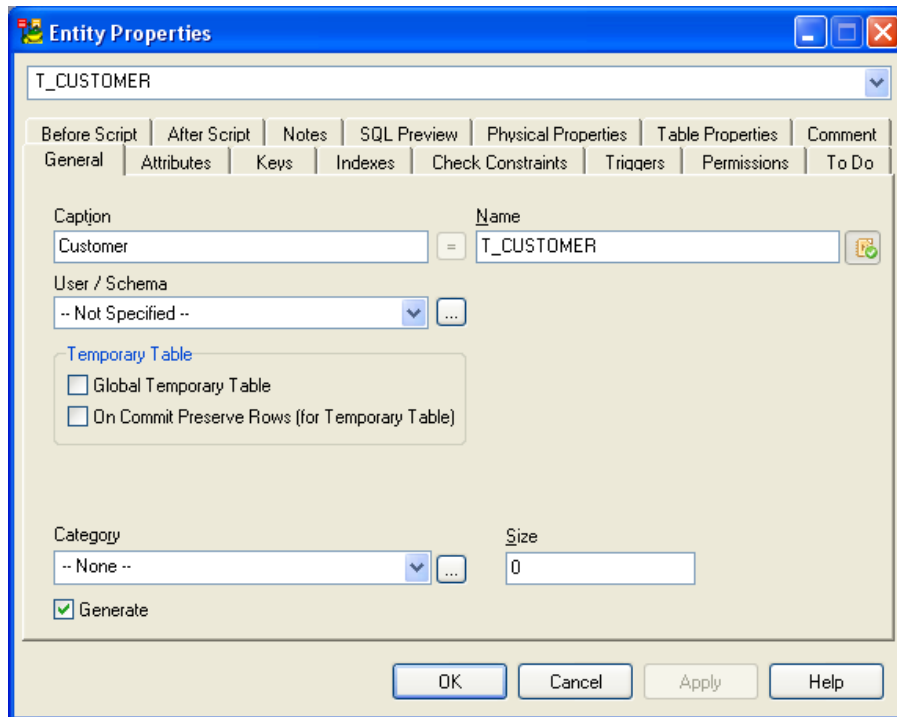
In the example below, some CSV files were imported from C:\Program Files\Quest Software\Toad Data Modeler 3\Naming Conventions\CSV.



7. Click **OK** to update the *Company Naming Conventions 1.txn* file.
8. Create a new entity on the Workspace and edit it. See the **Caption** and **Name** boxes.  
Take notice of the icon next to the **Name** box. It says: "Naming convention rules are turned on."



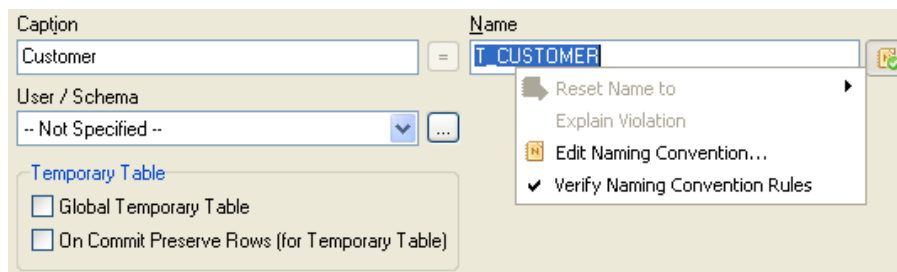
9. Enter the entity logical name to the **Caption** box: *Customer*. See the **Name** box then.



This way you will create other entities of your model. The naming convention rules will be applied automatically.

### Additional Information:

- Right-click the **Name** box in the **Entity Properties** form opens the pop-up menu for naming conventions. As you can see, you can edit the naming convention comfortably (**Edit Naming Convention**), run the naming convention verification etc.

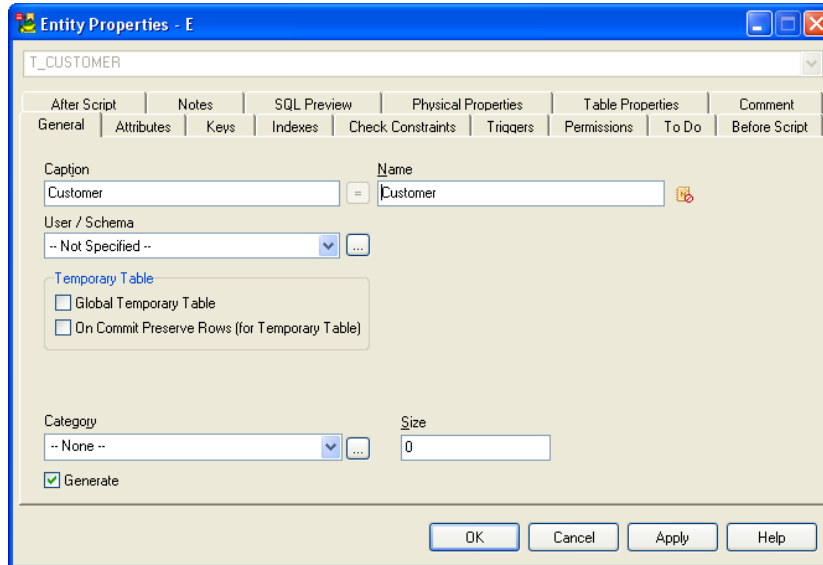


- Toad Data Modeler enables you to turn off the naming convention for the particular entity. – Click the icon next to the **Name** box. See the change in the screenshot below.

The icon now says: "Naming convention rules are turned off." In fact, it says that a naming convention for this model exists but is not applied for this entity.

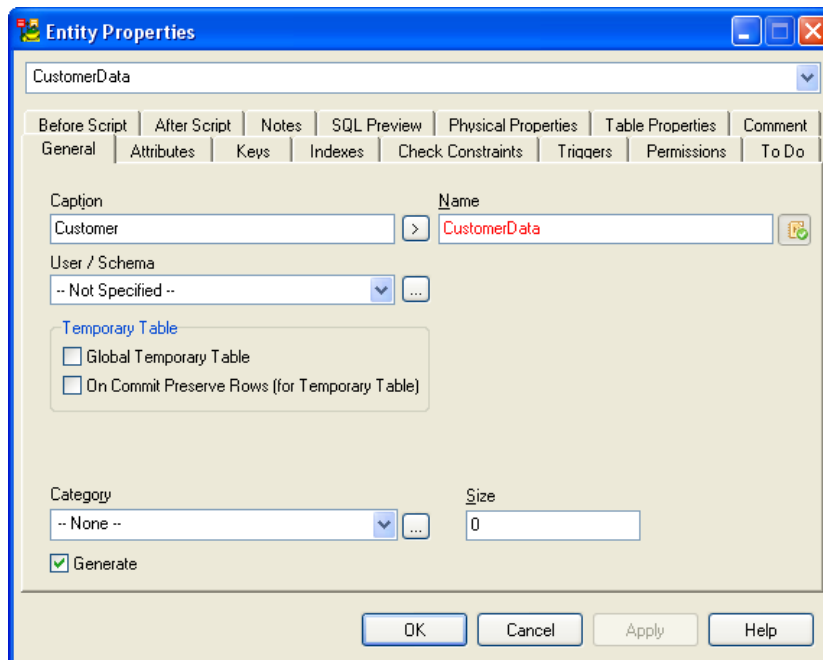


Caption and Name have been synchronized by the TDM default synchronization rules automatically.



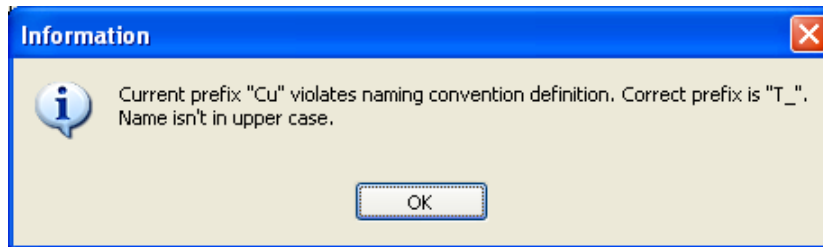
Now imagine this situation: You manually write a new physical name to the **Name** box - *CustomerData* - and click the icon again – to turn on the naming convention again.

The name becomes highlighted in red, which means that the name does not match the defined naming convention rules.

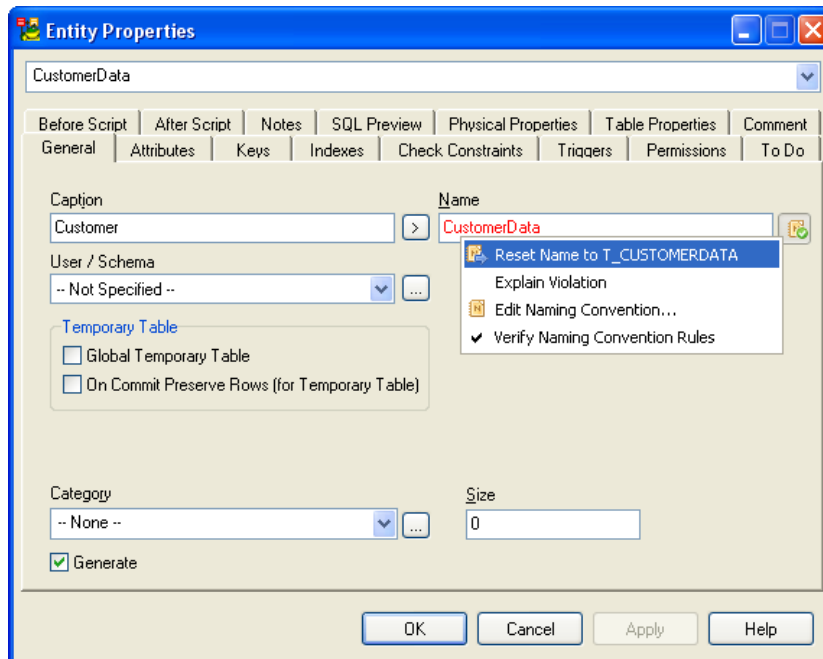


To find out what is wrong here, right-click the **Name** box and select **Explain**

**Violation.** The following dialog will display:



To apply the naming convention rules, right-click the **Name** box and select **Reset Name to...** Toad Data Modeler offers you a new physical name that matches the naming convention rules.



## Related Topics

Scenario 2 (page 119)

Scenario 3 (page 122)

Scenario 4 (page 125)


Use Cases - Different Naming Convention Settings (page 127)

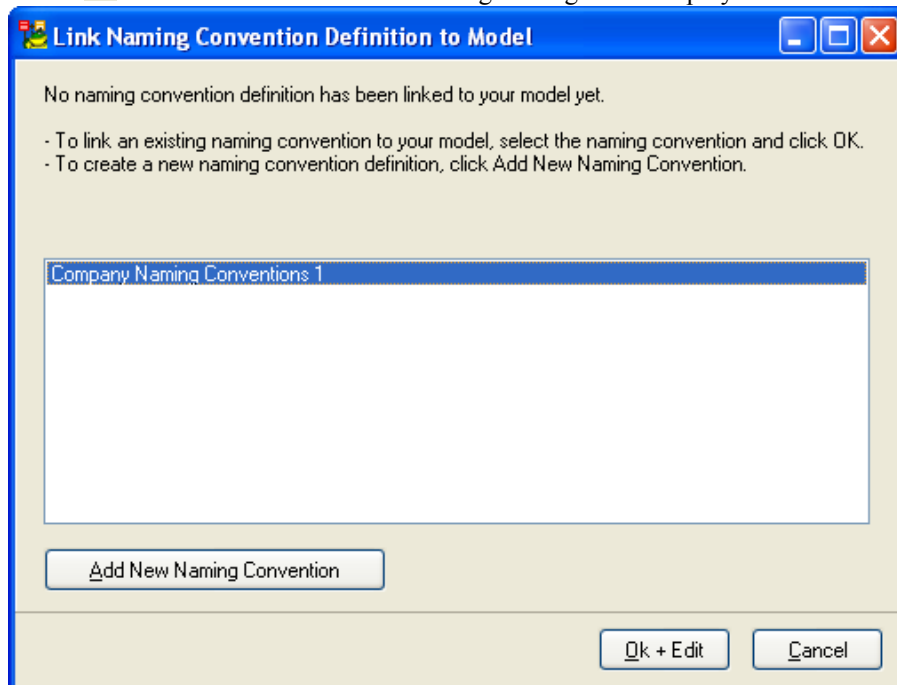
## Scenario 2

### Scenario

In Toad Data Modeler, you have Oracle 10g model *Videorental*. This model had been created before the naming conventions were available. Now you want to apply an existing naming convention in this existing model.

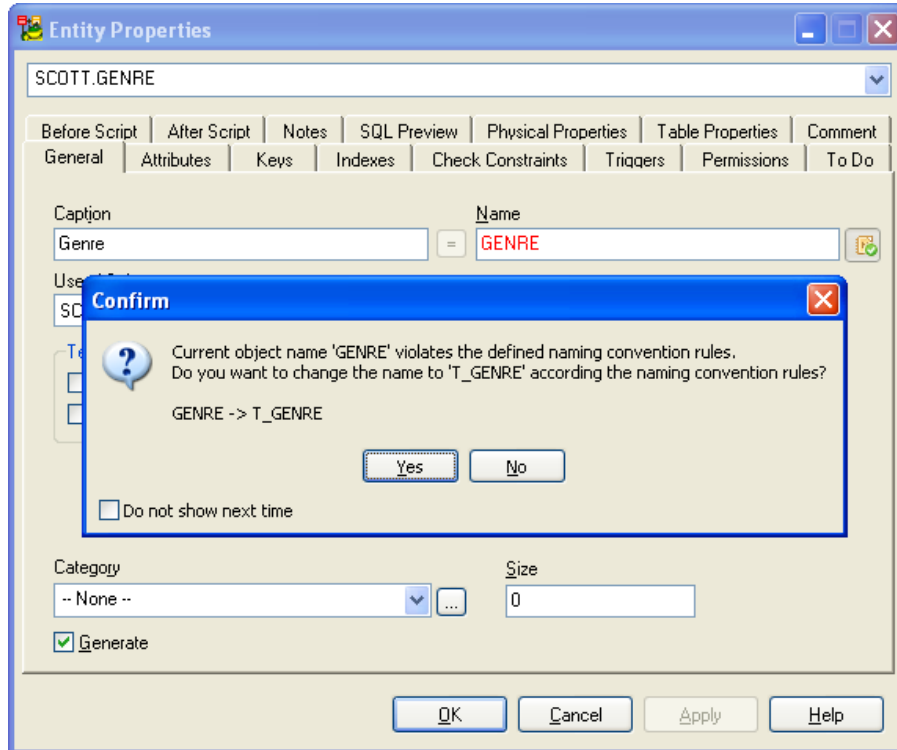
For new objects that you will create in the model, the defined naming convention will be applied automatically.

1. Open model *Videorental*.
2. Click  on the toolbar. The following dialog will display:



3. Select the existing naming convention and click **OK+Edit** to check its definition.  
**Note:** It is the naming convention defined in Scenario 1 (Entity => T\_ENTITY).
4. Verify whether the settings of the existing naming convention fit your needs. If so, click **OK**.

- Now check out e.g. the existing *Genre* entity | **General** tab.



As you can see, the naming convention rule has not been applied yet. Why? – Toad Data Modeler will not change any data without your confirmation. Also, it allows you to select items for which you want to apply the defined naming convention rules.

Now you can click Yes in this entity dialog and then go through other objects of your model, however it is not an ideal way. So, click **No** and let's do verification of the whole model.

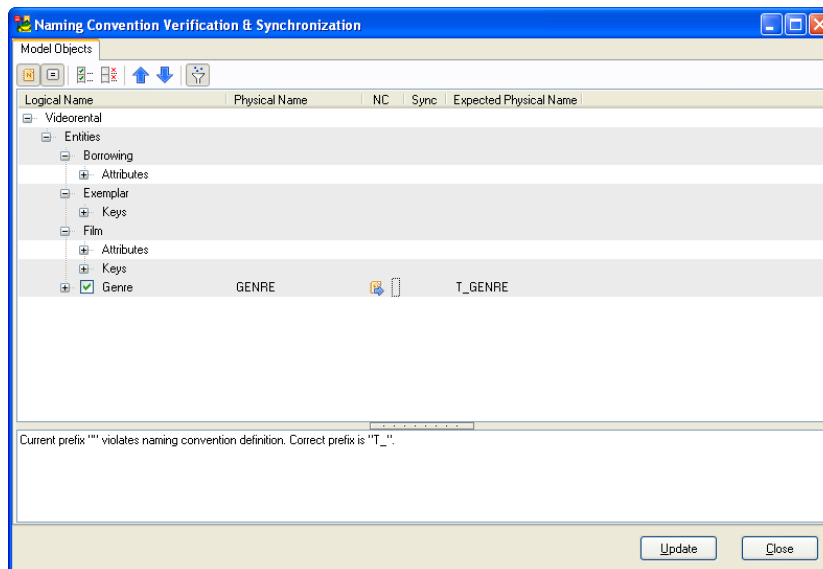
- Click  on the toolbar.

**Note:** You can run the verification also from the **Naming Convention Properties** dialog | **General** tab | **Verify and Synchronize**.

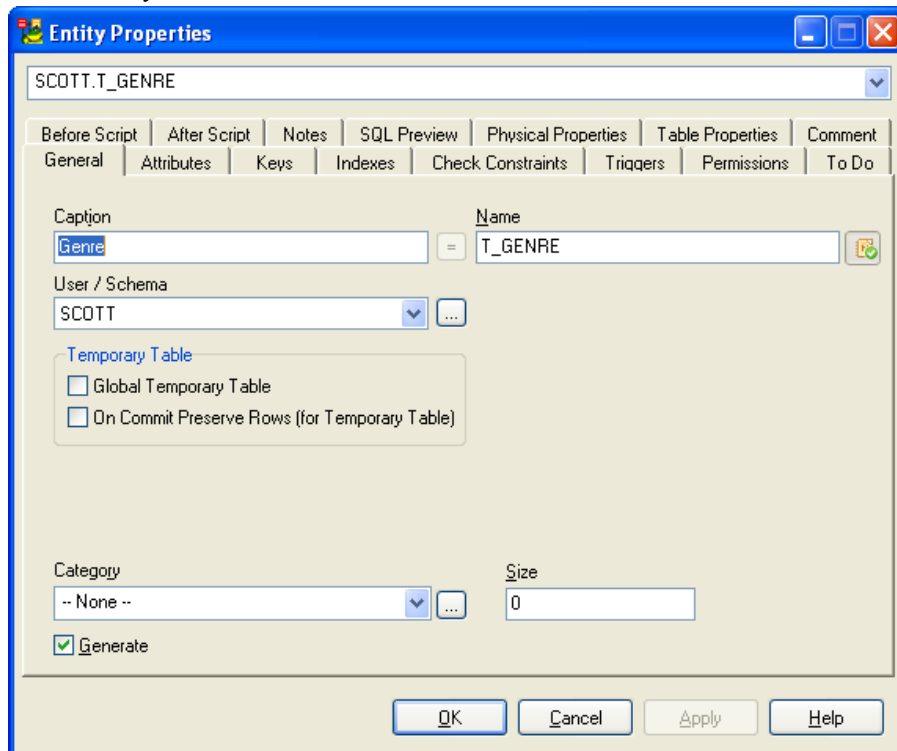
In the **Naming Convention Verification & Synchronization** dialog, you can see a list of items that do not match the naming convention rules (**NC** column) and the synchronization rules (**Sync** column).

Compare the items in the **Physical Name** column (current physical name) and **Expected Physical Name** column (physical name after the naming convention rules are applied).

Point the mouse cursor on the icons or click the particular row to learn the details.



7. Select all items to apply the naming convention and synchronization rules in all items of the model.
8. Click **Update** and **Close** then.
9. See the entities and attributes on the Workspace. For example, see details of the *Genre* entity:



## Related Topics

Scenario 1 (page 111)

Scenario 3 (page 122)

Scenario 4 (page 125)

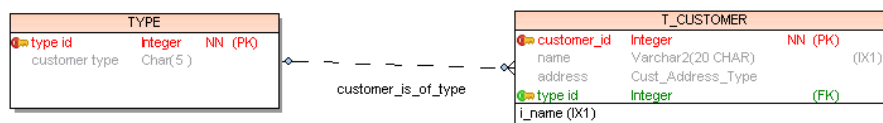
Use Cases - Different Naming Convention Settings (page 127)

## Scenario 3

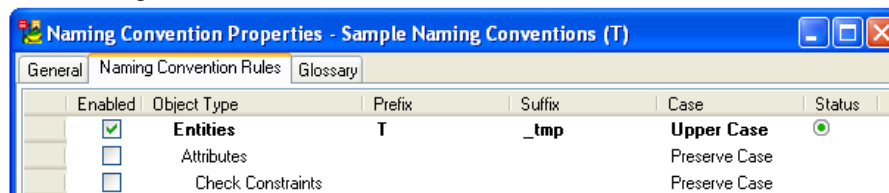
### Scenario

In your existing model, there is the *Type* entity. You have just defined a new naming convention (with rules: prefix *T* and suffix *\_tmp*) and linked it to the model. Now you are verifying the model with the naming convention rules. See what options Toad Data Modeler offers to you in this case.

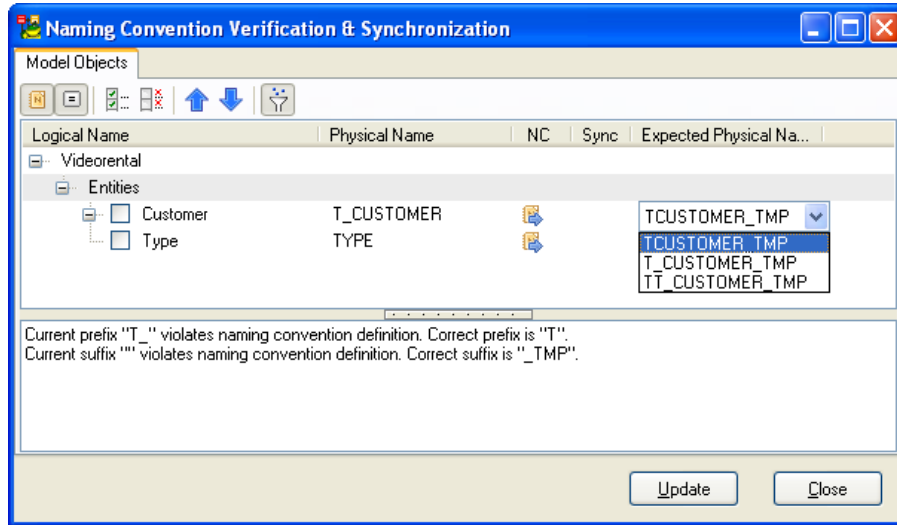
1. Entity on the Workspace:



2. The naming convention rules:



3. Click  on the toolbar.



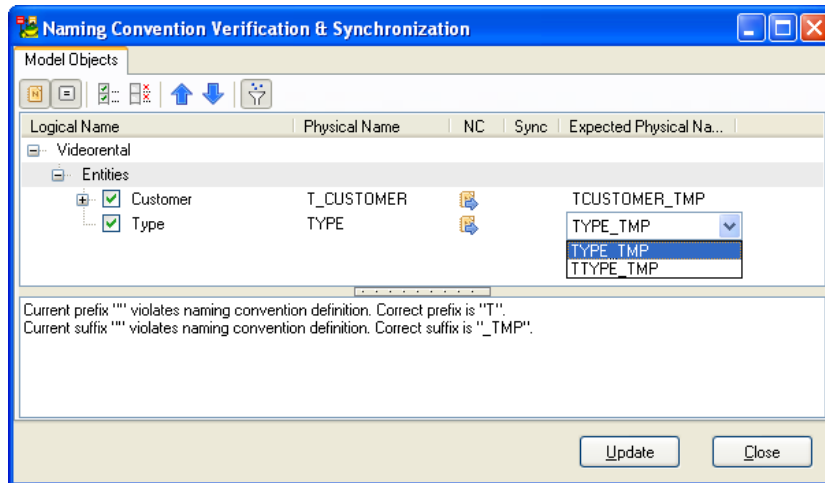
Physical names are not correct - that's what the dialog says.

**Note:** If the verification doesn't find any violations of the naming convention rules, the dialog is empty, as long as the filter icon is enabled.

Toad Data Modeler checks physical names and if there is any uncertainty about how the physical name should be defined, all possible options will be displayed in the combo box in the column **Expected Physical Name**. See the example below.

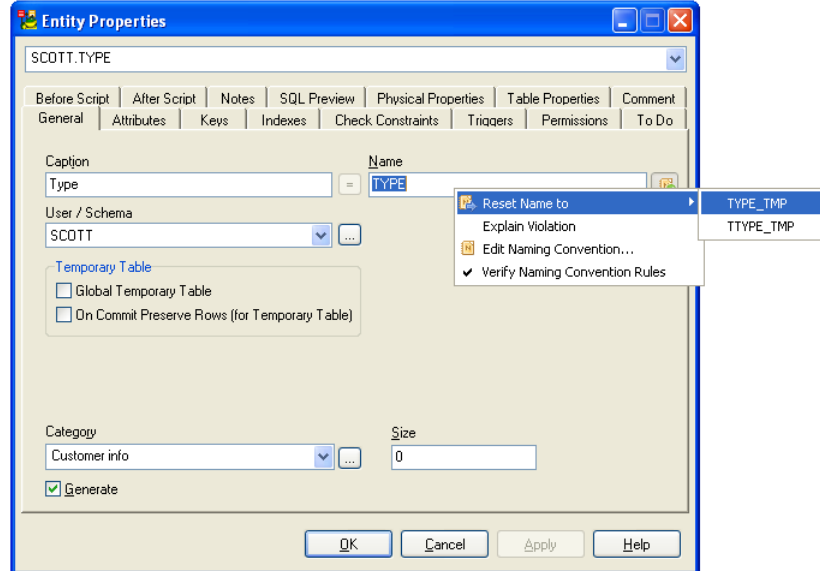
Click the *Type* item in the column **Expected Physical Name** and see variety of physical names that Toad Data Modeler offers to you for the particular entity - with respect to the defined naming convention rules.

The entity *Type* contains *T* char in the beginning. Prefix defined in the naming convention rules is *T*. Should the *Type* entity be renamed to *TTYPE\_TMP* or *TYPE\_TMP*? - The decision is up to you. Toad Data Modeler just offers you rational options and a way to change the names comfortably.



4. Select the preferred physical names and particular item checkboxes.
5. Click **Update** and **Close**.

**Note:** Using the **Naming Convention Verification & Synchronization** dialog you can change names at once, in a batch mode. The same action can be done e.g. for a particular entity in the **Entity** (or another object) **Properties** dialog. – Just right-click the **Name** box | **Reset Name to**.



## Related Topics

Scenario 1 (page 111)

Scenario 2 (page 119)

Scenario 4 (page 125)



Use Cases - Different Naming Convention Settings (page 127)

## Scenario 4

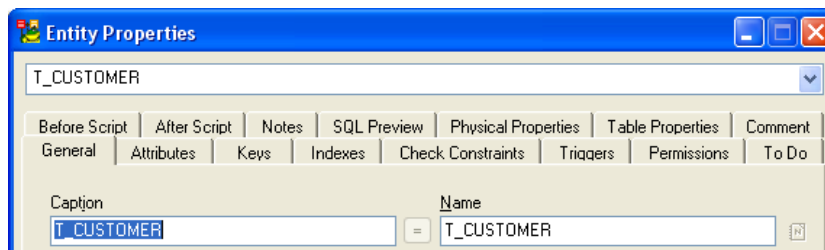
### Scenario

You reversed your Oracle 10g model to Toad Data Modeler. Now you want to modify logical names of objects but leave physical names unchanged.

Solution: Create a new naming convention and disable the Automatic Synchronization option.

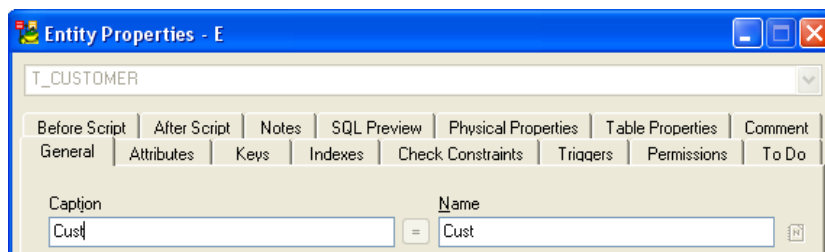
1. Activate the reversed model.


Just to check out, edit e.g. the *T\_CUSTOMER* entity. Logical name is *T\_CUSTOMER* and physical name is also *T\_CUSTOMER*.



As soon as you start editing the caption, you lose the physical name, which is what you do not want.

Click **Cancel** in the **Entity Properties** form and see how to solve this problem.

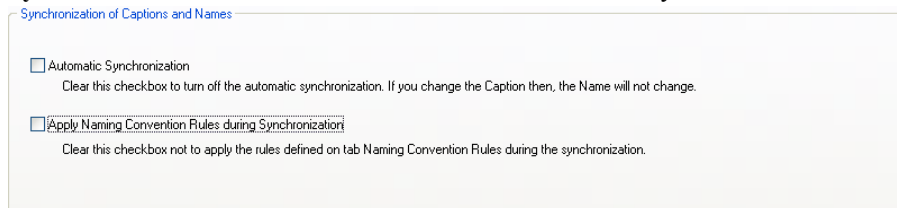


2. Click  on the toolbar.
3. Click **Add New Naming Convention**.
4. Define the new naming convention name and press **OK**.

5. Select the new naming convention and press **OK+Edit**.

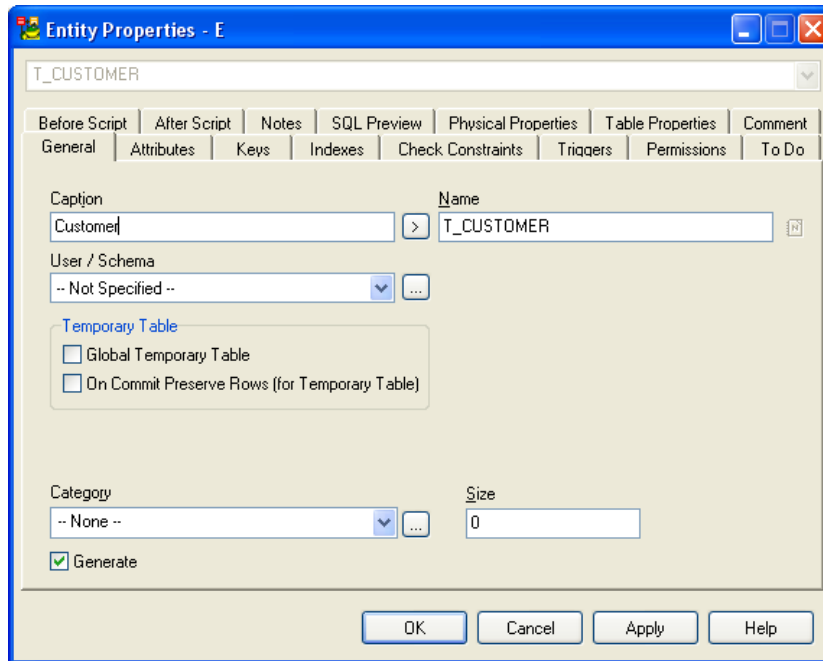


6. In the **Naming Convention Properties** dialog | **General** tab, clear the **Automatic Synchronization** checkbox to turn off the automatic synchronization.



7. Confirm **OK** in the **Naming Convention Properties** dialog.
8. Try to edit the entity again. Change its caption now.

As you can see, you can write a new logical name, the physical name will stay preserved.



## Related Topics


Scenario 1 (page 111)

Scenario 2 (page 119)

Scenario 3 (page 122)

Use Cases - Different Naming Convention Settings (page 127)

## Use Cases - Different Naming Convention Settings

Click  on the toolbar to open the **Naming Convention Properties** dialog of the naming convention that has been linked to your model.

On tab **General**, you can see options for synchronization of captions and names and options for verification of names. Read the following use cases for different settings of these options.

## Synchronization of Captions and Names

### A: Automatic Synchronization On, Apply Naming Convention Rules On

Synchronization of Captions and Names

☒ Automatic Synchronization  
Clear this checkbox to turn off the automatic synchronization. If you change the Caption then, the Name will not change.

☒ Apply Naming Convention Rules during Synchronization  
Clear this checkbox not to apply the rules defined on tab Naming Convention Rules during the synchronization.

If you enable both checkboxes, captions and names will be automatically synchronized, naming conventions rules will be applied automatically and glossaries will be used too.

**Example:** The new logical name defined as *Exemplar* will be converted to physical name to *T\_EXEMPLAR* automatically. *Customer* will be converted to *T\_CUS* etc.

**Use:** These settings are great for modeling new ER Diagrams according to naming convention rules. Automatic synchronization will create physical names based on logical names, naming convention plus glossary definitions automatically. Using these settings, you can save time during modeling new database structures.

### B: Automatic Synchronization Off, Apply Naming Convention Rules Off

Synchronization of Captions and Names

☐ Automatic Synchronization  
Clear this checkbox to turn off the automatic synchronization. If you change the Caption then, the Name will not change.

☐ Apply Naming Convention Rules during Synchronization  
Clear this checkbox not to apply the rules defined on tab Naming Convention Rules during the synchronization.

If you create a new naming convention and clear both checkboxes, you will turn off automatic synchronization.

**Use:** This is handy if you reverse engineer a physically existing database and start to modify captions (logical names). Names (physical names) will stay preserved.

### C: Automatic Synchronization On, Apply Naming Convention Rules Off

Synchronization of Captions and Names

☒ Automatic Synchronization  
Clear this checkbox to turn off the automatic synchronization. If you change the Caption then, the Name will not change.

☐ Apply Naming Convention Rules during Synchronization  
Clear this checkbox not to apply the rules defined on tab Naming Convention Rules during the synchronization.

Use these settings if you want to synchronize captions with physical names, but don't want to apply naming convention rules during synchronization.

**Example:** *T\_EXEMPLAR* (logical name) => *T\_EXEMPLAR* (physical name).

Prefix will not be automatically added to the physical name because you already defined it in caption (logical name).

**Use:** This can be useful when you need to rewrite the caption (logical name) and want the physical name to change automatically. Or if you are used to defining logical names in accordance with naming convention rules.

#### D: Automatic Synchronization Off, Apply Naming Convention Rules On

Synchronization of Captions and Names

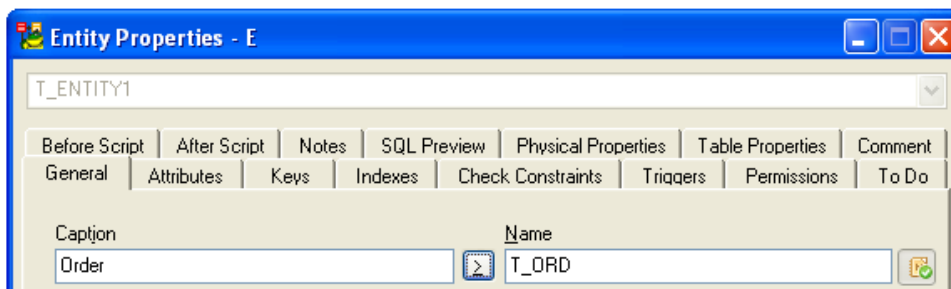
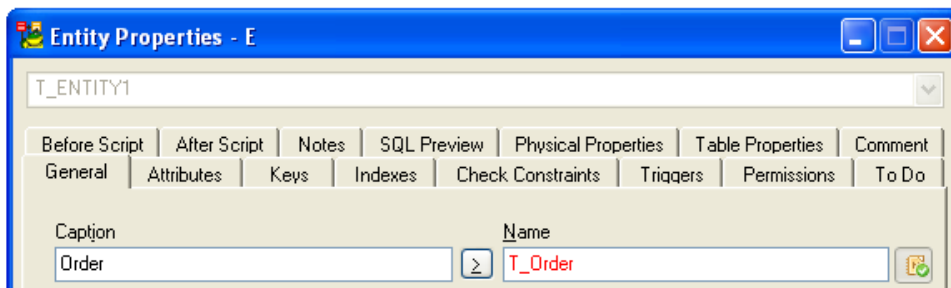
☐ Automatic Synchronization  
Clear this checkbox to turn off the automatic synchronization. If you change the Caption then, the Name will not change.

☒ Apply Naming Convention Rules during Synchronization  
Clear this checkbox not to apply the rules defined on tab Naming Convention Rules during the synchronization.

You turn off the automatic synchronization as you want to synchronize some selected items manually.

**Use:** Use these settings if you are used to defining both captions and names manually – with the advantage to apply the naming convention rules during manual synchronization.

How does it work? - E.g. in the **Entity Properties** form, you define the caption and physical name manually. Later you find out that something is wrong with the physical name (it violates the naming convention rules). -> Clicking the arrow button between the **Caption** and **Name** boxes (manual synchronization) will change the physical name and the naming convention rules will be applied (see the following screenshots).



## Verification of Names

Checkboxes in the **Verification of Names** area affect behavior of the Naming Convention Verification and Synchronization feature.

**Verification of Names**

☒ **Automatic Verification**  
It automatically verifies whether physical names of objects match the defined naming convention rules. Clear this checkbox to turn off the automatic verification.

☒ **Use Glossary for Naming Convention Verification**  
It automatically verifies whether physical names of objects match the definition on tab Glossary.

**Naming Convention Verification & Synchronization**

Model Objects

Logical Name | Physical Name | NC | Sync | Expected Physical Name

Logical Name	Physical Name	NC	Sync	Expected Physical Name
<b>Videorental</b>				
<b>Entities</b>				
<b>Customer</b>				
<b>Attributes</b>				
Customer ID	Customer ID			customer_id
Name	Name			name
Address	Address			address
type id	type id			type_id
Borrowing	TBORROWING_TMP			T_BORROWING
<b>Attributes</b>				
Exemplar ID	Exemplar ID			exemplar_id
Customer ID	Customer ID			customer_id
Start Date	Start Date			start_date
End Date	End Date			end_date
Total Price	Total Price			total_price
VAT	VAT			vat
Exemplar	TEXEMPLAR_TMP			T_EXEMPLAR
<b>Attributes</b>				
Exemplar ID	Exemplar ID			T_EXEMPLAR
Film ID	Film ID			film_id
Medium ID	Medium ID			medium_id
Price per Day	Price per Day			price_per_day

Current prefix "T" violates naming convention definition. Correct prefix is "T\_".  
Current suffix "\_TMP" violates naming convention definition. Correct suffix is "".

**Update** **Close**

## Related Topics

Basic Operations (page 108)

Glossaries, Import of CSV Files (page 130)

## Glossaries, Import of CSV Files

On the **Glossary** tab in the **Naming Convention Properties** dialog you can:

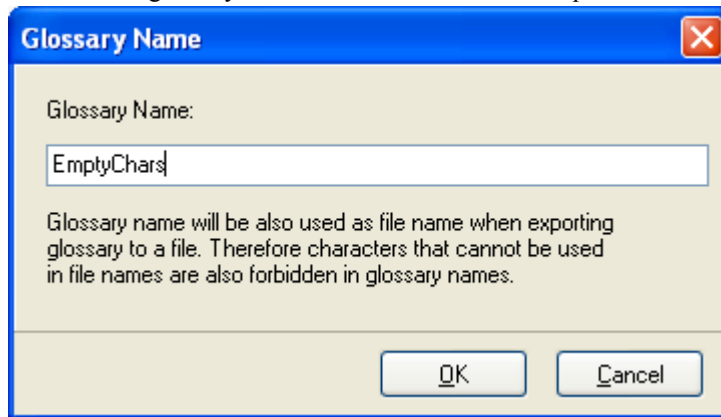
- Define which characters or words you want to replace.
- Import and export CSV files.

See how you can use it on the following example.

## Scenario

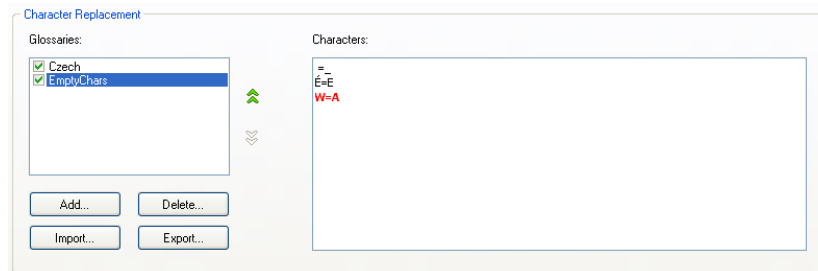
You want to use some abbreviations and automatically shorten particular names of entities and attributes in your newly created model.

1. Edit your existing naming convention.
2. On tab **Glossary | Character/Word Replacement** area, click **Add**.
3. Define the glossary name for character/word replacement.

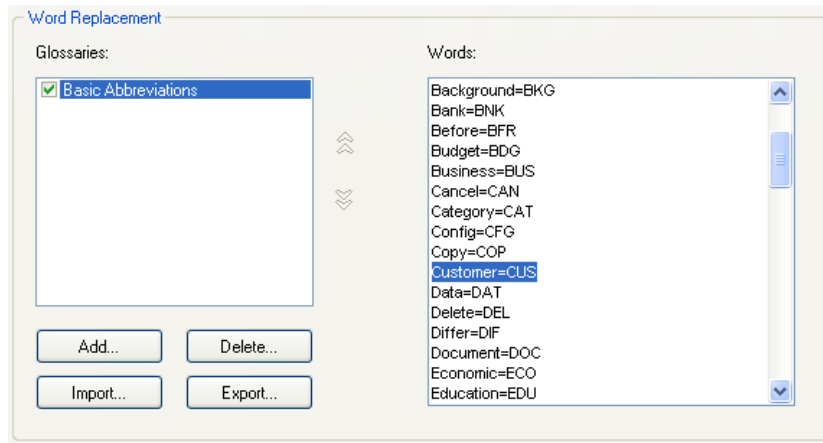


4. Specify the rules for replacement.

In the **Character Replacement** area, the **Characters** window, you can define rules for characters above ASCII 128 and space characters. Other definitions will be displayed in red and will not be taken into consideration.



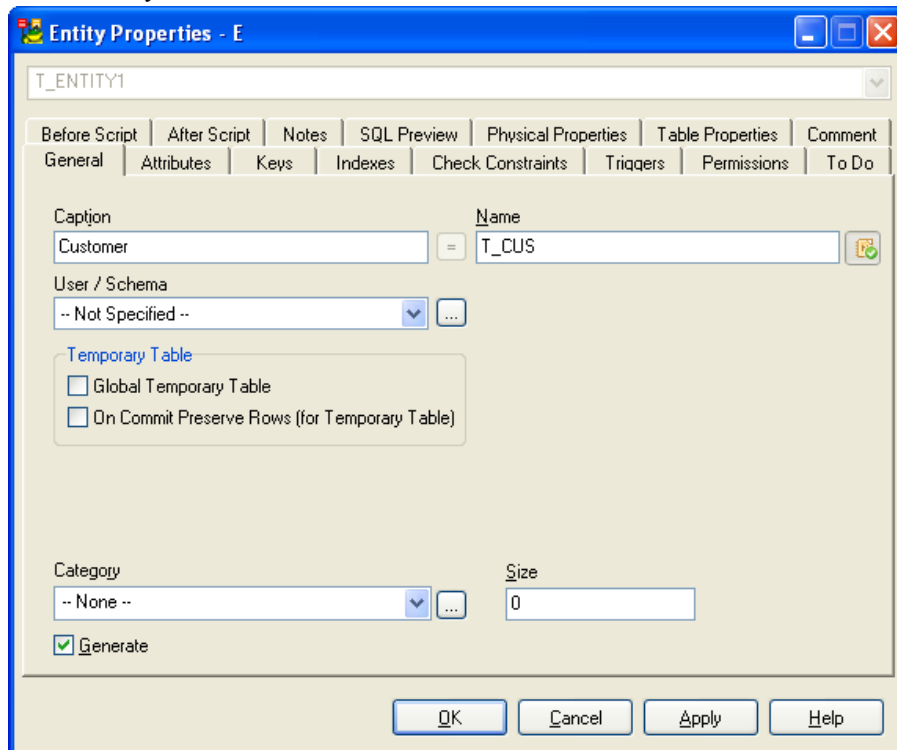
In the **Word Replacement** area, the **Words** window, define the glossary in the following format: *original word=new word*.



5. To import some other abbreviations from .CSV file, click **Import**.

Find e.g. the *Basic Abbreviations.csv* file in the following location: C:\Program Files\Quest Software\Toad Data Modeler 3\Naming Conventions\CSV.

6. In the **Naming Convention Properties** dialog confirm **OK**.
7. Create a new entity in your model and edit it.
8. Write a caption name *Customer*. As you can see, the physical name will automatically shorten to *CUS*.



## Related Topics



About Naming Conventions (page 106)

Basic Operations (page 108)

## Verification

### Model Verification

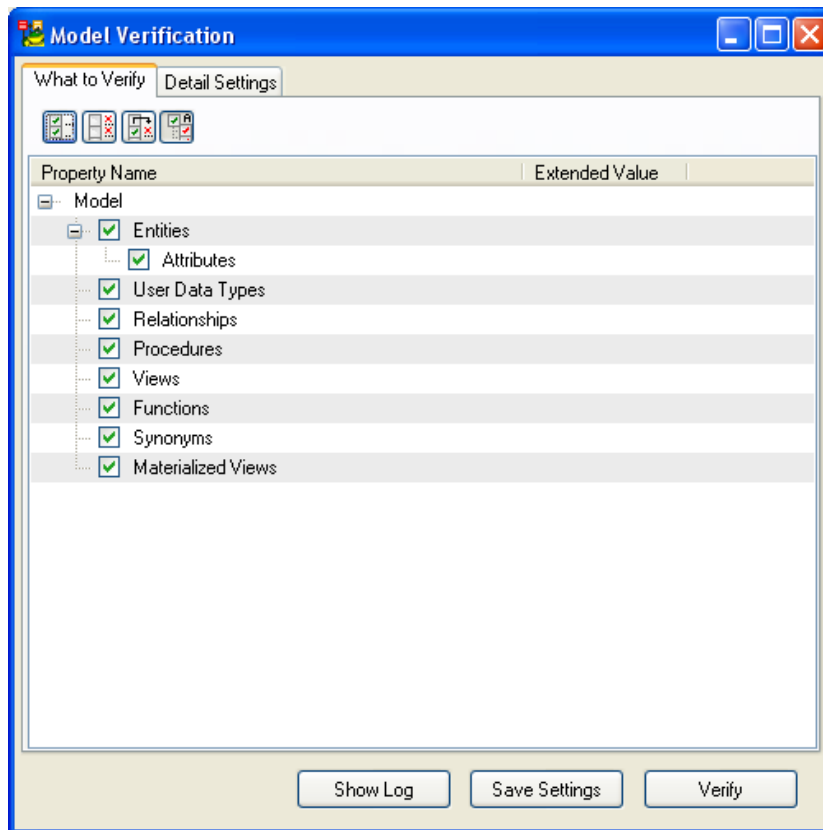
You have just created a model, defined advanced modeling options, and now it's time to check out if your model is correct.

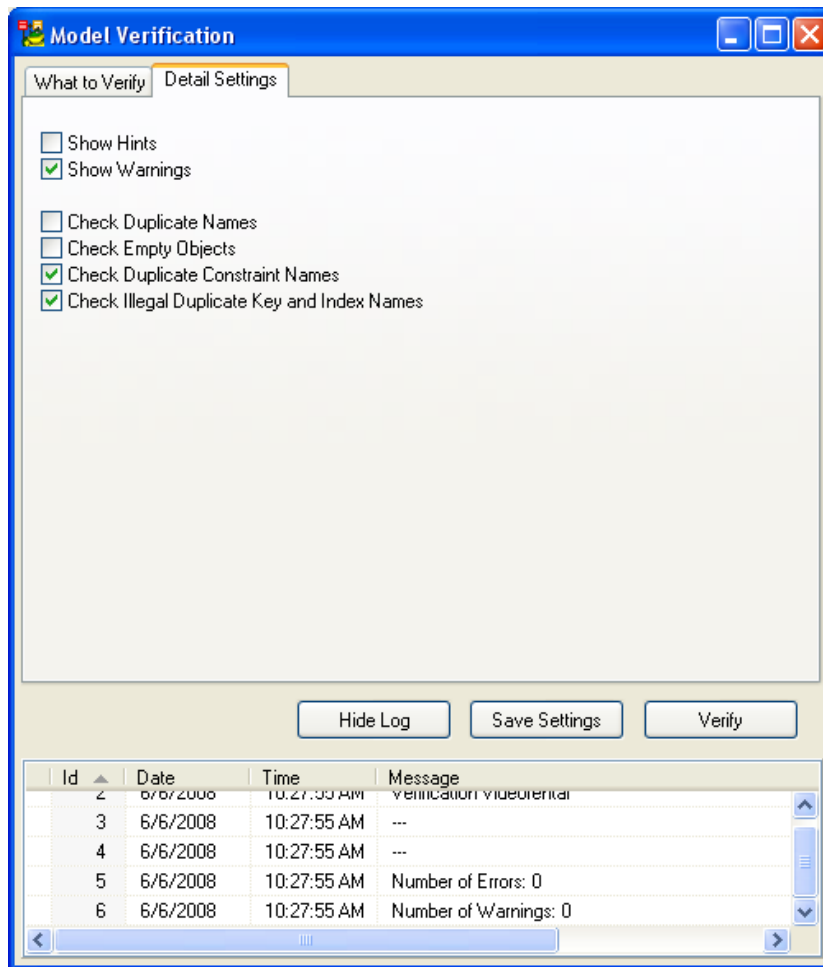
Toad Data Modeler verification returns Errors, Warnings and Hints.

#### Scenario

Verify your model *Videorental* (Oracle 10g).

1. Select **Model | Verify Model** (or press CTRL+F9).
2. On tab **What to Verify**, select Object Types and Properties that you want to verify.  
  
**Note:** More details on Object Types and Properties can be found in the Help file, "OTPs" topic.
3. On tab **Detail Settings**, select options for verification (of objects that you selected on tab **What to Verify**). To save the settings, click **Save Settings**.
4. Click **Verify**. To see the process of verification and final result, click **Show Log**.





Result: As you can see, there are no errors and no warnings in your model.

## Documentation

### Documentation to Physical Model

Toad Data Modeler provides you with a powerful documentation tool. You can:

- Generate reports in HTML, RTF and PDF formats, including Alter reports.
- Create user outputs for your physical models - e.g. HTML, PDF, CSV, text or XML output.
- Generate XSD file for your physical model.

### Related Topics

Report Generation (page 136)

Alter Reports (page 144)

Advanced Users (page 147)

XSD Generation (page 152)

## Report Generation

### Create HTML/RTF/PDF Report

After you create your ER diagram (regardless of whether you create it from scratch or reverse engineer it), you can generate HTML, RTF or PDF reports.

This function is available in the **Report Wizard**.

All the settings you set up for your report generation are saved with the model. So, there is no need to define the settings and select the options for the model all over again.

#### Scenario

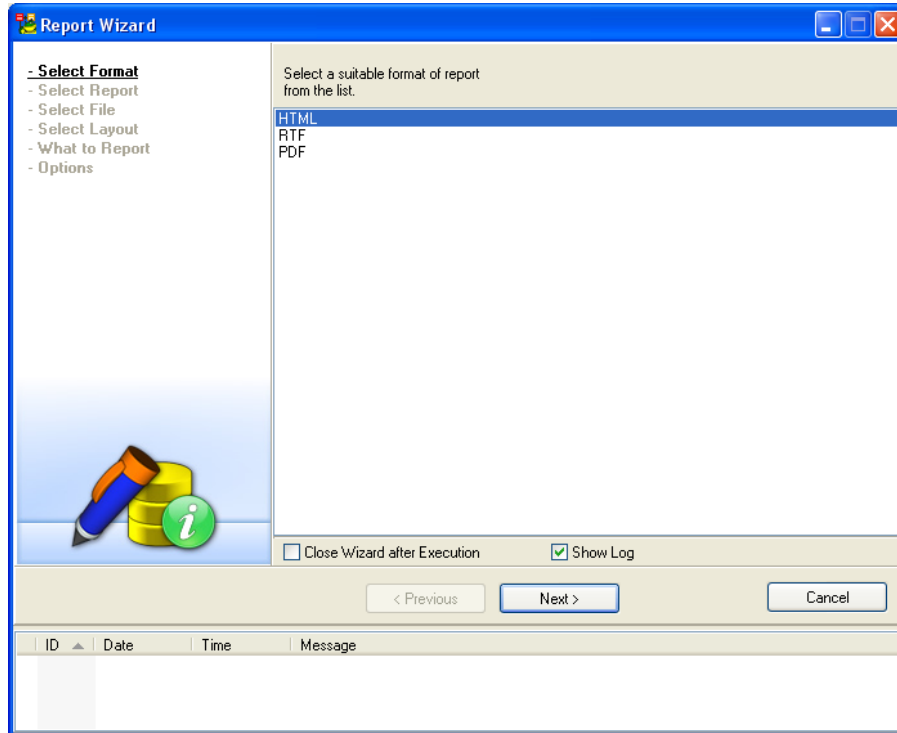
You want to generate HTML report for your *Videorental* Oracle 10g model.

#### *To open the Report Wizard*

1. Click  on the toolbar.

**Scenario:** Select format of report.

2. On the **Select Format** page, select *HTML*.



## Related Topics

Select Report Type (page 137)

## Select Report Type

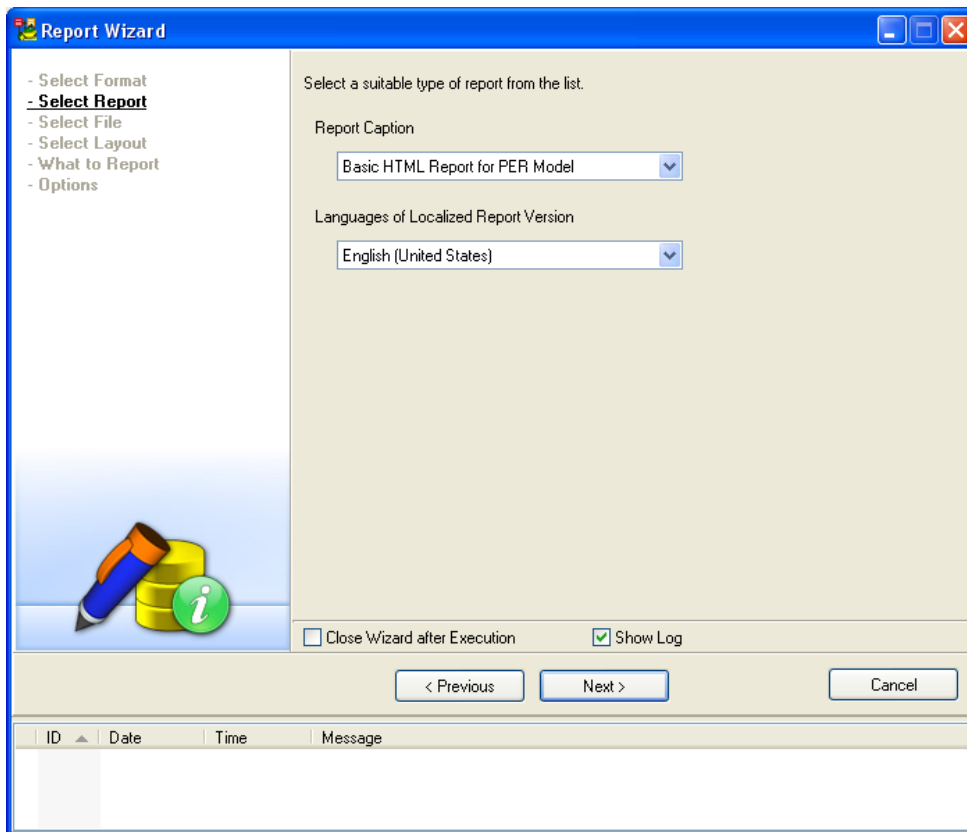
**Scenario:** Select type and language of the report.

1. From the **Report Caption** box, select a report type *Basic HTML Report for PER Model*.

**Note:** Generally, these report types are available: Basic HTML Report for PER Model, Basic HTML Report for LER Model, Basic RTF Report for PER Model, Basic RTF Report for LER Model.

2. From the **Languages** box, select the language in which you want to have the report - *English (United States)*.

**Tip:** You can select from several languages. However, first you need to download appropriate packages from the [Modeling community website](#).



## Related Topics

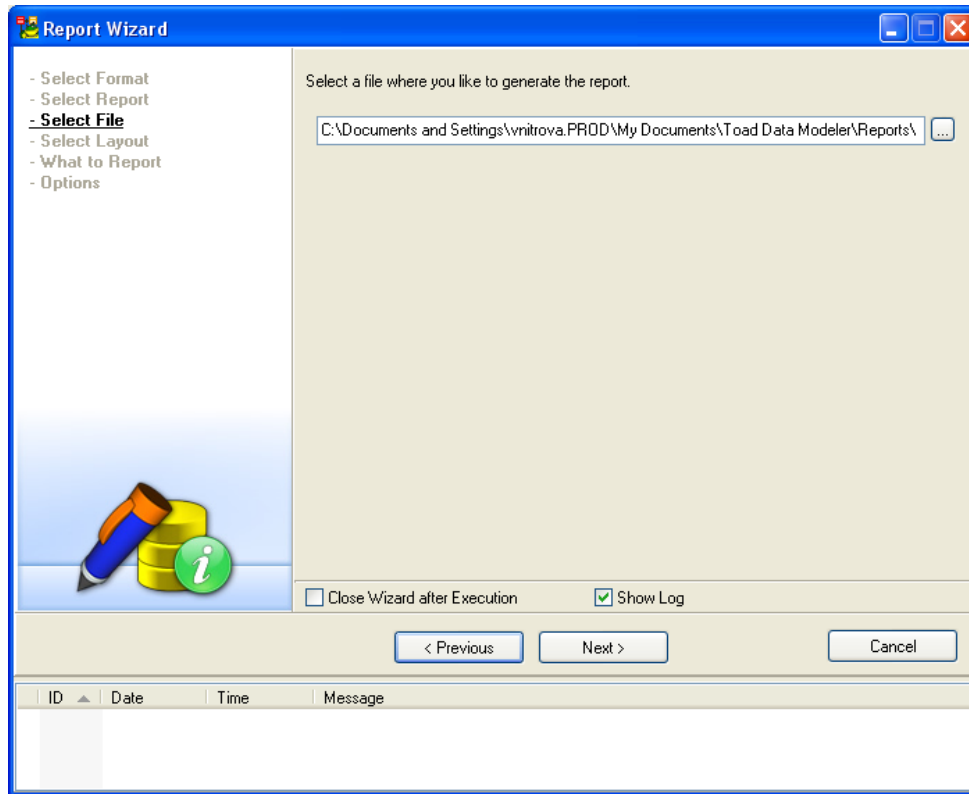
Select File (page 138)

## Select File

**Scenario:** Define a path where you want to generate the report.

You can change the path for particular model - simply click the button on the right. Or if you want to change the default path, feel free to do it in the **Settings** menu | **Options**.

**Note:** By default, the path defined in the **Settings** menu | **Options** | **Application** | **Paths** | **Reports** | **Path to HTML/RTF/PDF Reports** is set up here.



## Related Topics

Select Style (page 139)

## Select Style

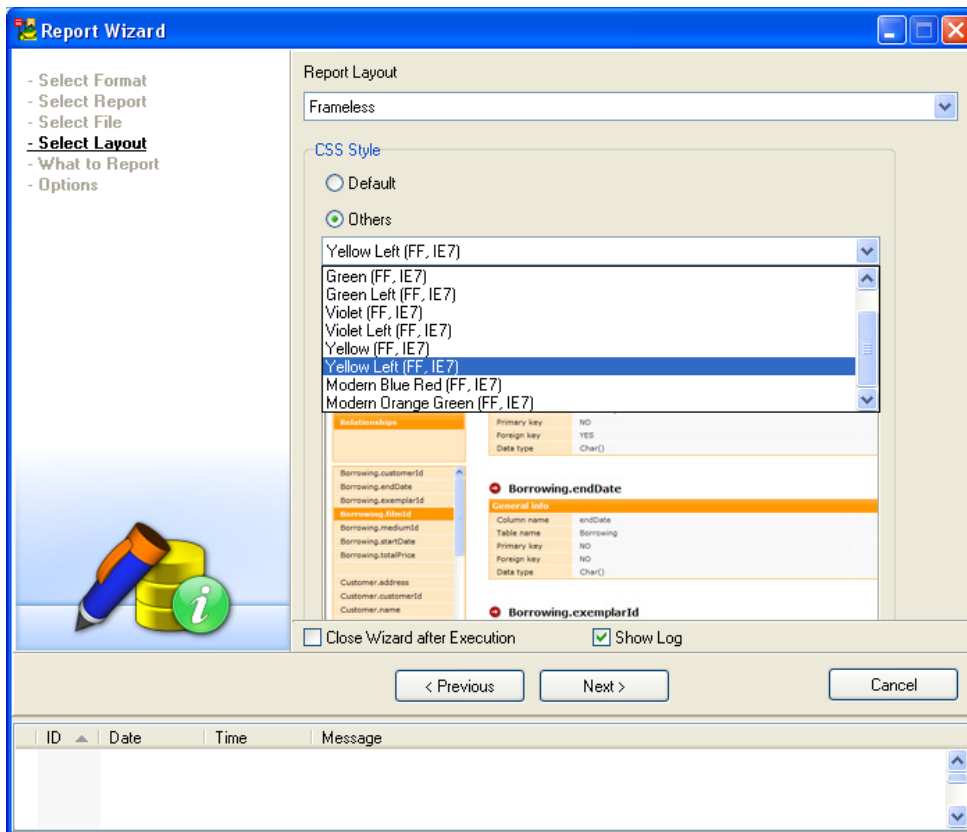
**Scenario:** On the **Select Layout** page, select a frame and style of your HTML report.

1. From the **Report Layout** box, select *Frameless*.

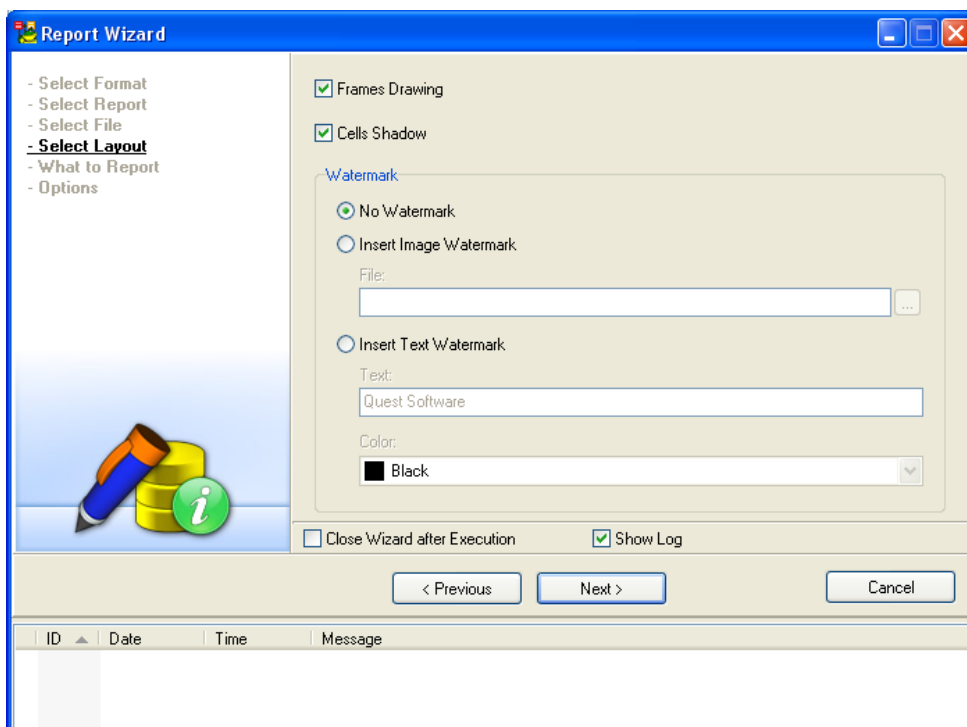
**Note:** For large models, *Frames - top menu* or *Frames - left menu* options are recommended. (Frameless report layout is not recommended as it uses Java script that goes through all objects, which takes too much time if your model is large.)

2. In the **CSS Style** area, select *Others* to be offered all available styles. See how the selected style looks like in the preview box. Select the one you like the most, e.g. *Yellow Left*.

**Tip:** You can generate the same report in different styles.



Options for RTF/PDF report:





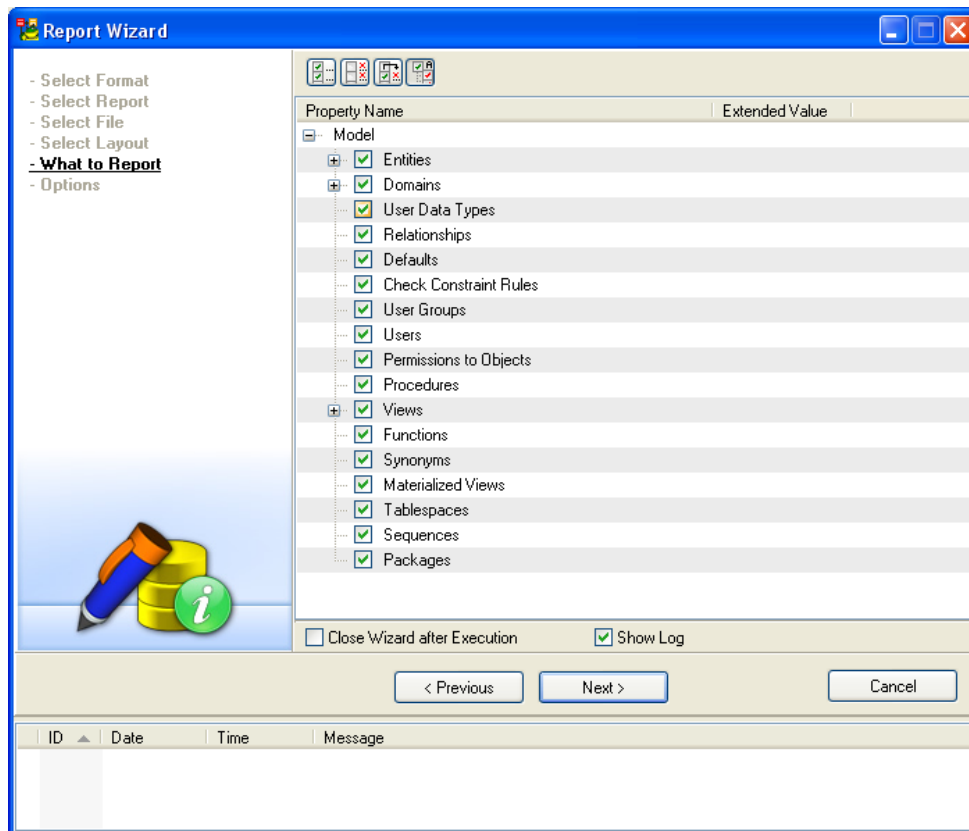
## Related Topics

Select Objects to Report (page 141)

## Select Objects to Report

**Scenario:** Select Object Types and Properties you want to contain in your report.

**Note:** More details on Object Types and Properties can be found in the Help file, "OTPs" topic.



## Related Topics

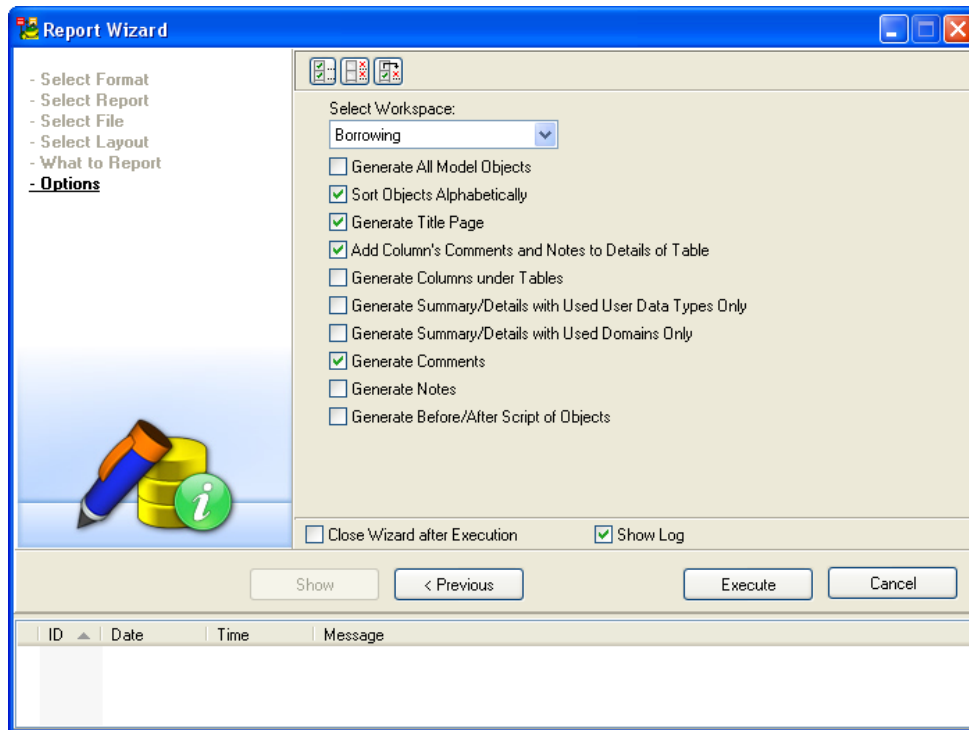
**Select Options Scenario:** Select options for the report generation. The options are specific for each report format and type. To generate a report for particular Workspace, select the Workspace from the Select Workspace box - Borrowing and clear the Generate All Model Objects checkbox.  
**Note:** ERD image of the selected workspace plus only objects that exist on the workspace will be generated in the report (e.g. tables, relationships, views etc.). The report will also contain all model objects that cannot be placed on Workspaces (e.g. User Data Types, Procedures etc.).  
**Other options:** To generate a report for the whole model, including images of all Workspaces of the model, select All Workspaces from the Select Workspace box. (The Generate

All Model Objects checkbox is selected automatically.)To generate a report for the whole model but only an ERD image of particular Workspace, select the Workspace from the Select Workspace box and select the Generate All Model Objects checkbox. Related TopicsGenerate Report (page 1)

## Select Options

**Scenario:** Select options for the report generation.

The options are specific for each report format and type.



- To generate a report for particular Workspace, select the Workspace from the **Select Workspace** box - *Borrowing* and clear the **Generate All Model Objects** checkbox.

**Note:** ERD image of the selected workspace plus only objects that exist on the workspace will be generated in the report (e.g. tables, relationships, views etc.). The report will also contain all model objects that cannot be placed on Workspaces (e.g. User Data Types, Procedures etc.).

Other options:

- To generate a report for the whole model, including images of all Workspaces of the model, select *All Workspaces* from the **Select Workspace** box. (The **Generate All Model Objects** checkbox is selected automatically.)

- To generate a report for the whole model but only an ERD image of particular Workspace, select the Workspace from the **Select Workspace** box and select the **Generate All Model Objects** checkbox.

## Related Topics

Generate Report (page 143)

## Generate Report

**Scenario:** After you define the settings on pages of the Report Wizard, you can generate the report.

1. Click **Execute**. To see the process, select **Show Log**.
2. As soon as the report generation is finished, the following message will display in the Wizard: *'Report was generated.'* Confirm the message **OK**.
3. Click **Show** to see the final report.
4. Click **Cancel** or **x** to close the Report Wizard.

## Tips:

- Leave checkbox **Close Wizard after Execution** cleared if you want to generate another report - with new options, another style etc. Click **Previous** to get back to previous steps to define new report options.
- HTML Reports: After you generate the new report and your previous report has remained open, press **Refresh F5** to update it.
- RTF Reports: If you want to generate another RTF report, remember to close the previous report first!

**Note:** Both HTML and RTF reports are saved in UTF-8 encoding.

Tables

SCOTT.T\_BORROWING

Model Info

ER Diagram

Tables

Columns

Indexes

Keys

Relationships

Domains

User Data Types

Check Constraints

Triggers

Procedures

SCOTT.T\_BORROWING

SCOTT.T\_CUSTOMER

SCOTT.T\_CUSTOMER\_RATING

SCOTT.T\_EXEMPLAR

SCOTT.T\_FILM

SCOTT.T\_GENRE

SCOTT.T\_MEDIUM

SCOTT.T\_BORROWING

Table properties

SchemaSCOTT

Columns

Key	Column name	Domain	Data type	Not null	Unique	Check	Default
PK	exemplar_id		Integer	YES	NO	NO	
FK	customer_id		Integer	YES	NO	NO	
	start_date		Date	NO	NO	NO	sysdate
	end_date		Date	NO	NO	NO	
	total_price		Price_Type	NO	NO	NO	
	VAT	VAT	Number(4,2)	NO	NO	NO	19

Relationships

Constraint (relationship) name	Relationship type	Parent table	Child table	Card.
makes	Non-identifying	SCOTT.T_CUSTOMER	SCOTT.T_BORROWING	1:5
is_related_to	Identifying	SCOTT.T_EXEMPLAR	SCOTT.T_BORROWING	1:1

Keys

PK?	Constraint name	Columns
PK	pk_T_BORROWING	exemplar_id

Check constraints


Check constraint name	check_end_after_start
Check	(end_date>start_date) or (end_date is null)

## Alter Reports

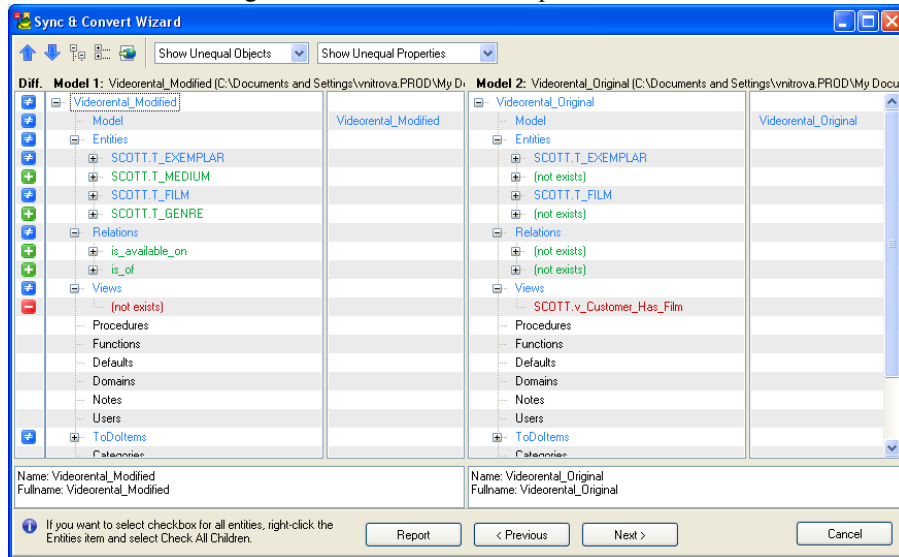
Toad Data Modeler allows you to generate a basic HTML, RTF and PDF alter report with changes between two physical models of the same database system.

### Scenario

You want to compare Oracle 10g models *Videorental\_Original* and *Videorental\_Modified* and generate HTML alter report for the differences.

1. Click  on the toolbar to open the **Sync & Convert Wizard**.
2. On the **Action** page, select **Compare Models and Generate Alter Report**.
3. On the **Select Left Side** page, click **File** and then the button next to the **File** box and find the model *Videorental\_Modified*.
4. On the **Select Right Side** page, click **Another Physical Model File**, click the button and find the model *Videorental\_Original*.
5. On the **Settings** page, select e.g. **Ignore Text Case**.
6. On the **Select Object Types** page, select **Compare All** and leave default selection of OTPs.

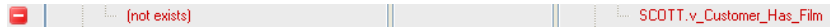
- On the **Select Items** page, you can see differences between the models. The differences will be generated in the alter report.




This icon and the blue color say: "There is a difference."

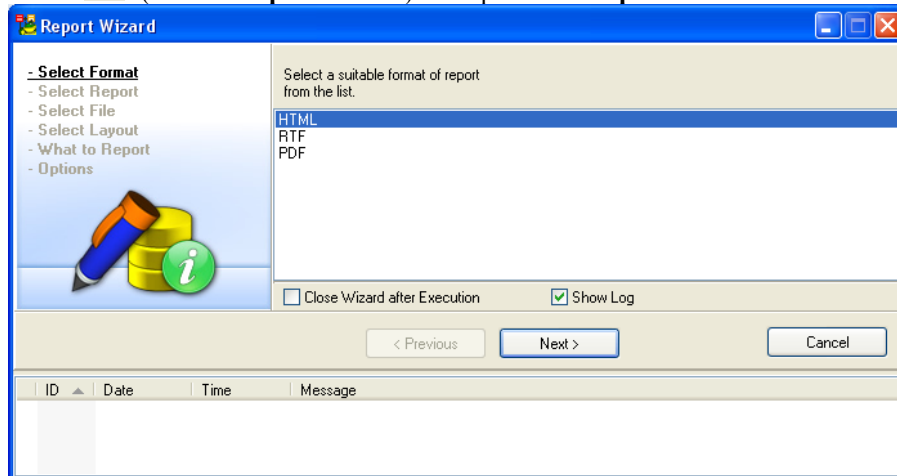


This icon and the green color say: "This item exists in Left model and is missing in Right model."

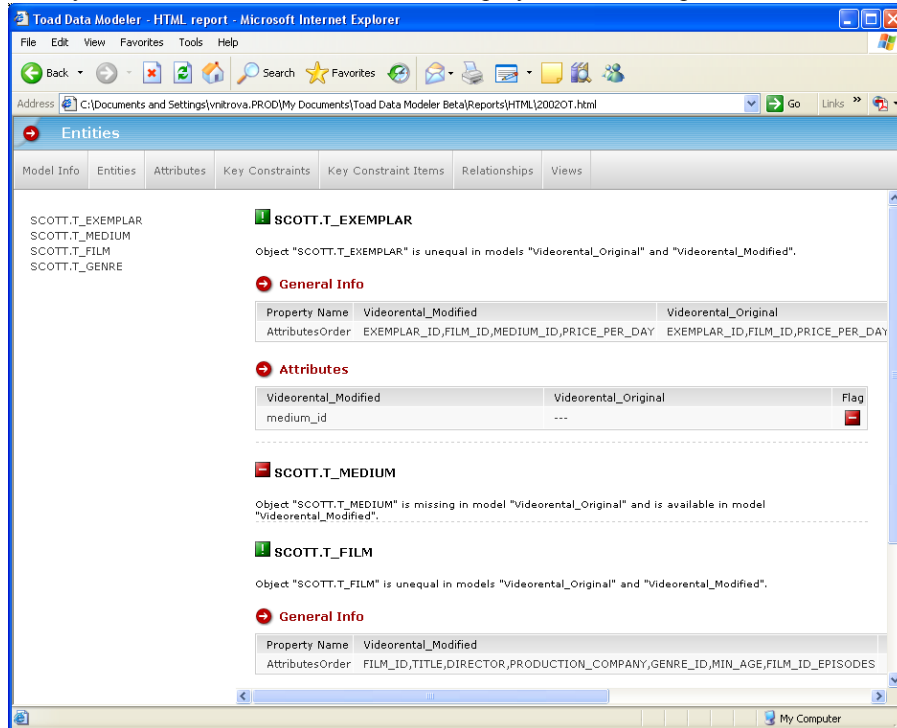


This icon and the red color say: "This item is missing in Left model and exists in Right model."

- Click  (or the **Report** button) to open the **Report Wizard**.



9. Select **HTML** and go through other steps in the **Report Wizard** that are already familiar to you.
10. Finally click **Execute** and **Show** to display the alter report.



11. Close the **Report Wizard** and click **Next** in the **Sync & Convert Wizard**.
12. On page **Review**, see the statistic information.
13. Close the wizard.

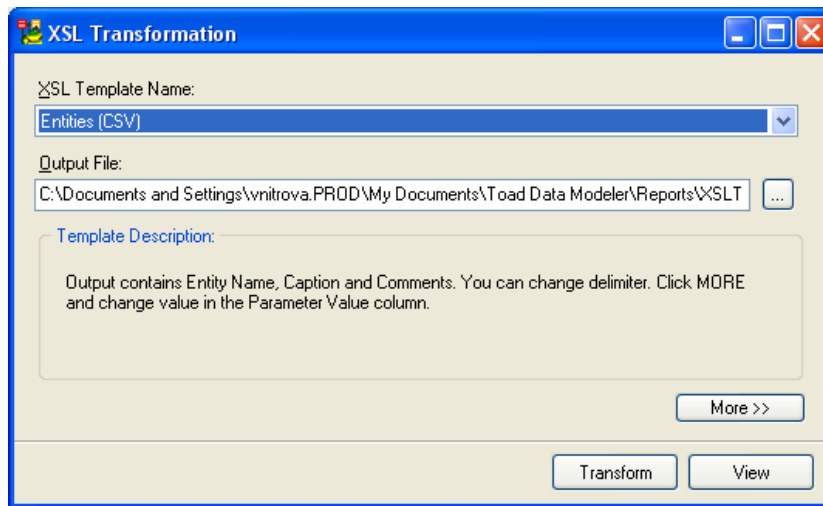
## XSL Transformation

### XSL Transformation

This feature allows you to create user outputs for your physical models very fast. The output can be in any format that supports XSL language - e.g. HTML, PDF, CSV, text or XML.

You select among various XSL templates. Each template generates a different output (HTML, CSV etc.) Advanced users can customize the templates and modify their content to generate an output that will fit their needs.

1. Select **Model | Reports / XSL Transformation**.



- From the **XSL Template Name**, select a template according to the output you want to get. Take notice of how the text in the Template Description area changes. In this example, **Entities (CSV)** template has been selected.

Available templates:

Comments (HTML)
Entities (CSV)
Notes (HTML)
Tablespaces (HTML)
ToDo Report (HTML)

- Check the path in the **Output File** box. This is the path where the output will be stored.
- Click **Transform**.
- Click **View** to see the result.

	A	B	C	D	E	F	G	H	I	J
1	T_CUSTOMER	Customer								
2	T_BORROWING	Borrowing	Borrowed items							
3	T_EXEMPLAR	Exemplar								
4	T_MEDIUM	Medium								
5	T_ORDER_RECORD	Order Record	All records are stored in list of records. It will be possible to book a movie.							
6	T_FILM	Film								
7	T_CUSTOMER_RATING	Customer Rating	Movie ratings (by customers)							
8	T_GENRE	Genre								
9										

### Advanced Users

If you click **More>>**, you can see options for advanced users interested in customization of the templates.

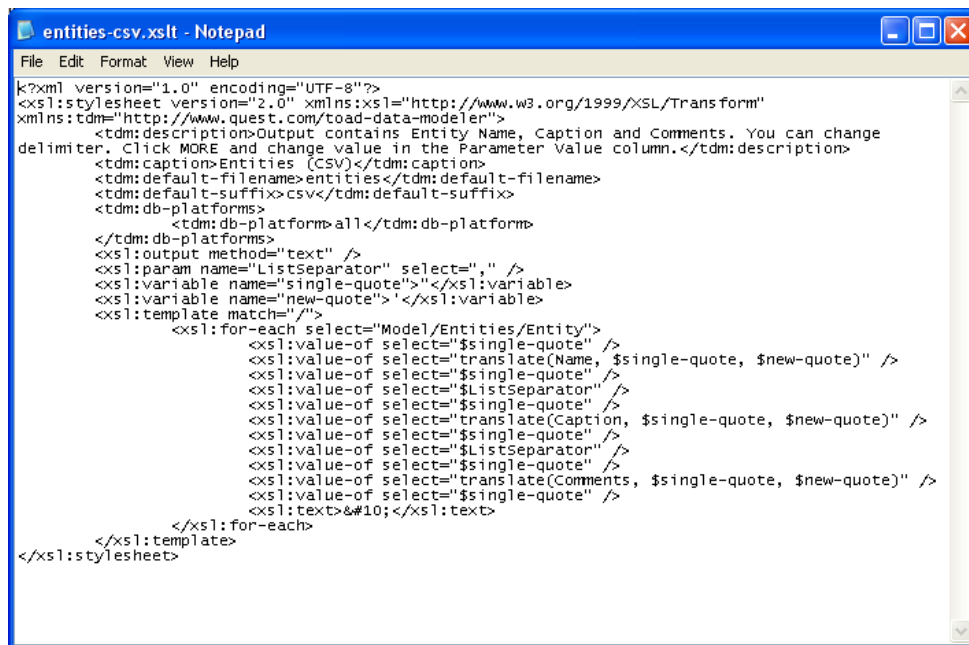
**XSL File Location** - Path where appropriate system template is stored. System templates shouldn't be modified. Via the button on the right, you can add other templates you created and that are missing in the box XSL Template Name.

**Input Values** - In this area, templates with parameters will be displayed. You can edit them directly here.

**Save XSD File As** - Allows you to save the XSD file without the necessity to open it in associated application.

**Show XSD File** - Shows the XSD file. The XSD file describes structure of source XML that is an input for XSL transformation.

Toad Data Modeler allows you to customize the templates. See the example below of how such a template can look like.



## Related Topics

Where to Find the Simplified XML File? (page 149)

## Customize XSLT Templates

XSLT is an XML based language used for transformation of XML source documents into other documents. Output files can be XML, HTML, TXT, CSV, SVG, XSD and so on.

In general, for transformation of XML files it is necessary to specify:

1. Source (XML)
2. Template with instructions on how to convert a source to output (XSLT file)
3. Output file (where to store the output)



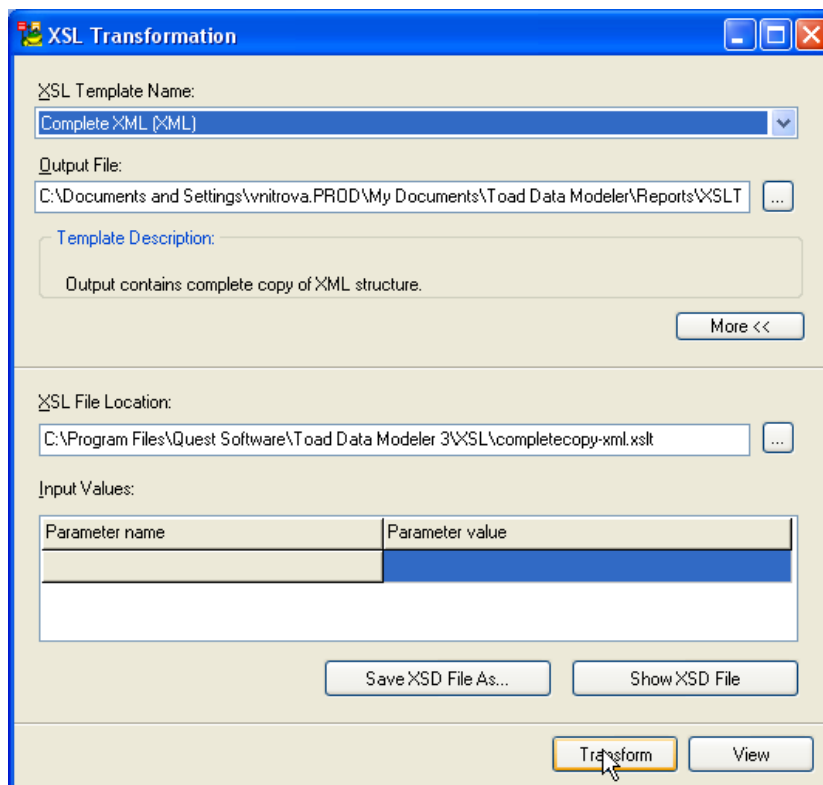
Toad Data Modeler allows you to transform simplified XML structure to the selected output. The simplified XML contains less XML data than standard TXP files (TXP is a standard file suffix for Toad Data Modeler models, however, its structure is also XML.)

#### Where to Find the Simplified XML File?

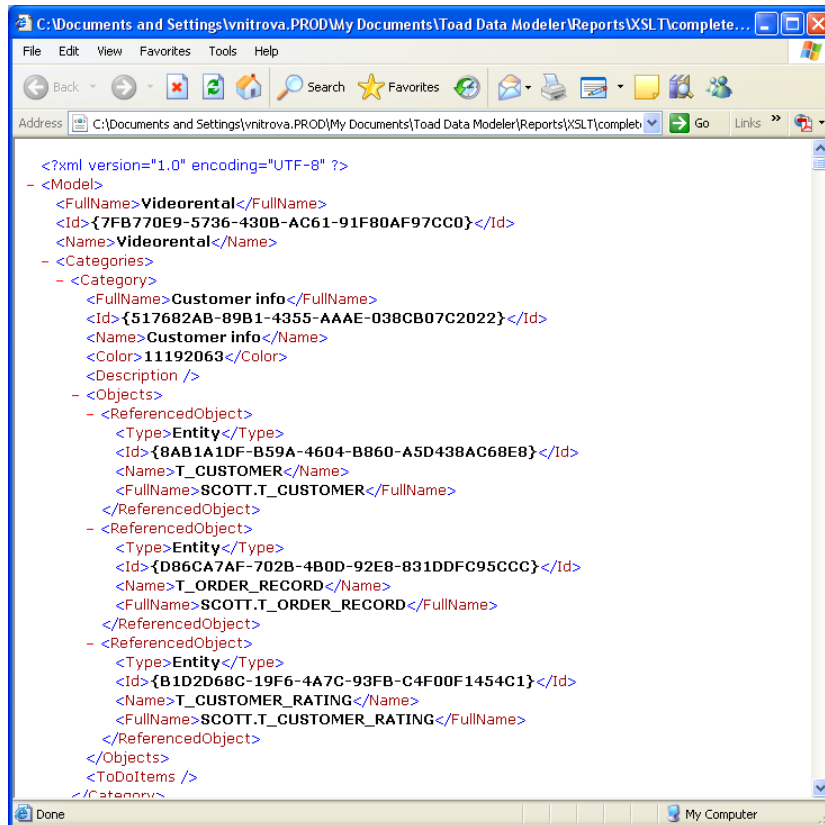
You cannot find it. You must generate it. Every database system contains different items (sequences...), that's why also simplified XML structures for MySQL and Oracle may differ.

#### To generate XML file

1. Click **Model | Reports / XSL Transformation** and from the **XSL Template Name** box select **Complete XML (XML)**.



2. Click **Transform** and **View** to open the generated simplified XML file for your database model. You will see a structure of XML file and a content of your model, including entity names, attribute names, information about relationships, comments etc.



For creation of new XSL templates, it is useful to work with XSD files. XSD files describe structure of XML source file and helps you to understand the XML structure, what nodes may appear in XML and so on. You can click **Show XSD File** or **Save XSD File As** to display or save XSD file for your simplified XML structure.

### Predefined XSL Templates

By default, the following reports or outputs can be generated:

- Comments (HTML)
- Complete XML (XML)
- Entities (CSV)
- Notes (HTML)
- Tablespace (HTML)
- ToDo Report (HTML)

Using these predefined items you can generate report of all ToDo items, create CSV files with information about entity names, captions and descriptions etc.

### Path to XSLT files

There are two folders where XSLT files are stored.

1. System folder: C:\Program Files\Quest Software\Toad Data Modeler 3\XSL
2. User-defined files: C:\Documents and Settings\<user name>\My Documents\Toad Data Modeler\<installation name>\XSL

### Sample XSLT File

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:tdm="http://www.quest.com/toad-data-modeler" xmlns:msxsl="urn:schemas-microsoft-com:xslt">
  <tdm:description>Output contains Entity Name, Caption and Comments. You can change List Separator. Click
  MORE and change value in the Parameter Value column.</tdm:description>
  <tdm:caption>Entities (CSV)</tdm:caption>
  <tdm:default-filename>entities</tdm:default-filename>
  <tdm:default-suffix>csv</tdm:default-suffix>
  <tdm:db-platforms>
    <tdm:db-platform>all</tdm:db-platform>
  </tdm:db-platforms>

  <msxsl:script language="JScript" implements-prefix="tdm">
    function closeInApostrophes( nodelist ) {
      var text = nodelist.nextNode().text;
      return( '""' + text.replace( /\'/g, '""' ) + '""' );
    }
  </msxsl:script>

  <xsl:output method="text" />
  <xsl:param name="ListSeparator" select="," />
  <xsl:template match="/">
    <xsl:for-each select="Model/Entities/Entity">
      <xsl:if test="position() = 1">
        <xsl:text>Entity Name</xsl:text>
        <xsl:value-of select="$ListSeparator" />
        <xsl:text>Entity Caption</xsl:text>
        <xsl:value-of select="$ListSeparator" />
        <xsl:text>Entity Description</xsl:text>
        <xsl:text>#10</xsl:text>
      </xsl:if>
      <xsl:value-of select="tdm:closeInApostrophes(Name)" />
      <xsl:value-of select="$ListSeparator" />
      <xsl:value-of select="tdm:closeInApostrophes(Caption)" />
      <xsl:value-of select="$ListSeparator" />
      <xsl:if test="(Comments)!=''">
        <xsl:value-of select="tdm:closeInApostrophes(Comments)" />
      </xsl:if>
      <xsl:text>#10</xsl:text>
    </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>
```

### Special instructions, parameters

In all XSLT files there must be the following special instructions:

```
<tdm:description>Description</tdm:description>
<tdm:caption>Caption (CSV) - will appear in combo box</tdm:caption>
<tdm:default-filename>output-file-name</tdm:default-filename>
<tdm:default-suffix>html</tdm:default-suffix>
<tdm:db-platforms>
  <tdm:db-platform>all</tdm:db-platform>
</tdm:db-platforms>
```

If you want to create a template only for selected target databases, use the following:

```
<tdm:db-platforms>
  <tdm:db-platform>OR9</tdm:db-platform>
  <tdm:db-platform>OR10</tdm:db-platform>
  <tdm:db-platform>OR11</tdm:db-platform>
</tdm:db-platforms>
```

Parameters specified this way:

```
<xsl:param name="GenerateModelInfo" select="True" />
```

will display in the **Input Values** table of the **XSL Transformation** dialog.

### Edit Existing XSLT Templates

1. Copy the system XSLT template to your user section. Default locations are:

System: C:\Program Files\Quest Software\Toad Data Modeler 3\XSL

User: C:\Documents and Settings\<user name>\My Documents\Toad Data Modeler\<installation name>\XSL

2. Edit the code then.

**Tips:**

- Visit the [Modeling community](#) for other available XSL templates.
- Read tutorials on customization on our [weblog](#).

## XSD Generation

### XSD File Generation

Toad Data Modeler allows you to generate XSD file to your physical models. XSD file describes structure of XML file - of your physical model created in Toad Data Modeler (TXP file). XSD shows how the TXP file looks like, how it is structured etc.

» Select **Tools | Generate XSD File**.

**Note:** This option is available only if Expert Mode is enabled.

By default, the XSD file is generated to:

C:\Documents and Settings\user\My Documents\Toad Data Modeler\Reports

**Tip:** Use a tool that can visualize the XSD.

**Example:** XSD file opened in Notepad

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema attributeFormDefault="unqualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema"><xs:element
name="PERModelOR10"><xs:complexType><xs:sequence><xs:element name="Id"
type="GUID"/><xs:element name="Name" type="xs:string"/><xs:element name="Ordinal"
type="xs:integer"/><xs:element name="HistoryID" type="xs:string"/><xs:element
name="GlobalOrder" type="xs:integer"/><xs:element name="ModelDefID"
type="xs:string"/><xs:element name="LogicalNames" type="xs:boolean"/><xs:element
name="WorkSpaces"><xs:complexType><xs:sequence><xs:element name="PERWorkSpaceOR10"
minOccurs="0" maxOccurs="unbounded"><xs:complexType><xs:sequence><xs:element name="Id"
type="GUID"/><xs:element name="Name" type="xs:string"/><xs:element name="Ordinal"
type="xs:integer"/><xs:element name="HistoryID" type="xs:string"/><xs:element
name="GlobalOrder" type="xs:integer"/><xs:element name="PenWidth"
type="xs:integer"/><xs:element name="PenColor" type="xs:integer"/><xs:element
name="BrushColor" type="xs:integer"/><xs:element name="FontCharset"
type="xs:integer"/><xs:element name="FontColor" type="xs:integer"/><xs:element
name="FontName" type="xs:string"/><xs:element name="FontStyle"
type="xs:integer"/><xs:element name="BackgroundColor" type="xs:integer"/><xs:element
name="RecalculateSize" type="xs:boolean"/><xs:element name="FullBackground"
type="xs:boolean"/><xs:element name="Shadow" type="xs:boolean"/><xs:element
name="DisplayLineNames" type="xs:boolean"/><xs:element name="FontHeight"
type="xs:integer"/><xs:element name="RightAngledLines" type="xs:boolean"/><xs:element
name="Grid" type="xs:boolean"/><xs:element name="GridVisible" type="xs:boolean"/><xs:element
name="GridSize" type="xs:integer"/><xs:element name="Description"
type="xs:string"/><xs:element name="ShapeList"><xs:complexType><xs:sequence><xs:element
name="WorkSpaceShapePERViewOR10" minOccurs="0"
maxOccurs="unbounded"><xs:complexType><xs:sequence><xs:element name="Id"
type="GUID"/><xs:element name="Name" type="xs:string"/><xs:element name="Ordinal"
type="xs:integer"/><xs:element name="HistoryID" type="xs:string"/><xs:element
name="GlobalOrder" type="xs:integer"/><xs:element name="ParentBaseID"
type="xs:string"/><xs:element name="PenWidth" type="xs:integer"/><xs:element name="PenColor"
type="xs:integer"/><xs:element name="BrushColor" type="xs:integer"/><xs:element
name="FontCharset" type="xs:integer"/><xs:element name="FontColor"
type="xs:integer"/><xs:element name="FontName" type="xs:string"/><xs:element
name="FontStyle" type="xs:integer"/><xs:element name="FormatLocked"
type="xs:boolean"/><xs:element name="FontHeight" type="xs:integer"/><xs:element name="Top"
type="xs:integer"/><xs:element name="Left" type="xs:integer"/><xs:element name="Z"
type="xs:integer"/><xs:element name="Width" type="xs:integer"/><xs:element name="Height"
type="xs:integer"/><xs:element name="dz" type="xs:integer"/><xs:element
name="RecalculateSizes" type="xs:boolean"/><xs:element name="UseWorkspaceRecalculateSizes"
type="xs:boolean"/><xs:element name="Shadow" type="xs:boolean"/><xs:element
name="FullBackground" type="xs:boolean"/><xs:element name="LineList"
type="referencedObjects"><xs:annotation><xs:documentation>List of referenced Line
List</xs:documentation></xs:annotation></xs:element></xs:sequence><xs:attribute
name="ObjectType" type="xs:integer" use="required" fixed="2503"/><xs:attribute
name="CSAName" type="xs:string" use="required"

```

## Script Generation

### Order of Generated Objects

Your ER diagram is done, verified, possibly you also generated a report for it. Now let's generate final DDL/SQL script for it.

Before you start to select options for the script generation, you can change order of objects for the script generation process. (For example, you prefer to generate Users before User Permissions to a table etc.)

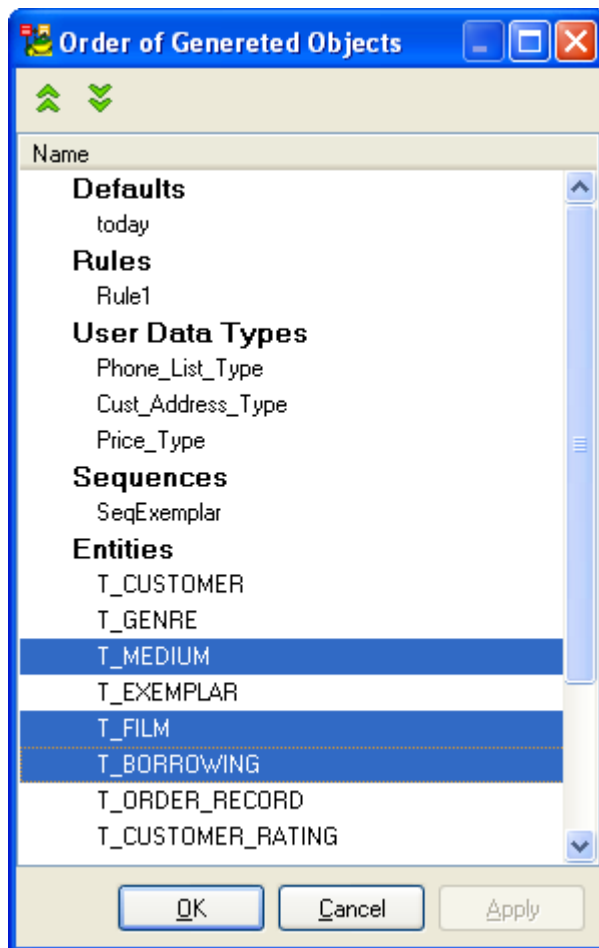
In Toad Data Modeler, you can change order of the following objects:

- Domains
- Entities
- Views
- Dictionary types
- Stored procedures
- Functions

- Users
- User data types

*To change the order*

» Select **Model | Order of Generated Objects**.



Objects placed above the selected item will be generated before the item. (E.g.: The *Genre* entity will be generated before the *Medium* entity, and *Order Record* entity after the *Borrowing* entity.)

Feel free to change order of an object, selection of objects or even whole groups of objects (e.g. Procedures and Views).

**Tips:**

- To select an object, click the selected object.
- To select a group, click the group name.
- To make multiple selection of objects, use SHIFT or CTRL.
- You can also use the pop-up menu or shortcuts (CTRL + Up, CTRL + Down).

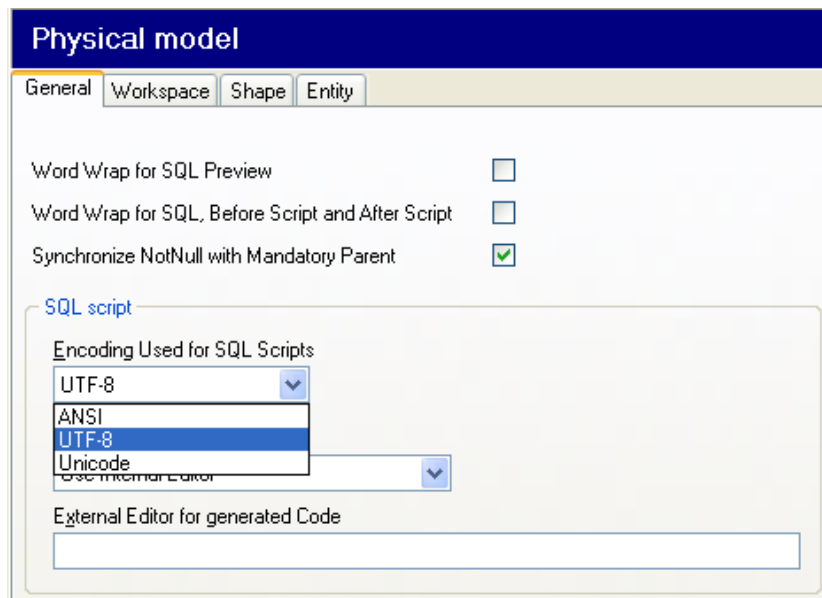
## Related Topics

Select Encoding (page 155)

## Select Encoding

*To set up encoding for the generated script*

1. Select **Settings** | **Options** | **Physical Model** | **General** tab | **SQL Script** area.
2. From the **Encoding Used for SQL Scripts** box, select the encoding you prefer, e.g. *UTF-8*.



**Note:** Here, you can also define External Editor for the generated SQL script or alter script. External editor enables you to **open** the file with generated SQL, it **doesn't allow you to edit** all SQL codes. Via this feature, you can view the generated SQL code in the program you prefer, and possibly execute the script from there.

See "Integration Options " (page 247) for more information.

## Related Topics

Select Items (page 155)

## Select Items


Now you are ready to start the SQL/DDDL Script Generation.

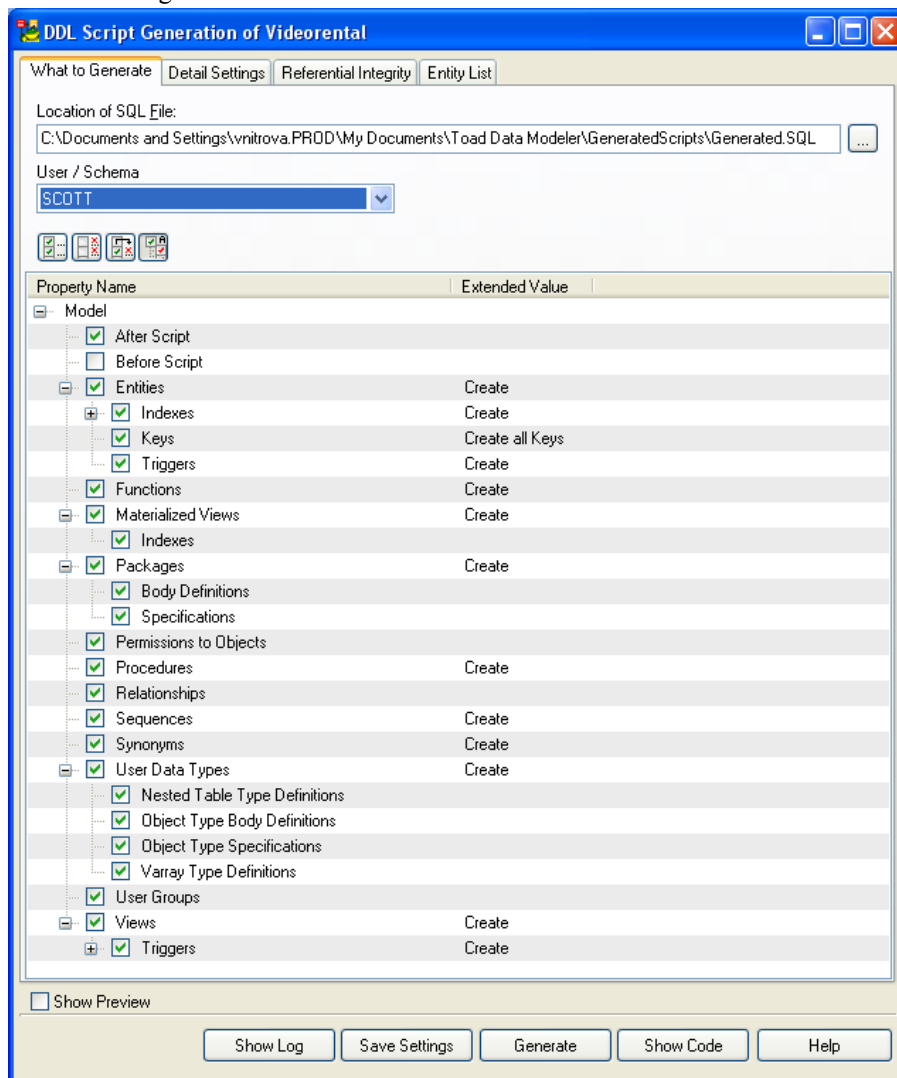
Toad Data Modeler respects as many database specifics as possible and therefore you can model almost any possible situation and generate very detailed and database specific SQL/DDL code.

## Scenario

Generate DDL script for your Oracle 10g *Videorental* model.

**Scenario:** Define options on tab **What to Generate**.

1. Click  on the toolbar (also **F9**).
2. Define settings on tab **What to Generate**.



3. In the **Location of SQL File** box, set up the path where the final DDL script should be saved. By default, the path defined in the **Settings** menu | **Options** | **Paths** | **File with Generated Script** is filled in automatically. However, you can define a different path for



each model of yours. To change the path for all your models, we recommend to change the default path in the **Settings** menu | **Options**.

**Tip:** To save the changed location for particular model (and also other settings you define in this dialog), you need to click **Save Settings** at the bottom of the dialog. The new path (including other settings for script generation) will be saved within the model. Next time you open the model, you will not have to set up the settings again.

4. From the **User/Schema** box, select a user/schema for which you want to generate the script. The schema selection here influences selection of entities on tab **Entity List**. If you select the item *--Not Specified--*, SQL script for all users/schemas will be generated.
5. Make selection of Object Types and Properties for the script generation. Feel free to use the **Select All**, **Deselect All** or **Invert Selection** buttons. The **Auto Check** button is enabled by default, which means that all sub-items of the selected main item will be automatically selected too. If a sub-item is selected, the main item will automatically be selected too.
6. Click the **Extended Value** column at a particular item to select a command - *Create, Drop, Drop and Create, Create or Replace* etc.

**Tips:**

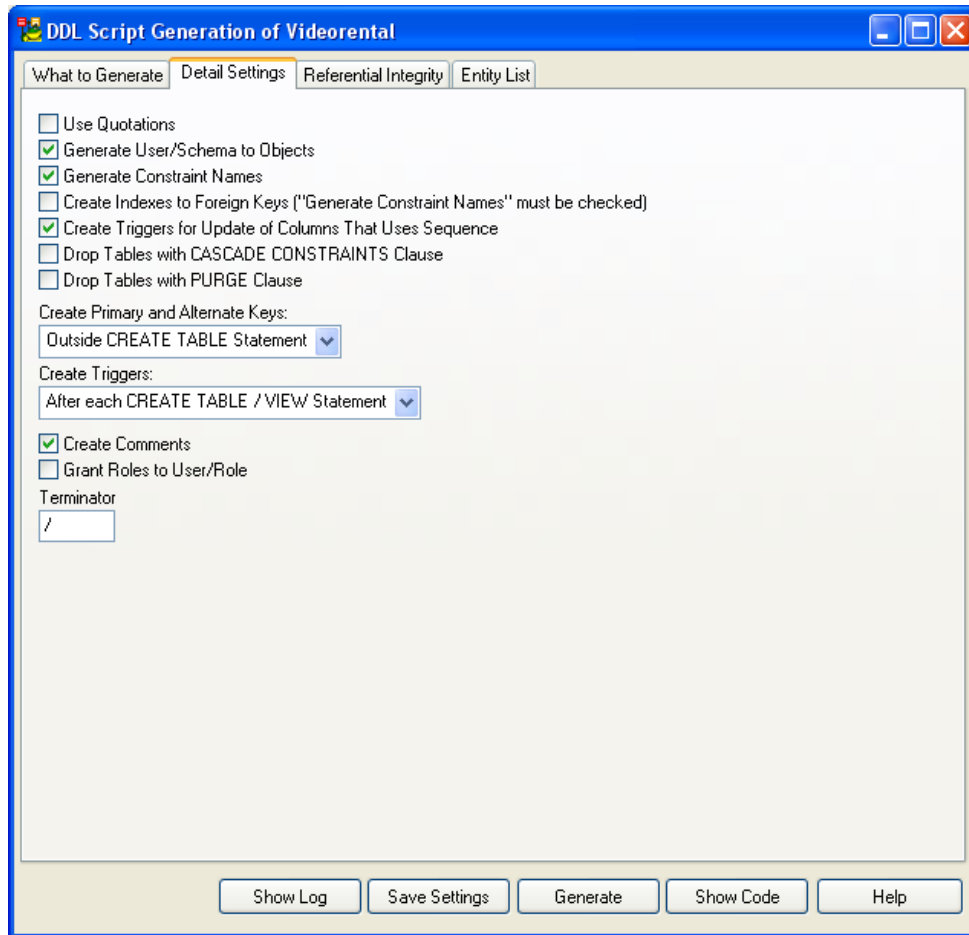
1. For Experts only:
  - a. You can change the default selected Object Types and Properties. See more details in the Help file, "OTPs" topic.
  - b. You can change the default values of the **Script Generation** dialog. See "Set Default Values" (page 295) for more information.
2. You can disable script generation for particular objects - see their **Properties** dialog and clear the **Generate** checkbox (e.g. in the **Entity Properties** form).

**Related Topics**

Detailed Settings (page 157)

## Detailed Settings

**Scenario:** On the **Detail Settings** tab, define other database dependent options.



**Note:** To see the details for the database you use, have a look at the Help file, "Databases" chapter, select your database and see its "Script Generating" topic.

## Related Topics

Referential Integrity (page 158)

## Referential Integrity

**Scenario:** On the **Referential Integrity** tab, define a method for script generation.

You can generate referential integrity for Parent and Child tables declaratively or via triggers (database dependent feature).

The image shows a screenshot of the 'DDL Script Generation of Videorental' dialog box, specifically the 'Referential Integrity' tab. The dialog has four tabs: 'What to Generate', 'Detail Settings', 'Referential Integrity' (selected), and 'Entity List'. The 'Referential Integrity' tab is divided into three sections: 'Parent Entity', 'Child Entity', and 'Generate Referential Integrity Only if'. The 'Parent Entity' section has a 'Restrict' dropdown and two columns of 'Update' and 'Delete' dropdowns, all set to 'Declaration'. The 'Child Entity' section has a 'Restrict' dropdown and two columns of 'Insert' and 'Update' dropdowns, all set to 'Declaration'. The 'Generate Referential Integrity Only if' section has three checkboxes: 'Child Table is Generated' (checked), 'Parent Table is Generated' (unchecked), and 'Child and Parent Table are Generated' (unchecked). At the bottom, there are five buttons: 'Show Log', 'Save Settings', 'Generate', 'Show Code', and 'Help'.

Parent Entity		
Restrict	Update Declaration	Delete Declaration
Cascade	Declaration	Declaration
Set Null	Declaration	Declaration
Set Default	Declaration	Declaration

Child Entity		
Restrict	Insert Declaration	Update Declaration

Generate Referential Integrity Only if

- ☒ Child Table is Generated
- ☐ Parent Table is Generated
- ☐ Child and Parent Table are Generated

Buttons: Show Log, Save Settings, Generate, Show Code, Help

## Related Topics

Select Entities/Workspaces and Save Settings (page 159)

## Select Entities/Workspaces and Save Settings

**Scenario:** On tab **Entity List**, you select if you want to generate SQL/DDL script for entire model or particular workspace or entity.

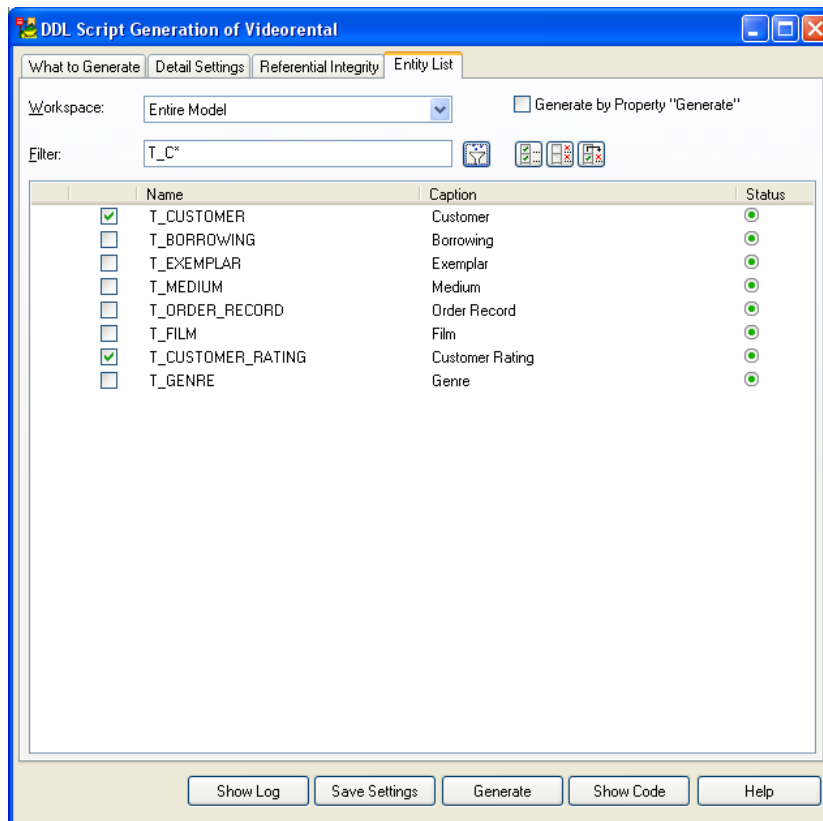
1. From the Workspace box, select *Entire Model* to generate DDL script for all Workspaces of the model.
2. Clear the **Generate by Property "Generate"** checkbox (it's selected by default).

Note: If you leave it selected, all items where the **Generate** checkbox is enabled in (see e.g. the Entity Properties form) will be selected for the script generation automatically. If

you clear it, you can make selection of individual entities that you want to generate, and enable the Filter.

3. In the **Filter** box, type *T\_C\** and click the filter icon to select all entities that fit your criteria.
4. Save all the script generation settings you have just defined in the dialog for particular model (selected options including the script location). - Click **Save Settings**.

Remember to save also your model to preserve the settings for next script generation!



## Related Topics

[Preview SQL Script \(page 160\)](#)

## Preview SQL Script

Toad Data Modeler allows you to preview the DDL script, check it out and decide whether you want to overwrite the existing .sql file or not.

See the **Show Preview** checkbox on the bottom left side of the dialog. It is clear by default. See below what happens if it is clear or selected after you press **Generate**.

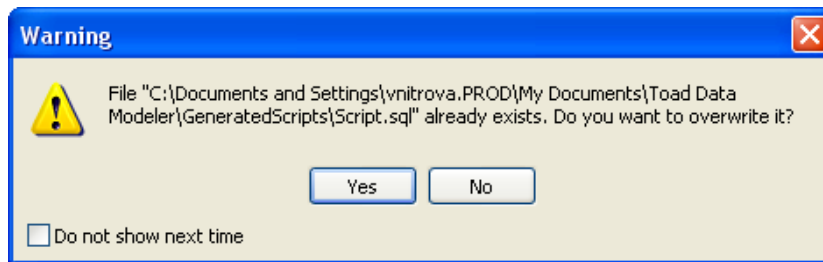
#### *To generate DDL/SQL script*

- » Click **Generate**.

**Tip:** To see the generation process, click **Show Log**.

### **A: The Show Preview Checkbox Is Clear**

1. To prevent you from overwriting the existing .sql file, this confirmation message will display:



2. Select **Yes**.

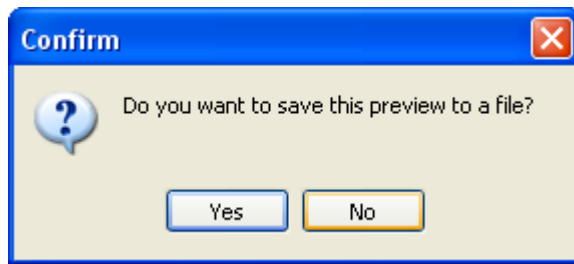
**Note:** If you selected **No**, the process would be interrupted and the .sql file would not be overwritten.

**Tip:** Select **Do not show next time** if you do not want to display the dialog next time.

3. Finally, in the log (or Message Explorer if it's opened) the following message will display: *"SQL Generation has been finished successfully."*
4. Click **Show Code** to open the generated script in the SQL File Viewer dialog.
5. Click **x** to close the dialogs.

### **B: The Show Preview Checkbox Is Selected**

1. The preview will be displayed in SQL File Viewer. Feel free to go through it. As soon as you close the preview, the following message will display:



2. Select **Yes** to overwrite the existing .sql file with the SQL script shown in the preview.

**Note:** **No** interrupts the process. The following message will display in the Message Explorer: *"SQL Generation has not been finished successfully. File cannot be saved."* You can make other modifications and generate another preview. Existing .sql file remains unchanged.

3. Click **Show Code** to open the generated script in the SQL File Viewer dialog.
4. Click **x** to close the dialogs.

### Related Topics

Example of Generated Script (page 162)

## Example of Generated Script

*Videorental* model - Oracle 10g database

**Tip:** Right-click the SQL File Viewer to see options Search, Find and Replace.

/\*

Created: 3/15/2007

Modified: 2/5/2008

Project: Videorental Project

Model: Videorental

Company: Quest Software, Inc.

Author: Radim Mario Tkacik

Database: Oracle 10g

\*/

-- Create user data types section -----

CREATE TYPE SCOTT.Price\_Type

```
AS OBJECT (price NUMBER(10,2),
  MEMBER FUNCTION total_price_VAT (vat number) RETURN NUMBER)
/
CREATE TYPE BODY SCOTT.Price_Type
IS
  MEMBER FUNCTION total_price_VAT (vat NUMBER)
  RETURN NUMBER IS
  BEGIN
    RETURN (price*((vat+100)/100));
  END;
END;
/
CREATE TYPE SCOTT.Cust_Address_Type
AS OBJECT
(
  street_address  VARCHAR2(40)
  , postal_code    VARCHAR2(10)
  , city           VARCHAR2(30)
  , state_province VARCHAR2(10)
  , country_id     CHAR(2)
  , phone         Phone_List_Type
)
/
CREATE TYPE SCOTT.Phone_List_Type
AS VARRAY(5) OF VARCHAR2(25)
/
-- Create sequences section -----
CREATE SEQUENCE SCOTT.SeqExemplar
INCREMENT BY 1
START WITH 1
```

```
NOMAXVALUE
NOMINVALUE
NOCACHE
/
-- Create tables section -----
-- Table SCOTT.T_CUSTOMER
CREATE TABLE SCOTT.T_CUSTOMER(
    customer_id Integer NOT NULL,
    name Varchar2(20 CHAR) NOT NULL,
    address Cust_Address_Type NOT NULL
)
/
-- Create indexes for table SCOTT.T_CUSTOMER
CREATE INDEX i_name ON SCOTT.T_CUSTOMER (name)
/
-- Add keys for table SCOTT.T_CUSTOMER
ALTER TABLE SCOTT.T_CUSTOMER ADD CONSTRAINT pk_T_CUSTOMER PRIMARY
KEY (customer_id)
/
-- Table SCOTT.T_CUSTOMER_RATING
CREATE TABLE SCOTT.T_CUSTOMER_RATING(
    title Varchar2(20 CHAR) NOT NULL,
    director Varchar2(20 CHAR) NOT NULL,
    rating Integer DEFAULT 3
    CONSTRAINT check_rating CHECK (rating > 0)
)
/
-- Add keys for table SCOTT.T_CUSTOMER_RATING
```



```
ALTER TABLE SCOTT.T_CUSTOMER_RATING ADD CONSTRAINT pk_T_CUSTOMER_
RATING PRIMARY KEY (title,director)
```

```
/
```

```
-- Table & Columns comments section
```

```
COMMENT ON TABLE SCOTT.T_CUSTOMER_RATING IS 'Movie ratings (by customers)'
```

```
/
```

```
-- Create procedures section -----
```

```
CREATE PROCEDURE SCOTT.p_Customer_Has_Num_Film(cid IN integer, num OUT integer)
```

```
AS
```

```
BEGIN
```

```
    SELECT count(*)
```

```
    INTO num
```

```
    FROM T_CUSTOMER c, T_BORROWING b, T_EXEMPLAR e, T_FILM f
```

```
    WHERE c.customer_id=cid
```

```
        and c.customer_id=b.customer_id
```

```
        and b.exemplar_id=e.exemplar_id
```

```
        and e.film_id=f.film_id;
```

```
END;
```

```
/
```

```
-- Create functions section -----
```

```
CREATE FUNCTION SCOTT.f_Customer_Has_Num_Film(cid IN integer)
```

```
RETURN integer
```

```
IS
```

```
sol integer;
```

```
BEGIN
```

```
    p_Customer_Has_Num_Film(cid,sol);
```

```
    RETURN (sol);
```

```
END;
```

```
/

-- Create views section -----

CREATE VIEW SCOTT.v_Customer_Has_Film AS

    SELECT distinct c.name, c.address.city as city, f.title, f.director

    FROM T_CUSTOMER c, T_BORROWING b, T_EXEMPLAR e, T_FILM f

    WHERE c.customer_id=b.customer_id

        and b.exemplar_id=e.exemplar_id

        and e.film_id=f.film_id

/

-- Create relationships section -----

ALTER TABLE SCOTT.T_CUSTOMER_RATING ADD CONSTRAINT is Rated FOREIGN
KEY (title, director) REFERENCES SCOTT.T_FILM (title, director)

/

-- Grant permissions section -----

insert into T_GENRE values (1,'sport')

/

insert into T_GENRE values (2,'documentary')

/

insert into T_GENRE values (3,'romantic')

/

insert into T_GENRE values (4,'horror')

/

insert into T_GENRE values (5,'action')

/

insert into T_MEDIUM values (1,'CD - DivX')

/

insert into T_MEDIUM values (2,'CD - VideoCD')

/

insert into T_MEDIUM values (3,'DVD')
```

```
/
insert into T_MEDIUM values (4,'Videotape')
/
insert into T_FILM values (3,'film3','producer3','production_company3',3,15,null)
/
insert into T_FILM values (2,'film2','producer2','production_company2',5,18,3)
/
insert into T_FILM values (1,'film1','producer1','production_company1',5,15,2)
/
insert into T_FILM values (4,'film4','producer4','production_company4',2,0,null)
/
insert into T_FILM values (5,'film5','producer5','production_company5',4,18,null)
/
insert into T_FILM values (6,'film6','producer6','production_company6',5,15,null)
/
insert into T_EXEMPLAR values (1,1,3,3)
/
insert into T_EXEMPLAR values (2,1,3,3)
/
insert into T_EXEMPLAR values (3,1,3,3)
/
insert into T_EXEMPLAR values (4,1,4,2)
/
insert into T_EXEMPLAR values (5,1,4,2)
/
insert into T_EXEMPLAR values (6,2,3,3)
/
insert into T_EXEMPLAR values (7,2,4,2)
/
```

```
insert into T_EXEMPLAR values (8,3,3,3)

/

insert into T_EXEMPLAR values (9,4,4,2)

/

insert into T_EXEMPLAR values (10,5,3,3)

/

insert into T_EXEMPLAR values (11,6,4,2)

/

insert into T_EXEMPLAR values (12,6,4,2)

/

insert into T_CUSTOMER values (1,'name1',cust_address_
typ('address1','post1','city1',null,null,null))

/

insert into T_CUSTOMER values (2,'name2',cust_address_
typ('address2','post2','city2',null,'cz',null))

/

insert into T_CUSTOMER values (3,'name3',cust_address_
typ('address3','post3','city3',null,null,phone_list_typ('541 123 456')))

/

insert into T_CUSTOMER values (4,'name4',cust_address_
typ('address4','post4','city4',null,null,phone_list_typ('541 123 456','596 815 641')))

/

insert into T_CUSTOMER values (5,'name5',cust_address_
typ('address5','post5','city5','state5','sk',null))

/

insert into T_BORROWING (exemplar_id,customer_id,start_date,end_date) values
(11,1,'1.1.2004','3.1.2004')

/

insert into T_BORROWING (exemplar_id,customer_id,start_date,end_date) values
(12,2,'2.1.2004','10.1.2004')

/
```

```
insert into T_BORROWING (exemplar_id,customer_id,start_date) values (1,2,'6.1.2004')
/
insert into T_BORROWING (exemplar_id,customer_id,start_date) values (6,4,'7.1.2004')
/
insert into T_BORROWING (exemplar_id,customer_id,start_date) values (7,3,'7.1.2004')
/
insert into T_BORROWING (exemplar_id,customer_id) values (9,3)
/
insert into T_ORDER_RECORD values (1,2,'1.1.2004')
/
insert into T_ORDER_RECORD values (1,1,'1.1.2004')
/
insert into T_ORDER_RECORD values (2,2,'3.1.2004')
/
insert into T_ORDER_RECORD values (3,3,'6.1.2004')
/
```

## Reverse Engineering

### About Reverse Engineering

Reverse engineering is another of the key features of Toad Data Modeler. This feature allows you to load already existing database structures and create new ER diagrams.

Toad Data Modeler offers you two different methods of object oriented reverse engineering:

- Standard RE - bulk loading of objects  
Indexes, comments, triggers, relationships, constraints, functions, views, procedures, user types etc. are loaded to the new ER diagram. You can select objects you want to reverse via the Object Types and Properties (OTP) selection on tab **What to Reverse**.
- Live RE - loading of even one particular object or a group of selected objects


Live RE offers an easy and quick way how to add new entities, views and synonyms to your existing models. This feature can be also used as a simple model update (limited). See "LIVE Reverse Engineering" (page 180) for more information.

**Note:** To see specific details on reverse engineering of you database, please see the Help file, "Databases" chapter and select your database.

### Scenario

Reverse engineer your Oracle 10g database to Toad Data Modeler (standard RE).

#### *To open the RE Wizard*

» Click  on the toolbar (also **File | Reverse Engineering**).

### Related Topics

Select Stored Alias (page 170)

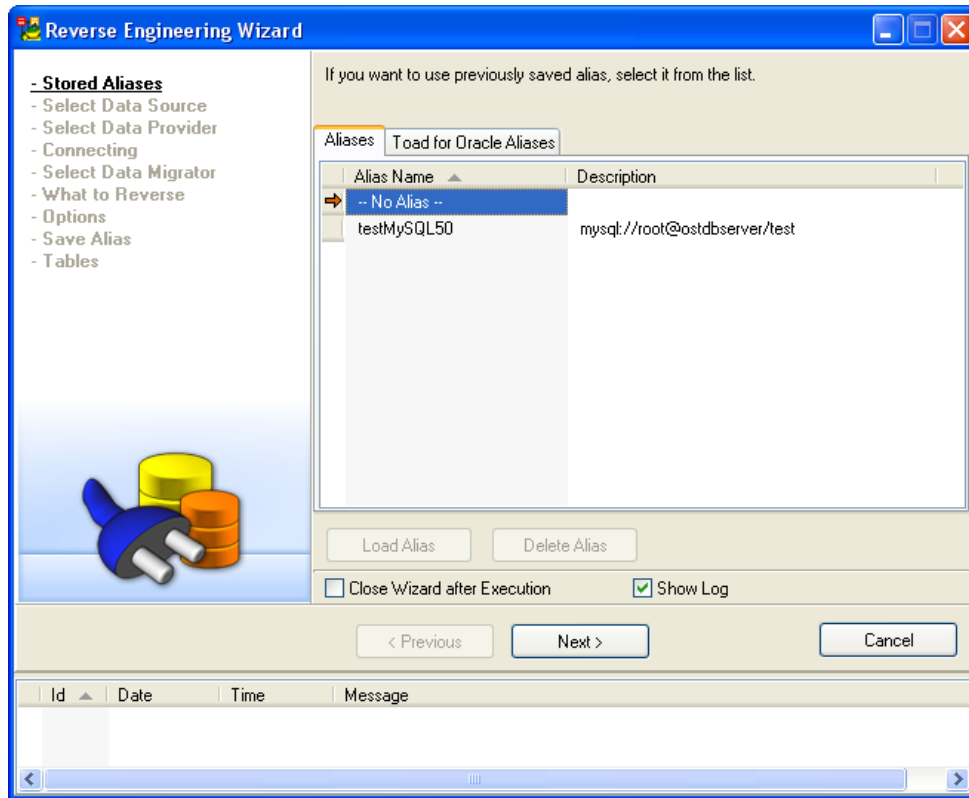
## Select Stored Alias

In Toad Data Modeler, you can save the connection settings and also the options selected in the RE Wizard under aliases. Aliases save your time. Next time you can select only particular alias and do not have to define the connection settings and other options again.

You can create new aliases, modify existing aliases, and delete aliases that you don't use.

**Scenario:** Select options on page **Stored Aliases** in the RE Wizard.

» You want to create a new alias (for Oracle 10g db), therefore Select *-No Alias-* and click **Next**.



#### Notes:

- If you select an existing alias from the list and click **Next**, you will go through all pages of the RE Wizard and can modify the existing alias. The settings on each page will be pre-set and ready for change.

**Tip:** Use this way if you want to create a new alias with similar parameters of the already existing one.

If you select an existing alias from the list and click **Load Alias**, all settings of the selected alias will be loaded automatically and you will switch to page **Tables**.

- On the **Stored Aliases** page, you can also delete aliases - see **Delete Alias**.
- Toad Data Modeler allows you to use Toad for Oracle aliases (it doesn't allow you to save changes to these aliases). If you want to save the Toad for Oracle alias including other settings and options defined in the RE Wizard for next usage, save them under Toad Data Modeler alias on page **Save Alias**.

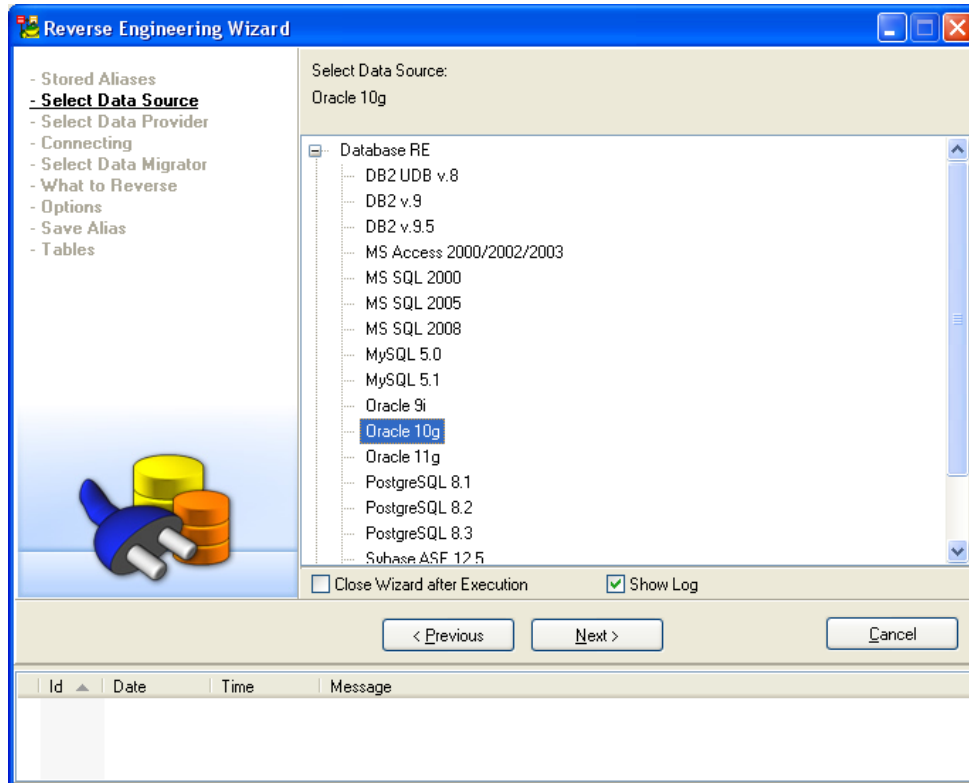
#### Related Topics

Select Data Source (page 172)

## Select Data Source

**Scenario:** On page **Select Data Source**, select a target database.

» Select *Oracle 10g* and click **Next**.



## Related Topics

Select Data Provider (page 172)

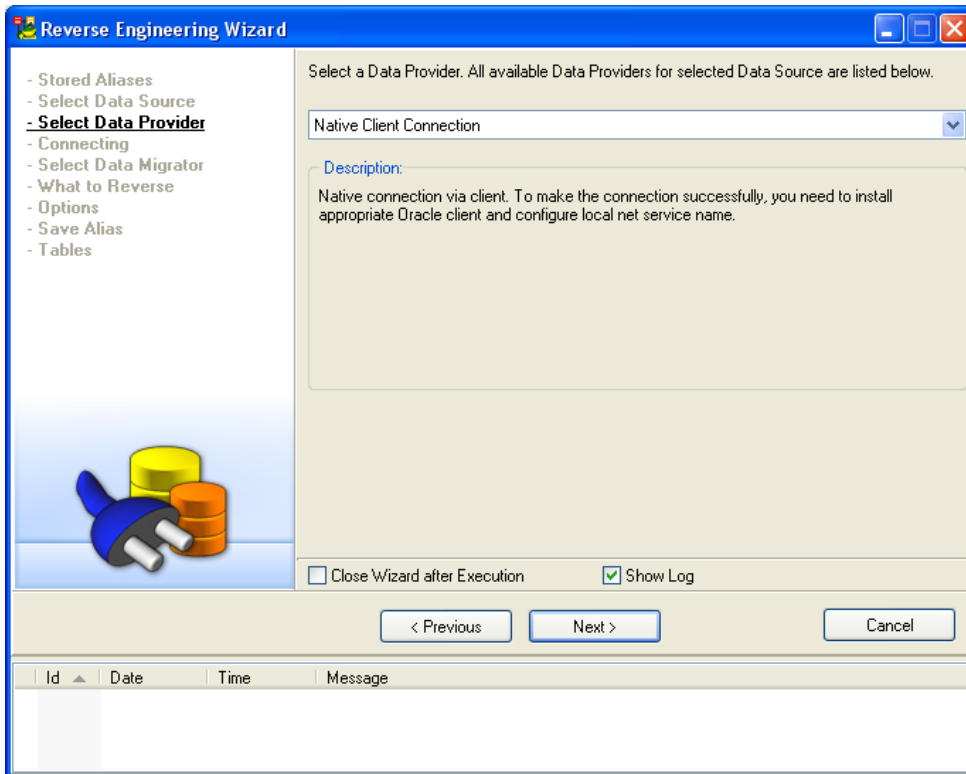
## Select Data Provider

**Scenario:** On the **Select Data Provider** page, select a connection method.

» Select *Native Client Connection*.

**Note:** Available connection methods for Oracle 10g: Native client connection, TCP/IP, ADO. According to the method you select, appropriate information will display in the **Description** area. All the methods are described in the Help file, "Databases", "Oracle 10g", "Reverse Engineering".





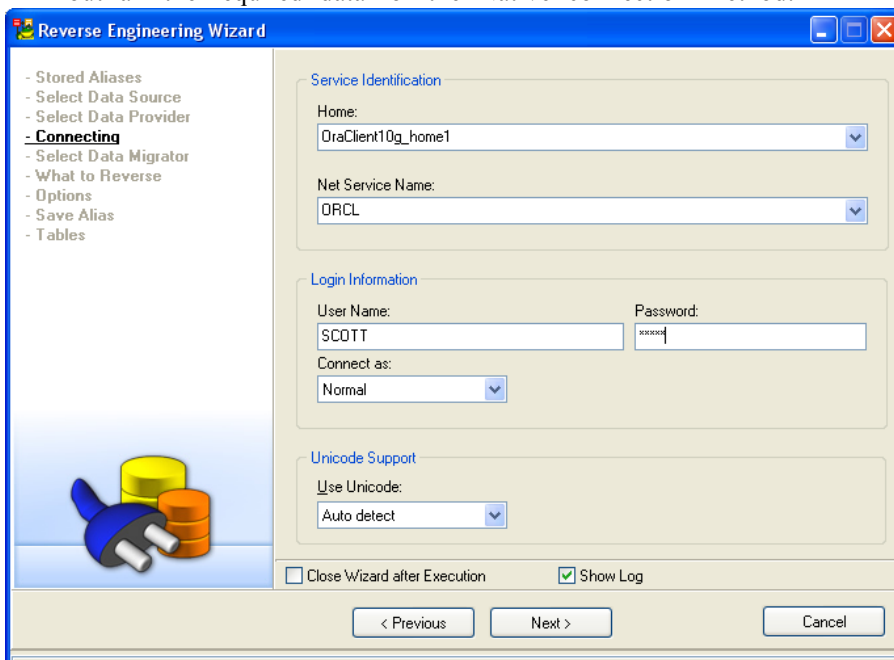
## Related Topics

[Connect to Database \(page 173\)](#)

## Connect to Database

**Scenario:** According to the Native connection method you selected on the previous page, appropriate options are available on this page.

1. Fill out all the required data for the Native connection method.



2. Define **User Name** and **Password**. - Do not fill out if they are contained in the Connection string.
3. Click **Next**.

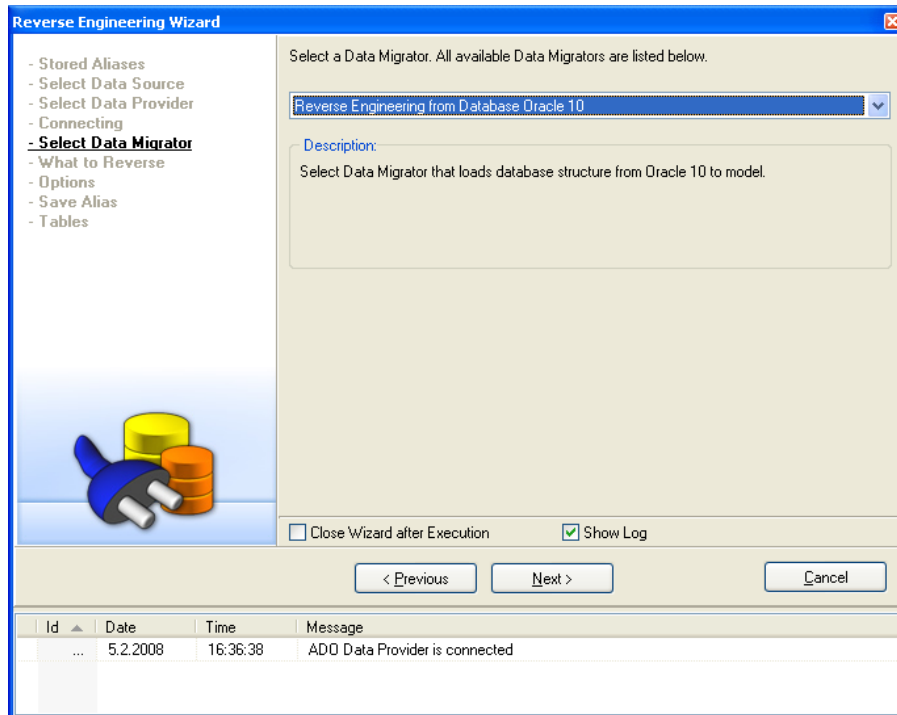
## Related Topics

Select Data Migrator (page 174)

## Select Data Migrator

**Scenario:** Data Migrator loads database structure from particular database to a model. Select the one for Oracle 10g db.

1. Select *Reverse Engineering from Database Oracle 10*.
2. Click **Next**.



## Related Topics

Select Objects to Reverse (page 174)

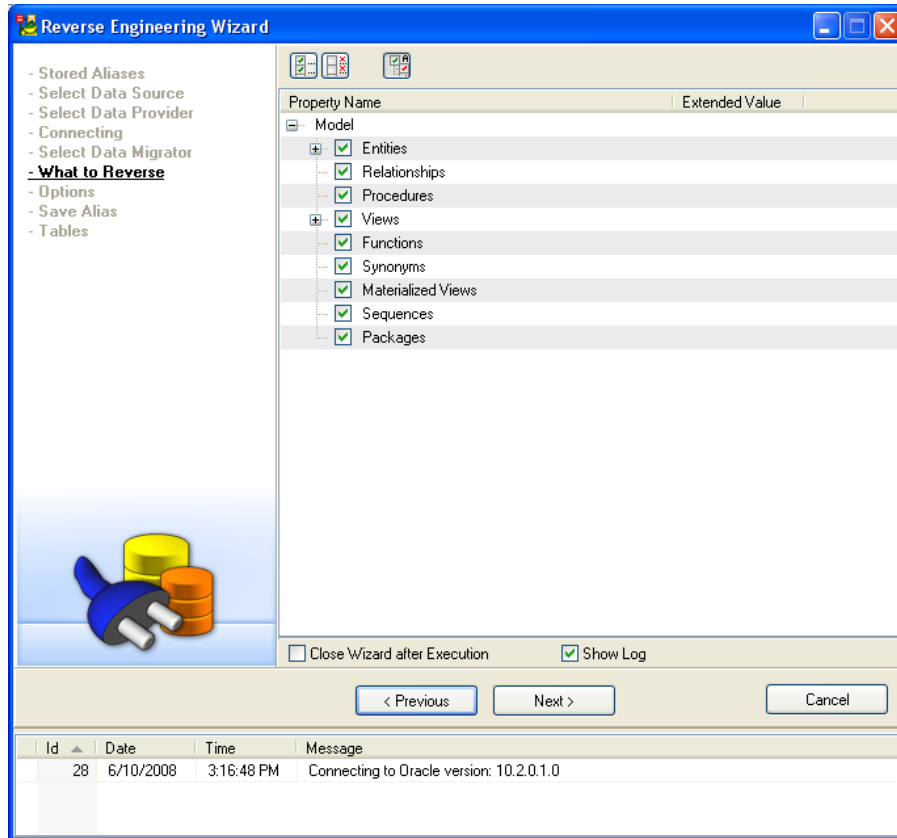
## Select Objects to Reverse

**Scenario:** On page **What to Reverse**, select Object Types and Properties (OTPs).

1. Select all objects to reverse all of them.

**Note:** More details on Object Types and Properties can be found in the Help file, "OTPs" topic.

2. Click **Next**.



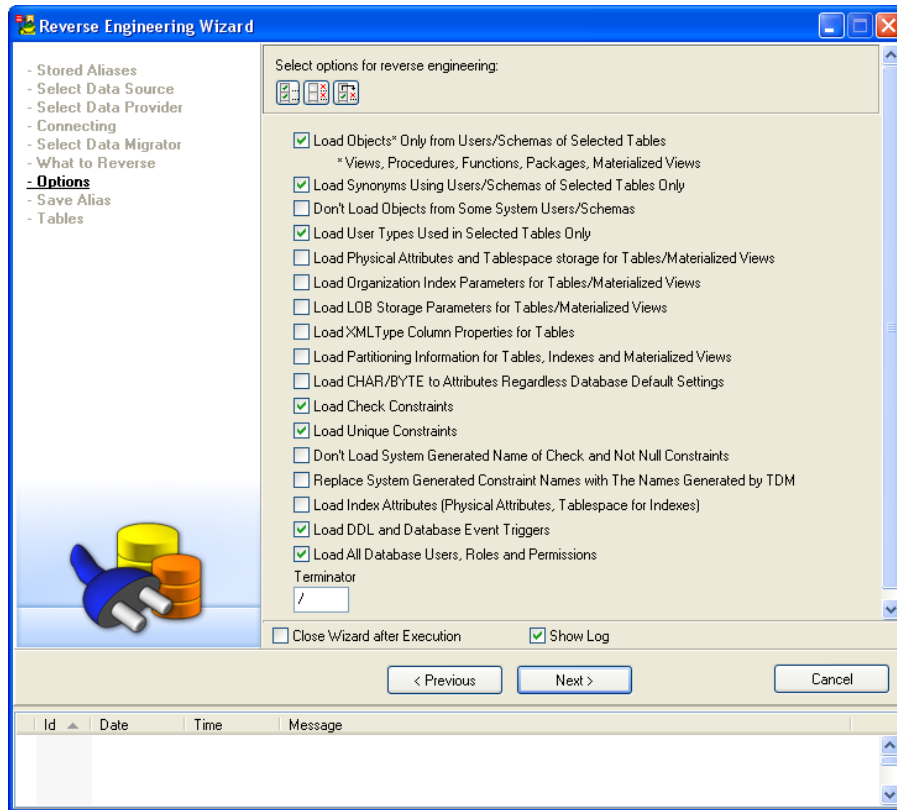
## Related Topics

Select Options (page 175)

## Select Options

**Scenario:** On the **Options** page, select options available for reverse engineering of Oracle 10g.

1. See the selection in the screenshot below.
2. Click **Next**.



**Notes:**

- LOB Parameters of XMLType are loaded. See the following checkboxes:

To load XMLType, select checkbox **Load XMLType Column Properties for Tables**.

To load LOB parameters of table columns and LOB parameters of XMLType parameter, select checkbox **Load LOB Storage Parameters for Tables/Materialized Views**.

XMLType parameters and their LOB parameters are displayed in **Entity Properties** form | **Table Properties** tab. Example:

XMLTYPE "A" STORE AS CLOB

(TABLESPACE "USERS"

CHUNK 8192)

**Note:** The table must have an attribute of XMLTYPE data type.

- To load partition information to indexes of primary and alternate keys, select checkbox **Load Partitioning Information for Tables, Indexes and Materialized Views**.
- The checkbox **Load Synonyms Using Users/Schemas of Selected Tables Only** loads synonyms that have the same schema as Selected Tables, synonyms with different schema but representing objects with the same schema as Selected Tables and public synonyms representing objects with the same schema as Selected Tables.

## Related Topics

Save Alias (page 177)

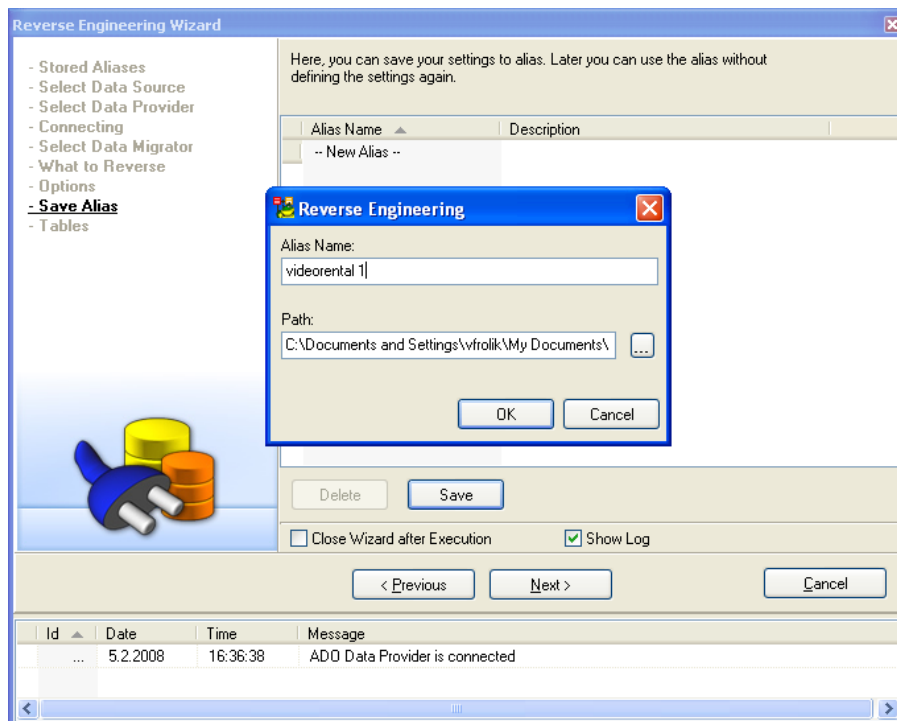
## Save Alias

**Scenario:** Save all the settings for next usage.

1. Click **Save**.
2. In the opened dialog, define the alias name and path where the alias should be saved. The default path is pre-set.

**Note:** Aliases are saved to .TXA files. These files are XML documents, and all data, including password, are saved in a text form. The default path to aliases is defined in the **Settings** menu | **Options** | **Reverse Engineering** | **Path to Aliases**.

3. Click **Next**.



## Related Topics

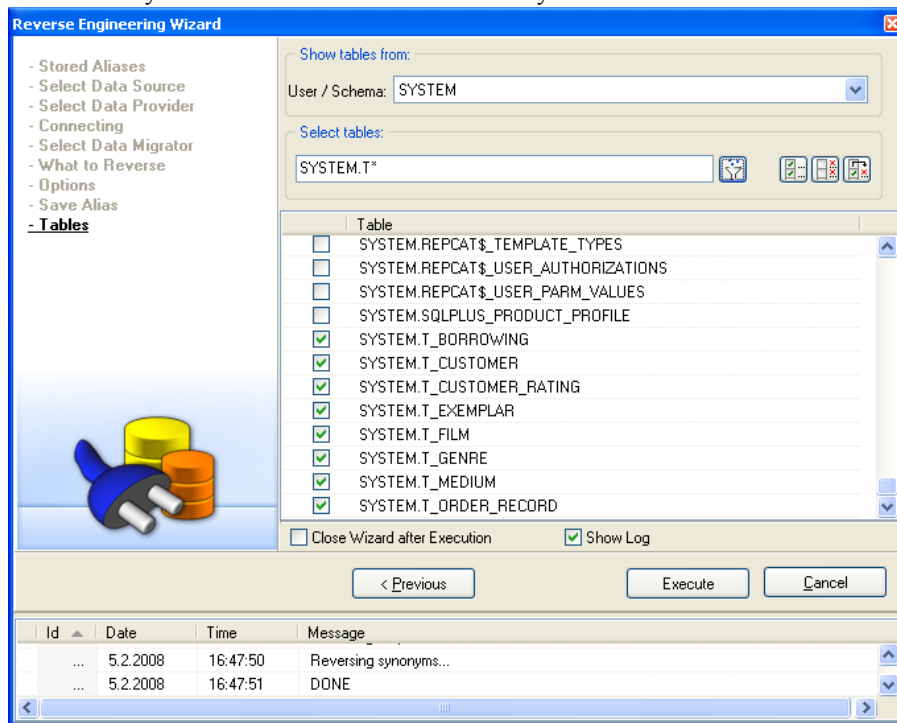
Select Schemas and Tables (page 178)

## Select Schemas and Tables

After the connection is established, a list of all available tables will display. For some databases, SCHEMA box is available.

**Scenario:** Select tables/schemas that you want to load.

1. From the **User/Schema** box, select the *SYSTEM* schema.
2. Take advantage of filter - type *SYSTEM.T\** and click the **Filter** icon to automatically select checkboxes that match your definition.



## Related Topics

Reverse Engineering (page 178)

## Reverse Engineering

**Scenario:** Start the reverse engineering process.

1. Click **Execute**.

**Note:** Follow the messages in the Log area or Message Explorer to see the process.

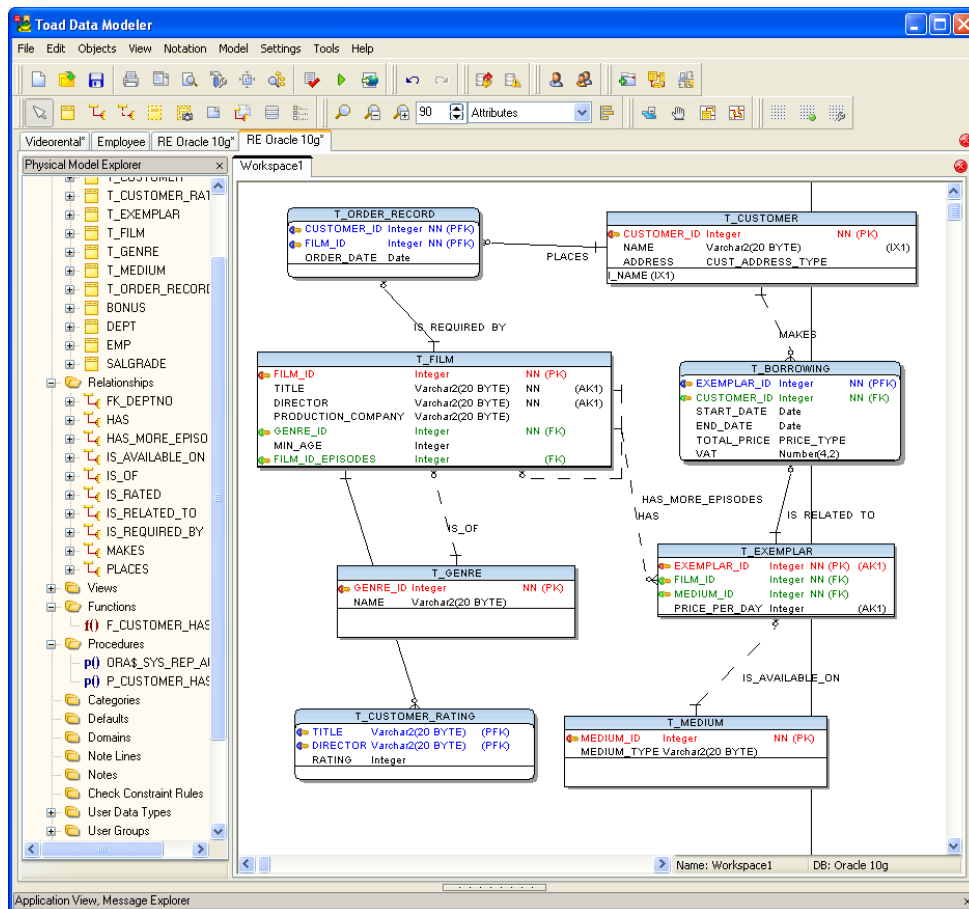
Finally, the following message will display if the RE is finished successfully.



**Tip:** Select **Do not show next time** to get the data and load the model in one operation without waiting for user input next time.

2. Click **OK** and then **Cancel** or close **x** the RE Wizard.

**Example** of reverse engineered ERD of Oracle 10g db



**Related Topics**

LIVE Reverse Engineering (page 180)

## LIVE Reverse Engineering

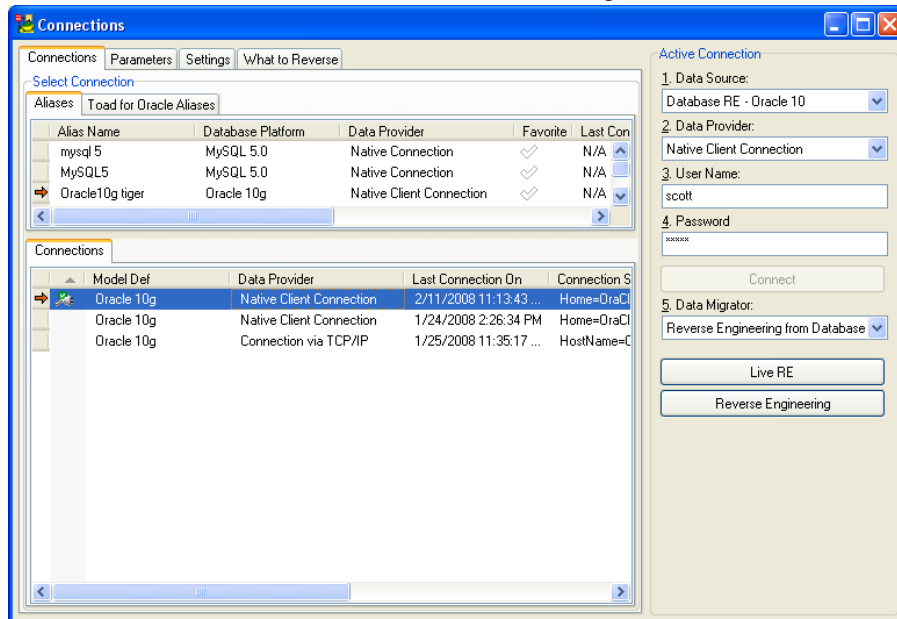
LIVE reverse engineering offers an easy and quick way how to add new entities, views and synonyms (provided that the particular database has synonyms) to your existing models. LIVE reverse engineering is available for all supported database systems.

### How does it work?


Generally, instead of going through steps in the **Reverse Engineering Wizard**, you will select what entities, views and synonyms you want to add to your existing Toad Data Modeler model, and via simple drag and drop techniques you will move them from the **Object Palette** dialog (which is similar to Toad for Oracle Object Palette) to particular Workspace or **Model Explorer**.

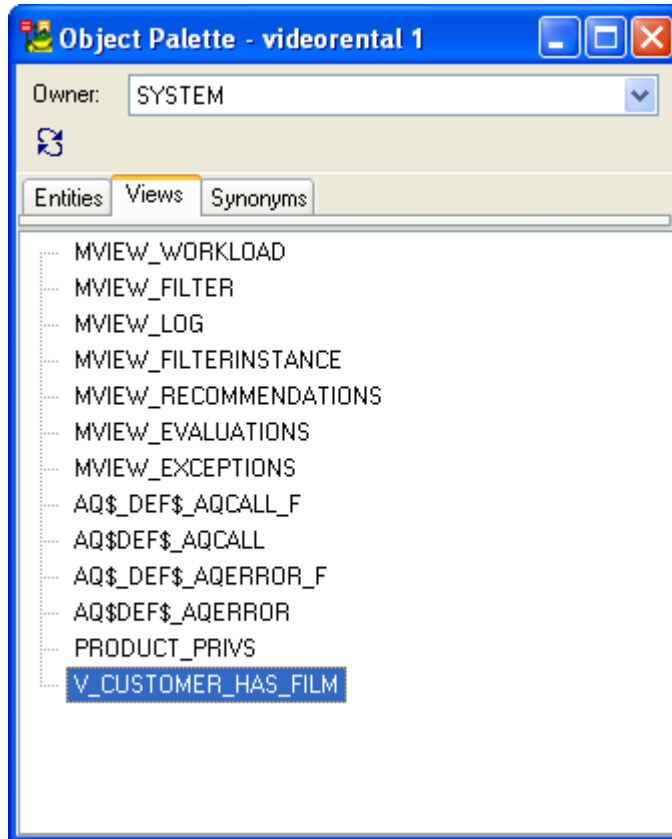
### To perform LIVE RE

1. Select **File | Connections**.
2. On tab **Connections**, you can:
  - a. Either select an alias from the Aliases or Toad for Oracle Aliases tab. Settings on the right in the Active Connection area will update accordingly and so will parameters on tab Parameters.
  - b. Or define new connection settings in the Active Connection area.
3. On tab **Settings**, modify options.
4. On tab **What to Reverse**, select Object Types and Properties (OTPs) you want to reverse.
5. Press **Connect** to connect to your database. In the lower Connections tab, you can see a list of last made connections, including this new one.



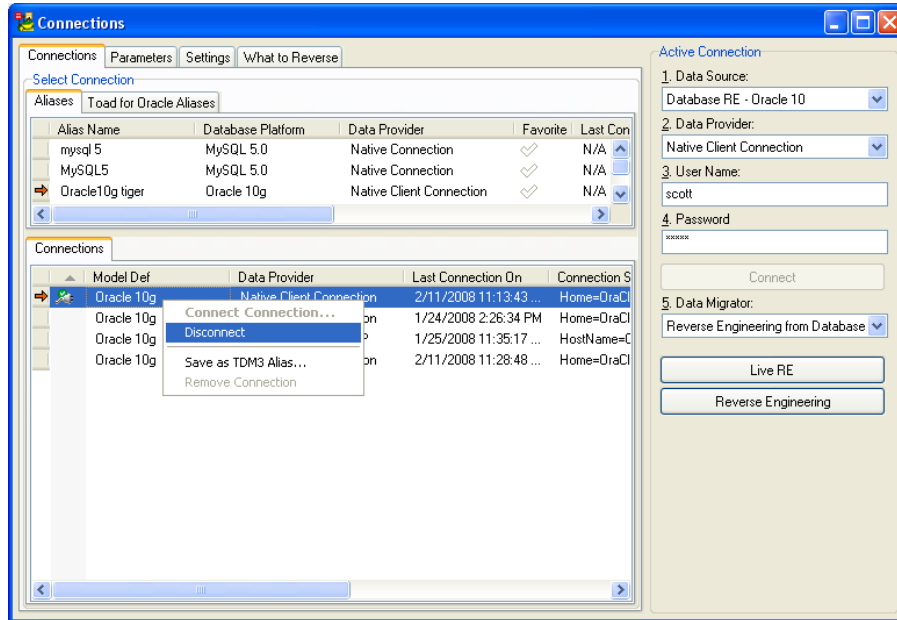


6. As soon as you are connected, this icon  will display next to the appropriate connection and the Connect option will become disabled.
7. Click **Live RE**. (If you clicked **Reverse Engineering**, standard **Reverse Engineering Wizard** would open on page Options.)
8. Select what entities, views and synonyms you want to add to your model. Use SHIFT for multiple selection.



9. Drag and drop the selected items from the Object Palette to particular Workspace or Model Explorer.
10. Close the **Object Palette** dialog.
11. Right-click the connection. You can disconnect from the database or save the connection settings under a new Toad Data Modeler alias.

12. Close the **Connections** dialog then.



## Load Models from DDL/SQL Script

Toad Data Modeler enables you to load models from SQL Script via:


- Standard reverse engineering in the **Reverse Engineering Wizard**
- LIVE Reverse Engineering in the **Connections** dialog

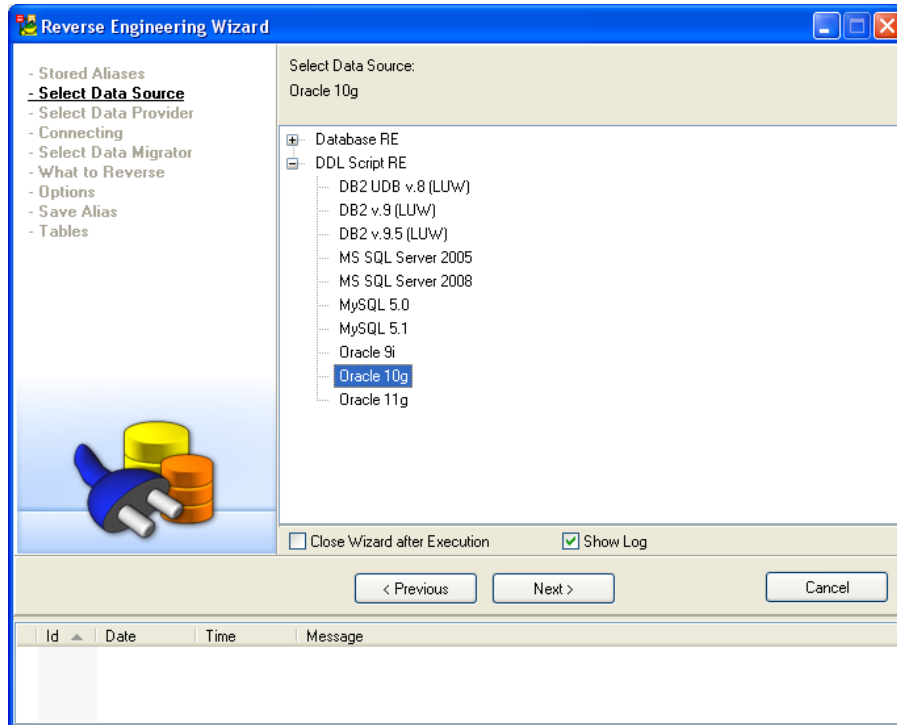
Import from DDL script is supported for the following databases:

- DB2 UDB 8, DB2 v9, DB2 v9.5
- MS SQL Server 2005 and 2008
- MySQL 5.0 and 5.1
- Oracle 9i, 10g and 11g

**Note:** See more details on this in the Help file, "Load Models from SQL Script" topic.

*To load models from SQL script in the Reverse Engineering Wizard*

1. Click  on the toolbar (or select **File | Reverse Engineering**).
2. Select the *Oracle 10g* db under item **DDL Script RE**.

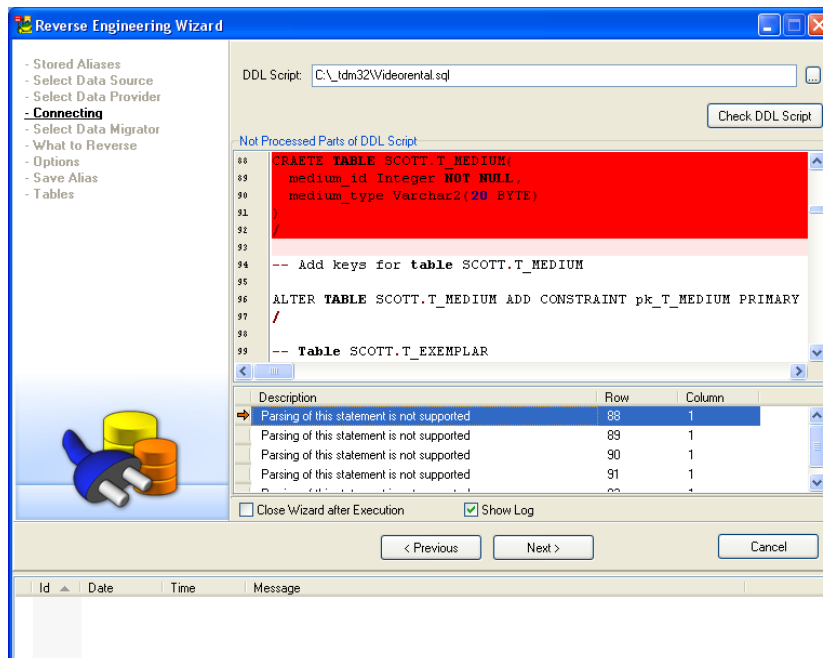


3. Select Data Provider.
4. On the **Connecting** page, use the button on the right of the DDL Script box to define a path to the .sql file. If you enter the path to the DDL Script box directly, press **Check DDL Script**.

If the Log area is clear, the script is O.K.

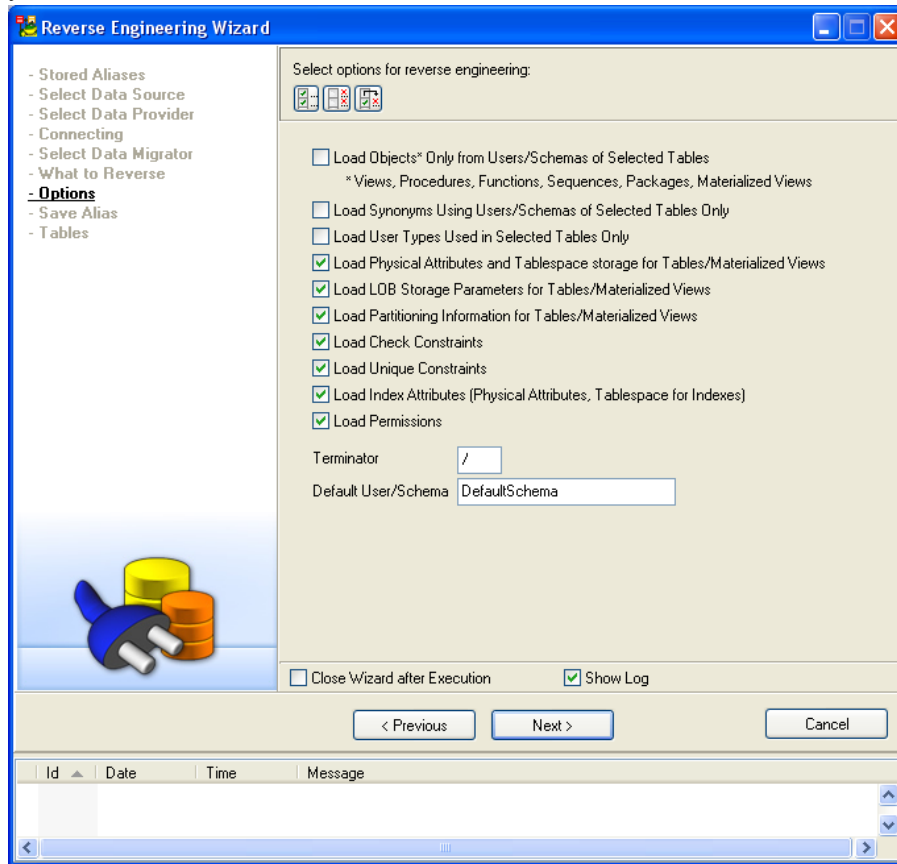
If a typo or unknown command is found, a message will display in the Log area | **Description** column. Double-click the message to highlight the problematic area to see visually in the script what is wrong.

**Example:** The table SCOTT.T\_MEDIUM will not be loaded. See the typo CRAETE instead of CREATE. You can correct the .sql file and load the .sql file again.

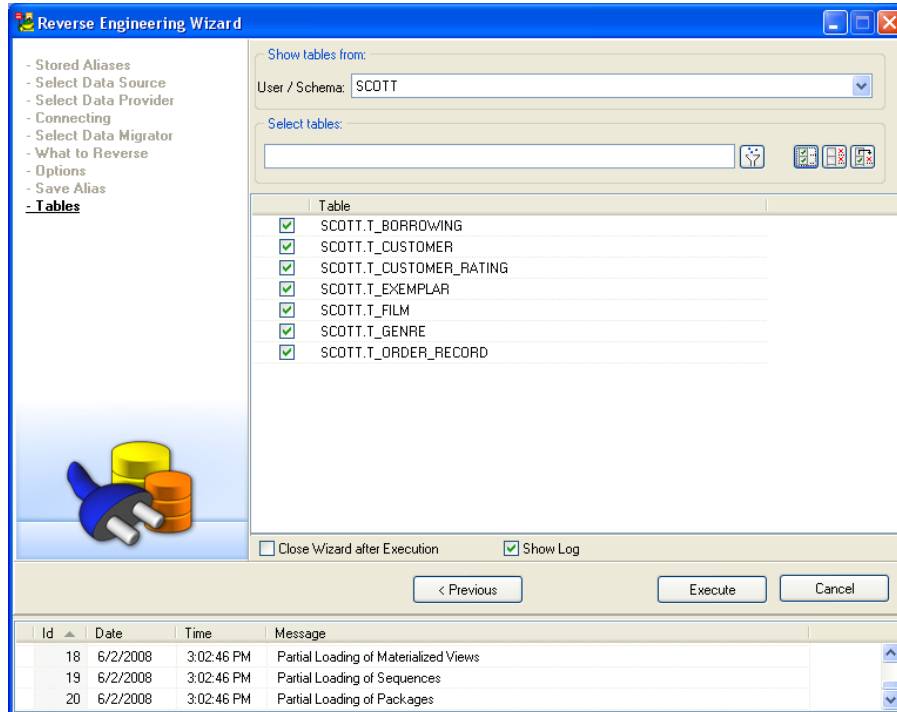


- Take other steps as during standard reverse engineering.
- On the **Options** page, take notice of the box **Default Schema**. A new schema with the name defined in this box will be created in the model. Feel free to change the name as

you want.



7. On the **Tables** page, select the tables to load and click **Execute**.



## Model Compare

### Model Compare

Toad Data Modeler allows you to compare models, see differences between them and generate alter report (HTML, RTF, PDF) containing the differences.

You can compare your models in the **Sync & Convert Wizard**.

See "Alter Reports" (page 144) for more information.

## Model Update/Model Merge

### Model Update

Toad Data Modeler allows you to update models to synchronize changes between your database and your model.

**Example:** You loaded the database structure of your database to Toad Data Modeler (Reverse Engineering). - Model *Videorental* was created. Then you made some changes in your database and now need to update model *Videorental*. For this purpose, you will use the Model Update feature.

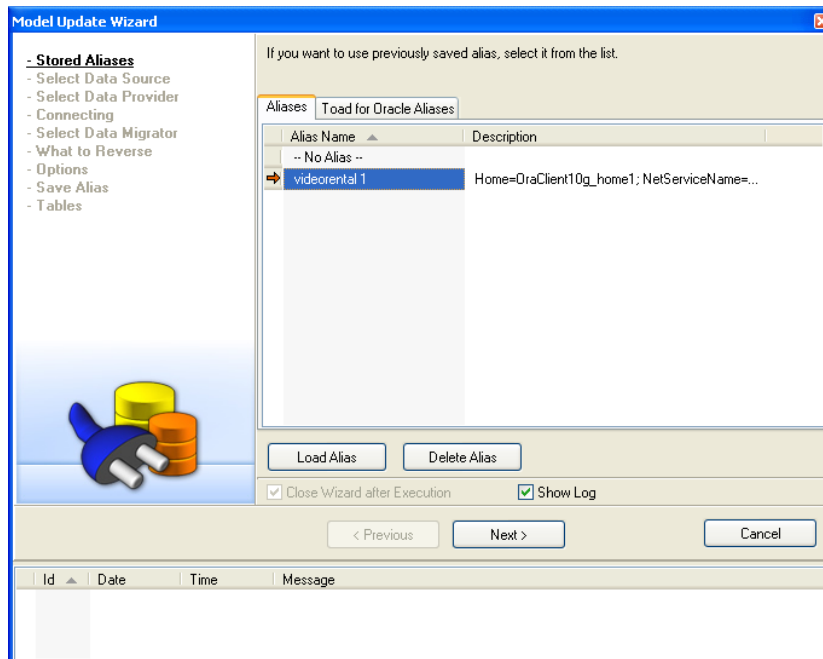
The model update feature includes the Reverse Engineering and Model Merge functions.

## Scenario

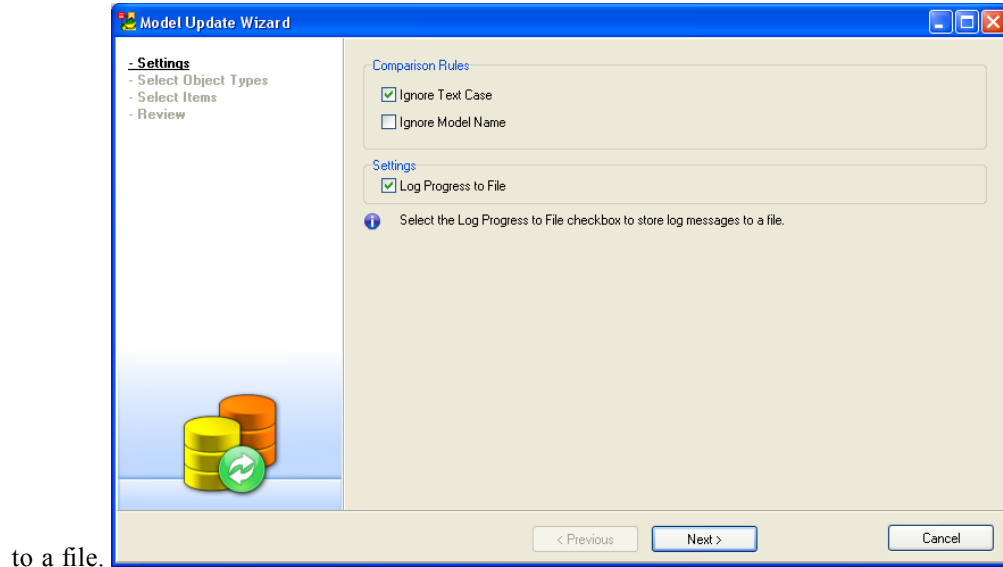
You want to update your Oracle 10g *Videorental* model.

1. Open model *Videorental* that you want to update.
2. Select **Model | Model Update** to open the **Model Update Wizard**.
3. Take the same steps as during the standard reverse engineering process.

**Tip:** Have a look at chapter "Reverse Engineering" and use the already existing alias for Oracle 10g model that you created. Click **Load Alias** to get on page **Tables**.

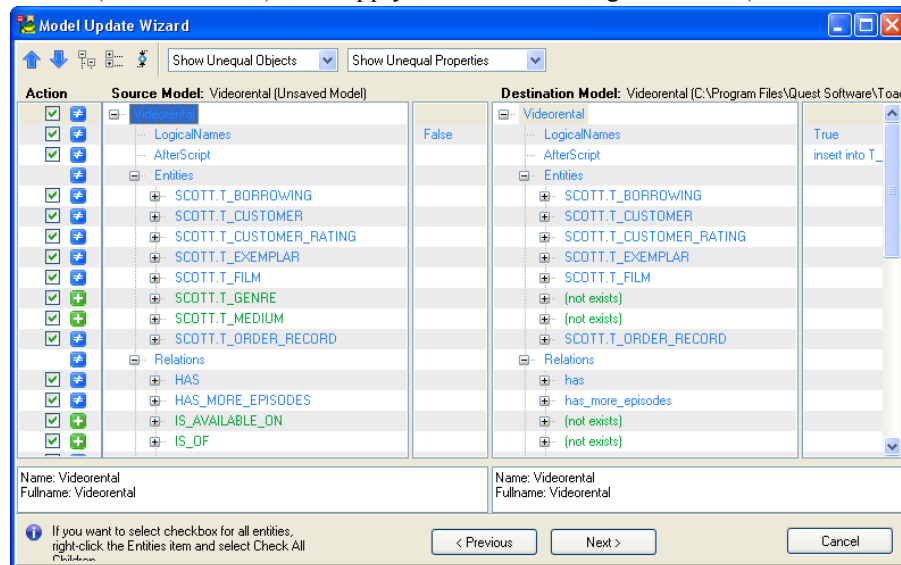


4. On the **Tables** page, select the tables that you want to load and click **Execute**.
5. Now it's time to define settings for the model merge.  
**Log Progress to File** - Select this checkbox to save the messages that the Wizard returns



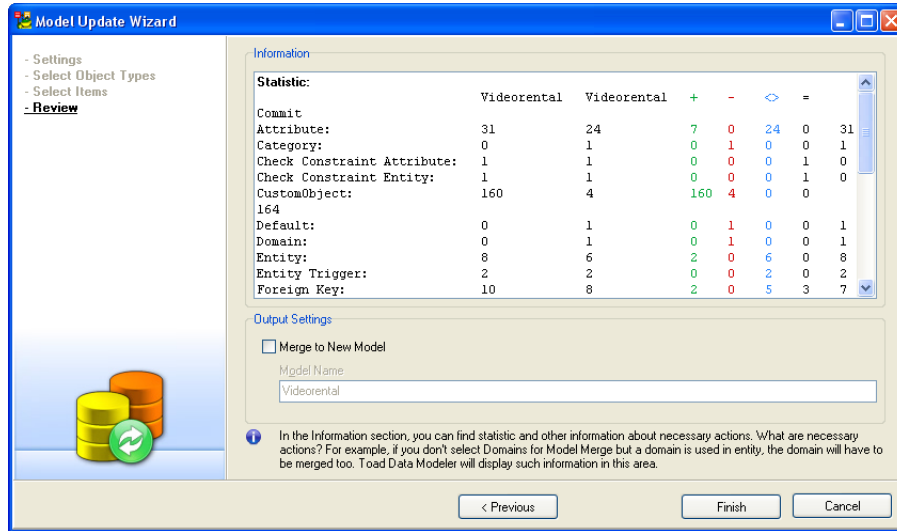
to a file.

6. On the **Select Object Types** page, select **Compare All**.
7. On the **Select Items** page, you can select or deselect the items for the model merge – select or clear the particular checkbox in the **Action** column. Leave the default selection, which means: "Take all differences from the Left model (Source Model) and apply them in the Right model (Destination model)."



8. On the **Review** page, see the statistic information. Here, you can also decide if you want to merge the changes to a new model or if you want to update the existing model *Videorental* (see the **Output Settings** area). You want to update the existing model.





9. Click **Finish**.
10. The Wizard closes and the updated model activates in the Application Window.

## Related Topics

Model Merge (page 189)

Synchronization (page 254)

## Model Merge

Toad Data Modeler offers you two options to merge your models:

- **Simple Model Merge** - allows you to merge physical models very quickly but without a possibility to select what particular items to merge. The target model will always be overwritten.
- **Model Merge in the Sync & Convert Wizard** - allows you to see differences between two models, select particular items to merge and merge the models either to already existing model or into a new model.

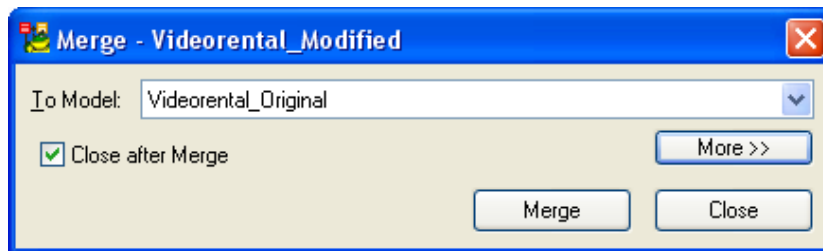
**Example:** You work with several models. Use Model Merge to compare the models and merge them either one to the other (an existing model will be overwritten), or to another model (a new model will be created).

Read the Help file, "Model Merge" topic for more details.

### Scenario 1

You want to merge models *Videorental\_Modified* and *Videorental\_Original* to apply the changes and update model *Videorental\_Original*. You decide to use the Simple Model Merge feature.

1. Open both models.
2. Activate model *Videorental\_Modified*.
3. Select **File | Sync & Convert | Simple Model Merge**.
4. From the **To Model** box, select a model that you want to update - *Videorental\_Original*. (This model will be overwritten.)




5. Select the **Close after Merge** checkbox to close the dialog automatically after the model merge.
6. Click **Merge**.

## Scenario 2

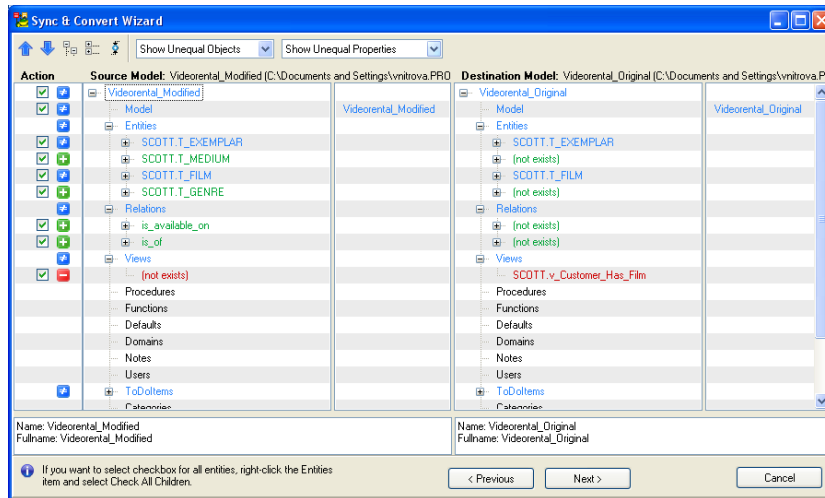
You made some changes in your Oracle 10g database and did the reverse engineering to Toad Data Modeler again. New model *Videorental\_Modified* was created after the RE. Now you want to update the existing *Videorental\_Original* model. -> You will merge models *Videorental\_Modified* and *Videorental\_Original*.

The reversed model *Videorental\_Modified* is open and currently active in the Application Window.

You decide to merge the models in the Sync & Convert Wizard.

1. Click  on the toolbar.
2. On the **Action** page, select **Merge Models**.
3. On the **Select Left Side** page, from the **Available Model** box, select model *Videorental\_Modified*.
4. On the **Select Right Side** page, select **File (\*.txp, \*.txl)**. Click the button on the right to find model *Videorental\_Original* in the **Open** dialog.
5. On the **Settings** page, select **Ignore Text Case** and **Log Progress to File** checkboxes.
6. On the **Select Object Types** page, select **Compare All**.
7. On the **Select Items** page, select or deselect the items for the model merge – select or clear the particular checkbox in the **Action** column.

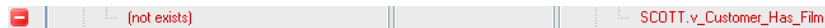
See what is shown here and how.



This icon and the blue color say: "There is a difference."



This icon and the green color say: "This item exists in Left model and is missing in Right model."



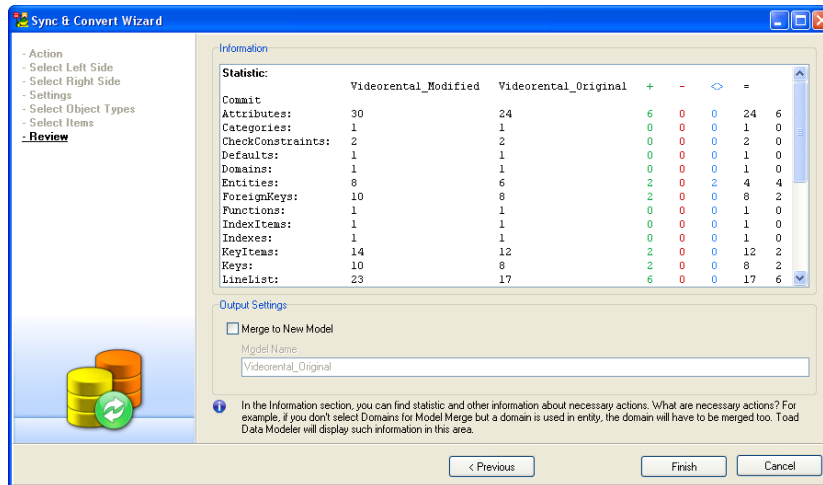
This icon and the red color say: "This item is missing in Left model and exists in Right model."

By default, all the **Action** checkboxes are selected, which means: "Take all differences from the Left model (Source Model) and apply them in the Right model (Destination model)."

In this example it means, e.g.:

- Model name line: The Right model *Videorental\_Original* (Destination Model) will be renamed to *Videorental\_Modified* after the model merge.
  - *SCOTT.T\_MEDIUM* entity: The entity will be added to the Right model.
  - *SCOTT.v\_Customer\_Has\_Film* view: The view will be deleted from the Right model .
8. On the **Review** page, see the statistic information. Here, you can also decide if you want to merge the changes to a new model or if you want to update the existing model *Videorental\_Original* (see the **Output Settings** area).

You want to update the existing model.



9. Click **Finish**.
10. The **Sync & Convert Wizard** closes and the updated model opens in the Application Window.

Model Update (page 186)

## Alter Scripts

### Alter Scripts

Toad Data Modeler allows you to generate alter scripts for the following databases:

- Oracle 9i, Oracle 10g, Oracle 11g
- MS SQL Server 2005, MS SQL Server 2008
- PostgreSQL 8.1, 8.2, 8.3
- DB2 UDB 8, DB2 v9, DB2 v9.5

Later you can execute the alter script generated in Toad Data Modeler in a third party software to apply the changes in your database (e.g. Toad for Oracle etc.). This way, you can synchronize your model in Toad Data Modeler with your database structure.

Alter scripts are generated in the **Sync & Convert Wizard**.

### Scenario

You made some changes in your Oracle 10g model *Videorental* and resaved it as model *Videorental Modified*. (You added and also deleted some items, changed some properties etc.).


Now you want to generate complete alter script, including CREATE,

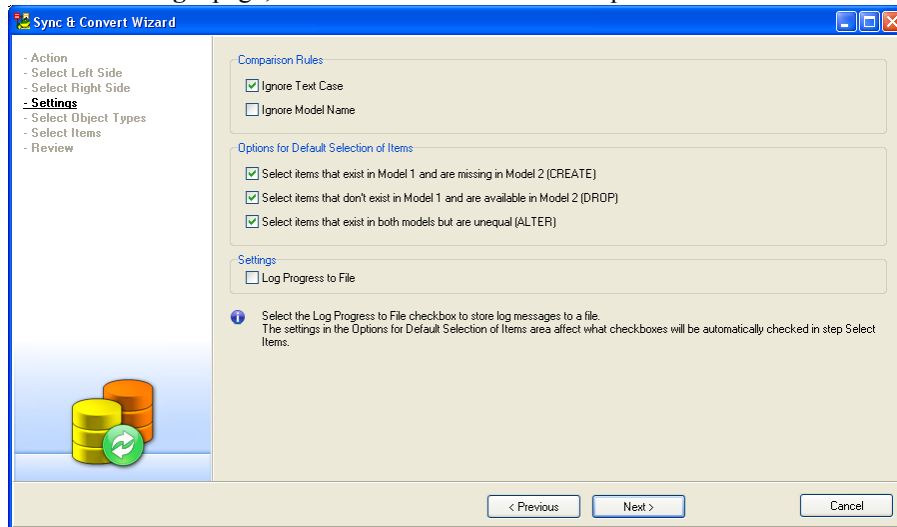
DROP and ALTER statements, for the model *Videorental*.

You have both models open in Toad Data Modeler.

Here is a list of changes you made and saved in model *Videorental\_Modified*:

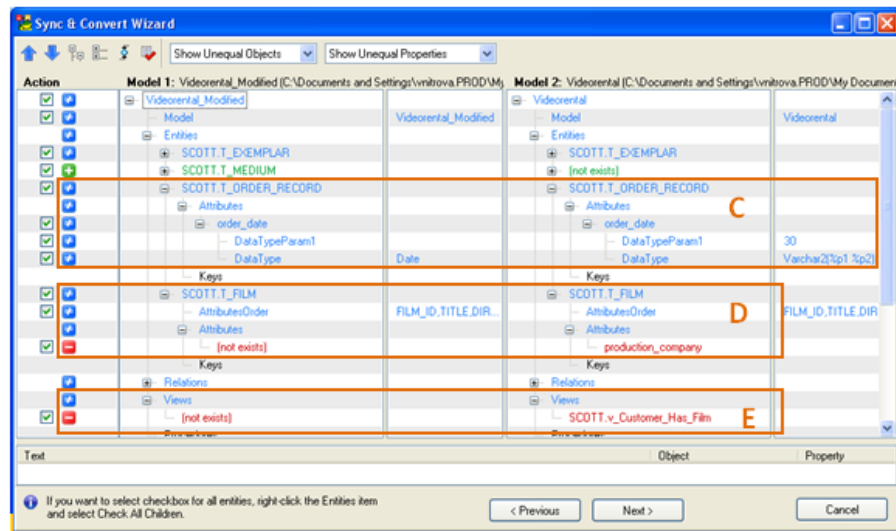
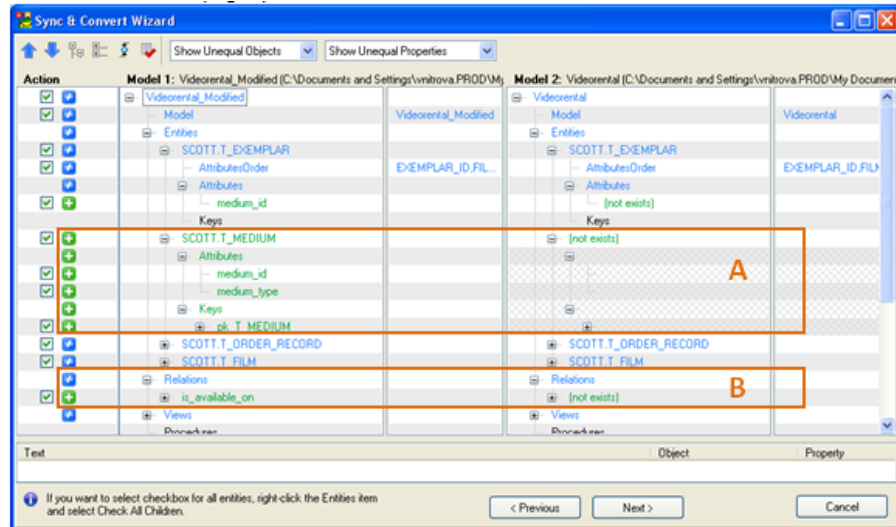
- A. Created entity *Medium* with attributes: *Medium ID* (PK) and *Medium Type*.
- B. Created relationship *Is Available On* connecting entities *Medium* and *Exemplar*.
- C. Changed data type for attribute *Order Date* (*Order Record* entity) from *Varchar2* to *Date*.
- D. Deleted attribute *Production Company* (*Film* entity).
- E. Deleted view *v\_Customer\_Has\_Film*.

1. Click  on the toolbar.
2. On the **Action** page, select **Generate Alter Script**.
3. On the **Select Left Side** page, from the **Available Model** box, select model *Videorental\_Modified*.
4. On the **Select Right Side** page, from the **Available Model** box, select model *Videorental*.
5. On the **Settings** page, select from the available options:



Particularly, take notice of options in the area **Options for Default Selection of Items**. You want to generate complete alter script (for items you added, deleted and also for properties you changed). So, leave the default selection – all of them are selected. The settings affect the default selection of items on page **Select Items**, which helps you to generate the alter script faster and more comfortably.

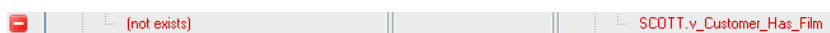
6. On the **Select Object Types** page, select **Compare All**.
7. On the **Select Items** page, you can see the differences between the models.



This icon and the blue color say: "There is a difference."



This icon and the green color say: "This item exists in Left model and is missing in Right model."



This icon and the red color say: "This item is missing in Left model and exists in Right model."

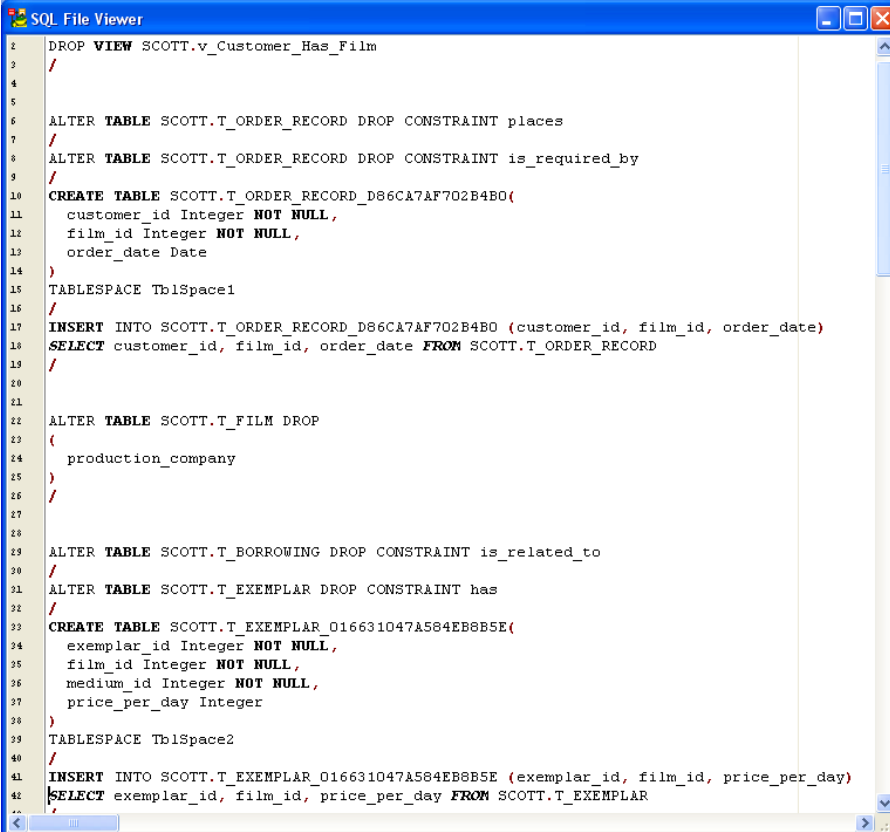
See the **Action** checkboxes. All of them are selected by default. => This selection matches the options you selected on page **Settings** in the **Options for Default Selection of Items** area. You want to generate CREATE, DROP and also ALTER statements, therefore for all the displayed differences the **Action** checkbox is selected.

The selected **Action** checkboxes mean: "Generate alter script for the selected differences between Left model (Model 1) and Right model (Model 2)."

8. On the **Review** page, you can see statistic information. In the **Output File** box, define where you want to save the generated alter script. Default path is: C:\Documents and Settings\user name\My Documents\Toad Data Modeler Beta\GeneratedScripts\AlterScript.sql
9. Click **Finish**.

**Note:** If the AlterScript.sql already exists, you are asked whether you want to replace it.

10. The generated alter script will open in SQL File Viewer.



```

1 DROP VIEW SCOTT.v_Customer_Has_Film
2 /
3
4
5
6 ALTER TABLE SCOTT.T_ORDER_RECORD DROP CONSTRAINT places
7 /
8 ALTER TABLE SCOTT.T_ORDER_RECORD DROP CONSTRAINT is_required_by
9 /
10 CREATE TABLE SCOTT.T_ORDER_RECORD_D86CA7AF702B4B0(
11     customer_id Integer NOT NULL,
12     film_id Integer NOT NULL,
13     order_date Date
14 )
15 TABLESPACE TblSpace1
16 /
17 INSERT INTO SCOTT.T_ORDER_RECORD_D86CA7AF702B4B0 (customer_id, film_id, order_date)
18 SELECT customer_id, film_id, order_date FROM SCOTT.T_ORDER_RECORD
19 /
20
21
22 ALTER TABLE SCOTT.T_FILM DROP
23 (
24     production_company
25 )
26 /
27
28
29 ALTER TABLE SCOTT.T_BORROWING DROP CONSTRAINT is_related_to
30 /
31 ALTER TABLE SCOTT.T_EXEMPLAR DROP CONSTRAINT has
32 /
33 CREATE TABLE SCOTT.T_EXEMPLAR_016631047A584EB8B5E(
34     exemplar_id Integer NOT NULL,
35     film_id Integer NOT NULL,
36     medium_id Integer NOT NULL,
37     price_per_day Integer
38 )
39 TABLESPACE TblSpace2
40 /
41 INSERT INTO SCOTT.T_EXEMPLAR_016631047A584EB8B5E (exemplar_id, film_id, price_per_day)
42 SELECT exemplar_id, film_id, price_per_day FROM SCOTT.T_EXEMPLAR
43 /
  
```

#### Notes:

1. Settings defined in **DDL Script Generation** dialog are taken for the alter script generation (quotations, schema prefix, primary key inside or outside statements).

2. It is recommended to set up the same DDL script generation settings for both models that are being compared.

## Model Conversion

### Model Conversion - PER - PER

Toad Data Modeler allows you to convert your PER models from one database system to another database system, e.g. model in Oracle 10g can be converted to SQL Server 2005 model.

The conversion can be executed via:

- **Sync & Convert Wizard (File | Sync & Convert)**
- **Simple Model Conversion (File | Sync & Convert)**


During the conversion:

- Data Types are converted.
- Permissions are compared by names.
- SQL code is copied and commented not to generate any code.

**Note:** More information on PER to PER conversion and Simple Model Conversion can be found in the Help file, "Model Conversion" chapter.

#### Scenario

Convert your Oracle 10g model *Videorental* to MS SQL 2008. Use the **Sync & Convert Wizard**.

1. Click  on the toolbar.
2. On the **Action** page, select **Convert Model to Another Target Database System**.
3. On the **Select Left Side** page, select **File (\*.txp, \*.txl)**. Click the button on the right to find model *Videorental* in the **Open** dialog.
4. On the **Select Right Side** page, select a target database system *MS SQL 2008*.
5. On the **Settings** page, you can select from the following options:

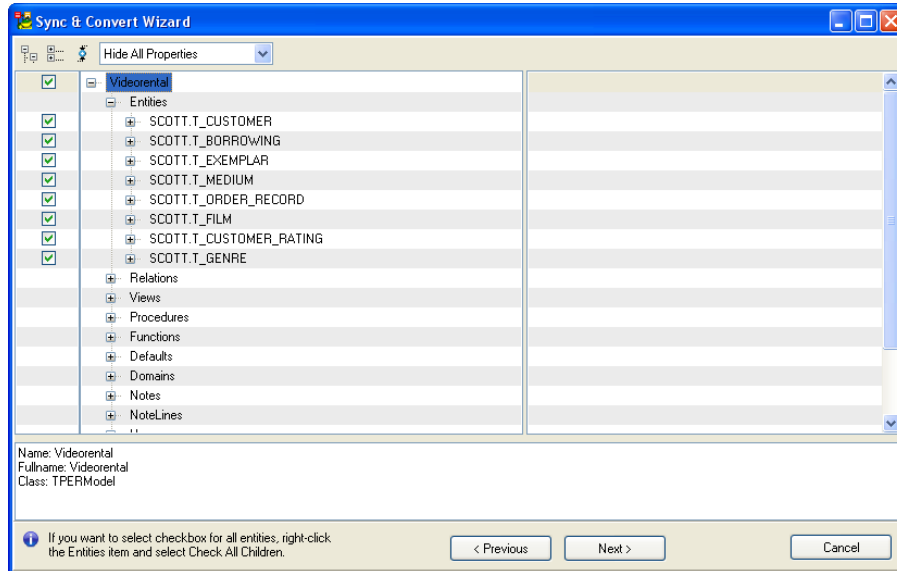
**Comment Out Database Specific Items** - Select it and see the final result shown below.

**Log Progress to File** - Select this checkbox to save the messages that the Wizard returns to a file.

Select both checkboxes.

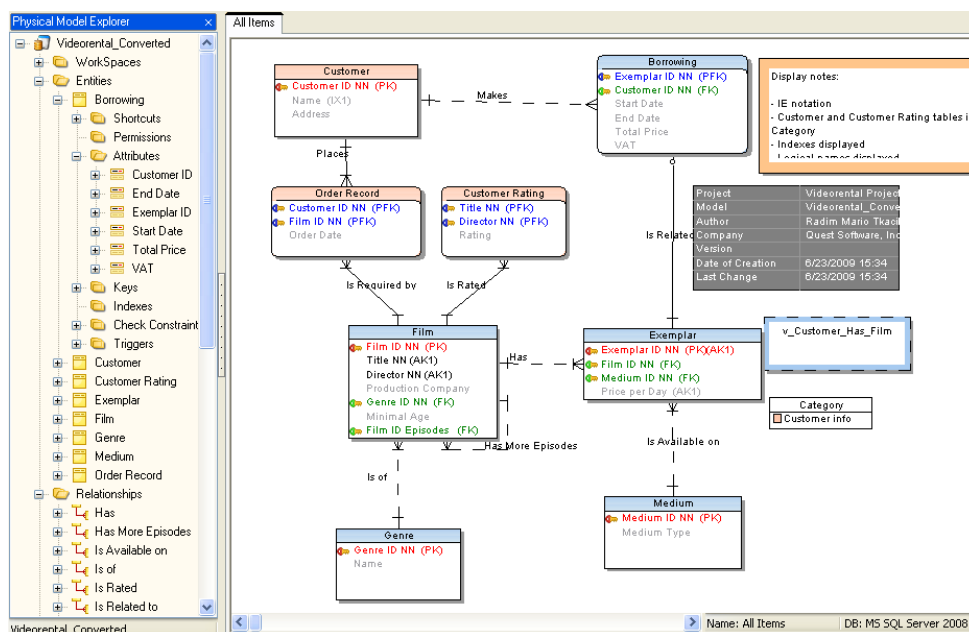


6. On the **Select Object Types** page, select **Convert All**.
7. On the **Select Items** page, you can select/deselect particular items of your model that you want or do not want to convert. Leave the default selection to convert all items of model *Videorental*.



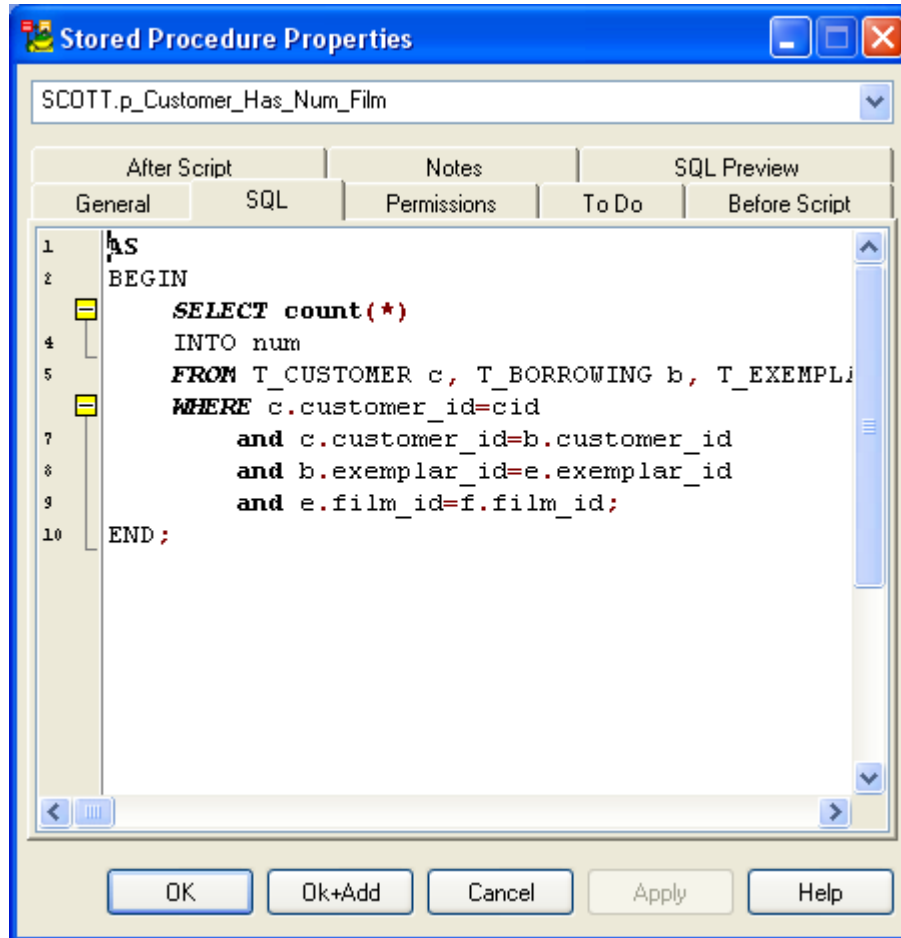
8. On the **Review** page, see the statistic information and define a model name for the converted model in the **New Model Name** box - *Videorental\_Converted*.
9. Click **Finish**.
10. The **Sync & Convert Wizard** closes and the new model *Videorental\_Converted* of MS SQL 2008 database opens in the Application Window automatically.

Result:

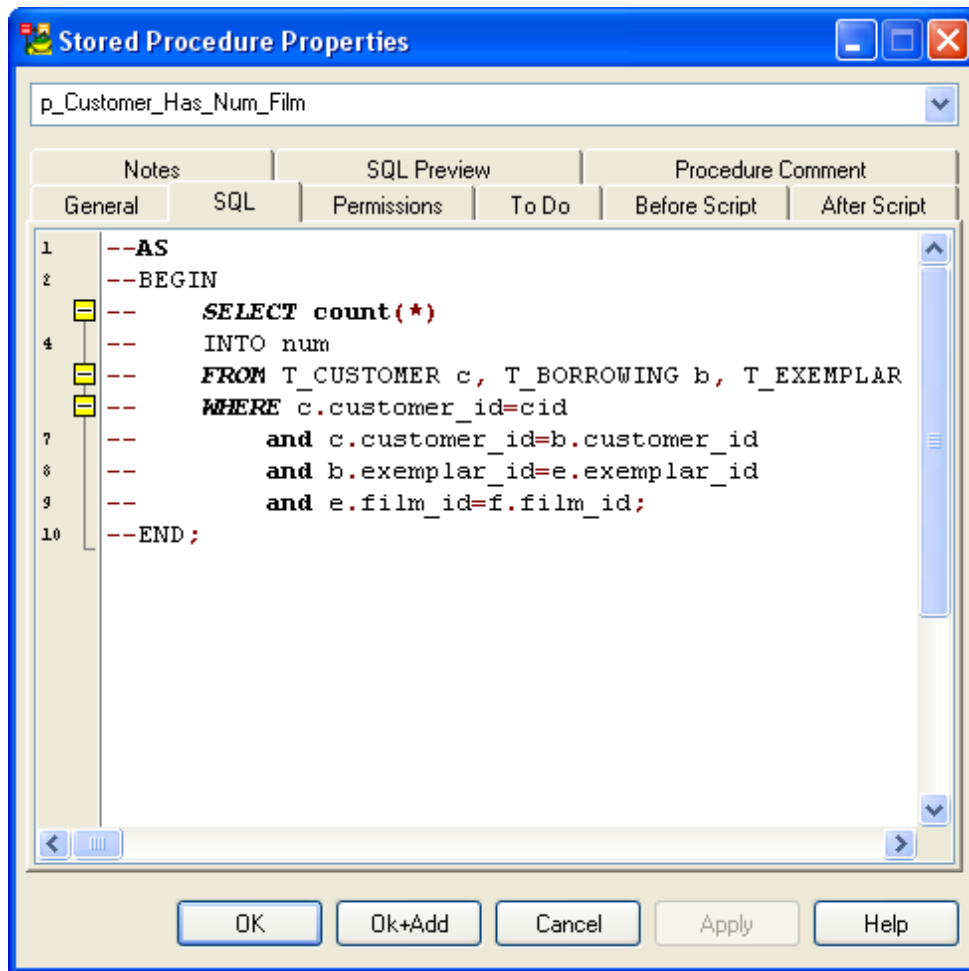


See how SQL code is commented:

Oracle 10g



Converted to MS SQL 2008



## Related Topics

Model Conversion - PER - LER (page 199)


## Model Conversion - PER - LER

The PER to LER conversion allows you to create new logical models from existing PER models comfortably, and also allows you to update your logical models.

All items supported in LER model should be converted from PER model to LER model properly. For more details on the changes and what is converted and how, please see the Help file, "Model Conversion" chapter.

### Scenario

You want to convert your model *Videorental* (Oracle 10g) to a new logical model.

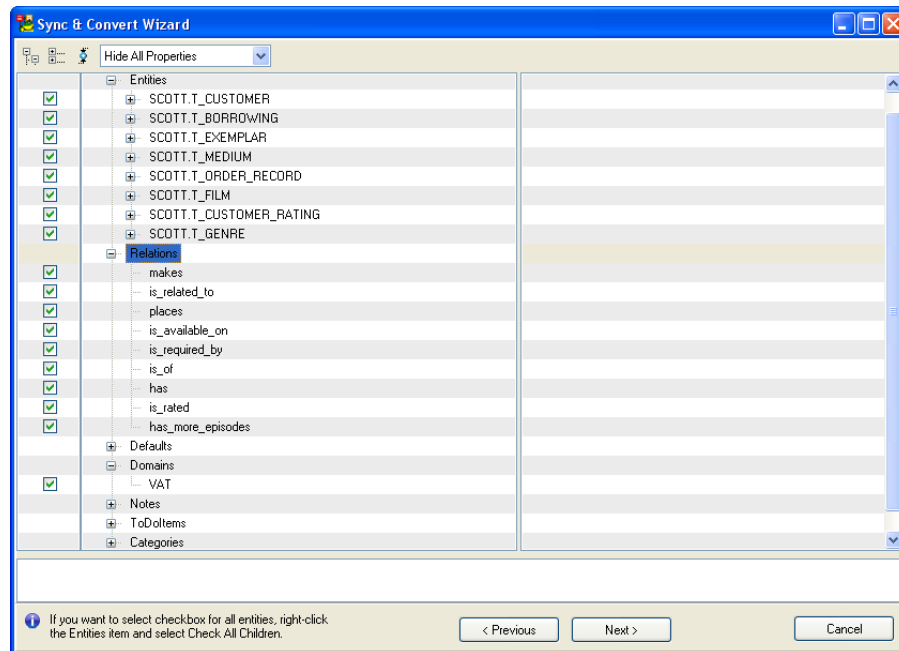
1. Click  on the toolbar.
2. On the **Action** page, select **Convert Model to Another Target Database System**.
3. On the **Select Left Side** page, select **File (\*.txp, \*.txl)**. Click the button on the right to find model *Videorental* in the **Open** dialog.
4. On the **Select Right Side** page, select *Logical Model*.
5. On the **Settings** page, you can select from the following options:

#### Comment Out Database Specific Items

**Log Progress to File** - Select this checkbox to save the messages that the Wizard returns to a file.

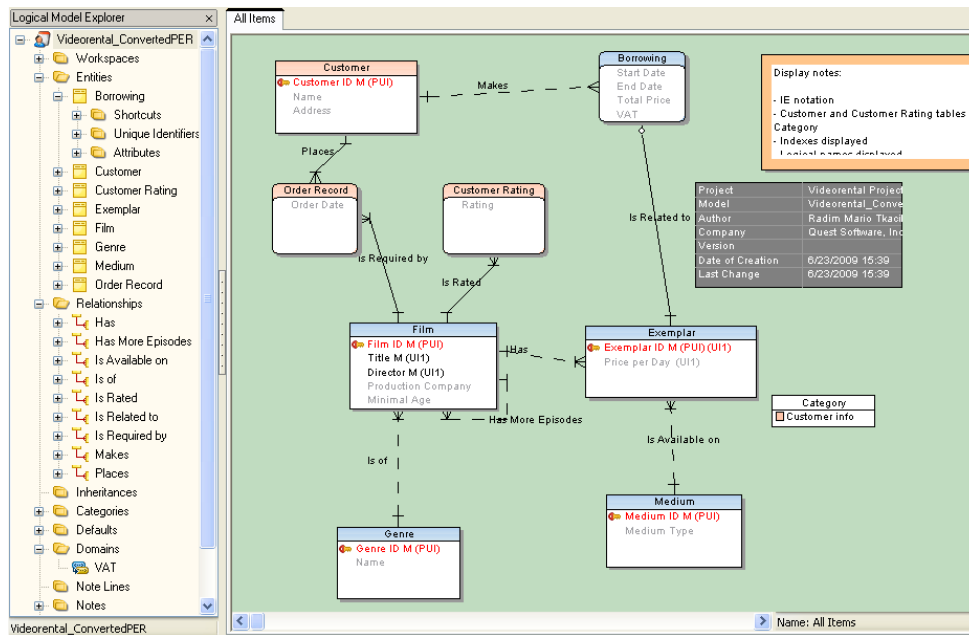
Select both checkboxes.

6. On the **Select Object Types** page, select **Convert All**.
7. On the **Select Items** page, you can select/deselect particular items of your model that you want or do not want to convert. Leave the default selection to convert all items of model *Videorental*.



8. On the **Review** page, see the statistic information and define a model name for the converted model in the **New Model Name** box - *Videorental\_ConvertedPER*.
9. Click **Finish**.
10. The **Sync & Convert Wizard** closes and the new model *Videorental\_ConvertedPER* opens in the Application Window automatically.

Result:



## Related Topics

Conversion LER - PER (page 219)

# Import Toad Data Modeler 2.x Models

## Import Toad Data Modeler 2.x Model

In Toad Data Modeler 3.x, you can open your models created in Toad Data Modeler 2.x - \*.dm2 and \*.dmx files.

*To open a file created in Toad Data Modeler 2.x*

1. In Toad Data Modeler 3.x, select **File** | **Open**.
2. From the **Files of type** box, select item *Toad Data Modeler version 2.x Model in binary format (\*.dm2)* or *Toad Data Modeler version 2.x Model in XML format (\*.dmx)*.
3. Select the file and click **Open**.
4. Resave the file in Toad Data Modeler 3.x format. - Select **File** | **Save Model as**.
5. From the **Save as type** box, select item *Physical Toad Data Modeler Models in XML Format (\*.txp)*.
6. Click **Save**.

# Logical Data Model

## About Logical Model

Toad Data Modeler allows you to design and maintain a logical model giving a complete picture of the business area. Logical model is independent of the database platform and is much simpler than physical model. It uses objects such as inheritance, valid values or M:N relationships. From the Logical ER (LER) diagram, you can build a Physical ER (PER) diagram of the selected database platform (LER to PER conversion).

In this chapter, you will find how to create a logical model in Toad Data Modeler, how to verify it, generate report for your logical model, convert it to physical model and more.

**Note:** In the installation package, you can find a sample logical model *Employee*. Default location: C:\Program Files\Quest Software\Toad Data Modeler 3\Samples.

### Related Topics


Create Logical Model (page 202)

## Create Logical ER Diagram

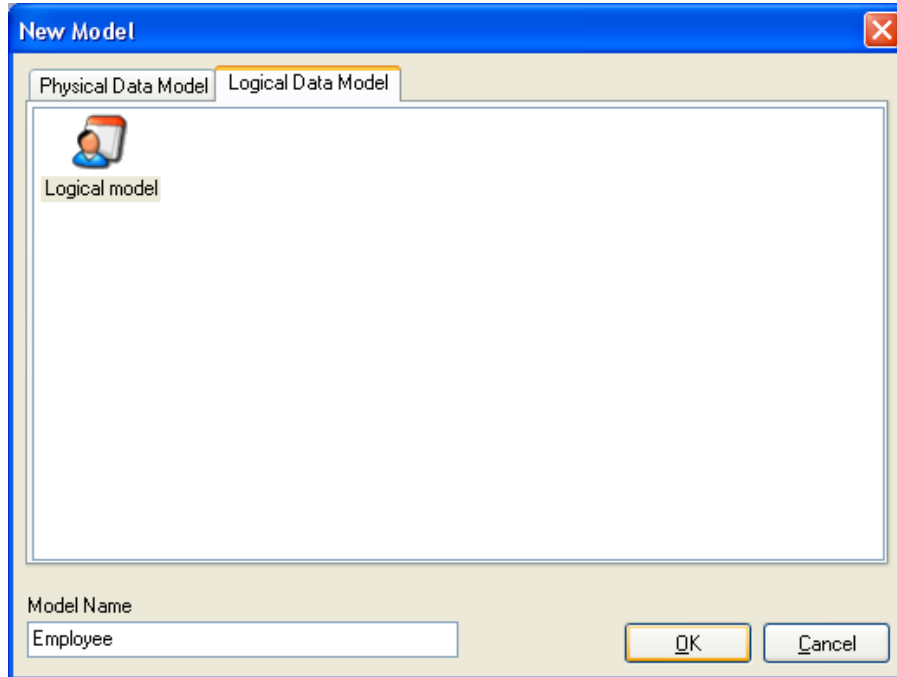
### Create Logical Model

#### Scenario

Create a logical model *Employee*.

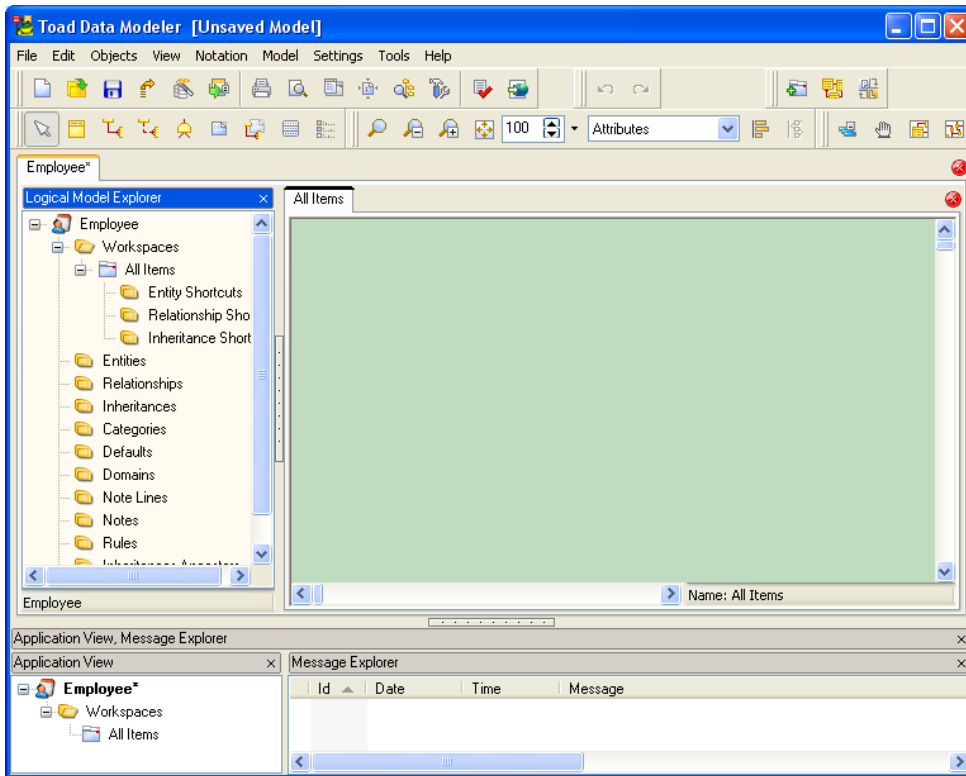
1. Click  on the toolbar (also CTRL+N).
2. On tab **Logical Data Model**, select **Logical model**.

3. In the **Model Name** box, define the model name *Employee* and confirm **OK**.



Result:

- The new logical model item is listed in the **Application View**.
- A designer for Workspace *All Items* of the new logical model will open in the **Application Window**.
- Main menu and the Designer toolbar for logical model will activate.



## Related Topics


Add Entities (page 204)

## Add Entities

In Toad Data Modeler, there are several ways how to create entities - on the Workspace, via **Model Explorer** and in the **Entities** dialog (**Model | Entities**). All the options are described in the Help file. Here, you will create entities directly on the Workspace.

### Scenario

Create entity *Department* on the Workspace in your *Employee* model.

1. Click  on the toolbar (or press CTRL + E) and then click anywhere on the work area.
2. Double-click the entity to edit it.
3. On tab **General**, define the entity name *Department*.

Caption	Logical entity name
---------	---------------------



Name	Physical entity name
<b>Note:</b> Appropriate entity name will be displayed on the Workspace according to the view you select ( <b>View</b> menu   <b>Show Logical Names</b> ).	
Logical Only	Select this option if you want to ignore the entity during conversion to physical model. In this example, leave it clear.

**Tip:** You can write description on the entity on the **Description** tab. To display the description directly in your logical ER diagram, select **Descriptions** in the **Display Level** box.

4. Define other properties and when you finish, confirm **Apply**.

The screenshot shows the 'Entity Properties' dialog box for the 'Department' entity. The 'General' tab is active. The 'Caption' field is set to 'Department' and the 'Name' field is also set to 'Department'. The 'Category' is set to 'Without Category' and the 'Size' is set to '0'. The 'Nature' field is empty. The 'Logical Only' checkbox is unchecked. The dialog has buttons for OK, Cancel, Apply, and Help.

## Related Topics

Add Attributes (page 205)

## Add Attributes

### Scenario

Create attribute *Department ID* in the *Department* entity.

1. Click the **Attributes** tab in the **Department Entity Properties** form.
2. Click **Add**.

3. Confirm **Apply**.
4. Click **Edit** or double-click the new attribute.
5. Define properties of the attribute.

The screenshot shows the 'Attribute Properties' dialog box for the 'Department ID' attribute. The 'Entity' is 'Department' and the attribute is 'Department ID'. The 'General' tab is selected, showing fields for 'Caption' (Department ID), 'Name' (Department ID), 'Data Type' (Integer), 'Domain' (empty), and 'Default Rule' (No default). Checkboxes for 'Logical Only', 'Mandatory', and 'Identity' are present, with 'Mandatory' checked. Buttons at the bottom include 'OK', 'Ok+Add', 'Cancel', 'Apply', and 'Help'.

## Related Topics

Create Unique Identifier (page 206)

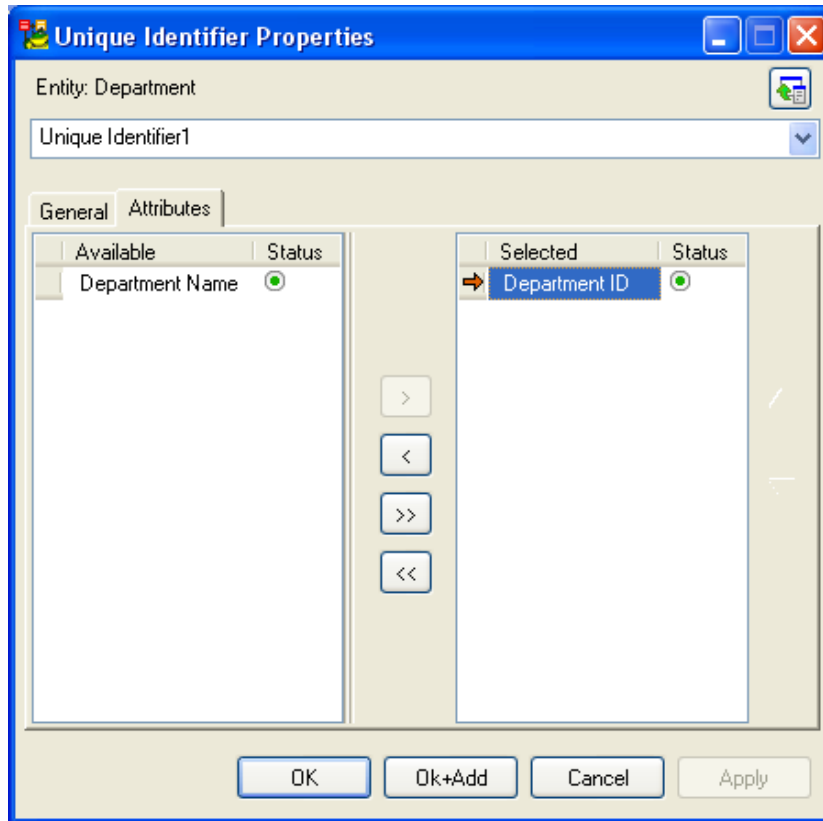
## Create Unique Identifier

### Scenario

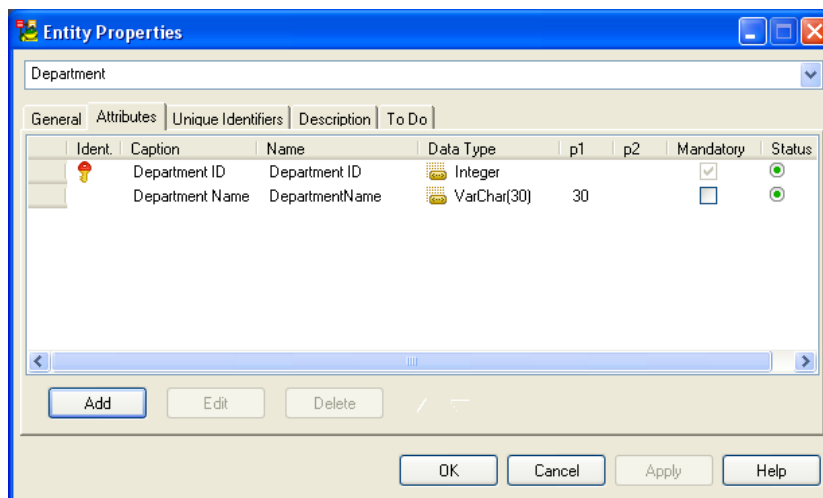
Assign a unique identifier to attribute *Department ID*.

1. In the **Department Entity Properties** form, click the **Unique Identifiers** tab.
2. By default, there's a *Unique Identifier1* that is empty.
3. Double-click it to edit it.

4. Define properties on tab **General** - Name and Caption - *Unique Identifier1*.
5. Click the **Attributes** tab.
6. Select attribute *Department ID* and click the **Add** arrow button.



7. Confirm **OK**. See the **Attributes** tab of the *Department* entity now:




## Related Topics

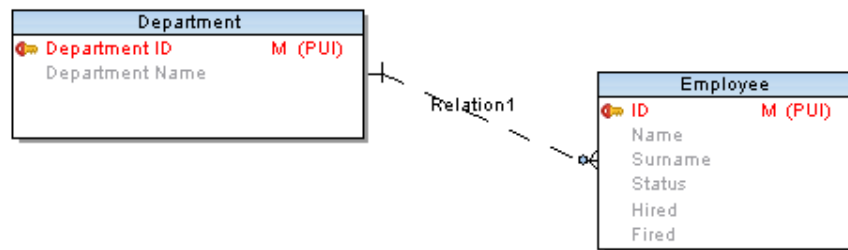
Create Relationship (page 208)

## Create Relationship

### Scenario

Create a non-identifying relationship between entities *Department* and *Employee*.

1. Click  on the toolbar.
2. Move your mouse cursor over the work area.
3. Click the *Department* entity (parent) and then the *Employee* entity (child).



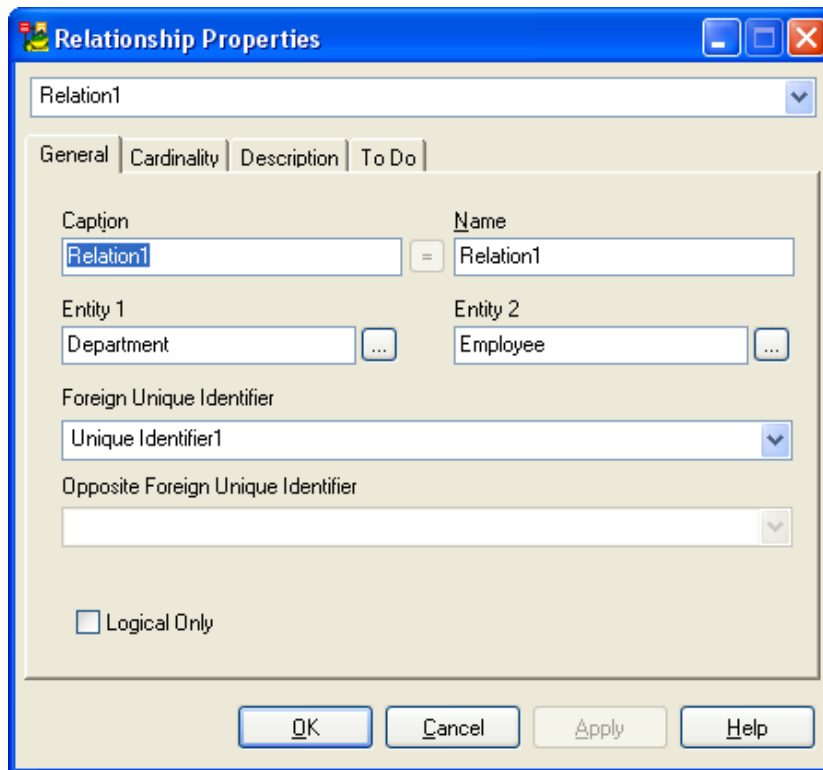
Keys do NOT migrate in Logical models when you create a relationship, but only in Physical models. In Logical model, only logical information is available. The relationships have only a logical meaning (therefore keys do not migrate). However, in Physical model the information on foreign keys is necessary. Therefore when you convert the Logical model to Physical model, FKs will be displayed properly in the physical model. It is done automatically and there is not any way to avoid doing this automatically.

## Related Topics

Specify Relationship Properties (page 208)

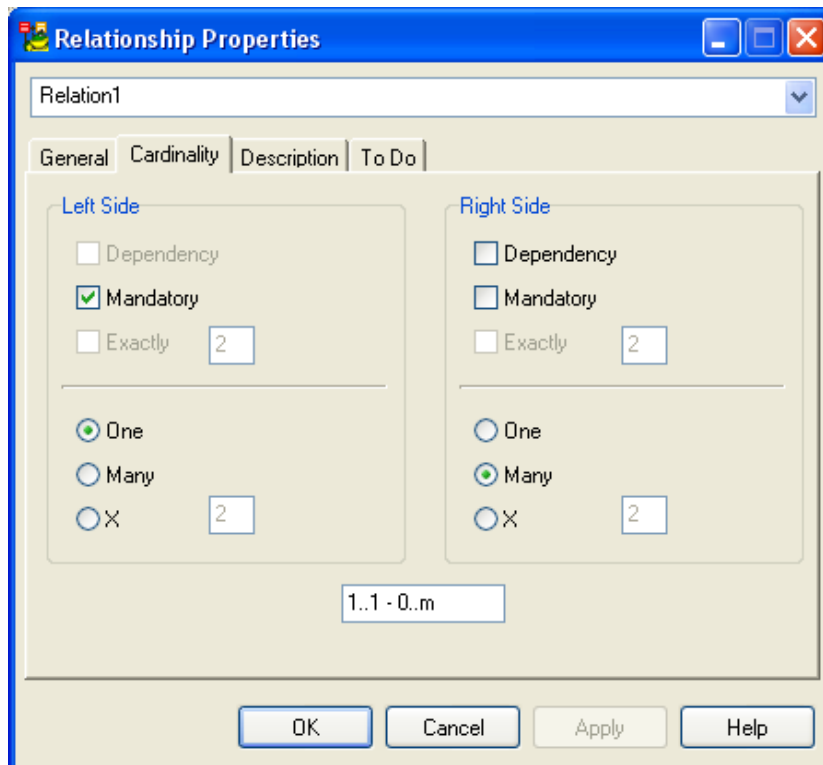
## Specify Relationship Properties

1. Double-click the relationship on the Workspace to open the **Relationship Properties** dialog.
2. On tab **General**, define physical and logical name of the relationship.
3. In the **Foreign Unique Identifier** box, you can select linking method for the relationship. (According to your selection, the LER model will be converted to PER model.) Select *Unique Identifier1*.



The 'Relationship Properties' dialog box is shown with the 'General' tab selected. The 'Relation1' dropdown is at the top. Below it are tabs for 'General', 'Cardinality', 'Description', and 'To Do'. The 'Caption' field contains 'Relation1' and the 'Name' field also contains 'Relation1'. 'Entity 1' is 'Department' and 'Entity 2' is 'Employee'. The 'Foreign Unique Identifier' is 'Unique Identifier1' and the 'Opposite Foreign Unique Identifier' is empty. The 'Logical Only' checkbox is unchecked. At the bottom are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

4. Set up cardinality on the **Cardinality** tab.



The 'Relationship Properties' dialog box is shown with the 'Cardinality' tab selected. The 'Relation1' dropdown is at the top. Below it are tabs for 'General', 'Cardinality', 'Description', and 'To Do'. The 'Left Side' and 'Right Side' sections are visible. In the 'Left Side', 'Mandatory' is checked, 'Exactly' is unchecked, and 'One' is selected. In the 'Right Side', 'Mandatory' is unchecked, 'Exactly' is unchecked, and 'Many' is selected. Both sides have a '2' in the adjacent text box. At the bottom, the cardinality '1..1 - 0..m' is displayed. At the bottom are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

5. Confirm **OK**.



## Related Topics

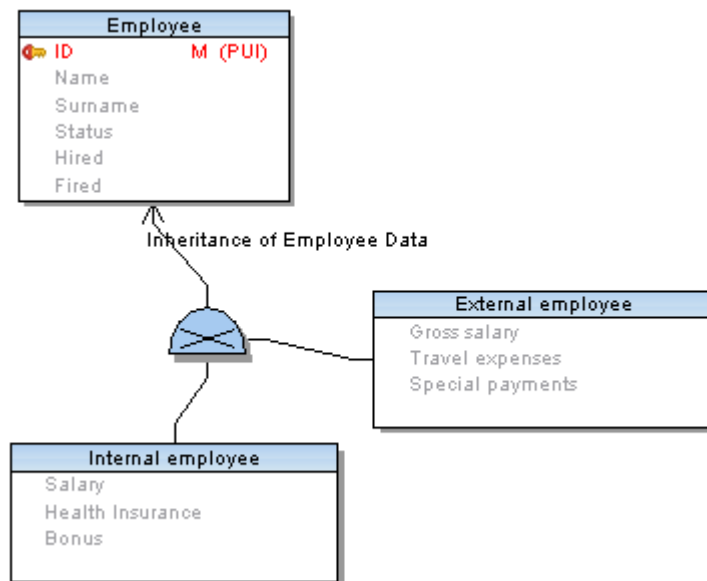
Create Inheritance (page 210)

## Create Inheritance

### Scenario

Create *Inheritance of Employee* data.

1. Click  on the toolbar (also CTRL+I).
2. Move your mouse cursor over the work area.
3. Click the *Employee* entity (ancestor) and then the *Internal Employee* entity (descendent).
4. To create another child of inheritance, click  on the toolbar again and then click the **Inheritance** on the Workspace and then the other descendent entity *External Employee*.



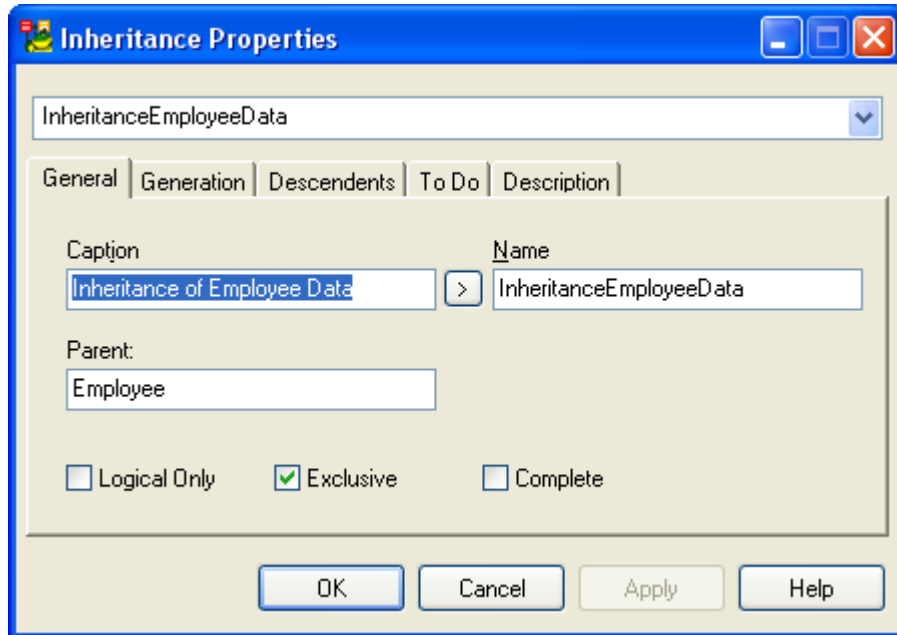
**Tip:** As well as for relationships, you can add handle points to inheritance, make only vertical and horizontal lines, and add multiple inheritances via SHIFT key.

## Related Topics

Specify Inheritance Properties (page 211)

## Specify Inheritance Properties

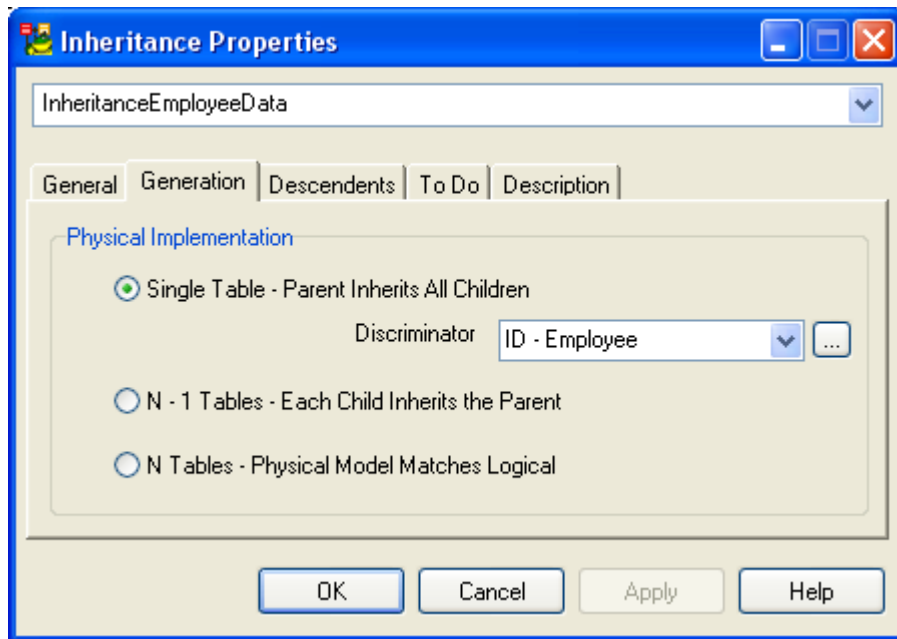
1. Double-click the inheritance on the Workspace.
2. Define physical and logical name of inheritance.
3. See the name of the parent entity in the **Parent** box.



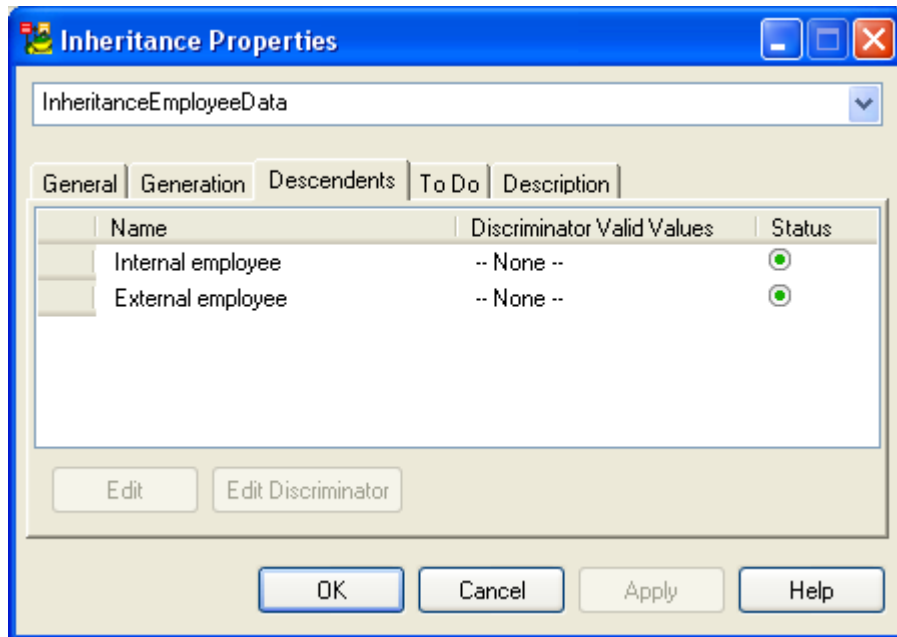
4. Select other properties.

Logical Only	Clear this checkbox not to ignore the inheritance during conversion to physical model.
Exclusive	Select this checkbox to set the inheritance exclusive. <b>Note:</b> For exclusive inheritances, Toad Data Modeler generates triggers that will perform check whether a correct record in siblings exists or not, and decide whether a record can be added to table or not etc. Exclusive inheritances are displayed with cross in the middle of the graphics.
Complete	This is just logical information that says that all records are complete.

5. Click tab **Generation**. On this tab, select how you want to convert inheritance during conversion of logical model to physical model. Select the first option.



6. Click tab **Descendents** to see the descendents of the inheritance.

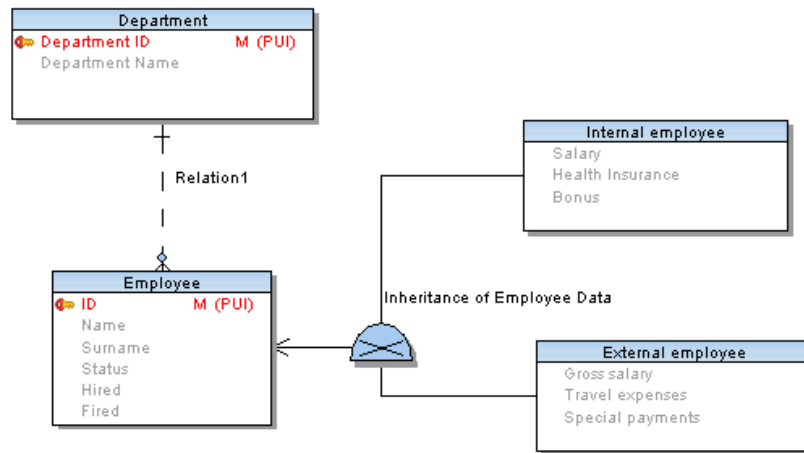


**Note:** Here, you can also define discriminators and edit their default values.

7. Confirm the changes **OK**.

Result:



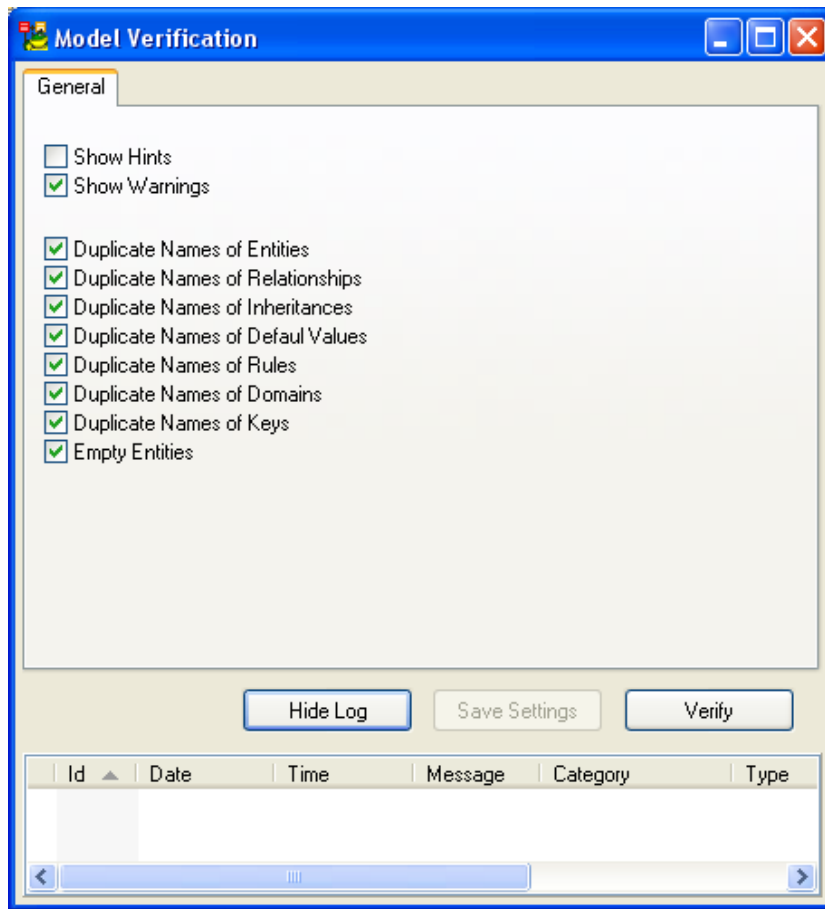


## Verification

### Model Verification

Check out your logical model and see what errors, warnings and hints the verification returns.

1. Select **Select Model | Verify Model** (also CTRL+F9).
2. Select options for verification. To save the settings, click **Save Settings**.
3. To see the process of verification and final result, click **Show Log**.
4. Click **Verify**.




## Report Generation

### Report Generation

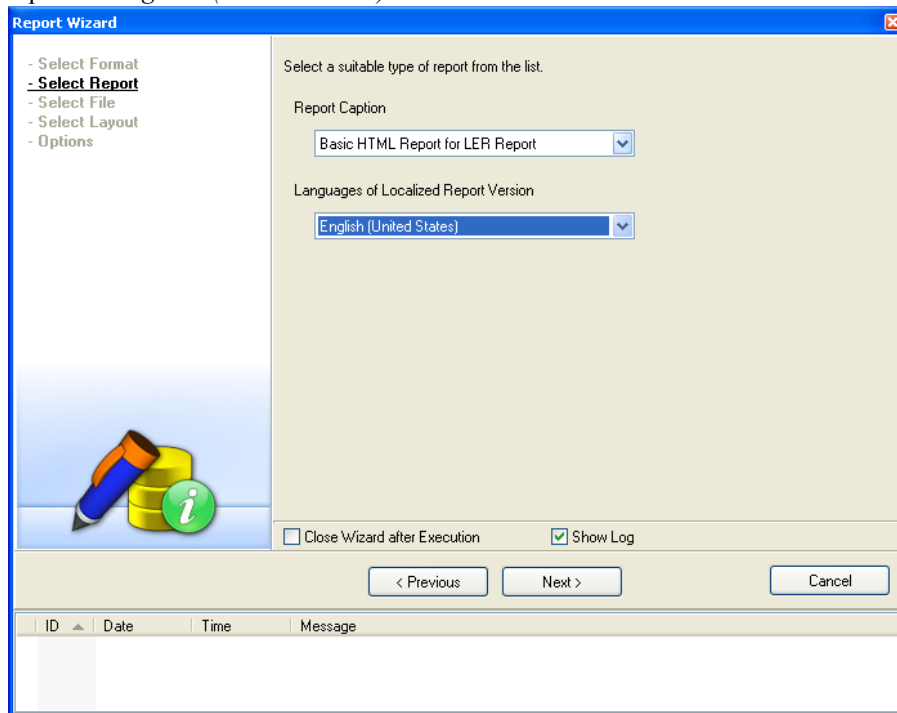
Toad Data Modeler enables you to generate HTML, RTF or PDF reports for your logical models.

#### Scenario

You want to generate HTML report for your *Employee* model.

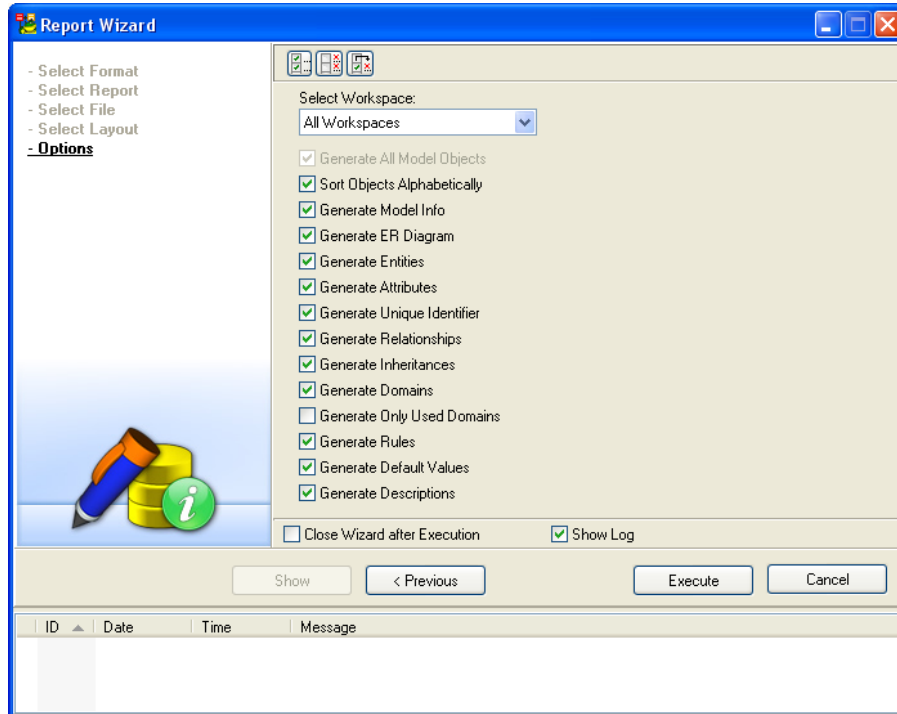
1. Click  on the toolbar.
2. On the **Select Format** page, select *HTML*.
3. On page **Select Report**, from the **Report Caption** box, select a report type *Basic HTML Report for LER Model*.

4. From the **Languages** box, select the language in which you want to have the report - *English (United States)*.



5. Select a file where you want to generate the report. By default, the path defined in the **Settings** menu | **Options** | **Application** | **Paths** | **Path to HTML/RTF Reports** is set up here.
6. On the **Select Layout** page, select a frame and style of your HTML report.

7. On the **Options** page, select options for the report generation.



8. Click **Execute**. To see the generation process, select **Show Log**.
9. As soon as the report generation is finished, the following message will display in the Wizard: 'Report was generated.' Confirm the message **OK**.
10. Click **Show** to see the final report.
11. Click **Cancel** or x to close the **Report Wizard**.


## Model Compare

### LER Model Compare

Toad Data Modeler allows you to compare your logical models - via the **Sync & Convert Wizard**.

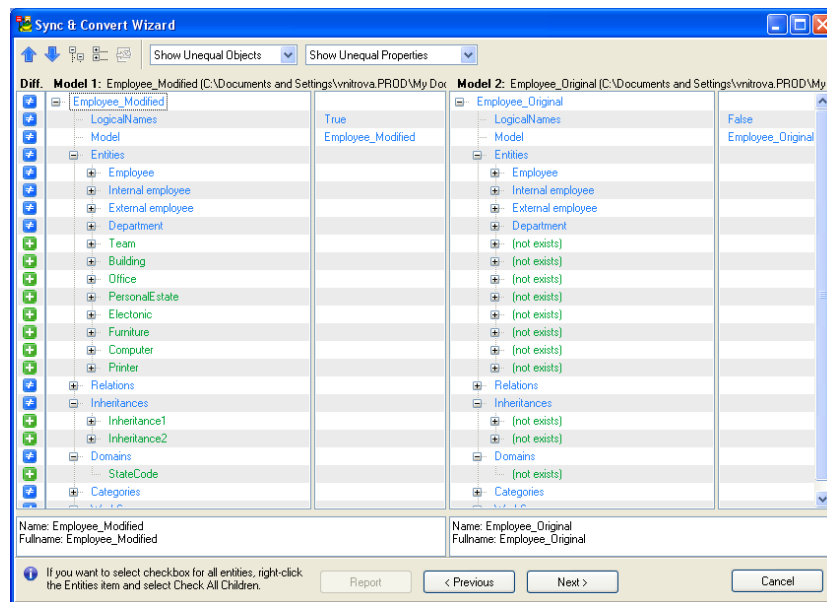
#### Scenario

You want to compare logical models *Employee\_Original* and *Employee\_Modified*.

1. Click  on the toolbar to open the **Sync & Convert Wizard**.
2. On the **Action** page, select **Compare Models and Generate Alter Report**.

3. On the **Select Left Side** page, click **File** and then the button next to the **File** box and find the model *Employee\_Modified*.
4. On the **Select Right Side** page, click **Another Physical Model File**, click the button and find the model *Employee\_Original*.
5. On the **Settings** page, select e.g. **Ignore Text Case**.
6. On the **Select Object Types** page, select **Compare All** and leave default selection of OTPs.
7. On the **Select Items** page, you can see differences between the models.

**Note:** Alter reports can be generated only for physical models.



The blue icon and the blue color say: "There is a difference."

The green icon and the green color say: "This item exists in Left model and is missing in Right model."

8. Click **Next** to get to page **Review** where you can see the statistic information.
9. Close the wizard.

## Model Update/Model Merge

### LER Model Merge

Toad Data Modeler allows you to see differences between your logical models and merge them - one to another or to a new model.

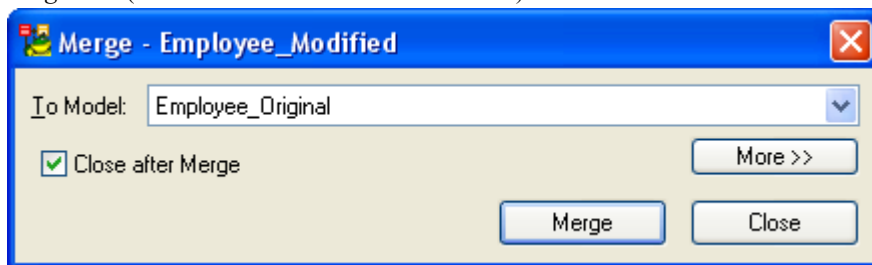
Toad Data Modeler offers you two options to merge your models:

- **Simple Model Merge** - allows you to merge your models very quickly but without a possibility to select what particular items to merge. The target model will always be overwritten.
- **Model Merge in the Sync & Convert Wizard** - allows you to see differences between two models, select particular items to merge and merge the models either to already existing model or into a new model.

## Scenario

You want to merge models *Employee\_Modified* and *Employee\_Original* to apply the changes and update model *Employee\_Original*. You decide to use the Simple Model Merge feature.

1. Open both models.
2. Activate model *Employee\_Modified*.
3. Select **File | Sync & Convert | Simple Model Merge**.
4. From the **To Model** box, select a model that you want to update - *Employee\_Original*. (This model will be overwritten.)



5. Select the **Close after Merge** checkbox to close the dialog automatically after the model merge.
6. Click **Merge**.

Model Merge (page 189)

## Model Conversion

### Information on LER - PER Conversion

Toad Data Modeler allows you to convert logical model to physical model for selected database system. For this purpose, you can use either the **Sync & Convert Wizard** or **Simple Model Conversion** feature (**File | Sync & Convert**).

Before you continue converting your LER model to PER model, you should know several information. Particularly the following:

- Physical model supports only non-identifying self relationship.
- Inheritance is not supported in PER model.
- Keys in LER models do not migrate.
- You can select a linking method in LER model.
- M:N relationships are supported in both models.
- Before you start the conversion, you can set up the conversion rules in the **Data Type Conversion Settings** dialog.

**Note:** This option is available only if Expert Mode is enabled.

For more details on LER to PER conversion, please see the Help file, "Model Conversion" chapter and "Logical Data Model" - "Conversion to Physical Model".

## Related Topics

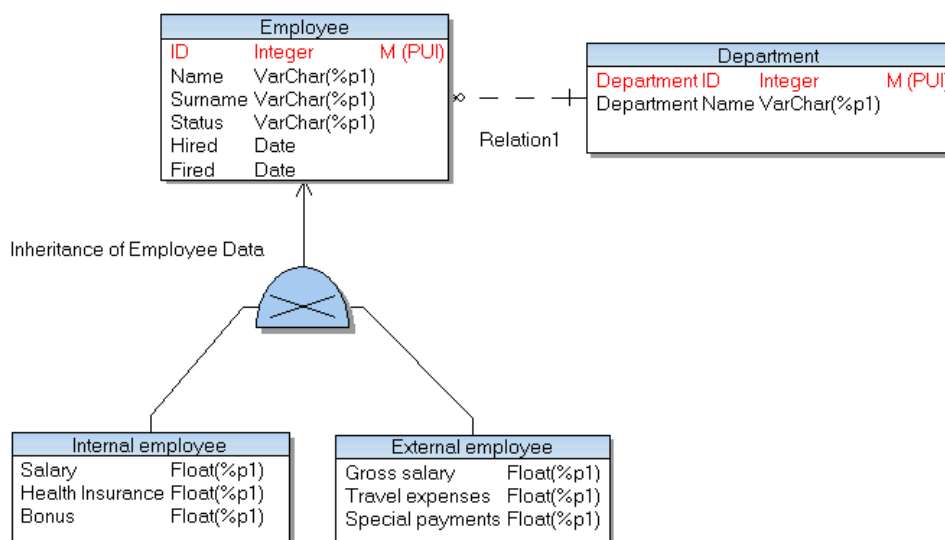
Conversion LER - PER (page 219)

## Conversion LER - PER

### Scenario

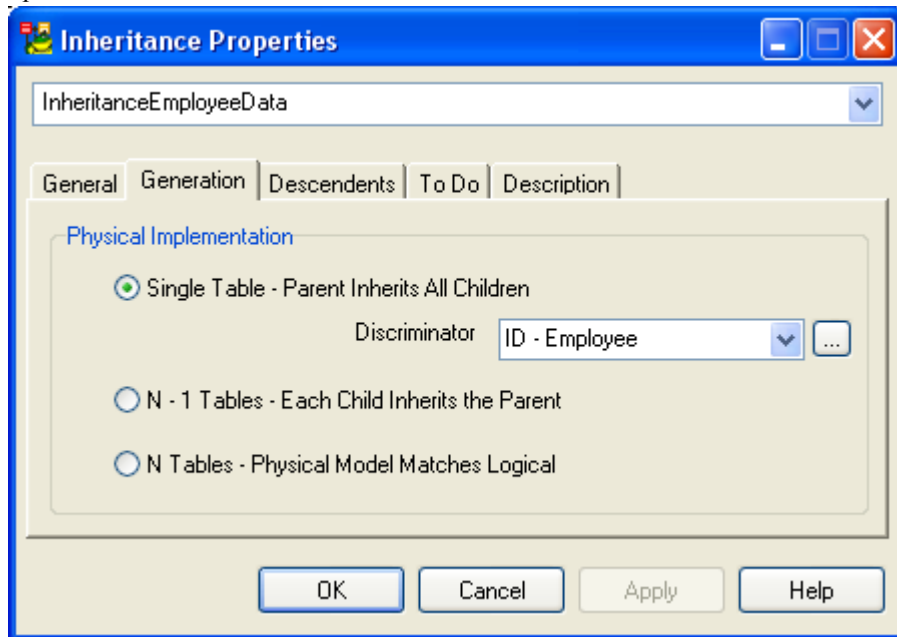
You want to convert Logical model *Employee* to PER model for Oracle 10g. Use the Sync & Convert Wizard. Model *Employee* is open and active.


This is how the logical model *Employees* looks like:



**Scenario:** Select a method for inheritance conversion.

1. Edit the inheritance and select **Single Table - Parent Inherits All Children** option.



2. Click  on the toolbar.
3. On the **Action** page, select **Convert Model to Another Target Database System**.
4. On the **Select Left Side** page, **Available Model** box, select the logical model *Employee*.
5. On the **Select Right Side** page, select a target database system *Oracle 10g*.
6. On the **Settings** page, you can select from the following options:

#### **Comment Out Database Specific Items**

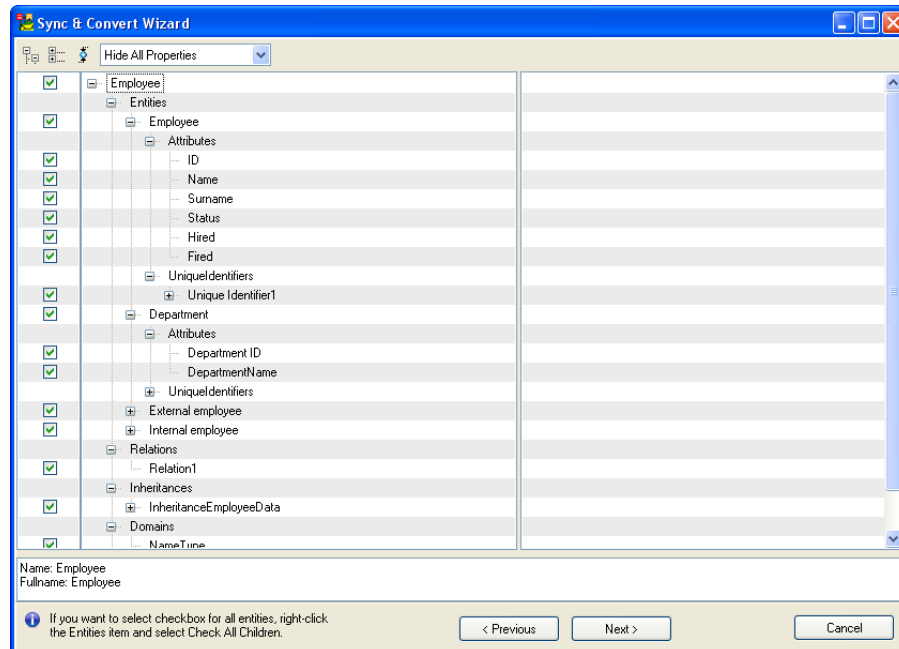
**Log Progress to File** - Select this checkbox to save the messages that the Wizard returns to a file.

Select both checkboxes.

7. On the **Select Object Types** page, select **Convert All**.
8. On the **Select Items** page, you can select/deselect particular items of your model that you want or do not want to convert. Leave the default selection to convert all items of the

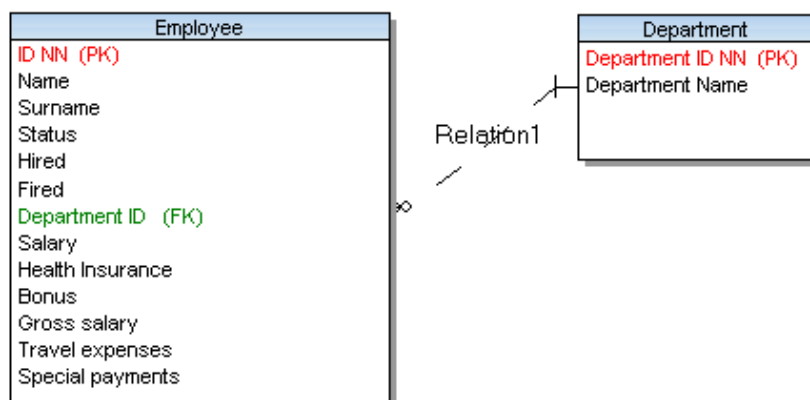


model.



9. On the **Review** page, see the statistic information and define a model name for the converted model in the **New Model Name** box - *Employee\_ConvertedLER*.
10. Click **Finish**.
11. The **Sync & Convert Wizard** closes and the new model *Employee\_ConvertedLER* opens in the Application Window automatically.

Result:



## Change the Look and Save Models

### Rename

*To rename your model name*

- » Right-click the model name in **Model Explorer** (or **Application View**) and select **Rename**.

*To rename objects on Workspace*

1. Select the object (entity, view etc.), press CTRL and double-click the object name.
2. Write a new name.
3. Click the Workspace then.

*To rename object in Model Explorer*


- » Select the object and press F2 or right-click and select **Rename**.

### Related Topics

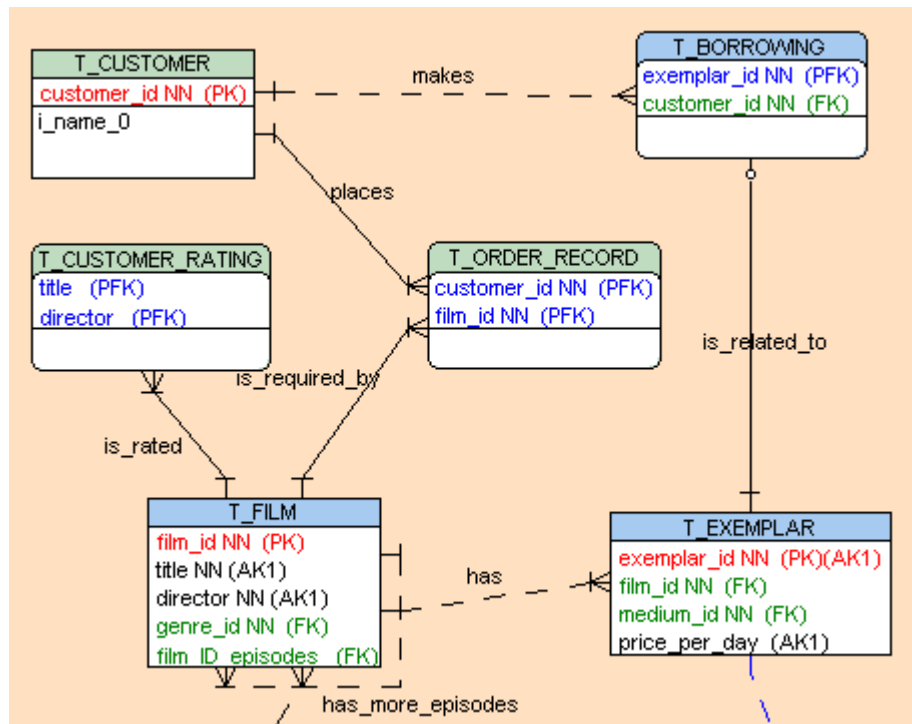
Inplace Editor (page 41)

### Display Modes

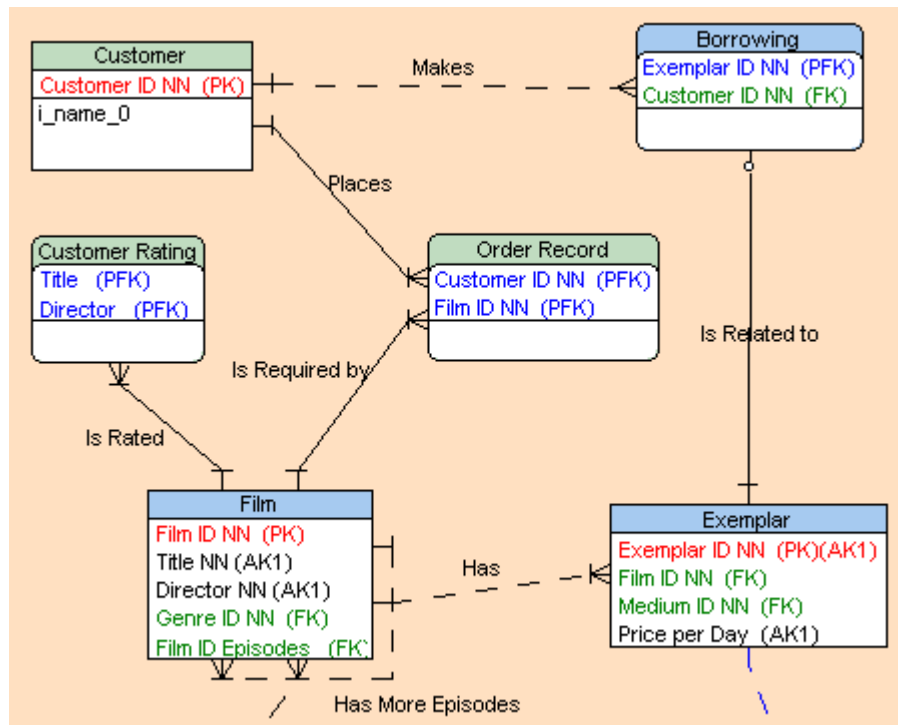
In Toad Data Modeler, you can switch between a logical and physical view.

- » Click  on the toolbar (also **View | Show Logical Names**).

Physical View



Logical View

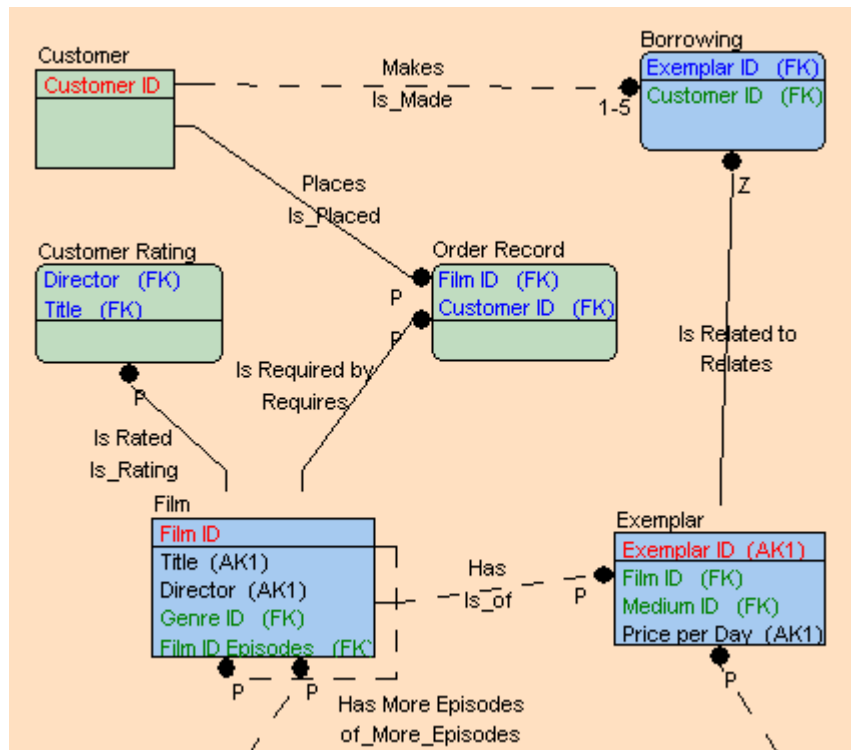


## Change Notation

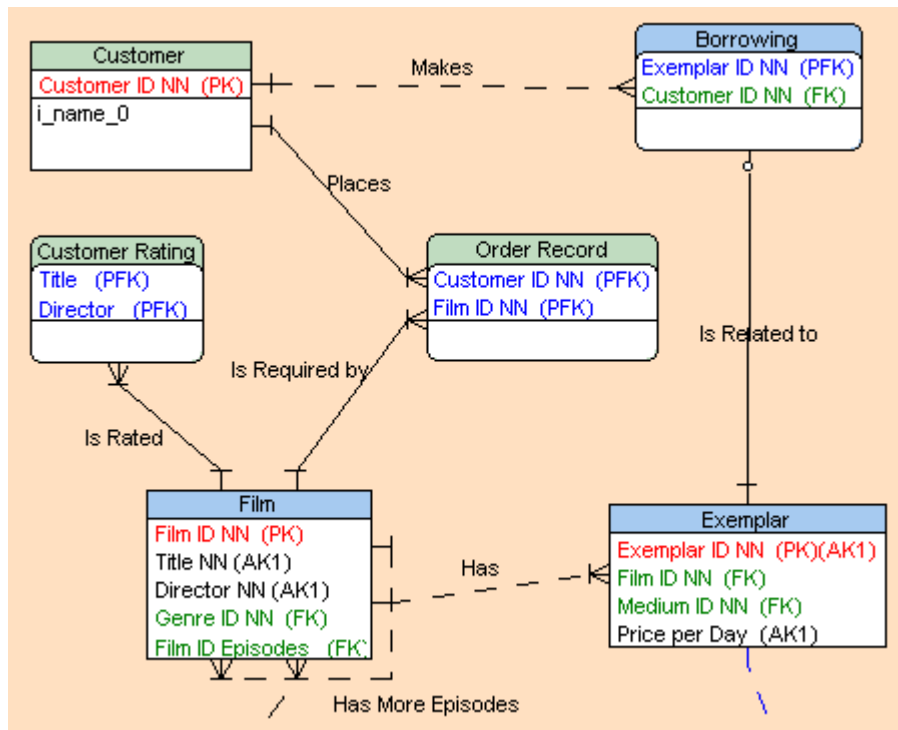
*To change or select notation for your model*

» Select **Notation** | **IE** or **IDEF1X**.

IDEF1X



IE



## Format Workspaces and Objects

You can set up format for all Workspaces of your model, for each Workspace separately and also for particular objects.

### *To set up format for new models (models that you will create)*

1. Select **Settings | Options | Model section | Physical/Logical Model**.
2. Define options on tabs **Workspace**, **Shape** and **Entity**.
3. Press CTRL+N to create a new model.

### *To change format of objects in existing models*

- » Right-click the Workspace and select **Workspace Format**.

### *To change format of a particular object*

- » Right-click the object on Workspace and select **Format**.

**Tip:** If you need to preserve format of a particular object against changes of format of your WS, select the **Lock Format** option in the **Object Format** dialog | **General** tab.

## Display Level for Entities

Toad Data Modeler allows you to display entities on Workspaces in several levels:

### PER Model

- Entities
- Primary Keys
- PK and FK keys
- All Keys
- Attributes

### LER Model

- Entities
- Primary Identifiers
- Unique Identifiers
- Attributes
- Descriptions - Text written on the **Description** tab of entity will be displayed on the WS.

*To set up default display level for new model (models that you will create)*

- » Select **Settings** | **Options** | **Physical/Logical Model** | **Entity** tab.

*To set up default display level for the selected Workspace*

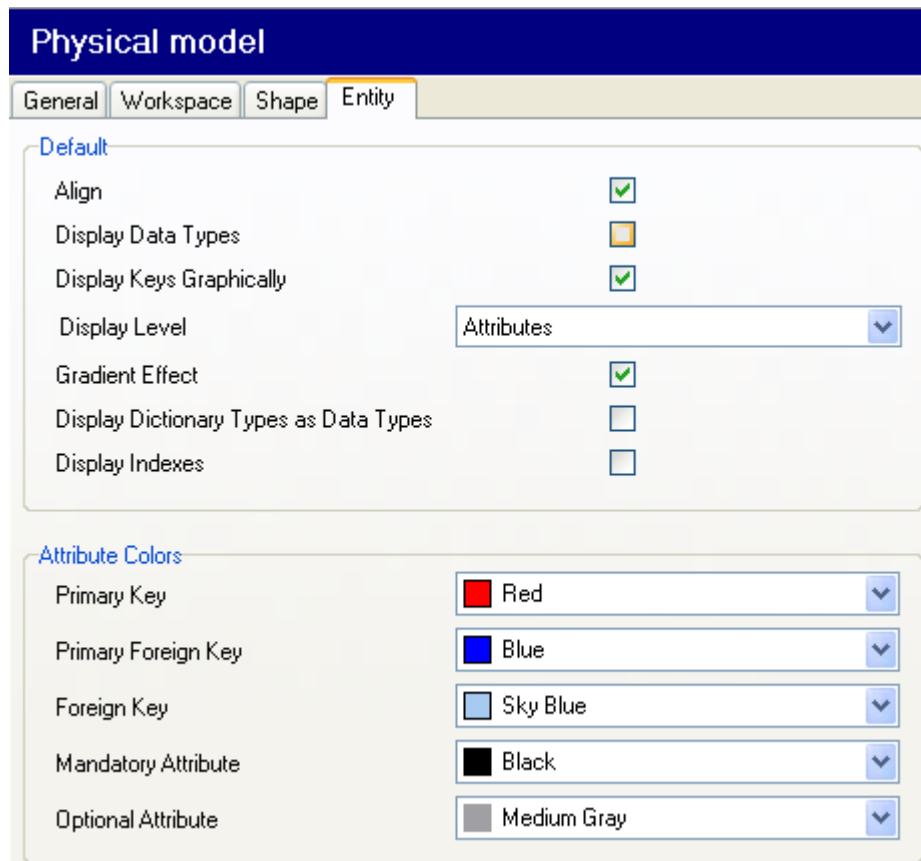
- » Change the display level from the **Display Level** box on the toolbar.

**Note:** Also right-click the work area | **Workspace Format** | **Entities** tab | **Display Level**.

## Select Colors for Attributes

*To define different colors for attributes on your Workspace*




- » Select **Settings** | **Options** | **Model** section | **Physical Model/Logical Model** | **Entity** tab | **Attribute Colors** area.

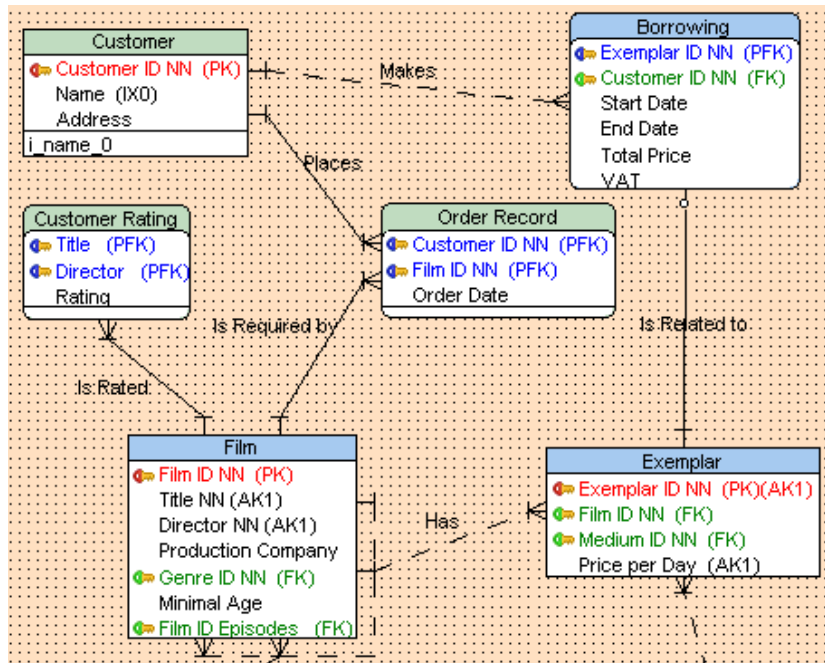


## Show Grid and Grid Size

The **Grid** options are available directly from the toolbar or from the **View** menu.


### *To show grid and set a grid size*

1. Click  to show grid.
2. To change a grid size, click .
3. To snap objects of your ERD to grid, click .



## Align Objects

### *To align objects on the Workspace*

1. Select objects you want to align.
2. Click  on the toolbar (also select **Edit | Align**).
3. Select the way to align the objects.

### *To align a self relationship*

- » Right-click the entity and select **Align Self Relationship**.

### *To align handle points of relationship*

1. Select the handle points (use ALT key to select a line segment, use SHIFT key to select more line segments).
2. Select **Edit | Align Handle Points**.

## Related Topics

Arrange Relationship Lines (page 229)



## Arrange Relationship Lines

In Toad Data Modeler, you can manage relationship lines in two modes:

- Straight mode
- Right Angled mode

### Straight Mode

In the Straight mode, you can:

- Add, move, remove and align handle points
- Switch between straight and right-angled lines

#### *To add handle points on the relationship line*

1. Select the relationship.
2. Press CTRL, hold it down and click the relationship line where you want to add the handle point.

#### *To move a handle point*

1. Select the relationship.
2. Point the mouse cursor at the handle point until a small black arrow displays.
3. Drag and drop it.

#### *To remove a handle point*

1. Select the relationship.
2. Press CTRL, hold it down and point your mouse cursor at a point you want to remove. -> The mouse cursor changes to a small black arrow.
3. Click the selected point to remove it.

#### *To align handle points*

1. Select the handle points (use ALT key to select a line segment, use SHIFT key to select more line segments).
2. Select **Edit | Align Handle Points**.

#### *To change the mode of relationship lines on particular Workspace*

- » Right-click the selected Workspace and click **Straight All Lines** or **Change Lines to Right Angled**.

*To change the mode of the selected relationship line on the Workspace*

- » Right-click the selected relationship line(s) and click **Straight Line** or **Change Line to Right Angled**.

## Right Angled Mode

In the Right-angled mode:

- You can add and remove handle points.
- You cannot move handle points (but can move the selected line segment).
- You cannot change the right-angled lines to straight lines (but can change the straight lines to right-angled).

## Move Relationship Anchor Points

Relationship automatically links the closest sides of entities. Anchor is always placed in the middle of the entity box side and it's not possible to alter it.

You can only remove an anchor point from one entity to another or from one entity shortcut to another. For this purpose, use the drag&drop techniques.

## Move Relationship Lines

*To move an entity with its relationship, including all handle points*

- » Select the entity and the appropriate relationship (use SHIFT) and drag and drop the entity.

Result: The selected entity, relationship and its handle points will move too. Position of handle points will be preserved.

*To move only the selected line segment*

- » Press ALT key, hold it down, select the line segment and move it by the drag and drop techniques or via the keyboard arrows.

**Note:** This option works only for angled lines (however, in both Straight and Right-angled modes).

## Move, Hide, Find Relationship Names

*To move the name of relationship on the Workspace*

- » Right-click the selected relationship in particular place and select **Move Caption Here**.

### *To hide relationship names on the Workspace*

- » Right-click the Workspace | **Workspace Format** | **General** tab | clear the **Display Line Names** checkbox.

### *To find the relationship line by its name on the Workspace*

- » Click the relationship name (caption). The appropriate relationship line will be highlighted on the Workspace by the blue color.

### *To find the relationship name (caption) by its line on the Workspace*

- » Click the relationship line, its name (caption) will be highlighted in a frame.

## Autolayout

To rearrange objects of your model automatically

- » Click  on the toolbar.

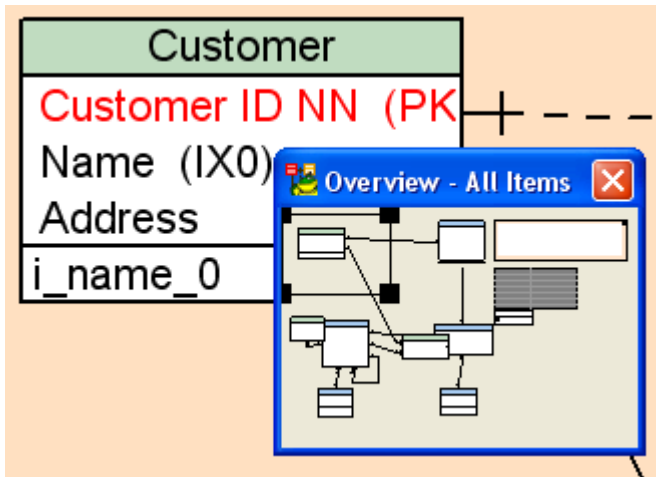
**Tip:** To set up a coefficient of distance between tables on the Workspace, select **Settings** | **Options** | **Graphics** | **Autolayout area**. (The lower the coefficient is, the smaller the distance between tables will be.)

## Zoom, Overview, Loupe, Move

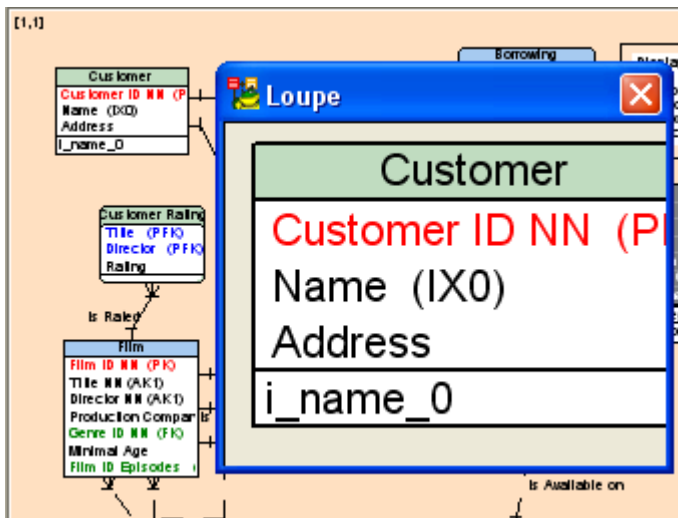
Toad Data Modeler offers you various view options for more comfortable work in your model - zoom in, zoom out, zoom box, overview, loupe and move.

All these options can be found on appropriate icons on the toolbar or in the **View** menu.

**Example:** Overview



Example: Loupe



## Tips

### Objects on the Workspace and Keyboard Arrows

- Move entities on the Workspace via keyboard arrows.

**Tip:** To set up the size of a step to move, select Settings | Options | Graphics | Move Objects by (mm/10) (in tenths of millimeters).

- Select an entity on the Workspace, press SHIFT, hold it down and use the keyboard arrows to change size of the entity box.

### Navigation on the Workspace

- CTRL + scroll mouse to zoom in/zoom out
- CTRL+, CTRL + Page Up to zoom in

- CTRL-, CTRL + Page Down to zoom out
- Scroll mouse to move up/down on the Workspace
- SHIFT + scroll mouse to move to the right/left on the Workspace
- Holding down the middle button to move on entire page/Workspace

## Categories

Toad Data Modeler allows you to create categories that colorfully distinguish parts of your model. You can assign entities that logically go together to one category and display them in a different color from others - on the Workspace and also in Model Explorer.

### Scenario

In your *Videorental* model, you want to distinguish entities that contain data on your customers. Create the *Customer info* category, select a color for it and then assign particular entities to the category.

**Scenario:** Create a category.

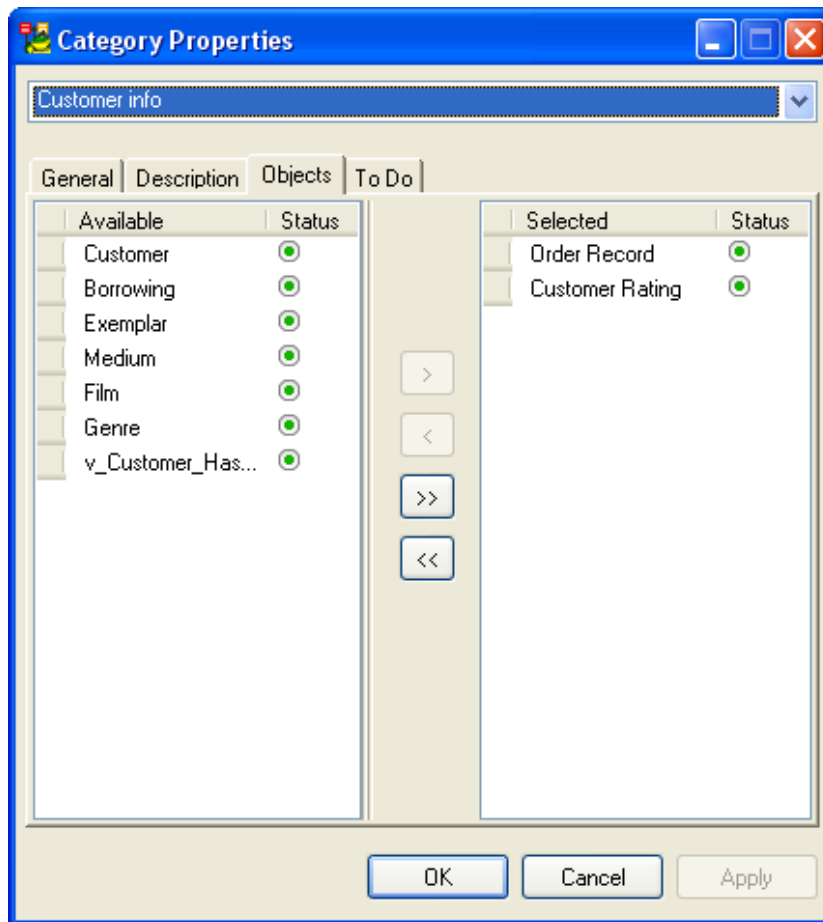
1. Select **Model | Categories** and click **Add** in the **Category List** dialog.
2. Confirm **Apply**.

**Scenario:** Select a color for the category.

3. Click **Edit**.
4. In the **Category Properties** dialog on tab **General**, write the name *Customer info* and select a color - *teal*.

**Scenario:** Select entities for the category.


5. In the **Category Properties** dialog, click the **Objects** tab.
6. Use SHIFT or CTRL key to make multiple selection of entities in window **Available** and click the **Add** arrow button to switch them to window **Selected**.
7. Confirm **OK**.



**Tips:**

- When you add a new entity to your model, you can assign the category to the entity directly in the **Entity Properties** form, tab **General**, **Category** box. Also, take notice of the small icon next to the box. Click it to access the **Category List** dialog where you can add a new category or edit/delete the existing ones.
- You can manage categories from Model Explorer too.

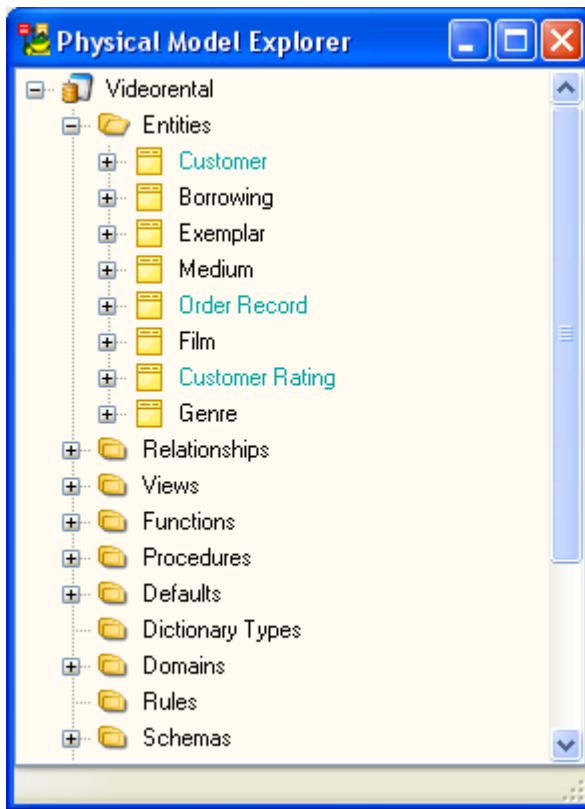
*To add caption of categories on the Workspace*

- » Click  on the toolbar and then click the Workspace.

*To display entities by category in Model Explorer*

- » Right-click the **Model Explorer** | **Settings** | and select **Use Colors of Category to Draw**.

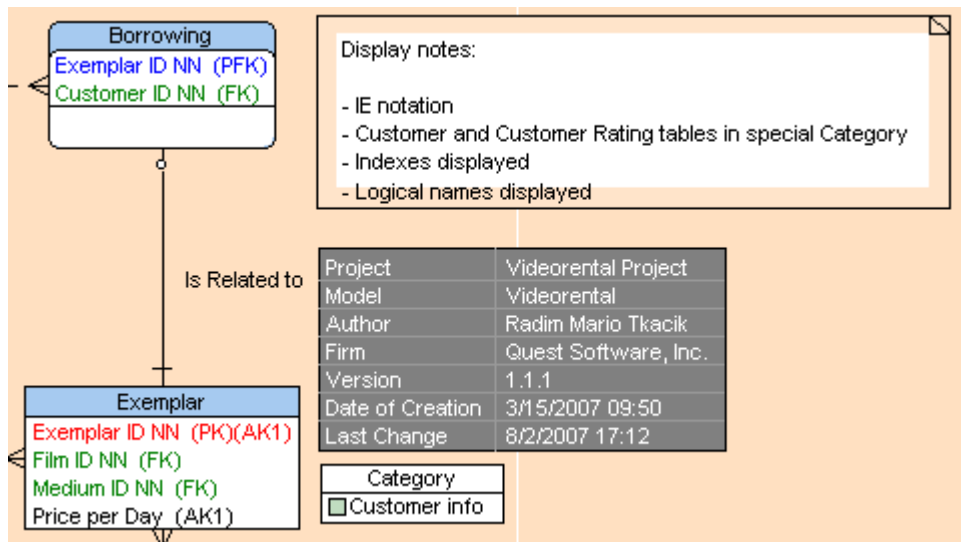
Result: Names of entities will be displayed in color of appropriate category.




## Add Notes and Note Lines

Toad Data Modeler enables you to place a note on Workspace or an object directly in ER diagram, and relate them via a note line.

**Example:**



#### *To add a note*


1. Click  on the toolbar and then click the work area where you want to place the note.
2. Double-click the note to edit it.

#### *To change format of notes*

- » Right-click the note and select **Format**.

**Tip:** You can define different background color, font type etc. for every note.

#### *To add a note line*

1. Click  on the toolbar.
2. Click the object and note that you want to relate together.

**Note:** Again, you can change format of note line at your convenience.

## Add Stamp

To display model properties on your model such as project name, author, company, version, date of creation etc. on the Workspace, you can add a stamp there.

#### *To add a stamp on the Workspace*

- » Click  on the toolbar and click the work area.



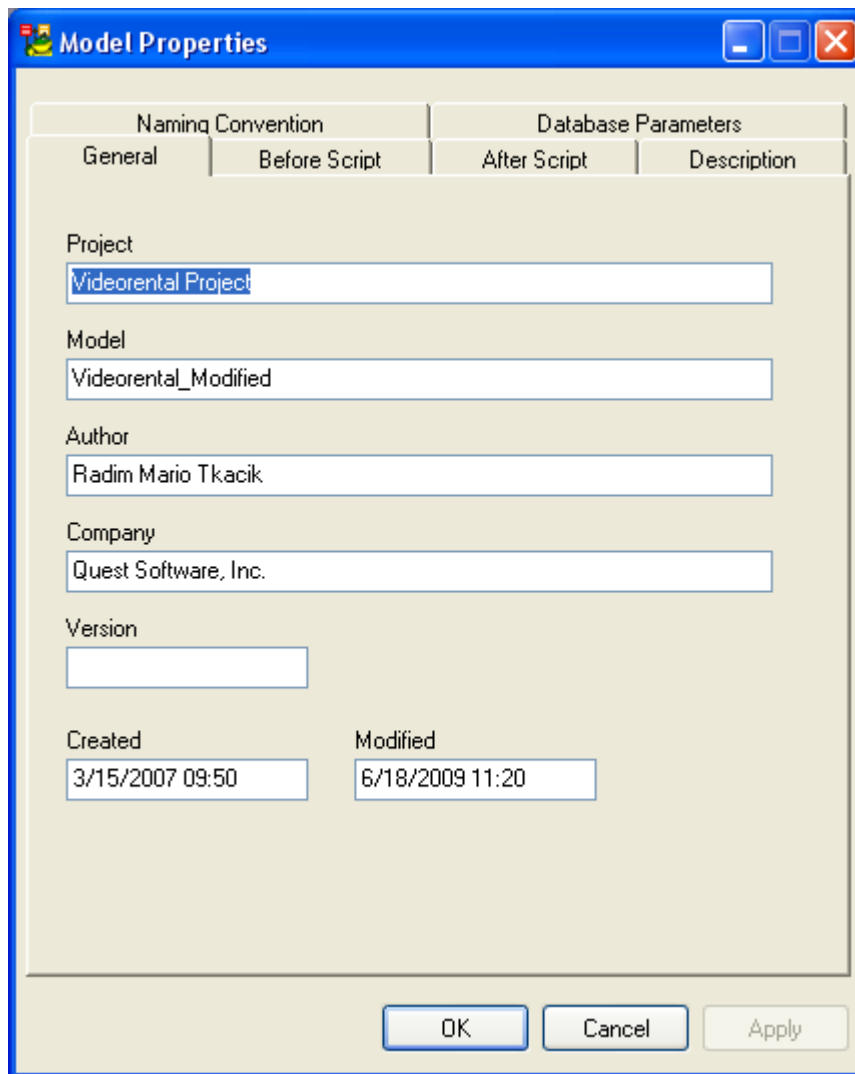
Project	Videorental Project
Model	Videorental
Author	Radim Mario Tkacik
Firm	Quest Software, Inc.
Version	
Date of Creation	3/15/2007 08:50
Last Change	2/5/2008 13:01

*To change format of Stamp*

- » Right-click the Stamp and select **Format**.

*To edit the Stamp*

- » Double-click the stamp to change the properties.



The image shows a 'Model Properties' dialog box with a blue title bar and standard Windows window controls. It features two main tabs: 'Naming Convention' and 'Database Parameters'. The 'Naming Convention' tab is active and contains four sub-tabs: 'General', 'Before Script', 'After Script', and 'Description'. The 'General' sub-tab is selected, displaying several text input fields for model metadata: 'Project' (Videorental Project), 'Model' (Videorental\_Modified), 'Author' (Radim Mario Tkacik), 'Company' (Quest Software, Inc.), and 'Version' (empty). At the bottom of this section are two date/time fields: 'Created' (3/15/2007 09:50) and 'Modified' (6/18/2009 11:20). At the very bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Apply'.

**Tip:** Select **Model | Information about Model** to see some statistic information on your model, objects and particular workspaces of your model.

## Save Model

### *To save your model*

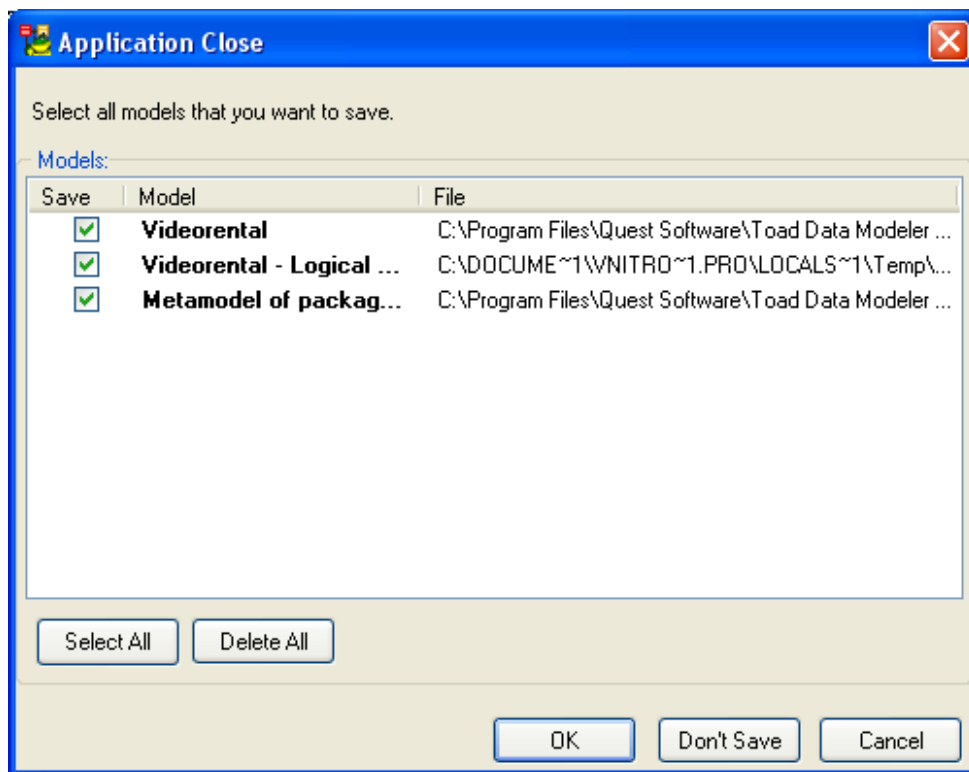
- » Select **File | Save** or **Save As**.

**Tip:** You can also save your models from the **Application View** or **Model Explorer**: Right-click your model name.

## Application Close Dialog

You can also close the application directly: Click **x** in the Application Window or select **File | Exit**.

After you select this option, the **Application Close** dialog will display:



In this dialog, all opened models are listed (opened in the Application View). Here, you can select what models should be saved, and also change location of models that have not been modified. Models where some changes were made are automatically selected to save.

Select the checkboxes of the models you want to save, or possibly change model locations, and confirm.




## Maintaining Models

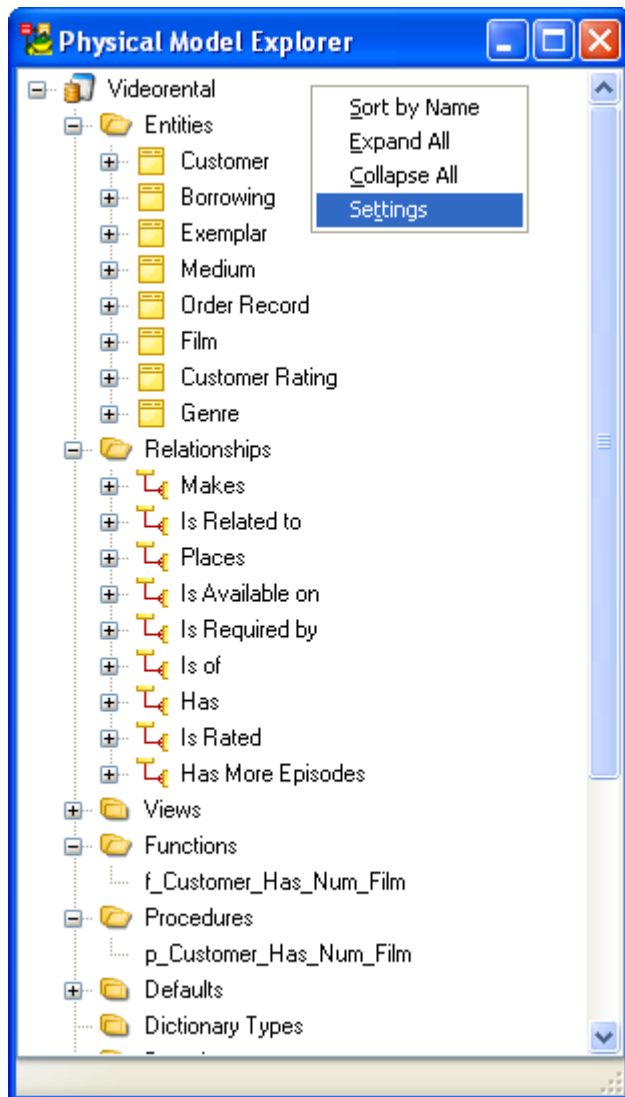
### Model Explorer

In Model Explorer, you can find all objects of your model , including dictionary types, domains, procedures etc. The content of Model Explorer depends on type of your model (PER, LER).

Model Explorer is by default docked on the left side of the Application Window. However, you can hide or close it.

*To display Model Explorer again*

- » Click **Model Explorer**  on the toolbar (or select **Tools | Physical/Logical Model Explorer**).



In Model Explorer, you can manage objects of your model easily:

- Manage model and particular Workspaces via the right-click options
- Manage and sort objects/items - Double-click the main item in Model Explorer opens appropriate dialog.

**Example:** Double-click the **Procedures** opens the **Procedures** dialog (same as when you select **Model | Procedures**). The same works for Entities, Relationships, Views, Functions, Users etc.

- Add, edit and delete
- Copy (CTRL + Drag&Drop techniques)
- Move (Drag&Drop techniques)
- Create shortcuts of entities (Drag&Drop techniques to the Workspace)

- Find shortcuts of objects on the Workspace quickly - Double-click the shortcut in Model Explorer highlights it on the Workspace.

## Examples of Use in Model Explorer

**Example 1:** Creating entities and defining their properties via Model Explorer.

1. Right-click the **Entities** folder and select **Add**. -> A new entity is created (*Entity1*)
2. Double-click the new entity to open its **Properties** dialog. Define its properties and confirm the changes **Apply**. -> The **Entity Properties** form remains opened.
3. Take the same steps in Model Explorer to create another entity (*Entity2*).
4. Click the still opened **Entity Properties** form. Unfold the Object Navigator box and select *Entity2* item.
5. Define its properties.

**Tip:** You can take the same steps while defining properties of any other item - attribute, procedure, schema etc. Leave the **Properties** dialog opened and use Model Explorer to add new items quickly and comfortably.

**Example 2:** Making a copy of attributes from Model Explorer to the Workspace.

1. Unfold the **Entities** folder | '*EntityName*' | **Attributes** tab in **Model Explorer** tree.
2. Make selection of attributes (use SHIFT or CTRL keys).
3. Press CTRL, hold it down.
4. Click any selected attribute and drag the attributes to the appropriate entity on the Workspace.
5. Release the mouse button and CTRL.

**Example 3:** Creating a shortcut of an entity from Model Explorer to appropriate Workspace directly.

1. In the Application Window, activate the Workspace, where you want to add a shortcut of the *Customer* entity.
2. In **Model Explorer**, find the *Customer* entity.
3. Click the *Customer* entity and drag it to the WS.

**Tip:** Have a look at the **Shortcuts** folder of the *Customer* entity in Model Explorer. A new shortcut has been created.

**Example 4:** Moving entities between Model Explorers of two models - from *Model A* to *Model B*:

1. Open Model Explorers for both models. You can leave them undocked, or at least one of them to be able to drag entities between them.
2. Make selection of entities in *Model Explorer A*.
3. Click any of the selected entities and drag them to the **Entities** folder of the *Model Explorer B*.
4. Release the mouse button.

## Message Explorer

In Message Explorer, hints, errors, warnings and other messages that occur during your work with Toad Data Modeler are displayed.

Message Explorer is by default docked at the bottom of the application window. However, you can hide or close it at any time.

### *To open Message Explorer again*

- » Select **Tools | Message Explorer**.

Id ▲	Date	Time	Message
1	7.8.2007	11:33:33	Connecting to database
2	7.8.2007	11:33:33	Connecting to Oracle version: 10.2.0.1.0
3	7.8.2007	11:33:33	Fetching tables from schema SYSTEM...
4	7.8.2007	11:33:34	Disconnected from database
5	7.8.2007	11:33:56	Connecting to database.
6	7.8.2007	11:33:57	Reversing table details...
7	7.8.2007	11:33:58	Reversing table comments...
8	7.8.2007	11:33:58	Reversing columns...
9	7.8.2007	11:34:00	Reversing primary keys...

You can sort messages by IDs, date, time and message type. - Simply click the appropriate column.

### *To save the messages*

1. Right-click in Message Explorer and select **Save Messages**.
2. Define a path where the *LocalLog.txt* file should be saved.

## To Do List

To-Do List allows you to keep records of tasks and make notes on unfinished actions.

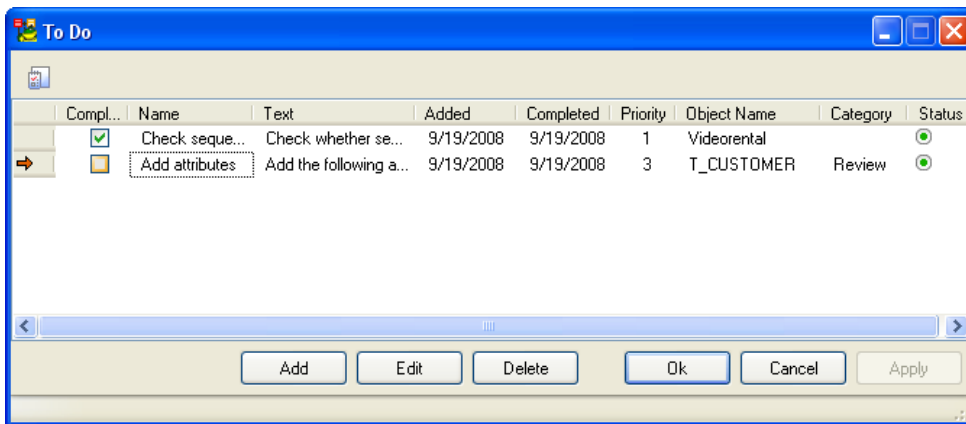
You can assign tasks to:

- Particular object of your model in its Properties dialog (see the Entity Properties form | To Do tab)
- Main To Do dialog - see the Model menu | To Do (A complete list of all To Do items can be found here.)

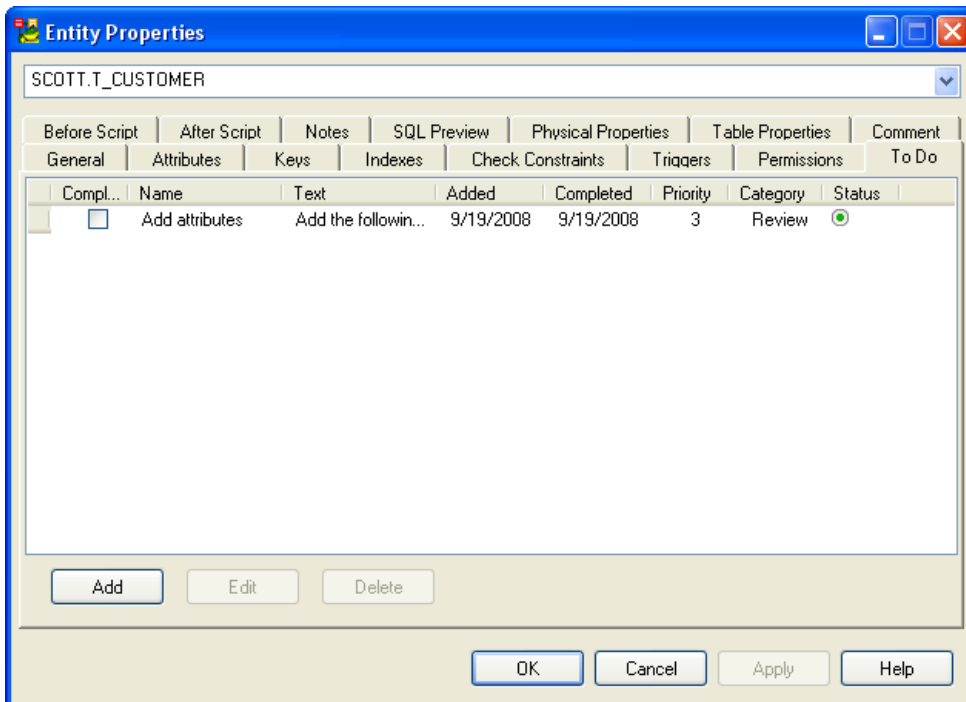
#### *To add a new To Do item*

» Select **Model | To Do | Add**.

**Note:** Or edit the object (e.g. entity) and in the **Properties** dialog, click **To Do** tab | **Add**.



**Example: To Do** tab in the **Entity Properties** form







### *To edit a To Do item*

1. Select **Model | To Do** | double-click the selected item.
2. Define/edit properties of the item.

The screenshot shows the 'To Do Properties' dialog box with the 'General' tab selected. The 'Name' field contains 'Add attributes'. The 'Category' dropdown is set to 'Review' and the 'Priority' spinner is set to '1'. The 'Date of Creation' is '2/ 6/2008' and the 'Deadline' is '2/ 8/2008'. The 'Completed' checkbox is unchecked. Below these fields is a list box titled 'Add the following attributes to this entity:' containing a bulleted list: '- phone', '- email', and '- cell'. At the bottom of the dialog are four buttons: 'OK', 'Cancel', 'Apply', and 'Help'.

## Undo/Redo Functions

In Toad Data Modeler, the Undo and Redo functions are available (see  and  on the the toolbar or in the **Edit** menu). These functions allow you to undo and redo steps only during your work with **models** and **packages**. So, they are not available e.g. in Version Manager, during scripting (in Scripting Window) etc.

Generally, you can use Undo/Redo:

- During your work with models on the Workspace, in Application View and Model Explorer.
- During your work with models in forms and dialogs (e.g. the Entity Properties form).

The functions will always return to a state before the confirmation **OK** or **Apply**.

- During your work with packages and scripts in Package and Script Explorers.

**Tip:** You can set up number of Undo/Redo steps in the **Settings** menu | **Options** | **Application** section | **General**.

# Toad for Oracle Integration

## Integration Options

Toad Data Modeler and Toad for Oracle products have started the integration process.

Here are the features for Toad for Oracle users that are available in Toad Data Modeler:

- Loading of Toad® for Oracle aliases
- Opening Toad® for Oracle projects
- Importing Toad® for Oracle ER diagrams
- Possibility to define Toad® for Oracle as a default editor for generated SQL scripts
- Possibility to use Toad® for Oracle icons in Toad Data Modeler

*To define default settings for Toad for Oracle integration*

» Select **Settings | Options | Application | Toad for Oracle Integration**.

## Toad for Oracle Aliases

You can use Toad for Oracle in the:

- **Connections** dialog (**File | Connections | Toad for Oracle Aliases**).
- **Reverse Engineering Wizard | Stored Aliases** step (**File | Reverse Engineering | Toad for Oracle Aliases**).

## Toad for Oracle as a Default Editor

If you define Toad for Oracle as a default editor, every generated SQL script will open in Toad for Oracle (Oracle models only).

The screenshot shows the 'Oracle 10g' configuration window. It has a dark blue header with the text 'Oracle 10g'. Below the header, there are three main sections. The first section is 'Code Editor Type' with a dropdown menu showing 'Use Custom Editor'. The second section is 'External Editor for Generated Code' with a text box containing 'C:\Program Files\Inno Setup5\Compile32.exe%CODE\_FILE\_NAME%'. The third section is 'Default DataType' with a dropdown menu showing 'Number(%p1,%p2)', and two input boxes for 'Precision' (10) and 'Scale' (0).

## Toad for Oracle Icons in Toad Data Modeler

- » Click **View | Icons Theme** and select **Toad for Oracle Icons**.

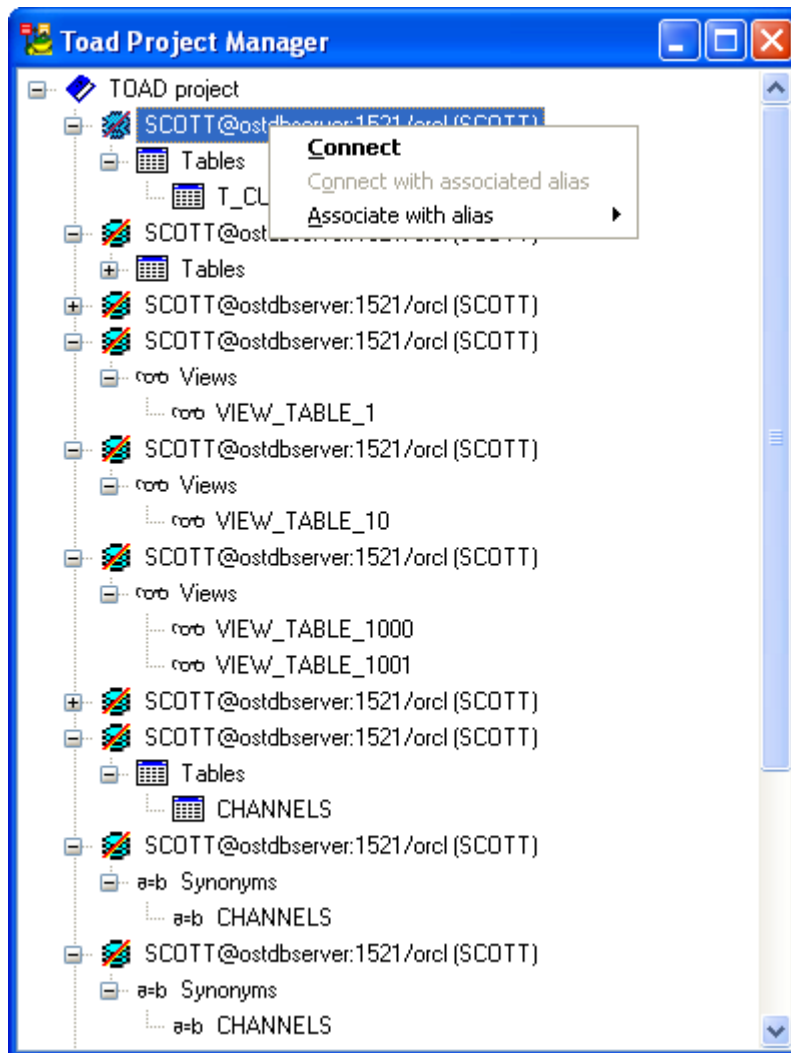
### Related Topics

Open Toad for Oracle Projects (page 248)

Import Toad for Oracle ER Diagrams (page 250)

## Open Toad for Oracle Projects

1. Select **File | Open Toad for Oracle Project**.
2. From the **Open** dialog, select \*.tpr file. **Toad Project Manager** opens.
3. Select a connection type, right-click it and select **Connect**.



4. After you are connected, you can select items from the Toad Project Manager and drag them to Workspace or Model Explorer of your model in Toad Data Modeler.
5. Right-click the connection and select **Disconnect** or simply close **Toad Project Manager**.

**Note:** Tables that already exist in the model cannot be added to the model.

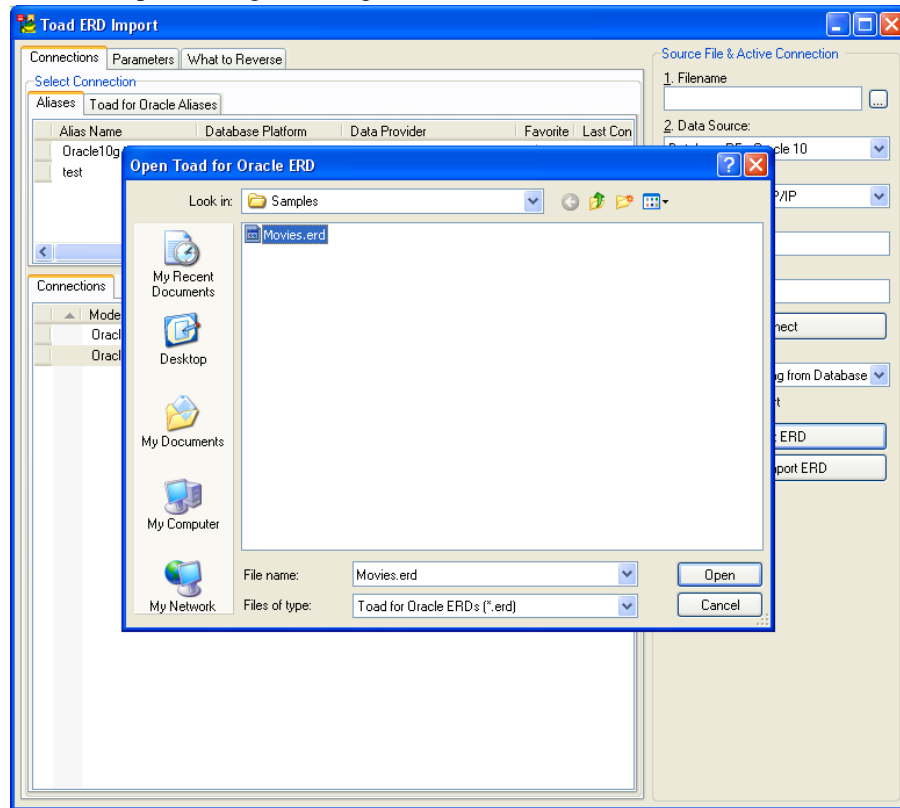
### Related Topics

Integration Options (page 247)

Import Toad for Oracle ER Diagrams (page 250)

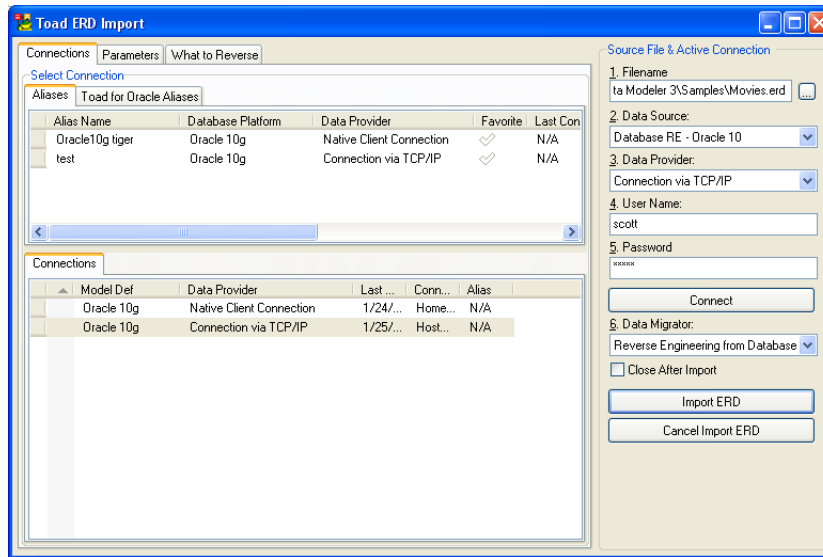
## Import Toad for Oracle ER Diagrams

1. Select **File | Import Toad for Oracle ERD**.
2. From the **Open** dialog, select particular .erd file.

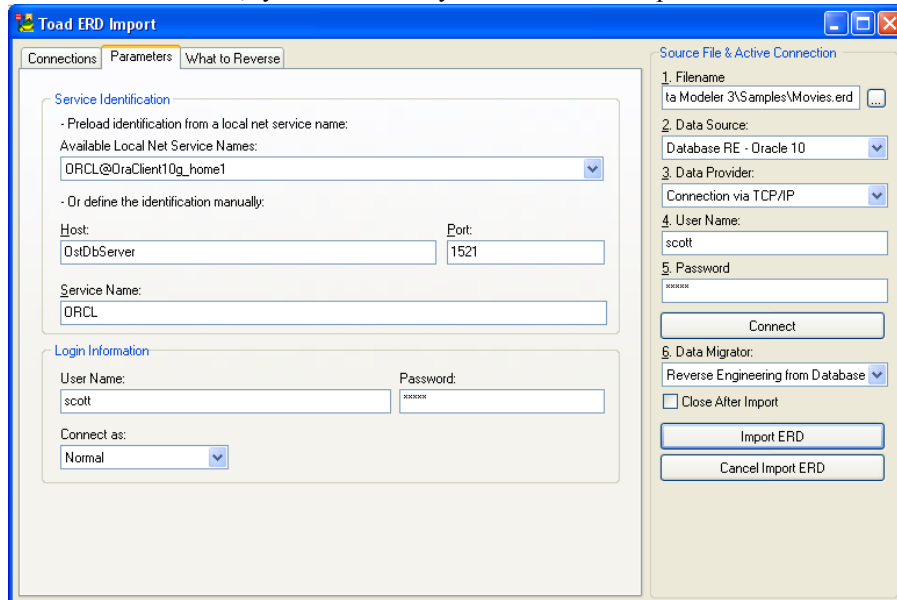


The file will be loaded to the **Filename** box on the right.

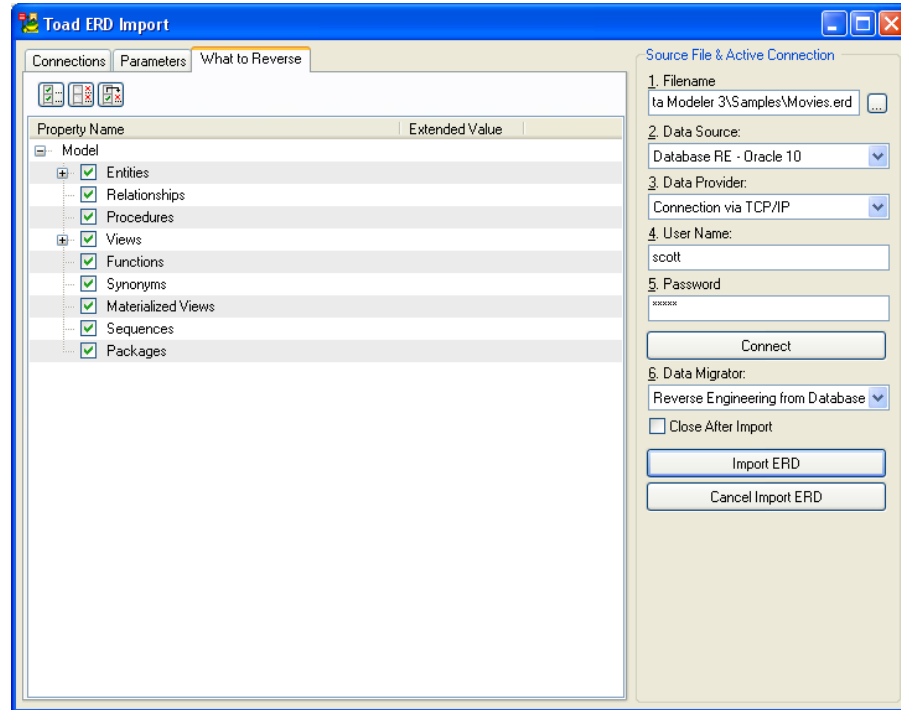
In the .erd file, there is only information on tables. Now it is necessary to connect to database to load information on attributes and other items of the ER diagram. In other words, reverse engineering has to be done to create a new model in Toad Data Modeler. Follow other steps.




3. On tab **Connections**, you can:
  - a. Select an alias from the **Aliases** or **Toad for Oracle Aliases** tab. Settings on the right in the **Active Connection** area will update accordingly and so will parameters on tab **Parameters**.
  - b. Define new connection settings in the **Active Connection** area.
4. On tab **Parameters**, you can modify the connection parameters.

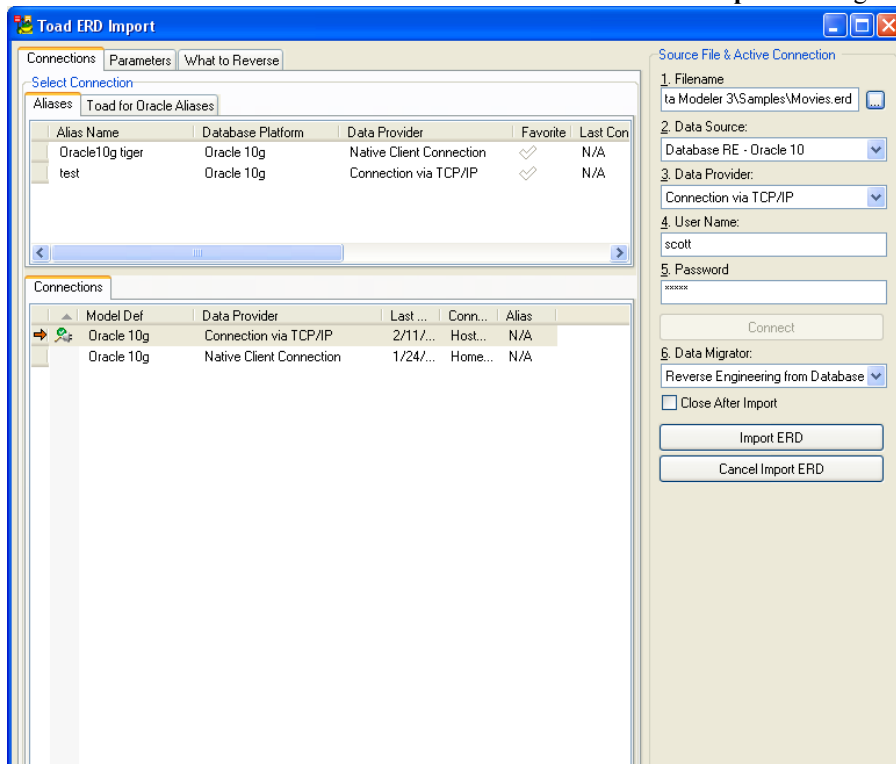


- On tab **What to Reverse**, select Object Types and Properties (OTP) you want to



reverse.

- On tab **Connections**, press **Connect** to connect to your database. In the lower **Connections** tab, you can see a list of last made connections, including this new one.
- As soon as you are connected, the following icon will be displayed  next to appropriate connection and the **Connect** option will become disabled. (You can also open the Message Explorer to see the process.)
- Click **Import ERD** to import the ER diagram to Toad Data Modeler and open it on the Workspace.
- Disconnect from the database and close the **Toad ERD Import** dialog.







# Synchronization

## Synchronization

Toad Data Modeler allows you to synchronize:

- Database and model
- Physical model and logical model

## Database and Model Synchronization

### Situation A

You have reverse engineered your database to Toad Data Modeler. -> A new physical ER diagram has been created. Later you have made some changes in your database.

Solution: Use the Model Update or Model Merge features to update your physical model in Toad Data Modeler.

### Situation B

You have made changes in your physical ER diagram in Toad Data Modeler and need to apply the changes in your database.

Solution: Compare the models and generate Alter Scripts in Toad Data Modeler. Then use a third party software to connect to your database and execute the script generated in Toad Data Modeler in your database. (Note: Alter script generation is not supported for all databases.)

## Physical Model and Logical Model Synchronization

You have created a Logical model in Toad Data Modeler. You need to convert the LER model to Physical model for the selected database system.

Solution: Use the Sync & Convert Wizard or Simple Model Conversion feature for LER to PER model conversion.

### Related Topics

Model Update (page 186)

LIVE Reverse Engineering (page 180)

[Alter Scripts \(page 192\)](#)

[Model Conversion - PER - PER \(page 196\)](#)

[Model Conversion - PER - LER \(page 199\)](#)


## Version Manager

### Create New Project and Add Files to Projects

In Toad Data Modeler Version Manager, you can create projects, add your models and other files to your projects, modify models, create their versions/revisions, send versions as files etc.

**Note:** Your project can contain files of any types, not only models. They can be text files, images, or other programs. Nevertheless, Toad Data Modeler is not able to work with such files - it's not possible to open e.g. a text file as a standard Toad Data Modeler model. You will be able to view them only (the **Show Version (only to read)** option).

#### To open Version Manager



1. Enable Expert Mode: select **Settings** | **Options** | **General** | select the **Expert Mode** checkbox.
2. Click  on the toolbar (or select **Tools** | **Version Manager**).

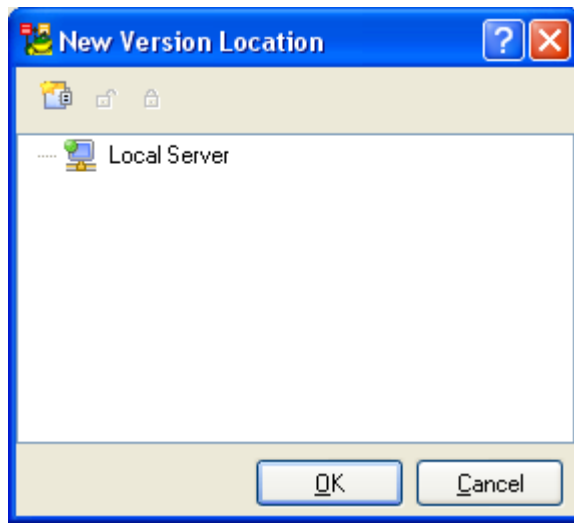
There are several ways how to create a project and add files to it:

- Method A: Add a model and create a project almost at one jump. (Recommended when you want to add a single Toad Data Modeler model to a new or already existing project.)
- Method B: Add a group of files and create a project almost at one jump. (Recommended when you store files that you all want to add to the new project in one directory. Available only for new, just being created projects.)
- Method C: Add a single file (no matter what type it is - model, text document, screenshot etc.) to already existing project.

#### Scenario

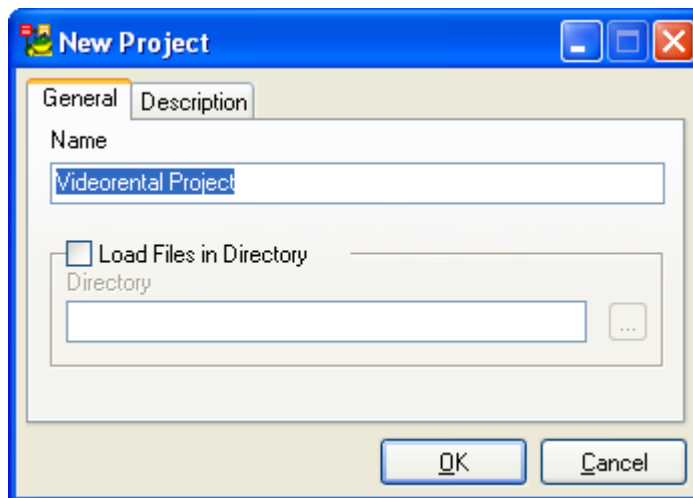
**Method A:** You want to add your Oracle 10g model *Videoreantal* that you have just created to Version Manager. For now, no project exists in Version Manager.

1. Open model *Videoreantal*.
2. Click  to add model *Videoreantal* to Version Manager.
3. In the **New Version Location** dialog, click  to create a new project.

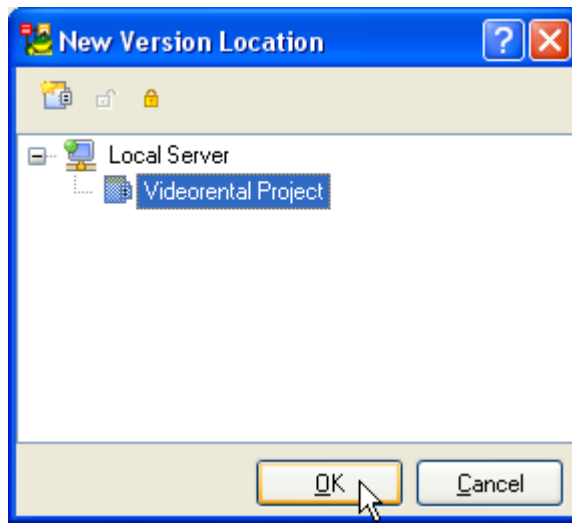


**Note:** If the project where you want to add the model existed, you would simply select the project and click **OK**. (And followed step 7.)

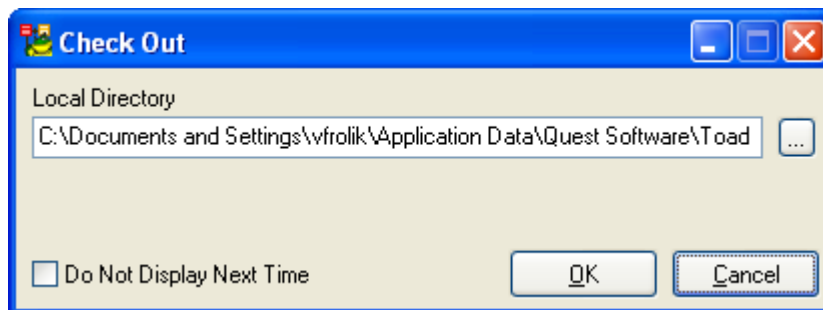
4. In the **New Project** dialog, define name and description of your project.



5. Confirm **OK**. -> You get back to the **New Version Location** dialog.
6. Select the new project *Videorental Project* and confirm **OK**.

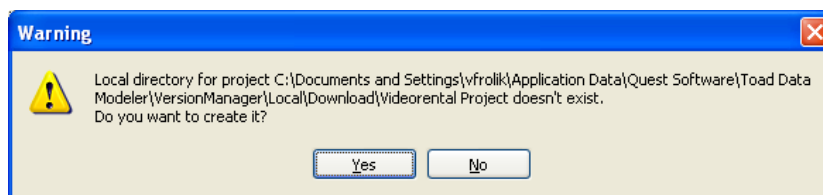


7. After your confirmation:
  - a. The selected model will be added to the project.
  - b. Version 1.0 of the model will be created automatically and checked out. Before it is checked out, you will be asked the following question:



In the **Local Directory** box, you are asked to define/confirm a path where you want to load the version.

8. The file doesn't exist and therefore will be created. Click **Yes**.



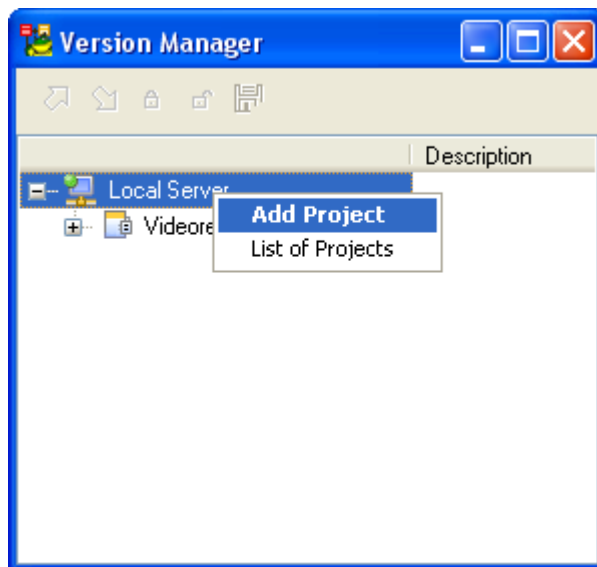
Designer of the version will open in the **Application Window** and the file will be listed also in the **Application View**.

## Scenario

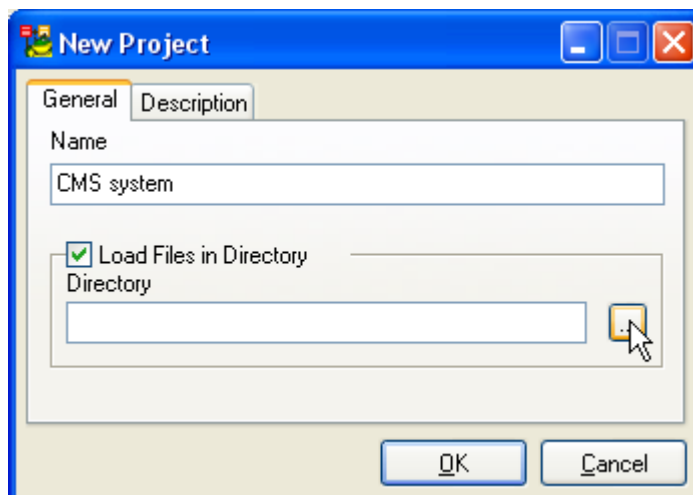
**Method B:** Create a new project and add a group of files to the project.

Note: You already have all the files that you want to add to the project stored in one directory.

1. Open **Version Manager**.
2. Right-click the *Local Server* and select **Add Project** to create another project.



3. Define name and description of the project.
4. To load files that you want to add to the project, select **Load Files in Directory** and click the small icon on the right.

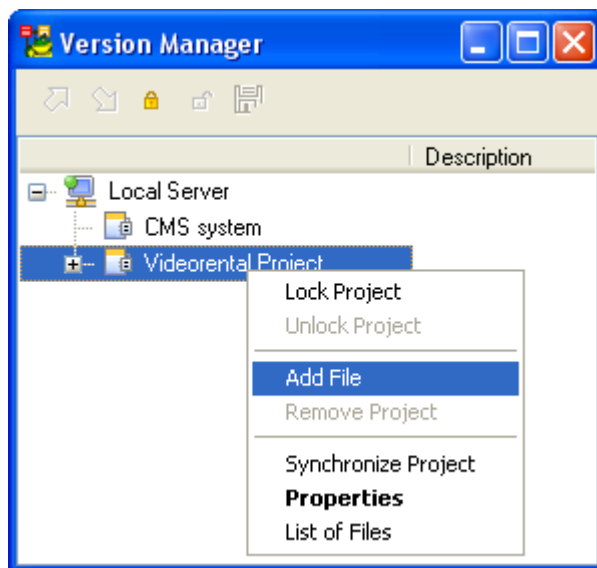


5. Find a directory where the files are stored.
6. Confirm **OK**.

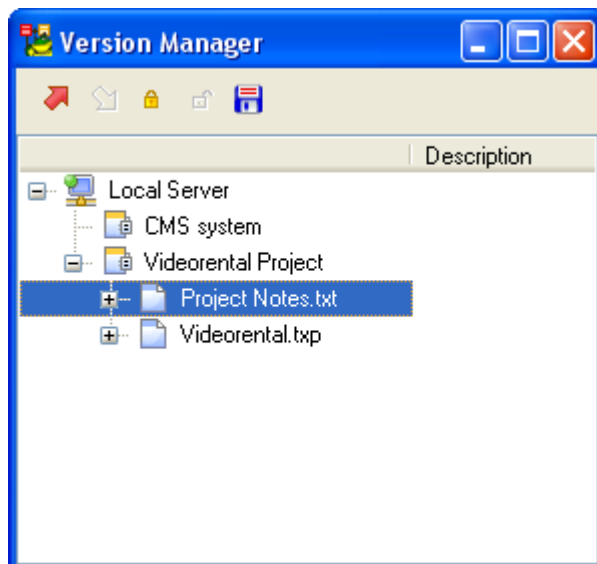
## Scenario

**Method C:** Add a single TDM model (or any other file) to already existing project.

1. Open **Version Manager**.
2. Right-click the selected project and select **Add File**.



3. From the **Files of Type** box, select the appropriate type, find the file and confirm **Open**.





## Related Topics

Work with Versions/Revisions (page 261)

View and Sort VM Items (page 269)


## Work with Versions/Revisions

In the previous step, you created some new projects and added some models/files to them.

Now you will learn about how to use versions/revisions of the files/models in Version Manager (VM), and also get some helpful tips and practices.

## Version Check Out

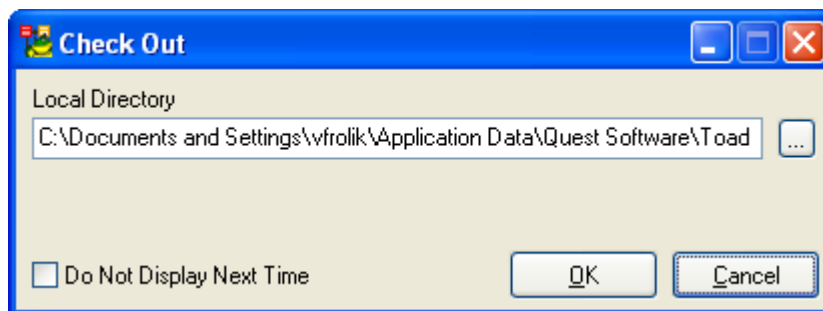
*To start editing a version/revision*

1. Select the version in **Version Manager** (VM) and click .
2. You are asked to define a path where you want to load the selected version. In this place, the version will be saved and you will work with the version from here. By default, the path is set up in the **Settings** menu | **Options** | **Version Manager** | Local Directory.

Possible path is:

C:\Documents and Settings\user\Application Data\Quest Software\Toad Data Modeler\Installation name\VersionManager\Local\Download\Project\_Name.

**Tip:** If you don't want Toad Data Modeler to ask you about the local directory path for every check out, you can select the **Do Not Display Next Time** checkbox.



3. Confirm **OK**. See what will happen:
  - Selected version will be loaded from a server to your local file. The version will

be locked automatically on the server. (Version lock properties show information on who locked the project and when.

**Note:** For now, all users in Version Manager are Admins.

- The version item will display in the **Application View** from where you can manage it as well (see the pop-up menu or main menu toolbar).
  - The version will open in the **Application Window**.
4. Feel free to work with the version.

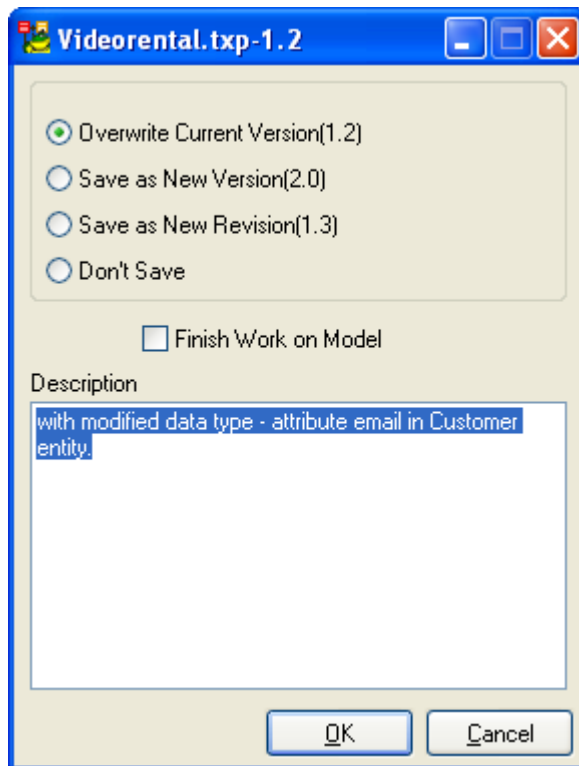
**Note:** Multiple version Check Out is possible. - Use SHIFT key to select versions and click **Check Out** then.

## Version Check In

*To save the changes you've made in the version*

1. Click  in **Version Manager** or on the main toolbar.

**Note:** Multiple version Check In is possible. - Use SHIFT key to select versions and click **Check In** then.

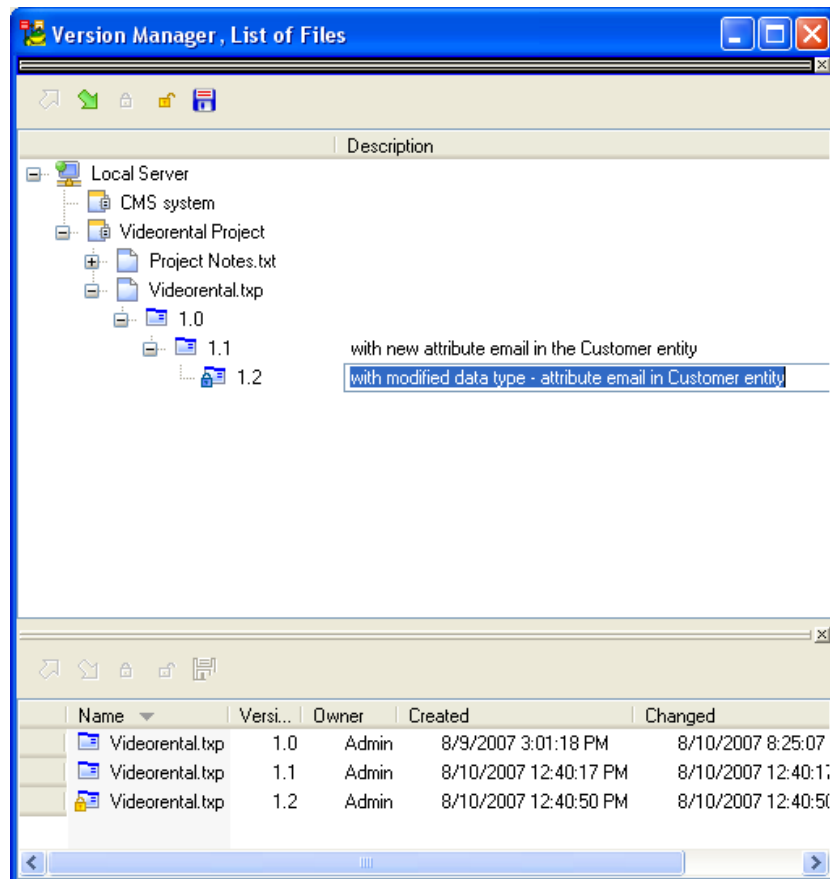


2. Select how you want to save the changes - overwrite the current version, save as a new version, revision or do not save. Select the **Overwrite** option.

**Tip:** If you select **Overwrite Current Version** and leave the **Finish Work on Model** checkbox clear, the changes will be saved and the version will remain open for another edit. This allows you to continuously save the changes and be sure you will not lose them. You can overwrite the version until you are sure the version is a "milestone" version. Then you will select **Save as New Version/Revision**. This option prevents you from creating many new versions that you find useless later.

3. Write description on the changes you've made in the version. The description will be displayed in the VM next to the version number.

**Tip:** You can edit the description in the VM directly via the inplace editor!



4. Confirm **OK**. Result:
  - The version will be saved to your local disc and possible changes will be applied on a server.
  - Version will be automatically unlocked and will become accessible for other team members.

**Note:** Projects.xml file is saved after every change made in the project. The file is

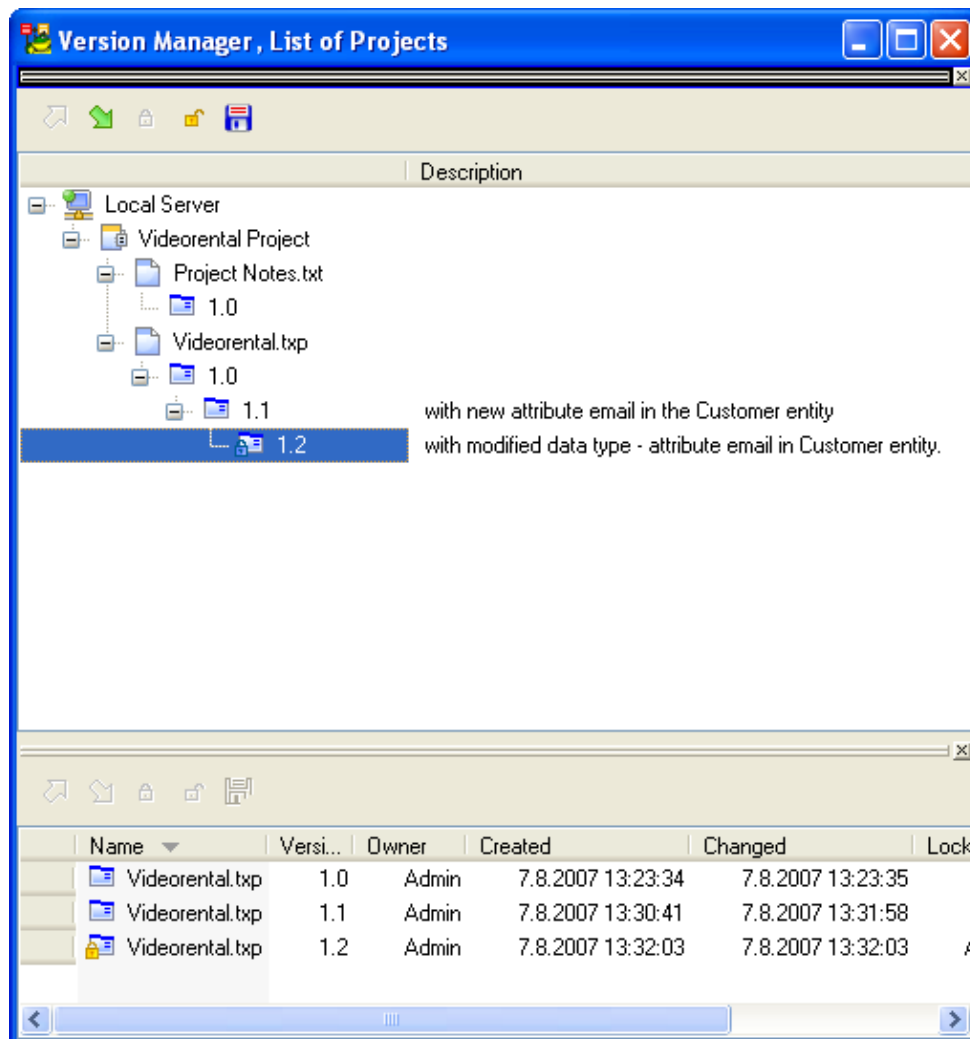
being updated continuously – changes made and saved by user A will be visible to user B.

## Places to Manage Versions

You can work with the versions in:

- **Version Manager (VM)**
- **Application View** and via the main toolbar (After **Check Out** from VM)
- **List of Versions**

**Tip:** Dock the List on the Version Manager dialog to have a different view of the versions (files).



### Versions in the Application View

- Version Manager can be closed.
- **Check In** (with an option to save changes and leave the version opened for other edit) is available on the main toolbar.
- Other options for particular version/revision are available from the version pop-up menu in the AV.
- More space for your modeling (use the splitter to hide the AV.)
- Helpful for longer editing (you know you don't need to open VM soon).

### Versions in the Version Manager

- VM allows you to view gradual process of how you created the versions/revisions.
- You can see what version precedes which one, what version has been derived from which one etc.
- You can dock VM and hide it too.
- From the VM, you can open the **List of Files** or **Versions**.

### Versions in the List of Versions

- It offers a different view on your versions/revisions.
- You can sort items by version number, date of creation, change etc. (click appropriate column header),
- You can check them in, out, lock and unlock them, or save selected version as a standard file. (See the icons in the toolbar that are same as those on the Version Manager toolbar).
- You can dock it and later hide it too.

### Tips on Version Right-click Options

Right-click the version to see the following options:

#### Lock/Unlock

Versions that you check out are locked automatically. However, you might need to lock a version of which edit has already been finished - not to overwrite it. Simply click the **Lock** icon. If you need to unlock it later, simply click **Unlock**.

How does it work? - If you check out a version, the version locks and the corresponding file on the server becomes read-only, however it is editable in your local file.

When you check in the version, the version unlocks and the corresponding file on the server can be modified. In your local file, it is locked and becomes read-only.

**Note:** Generally, there are two colors of padlock in Version Manager. Blue padlock means that you are the person who locked the project/file/version, so only you can modify it. Yellow padlock means that another person locked the project/file/version, so you are not allowed to make any changes in it unless the project/file/version is unlocked by the particular person again.

**Tip:** Use **Unlock** as Undo after **Check Out**. E.g. You checked out a version, made some changes and before you check the version in, you find the changes wrong. Simply click **Unlock** and all the changes will be canceled. The version will close from the AV and AW, and will be prepared for new Check Out in VM.

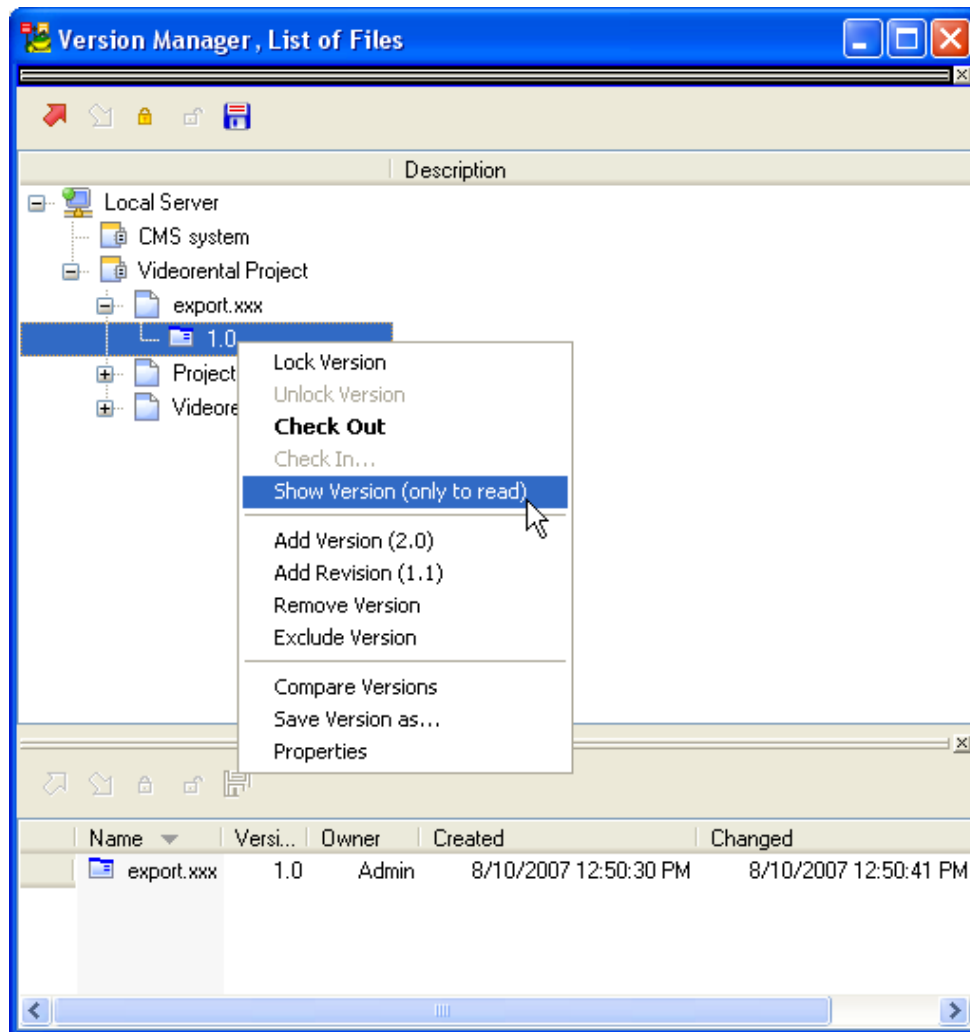
### Save Version as

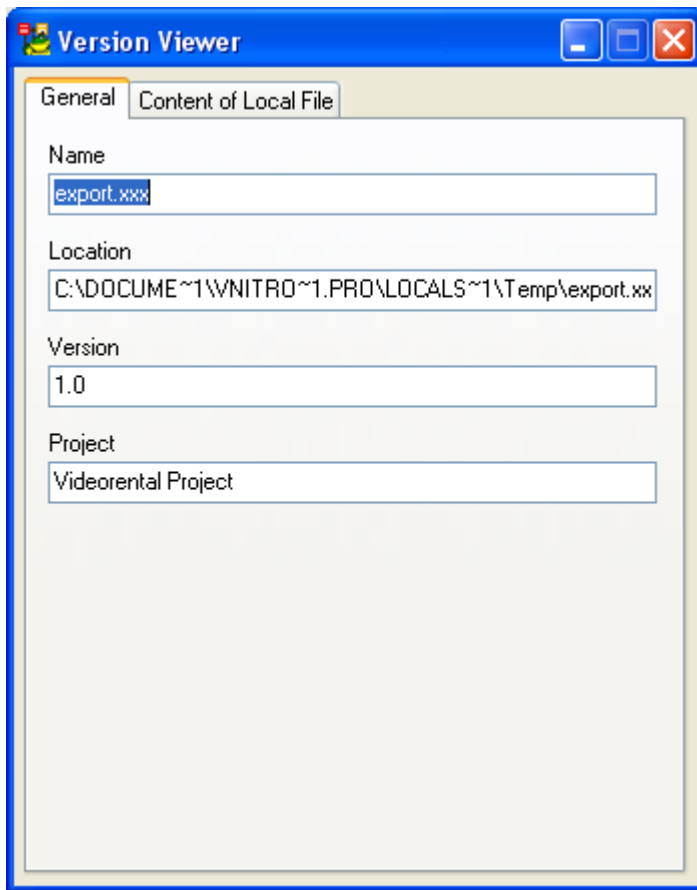
This option allows you to save selected version as a standard file (Toad Data Modeler models with extension .txp or .txl) and e.g. send them to your colleagues, customers etc. Also, you can add such a file to another project in Version Manager.

### Show Version (Only to Read)

If you select this option, the version will open in appropriate program, e.g. Word, Notepad, Windows Viewer, Acrobat Reader etc. Toad Data Modeler models will open in the Application Window.

If your file has an extension that does not associate with any program, it will open in the **Version Viewer** dialog (see the screenshots below). On tab **Content of Local File**, you will see the text.





### Remove Version

Selected version and all its revisions will be removed from Version Manager.

### Exclude Version

Use this option if you have created some intermediate revisions that you find useless now and want to get rid of them. Only the selected version will be excluded, its revisions will remain.

**Note:** For other options, please see the Help file, "Version Manager" chapter.

### Related Topics

Create New Project and Add Files to Projects (page 256)

View and Sort VM Items (page 269)



## View and Sort VM Items

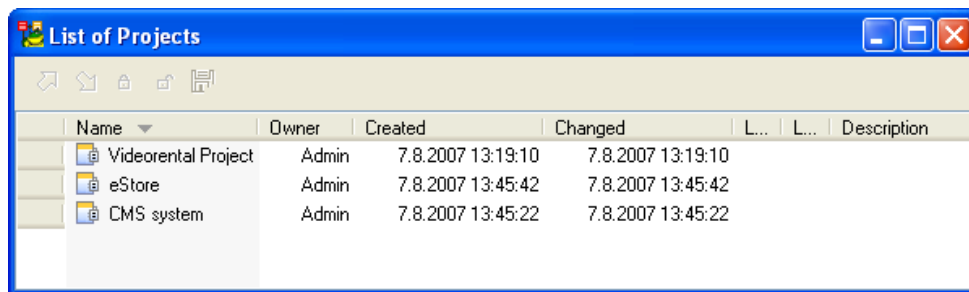
In Version Manager, the items are sorted this way:

- Projects are listed alphabetically.
- Under projects, files are listed in the alphabetical order too. Under files, their versions and revisions are displayed.
- Versions and revisions of files are sorted in the order they were created. This sorting allows you to see what version precedes which one, what version has been derived from which one etc.

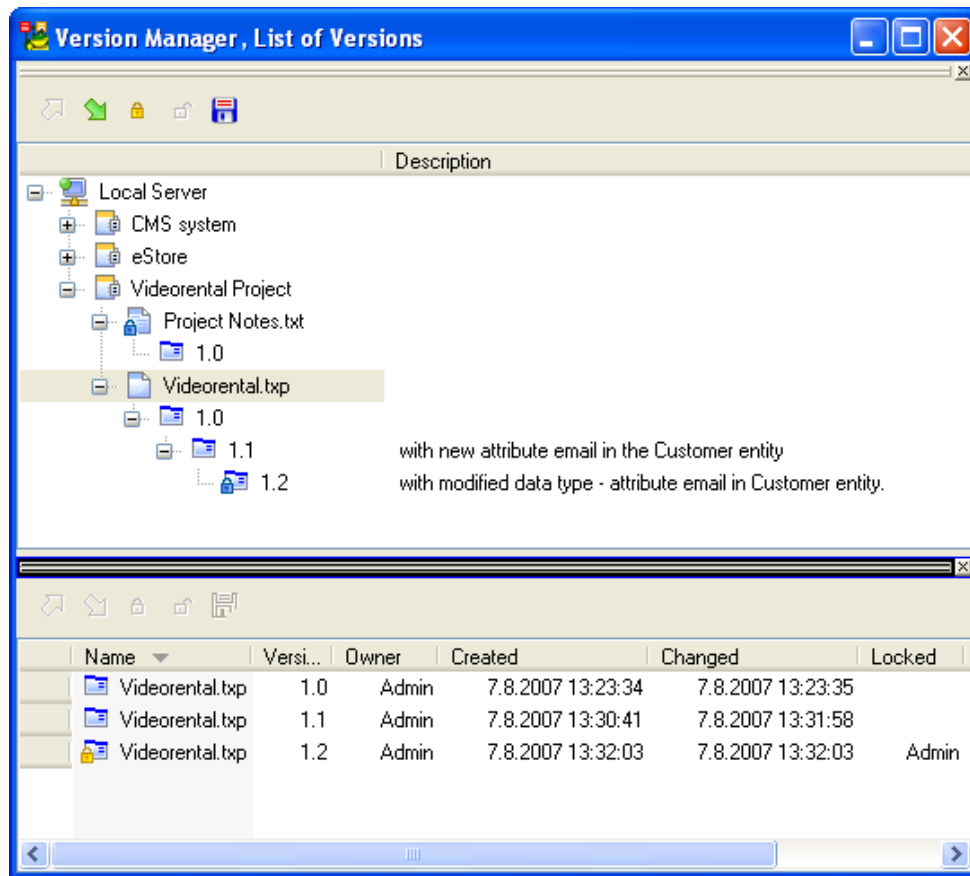
Nevertheless, Toad Data Modeler offers you an option to display the VM items in a different view and sort them - on the **List**.

The **List** displays information on items in Version Manager - projects, files and versions. Here, you can sort the items by different conditions (by name, version number, date of creation etc.), however you are not able to edit them.

The **List** opens together with Version Manager. You can dock it wherever you want. If you close it, you can find it later in appropriate pop-up menus of items in Version Manager. E.g. Right-click the server and select **List of Projects**.



**Tip:** You don't have to close the **List of Projects** to open **List of Files**. To see all files of the selected project, simply click the project in the Version Manager tree and the List of Projects will change to List of Files automatically. To see all versions/revisions of a file, simply click a file in Version Manager tree to display the List of Versions.



## Related Topics

Create New Project and Add Files to Projects (page 256)

Work with Versions/Revisions (page 261)

## Configuration

### Options

#### *To configure global settings*

- » Select **Settings| Options**.

In the **Options** dialog, you can define:

- Default options (e.g. options for models) - If you change the default options, the changes will be applied for **new models**, not in already existing models.
- Other options (e.g. number of Undo/Redo steps, paths in Version Manager) - You can change them at any time. The changes will be applied immediately for currently opened model.

There are two main sections in the **Options** dialog:

- **Application** where you can modify all basic settings of Toad Data Modeler.
- **Model** where you can modify settings for every new model you will create.

See below some information on a few of them. Their complete description can be found in the Help file, "Options and Configuration" chapter.

#### Expert Mode Page

Expert mode is a feature for experts and advanced Toad Data Modeler users who are interested in customizing the application, writing or editing scripts, creating new packages, working in Version Manager etc.

#### *To enable Expert Mode*

- » Click the **General** page and select the **Expert Mode** checkbox.

After it is selected, take notice of the new **Expert Mode** page above page **General**.

See "Our Goal" (page 274) for more information about customization sample.

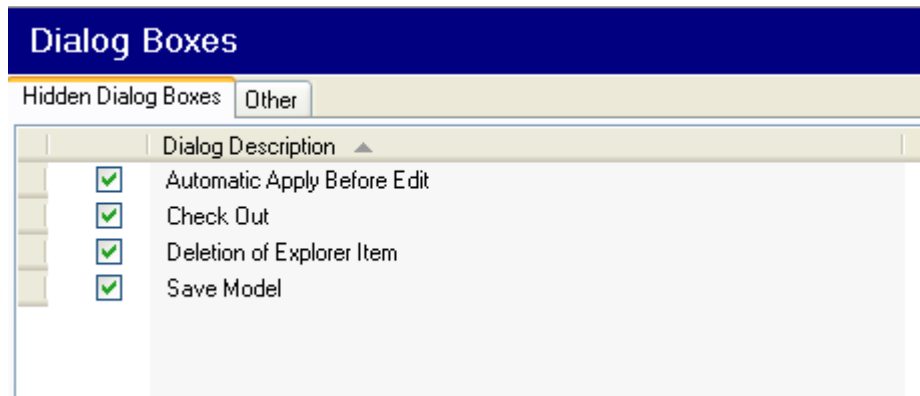
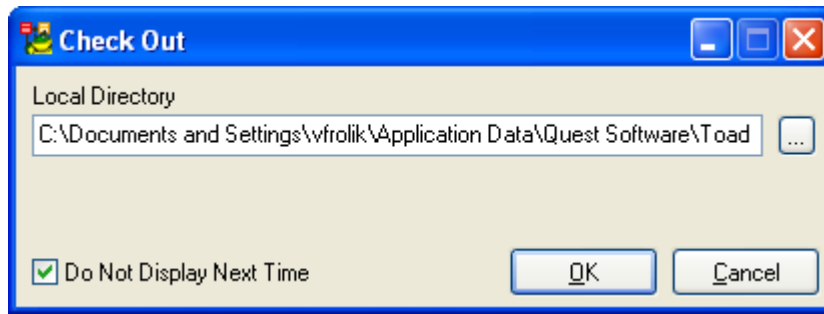
#### Dialog Boxes Page

This tab is empty until you forbid a dialog by selecting the 'Do not show next time' option in the dialog itself. - If you select this option, a new checkbox on the particular operation will display here - on the **Dialog Boxes** page.

To enable the particular dialog/operation again, select the appropriate checkbox here.

**Example:**

If you click the 'Do Not Display Next Time' checkbox in the **Check Out** dialog, the item will appear in the list of **Dialog Boxes**.



Other examples:

**Automatic Apply before Edit** - If it's selected, you can add e.g. new attributes to entities without the **Apply** confirmation and edit them directly.

**Deletion of Explorer Items** - If it's selected, the item for which you select **Delete Item** in Model Explorer, will be deleted from your model without further confirmation.

**Save Model** - If it's selected, the model for which you select **Close Model** will have to be saved.

## Customization - Sample

### Expert Mode

Before you start customizing the application, you need to enable the **Expert Mode** option in Toad Data Modeler.

After the Expert Mode is turned on, particular functions and options, hidden so far in Toad Data Modeler, will become available.

1. Select the **Settings** menu | **Options** | **General** | **Expert Mode**.
2. The following functions/options become available:
  - Script Explorer
  - Package Explorer
  - Version Manager
  - Scripting Window
  - Generate XSD File
    - Items in the **Tools** menu + particular icons in the toolbar.
  - Data Type Conversion Settings
  - Available OTPs
    - Items in the **Settings** menu.
  - New Package
  - Add to Version Manager
    - Items in the **File** menu.
  - Reference
    - Item in the **Help** menu.
  - Add to Version Manager
  - Test Model
  - Repair Model
    - Model options in the **Application View**.
  - Options for defining, loading and storing Object Types and Properties

(OTPs)

- in various forms and dialogs. More details on OTPs can be found in the Help file, "OTPs" topic.

3. After you select the **Expert Mode** checkbox, take notice of the new **Expert Mode** item in the **Options** dialog (above **General**). Here, you can see this option - **Save the definitions to the 'My Package'**. - Clear this checkbox.

Notes:

- If you wanted to save all changes you made in the application (e.g. while customizing a form) to My Package, you would leave this checkbox selected.
- If you wanted to save the changes you made to another package (e.g. new add-on package that you created), you would clear this checkbox. - This is what we will do in our Customization Sample. Follow the next topics.

### Related Topics

[Our Goal \(page 274\)](#)

## Script and Package Explorers

### Script Explorer

In Script Explorer, you can edit existing scripts, extend functionality of existing scripts, create your own scripts etc.

### Package Explorer

Package Explorer displays structure of packages in Toad Data Modeler. Here, you can work with objects saved in the packages - rename them, move, copy etc. You can also access all Meta Models from the Package Explorer.

## Our Goal

The following tutorial guides you through some of the customization options available in Toad Data Modeler.

Read the goal definition below and go step by step to accomplish it (see the topics one by one in this section).

During your work, you will use various tools (e.g. Package or Script Explorer etc.) and features (e.g. form editing or setting default values etc.). To follow the point of the example fluently and clearly without digressions, details on these features (their functionalities and options) are not described. For more information on them, please see the appropriate topics/chapters in the Help file.

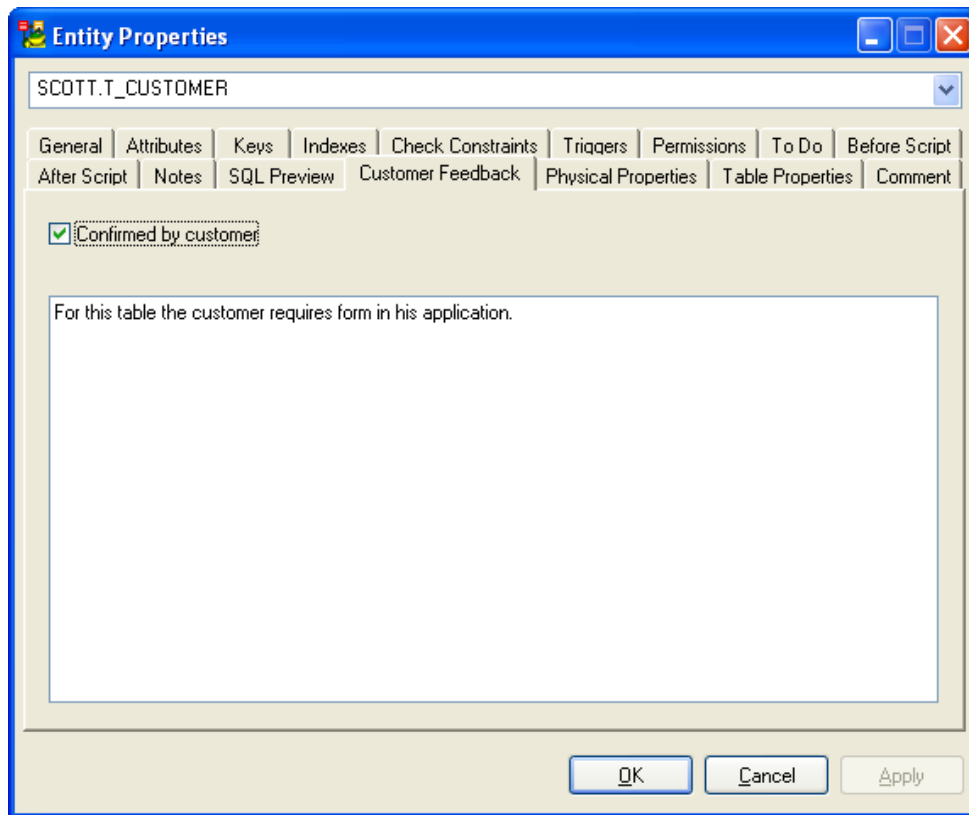
## Our Goal Definition:

- To be able to mark an entity as "Confirmed" By Customer.
- To be able to write Customer Notes to entity, but don't mix them with Comments or Notes in Toad Data Modeler. Customer notes and developer notes must be differentiated.
- Define both values in the Entity Properties form, if possible.
- Store both values in a model.
- Write a list of Confirmed and Not Confirmed entities to Message Explorer (Log). Just to get a quick overview.
- Generate modified HTML reports with both the entity status and notes from customer.
- All for Oracle 10g only. Other models for other databases must remain untouched.

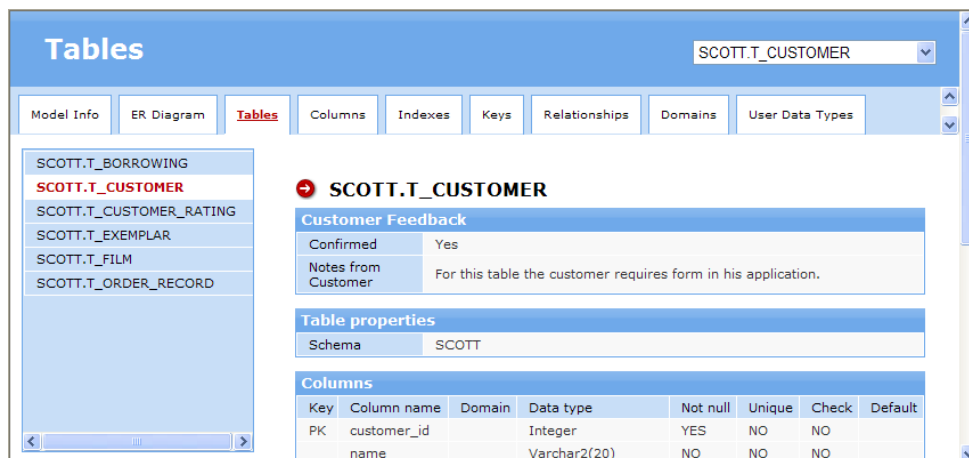
Think about how you could achieve this task in another application (Toad Data Modeler or any other modeling tool), and then read the rest of this section to find out how easily you can achieve this in Toad Data Modeler.

Before you start, see the following two screenshots for an inspiration:

1. There is no **Customer Feedback** tab in any standard **Entity Properties** form. This tab has been added to the form by end-user - by YOU!



2. Standardly, in generated HTML reports, there is no section **Customer Feedback**. As you can see below, users may use the values defined on modified forms, and generate modified HTML reports. The section Customer Feedback has been added to the reports by end-user - by YOU! (And generated automatically, of course.)



## Related Topics

Create Package (page 277)



## Create Package

### Why Do We Need a New Package?

Packages are containers for groups of scripts, customized Form definitions, metamodels etc. In Toad Data Modeler, the following three types of packages may exist.

- **System packages** - have the lowest priority (distributed with Toad Data Modeler application).
- **Add-on packages** - have higher priority than system packages (can be downloaded from web site, shared among users etc. No add-on package exists after installation.).
- **My Package** - has the highest priority (created automatically upon installation of Toad Data Modeler).

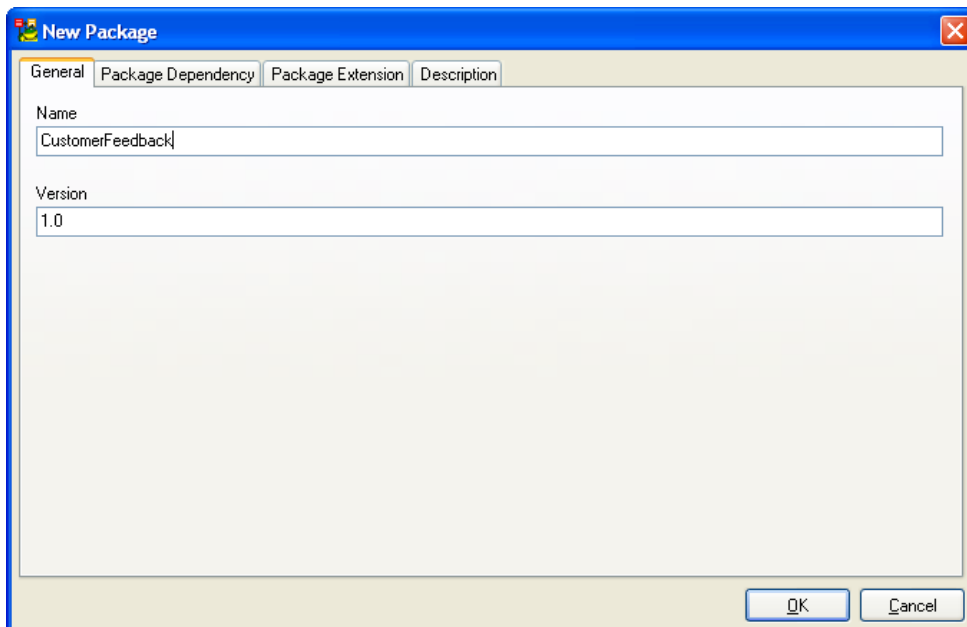
All the packages exist as separate XML files with extension .TXG.

You can make your modification without the necessity to create a new package, but all scripts and modifications you will ever make will be stored in the My Package.txg file. If you plan to share your modifications with others, it's a good idea to create a new package for this purpose. In this example, we will create a new package *CustomerFeedback*, and store all scripts used in this tutorial, metamodel and form modifications into this package. It will give us the possibility to share the CustomerFeedback.txg file with others.

## Create a New Package

Click **File | New Package** to open the following dialog.

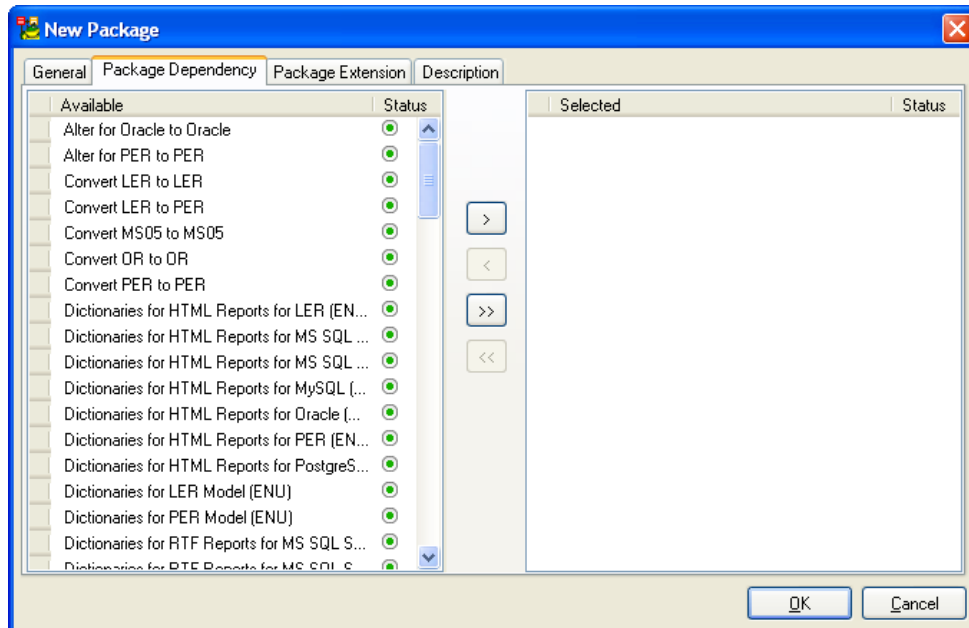
Define Name of the package.



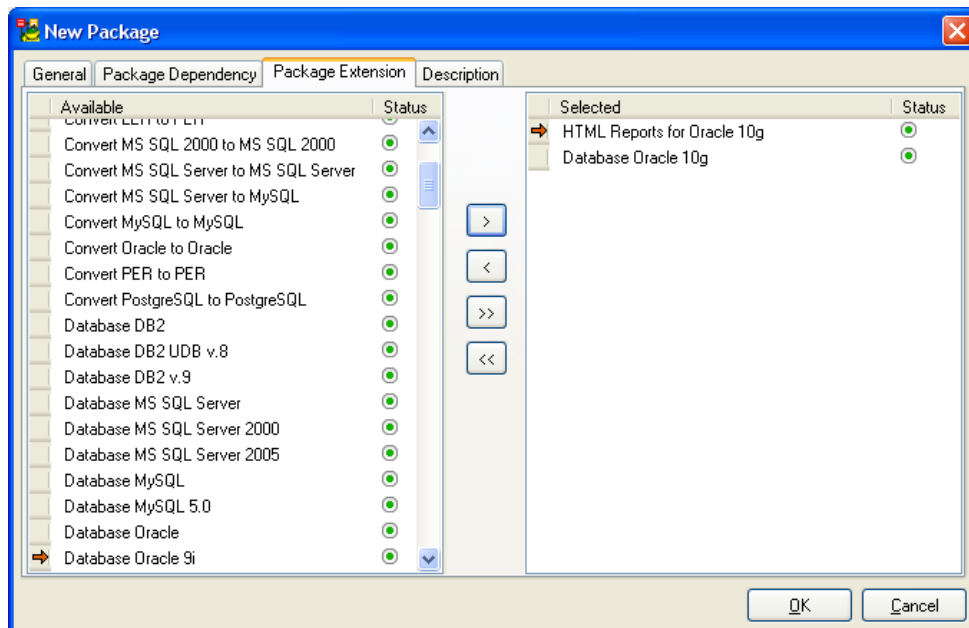
The screenshot shows a 'New Package' dialog box with a blue title bar and a standard Windows-style border. It features four tabs: 'General', 'Package Dependency', 'Package Extension', and 'Description'. The 'General' tab is active, displaying two text input fields. The 'Name' field contains the text 'CustomerFeedback' and the 'Version' field contains '1.0'. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

Select packages on which the new package depends. In this example, the Customer Feedback will not be dependent package. It will just extend existing packages.

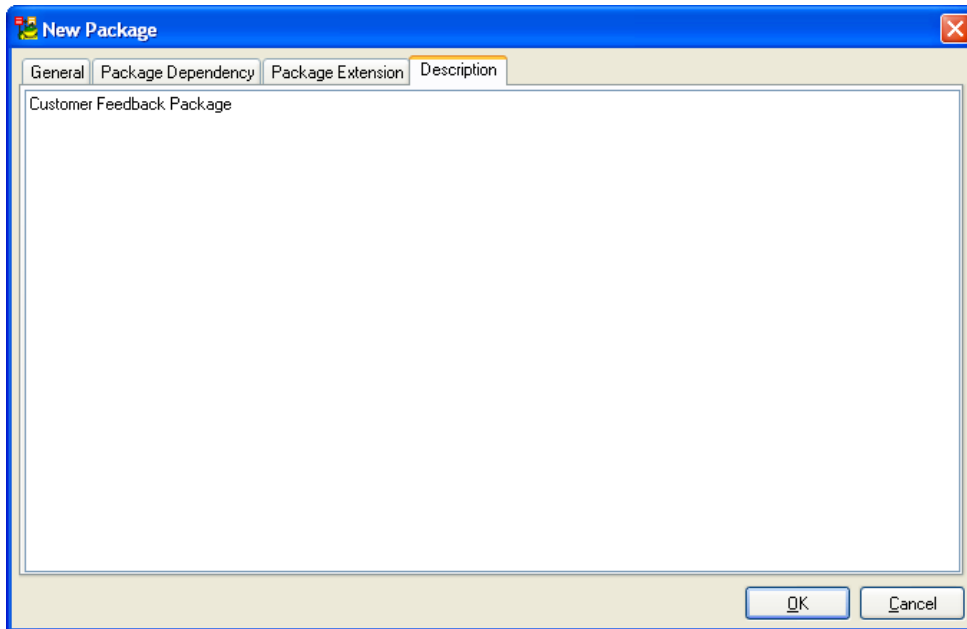
**Note:** Example of dependent package: Package **RE MS SQL Server 2005** depends on **RE MS SQL Server** package and extends **MS SQL Server 2005** package. (RE is abbreviation for Reverse Engineering.)



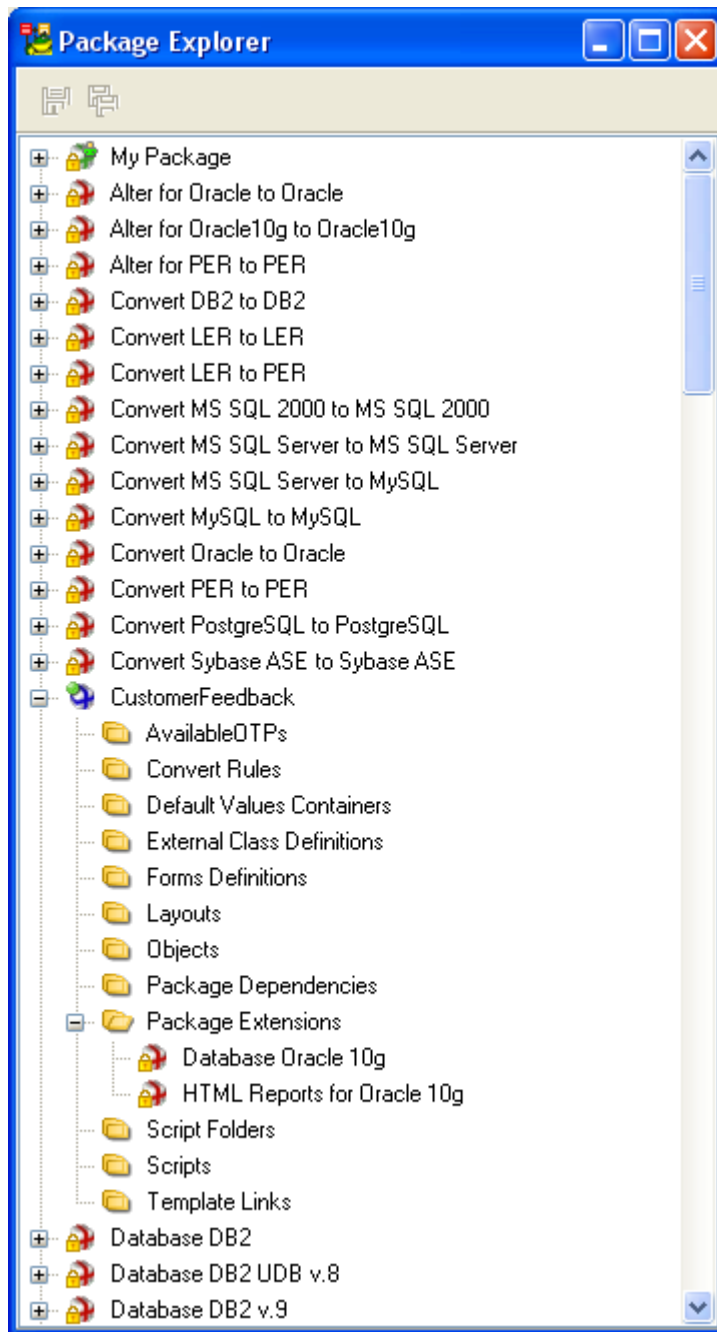
Select packages you want to extend. In our example, we will be extending Oracle 10g package and HTML Reports for Oracle 10g.



You can write description to the **Description** tab.



Newly created package will appear in the **Package Explorer**. You can also see package extensions there. Add-on packages have blue icon.



Just to compare, see My Package (where all modifications are stored if you don't use add-on packages) - it has a green icon and is listed at the top.

Well, a new package exists, let's continue adding new properties.

## Related Topics

[Add New Properties in Metamodel \(page 281\)](#)

## Add New Properties in Metamodel

Properties and methods can be added visually, via Metamodel.

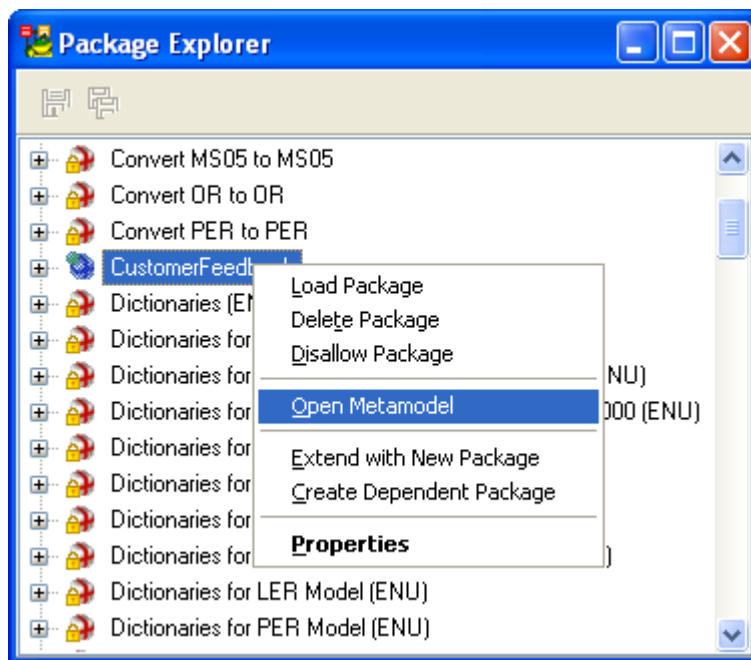
For our purpose, we will need two new properties.

- **ConfirmedByCustomer** (boolean)
- **NotesFromCustomer** (string)

This is where our values will be stored. The properties will be assigned to items that will appear in **Entity Properties** form. The ConfirmedByCustomer property value will be assigned to a checkbox, and the NotesFromCustomer value will be assigned to a text box.

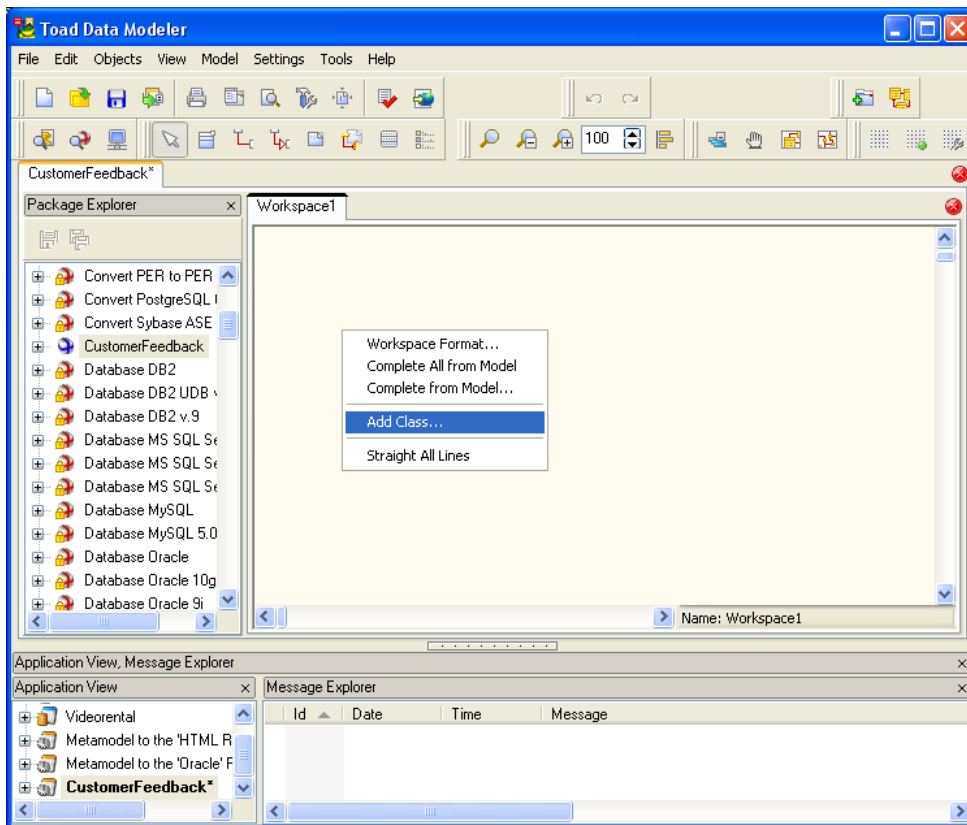
## How To Add New Properties

Right-click the CustomerFeedback package and select **Open Metamodel**.



The following will be displayed.

Right-click the metamodel workspace and select **Add Class...**

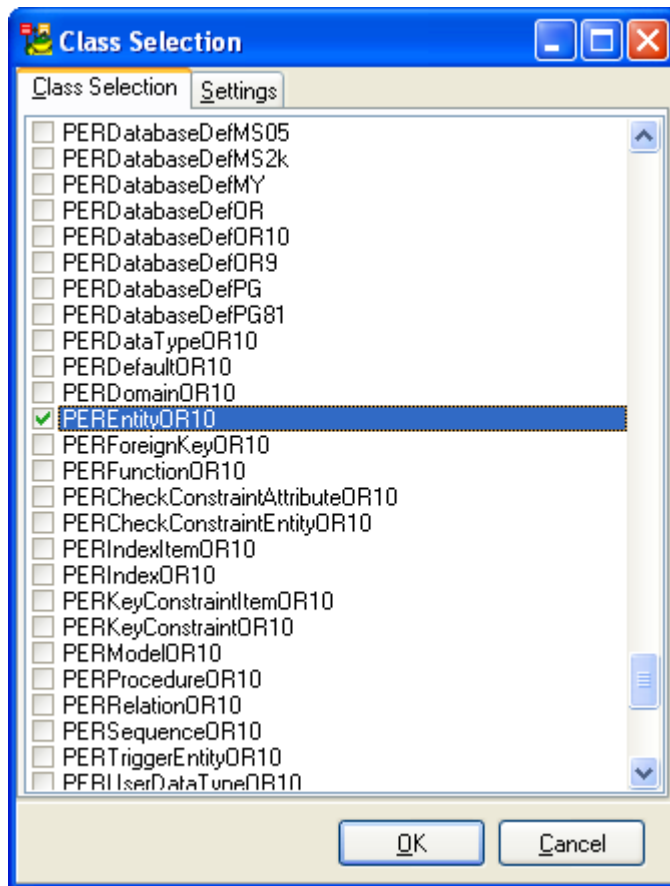


Select class you want to extend. For our purpose, we need to extend **PEREntityOR10** class.

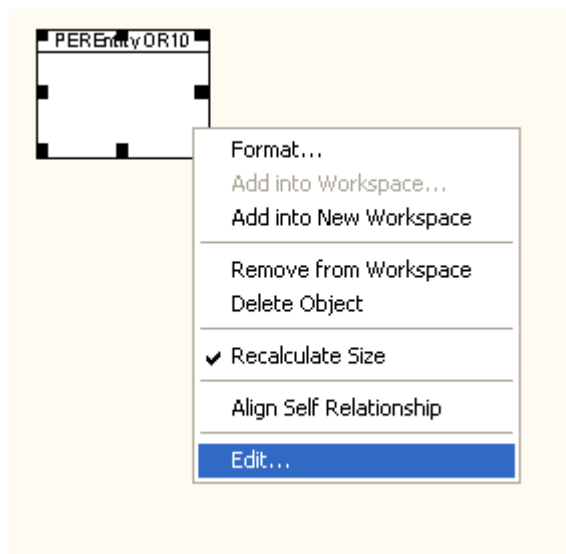
**PER** - Physical Entity Relationship model.

**Entity** - Items must be accessible in the Entity Properties form.

**OR10** - Modification will be made for Oracle 10g only.

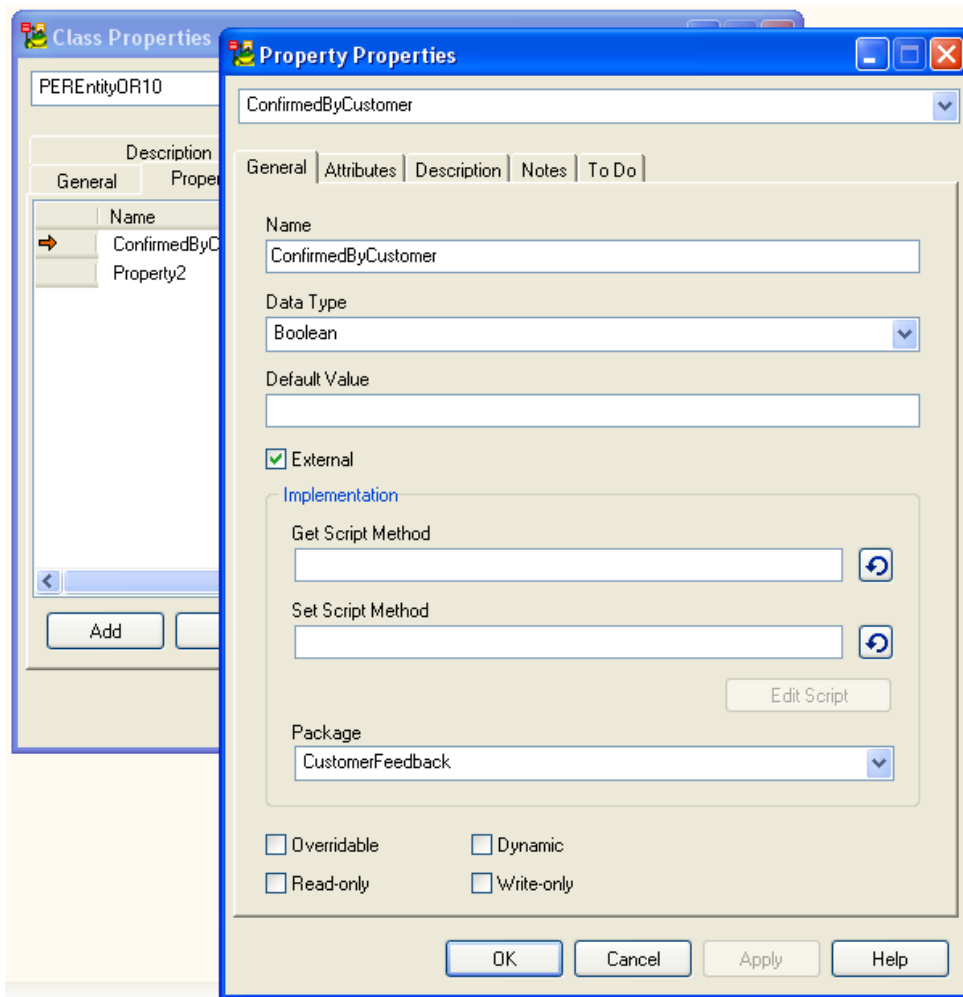


Select the newly added class and right-click it. Select **Edit**.



Add two new properties to the class.

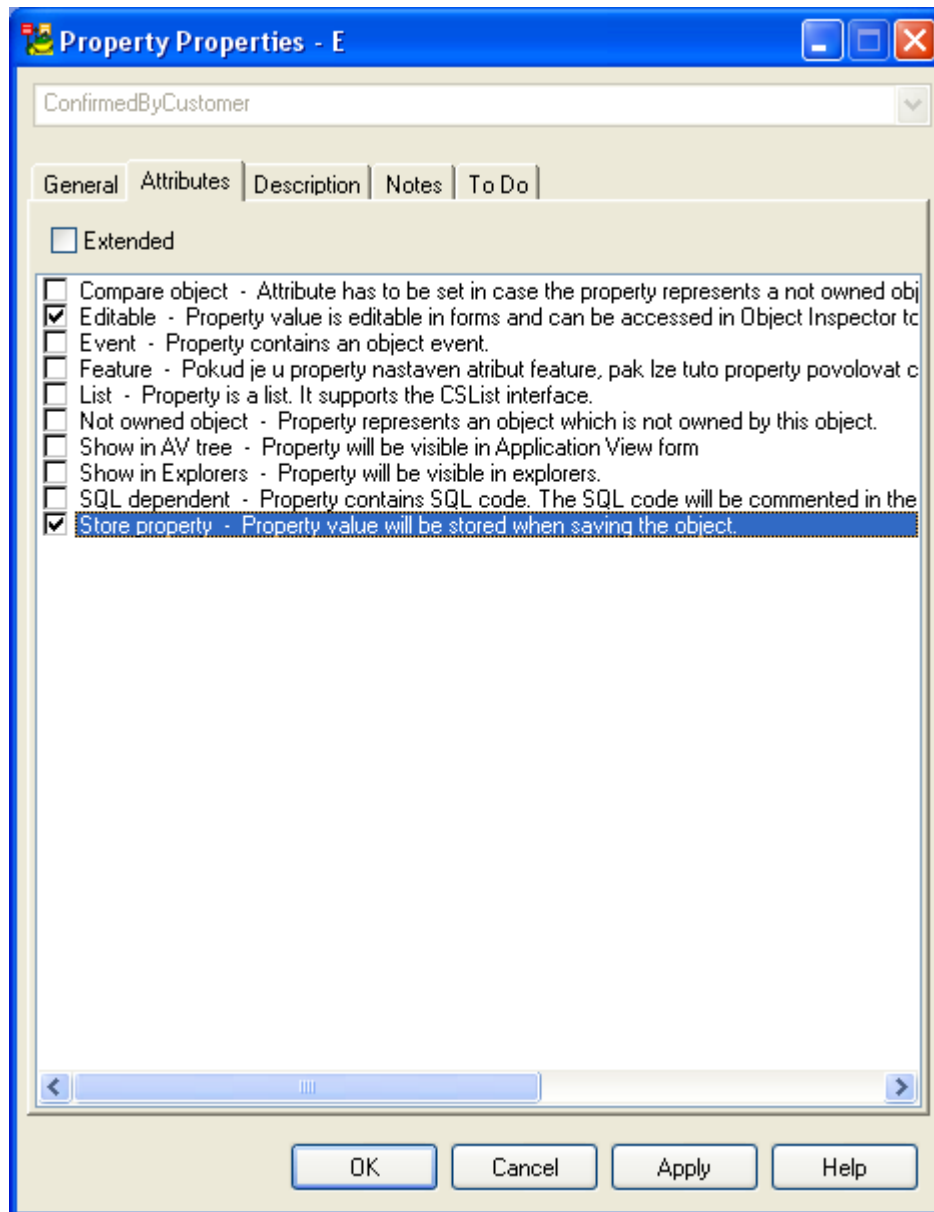
**ConfirmedByCustomer**, data type **Boolean**.



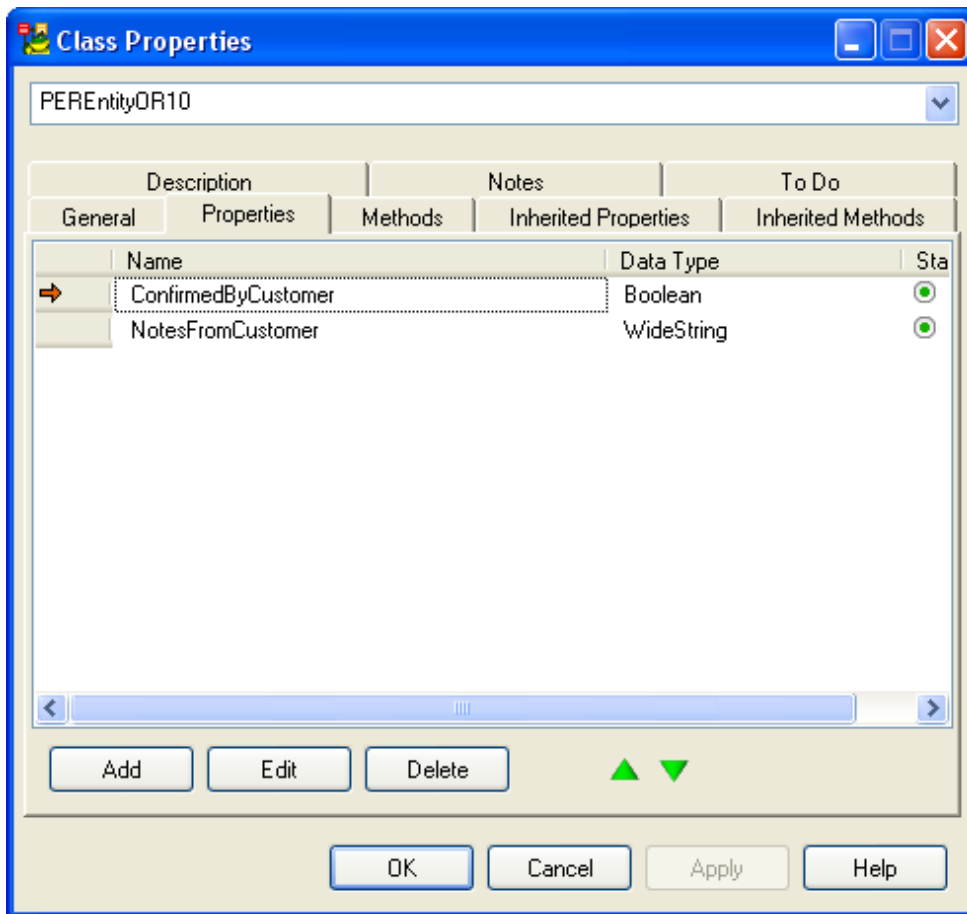
On tab **Attributes** of the **Property Properties** dialog, you can define property Attributes. Select **Editable** (we need to be able to edit the values) and **Store Property** (we want to store the values with model).

Now define the second Property NotesFromCustomer, on tab **Attributes**, select the same attributes - **Editable** and **Store Property**.

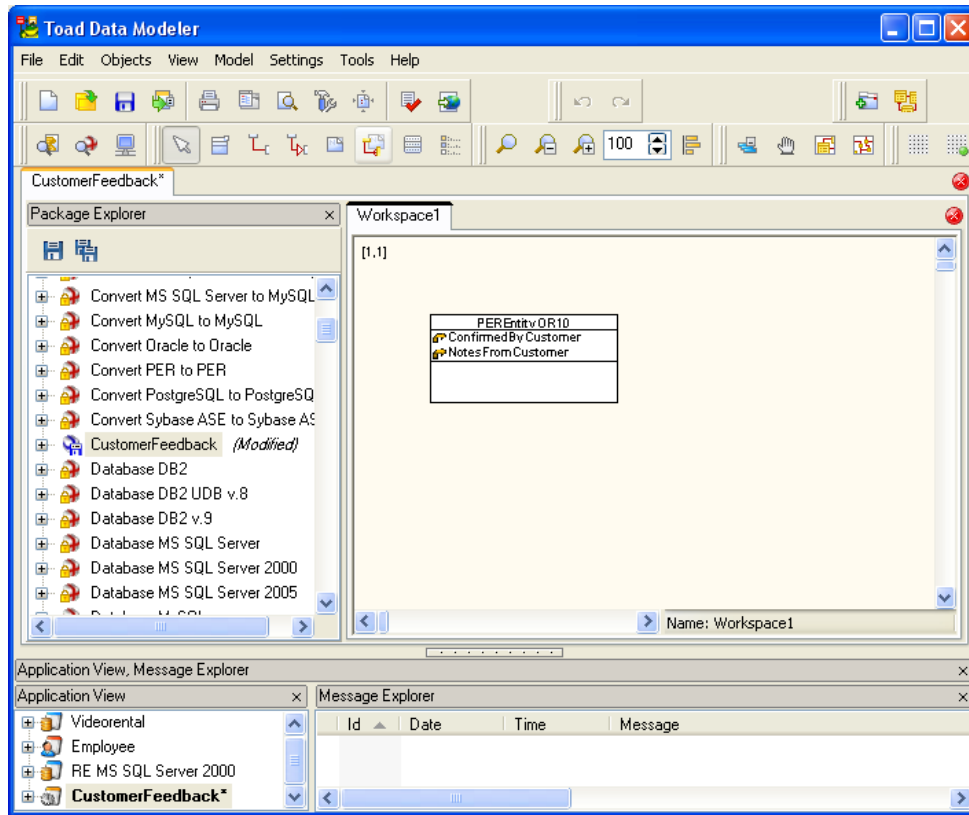




This is how the **Class Properties** dialog should look like after adding the two properties.



And this is how the class is displayed in the metamodel of the CustomerFeedback package.



Properties have been added to the metamodel. **Now it's necessary to save the metamodel and restart the application.** Then we can continue modifying **Entity Properties** form.

**Note:** Metamodels are XML documents stored as .TXM files.

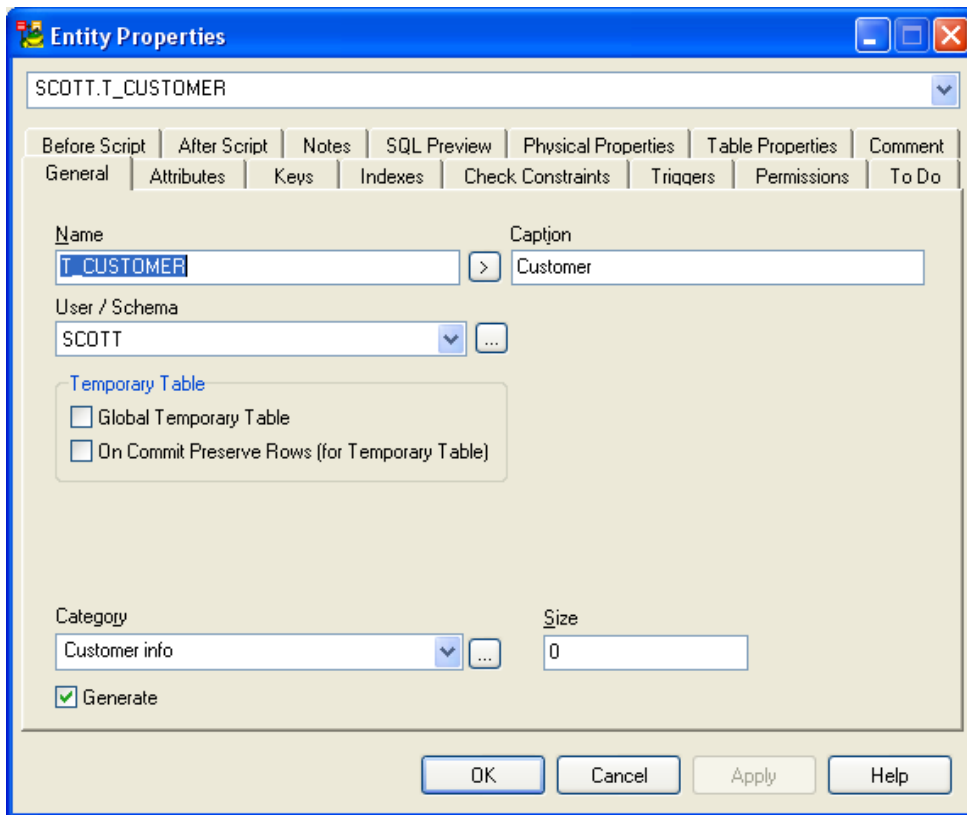
## Related Topics

Modify Form (page 287)

## Modify Form

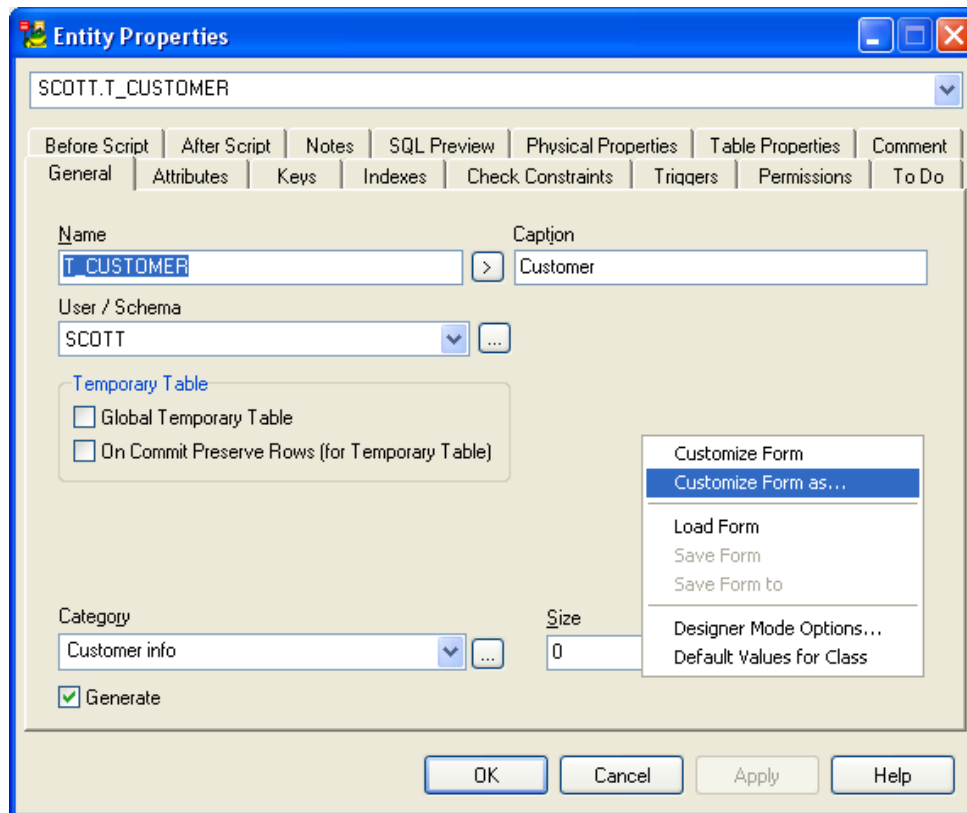
Back to our Oracle 10 physical model.

Edit an entity to open standard **Entity Properties** form.

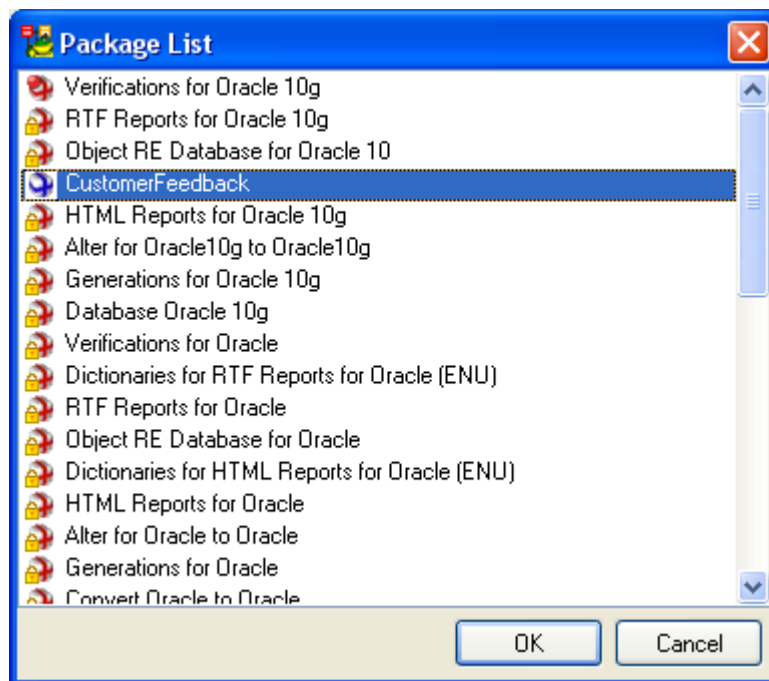


The image shows the 'Entity Properties' dialog box in Toad Data Modeler. The title bar is blue with the text 'Entity Properties' and standard window controls. Below the title bar is a dropdown menu showing 'SCOTT.T\_CUSTOMER'. A tabbed interface follows, with tabs for 'Before Script', 'After Script', 'Notes', 'SQL Preview', 'Physical Properties', 'Table Properties', 'Comment', 'General', 'Attributes', 'Keys', 'Indexes', 'Check Constraints', 'Triggers', 'Permissions', and 'To Do'. The 'General' tab is selected. It contains several fields: 'Name' with a text box containing 'T\_CUSTOMER' and a '>' button; 'Caption' with a text box containing 'Customer'; 'User / Schema' with a dropdown menu showing 'SCOTT' and an ellipsis button; a 'Temporary Table' section with two unchecked checkboxes: 'Global Temporary Table' and 'On Commit Preserve Rows (for Temporary Table)'; 'Category' with a dropdown menu showing 'Customer info' and an ellipsis button; 'Size' with a text box containing '0'; and a 'Generate' checkbox which is checked. At the bottom are four buttons: 'OK', 'Cancel', 'Apply', and 'Help'.

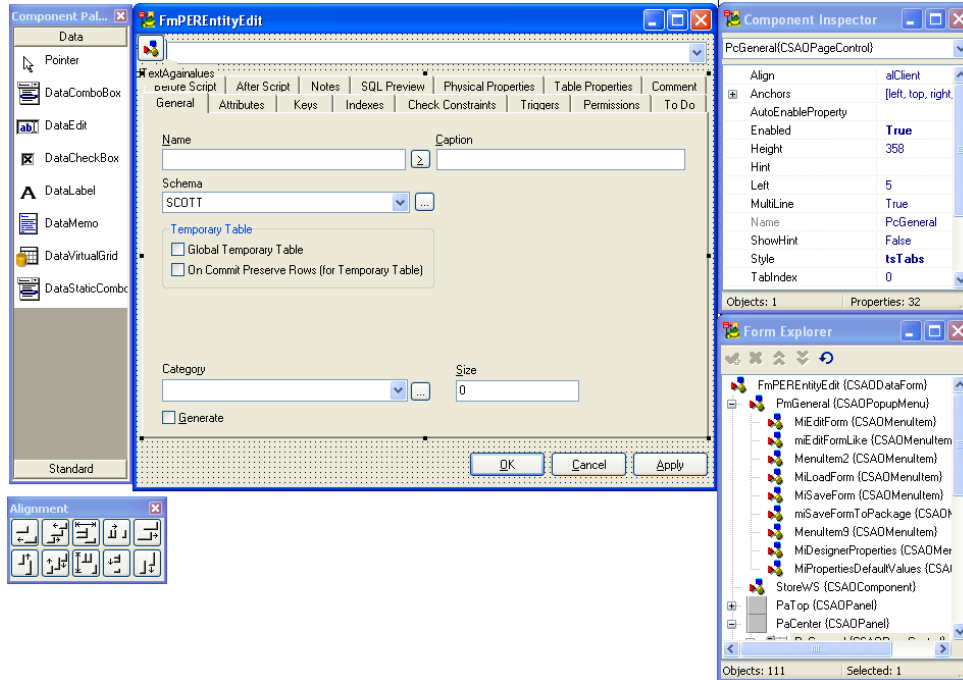
Right-click the form and select **Customize Form as...**



Select a package you want to store modifications in.

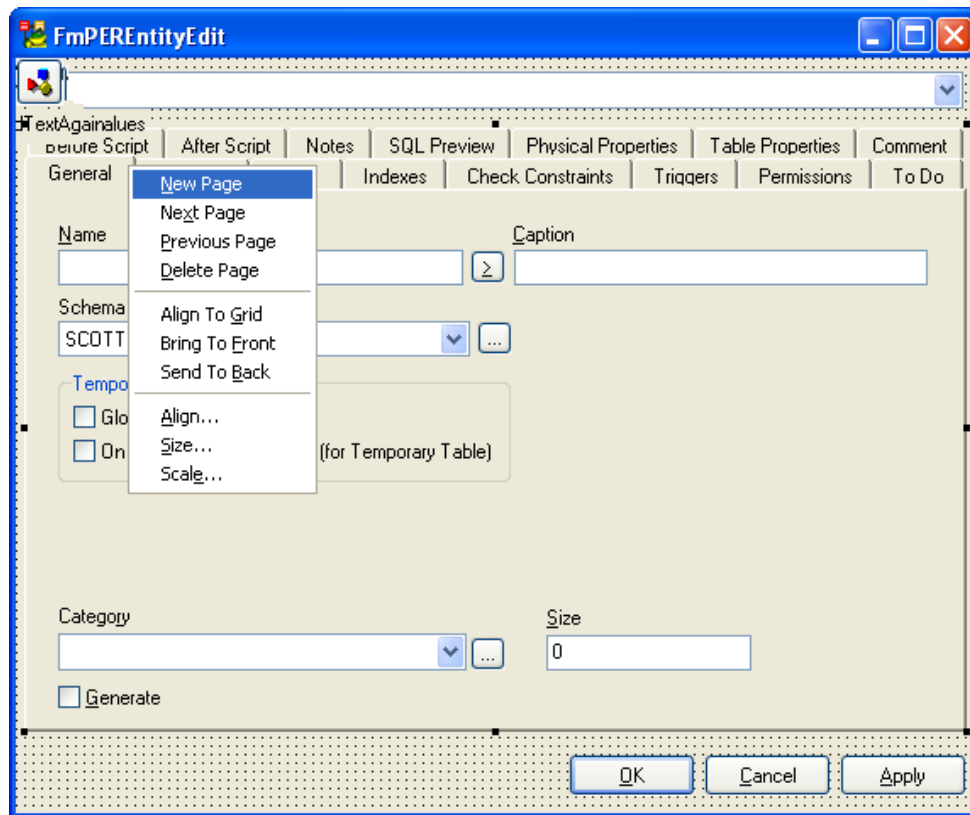


The following forms and palettes will appear. Note that the **Entity Properties** form has dotted grid now:

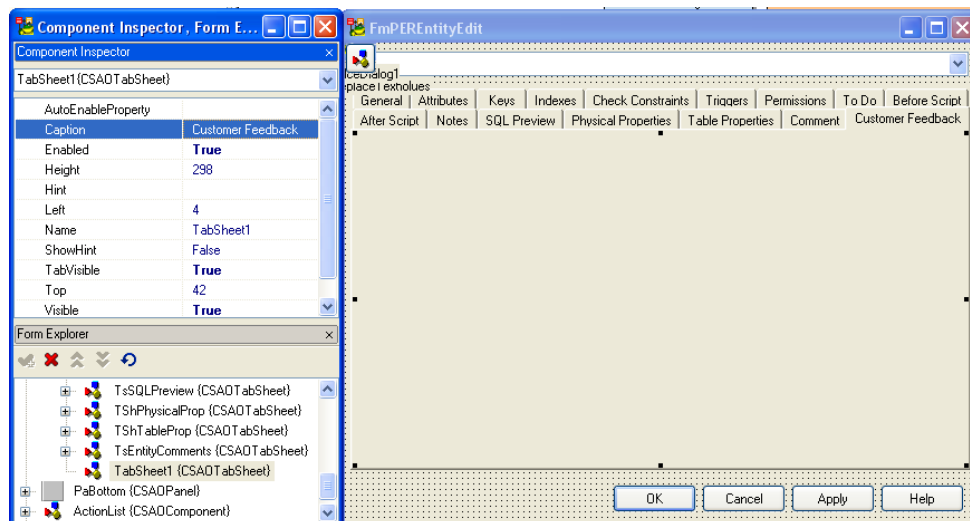


In the **Form Explorer**, see that the form name is **FmPEREntityEdit**. We will need this information later.

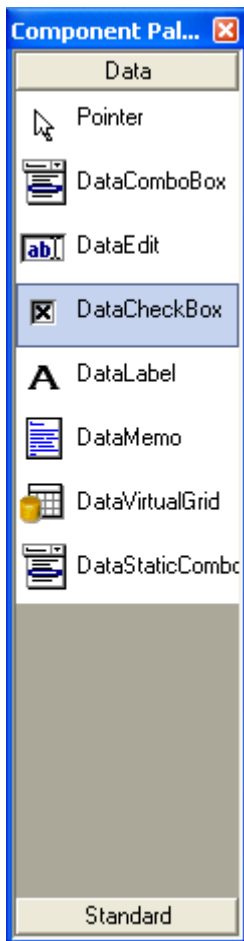
To add a new tab to the Entity form, right-click any tab in the form and select **New Page**.



Define Caption for the new tab in **Component Inspector**.



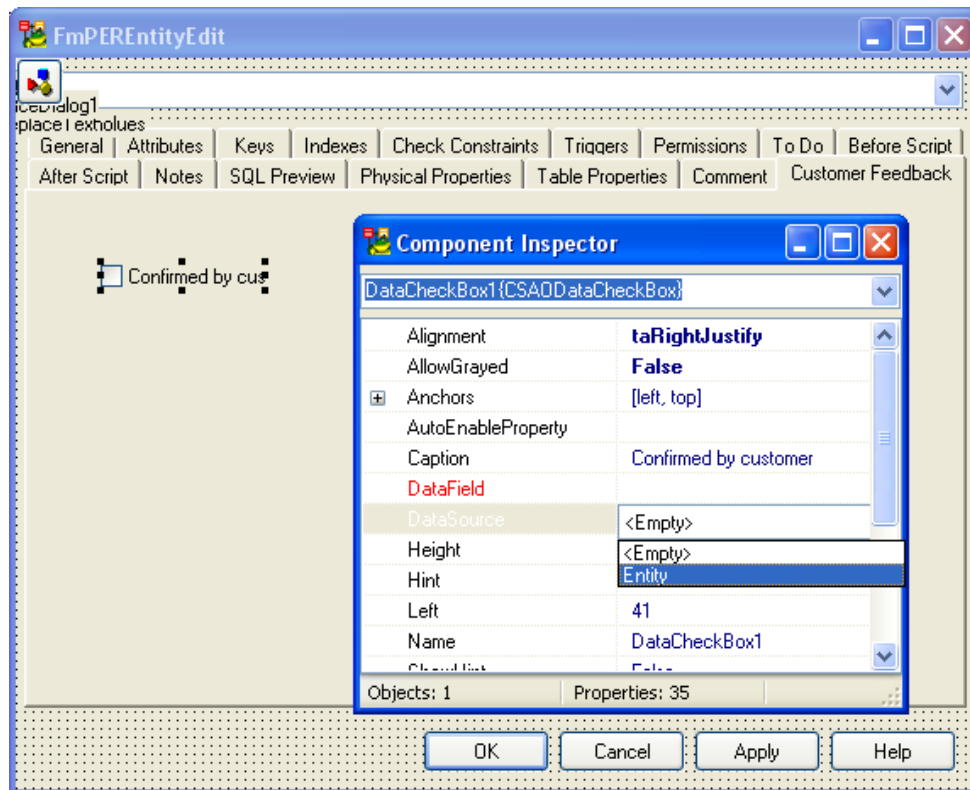
Then select **DataCheckBox** item from the **Component Palette**.



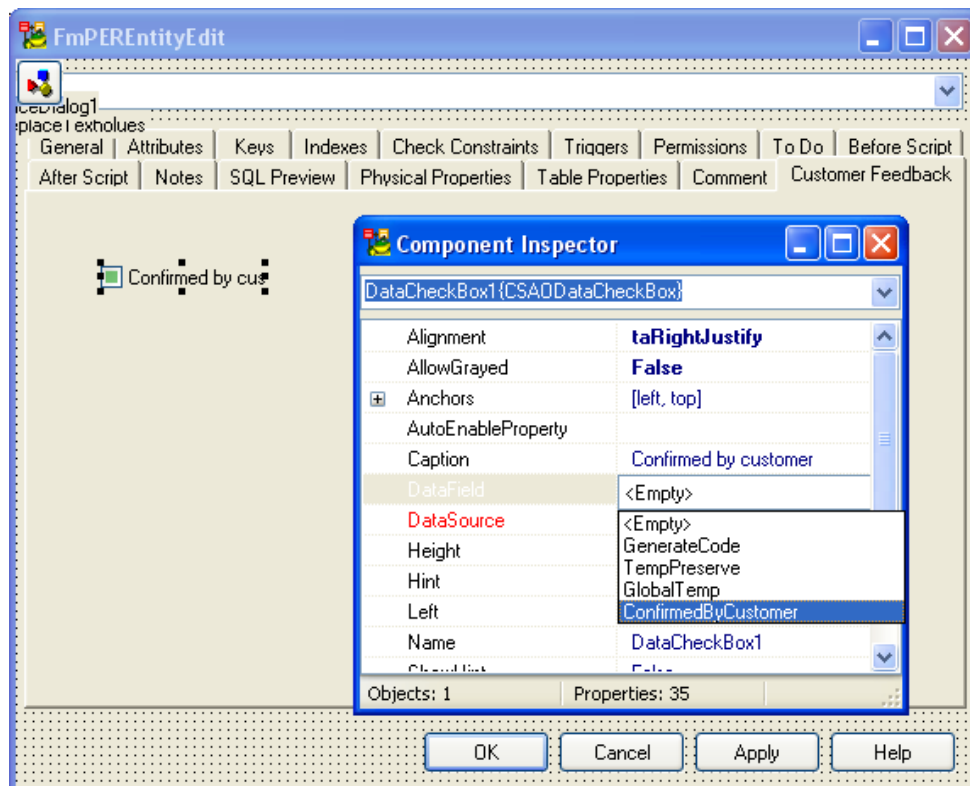
And add a new checkbox to the **CustomerFeedback** tab.

Select the **Entity** item from **DataSource** field in the **Component Inspector**.

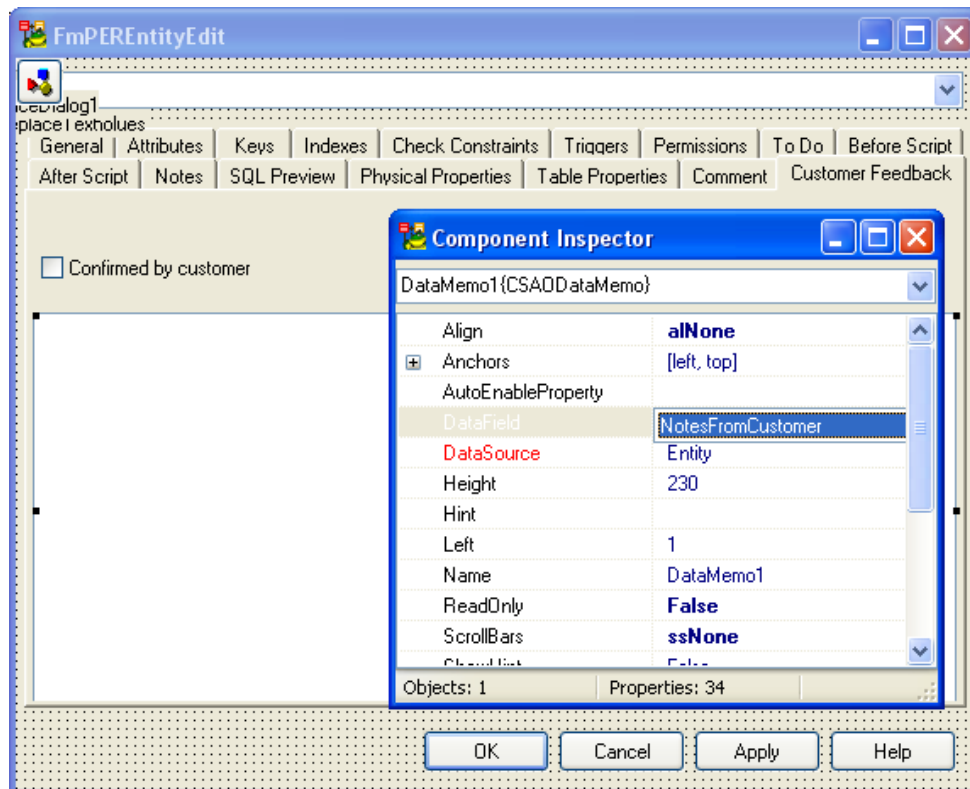




Then select **ConfirmedByCustomer** item from **DataField**.

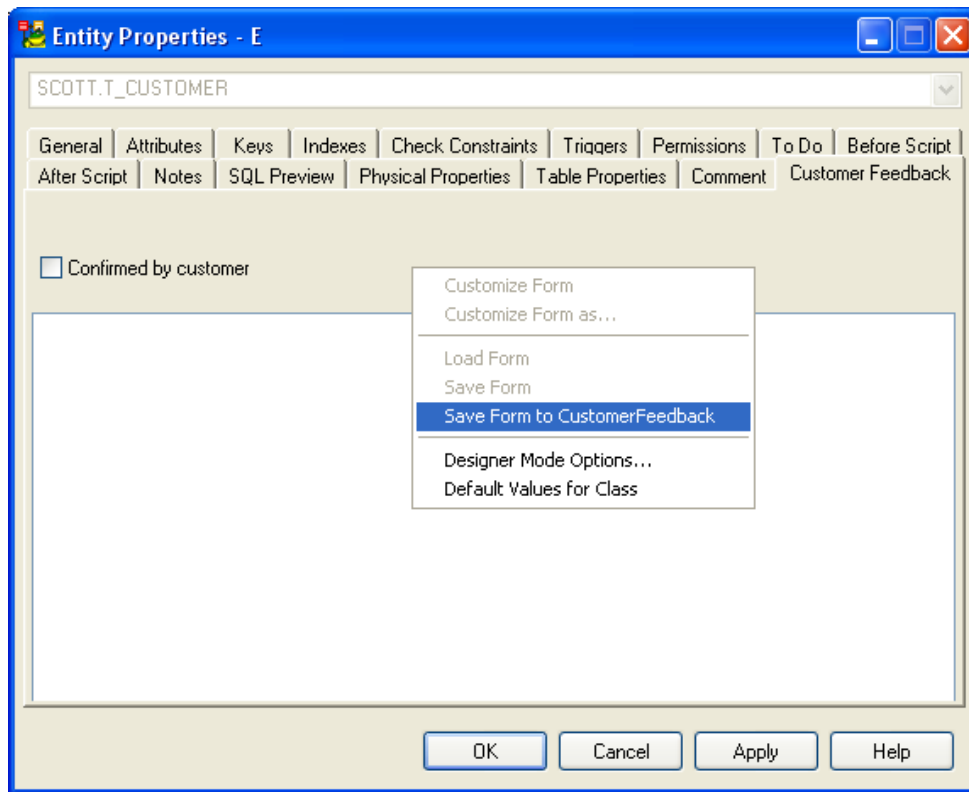


Select **DataMemo** from **Component Palette** and add new text field (datamemo) item to the **Customer Feedback** tab.



Close the **Entity Properties** form by clicking the red X button at top of the form. Component Inspector, Component palette, Form Explorer will disappear.

Then right-click the form and select **Save Form to CustomerFeedback**.



New items are in the form.

## Related Topics

Set Default Values (page 295)

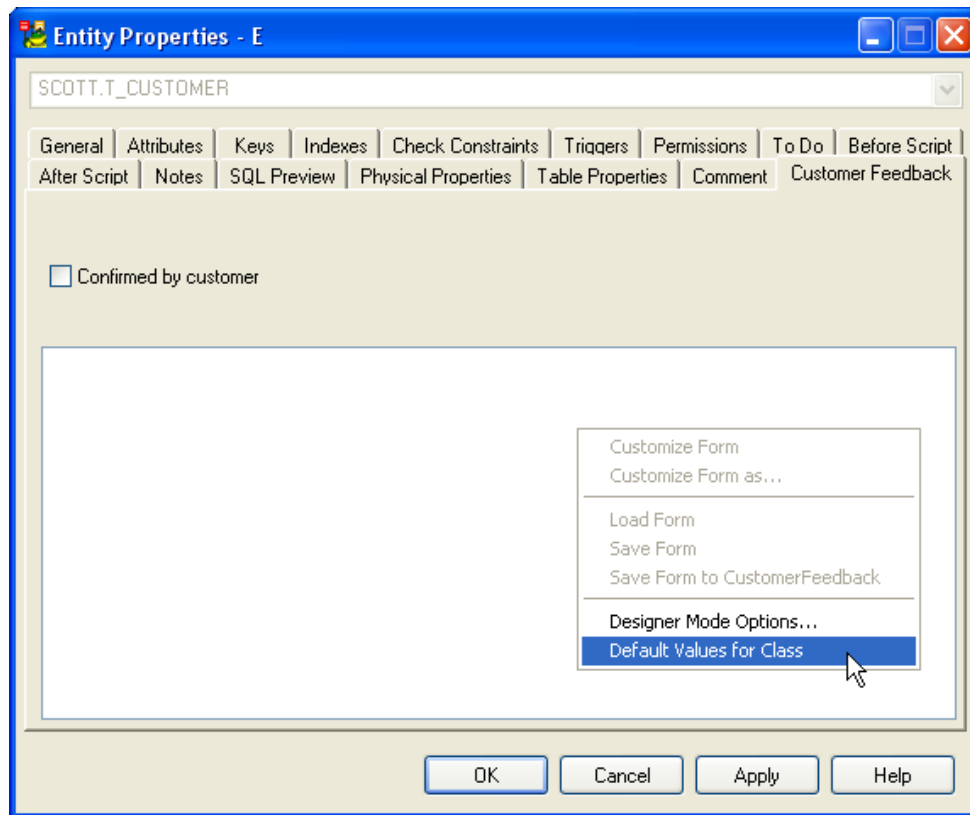
# Set Default Values

## How to Define Default Values

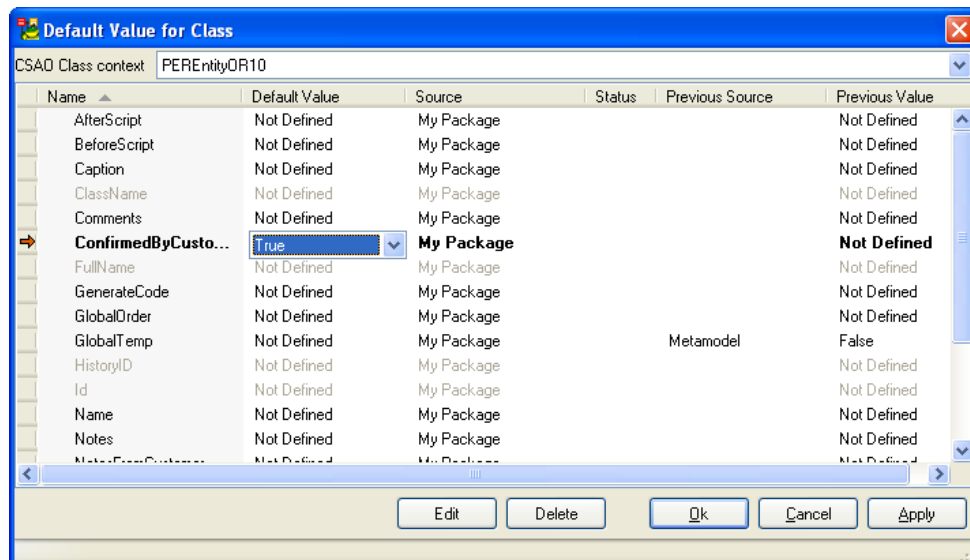
If you need to change the default value for new items, do the following:

Right-click the form (**Entity Properties** form in our example) and select **Default Values for Class**.

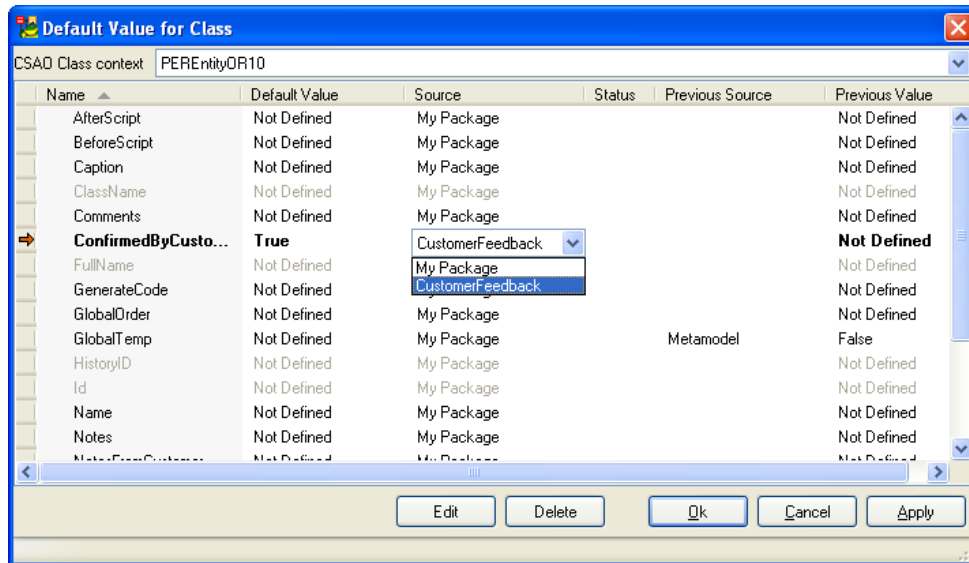
We want the **Confirmed by customer** checkbox to be selected by default for new entities.



Select property name and click the **Default Value** column. Then press F2 to edit the value.



Select where the definition will be stored. In our example, we need to store it into the **CustomerFeedback** package.



Click the dialog to confirm your selection in combo box and then confirm **OK**.

Done.

## Related Topics

Add Events (page 297)

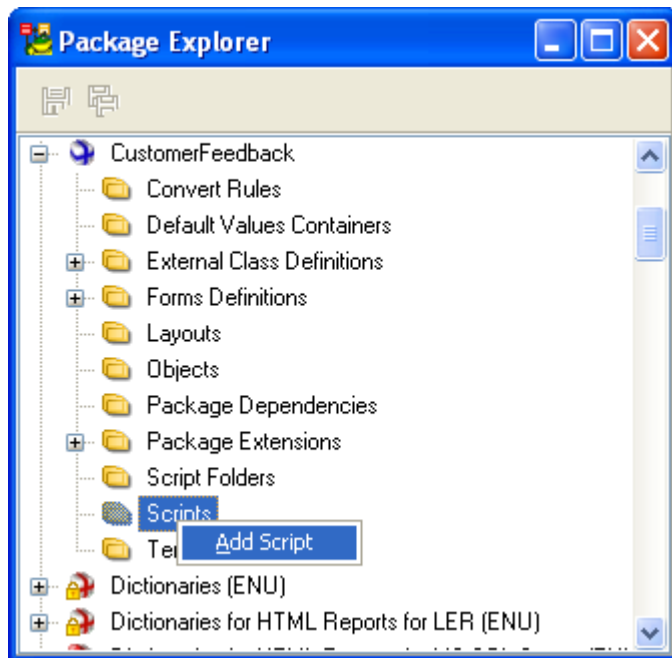
## Add Events

The following events are available in Toad Data Modeler scripting:

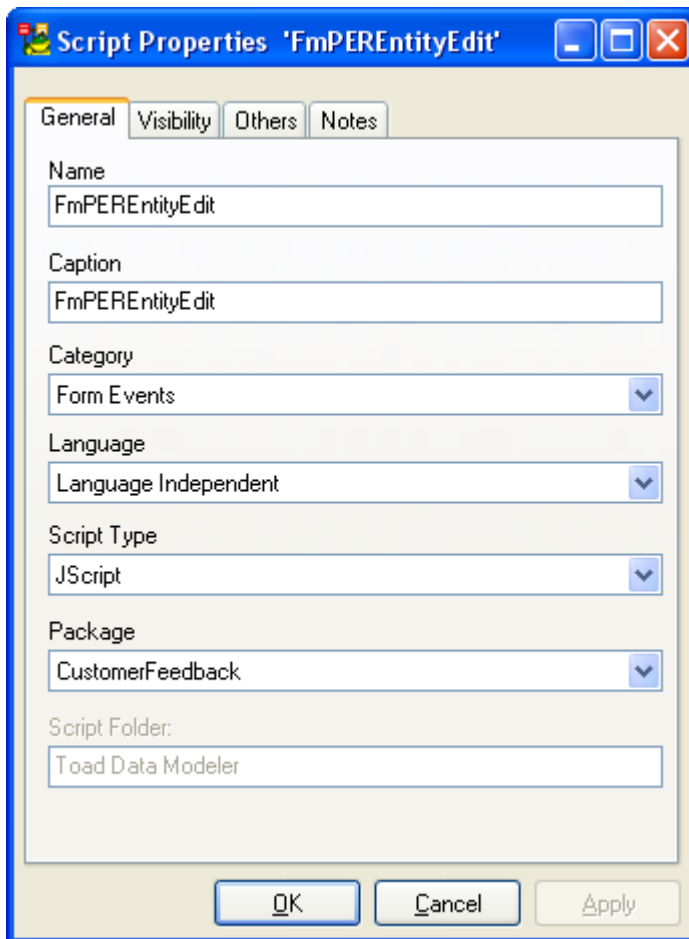
- OnCreate
- OnCheck
- OnChange
- OnClick
- OnClickSilent
- OnClose

Let's add **OnCheck** event to the checkbox on the **Customer Feedback** tab of the **Entity Properties** dialog. When the checkbox is selected, the text box with **Notes from Customer** will be visible. When the checkbox is clear, the text box will disappear.

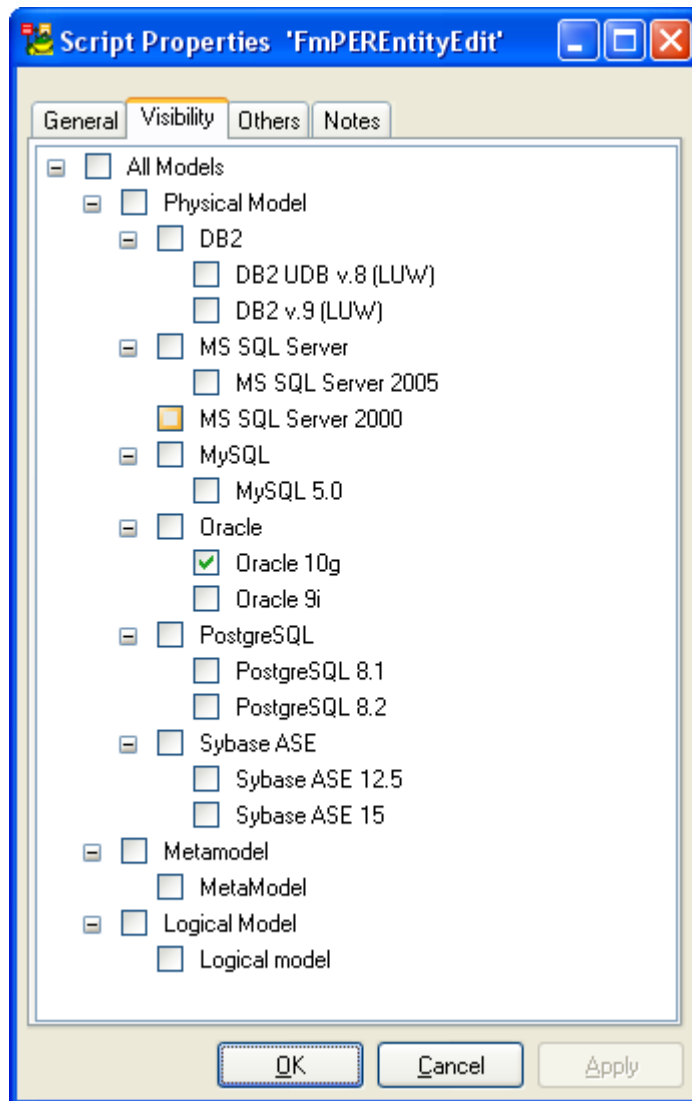
In Package Explorer, select the Script folder under the CustomerFeedback package. Right-click it and select **AddScript**.



Set the script name to **FmPEREntityEdit**. - This is the name of the form we want to write the script for. See the "Modifying a Form" topic to find out where the form name is defined.

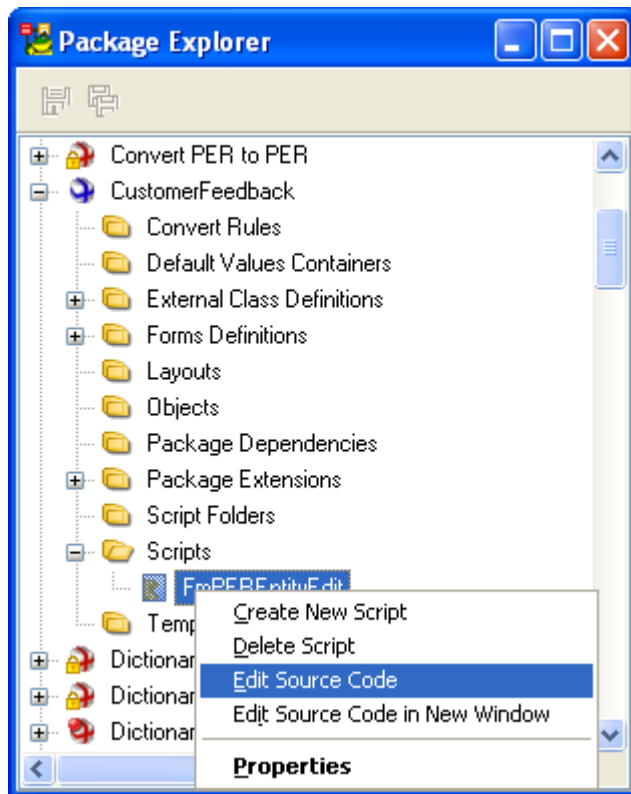


Set visibility to Oracle 10g only.

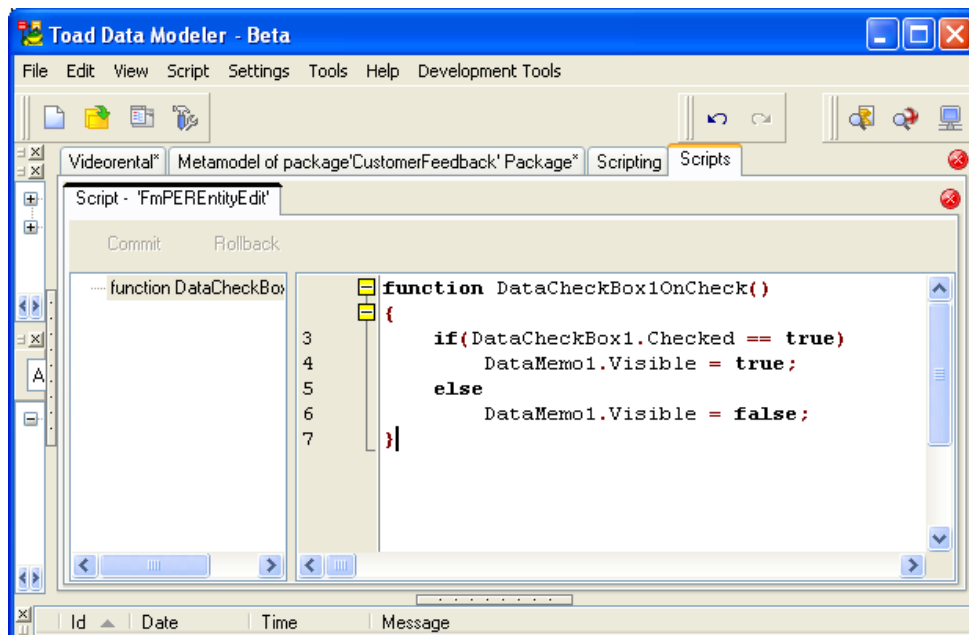


Close the window, right-click the script again and select **Edit Source Code**.





Add there the event function.



Code:

```
function DataCheckBox1OnCheck()
```

```
{  
    if(DataCheckBox1.Checked == true)  
        DataMemo1.Visible = true;  
    else  
        DataMemo1.Visible = false;  
}
```

Explanation of items in bold.

#### **DataCheckBox1OnCheck()**

- DataCheckBox1 - name of item that has been added to the Entity form.
- OnCheck - name of event.

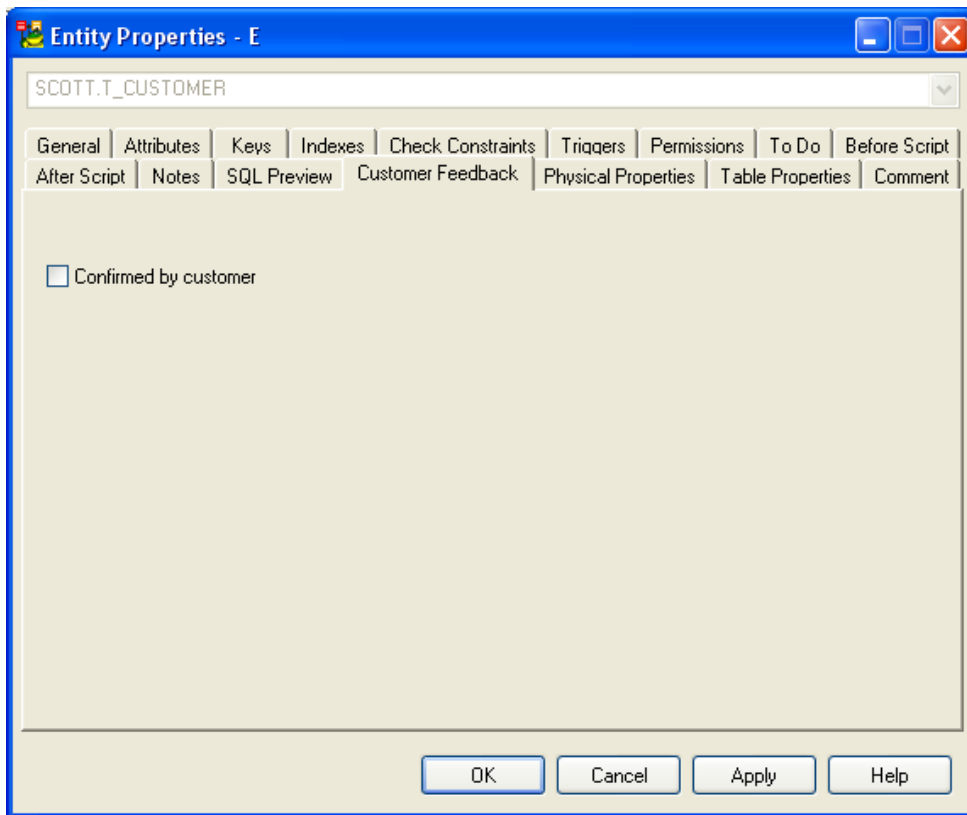
#### **DataCheckBox1** and **DataMemo1**

- Both are names of items that were added to the Entity form. See the [Modifying a Form](#) topic.

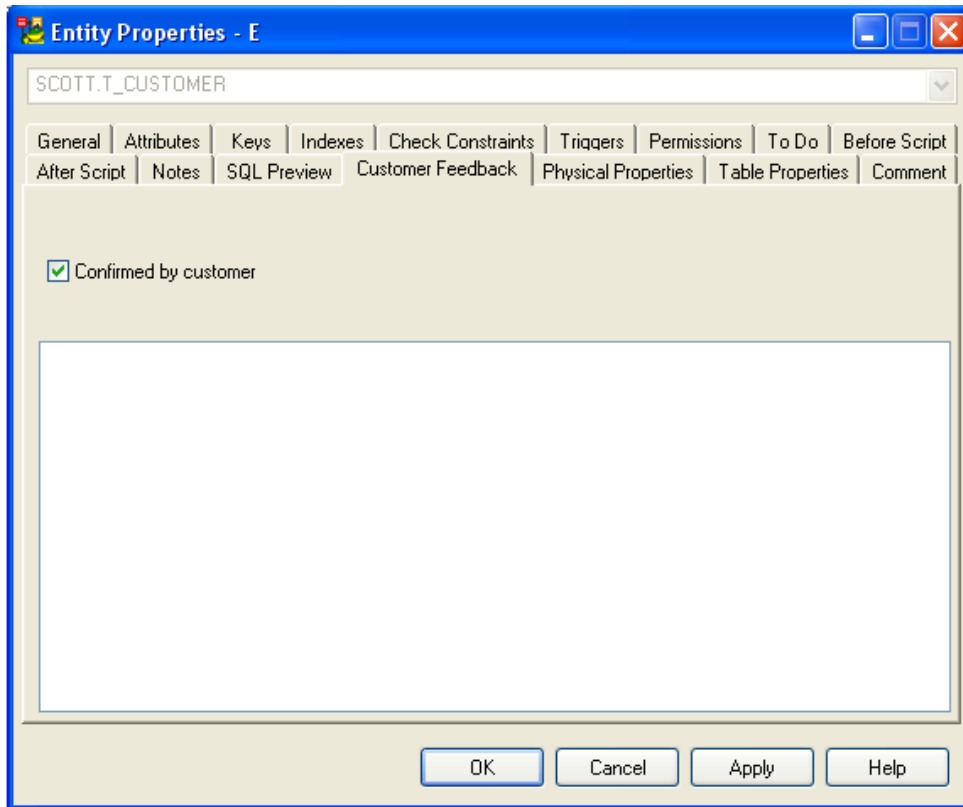
Press **Commit** to confirm the script.

Result:

When the checkbox is clear, the text area is hidden.



If you select the checkbox, the text area will display.



## Related Topics

Access Property Values via Scripting Window (page 304)

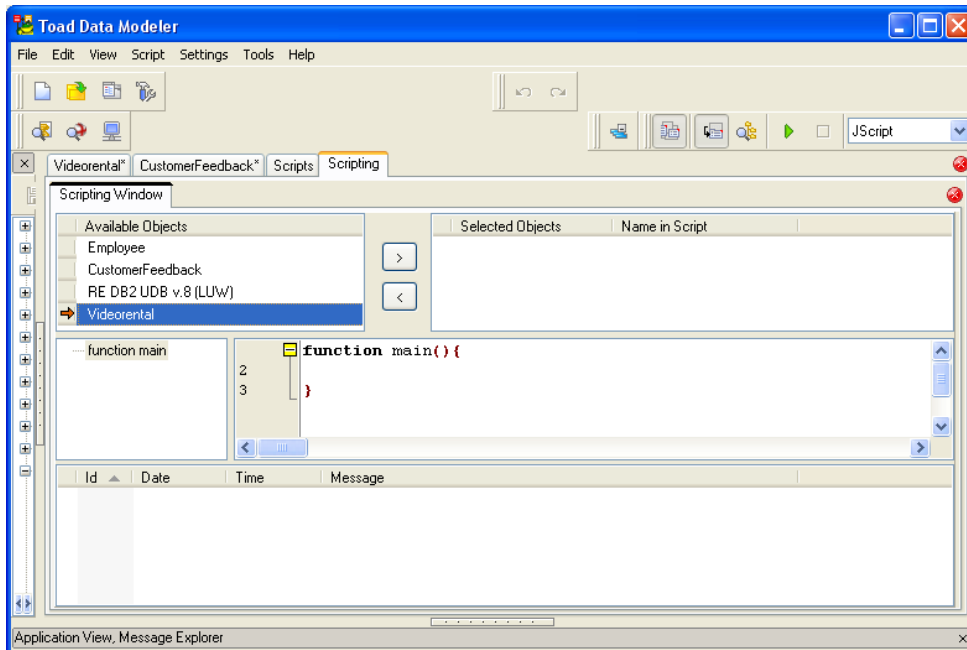
## Access Property Values via Scripting Window

You can write scripts in Toad Data Modeler, save the scripts to packages, distribute the packages etc. - This will be explained later. Now you will see how to work with **Scripting Window** that allows you to run scripts at once, without the necessity to have them stored in packages.

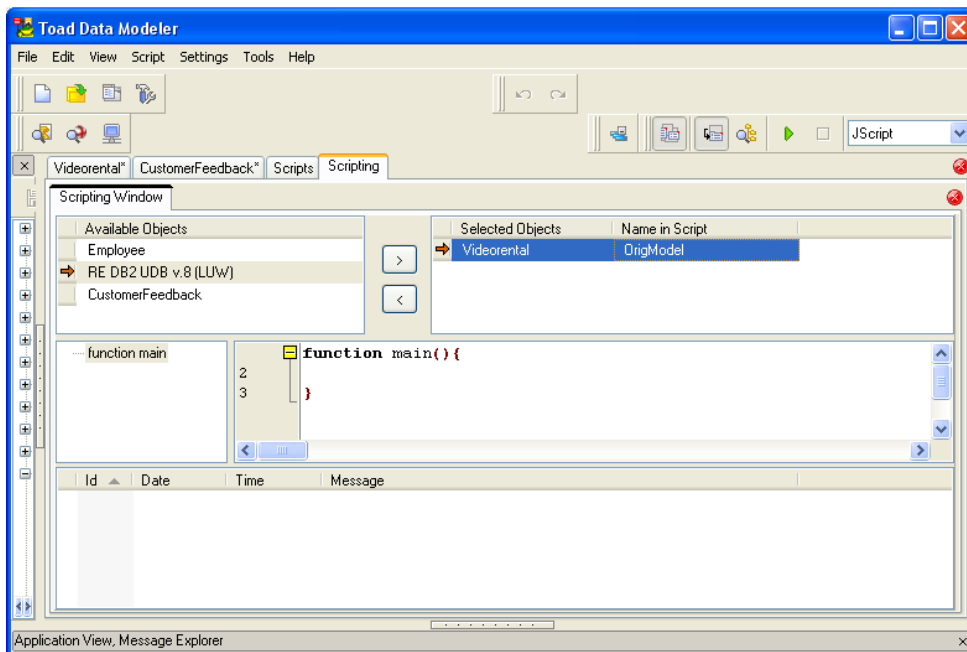
Click **Tool | Scripting Window** to open it. (Of course, Expert Mode has to be turned on.)

The following dialog appears. If you don't see the upper part of the **Scripting Window**, select **View | Show Registered Objects**.

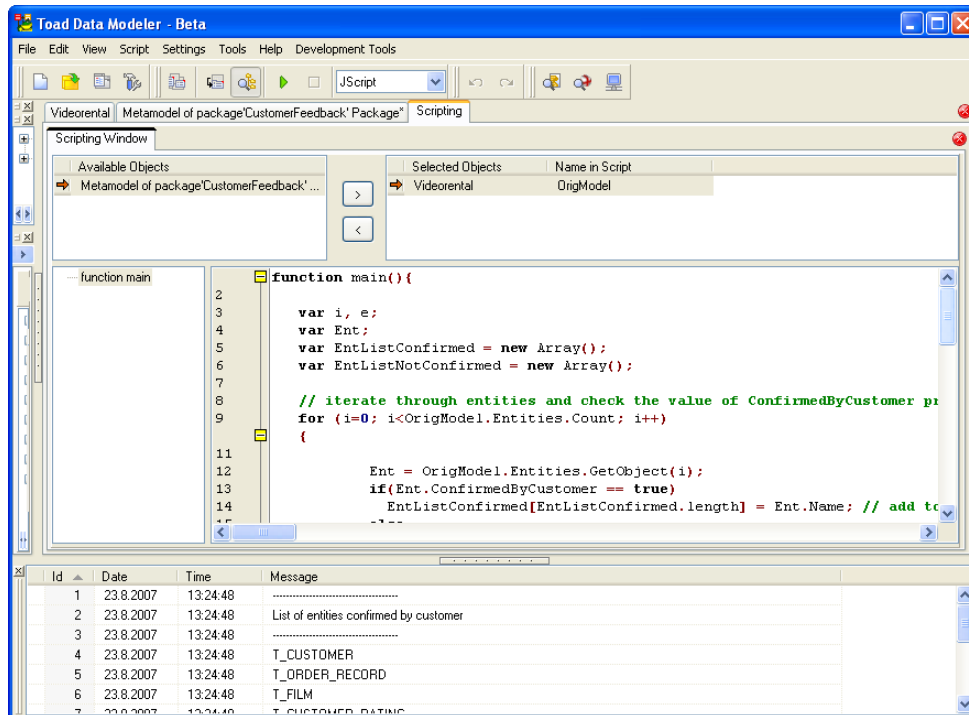
On the left, you can see available models. Use the arrows to select model you want to work with. In our example, we will execute script for *Videorental model* (for Oracle 10g).



In the **Name in Script** column, you can define name that will be used in the script. Our OrigModel value will represent the selected Videorental model.



Write script to the main() function.



## Code:

```
function main() {

    var i, e;

    var Ent;

    var EntListConfirmed = new Array();

    var EntListNotConfirmed = new Array();


    // iterate through entities and check the value of ConfirmedByCustomer
    property

    for (i=0; i<OrigModel.Entities.Count; i++)

    {

        Ent = OrigModel.Entities.GetObject(i);

        if(Ent.ConfirmedByCustomer == true)

            EntListConfirmed[EntListConfirmed.length] = Ent.Name; // add to
            list of confirmed entities
    }
}
```

```
        else

            EntListNotConfirmed[EntListNotConfirmed.length] = Ent.Name; //
add to list of not confirmed entities

    }

    // write list of confirmed entities to Log.
    Log.Information ("-----");
    Log.Information ("List of entities confirmed by customer");
    Log.Information ("-----");
    for (e=0; e<EntListConfirmed.length; e++)
    {
        Log.Information(EntListConfirmed[e]);
    }

    Log.Information ("# Number of confirmed entities:
"+EntListConfirmed.length.toString());

    // write list of NOT confirmed entities to Log.
    Log.Information ("-----");
    Log.Information ("List of entities NOT confirmed by customer");
    Log.Information ("-----");
    for (e=0; e<EntListNotConfirmed.length; e++)
    {
        Log.Information(EntListNotConfirmed[e]);
    }

    Log.Information ("# Number of NOT confirmed entities:
"+EntListNotConfirmed.length.toString());

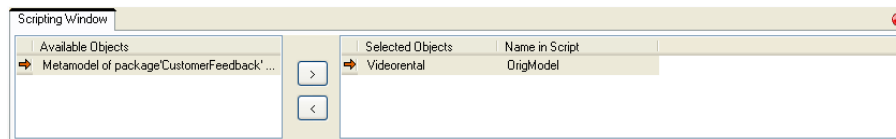
}
```

## Where to find information about objects and their properties and methods?

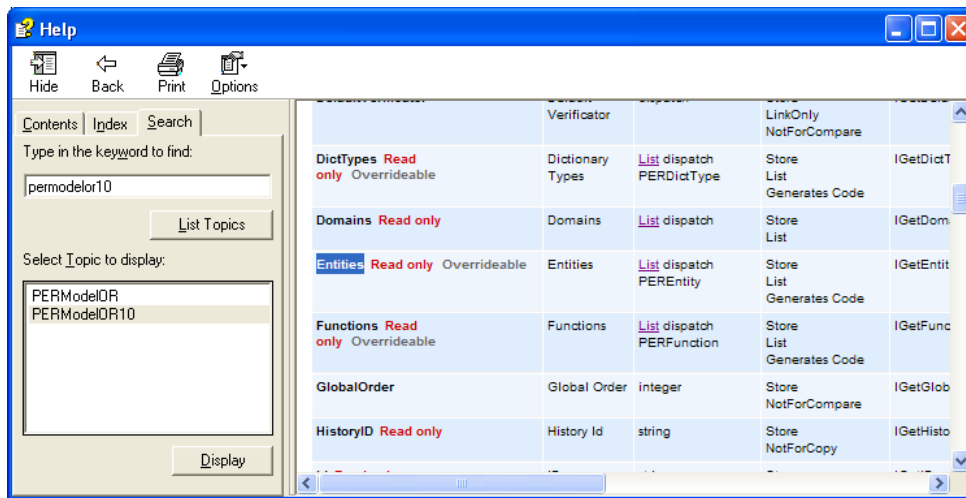
### Explanation of Items in Bold:

#### OrigModel.Entities.Count

- **OrigModel** - represents object assigned in the upper part of the Scripting Window (Videorental object renamed to OrigModel).

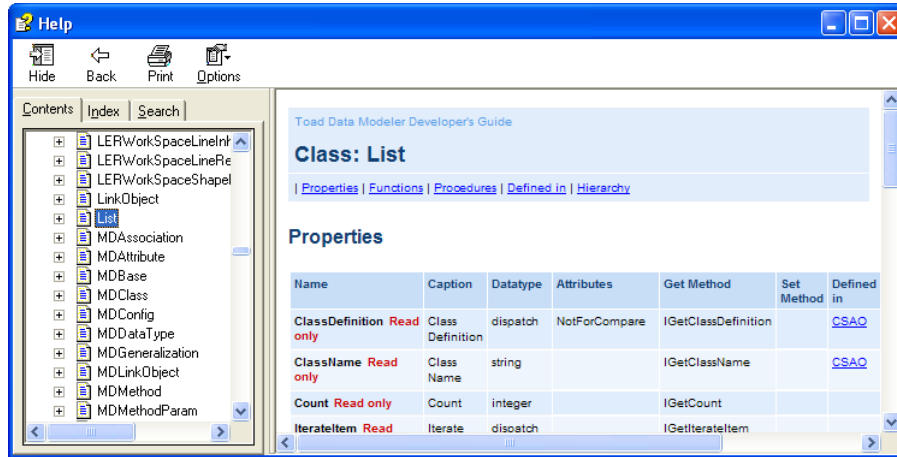


- **Entities** - we work with Physical Entity Relationship model, therefore we need to search for PER object. Model is for Oracle 10g, let's find the PERModelOR10 object in the Reference.



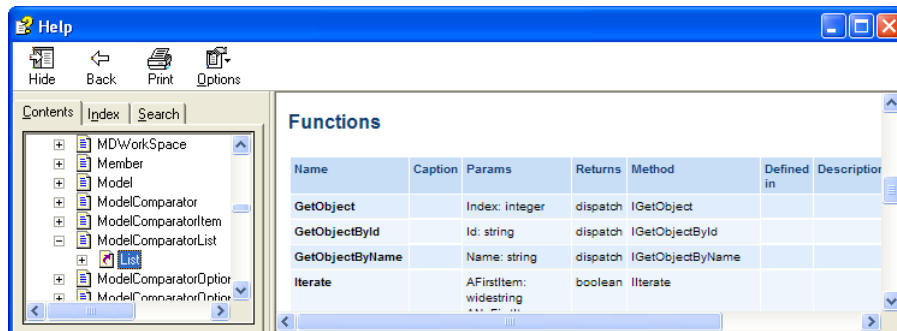
- **Count** - represents a feature that is available for all List objects. On the screenshot above, you can see that the Entities datatype is a List. Let's click the List link and see details of the List class.





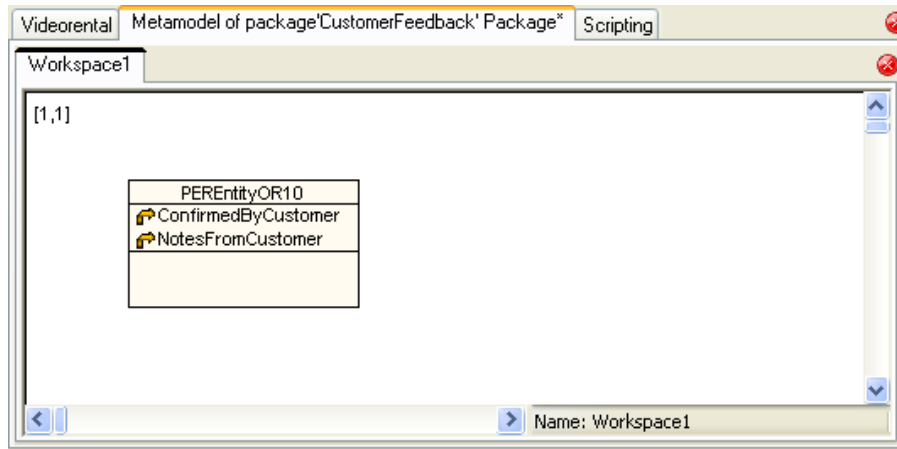
`OrigModel.Entities.GetObject(i)`

- **GetObject** - belongs to the List class.



`Ent.ConfirmedByCustomer`

- **Ent** - is a variable that holds assigned Entity objects (assigned earlier using the `OrigModel.Entities.GetObject(i)` function).
- **ConfirmedByCustomer** - property of PEREntityOR10 object, added to Metamodel of the **CustomerFeedback** package.



#### Ent.Name

- **Ent** - is a variable that holds assigned Entity objects (assigned earlier using the `OrigModel.Entities.GetObject(i)` function).
- **Name** - property of PEREntityOR10 object. We still work with PER model and now we need to find property of Entity in Oracle 10g model. Let's see properties of the PEREntityOR10 object.

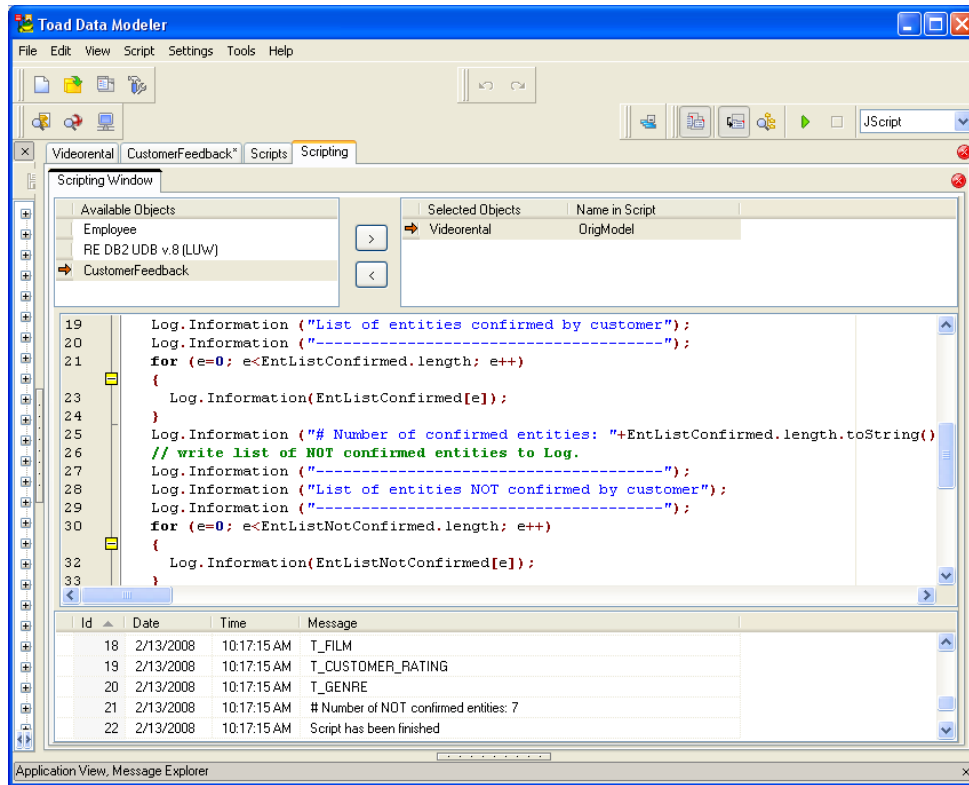


#### .length and .toString()

- both are standard JavaScript items.

## Executing the Script

Click **Execute Script** . Result will be displayed in the **Message Explorer** and **Log** area.



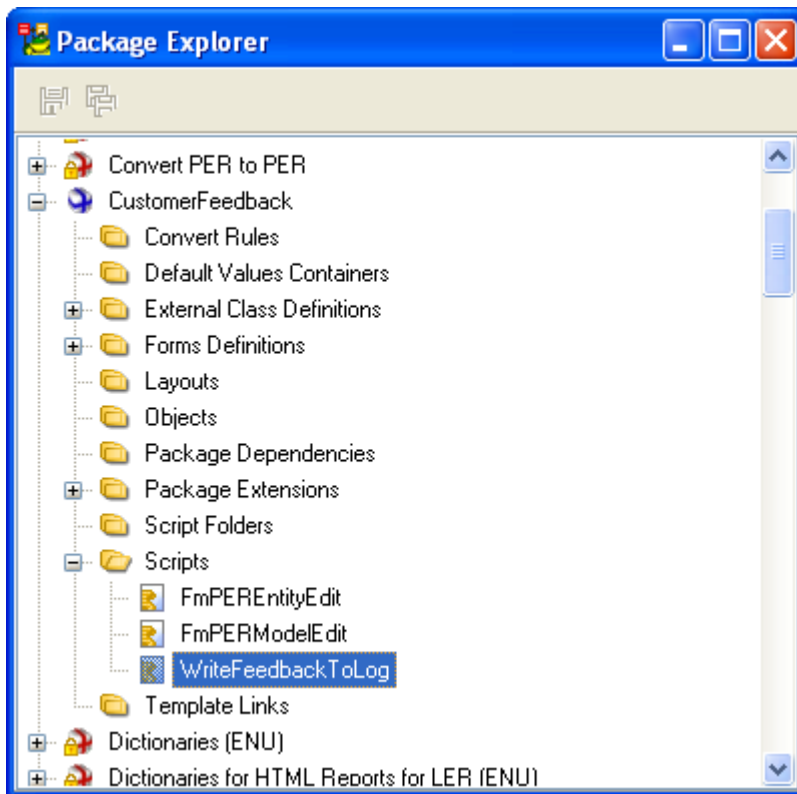
## Related Topics

Create Script (page 311)

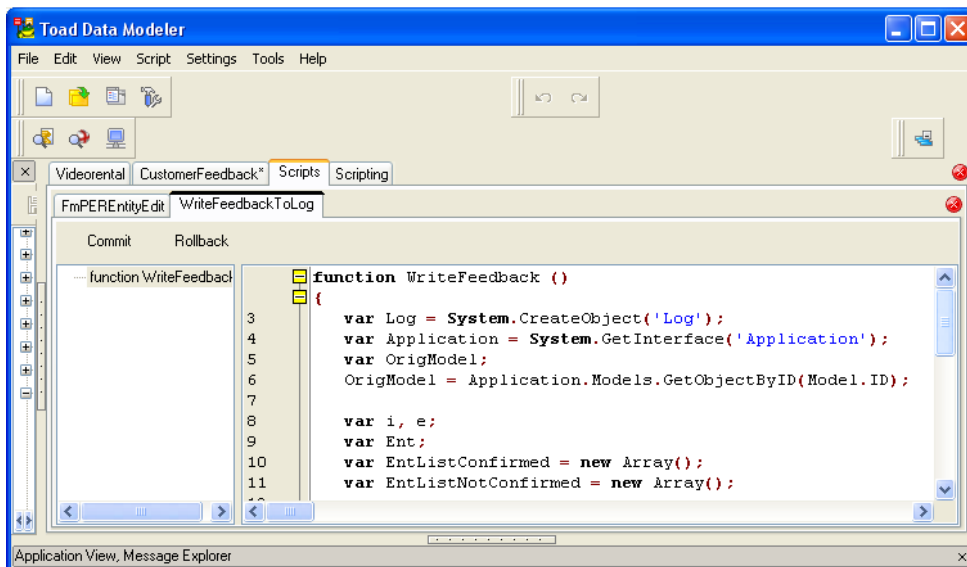
## Create Script

You know how to execute scripts from the Scripting Window. If you want to store the script and call it from another form in the application, for example, do the following:

Create a new script WriteFeedbackToLog. See the "Adding Events" topic to find out how to create new scripts.



Write there function WriteFeedback.



## Code

```
function WriteFeedback ()  
{
```

```
var Log = System.CreateObject('Log');  
  
var Application = System.GetInterface('Application');  
  
var OrigModel;  
  
OrigModel = Application.Models.GetObjectByID(Model.ID);  
  
....  
  
....  
  
}
```

## Explanation

The WriteFeedback function is almost identical to the Main function we were executing from the Scripting Window.

The only difference is in the definition of OrigModel object. In the **Scripting Window**, we could select Videorental and define the OrigName name.

However, now we have no means to select the object visually (and we do not need it, the function will be executed for active model). Therefore we need to define the OrigModel object via Application.Models.GetObjectByID method, with parameter Model.ID.

This way we can get the currently active model.

We also need to register object Log. (It is not necessary to register Log in the Scripting Window. Log is registered in the Scripting Window automatically.)

The rest of the script is identical.

## Related Topics

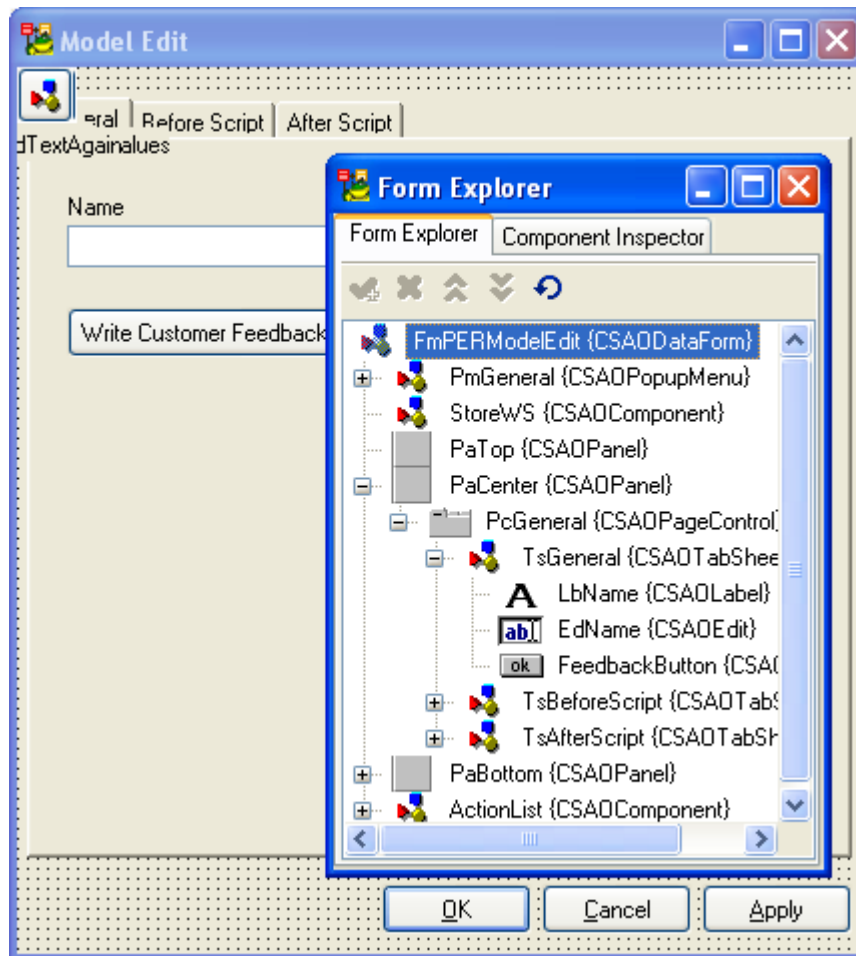
Call Existing Script from Model Properties Form (page 313)

## Call Existing Script from Model Properties Form

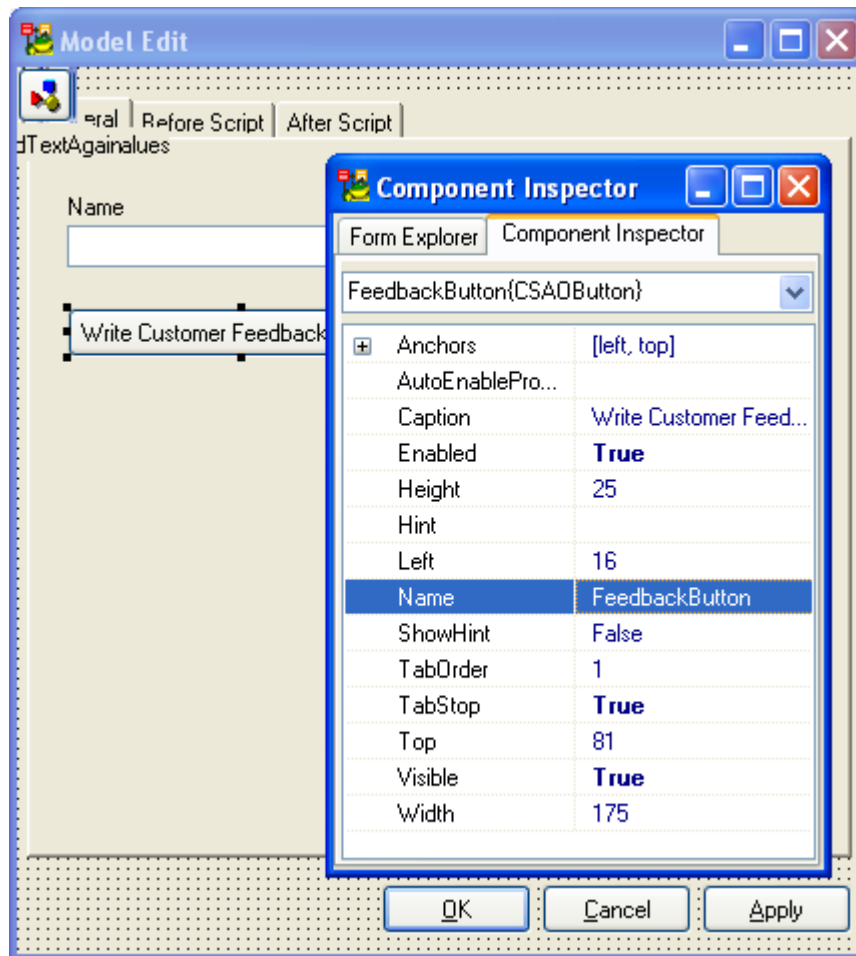
Edit the **Model Properties** form. See the "Modify Form" topic to find out how to edit existing form.

Add there a new button and remember the name of the form - **FmPERModelEdit**. The name can be found in the **Form Explorer**.

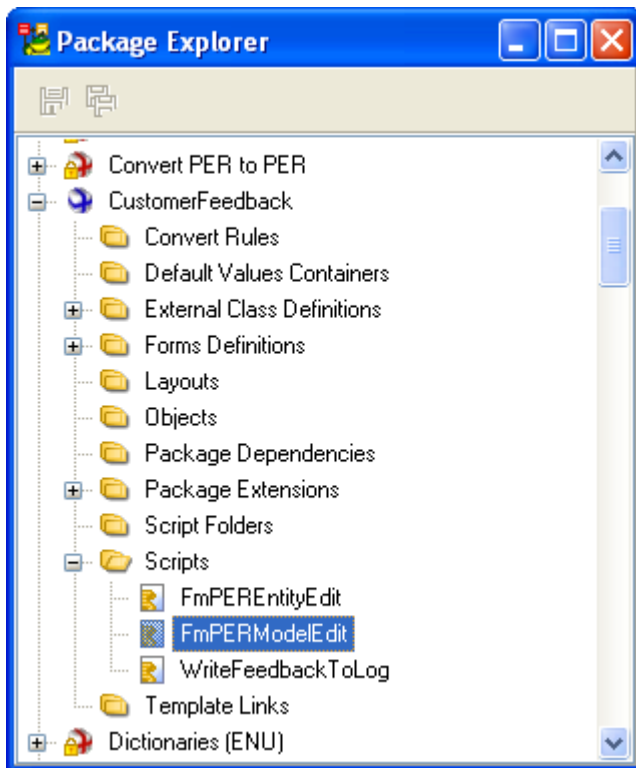
Defined caption for the button - **Write Customer Feedback To Log**.



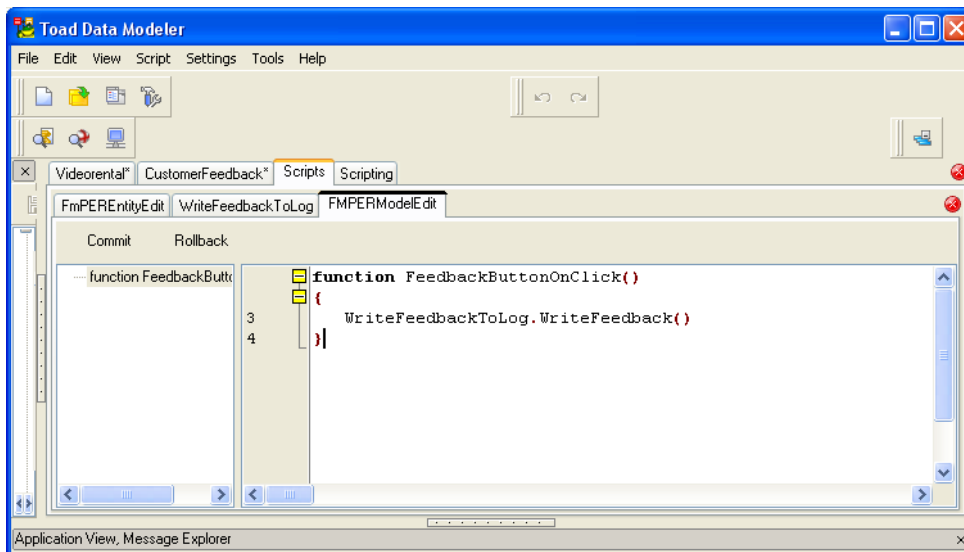
Set the name of the button to **FeedbackButton**.



Create a new script with the name of the **Model Properties** form - **FmPERModelEdit**.



Write event function to the script.



## Code

```
function FeedbackButtonOnClick()  
{  
    WriteFeedbackToLog.WriteFeedback()  
}
```

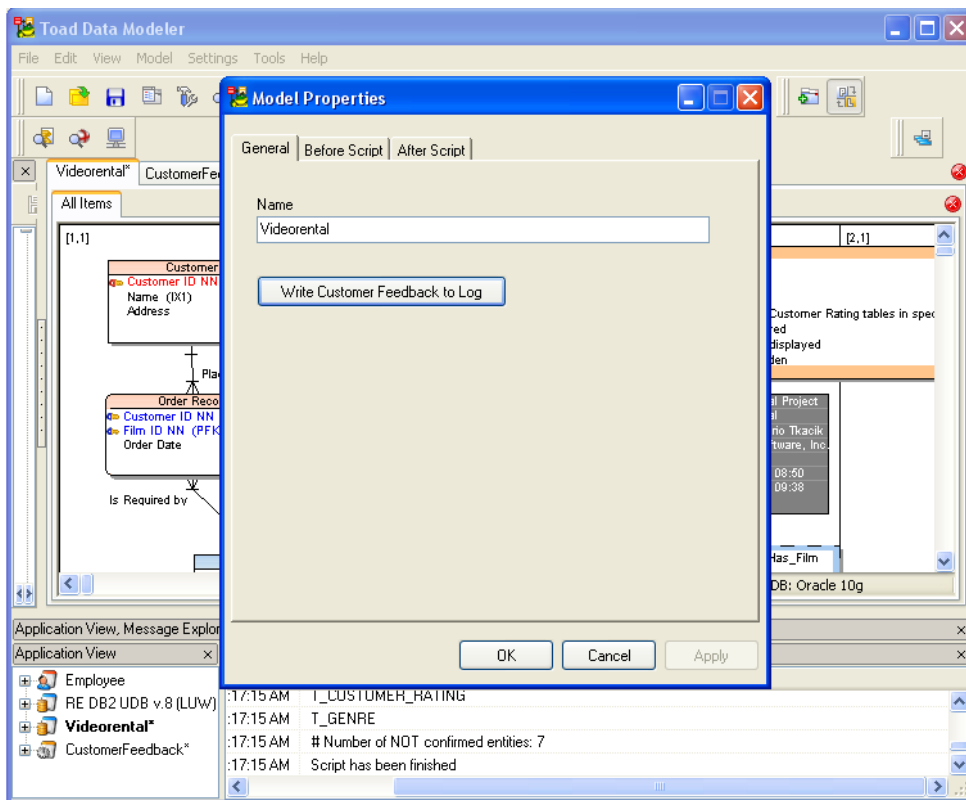


```
}
```

## Explanation

- **FeedbackButton** = name of the button.
- **OnClick** = event.
- **WriteFeedbackToLog** = name of script that contains called function.
- **WriteFeedback** = called function.

When you click the button, an output will be displayed in **Message Explorer (Log)**.



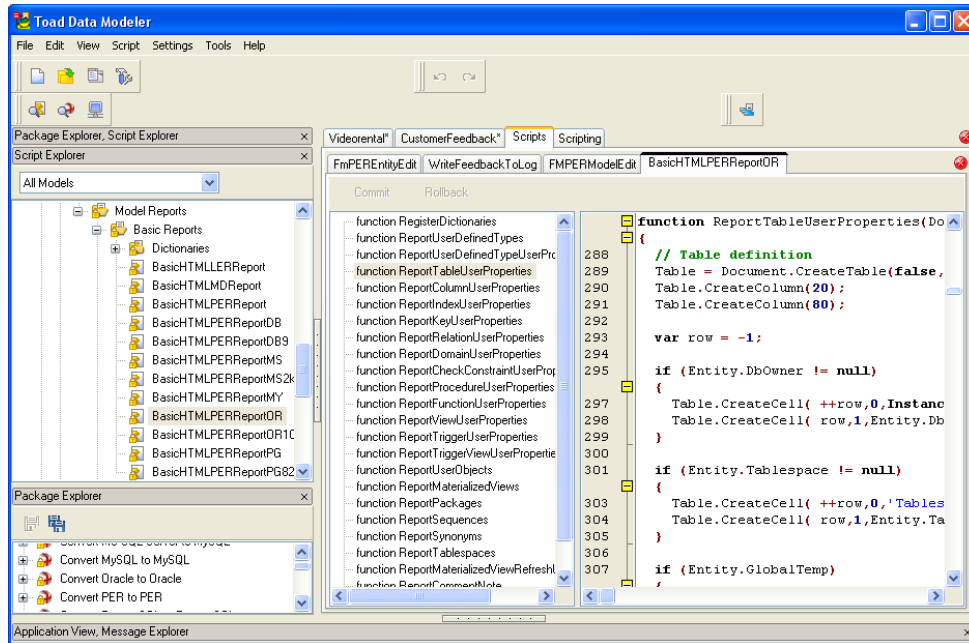
## Related Topics

Modify HTML Reports (page 317)

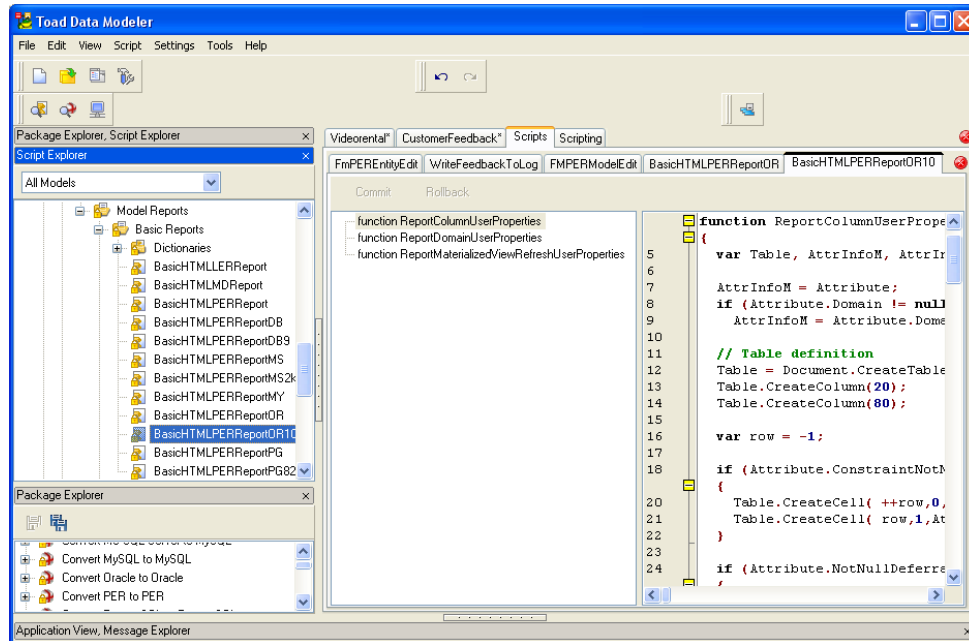
## Modify HTML Reports

To modify HTML reports, we need to extend existing method. The first thing we need to do is to find out what script should be extended.

In **Script Explorer**, you can see **BasicHTMLPERReportOR** script with function **ReportTableUserProperties**. This is the script that generates Tables pages in HTML reports, specifically the section **Table Properties**.



You can also see script **BasicHTMLPERReportOR10** that extends the **BasicHTMLPERReportOR** script.

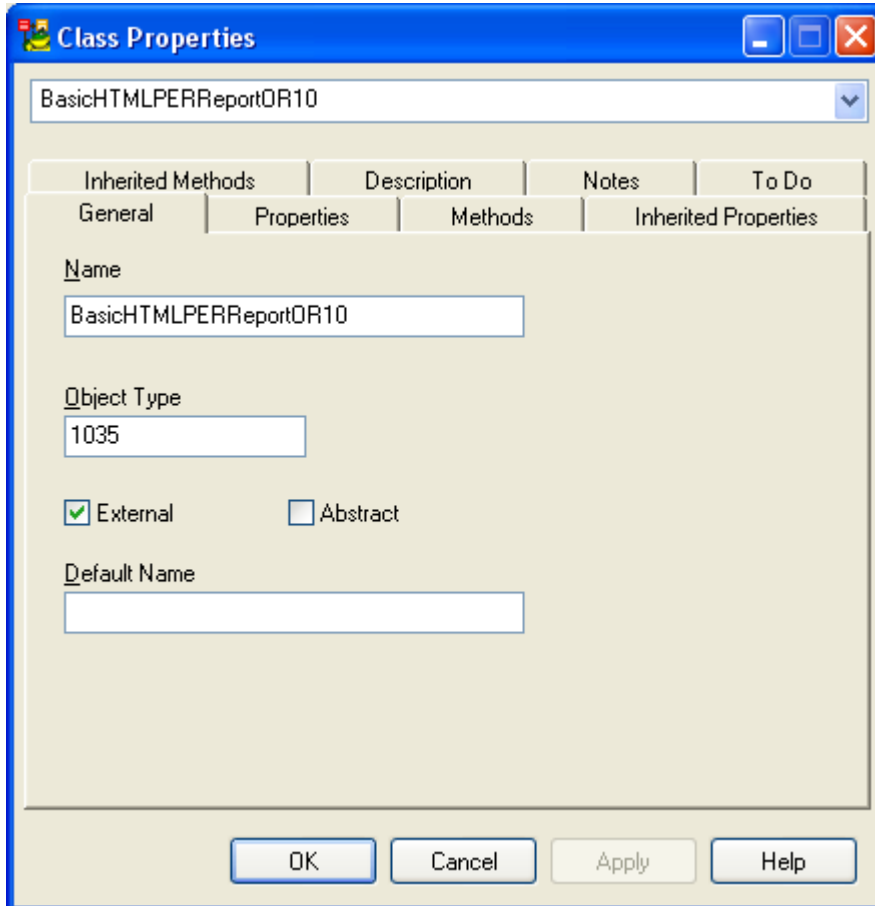


Now we now need to write a script that will extend the **ReportTableUserProperties** function defined in the **BasicHTMLPERReportOR** script.

For that purpose, we need to open our **CustomerFeedback** metamodel and the method there.

Open the **CustomerFeedback** metamodel, add there a new class (see the **Class** icon in the toolbar), edit the class and set the name to **BasicHTMLPERReportOR10**. (One extension of that class already exists, in our metamodel we will create another extension of the class).

Define **Object Type** (the value can be currently found out in metamodel to **HTML report for Oracle 10g** package).

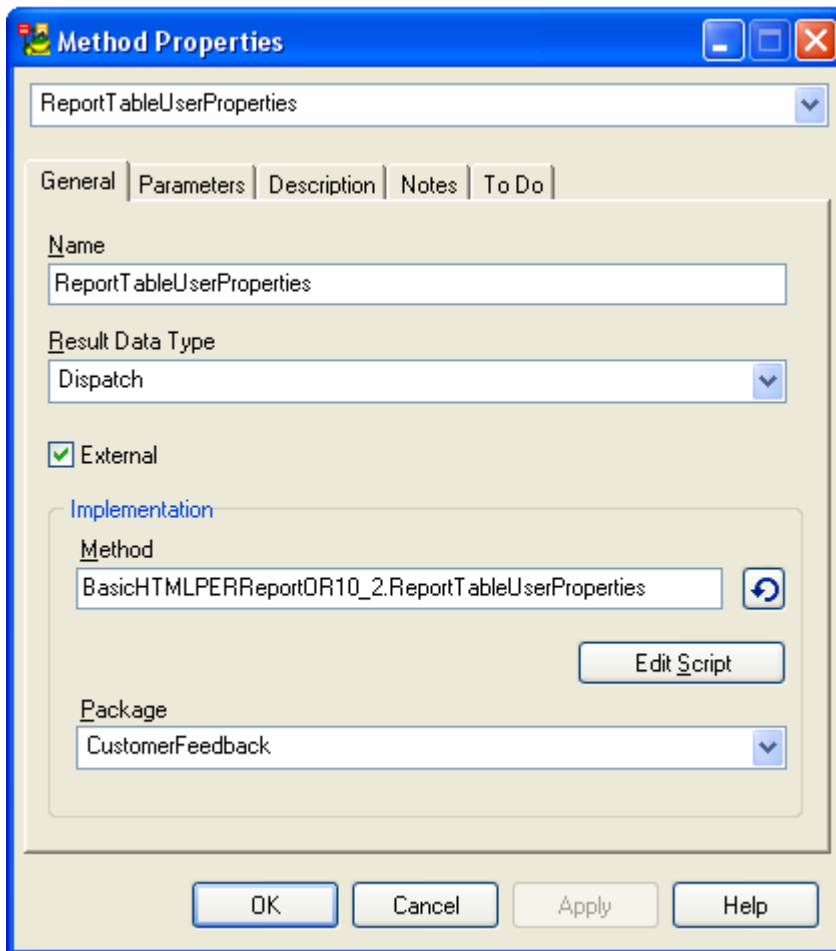


The screenshot shows the 'Class Properties' dialog box for the class 'BasicHTMLPERReportOR10'. The dialog has a title bar with standard Windows window controls. Below the title bar is a dropdown menu showing 'BasicHTMLPERReportOR10'. The main area is divided into four tabs: 'General', 'Properties', 'Methods', and 'Inherited Properties'. The 'General' tab is selected. It contains the following fields and options:

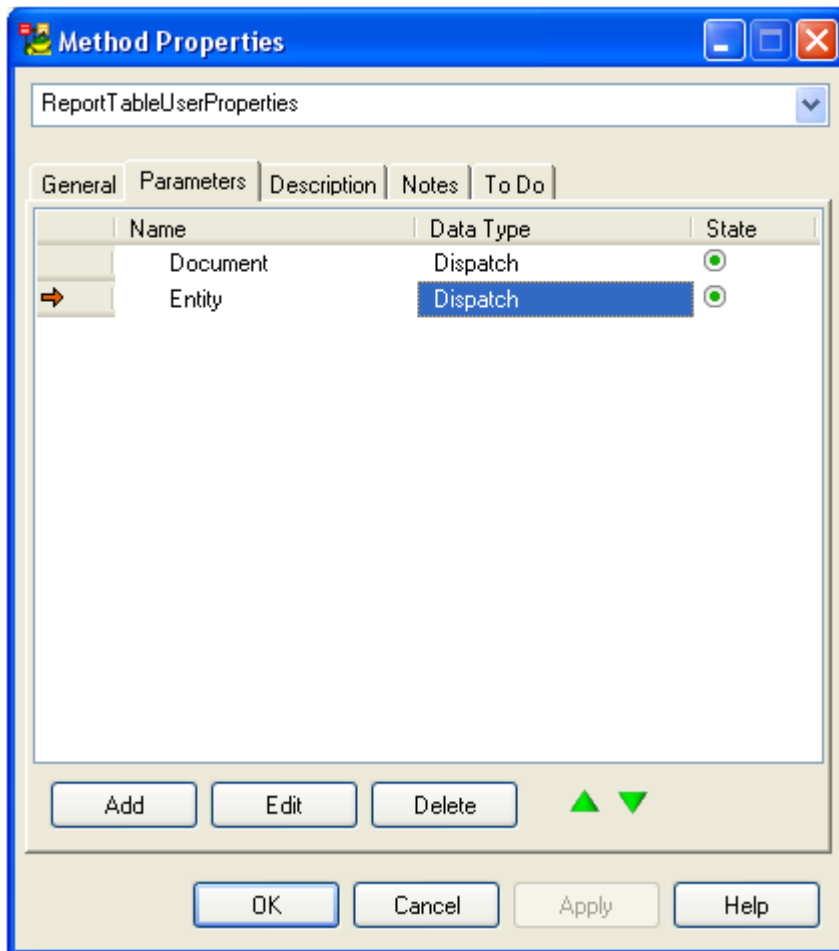
- Name:** A text box containing 'BasicHTMLPERReportOR10'.
- Object Type:** A text box containing '1035'.
- External:** A checked checkbox.
- Abstract:** An unchecked checkbox.
- Default Name:** An empty text box.

At the bottom of the dialog are four buttons: 'OK', 'Cancel', 'Apply', and 'Help'.

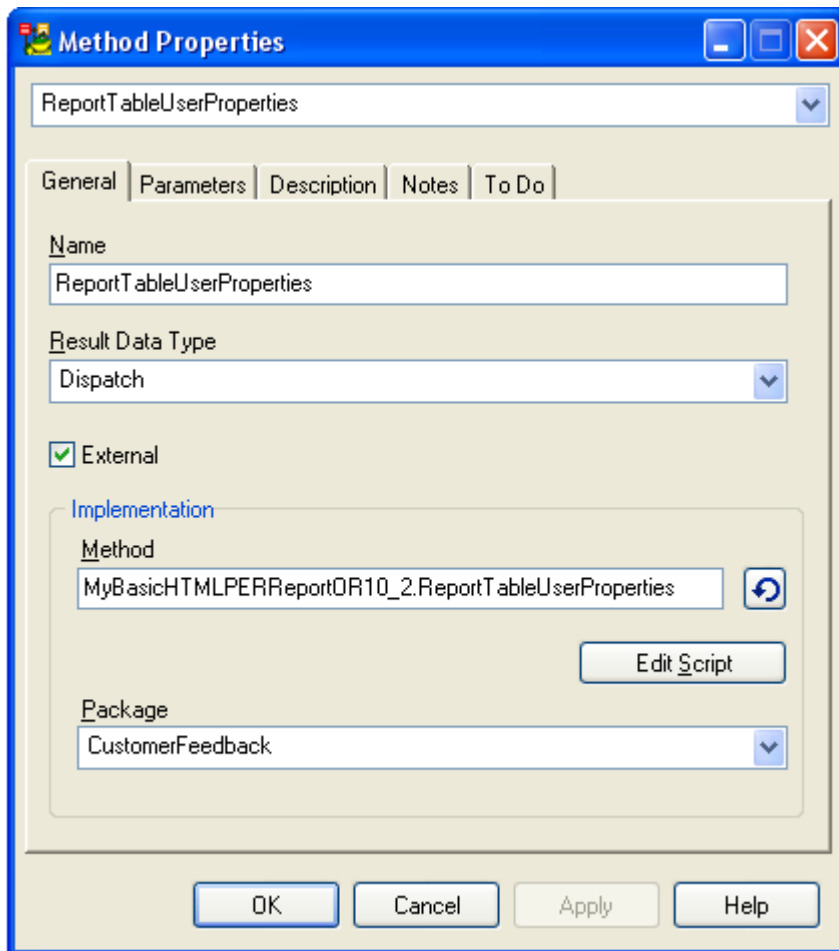
Add the **ReportTableUserProperties** method to the class.



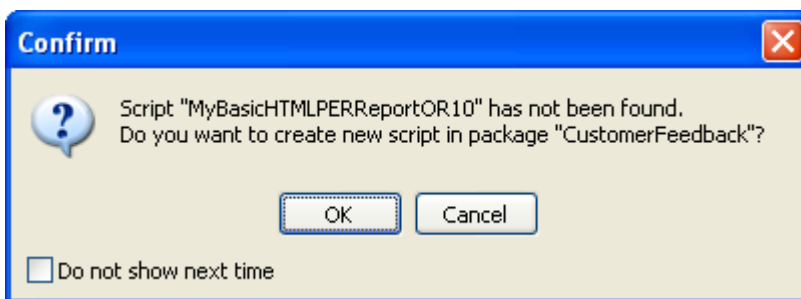
Add two new parameters to the method. (The method name and number of parameters must be identical to the original method - see the first screenshot).



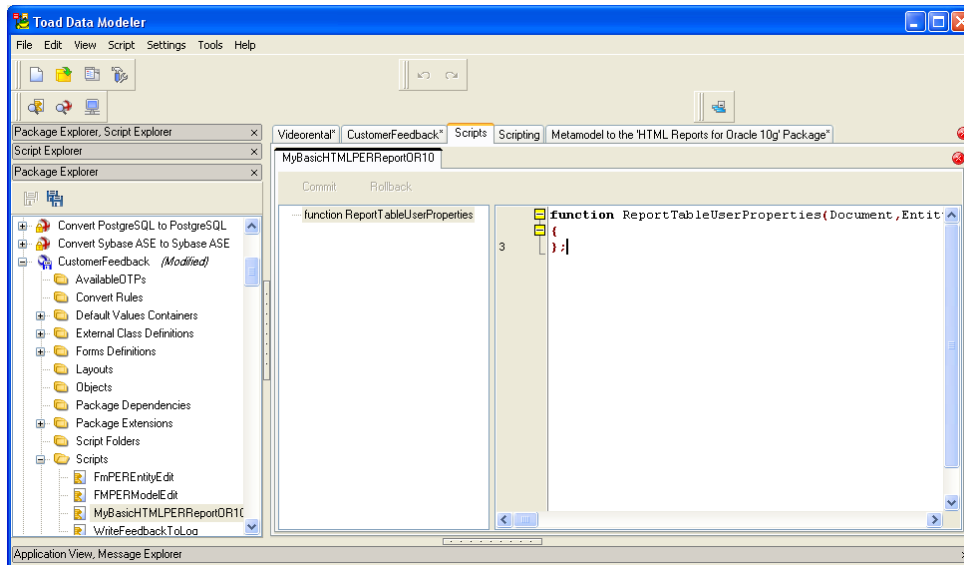
Return back to the **General** tab. Click **Reload**. Script name and method name will appear there. Add a prefix **My** to it (this will be changed in future, no manual modification will be required).



Click **Edit Script**. If the script doesn't exist, the following message will appear.



Click **OK** and define code for the **ReportTableUserProperties** method that extends the existing method of the same name.



## Code:

```
function ReportTableUserProperties(Document, Entity)
{

    // Table definition
    Table = Document.CreateTable(false,true);

    Table.CreateColumn(20);
    Table.CreateColumn(80);

    var row = -1;

    if (Entity.ConfirmedByCustomer == true)
    {
        Table.CreateCell( ++row,0, 'Confirmed' );
        Table.CreateCell( row,1, 'Yes' );
    }

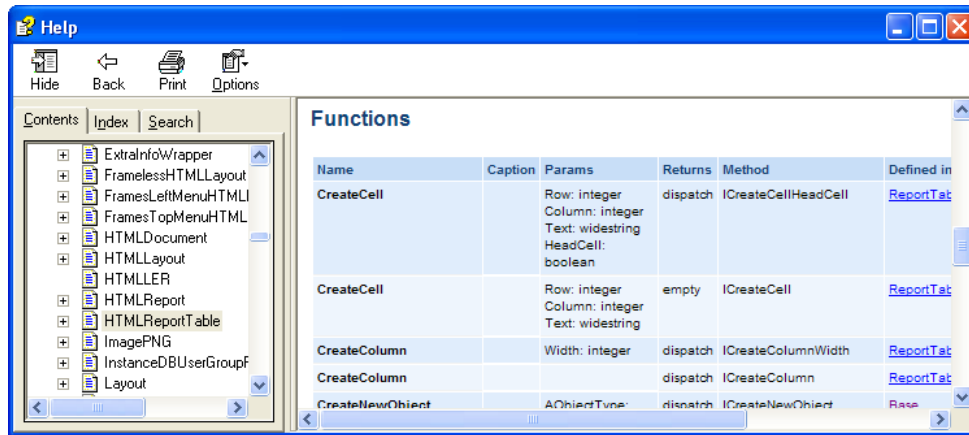
    else if (Entity.ConfirmedByCustomer == false)
    {
        Table.CreateCell( ++row,0, 'Confirmed' );
        Table.CreateCell( row,1, 'No' );
    }
}
```

```
    }  
    else  
    {  
        Table.CreateCell( ++row,0,'Confirmed' );  
        Table.CreateCell( row,1,'Undefined' );  
    }  
  
    if (Entity.NotesFromCustomer.length > 0)  
    {  
        Table.CreateCell( ++row,0,'Notes from Customer' );  
        Table.CreateCell( row,1, Entity.NotesFromCustomer );  
    }  
  
    if (row > -1)  
    {  
        Document.WriteStyled( 'CAPTION2', 'Customer Feedback' );  
        Table.Draw();  
        Table.Close();  
    }  
  
    Instance.ReportTableUserProperties(Document, Entity);  
  
};
```

## Explanation

**Table.CreateColumn(20)** - the CreateColumn function belongs to the HTMLReportTable class. All functions related to the Table object can be found in the Toad Data Modeler Reference.





**Entity.ConfirmedByCustomer** - represents the variable we added earlier to the **CustomerFeedback** metamodel.

**Entity.NotesFromCustomer.length** - standard JavaScript function that returns number of characters of the NotesFromCustomer string.

**Document.WriteStyled** - represents function that belongs to the HTMLDocument class.



**Instance.ReportTableUserProperties (Document, Entity) ;**

- **Instance** - using the Instance keyword, we can call existing function we extended. We could copy and paste the content of the **ReportTableUserProperties** function defined in the **BasicHTMLPERReportOR** script. However, if a change was made to the script later, we would have to update our script too, which would be difficult to maintain. That's why it's better to write code that will extend the existing functionality only, and call the rest from existing script via the **Instance** keyword.
- **ReportTableUserProperties** - represents existing function we call.


When you generate HTML reports now, you will see the following output. New section **Customer Feedback** is generated on top, followed by the **Table Properties** part, as originally defined in the ReportTableUserProperties function in script BasicHTMLPERReportOR.

The screenshot shows the 'Tables' tab in Toad Data Modeler. On the left, a list of tables includes SCOTT.T\_BORROWING, SCOTT.T\_CUSTOMER (highlighted in red), SCOTT.T\_CUSTOMER\_RATING, SCOTT.T\_EXEMPLAR, SCOTT.T\_FILM, and SCOTT.T\_ORDER\_RECORD. The main area displays the properties for SCOTT.T\_CUSTOMER. It includes a 'Customer Feedback' section with 'Confirmed' set to 'Yes' and a note: 'For this table the customer requires form in his application.' Below this is the 'Table properties' section with 'Schema' set to 'SCOTT'. The 'Columns' section contains a table with the following data:

Key	Column name	Domain	Data type	Not null	Unique	Check	Default
PK	customer_id		Integer	YES	NO	NO	
	name		Varchar2(20)	NO	NO	NO	

## Printing Models

### Page Format Setup

1. Click  on the toolbar.
2. Select page size, margins, orientation etc.

If you want to define the size on your own, select **Custom** from the **Page** box. Define Height and Width on the right. The measure unit can be set up in **Settings | Options | General | Select Unit of Length** - inches or millimeters.

Under **Print Options**, the two options are available:

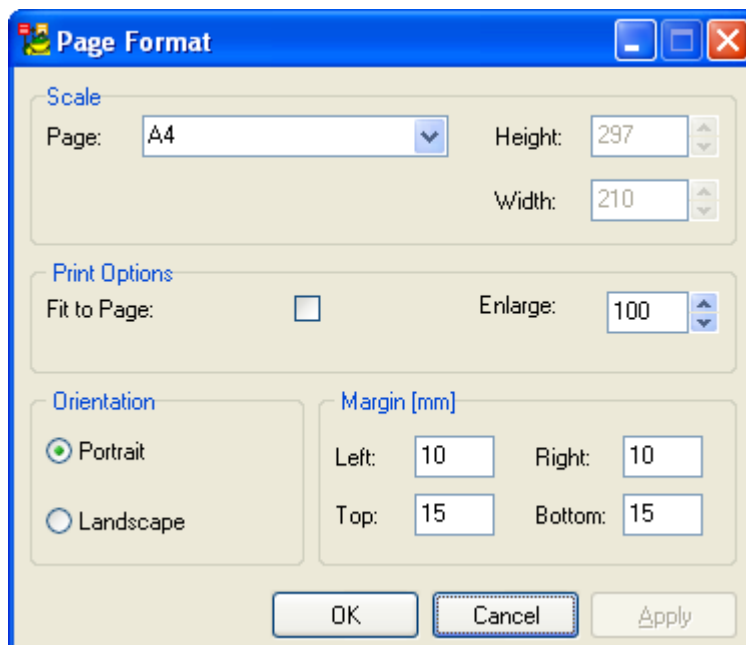
**Fit to Page** - If you select this option, the entire model will be printed on one page. This option is model-dependent.

**Enlarge** - Select a scale of your model for print.


The options in the **Orientation** area defined in the **Page Format** dialog will be automatically set up in the **Printer Properties** dialog.

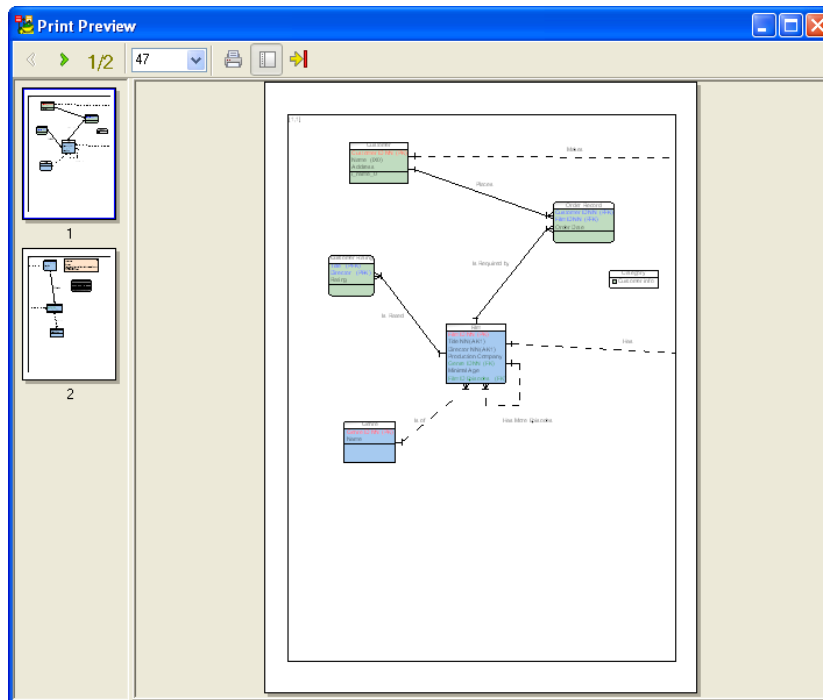
Toad Data Modeler allows you to set up a different orientation for every model, the option is model-dependent.

3. Confirm **OK**.



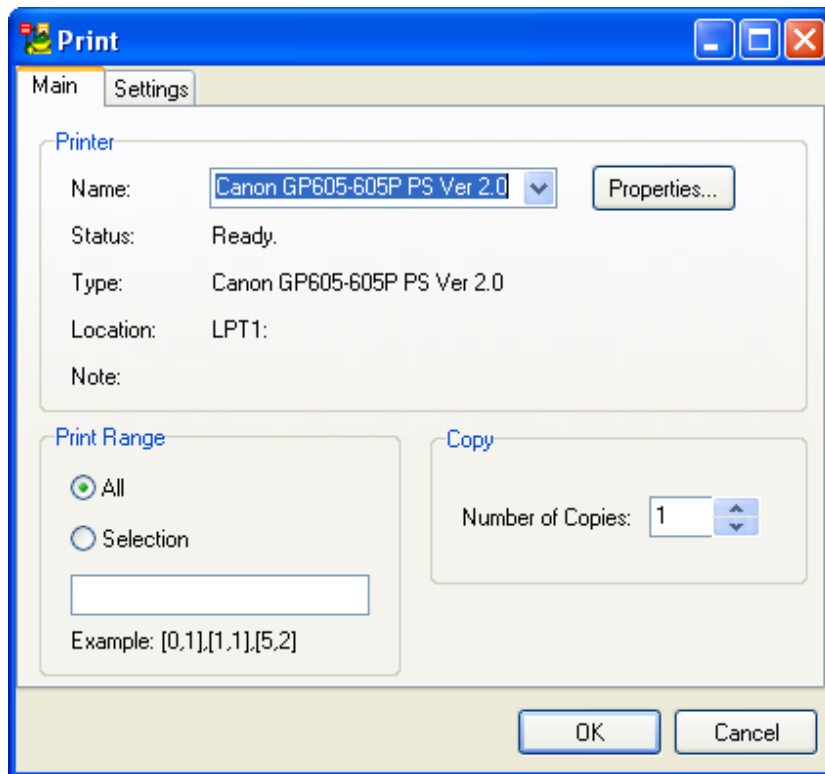
## Preview and Print

1. Click  on the toolbar to see your model preview.



2. Select appropriate printer and click **Properties...** for other configuration.

**Note:** Print on plotter - We recommend to print your ER diagram to PDF format first.

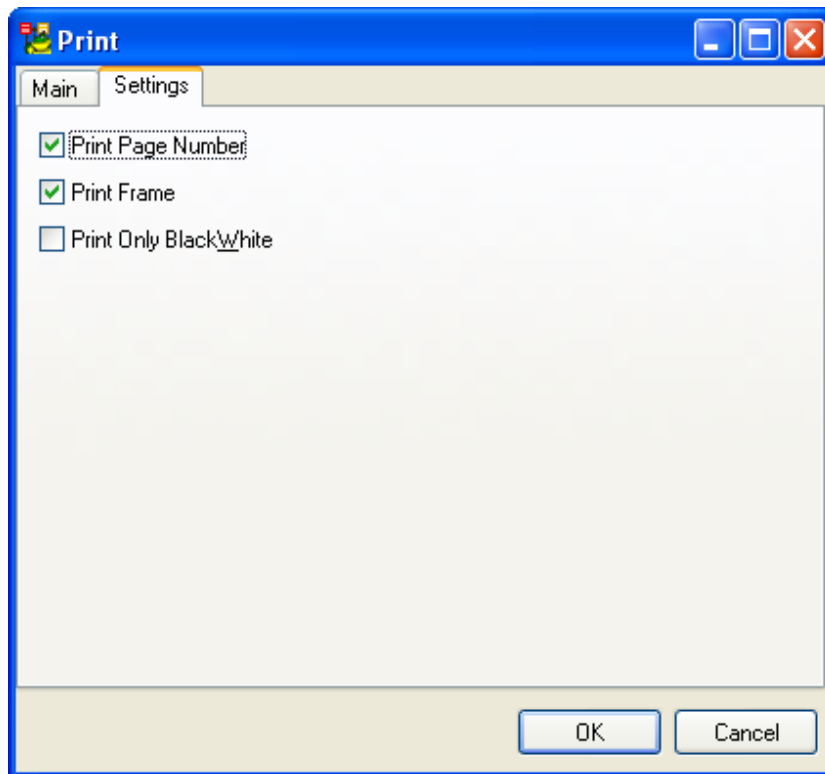


On the **Printer Properties** dialog | **Main** tab, you can see that the page orientation is set up according to the option selected in the **Page Format** dialog.

**Tips:**

- If you want to print selection of pages, follow the example in the **Print** dialog. To find out numbers of pages you want to print, go to the **Preview** dialog.
- If you want to print the entire model on one page, go to the **Page Format** dialog and select **Fit to Page**.

3. Select other options on tab **Settings**.



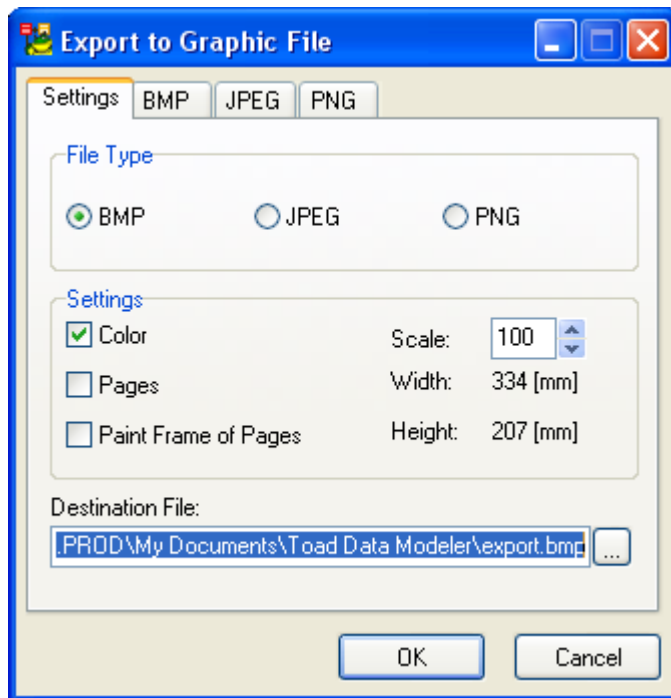
**Tips:**

- You may print to PDF using "PDF printer".
- In **Settings | Options | General**, clear the **Print Gradients** checkbox for much faster print performance. (It is disabled by default.)

## Export to Graphic File

You can export your model into graphical format.

1. Select **File | Export to Graphic File**.
2. On tab **Settings**, select some of the supported formats - BMP, JPEG or PNG.
3. Define the settings here and also on other tabs according to the selected format.
4. On the **Settings** tab, in the **Destination File** box, define a path to a file where the output will be saved. By default, the output is stored in the Documents and Settings directory, e.g. C:\Documents and Settings\user\My Documents\Toad Data Modeler\export.bmp.



**Note:** Large images cannot be exported into JPG. Generate PNG files instead of JPG if you expect the image size to be more than 2 MB.

# Appendix: Contact Quest

## Contact Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a commercial version and have a valid maintenance contract. Quest Support provides around the clock coverage with SupportLink, our web self-service. Visit SupportLink at:

<http://support.quest.com>

From SupportLink, you can do the following:

- Quickly find thousands of solutions (Knowledgebase articles/documents).
- Download patches and upgrades.
- Seek help from a Support engineer.
- Log and update your case, and check its status.

View the *Global Support Guide* for a detailed explanation of support programs, online services, contact information, and policy and procedures. The guide is available at:

[http://support.quest.com/pdfs/Global Support Guide.pdf](http://support.quest.com/pdfs/Global%20Support%20Guide.pdf)

## Contact Quest Software

Email      [info@quest.com](mailto:info@quest.com)

Quest Software, Inc.

World Headquarters

Mail        5 Polaris Way

Aliso Viejo, CA 92656

USA

Web site   [www.quest.com](http://www.quest.com)

See our web site for regional and international office information.

## About Quest Software

Quest Software, Inc., a leading enterprise systems management vendor, delivers innovative products that help organizations get more performance and productivity from their applications,



databases, Windows infrastructure and virtual environments. Through a deep expertise in IT operations and a continued focus on what works best, Quest helps more than 100,000 customers worldwide meet higher expectations for enterprise IT. Quest Software helps organizations deliver, manage and control complex database environments through award-winning products for Oracle, SQL Server, IBM DB2, Sybase and MySQL. Quest Software can be found in offices around the globe and at [www.quest.com](http://www.quest.com).

# Index

## A

alias	170, 177
align objects	228
alter reports	144
alter scripts	192
alternate keys	84
aplication view	22
application close dialog	238
application window	18
apply naming convention rules	127
attributes	52
create in LER Model	205
create in PER Model	52
autolayout	231
automatic synchronization	127

## C

categories	233
check constraints	89
check in	261
check out	261
command line parameters	31
compare	
alter reports	144
compare LER models	216
compare PER models	186
component inspector	292

component palette	291
compound keys	70
connection during RE	172
connect to database	173
conversion	196, 199
LER - PER	219
LER - PER information	218
PER - LER	199
PER - PER	196
create ERD	48
create script	311

## D

database	14
select target database for reverse engineering	172
supported databases	14
ddl script generation	155, 157
referential integrity	158
select entities/workspaces	159
default data type	54
default values	295
defaults	98
designer	18
dialog boxes	271
dictionary	75
dictionary types	81
domains	82
export/import	76
user data types	76

dictionary types	81	workspaces and objects	225
display level	226	functions	98
display modes	222	<b>G</b>	
docking	36	generate sql/ddl script	155
domain	82	generated objects	153
<b>E</b>		order	153
entities	50, 226	glossary	130
create - LER model	204	graphic file export	330
create - PER model	50	grid	227
display level	226	grid size	227
expert mode	273	<b>H</b>	
explorers	23	handle points	229
export csv files	130	hot keys	33
export of dictionary	75	html report	136
exporting to image	330	<b>I</b>	
<b>F</b>		image export	330
file		import csv files	130
saving	238	import ERD from toad for oracle	250
filter		import model from script	182
ddl script generation	159	import of dictionary	75
find function	44	import of tdm v2 models	201
find object in erd	44	import projects from toad for oracle	248
find object/item in tree	46	indexes	86
fit to page	327	inheritance	
foreign keys	70	conversion	218
foreign keys mapping	71	create	210
formatting		inheritance properties	211
relationships	229		

inplace editor	41	model merge	189
<b>J</b>		model update	186
jpeg	330	model verification	133
<b>L</b>		modeless dialogs/forms	38
layout description	15	modify form/dialog	287
linking method	69	move, zoom	231
live reverse engineering	180	<b>N</b>	
loading model from script	182	naming conventions	
logical model	202	apply in existing model	119
create	202	basic operations	108
LER to PER model conversion	219	change logical names	125
LER to PER model conversion info	218	create for new models	111
model compare	216	default synchronization	109
model merge	217	glossary	130
model verification	213	import/export csv files	130
logical view	222	introduction	106
loupe	231	use cases	127
<b>M</b>		verify	122
m:n relationship	61	navigation in trees	46
main menu	16	new project - create	256
materialized views	96	notation	224
merge	189	note	235
message explorer	243	note line	235
model		<b>O</b>	
renaming	222	object navigator box	42
model compare	186	object viewer	43
model explorer	240	objects	44
		rename	222

search	44	relationship names	
options dialog	271	move, hide, find	229
order of generated objects	153	relationship properties LER	208
orientation	327	relationship types	61
overview	231	relationships	229
		remove object	24
<b>P</b>		rename	222
package	277	report - xsl transformation	146
tutorial - create package	277	reports	
page format setup	327	alter	144
parent attributes	73	reverse engineering	178
partiality	66	from script	182
pdf	136	live	180
pdf report	136	reverse engineering wizard	169
PER - LER model conversion	199	right angled mode	229
PER - PER model conversion	196	right angled relationship line	229
permissions	105	rolenames	73
physical view	222	rtf report	136
png	330	<b>S</b>	
preview before print	328	saving model	238
preview of sql script	160	schemas	98
primary key	54	script	311
print	328	tutorial - create script	311
procedures	98	script generation	155, 157
projects in version manager	256	script preview	160
<b>R</b>		search objects/items	44, 46
referential integrity	67	select objects by owner/category	45
relationship		self-relationship	60
create in LER model	208		

shortcuts	24, 26	triggers	89
show grid	227	txi files	75
simple model merge	189	<b>U</b>	
sorting version manager items	269	undo/redo functions	245
sql preview in script generator	160	unique identifier	206
sql script generation	155	user data types	76
stamp	236	user groups	100
straight mode	229	users	100
supported databases	14	<b>V</b>	
sync & convert wizard	254	verification of names	127
alter scripts	192	verify LER model	213
compare,alter reports	144	verify PER model	133
conversion LER-PER	219	version manager	256, 269
conversion PER-PER	196	add files to project	256
model merge	189	exclude version	268
model update	186	remove version	268
synchronization		sort items	269
models and database	254	versions/revisions	261
not null and mandatory parent	65	view objects	43
synchronization caption and name	109	views	93
synonyms	98	<b>W</b>	
<b>T</b>		workspace	18, 46
tdm v2 model import	201	navigation on workspace	47
tips and hints	46	<b>X</b>	
to do list	243	xsd file generation	152
toad for oracle integration	247	xsl transformation	146
import toad for oracle erd	250	xslt templates	148
open toad for oracle projects	248		

**Z**

zoom 231