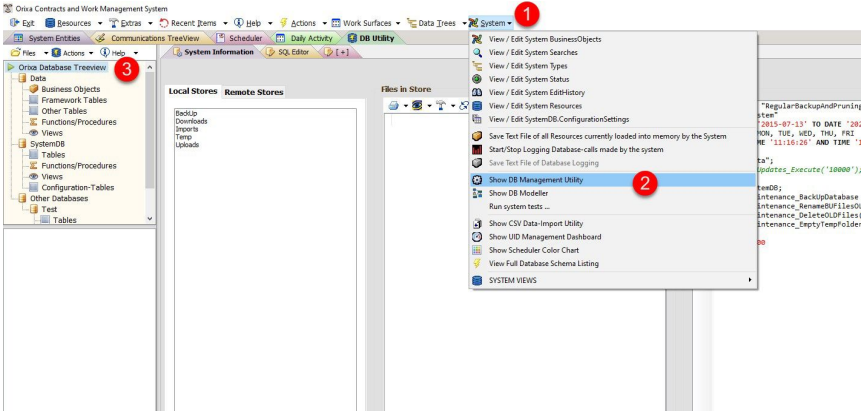


# Database Management Utility

The DB Management Utility is a program which can run independent of your Orixia App, or open as a screen within your Orixia App.

It gives access to a wide range of database administration tasks, such as access to procedures for backup, restoration and a full suite of other actions which are explained in this article.

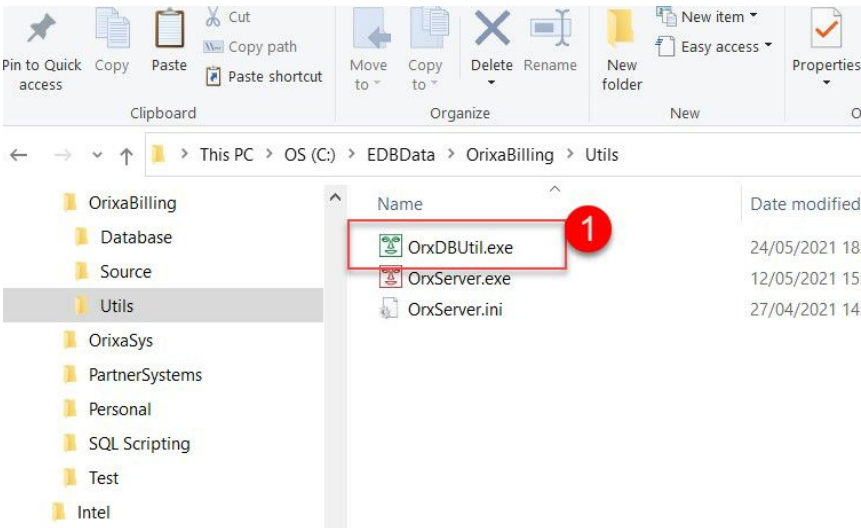
## Opening the DB Management Utility within your Orixia App



Accessing the DB Management Utility

1. Click on the "System" menu. This item will not be showing unless you have SecurityLevel settings for an Administrator.
2. Click Show DB Management Utility
3. The "DB Utility" tab will open, with the "Treeview" on the left. Click on any item in the Treeview to see different elements of the Orixia Database for your App.

## Outside Your Orixia App "Stand-alone"



DB Utility in Utils Folder

The **OrxDBUtil.exe** should have been included in your Orixia system download. It is usually stored in the "Utils" folder for your system. It is possible it has been moved elsewhere. To run it just double-click on the executable in Windows Explorer. If you want to use it regularly, add a short-cut to your desktop so you can access it at anytime.

Once the DB Utility is loaded you can do System Tasks such as Backup, Restore or Reverse Engineer your database.

Click on the items in the Orixia Database Treeview on your left to access information about all aspects of your Orixia database including access to tables, functions, procedures and views.

## If the utility is run stand-alone it will not be connected:

1. Pick an "App Name" from the drop-down-list in the top left corner of the Utility.
2. Click the "Connect" button. All database schema elements of the chosen App should load into the database-treeview for you to view.
3. If there are no "App Names" in the drop-down list, follow the steps below to create new App details.

## The DB Utility can be used to create your own Orixia App

This is a complex topic, which cannot be covered full in this document, but it is important to understand that most of the work done to create an Orixia App is done using the DB Utility. It contains numerous tools to assist with the creation of elements of an Orixia App. This document covers more day-to-day usage of the DB Utility, but the tool can be used for App Creation as well.

### Summary steps to use the DB Utility to create a new Orixia App

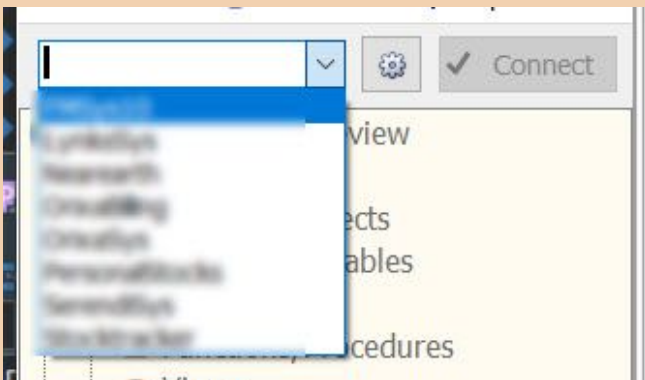
1. If you are connected, Disconnect from the current database by clicking "Actions", "Disconnect All".
2. Click on "Files", "Create New Session" to set up Registry details for your New App. Here you will need to give details such as the on-disk folder and name for your App.  
Neither the App nor the Database need to exist, but these details will be used in the creation-steps which follow.  
If you want to create a new remote database, the database server must be running on the destination IP Address and port.  
After you close the New Sessions form, SQL scripts will run to create the Database, SystemDB and a few other essential database schema elements.
3. Click on "Actions", "Run Database Changes Script" to load and run a script to create your database. You will need a Creation Script (several are available on the Orixia website, or you can write your own).  
Running this script should create database schema elements and Business Objects needed .

A full document with step-by-step instructions is available here:

[Creating a new App from scratch in the Orixia DB Utility](#)

## Differences Using in-App versus Stand-alone

The stand-alone program is not linked to any App-database, when it opens. The developer chooses a connection from those available on their computer, or can use the DB Utility to create new databases and new Apps. The main advantage of this situation is that without any database connection the Stand-alone DB Utility can make some changes to the App which are not possible within the App itself. With the App running most data-tables in the App will be open, as will be some Views, Procedures, Functions and other database schema elements. It is not possible to run SQL to ALTER any open schema element, so changes to these will be impossible unless you run the DB Utility in Stand-alone mode.



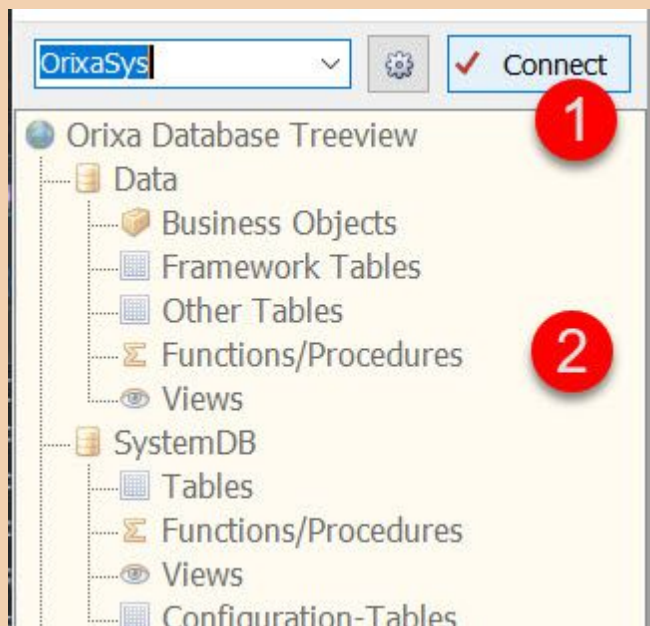
DB Utility App-Sessions List

In the stand-alone App a small connection tool-bar is added, above the "Treeview", as shown on the left.

This includes a drop-down list which will show all Orixia Apps accessible from on your computer (both local and remote), a "settings" and "Connect" button.

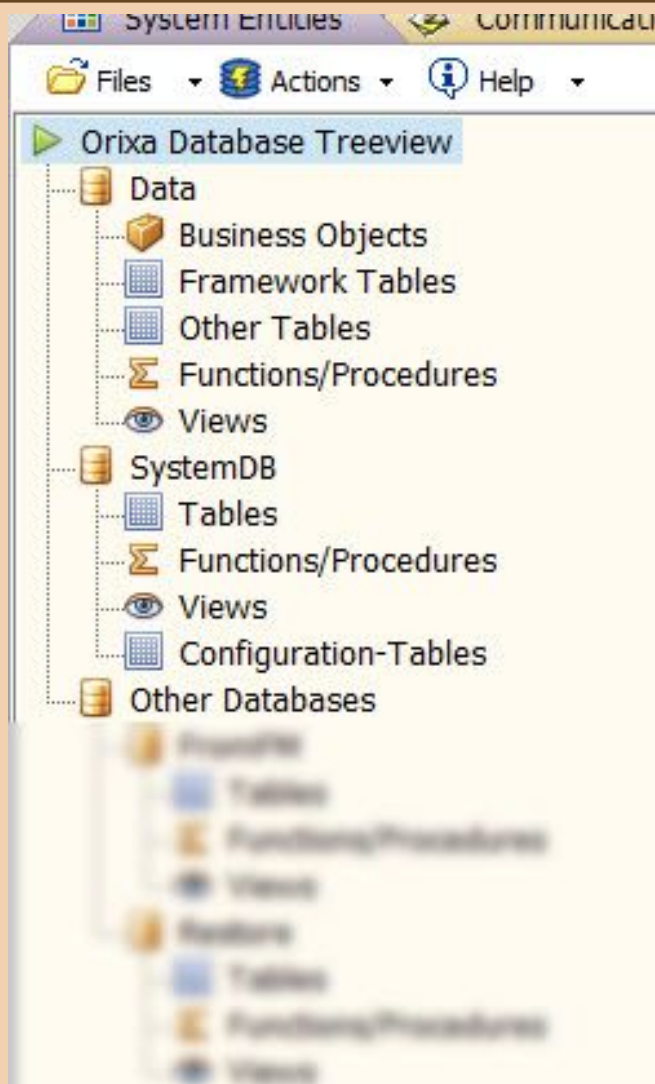
Pick one of the Apps from the drop-down list then:

1. Click the connect button to load all the database details for that App.
2. Note that until the Utility is connected the Treeview will show grayed out (disabled).



DB Utility Connect Session

## The Management Utility Treeview



DB Management Utility Tree-view

Most navigation within the DB Utility is done by selecting items in the "Database Treeview" (shown on the left).

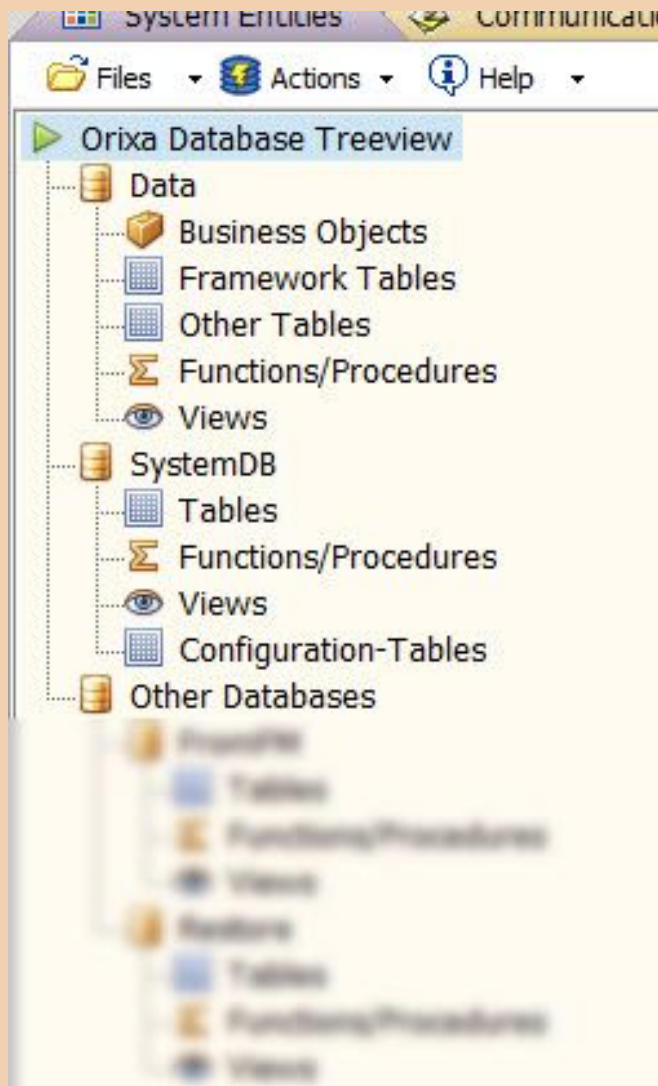
The different headings of the Treeview give access to the contents of different parts of the database(s) of your App.

All Oria Apps include a main "Data" database, and a "SystemDB" database. Many Apps may have additional "Other Databases" (as shown blurred on the left).

The main "Data" database contains all the data-tables and database schema elements that make your specific App work.

The "SystemDB" database holds a few smaller data-tables and schema elements which are used locally by the App, for example to hold users Configuration Settings.

The "Other Databases" may include cloud-databases used to hold shared data, or test and backup databases which are used for development of your App.



Tree-view Main Database

The main "Data" database tree items are separated to make your App easier to understand.

All the data-tables which are specific to your App, and contained in the "BusinessObjects" system-data-table are accessed from the "Business Objects" tree item.

All the data-tables which are used in all Orix Apps, and are part of its general framework are included in the "Framework tables" tree item.

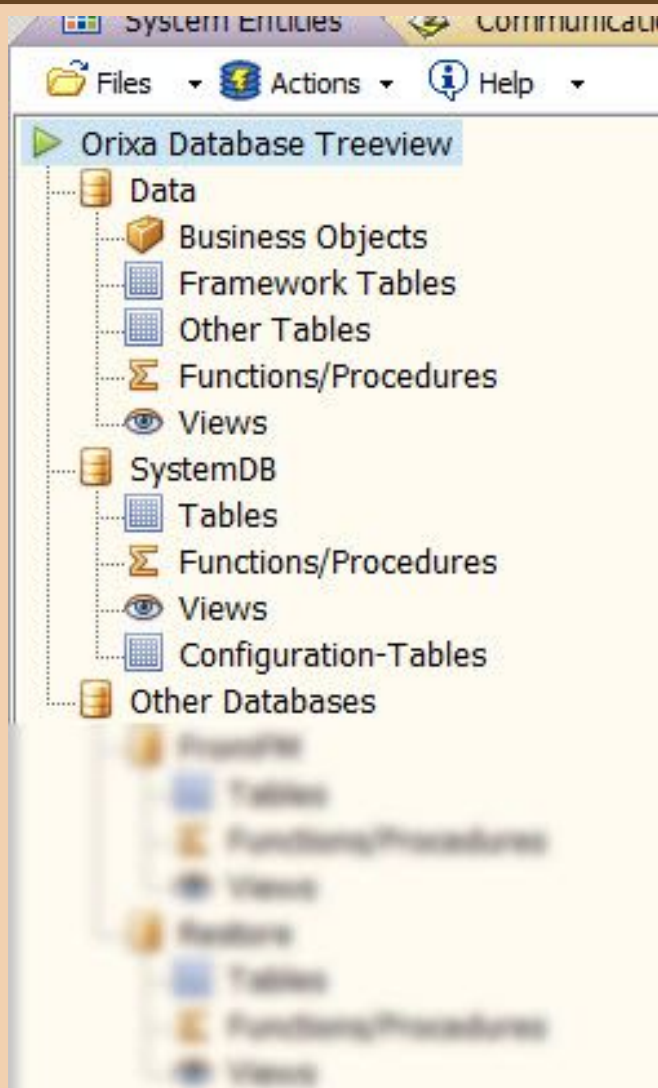
Any other tables at all are listed in the "Other Tables" tree item. This will include data-tables to hold temporary data, or computation data.

All of the main "Data" database Functions, Procedures and Views can also be accessed by clicking the appropriate Treeview Item.

The SystemDB tree items list all the contents of the Orix SystemDB database which is used to hold local settings and data for your App.

Click on the different items under this heading to see the data-tables and other schema elements of the database, and review and access their data.

Note that the SystemDB tree items include a heading "Configuration-tables" which holds links to meta-data tables for the operation of the database itself such as listings of current connected Sessions, Server Event Logs and other statistics.

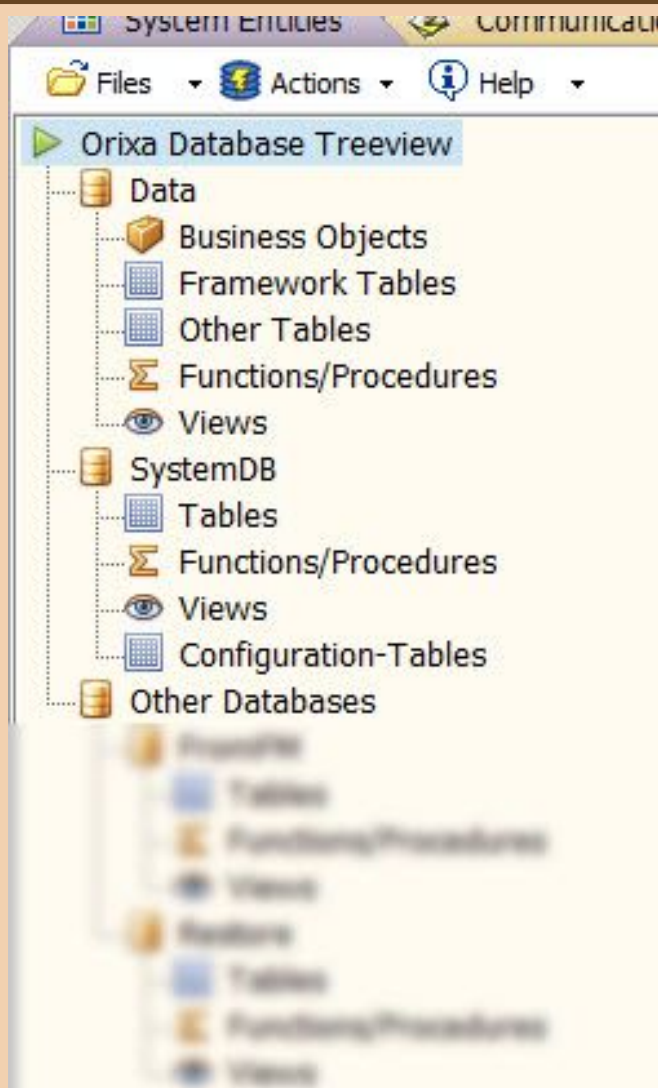


Tree-view SystemDB Database

If your App includes any other databases which have been created by your developer they will be listed under the "Other Databases" tree item.

Apps which work in a single premises may not have any entries under this item. But if your App includes links to a Cloud-Server, or additional databases for purposes such as Testing or Development these will be listed here.

Click on the items under each database's name to access data-tables and other schema elements.



Tree-view Other Databases

#### Accessing table-details

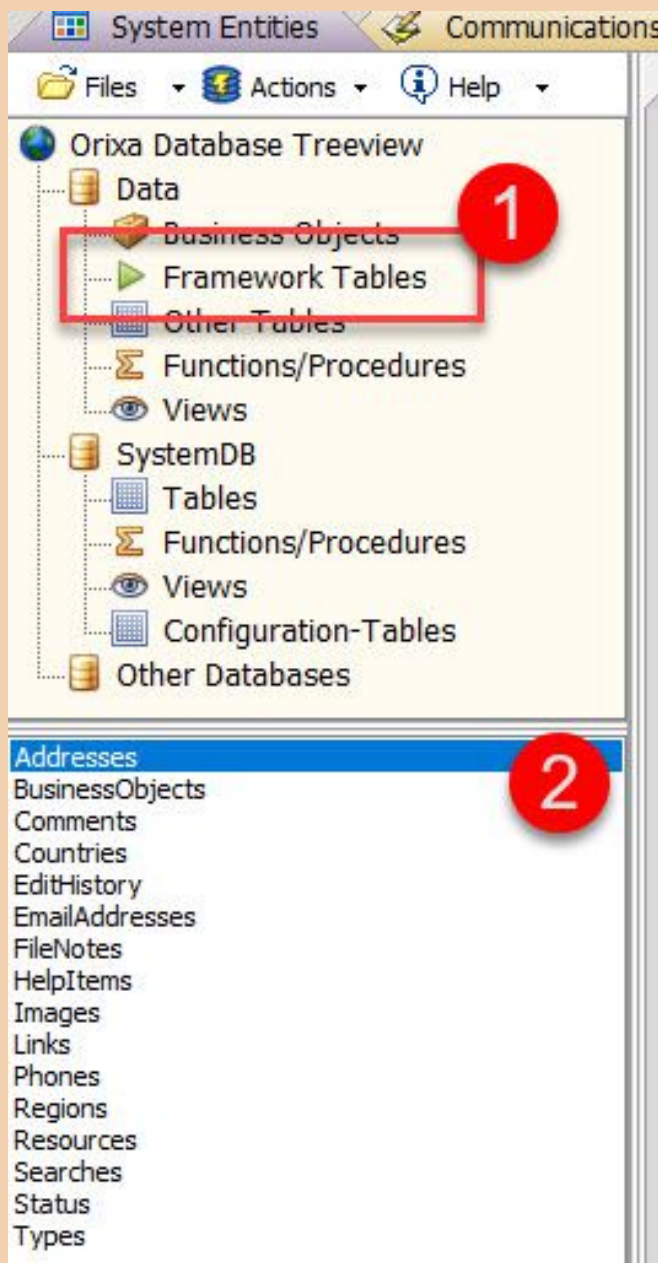
1. Select any Treeview item that displays tables.
2. The list of tables in that part of your App will be shown.

Click on any table-name in the list and detailed information about that table will be displayed.

Right click with the mouse on the selected table to see a list of options for that data-table.

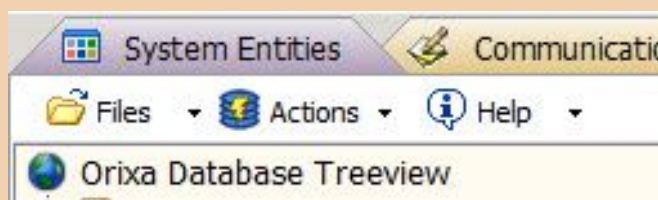
Note that when "Views" are selected in the tree view the same visual interface displays in the Database Utility.





DB Management Utility Framework Table-details

## Using the Utility: The Toolbar



DB Utility Toolbar

The top left part of the Database Utility window contains a small toolbar.

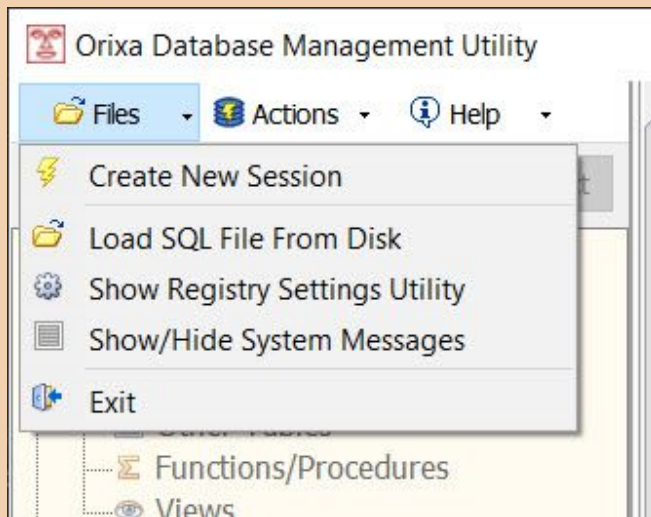
Use it to access important commands for managing and developing Orix Apps

### The "Files" Toolbutton

"Create New Session" allows creation of a wholly new database which can be used for a new App. Developers should only use this option if they want to create a completely new system.

A full document with step-by-step instructions is available here:

[Creating a new App from scratch in the Orix DB Utility](#)

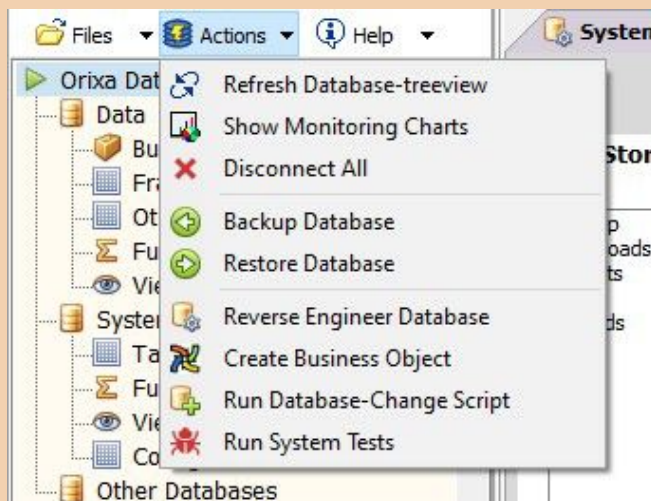


Toolbar Files-button

Details of using the "Registry Settings Utility" this are provided below.

Show/Hide System Messages toggles the visibility of the System Messages screen which is only made visible when SQL Scripts are being run or the utility returns complex messages to a user.

If the Utility is open as an independent application an "Exit" button can be seen at the bottom of the list, if the utility is opened in an App this option is not shown.



Toolbar Actions-button

### The "Actions" Toolbutton

This toolbutton gives access to a set of powerful actions. The more complex actions are explained in their own help-sections below.

"Refresh Database-treeview" simply updates the contents of the DB Utility Treeview. This is useful if you run a change script, the treeview does not reload until this command is called.

"Disconnect All" use this command to completely disconnect all currently open data-connections. This should be done prior to running any change-script.

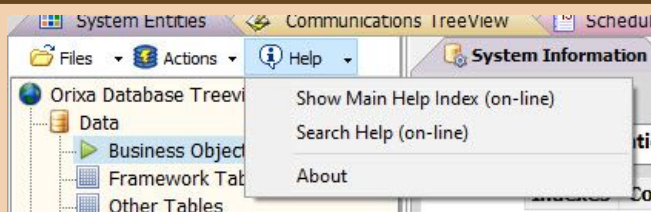
"Backup Database" click here to run the "Backup Database" procedure. This procedure creates backup files of the main database and SystemDB. Developers can rewrite this procedure to extend it to do extra work if they want to.

"Restore Database" click here to select a file from the Backup Store to use to Restore the main database. Note that changes made since the backup was done will be lost.

"Create Business Object" use of this is described here: [Create Business Object Form](#)

"Run Database-Change Script" first disconnects all existing data-connections then opens a window with a dedicated connection to your database to allow Administrators to run database change scripts.

The other commands are described in later sections.



Toolbar Help-button

### The "Help" Toolbutton

This toolbutton simply provides a direct link to the Orixa website help and documentation.

"Show Main Help Index" opens a browser window to the main Help Index.

"Search Help" opens a browser window to a page generated by help items that match the search term you enter. Note that it is best to keep search terms simple if you want to see any results.



Orixa Session Management & Registry Settings Utility

Orixa App Saved Registry Settings

App Name:  Find Values

Database Name:

Catalogue Folder:  Pick

IP Address / Port:

System Password:

a password is stored by the system

Use this screen to set the network and local computer locations for the Database system. Existing App-settings are shown in the "App-Name" list.

Type a new name and settings in to the fields and click "Save Settings" to add a new App to this computer.

Note that if a password has been stored it will NOT be shown here, you can type a new password into the "System Password" field if you need to update or change the password.

Save Settings

Close

### Registry Settings Utility

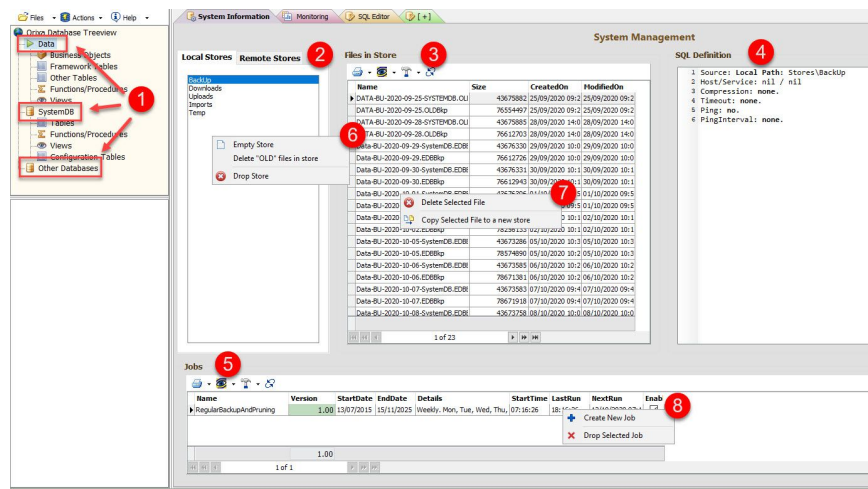
### Registry Settings Utility

Details of how to use this utility are detailed here:

[Using the Registry Settings Utility](#)

# Screens within the Database Management Utility

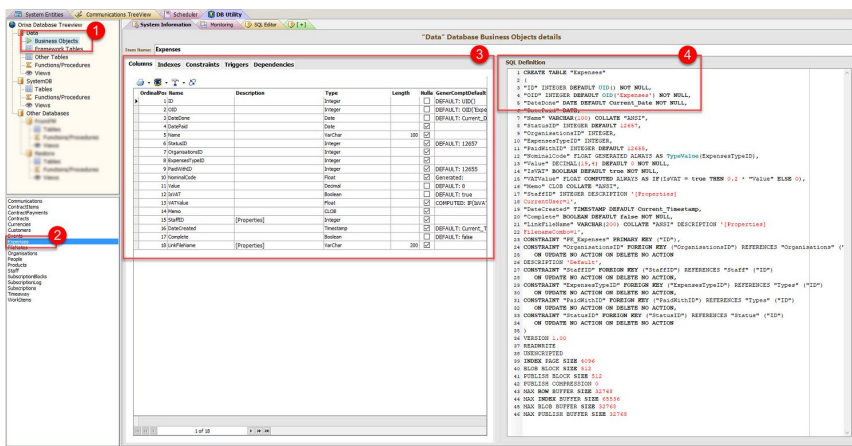
## System Management Screen



DB Management Utility System Management Screen

1. If any "Database" heading is selected, the "System Management Screen" (shown above) will display. This screen provides a visual context to review and work on system-level database elements: Stores and Jobs.
2. Local and Remote Store stores for your App are listed. Clicking on any one will show data about it in the other parts of the screen.
3. "Files in Store" will fill with the file contents of any store selected from the Local Stores or Remote Stores lists.
4. The SQL definition of the selected Store or Job is displayed, giving the Developer a view of the functionality of the selected item.
5. The Jobs Grid, lists all the database Jobs in your App. Jobs are pieces of programmed SQL which can be set to run automatically, for example to automate a database backup every night, or to automatically run a particular report etc. Once a Job is selected its SQL definition is displayed in the SQL Definition at 4.
6. Right-click on an item in the Store-list to show the command menu. Here you can choose to Empty Store, Delete "Old" files in store, and Drop the selected Store. Note that "Drop Store" is **permanent** and results in the files contained in the store being deleted from disk.
7. Right-click on an item in the Files-list to show the command menu. Here you can choose to Delete the selected file, or copy it to a different store.
8. Right-click on an item in the Jobs-list to show the command menu. Here you can choose to Creat New Job or Drop Deleted Job.

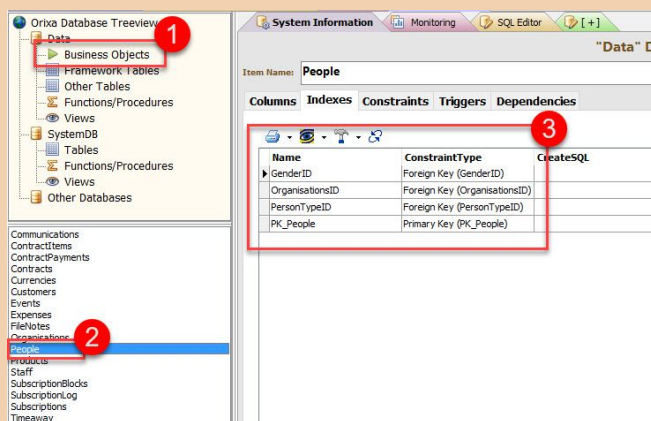
## Table Details Screen



DB Management Utility Table-details

1. A "Tables" item is selected in the Treeview. This results in the list of tables filling at 2.
2. Select any table in the tables-list and its details will display at 3., and 4.
3. A grid containing all critical meta-data for an individual data-table. The first grid shows the data-columns for the table, with name, data-type and other meta-data.
4. The full SQL Definition for the selected data-table is displayed.

## Viewing Data-table Indexes, Constraints, Dependencies and Triggers



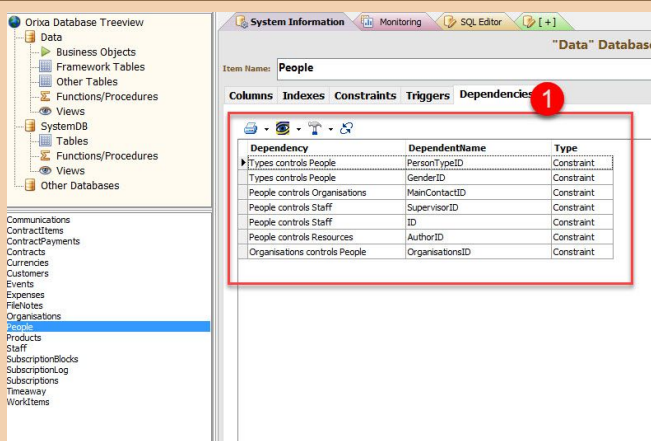
Indexes

### Data-table Indexes

1. Selected a "Tables" Treeview Item.
2. Click on any table
3. In the table-details screen click on the "Indexes" tab.

A list of the Indexes of the current table will display.

This is useful as it shows fields in the data-table that will allow for very efficient searching and querying of the data-table.



Dependencies

### Dependencies

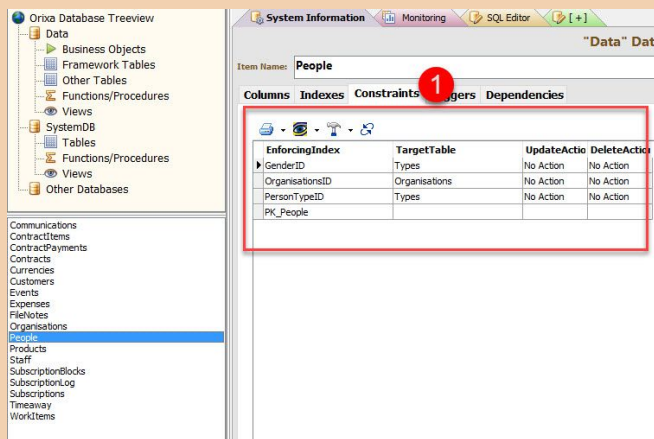
The Dependencies Grid shows how the selected data-table is called on by other elements in the database.

For example a data-table may be used by another table, or by a View, Function or Procedure.

In such cases any changes made to the selected table may impact on the dependent elements.

**Prior to any database-change, always review the dependencies of any database schema element to ensure the change does not break some other aspect of the database.**

Note that MANY database elements can have Dependencies. The Database Management Utility shows a Dependencies Grid in each case, allowing you to check dependencies.



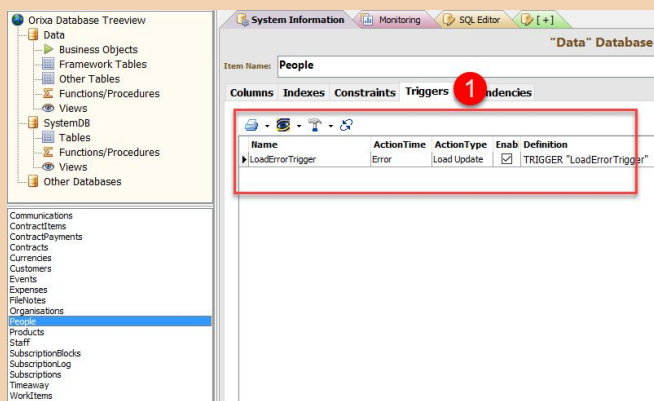
### Constraints

### Constraints

A Constraint is a limit or restriction placed on a row in a data-table.

Many App data-tables include Constraints, the most common is a "link constraint", where the value of one field is restricted to only contain data that appears in a column of another table.

Orixa uses these types of link constraint to build the inter-linkages that make Orixa Apps work.



### Triggers

### Triggers

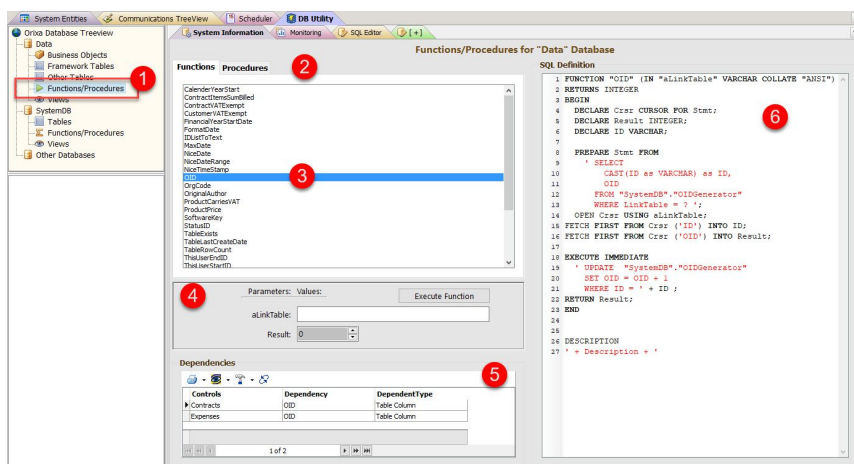
When a change is made to a row in a data-table Orixa checks whether any Triggers are linked to this change, and if they are the SQL in the trigger is run.

Triggers can be used to enforce changes or update records when a change happens.

Triggers are powerful because they are called "server side", if database data is changed for any reason a trigger will still run.

This means users editing Orixa Data in Microsoft Excel, or edits that occur as a result of automated database Jobs will also cause triggers to activate. Triggers are not limited to act only when data is edited in an Orixa App.

## Functions and Procedures Screen



### DB Management Utility Table-details

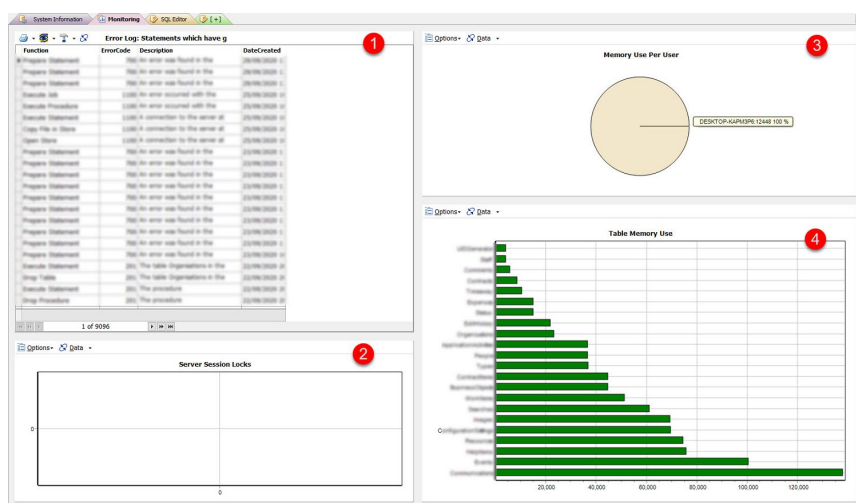
1. When the "Functions/Procedures" Treeview item is selected for any database the Function/Procedures screen displays, as shown above.
2. All Functions and Procedures for the selected database are listed.
3. Click on any one and its details will display at 4., 5., and 6.
4. All the Parameters of a Function or Procedure will be listed. Enter values into these fields and click the "Execute Function" ("Execute Procedure") button and the function / procedure will be executed and any results will be displayed. If the Procedure returns a data result-set this will be displayed as a data-grid.
5. Dependencies Grid. As detailed above for the Table-details Screen, Functions and Procedures can have dependencies. For example a Row in a data-table may be computed. In such a case the result of the

- computation may come from a Function in the database. As with other dependencies it is important to check that changing any function will not break these dependent computations.
- SQL Definition. This shows the full SQL for the Function or Procedure.

## Monitoring Screen

Click on the "Monitoring" tab of the Database Management Utility to see a simple set of charts and a grid detailing activity in the database of your App. Two of the charts show memory use, allowing you to determine the level of stress on your server from the Orixas system.

The Monitoring Screen is not opened by Default. To create the Monitoring Screen click on "Actions", "Show Monitoring Charts" as explained under "Actions ToolButton" above.



Monitoring Dashboard

- The Error Log Grid. If a command runs in your App and this results in an error, a log of the error is saved to the System LogEvents data-table. Developers can review the details of errors here. Double-click on the "Description" field to see a full listing of the error.
- Server Session Locks. This chart is empty in the image above, as the system has just started and is being used by a single user. In multi-user use, all the record-locks in place for all users will be displayed. This is useful as it allows the System Administrator to see which users are touching which parts of the database.
- Memory Use Per User. In the images above this pie-chart is showing details for a single user but in multi-user cases this chart will show the amount of memory being used by different users of the system.
- Table Memory Use. This chart shows the total memory in-use on the server broken down on a per-table basis. This allows the System Administrator to see which tables are demanding most resource from the server.

## Using SQL Editors

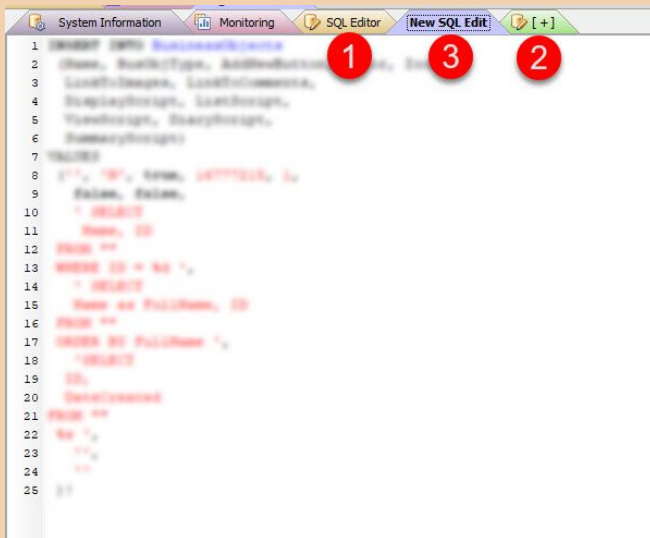
It is often useful to open a SQL editor to allow diverse queries to be run against the database. This can be done from the "SQL Editor" or by opening new SQL Edit tabs.

[Detailed information about the SQL Editor](#)

**NOTE:** Your Orixas App will include at least 2 databases, the main "Data" database and the "SystemDB", it may contain others (such as a "Test" database).

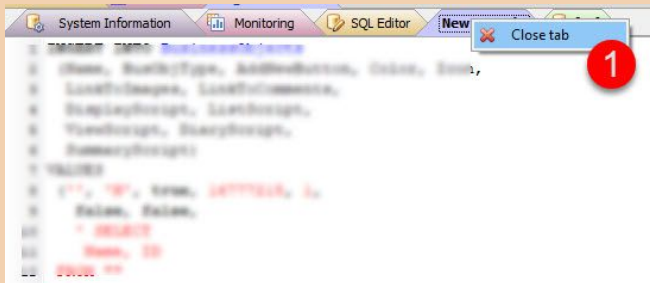
In general SQL run from the "SQL Editor" will run against the "Data" database. To run SQL against the SystemDB or other databases see the note further down in this document.





SQL Editor Windows

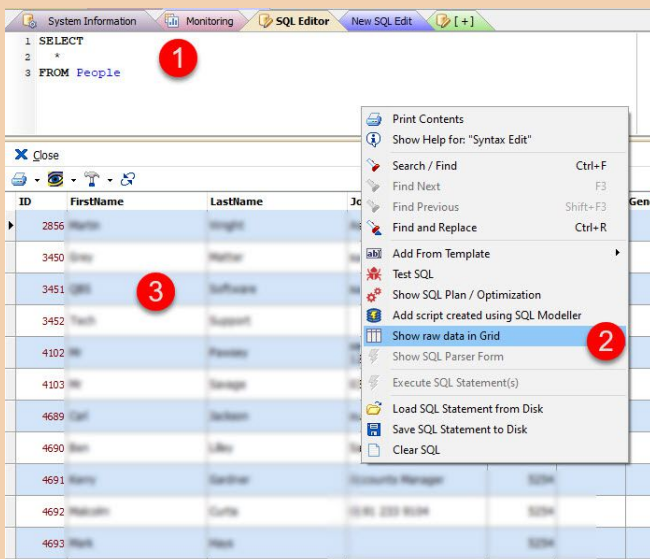
1. Click on the "SQL Editor" tab to show a view of current SQL and to run the statement(s) that have been written.
2. Click on the "Add tab" "+" sign to add new SQL Edit tabs.
3. A New SQL Edit tab, with a SQL Statement added to it.



Close SQL Tab

1. You can close any New SQL Edit tab by right-clicking on the tab-heading and clicking on "close tab".

Note that if you have unsaved SQL in the New SQL Edit tab you will be prompted for whether you would like to save it before exiting.



SQL Editor in use, with menu

1. SQL Statement
2. Right click on the SQL Editor window to access the menu, select "Show raw data in Grid"
3. Data is displayed in a grid below the statement

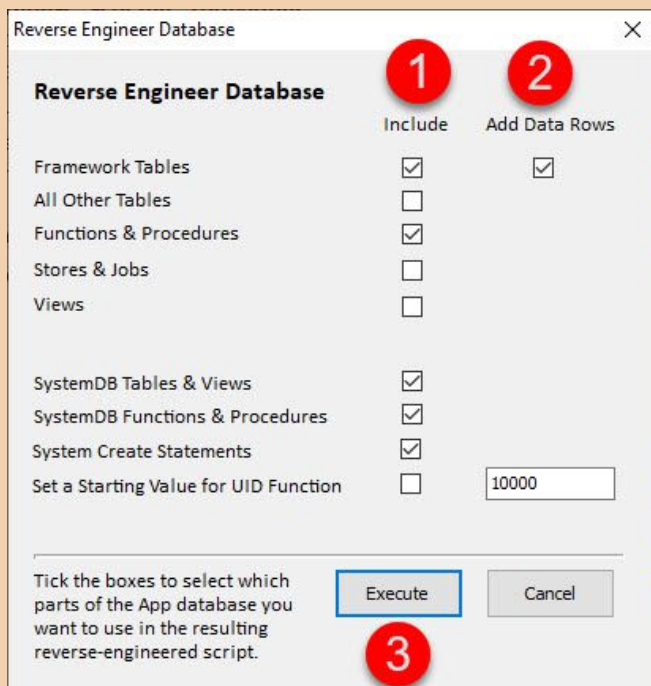
Note that it is possible to edit data in the grids returned by the SQL Editor if the statement is a simple SELECT for a single table. Complex multi-part queries do not result in "live" data.

## Other actions

### Reverse Engineer Database Window

A strong feature of Orixa is that you can use any existing Orixa system to create a new system.

Create an Orixa compliant database, and when the Orixa App runs it will build your App based on the contents of the database.

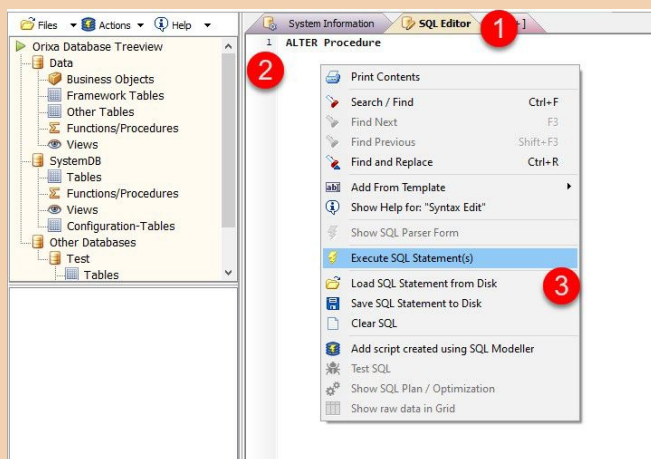


Reverse Engineer database

The Reverse Engineer Database Window allows Developers to create the extremely long and complicated SQL statements which are required to do this.

Developers can also use this window to extract useful sets of SQL which they can add to and extend to create new systems.

1. Items ticked in the "include" column will have SQL scripts generated to re-create their database schema elements and meta data.
2. The row data for the Framework tables can also be added.
3. Click Execute to generate the script, it will be added to the SQL Editor.

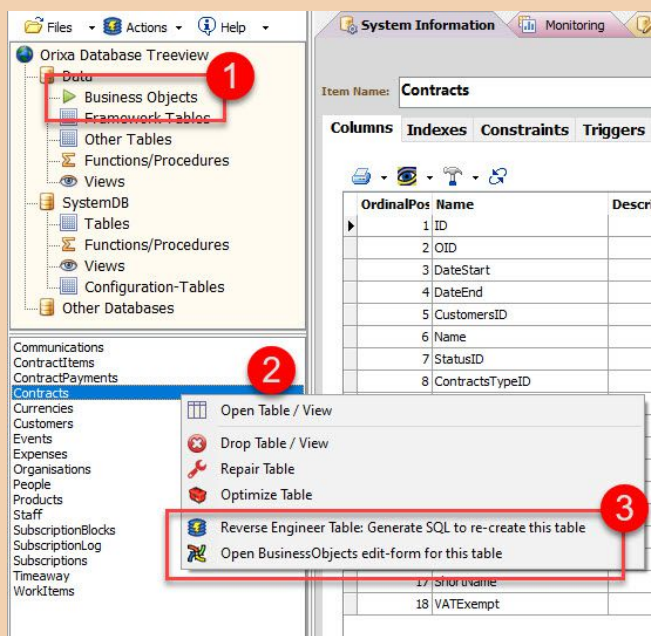


DB Utility SQL Editor

The DB Utility includes a SQL Editor, in which the developer can write and execute SQL statements, including statements to change the structure of their database.

Great care must be taken running such scripts and the effects of changes must be thought through carefully. It is usually best to run change-scripts against a **test database** first, and then against the main Database.

1. SQL Editor Window.
2. Main writing area to add SQL.
3. Right click on the SQL editor surface to open a menu of actions including "Execute SQL Statements"



### View Table Actions

The Tables / Views list includes a useful menu.

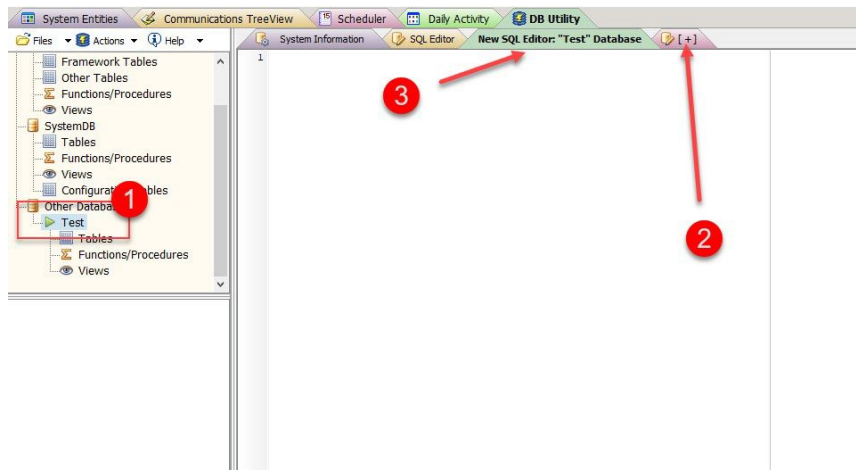
1. Select any table or views Treeview item.
2. Right click on an item from the list. The Menu will appear.
3. This menu includes a number of options, including "Reverse Engineer the selected table" and "Open BusinessObjects Edit Form."

Options in this menu include "Drop Table / View", "Repair Table" "Optimize Table". These are useful actions which you can start just by clicking on the menu-item.

The **Reverse Engineer table** option is useful as it creates SQL INSERT statements which duplicate the contents of any data-table. This can be useful for copying data from one system into another.

The **Open Business Objects** option is useful for direct access to the scripts that create the business object.

## Running SQL against the "SystemDB" or Other Databases



### Database-specific SQL Editor

If you need to run SQL against either the SystemDB or other databases

1. Click on the Tree-view heading that corresponds to your selected database.
2. Click on the "[+]" to add a new SQL Editor.
3. A new SQL Editor will appear, note that the name of the database it is targetted is included in the title of the tab it is linked to.

## Features of the SQL Editor

Writing SQL in the Database Management Utility is made as easy as possible with a fully-featured, modern SQL Editor, that has many usability enhancements.

To see an explanation of the features of the editor check this article: [The SQL Syntax Editor](#)