

ICONICS, Inc.
100 Foxborough Blvd.
Foxborough, MA 02035
Phone: (508) 543-8600
Fax: (508) 543-1503
[Worldwide Support Centers](#)
www.iconics.com



© ICONICS, Inc. All rights reserved.
Specifications are subject to change without notice.
GENESIS, GENESIS32, Pocket GENESIS, BizViz
and their respective modules, OPC-To-The-Core, and
Visualize Your Enterprise are trademarks of ICONICS,
Inc. Other product and company names mentioned
herein may be trademarks of their respective owners.

Introduction

The Data Mining ActiveX is included in the GENESIS32 installation, and it can be inserted into any container that has the capability to embed ActiveX objects, including GraphWorX32, Visual Basic forms, Microsoft Word, and Microsoft Excel. The basic purpose of the Data Mining ActiveX is to read data from a specified data source (e.g. Microsoft Access, Microsoft SQL Server, and ODBC databases) and display the chosen data set in a data grid during runtime mode.

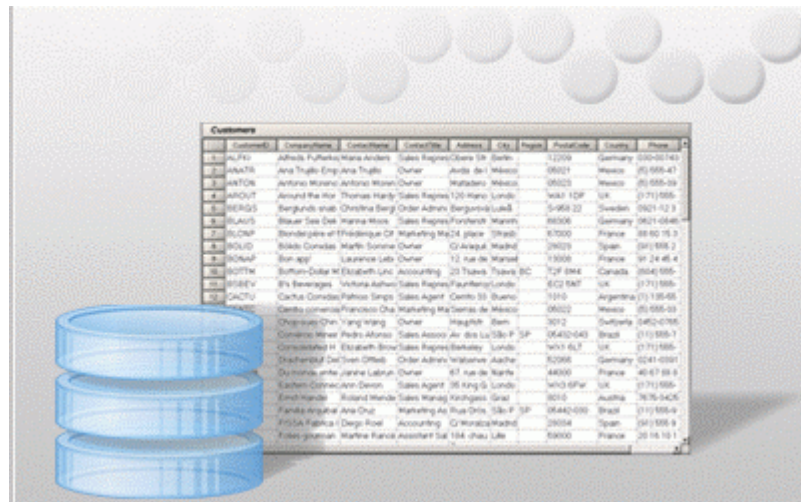
Data Mining ActiveX Features

Key features of the Data Mining ActiveX include:

- Data Mining Grid for displaying data during runtime
- Support for open database connections, including Microsoft Access, Microsoft SQL Server, Microsoft Excel, Oracle, and ODBC databases.
- TraceWorX32 support
- Global aliasing support
- Column aliasing for data items
- Smart aliasing and column comparison
- Column sorting in runtime mode
- Column hiding
- Auto/manual grid refresh/updates
- View the Data Mining ActiveX via WebHMI

Inserting the ActiveX Into a Display

To insert the Data Mining ActiveX, choose **Insert Object** from the **Edit** menu in GraphWorX32, or click the **Data Mining ActiveX** button on the **ActiveX** toolbar. The **Data Mining ActiveX** will appear as shown below.



Data Mining ActiveX

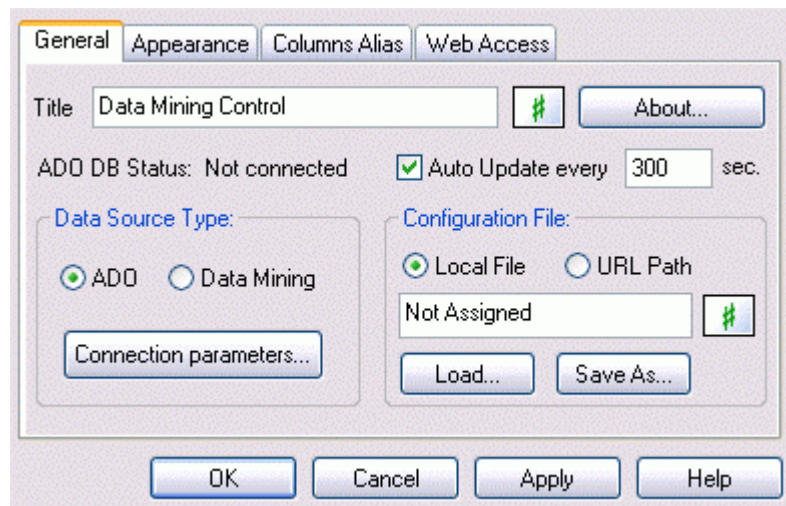
Data Mining ActiveX Properties

Double-clicking the Data Mining ActiveX opens the **Data Mining ActiveX Properties** dialog box, which contains the following property pages:

- **General**
- **Appearance**
- **Columns Alias**
- **Web Access**

General

The **General** tab of the **Data Mining ActiveX Properties** dialog box, shown below, allows you to connect to the desired data source.



Data Mining ActiveX Properties: General Tab

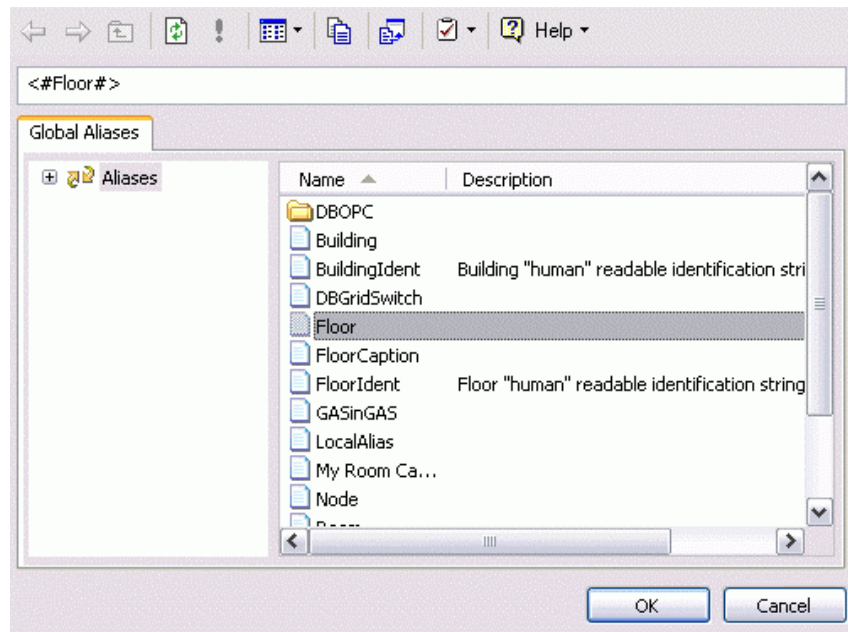
Configure the following **General** tab settings:

- **Title:** Enter a title for the ActiveX.
- **About:** Clicking the **About** button displays both technical support information and the About Box, which contains registration and serial number information as well as system resource information, such as memory and disk space available.
- **Auto Update (Used only for ADO Data Source type):** Specifies the frequency (in seconds) at which the Data Mining ActiveX updates data from the connected database.
- **Configuration File:** By default, information is stored as part of the container application's document file. You can also store information as part of a separate (.dbm) file. To save the .dbm file, under **Configuration File** click **Save As** to open the **Save Configuration Info** dialog box. Enter a name for the .dbm file in the **File Name** field, and then click **Save**.

You can also open an existing configuration file by clicking the **Load** button on the **General** tab and browsing for the file. The ActiveX will attempt to load the configuration file. If **URL Path** is selected, the ActiveX will use the specified network URL path upon loading the file.

Note: The ActiveX cannot be saved to a URL path. To create a URL file, save a configuration to a local file and copy the file to the desired network location. When the **URL Path** option is selected, you can also enable the **Use Local Settings** option, which, if checked, stores runtime changes to a local copy of the remote configuration file. It will use this local file every time the HTML page that hosts the Viewer is loaded inside the Web browser.

You can also select aliases to use for the configuration file. Clicking the **#** button and selecting **Insert Global Alias** from the pop-up menu opens the Global Aliases tab of the Unified Data Browser, as shown in the figure below. Select a global alias from the Unified Browser, which includes all global aliases in the global alias database. This eliminates the need to manually type in the alias name. All global aliases that are configured in the Global Alias Engine Configurator are conveniently available to choose from inside the browser. The tree control of the Global Alias Engine Configurator is mimicked in the tree control of the Global Alias Browser. Select a global alias by double-clicking the alias name (e.g. "Floor" in the figure below). The alias name appears at the top of the browser, which automatically adds the **<#** and **#>** delimiters to the alias name. Click the **OK** button.



Selecting an Alias From the Global Alias Browser

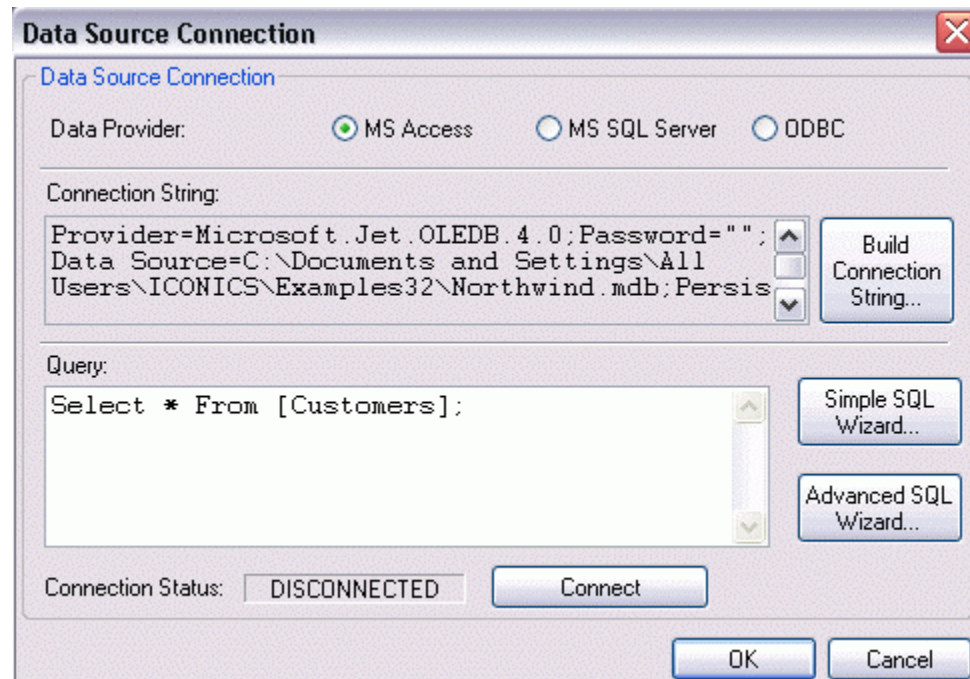
- Data Source Type:** The Data Mining ActiveX allows you to connect to two types of data sources: **ADO** and **Data Mining**. An **Active Data Object (ADO)** connection is a direct connection to a data source. A data mining connection is a connection to a data source through a preconfigured data mining tag (i.e. a data tag that was configured using the Data Mining Configurator). To define your database connections, clicking the **Connection Parameters** button. ADO and Data Mining data source connections are described in detail in the sections below.

Connecting to an ADO Data Source

An **Active Data Object (ADO)** connection is a direct connection to a data source.

To connect to an ADO data source:

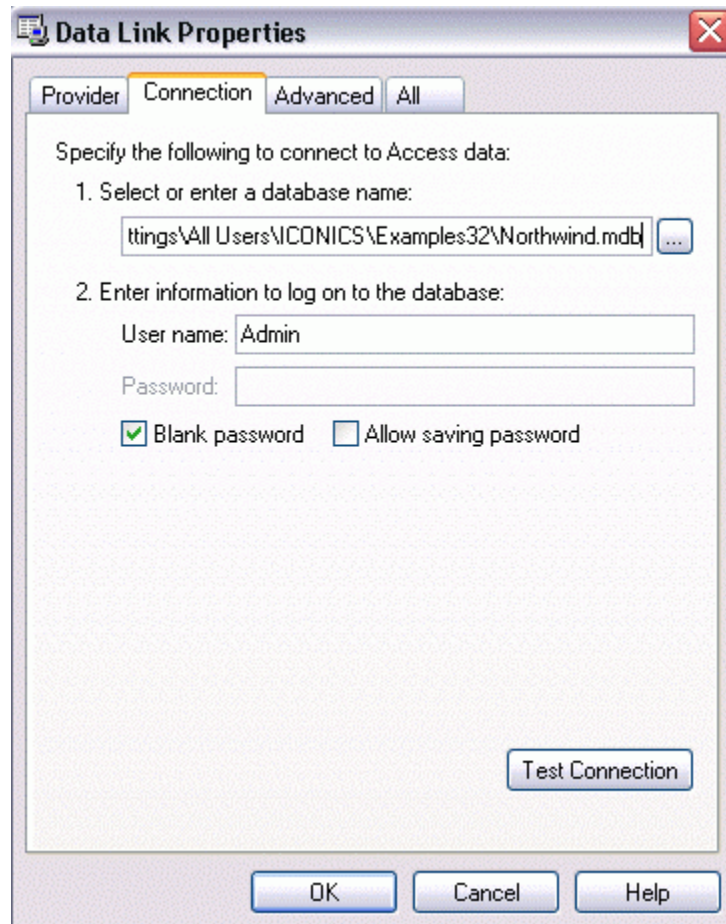
1. Select the **ADO** radio button on the **General** tab of the **Data Mining ActiveX Properties** dialog box.
2. Click the **Connection parameters** button. This opens the **Database Connection** dialog box, as shown below.
3. Under **Data Provider**, select from Microsoft Access, Microsoft SQL Server, or ODBC (Open Data Base Connectivity), as shown in the figure below. In this example, we will select Microsoft Access.



Connecting to a Database

4. Click the **Build Connection String** button. This opens the Microsoft **Data Link Properties** dialog box, as shown in the figure below. In the **Connection** tab specify the data source and then click **OK**. In this example, we will use the **Northwind.mdb** Microsoft Access database, which is included in the GENESIS32 installation in the **\Documents and Settings\All Users\ICONICS\Examples32** directory. For more information about database mining examples, please refer to the Data Mining Configurator help documentation.

Note: All data source connections are made through the **Data Link Properties** dialog box. The **Connection** tab settings may vary depending on which data provider you selected. Click the **OK** button. Click the **Help** button at any time to view the Microsoft Data Link help documentation.



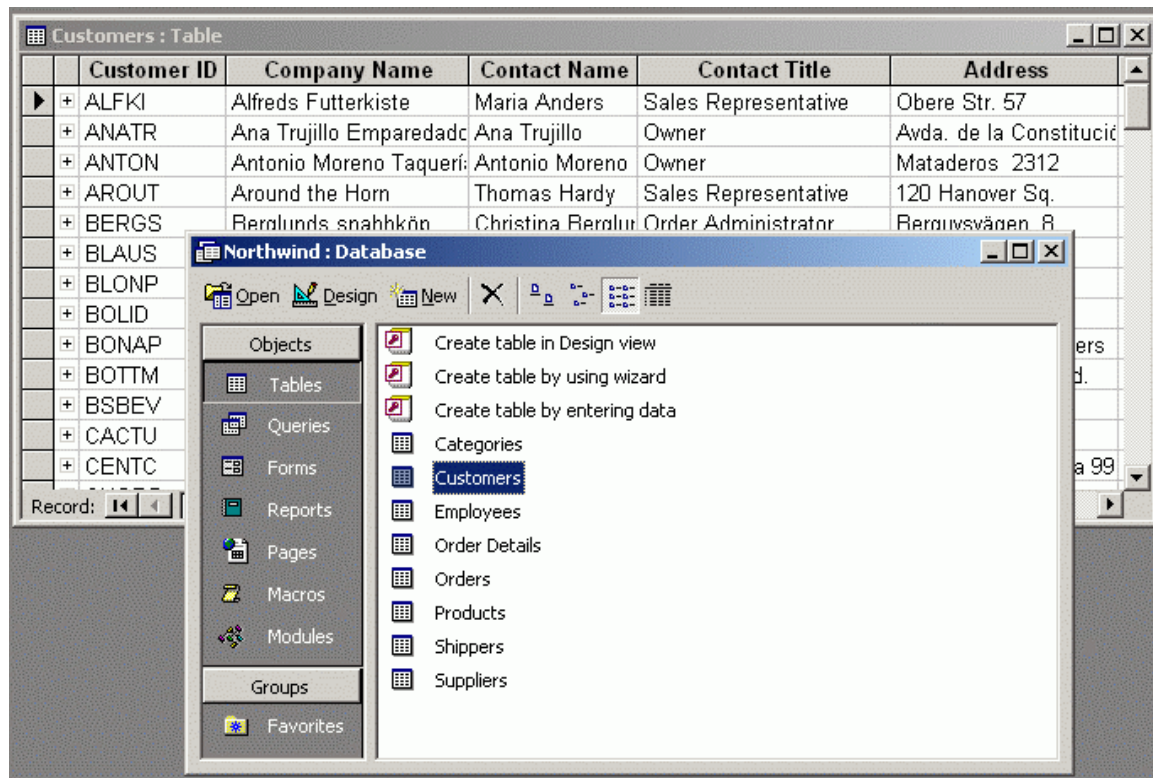
Specifying a Data Source

5. The data source reference appears in the **Database Connection** dialog box in the **OLE DB Connection String** field, as shown below. In the **Query** field, you must type in a database query to define which recordset will be read from the connected data source. In this example, we will use the following query:

Select * From [Customers];

This query will read all data from the **Customers** table in the Northwind.mdb database. The figure below shows this table as viewed in Microsoft Access. As you can see, it has several columns of data (e.g. Customer ID, Company Name, Contact Name, Contact Title, etc.).

Note: You can also use the Simple and Advanced query wizards to build your database query. For more information about the wizards, please see the SQL query wizards sections below. For more information about writing database queries, please refer to the Microsoft SQL Server or Oracle help documentation.



Customers Table in Northwind Database

- Click the **Connect** button to connect to the database, as shown in the figure below. Click **OK** to return to the **General** tab. Then click **OK** again.



Database Connection

7. When you enter the display into runtime mode, you will see the data items from the **Customers** table displayed in the Data Mining Grid, as shown in the figure below. The columns of data (e.g. Customer ID, Company Name, Contact Name, Contact Title, etc.) match the columns in the connected Northwind.mdb database.

Data Mining Control					
	CustomerID	CompanyName	ContactName	ContactTitle	Address
1	ALFKI	Alfreds Futterkiste	Maria Ander	Sales Repres	Obere Str. 57
2	ANATR	Ana Trujillo Emparedado	Ana Trujillo	Owner	Avda. de la Constit
3	ANTON	Antonio Moreno Taquería	Antonio Mor	Owner	Mataderos 2312
4	AROUT	Around the Horn	Thomas Har	Sales Repres	120 Hanover Sq.
5	BERGS	Berglunds snabbköp	Christina Ber	Order Admini	Berguvsvägen 8
6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Repres	Forsterstr. 57
7	BLONP	Blondel père et fils	Frédérique C	Marketing Ma	24, place Kléber
8	BOLID	Bólido Comidas preparad	Martín Som	Owner	C/ Araquil, 67
9	BONAP	Bon app'	Laurence Le	Owner	12, rue des Bouch
10	BOTTM	Bottom-Dollar Markets	Elizabeth Lin	Accounting	23 Tsawassen Blv
11	BSBEV	B's Beverages	Victoria Ash	Sales Repres	Fauntleroy Circus
12	CACTU	Cactus Comidas para llev	Patricio Simp	Sales Agent	Cerrito 333
13	CENTC	Centro comercial Moctez	Francisco C	Marketing Ma	Sierras de Granad
14	CHOPS	Chop-suey Chinese	Yang Wang	Owner	Hauptstr. 29
15	COMMI	Comércio Mineiro	Pedro Afons	Sales Associ	Av. dos Lusíadas,
16	CONSH	Consolidated Holdings	Elizabeth Br	Sales Repres	Berkeley Gardens
17	DRACD	Drachenblut Delikatesse	Sven Ottlieb	Order Admini	Walserweg 21
18	DUMON	Du monde entier	Janine Labru	Owner	67, rue des Cinqua
19	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George

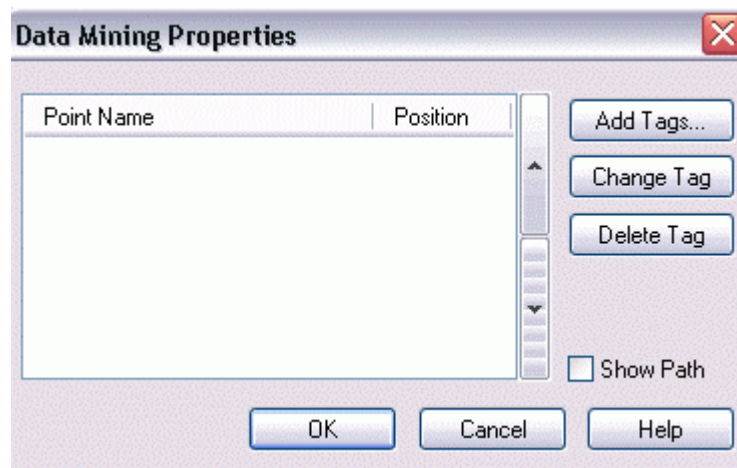
Table Column Displayed in Data Mining ActiveX Grid

Connecting to a Multidimensional Data Mining Data Source

A data mining connection is a connection to a data source through a preconfigured data mining tag (i.e. a data tag that was configured using the Data Mining Configurator).

To connect to a Data Mining data source:

1. Select the **Data Mining** radio button on the **General** tab of the **Data Mining ActiveX Properties** dialog box.
2. Click the **Connection parameters** button. This opens the **Data Mining Properties** dialog box, as shown below.
3. To create a connection, click the **Add Tags** button, as shown in the figure below.

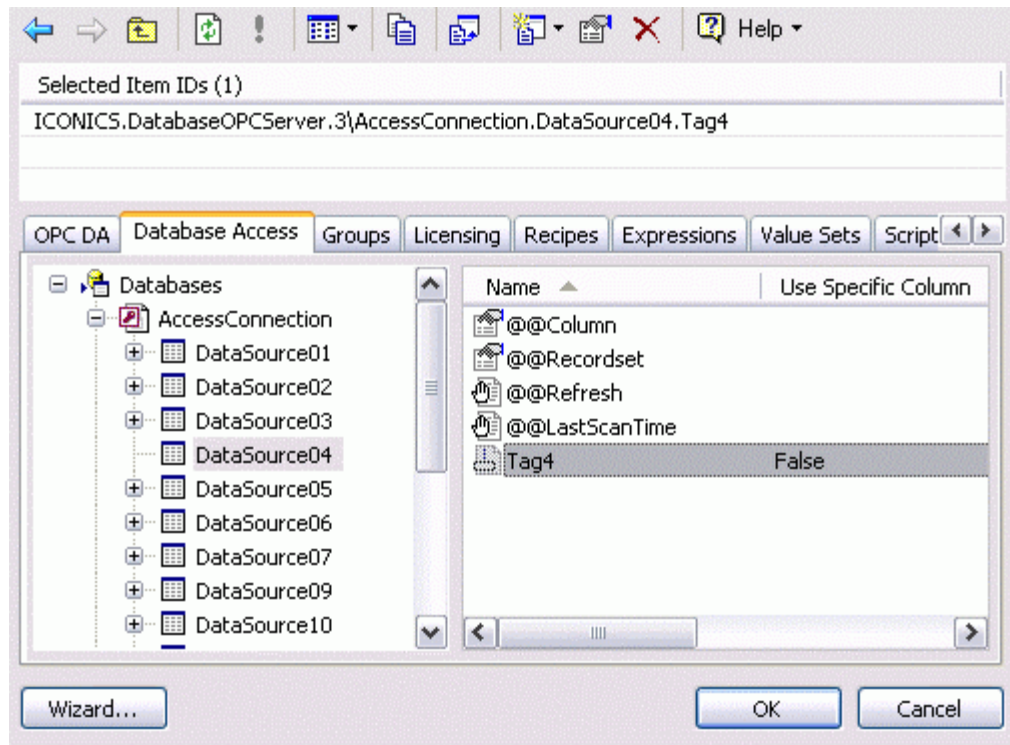


Data Mining Properties

4. This opens the Unified Data Browser. Select one or more data tags from the Unified Data Browser, as shown in the figure below. In this example, we will use the following multidimensional database mining tag:

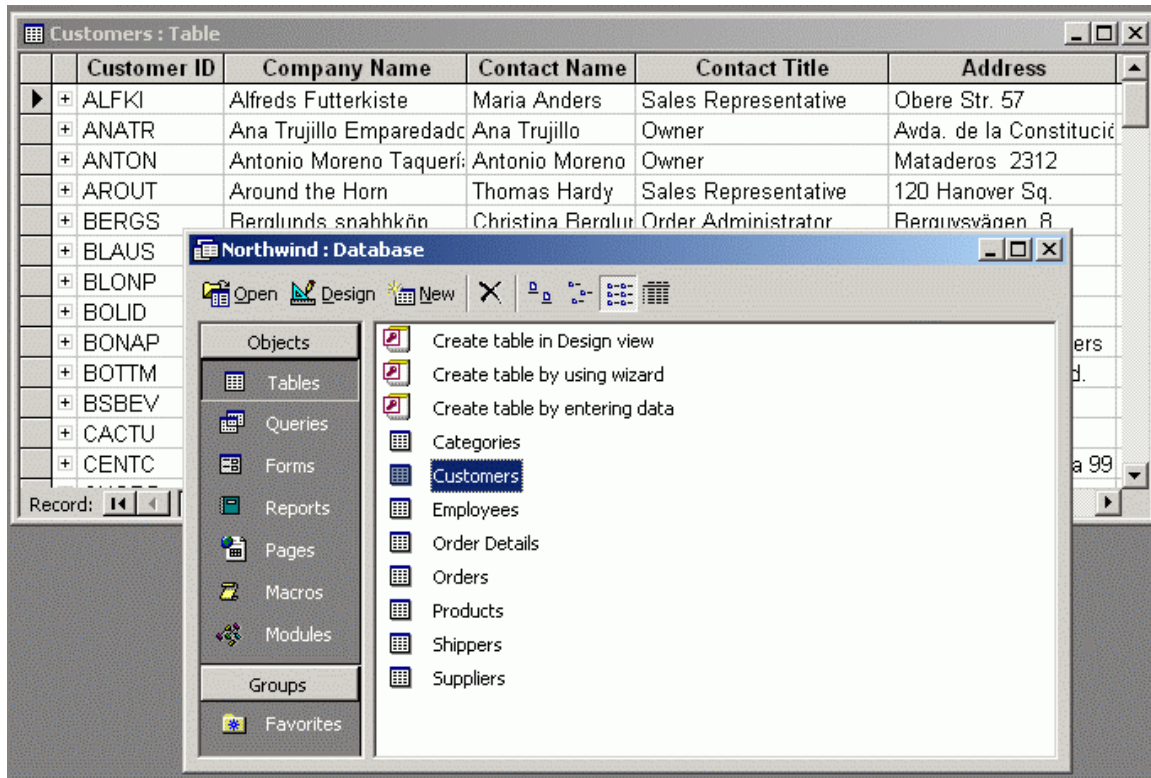
ICONICS.DatabaseOPCServer.3\AccessConnection.DataSource04.Tag4

This tag is defined in the **DBOPCConfig.mdb** example data mining configuration database, which is included in the GENESIS32 installation in the **Documents and Settings\All Users\ICONICS\Examples32** directory. This configuration database connects to the **Northwind.mdb** Microsoft Access database. For more information about database mining examples, configuring database access items and how to load the **DBOPCConfig.mdb** configuration database, please refer to the Data Mining Configurator help documentation.



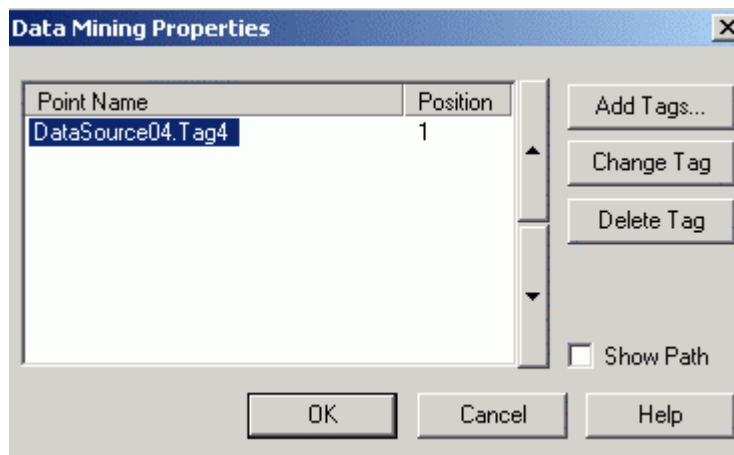
Selecting a Data Tag From the Unified Data Browser

5. This example data access tag, **AccessConnection.DataSource04.Tag4**, is a multidimensional array (i.e. reads data from multiple columns in the specified database table) that reads data from the **Customers** table in the Northwind.mdb database. The figure below shows this table as viewed in Microsoft Access. As you can see, it has several columns of data (e.g. Customer ID, Company Name, Contact Name, Contact Title, etc.). Click **OK**.



Customers Table in Northwind Database

6. The selected data tag appears in the **Data Mining Properties** dialog box under the **Point Name** list, as shown in the figure below. You have the following additional options:
 - **Add Tags:** You can choose additional data items to add to the list. When multiple data items are selected, click the up and down arrow buttons to change the order in which the data items are displayed in the Data Mining ActiveX Grid. The **Position** number indicates the position of each data item in the grid display.
 - **Change Tag:** Opens the Unified Browser, enabling you to replace the currently selected data item with a different data item.
 - **Delete Tag:** Removes the selected data item from the list.
 - **Show Path:** Displays the full path (i.e. server name and location) of the selected data item(s).



Data Mining List of Tags Present Inside the Grid

- Click **OK** to return to the **General** tab. Then click **OK** again. When you enter the display into runtime mode, you will see the multidimensional array of data items from the **Customers** table displayed in the Data Mining Grid, as shown in the figure below. The columns of data (e.g. Customer ID, Company Name, Contact Name, Contact Title, etc.) match the columns in the connected Northwind.mdb database as defined by the connected **AccessConnection.DataSource04.Tag4** data tag.

	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region
1	ALFKI	Alfreds Futterk	Maria Anders	Sales Repres	Obere Str.	Berlin	<Empty>
2	ANATR	Ana Trujillo Emp	Ana Trujillo	Owner	Avda. de l	México	<Empty>
3	ANTON	Antonio Moreno	Antonio Moren	Owner	Matadero	México	<Empty>
4	AROUT	Around the Hor	Thomas Hardy	Sales Repres	120 Hano	Londo	<Empty>
5	BERGS	Berglunds snab	Christina Bergl	Order Admini	Berguvsvä	Luleå	<Empty>
6	BLAUS	Blauer See Deli	Hanna Moos	Sales Repres	Forsterstr.	Mannh	<Empty>
7	BLONP	Blondel père et f	Frédérique Cit	Marketing Ma	24, place	Strasb	<Empty>
8	BOLID	Bólido Comidas	Marín Somme	Owner	C/ Araquil,	Madrid	<Empty>
9	BONAP	Bon app'	Laurence Lebi	Owner	12, rue de	Marseil	<Empty>
10	BOTTM	Bottom-Dollar M	Elizabeth Linc	Accounting	23 Tsawa	Tsawa	BC
11	BSBEV	B's Beverages	Victoria Ashwo	Sales Repres	Fauntleroy	Londo	<Empty>
12	CACTU	Cactus Comidas	Patricio Simps	Sales Agent	Cerrito 33	Bueno	<Empty>
13	CENTC	Centro comercia	Francisco Cha	Marketing Ma	Sierras de	México	<Empty>
14	CHOPS	Chop-suey Chin	Yang Wang	Owner	Hauptstr.	Bern	<Empty>
15	COMMI	Comércio Mineir	Pedro Afonso	Sales Associ	Av. dos Lu	São P	SP
16	CONSH	Consolidated H	Elizabeth Brow	Sales Repres	Berkeley	Londo	<Empty>
17	DRACD	Drachendorfer	Gregg O'Neil	Order Admini	Wolfsgr	Aachen	<Empty>

Viewing Data Items in Runtime Mode: Multidimensional Array

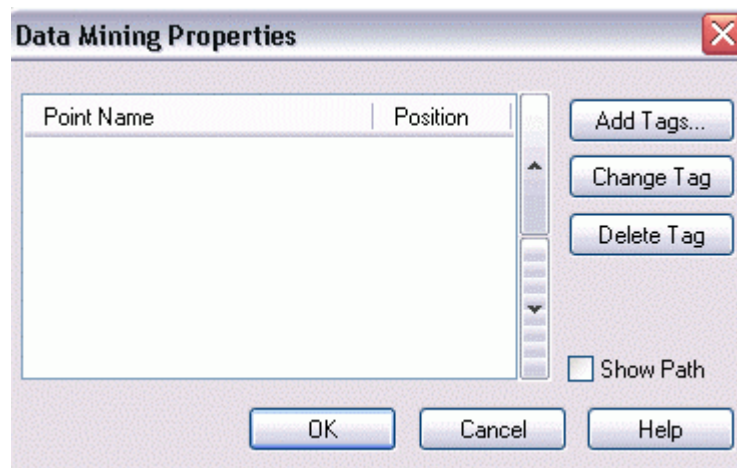
Note: If you want to browse for data mining tags on a remote computer, go to the **OPC DA** tab of the Unified Data Browser and browse to the computer in the **Network Neighborhood** tree.

Connecting to a One-Dimensional Data Mining Data Source

A data mining connection is a connection to a data source through a preconfigured data mining tag (i.e. a data tag that was configured using the Data Mining Configurator).

To connect to a Data Mining data source:

1. Select the **Data Mining** radio button on the **General** tab of the **Data Mining ActiveX Properties** dialog box.
2. Click the **Connection parameters** button. This opens the **Data Mining Properties** dialog box, as shown below.
3. To create a connection, click the **Add Tags** button, as shown in the figure below.

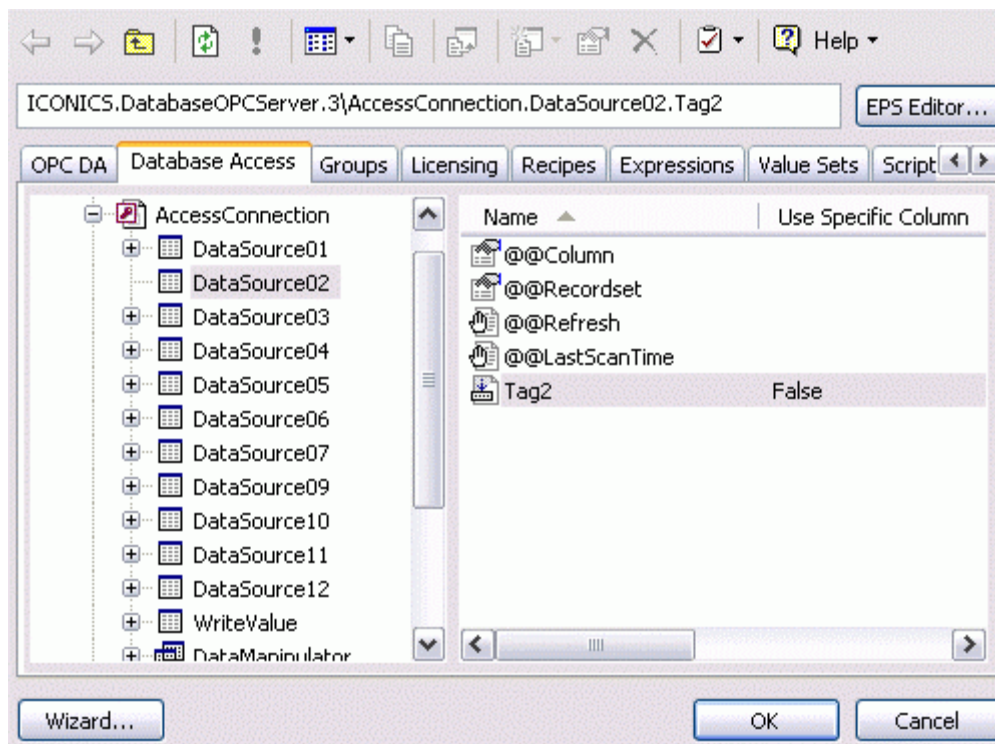


Data Mining Properties

4. This opens the Unified Data Browser. Select one or more data tags from the Unified Data Browser, as shown in the figure below. In this example, we will use the following one-dimensional database mining tag:

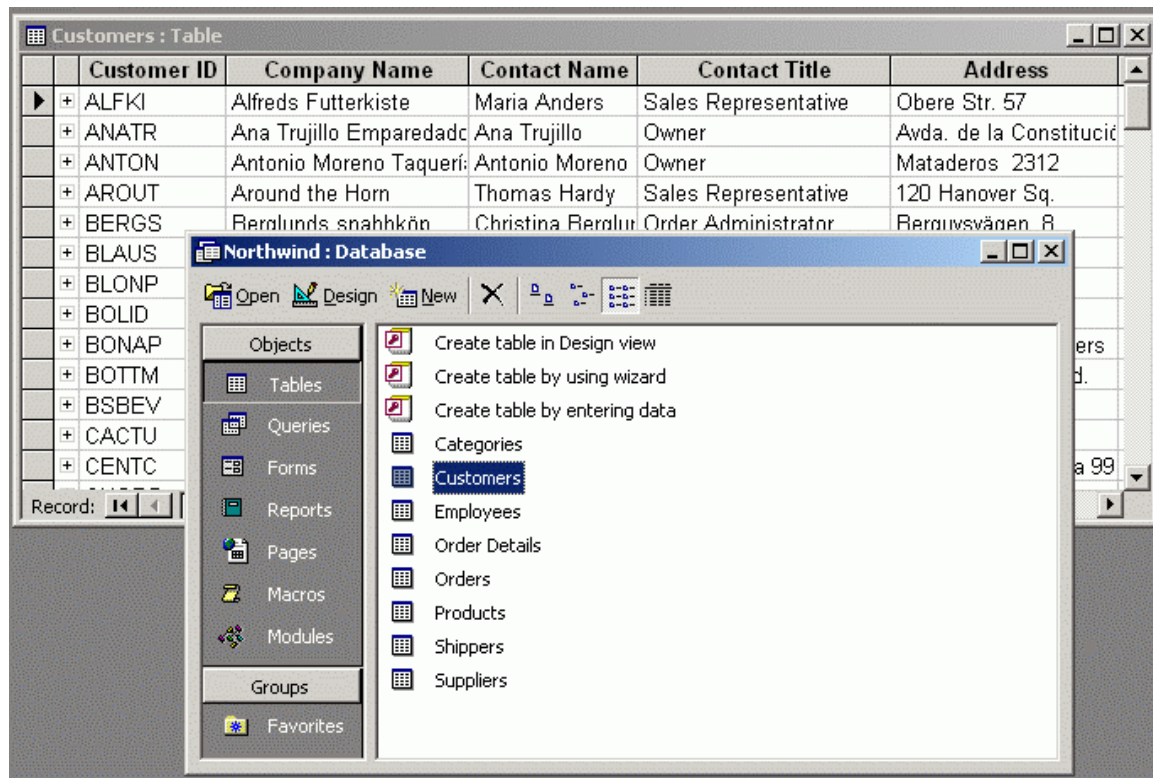
ICONICS.DatabaseOPCServer.3\AccessConnection.DataSource02.Tag2

This tag is defined in the **DBOPCConfig.mdb** example data mining configuration database, which is included in the GENESIS32 installation in the **Documents and Settings\All Users\ICONICS\Examples32** directory. This configuration database connects to the **Northwind.mdb** Microsoft Access database. For more information about database mining examples, configuring database access items and how to load the **DBOPCConfig.mdb** configuration database, please refer to the Data Mining Configurator help documentation.



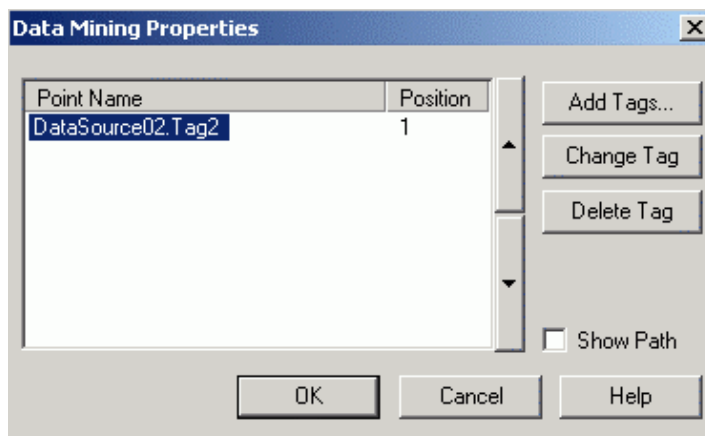
Selecting a Data Tag From the Unified Data Browser

5. This example data access tag, **AccessConnection.DataSource02.Tag2**, is a one-dimensional array (i.e. reads data from a single column in the specified database table) that reads data from the **Customers** table in the Northwind.mdb database. The figure below shows this table as viewed in Microsoft Access. As you can see, it has several columns of data (e.g. Customer ID, Company Name, Contact Name, Contact Title, etc.). This particular tag reads data only from the **ANTON** rowset in the **Customer ID** column. Click **OK**.



Customers Table in Northwind Database

6. The selected data tag appears in the **Data Mining Properties** dialog box under the **Point Name** list, as shown in the figure below. You have the following additional options:
 - **Add Tags:** You can choose additional data items to add to the list. When multiple data items are selected, click the up and down arrow buttons to change the order in which the data items are displayed in the Data Mining ActiveX Grid. The **Position** number indicates the position of each data item in the Grid display.
 - **Change Tag:** Opens the Unified Browser, enabling you to replace the currently selected data item with a different data item.
 - **Delete Tag:** Removes the selected data item from the list.
 - **Show Path:** Displays the full path (i.e. server name and location) of the selected data item(s).



Data Mining List of Tags Present Inside the Grid

7. Click **OK** to return to the **General** tab. Then click **OK** again. When you enter the display into runtime mode, you will see the one-dimensional array of data items from the **Customers** table displayed in the Data Mining Grid, as shown in the figure below. The grid displays the entire rowset ANTON in the **Customer ID** column in the connected Northwind.mdb database as defined by the connected **AccessConnection.DataSource02.Tag2** data tag.

Note: The array can be a horizontal array (i.e. view an entire rowset), or it can be a vertical array (i.e. it shows the values in a column). No matter what array you want to view, in the grid it is always shown as a column. Thus, even though the example shown in the figure below displays a rowset, the data in the row are displayed in the grid as a column.

Data Mining Control	
	CustomerID
1	ANTON
2	Antonio Moreno Taquería
3	Antonio Moreno
4	Owner
5	Mataderos 2312
6	México D.F.
7	<Empty>
8	05023
9	Mexico
10	(5) 555-3932
11	<Empty>

Viewing Data Items in Runtime Mode: One-dimensional Array

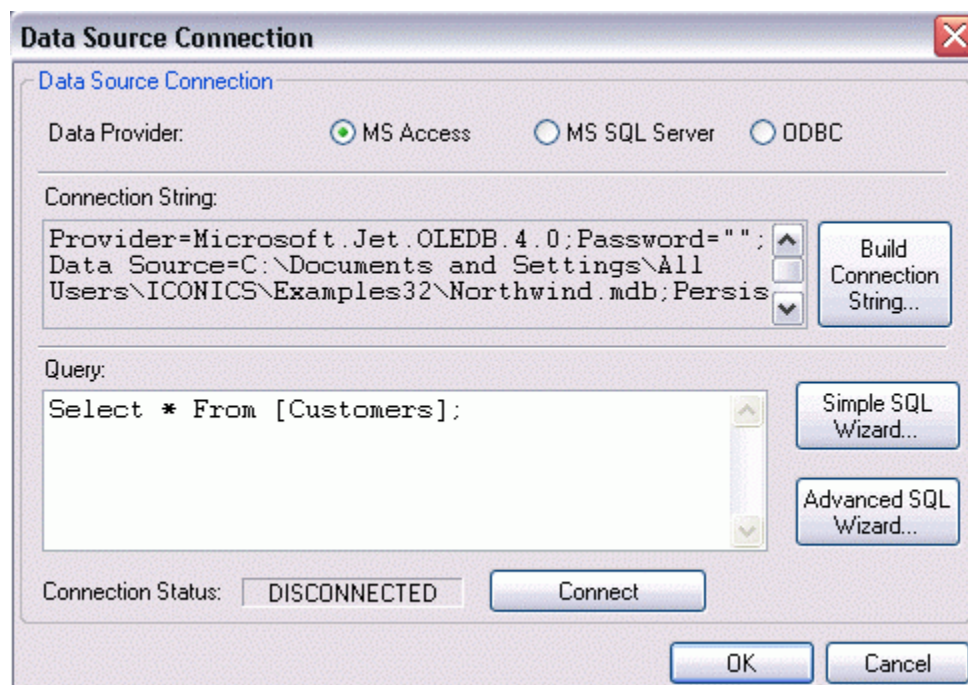
Note: If you want to browse for data mining tags on a remote computer, go to the **OPC DA** tab of the Unified Data Browser and browse to the computer in the **Network Neighborhood** tree.

Using SQL Wizards to Build Database Queries

When you are connecting to an ADO data source, you can either edit your database query manually in the **Data Source Connection** dialog, or you can use one of the following wizards to read data from the connected database, as shown in the figure below.

- Simple SQL Query Wizard
- Advanced SQL Query Wizard

Note: For more information about writing database queries, please refer to the Microsoft SQL Server or Oracle help documentation.



Connecting to a Database

Simple SQL Query Wizard

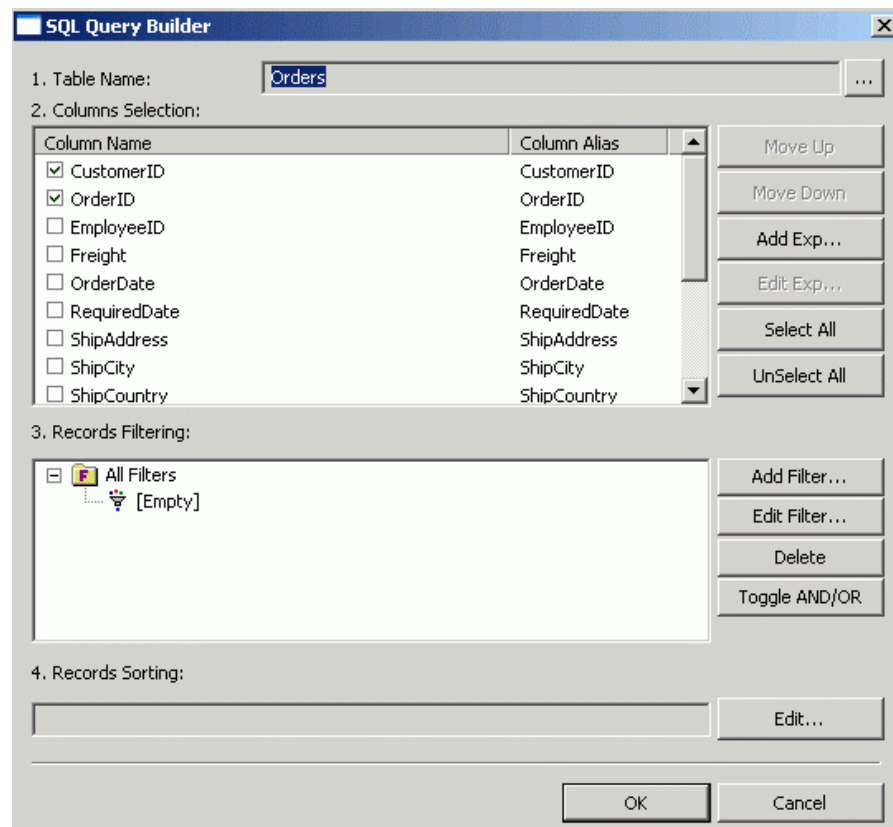
To create a basic SQL query for reading data from the connected database:

1. Click on the **Simple SQL Wizard** button on the **Data Source Connection** dialog, as shown in the figure below, as shown in the figure below.



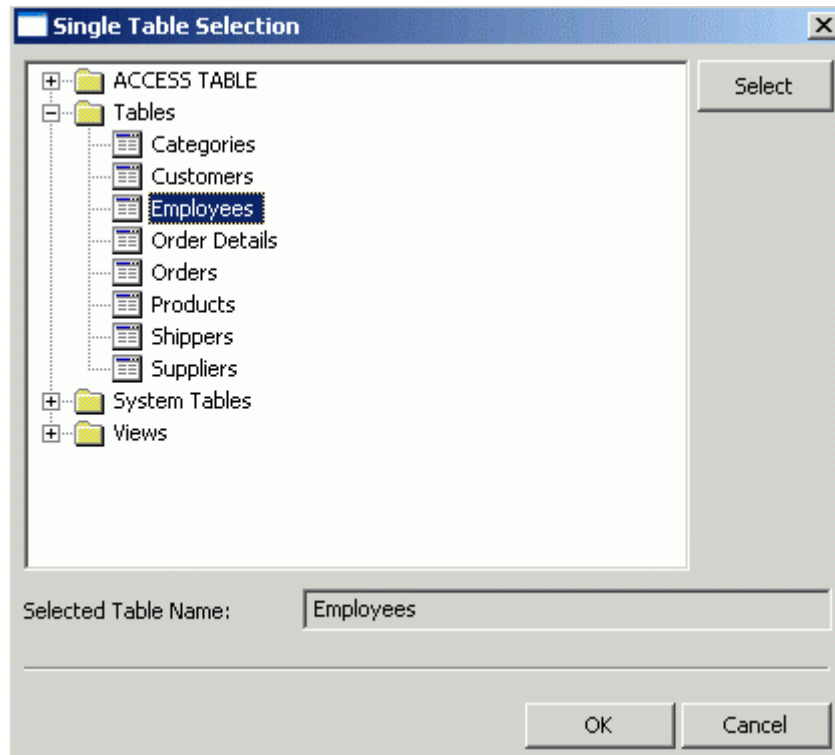
Creating a Simple SQL Query

1. This opens the **SQL Query Builder**, as shown in the figure below. This allows you to build a SQL read query by choosing from columns in a database table and filtering records from those columns.



SQL Query Builder

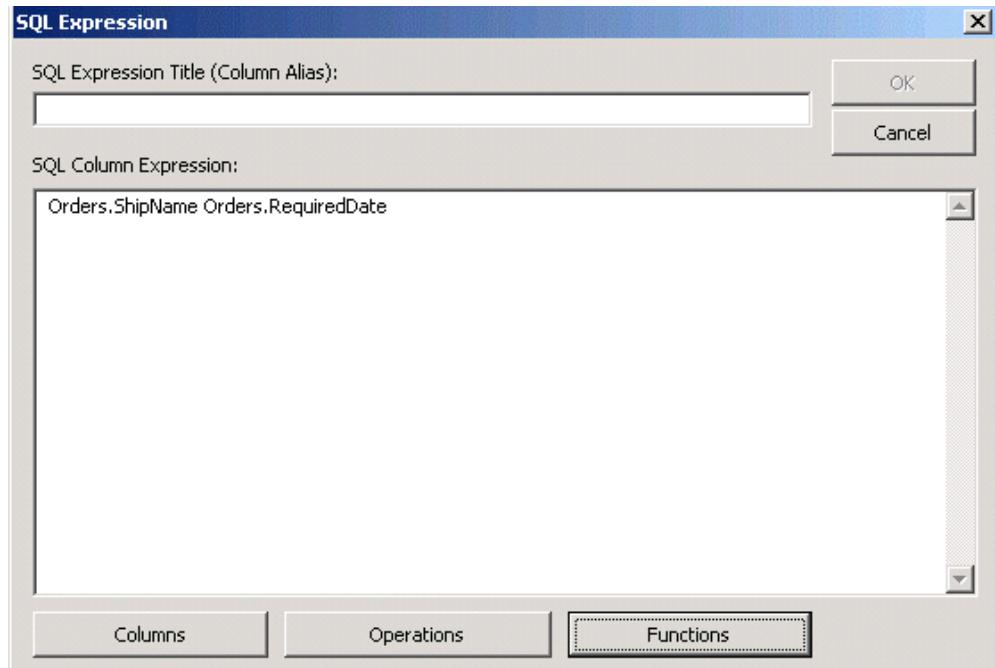
2. Under **Table Name**, click the ... button to select a table from a list of all the tables (e.g. "Employees") in the connected database.



Selecting Database Tables

3. The **Column Selection** field lists the names of all the columns (e.g. EmployeeID, LastName, etc.) in the selected table. Check all columns from which you want to read data. You can change the column order by highlighting column names and clicking the **Move Up** or **Move Down** buttons.
4. You can also add and/or edit SQL column names/expressions to each column by clicking the **Add Expression** button. In the **SQL Expression** dialog, you can edit the **Expression Title** and **SQL Column Expression** fields, as shown in the figure below. When you are finished, click **OK**. Then click the **Next** button to continue. Or you can build an **Expression Filter**, which allows you to manually edit SQL filtering expressions, as shown in the figure below. You can use the **Columns** button (lists column names from database), **Operators** button (e.g. addition, subtraction, multiplication, etc.), or **Functions** button (mathematical, string, date/time, etc.) button to browse for the corresponding SQL elements.

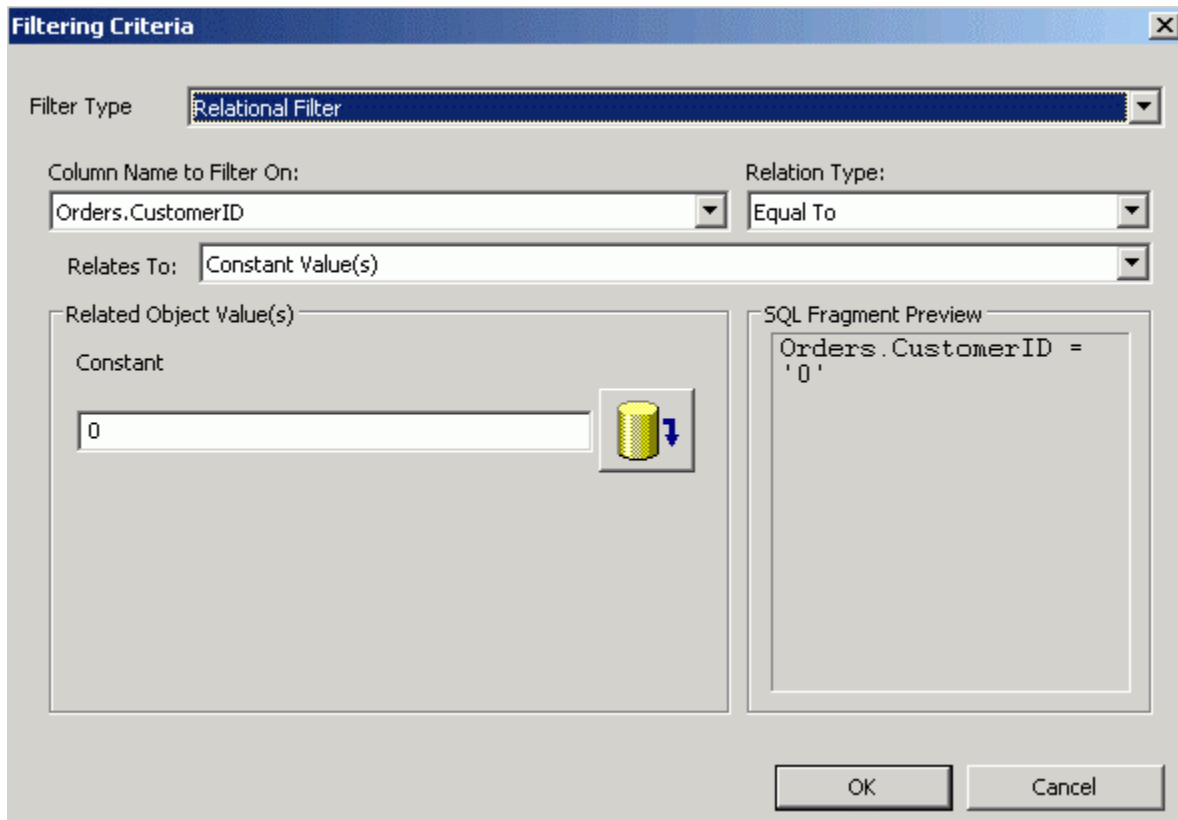
Note: Please refer to the SQL language documentation for details about SQL expressions.



Editing SQL Column Expressions

5. Now that you have selected which database column information will be displayed, you can now define parameters for filtering or restricting data from those columns. Enter any filters you would like to apply to the data by clicking the **Add Filter** button. The tree-like structure for record filters in the SQL Query Builder displays how they are combined, using AND / OR Boolean operators and parentheses. The rule is simple; each filter node is first combined with its "children," and then the resulting expression is combined with the "siblings." Click the **Toggle AND/OR** button to switch between the AND / OR boolean operators.
6. In the **Filtering Criteria** dialog box, shown in the figure below, helps you write proper syntax for SQL filtering expressions. The **Column Name to Filter On** drop-down list contains the columns you can use in this expression, and **Relation Type** list includes SQL functions.

Note: Please refer to the SQL language documentation for details about SQL expressions.



The **Filtering Criteria** dialog box is shown with the following settings:

- Filter Type:** Relational Filter
- Column Name to Filter On:** Orders.CustomerID
- Relation Type:** Equal To
- Relates To:** Constant Value(s)
- Related Object Value(s):**
 - Constant: 0
- SQL Fragment Preview:**

```
Orders.CustomerID = '0'
```

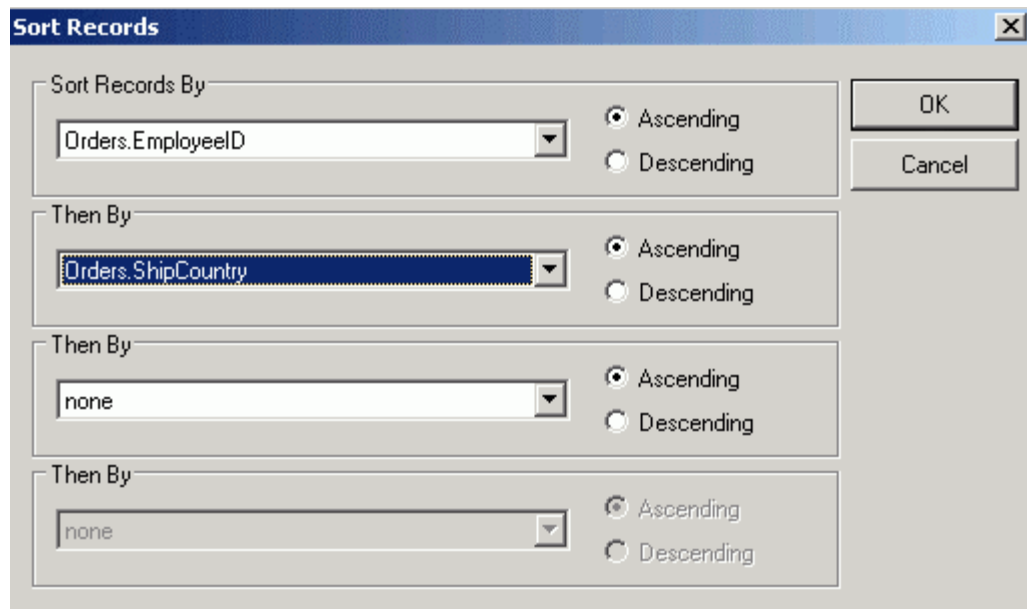
Buttons: OK, Cancel

Records Filtering Criteria

- To sort the database records, click the **Edit** button on the **SQL Query Builder** dialog box. The **Sort Records** dialog box appears, as shown in the figure below. Now that you have defined parameters for filtering or restricting data from columns, here you will specify the priority by which the columns and the records in each column are sorted. You can choose to sort on any of the selected columns, as shown in the figure below, or leave them unsorted.

To determine the order in which the data will be displayed within in each column, select that column from the drop-down list and select one of the following sorting options:

- **Ascending:** Sorts the data in the column in ascending order (i.e. from least to greatest).
- **Descending:** Sorts the data in the column in descending order (i.e. from greatest to least).



SQL Records Sorting

8. When you are finished editing your SQL query, click the **OK** button on the **SQL Query Builder** dialog box. The SQL statement is shown in the **Data Source Connection** dialog.

Advanced SQL Query Wizard

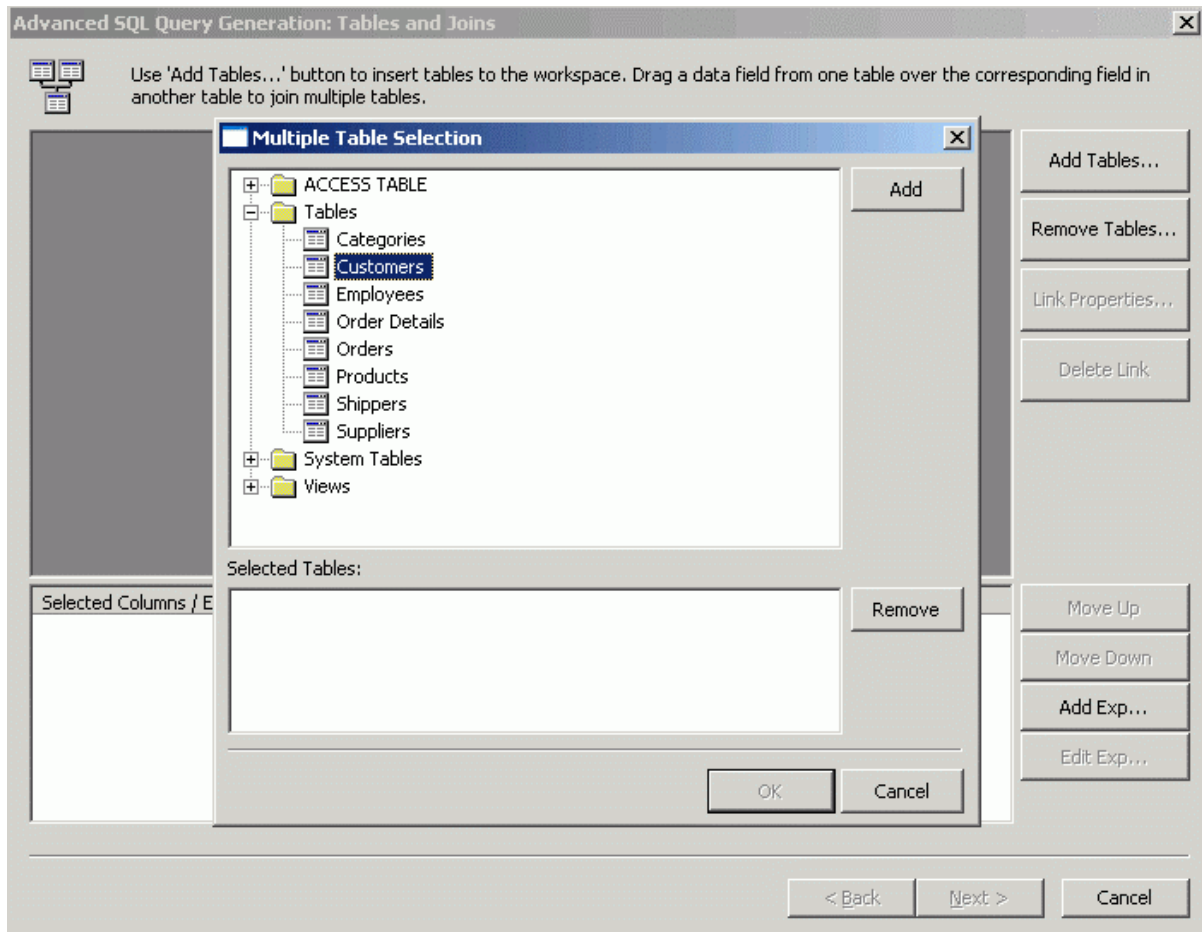
To create an advanced SQL query for reading data from the connected database:

1. Click on the **Advanced SQL Wizard** button on the **Data Source Connection** dialog, as shown in the figure below.



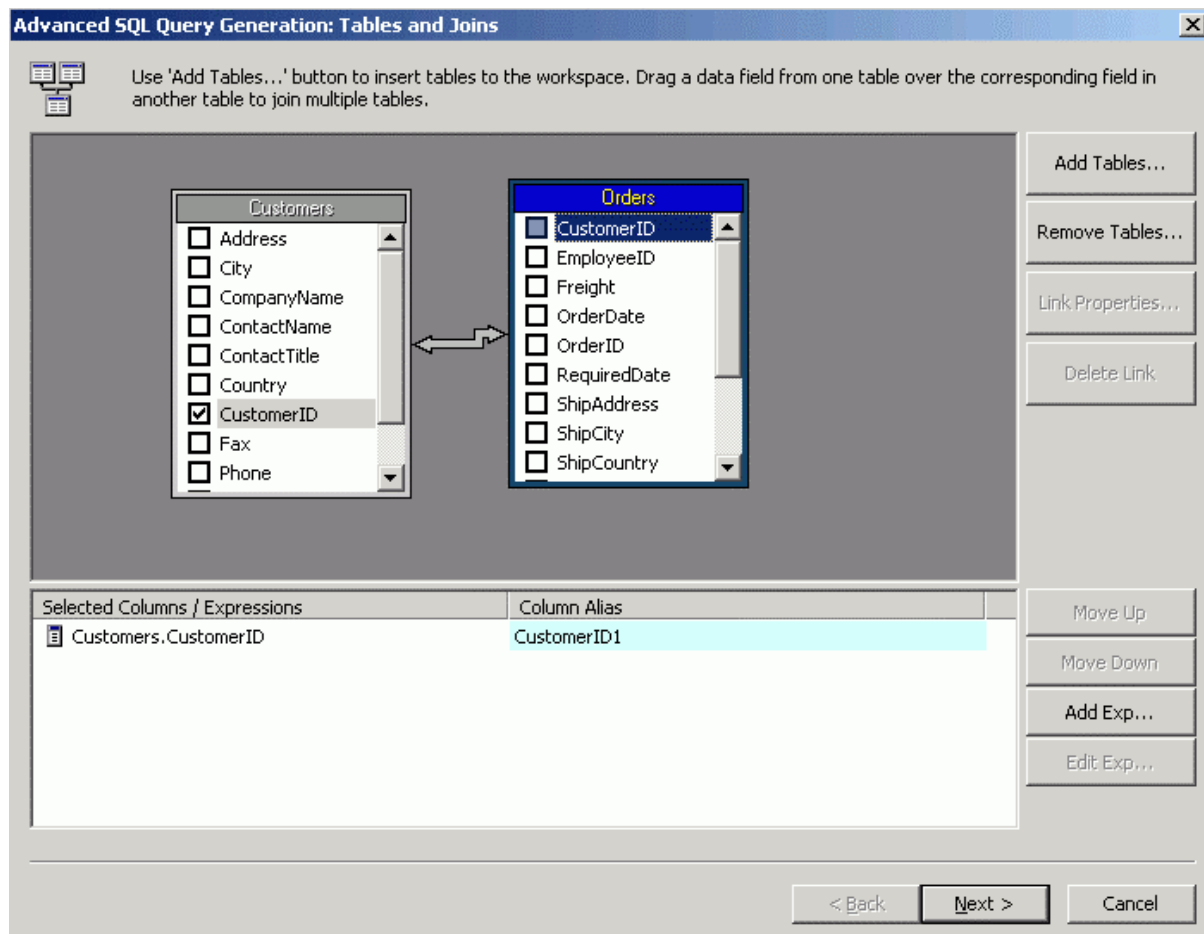
Creating an Advanced SQL Query

2. After the database schema loads (this may take several minutes depending on the size and location of the selected database), the **Advanced SQL Query** dialog box opens, as shown in the figure below. The query wizard walks you through a series of dialogs to help you create your SQL query.
3. Click the **Add Tables** button to display a list of the tables in the database, as shown in the figure below. This dialog displays a list of data tables from the data source that you specified.
4. Select the data tables to include in the database query and click the **Add** button, as shown in the figure below. Click **OK**.



Selecting Data Tables to Include in Database Query

5. The wizard displays all columns from all the tables you selected. For each table you chose, select which data columns to include in the database query by checking the box next to the name of each column, as shown in the figure below. For this example database query, the **Orders** and **Customers** tables are selected, as shown in the figure below.
6. If you select multiple tables, you can join the tables together by simply dragging a data column from one table over to the corresponding field in another table. A double-headed arrow that points to both connected tables will be drawn automatically, indicating that a joint between the two tables exists, as shown in the figure below. In this example, the **CustomerID** column in the **Customers** table is linked to the **CustomerID** column in the **Orders** table.



Selecting Data Columns to Include in Query

- To view the link details, click on the **Link Properties** button. This opens the **Table Link Properties** dialog box, which lists the names of the tables and columns that are joined. Use the expression options under **SQL Join Type** to define the relationship between the two columns (e.g. Equal, Greater than, Less than, etc.). You can also manually write a **Custom Join Condition** by clicking the **Edit** button.

Note: Please refer to the SQL language documentation for details about SQL joins.

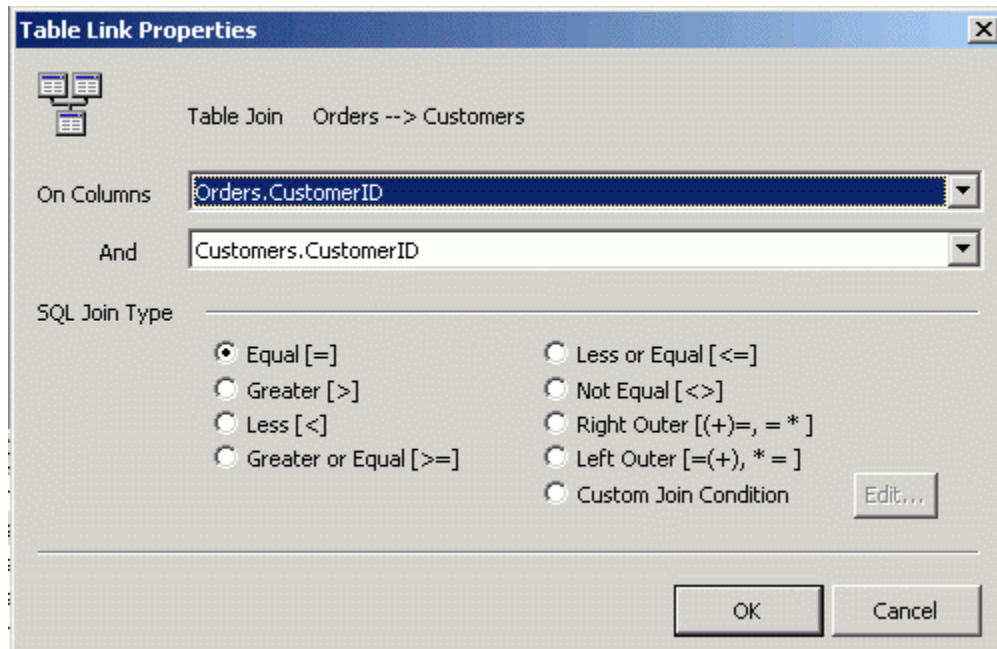
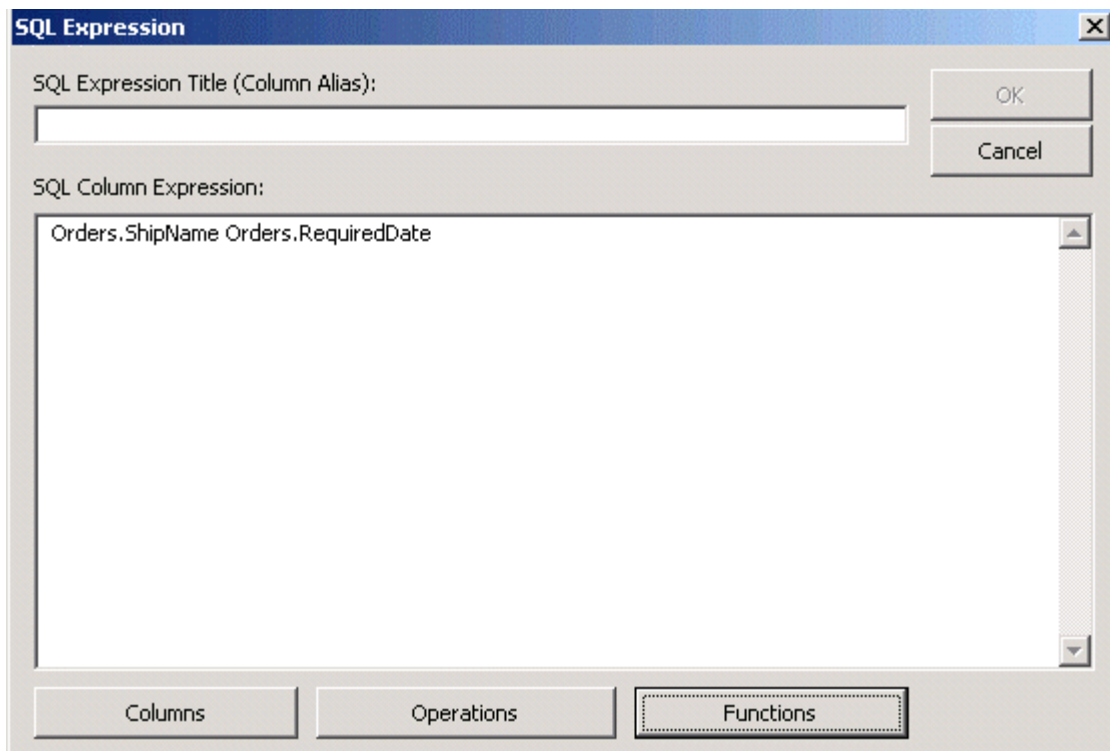


Table Link Properties

8. The selected data columns are also listed in the bottom pane of the window. You determine which columns or expressions will be included in the "SELECT" part of the SQL query as well as the column titles that will be displayed on the mobile device. Click the **Move Up/Move Down** buttons to change the order in which the columns will be displayed. To change an existing column/expression, click the **Edit Exp...** button. To insert a column or to create a new expression, click the **Add Exp...** button. This opens the **SQL Expression** dialog, as shown in the figure below. In the **SQL Expression** dialog, you can edit the **Expression Title** and **SQL Column Expression** fields, as shown in the figure below. When you are finished, click **OK**. Then click the **Next** button to continue. Or you can build an **Expression Filter**, which allows you to manually edit SQL filtering expressions, as shown in the figure below. You can use the **Columns** button (lists column names from database), **Operators** button (e.g. addition, subtraction, multiplication, etc.), or **Functions** button (mathematical, string, date/time, etc.) button to browse for the corresponding SQL elements.

Note: Please refer to the SQL language documentation for details about SQL expressions.

Click the **Next** button to continue working through the SQL Query Wizard.



Editing SQL Expressions

9. The **Records Grouping and Aggregation** dialog box appears, as shown in the figure below, listing the selected tables and columns. Grouping applies to records in a data column that have the same value. Click in the **Grouping** field to change the grouping options for the data tables (or leave them unchanged). Be sure to assign an **Aggregate Function** (e.g. Count, Average, Maximum, Minimum, Sum, Standard Deviation, etc.) to each of the columns you did not include in the grouping list. The records of columns that are assigned the same aggregate function will be grouped together in the query result.

Advanced SQL Query Generation: Records Grouping and Aggregation

Click in the 'Grouping' field to modify the grouping options for the record fields, or click the 'Next' button to leave the grouping options unchanged.

Please note: if you enable grouping option on certain fields, you should also select an aggregation function for every field that doesn't have grouping attribute.

Grouping type: Grouping using the SQL 'GROUP BY' statement

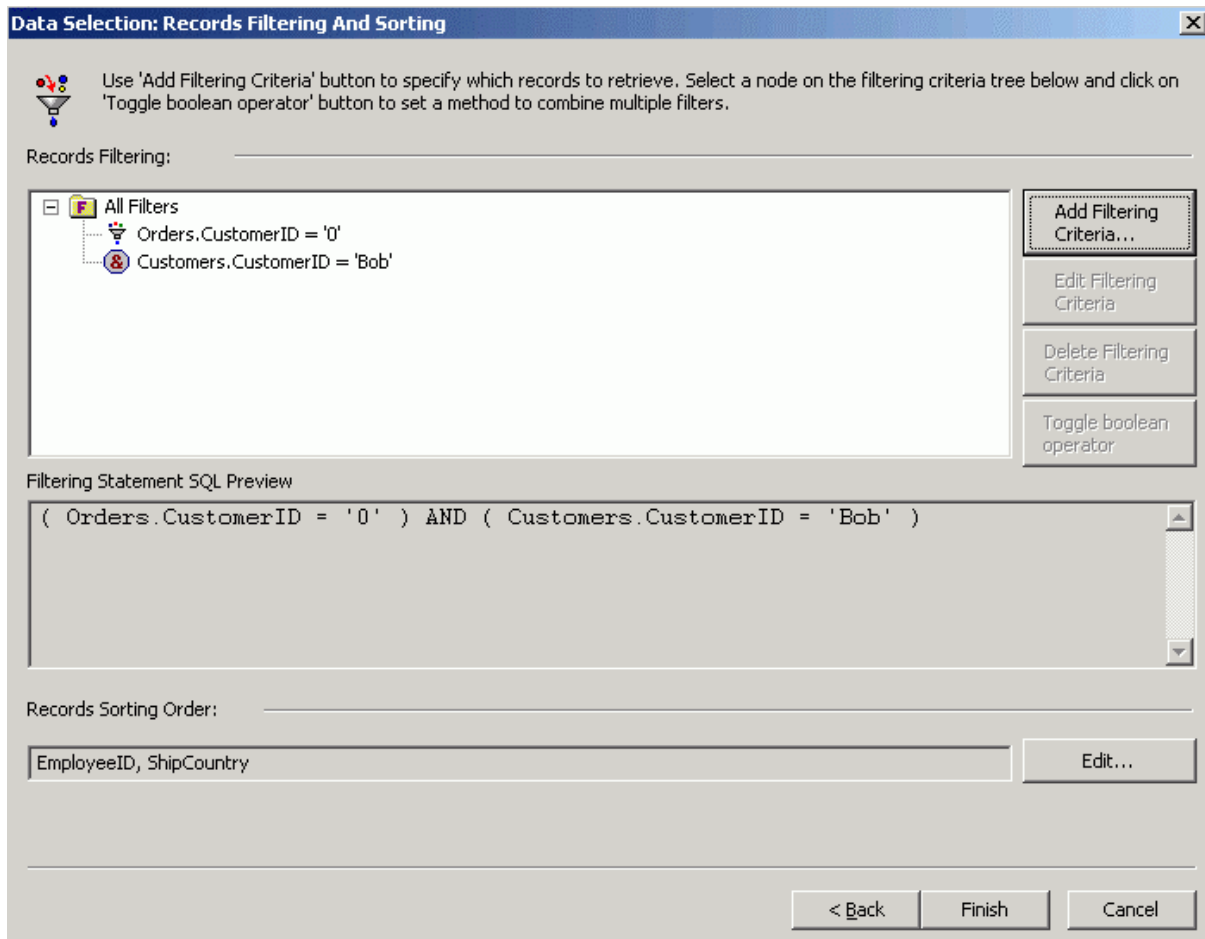
Selected Fields	Grouping	Aggregate Function
Orders.CustomerID	On	N/A
Customers.CustomerID	On	N/A
Orders.OrderDate	Off	none
Orders.RequiredDate	Off	none
Customers.Fax	Off	none

< Back Next > Cancel

Records Grouping and Aggregation

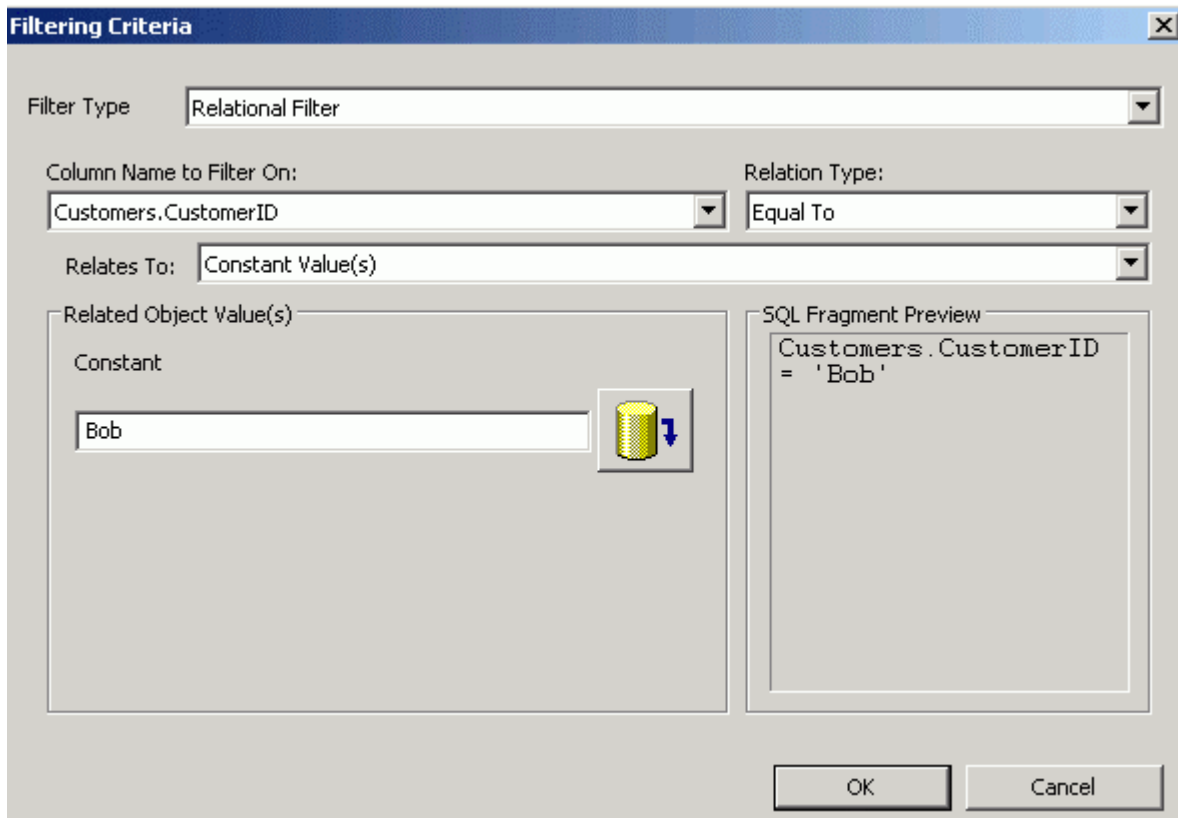
- Click the **Add Filtering Criteria** button to enter any filters you would like to apply to the data, as shown in the figure below, or click **Next** to bypass the filter configuration page. You can drag and drop filters from one tree node to another, or if you create a new filter while some other filter is selected on the tree, the newly created one will be shown as a child tree node. The tree-like structure for filters displays how they are combined, using AND / OR Boolean operators and parentheses. The rule is simple; each filter node is first combined with its "children," and then the resulting expression is combined with the "siblings." Try dragging and dropping several filters onto one another and watch how the SQL preview changes to get an idea of how this works. Click the **Toggle Boolean Operator** button to switch between the AND / OR boolean operators.

Note: Please refer to the SQL language documentation for details about SQL expressions.



SQL Records Filtering

11. The **Filtering Criteria** dialog box appears, as shown in the figure below. Now that you have selected which database column information will be displayed, you can now define parameters for filtering or restricting data from those columns. This dialog allows you to establish relationships between different types of columns in the data tables you selected. Under **Filter Type**, select either **Relational Filter**, as shown in the figure below, or **Expression Filter**, which allows you to manually edit SQL filtering expressions. Under the **Column Name to Filter On** field, select a column name from the drop-down list (e.g. [Customers].[Customer ID]). Under **Relates To**, select a column or a constant value with which the column name is associated. In the example below, the Customer ID is equal to "Bob" (a constant value). Under **Relation Type**, select a relational filter expression (e.g. Equal To, Greater Than, Less Than, etc.) from the drop-down list to define the relationship between the values. Click **OK**. Click the **Next** button to continue working through the SQL Query Wizard.



The **Filtering Criteria** dialog box is shown with the following settings:

- Filter Type:** Relational Filter
- Column Name to Filter On:** Customers.CustomerID
- Relation Type:** Equal To
- Relates To:** Constant Value(s)
- Related Object Value(s):**
 - Constant: Bob
- SQL Fragment Preview:**

```
Customers.CustomerID
= 'Bob'
```

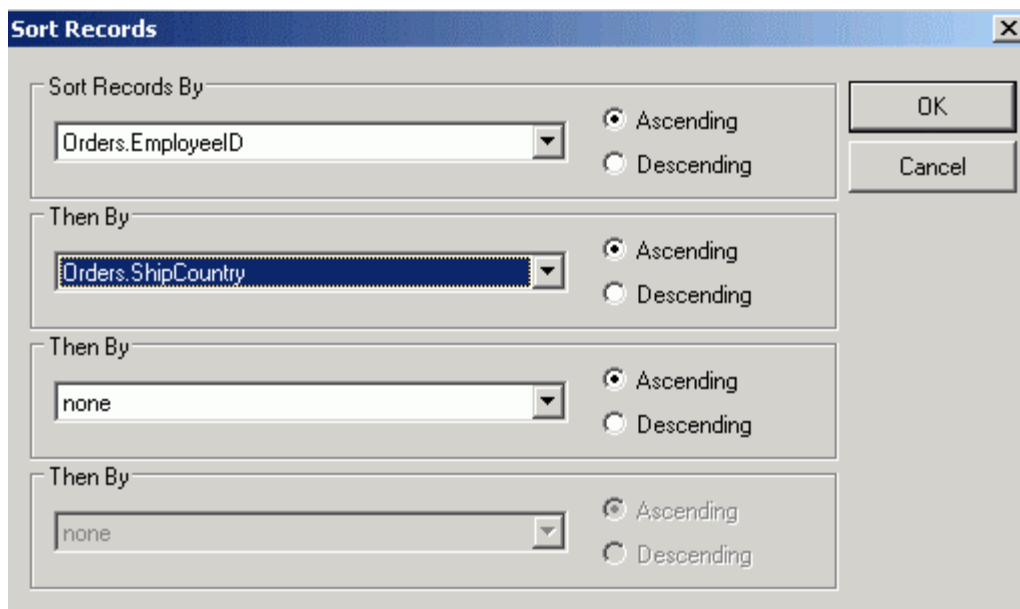
Buttons: OK, Cancel

Filtering Criteria

12. To sort the database records, click the **Edit** button on the **SQL Query Builder** dialog box. The **Sort Records** dialog box appears, as shown in the figure below. Now that you have defined parameters for filtering or restricting data from columns, here you will specify the priority by which the columns and the records in each column are sorted. You can choose to sort on any of the selected columns, as shown in the figure below, or leave them unsorted.

To determine the order in which the data will be displayed within in each column, select that column from the drop-down list and select one of the following sorting options:

- **Ascending:** Sorts the data in the column in ascending order (i.e. from least to greatest).
- **Descending:** Sorts the data in the column in descending order (i.e. from greatest to least).

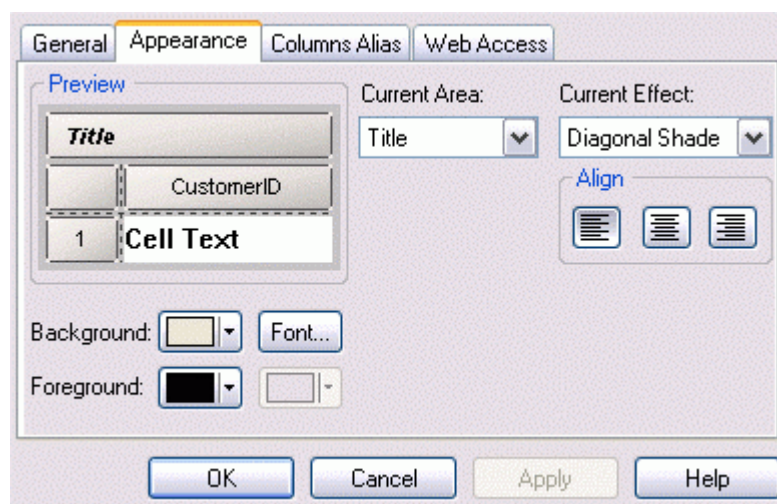


SQL Records Sorting

13. Click the **Finish** button to complete the query build.

Appearance

The **Appearance** tab of the **Data Mining ActiveX Properties** dialog box, shown below, determines how the data grid looks in runtime mode. The **Preview** window allows you to see your changes in configuration mode.



Data Mining ActiveX Properties: Appearance Tab

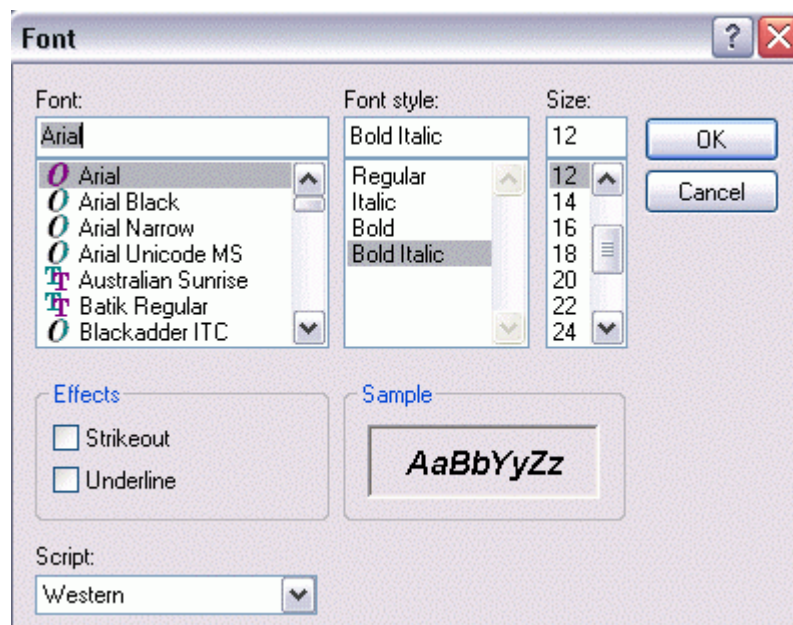
Under **Current Area**, select a data grid item to configure (Title, Headers, Cell Text, or Background) and then configure the appearance of the item.

The colors section allows you to choose colors for the background of the display, the background color of the text, the background color of the header and the foreground text color. Select colors for the **Background** and **Foreground** by clicking the color palettes. For the background, you can also choose the color of

the horizontal and vertical grid lines in the data grid. A **Color** box appears, allowing you to choose a preconfigured color or to create your own color using the **Other** button.

Font

For **Title**, **Cell Text** and **Headers**, click the **Font** button to specify the font size, style, and text effects (i.e. underline and strikethrough) for the data grid, as shown in the figure below.



Selecting a Font Style for the Viewer

Visual Effects and Alignment

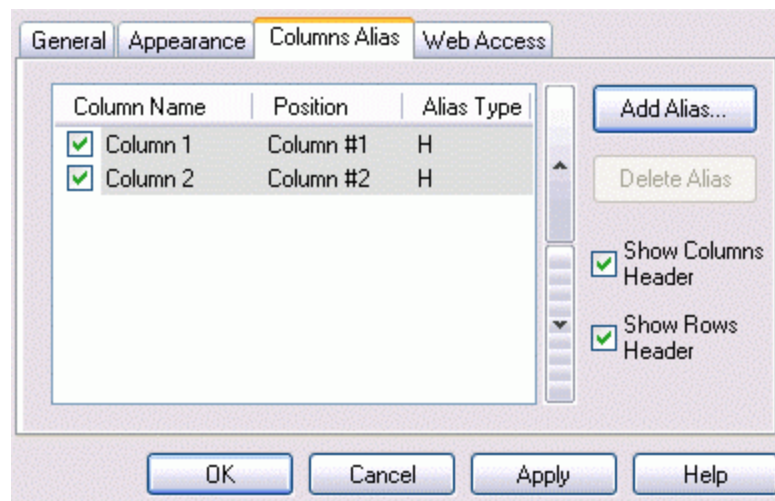
For **Title**, **Cell Text** and **Headers**, the **Align** buttons determine the justification (left, center or right) of the text.

For **Title** and **Headers**, you can apply various stylistic graphical effects (e.g. bumps, gradients, and shading) by selecting from the drop-down list under Current Effect.

For the background, you can select **Line Style** (e.g. solid, dashed, dotted, etc.) for both **Horizontal** and **Vertical** grid lines that appear in the data grid.

Columns Alias

The **Columns Alias** tab of the **Data Mining ActiveX Properties** dialog box, shown below, allows you to create column header names and associate each name with a column that is displayed in the Data Mining ActiveX Grid. You can also create "smart" aliases that allow you to substitute some text or an image for a value or a string in the selected column.

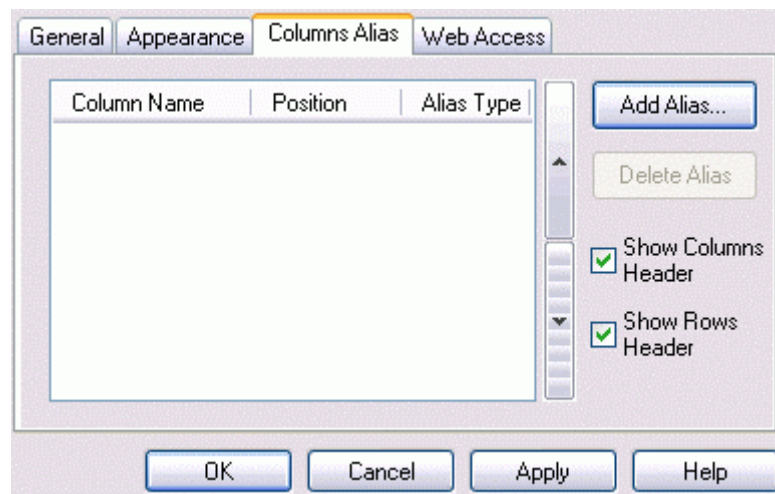


Data Mining ActiveX Properties: Columns Alias Tab

Creating Column Header Aliases

To create a column header alias:

1. On the **Columns Alias** tab, click the **Add Alias** button, as shown in the figure below.



Adding a Column Alias

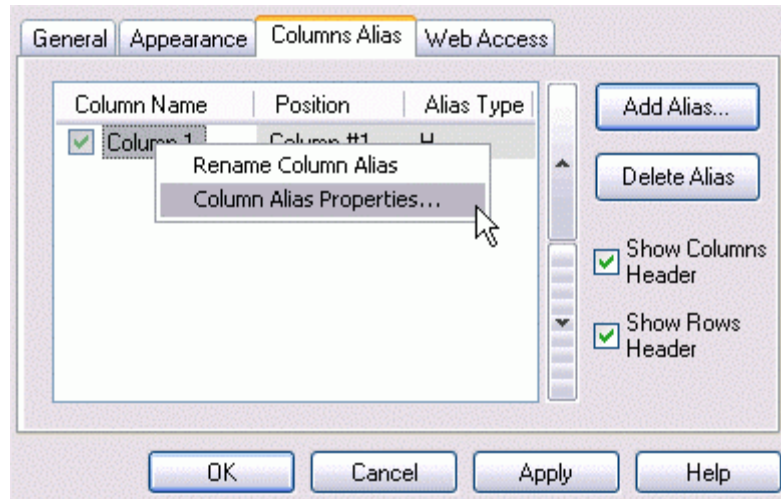
2. This opens the **Column Alias Property** dialog box, as shown in the figure below. Check the **Header Alias** check box. In the **Alias Name** field, specify a column name. Then specify a **Column Position** number, which corresponds to the position of the column where the alias will be applied in the Grid display. You can also specify the justification (left, right, or center) of the alias text in the column. Click **OK**.

Note: With column aliasing, you can define aliases for the columns that you need. Column positions do not need to be sequential. For example, you could define two aliases: one for column position 3 and another for column position 13.

Note: You can also click the # button to select a global alias from the global alias browser.

Column Alias Property

3. The new column alias and position number appear in the under the **Column Name** list, as shown in the figure below. You have the following additional options:
 - **Add Alias:** You can add additional column aliases to the list. When multiple column aliases are selected, click the up and down arrow buttons to change the order in which the columns are displayed in the Data Mining ActiveX Grid. The **Position** number corresponds to the position of each data item in the Grid display.
 - **Delete Alias:** Deletes the selected column alias from the list.
 - **Rename Column Alias:** Enables you to change the name of the selected column alias.
 - **Column Alias Properties:** Opens the **Column Alias Property** dialog box for the selected column alias.



Editing Column Alias Properties

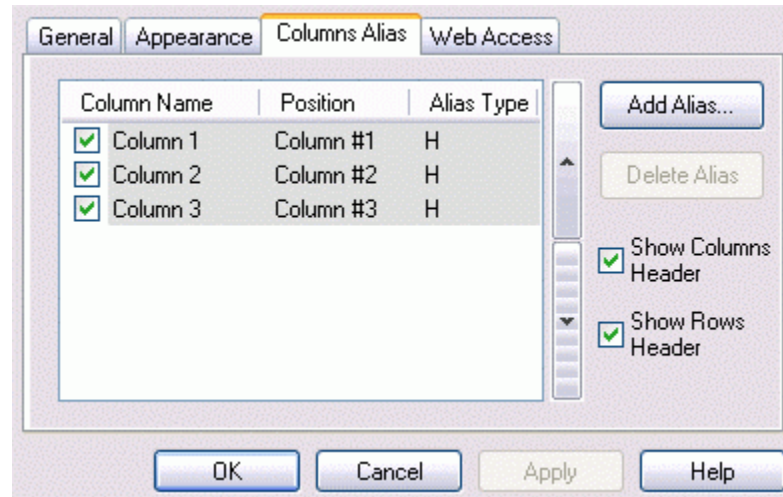
4. When you enter the display into runtime mode, you will see the associated data items displayed in the Data Mining Grid with the substituted column aliases, as shown in the figure below. You can change the column width by moving the column dividers.

<i>Data Mining Control</i>			
	Column 1	Column 2	Column 3
1	Jack	Hughman	10
2	Smith	Jones	11
3	Peter	Williams	13
4	20	05	17

Column Aliases Displayed in Data Mining Grid

Hiding Columns

Suppose that you have configured a database connection that reads a data set with multiple columns, and you have created a column alias for each column of data that is read from the database, as shown in the figure below. You can hide each columns from the display by simply unchecking the check box next to the column.



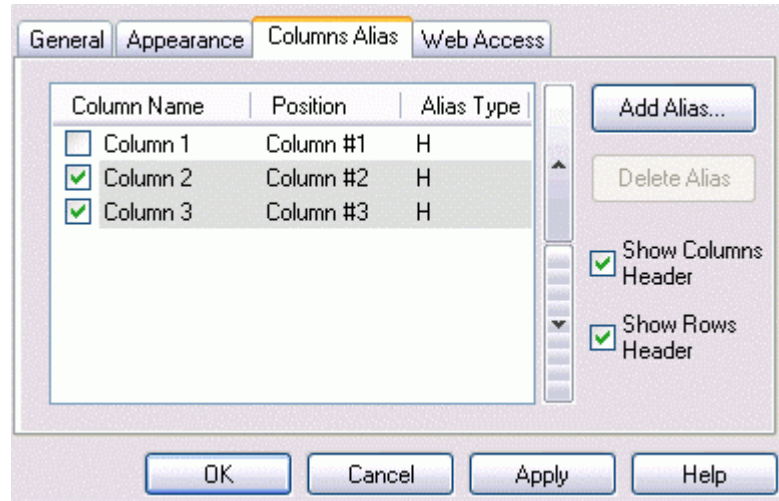
Multiple Column Aliases

For example, in runtime mode the grid display appears as shown in the figure below. As you can see, the column alias names are displayed at the top of the grid. Column 1 contains some data (e.g. abbreviations, cryptic characters, etc.) that may not make sense to some users and therefore should be removed from the display. Thus, we will hide Column 1.

<i>Data Mining Control</i>			
	Column 1	Column 2	Column 3
1	Jack	Hughman	10
2	Smith	Jones	11
3	Peter	Williams	13
4	20	05	17

Grid Display With Multiple Columns

Go back to the **Columns Alias** tab and uncheck the check box next to Column 1, as shown in the figure below. Click **OK**.



Hiding a Column

When you start runtime mode, you can see that Column 1 is now hidden, and the first column is now Column 2, as shown in the figure below.

Note: You can also hide the column headers and row headers. To hide the column names in the Grid, uncheck the **Show Columns Header** check box on the **Columns Alias** tab. To hide the row names (or numbers) in the Grid, uncheck the **Show Rows Header** check box on the **Columns Alias** tab.

Data Mining Control		
	Column 2	Column 3
1	Hughman	10
2	Jones	11
3	Williams	13
4	05	17

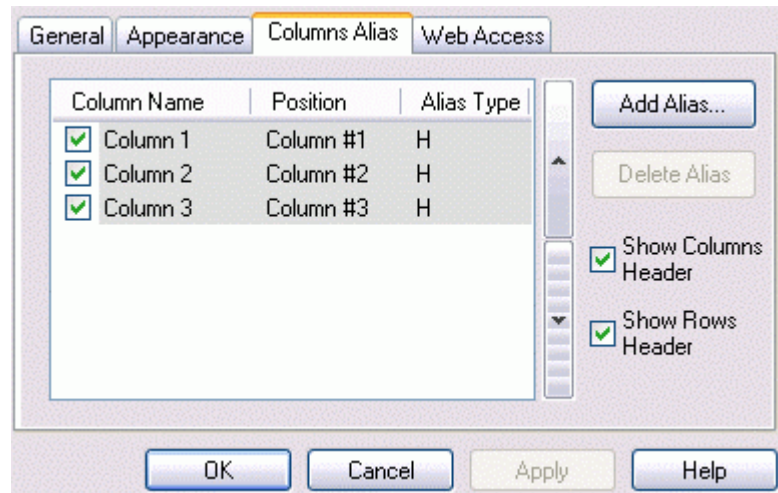
Grid Display With Hidden Columns

Creating Smart Aliases for Columns

In the **Columns Alias** tab, you can also create "smart" aliases that allow you to substitute some text or an image for a value or a string in the selected column. The basic concept of the smart alias configuration is to create a statement that defines the conditions under which a specified value or string will be replaced by a specified image or a text string in the data grid. For example:

1. Suppose you have already configured three column header aliases in the **Columns Alias** tab for a connected data source, as shown in the figure

below. As you can see, under the **Alias Type** column, each column header alias is indicated by an "H."



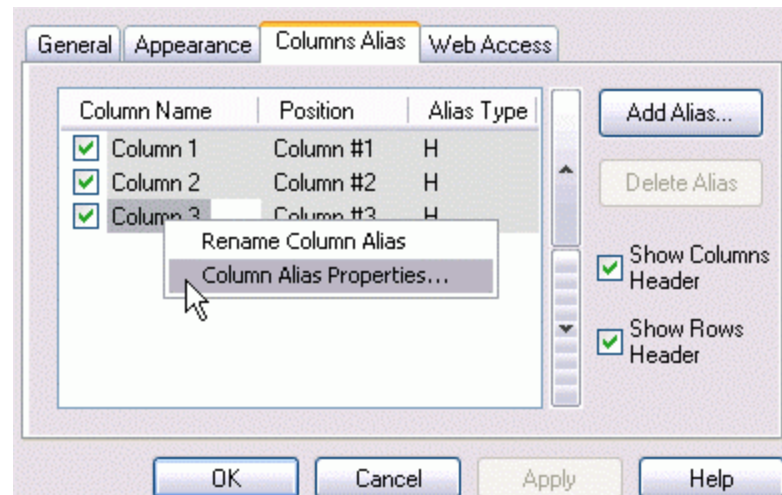
Column Aliases

2. In runtime mode, the Data Mining ActiveX grid appears with the column header aliases, as shown in the figure below. As you can see, column 3 lists several numerical values for this data source (e.g. 10, 11, 13, and 17). Let's suppose that you want to apply a smart alias to column 3 so that every time the value "10" occurs in column 3, that value will be replaced by an image of your choice.

<i>Data Mining Control</i>			
	Column 1	Column 2	Column 3
1	Jack	Hughman	10
2	Smith	Jones	11
3	Peter	Williams	13
4	20	05	17

Column Header Aliases in Runtime Mode

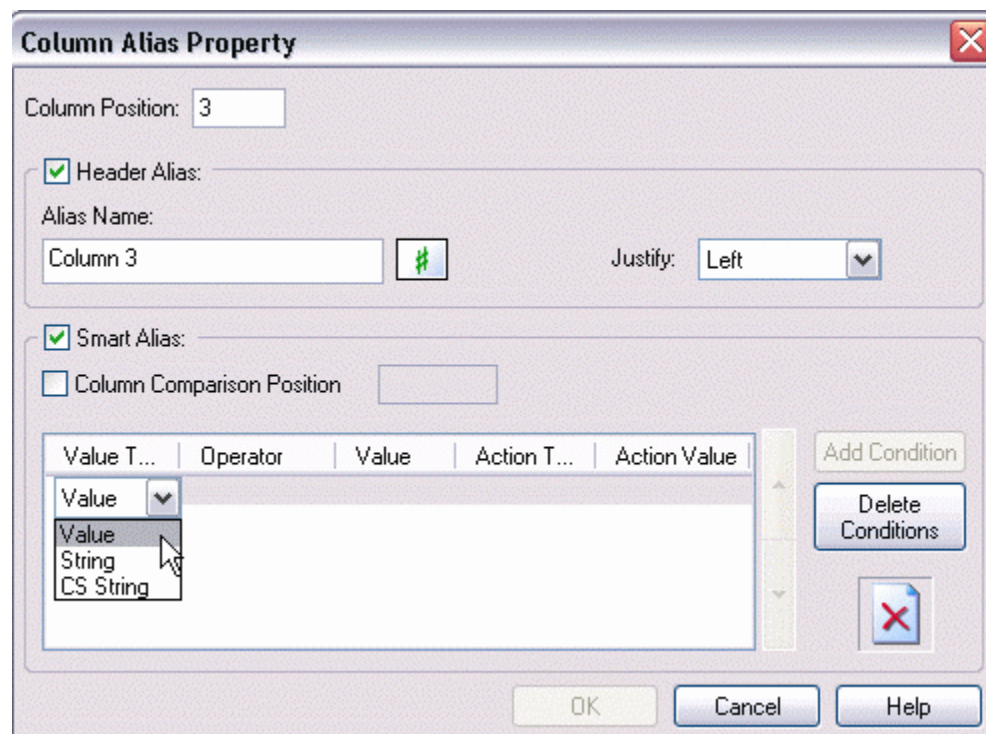
3. To do this, we will need to edit the alias properties for column 3. In the **Columns Alias** tab, right-click on **Column 3** and select **Column Alias Properties**, as shown in the figure below.



Opening the Alias Properties for Column 3

4. This opens the **Column Alias Property** dialog box, as shown in the figure below. Check the **Smart Alias** check box, and then click the **Add Condition** button. Click on the **Value Type** column and select **Value** from the drop-down list.

Note: Under **Value Type**, you can also select **String** or **Case-Sensitive (CS) String**.



Creating a Smart Alias for the Column

5. Now you can create a statement that tells the Data Mining ActiveX under which conditions to alias the value "10" in column 3. In the example shown below, the "equals" symbol is selected as the **Operator**, the **Value** is

specified as "10" and the specified **Action Type** is "Image." For the **Action Value**, click the red arrow button to browse for an image to use as a replacement for the value.

Note: For the value **Operator**, you can also select symbols for "greater than," "less than," "greater than or equal to," "less than or equal to," or "greater than and less than." In the **Value** field, you must type in a value or a string, depending on which value type you are using. Under **Action Type**, you can select image or text. If you select text for the **Action Type**, you must type the text for the **Action Value**.

For the string and case-sensitive string **Operator**, you can also select symbols for "greater than and less than," "prefix," "suffix," "contained," or "not contained." In the **Value** field, you must type in a value or a string, depending on which value type you are using. Under **Action Type**, you can select image or text. If you select text for the **Action Type**, you must type the text for the **Action Value**.

Column Alias Property

Column Position: 3

☒ Header Alias:

Alias Name: Column 3 Justify: Left

☒ Smart Alias:

☐ Column Comparison Position

Value Type	Operator	Value	Action Type	Action Value
Value	=	10	Image	<Image>

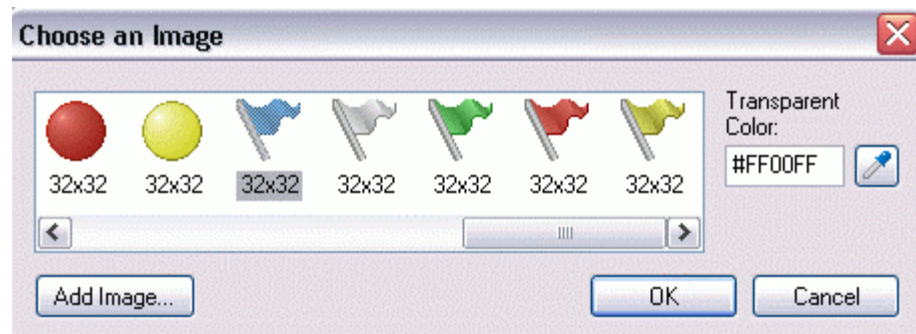
Add Condition
Delete Conditions

OK Cancel Help

Creating a Conditional Statement for the Alias

- You can choose one of the provided images, or you can click the **Add Image** button to select your own image file. (There are two sets of images, 16x16 and 32x32, installed by default inside the installation directory, usually "Program Files/ICONICS/GENESIS32/Bin/Icons".) For this example, we will choose the blue flag, as shown in the figure below.

Note: The maximum resolution for images is 32 x 32 pixels. You can also specify a **Transparent Color** (represented as a hexadecimal value), which is the color that will be used for the background of the cell(s) in which the selected image will appear. Clicking the eyedropper button allows you to choose a color.



Choosing an Image File for the Action Value

7. The selected image is displayed in the smart alias configuration, as shown in the figure below. Thus, the completed alias condition statement for our example reads as follows: "If a value in column 3 equals 10, then the value will be replaced by the blue flag image." Click **OK**.

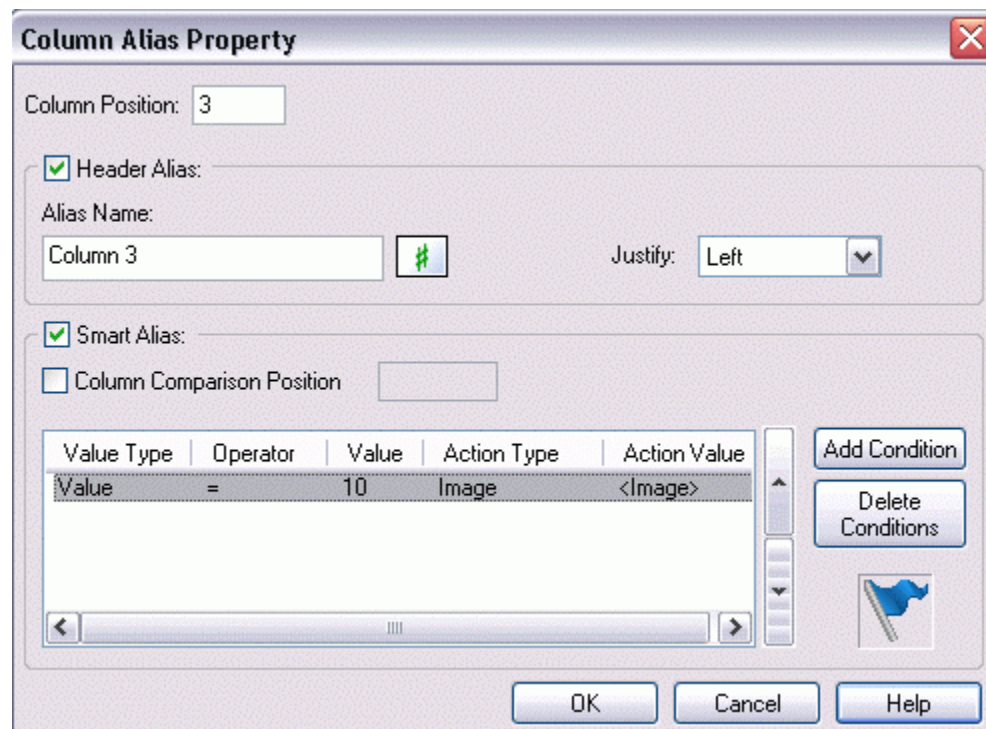
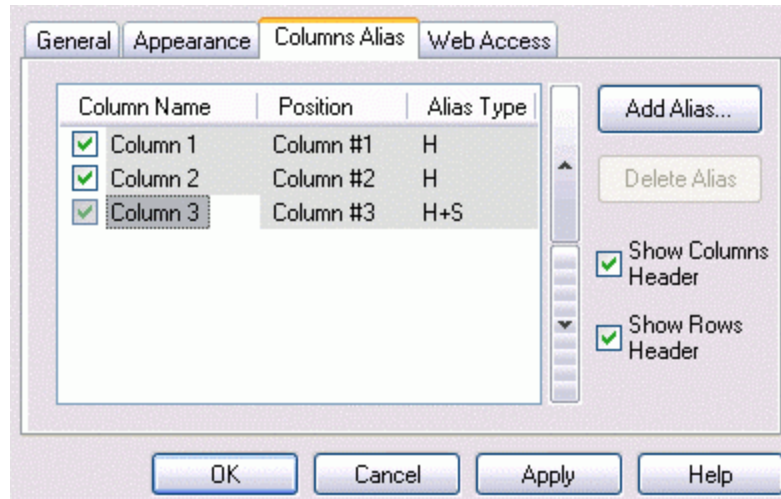


Image Selected for Action Value

8. Now the smart alias for column 3 is indicated in the **Columns Alias** tab in the **Alias Type** column, which says "H+S." This means that column 3 now contains both a header alias and a smart alias.



Smart Alias Indicated in Columns Alias Tab

9. In runtime mode, you can see how the smart alias is resolved in column 3, as shown in the figure below. As you can see, the value 10 in column 3 has been replaced by the blue flag image in the Data Mining ActiveX grid.

<i>Data Mining Control</i>			
	Column 1	Column 2	Column 3
1	Jack	Hughman	
2	Smith	Jones	11
3	Peter	Williams	13
4	20	05	17

Smart Alias Resolved in Runtime Mode

Creating a Column Comparison in a Smart Alias

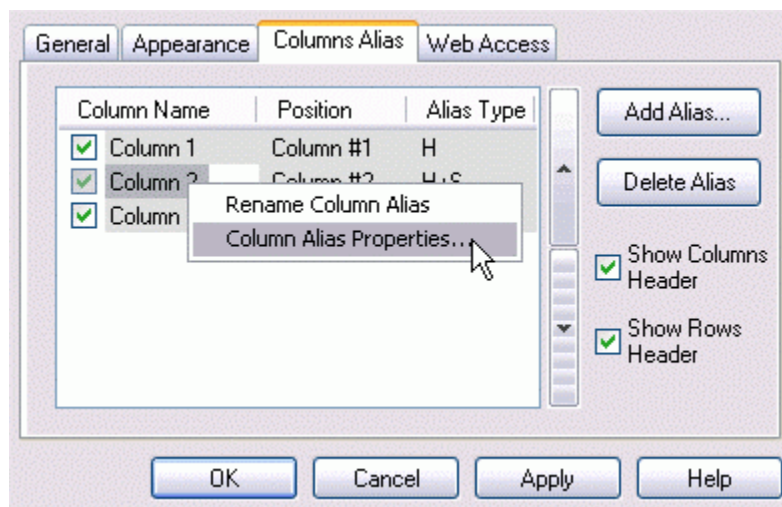
In the **Columns Alias** tab, you can also set up a "smart" alias that will compare the values of two different columns. For example:

1. Suppose you have already configured a data source connection with multiple columns. In runtime mode, the Data Mining ActiveX grid appears as shown in the figure below. For this example, we will set up a smart alias for Column 2 that compares the values with the values in Column 3.

Data Mining Control				
	Column 1	Column 2	Column 3	Co
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sale
2	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	(
3	ANTON	Antonio Moreno Taquería	Antonio Moreno	(
4	AROUT	Around the Horn	Thomas Hardy	Sale
5	BERGS	Berglunds snabbköp	Christina Berglund	Ord
6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sale
7	BLONP	Blondel père et fils	Frédérique Citeaux	Marl
8	BOLID	Bólido Comidas preparadas	Martín Sommer	(
9	BONAP	Bon app'	Laurence Lebihan	(

Column Header Aliases in Runtime Mode

- To do this, we will edit the alias properties for Column 2. In the **Columns Alias** tab, right-click on **Column 2** and select **Column Alias Properties**, as shown in the figure below.



Opening the Alias Properties for Column 2

- This opens the **Column Alias Property** dialog box, as shown in the figure below. Check the **Smart Alias** check box, and then click the **Add Condition** button. Also check the **Column Comparison Position** check box and type 3 in the edit box. This means that the smart alias value type specified for Column 2 will be compared with the values in Column 3. Click on the **Value Type** column and select **String** from the drop-down list. Now you can create a statement that compares the string with the values in Column 3. In the example shown below, "contained" is selected as the **Operator**, the **Value** is automatically specified as Column 3, and the specified **Action Type** is "Image." A green smiley face image is already selected to use as a replacement for the values. Thus, the completed alias condition statement for our example reads as follows: "If a string in Column 2 is also contained by Column 3, then the string will be replaced by the green smiley face image." Click **OK**.

Column Alias Property

Column Position: 2

☒ Header Alias:

Alias Name: Column 2  Justify: Left

☒ Smart Alias:

☒ Column Comparison Position 3

Value T...	Operator	Value	Action T...	Action Value
String	Contained	Image	Image	<Image>

Add Condition
Delete Conditions


OK Cancel Help

Creating a Smart Alias for the Column

- In runtime mode, you can see how the column comparison alias compares the values in each row of the two columns and then resolves the alias based on the conditional statement, as shown in the figure below. As you can see, for each row in Column 2 that contains a string from the same row in Column 3, the string values in Column 2 are replaced by the green smiley face image, as defined in the smart alias conditional statement. For example, in Row 2 both Column 2 and Column 3 contain the string "Ana Trujillo," and in Row 3 both Column 2 and Column 3 contain the string "Antonio Moreno."

Data Mining Control				
	Column 1	Column 2	Column 3	Co
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sale
2	ANATR		Ana Trujillo	(
3	ANTON		Antonio Moreno	(
4	AROUT	Around the Horn	Thomas Hardy	Sale
5	BERGS	Berglunds snabbköp	Christina Berglund	Ord
6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sale

Data Mining Control				
	Column 1	Column 2	Column 3	Co
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sale
2	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	(
3	ANTON	Antonio Moreno Taquería	Antonio Moreno	(
4	AROUT	Around the Horn	Thomas Hardy	Sale
5	BERGS	Berglunds snabbköp	Christina Berglund	Ord
6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sale
7	BLONP	Blondel père et fils	Frédérique Citeaux	Marl
8	BOLID	Bólido Comidas preparadas	Martín Sommer	(
9	BONAP	Bon app'	Laurence Lebihan	(

Column Comparison Resolved in Runtime Mode

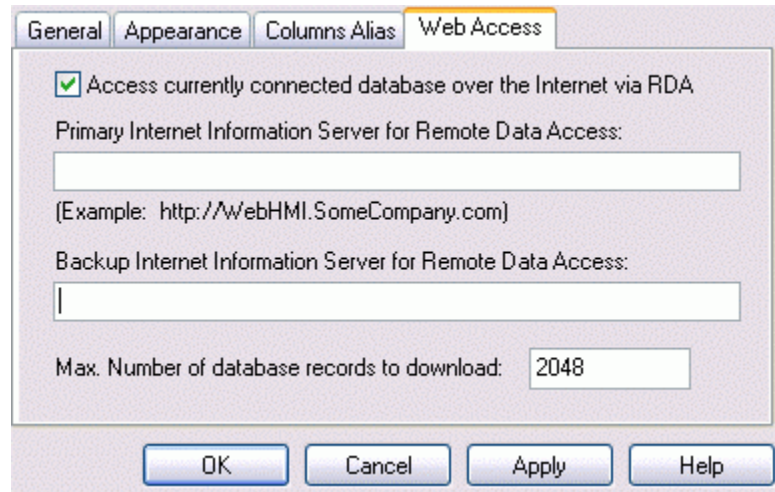
Web Access

The **Web Access** tab of the **Data Mining ActiveX Properties** dialog box, shown below, allows you to access the currently connected database over the Internet via a **Remote Data Service (RDS)**. The RDS, which is hosted by an Internet Information Server (IIS), enables the downloading of data from the connected database to a client over the Internet. Simply specify the URL name or IP address of the primary IIS Web server in the **Primary Internet Information Server for RDA** field, as shown in the figure below. The **Remote Recordset Page Size** field specifies the size of the recordset portion in the database records.

Note: You also may designate a redundant IIS Web server by specifying the URL name or IP address of the redundant IIS Web server in the **Backup Internet Information Server for RDA** field, as shown in the figure below.

For more information about Remote Data Access, please refer to the AlarmWorX32 Viewer ActiveX help documentation.

Note: The **Web Access** tab only applies when you are connecting via an ADO database connection. If you are using a data mining tag, you need only supply the node name or IP address of the server in front of the tag name.



The screenshot shows the 'Web Access' tab of the 'Data Mining ActiveX Properties' dialog box. The 'General' tab is selected, and the 'Web Access' tab is active. The 'Access currently connected database over the Internet via RDA' checkbox is checked. Below this, there are two text input fields: 'Primary Internet Information Server for Remote Data Access:' and 'Backup Internet Information Server for Remote Data Access:'. An example URL 'http://WebHMI.SomeCompany.com' is shown below the first field. At the bottom, there is a 'Max. Number of database records to download:' field with the value '2048'. The dialog box has 'OK', 'Cancel', 'Apply', and 'Help' buttons at the bottom.

Data Mining ActiveX Properties: Web Access Tab

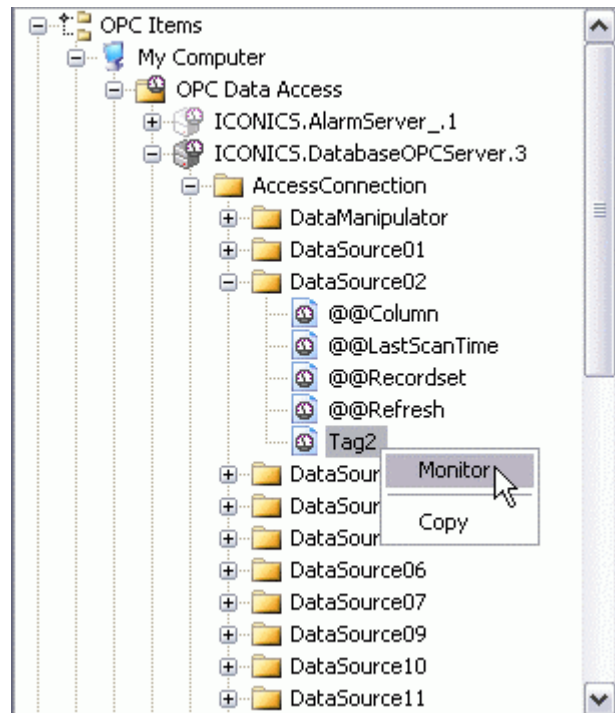
Viewing Data in the Runtime Grid

Once you have established your data source connections, you are ready to view the data in the Data Mining ActiveX Grid.

Testing Your Data Tags

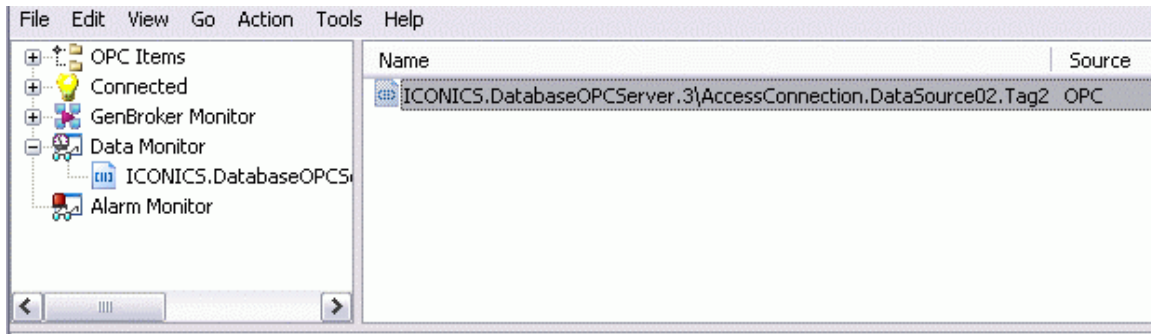
Before you try to connect to data sources and view your data tags, it is highly recommended that you verify if the tags present inside the display are well configured from the server side. To do that you have to check the quality of your data tags using OPC DataSpy:

1. Start OPC DataSpy from the Windows **Start** menu by selecting **Programs > ICONCS Tools > OPC DataSpy**.
2. In OPC DataSpy, expand the tree control and browse to your data tag. Right-click on the data tag(s) and select **Monitor** from the pop-up menu, as shown in the figure below.



Browsing for Data Tags in OPC DataSpy

3. The data tag(s) now appear in under the Data Monitor tree control, as shown in the figure below. The tag quality is indicated in the right-hand pane. If the tag quality is good, then the tag is ready for data mining. If the tag quality is bad, go back to the Database Mining Configurator and check your data source and data item configurations.



Data Tag Quality Indicated in DataSpy Data Monitor

Viewing One-Dimensional Arrays

The figure below shows a one-dimensional array in runtime mode. This example (refer to the "Connecting to a One-Dimensional Data Mining Data Source" section above for details) displays the entire rowset ANTON in the **Customer ID** column in the connected Northwind.mdb database, as defined by the connected **AccessConnection.DataSource02.Tag2** data tag. You can change the column width by moving the column dividers.

Note: The array can be a horizontal array (i.e. view an entire rowset), or it can be a vertical array (i.e. it shows the values in a column). No matter what array you want to view, in the grid it is always shown as a column. Thus, even though the example shown in the figure below displays a rowset, the data in the row are displayed in the grid as a column.

Data Mining Control	
	CustomerID
1	ANTON
2	Antonio Moreno Taquería
3	Antonio Moreno
4	Owner
5	Mataderos 2312
6	México D.F.
7	<Empty>
8	05023
9	Mexico
10	(5) 555-3932
11	<Empty>

Viewing Data Items in Runtime Mode: One-dimensional Array

Viewing Multidimensional Arrays

The figure below shows a multidimensional array in runtime mode. This example (refer to the "Connecting to a Multidimensional Data Mining Data Source" section above for details) displays columns of data (e.g. Customer ID, Company Name, Contact Name, Contact Title, etc.) that match the columns in the connected Northwind.mdb database, as defined by the connected

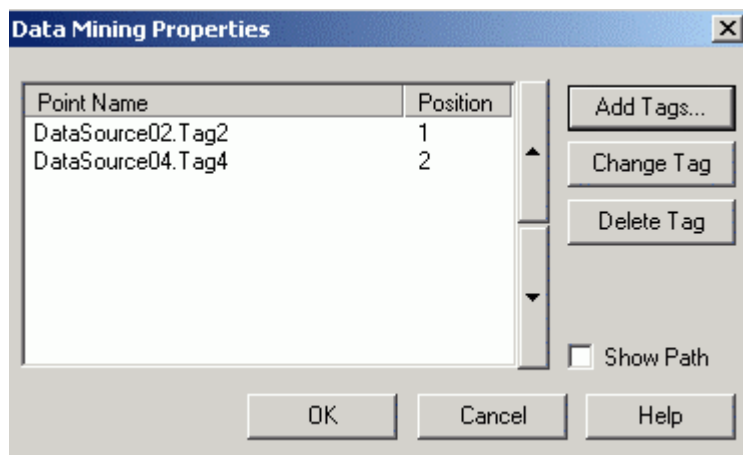
AccessConnection.DataSource04.Tag4 data tag. You can change the column width by moving the column dividers.

Data Mining Control							
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region
1	ALFKI	Alfreds Futterkiss	Maria Anders	Sales Repres	Obere Str.	Berlin	<Empty>
2	ANATR	Ana Trujillo Emp	Ana Trujillo	Owner	Avda. de l	México	<Empty>
3	ANTON	Antonio Moreno	Antonio Moren	Owner	Matadero	México	<Empty>
4	AROUT	Around the Hor	Thomas Hardy	Sales Repres	120 Hano	Londo	<Empty>
5	BERGS	Berglunds snab	Christina Bergl	Order Admini	Berguvsvä	Luleå	<Empty>
6	BLAUS	Blauer See Deli	Hanna Moos	Sales Repres	Forsterstr.	Mannh	<Empty>
7	BLONP	Blondel père et f	Frédérique Cit	Marketing Ma	24, place	Strasb	<Empty>
8	BOLID	Bólido Comidas	Martín Somme	Owner	C/ Araquil,	Madrid	<Empty>
9	BONAP	Bon app'	Laurence Lebi	Owner	12, rue de	Marseil	<Empty>
10	BOTTM	Bottom-Dollar M	Elizabeth Linc	Accounting	23 Tsawa	Tsawa	BC
11	BSBEV	B's Beverages	Victoria Ashwo	Sales Repres	Fauntleroy	Londo	<Empty>
12	CACTU	Cactus Comidas	Patricio Simps	Sales Agent	Cerrito 33	Bueno	<Empty>
13	CENTC	Centro comercia	Francisco Cha	Marketing Ma	Sierras de	México	<Empty>
14	CHOPS	Chop-suey Chin	Yang Wang	Owner	Hauptstr.	Bern	<Empty>
15	COMMI	Comércio Mineir	Pedro Afonso	Sales Associ	Av. dos Lu	São P	SP
16	CONSH	Consolidated H	Elizabeth Brow	Sales Repres	Berkeley	Londo	<Empty>
17	DRACD	Drachkötter Del	Gunnar O'Hish	Order Admini	Waldsee	Ascher	<Empty>

Viewing Data Items in Runtime Mode: Multidimensional Array

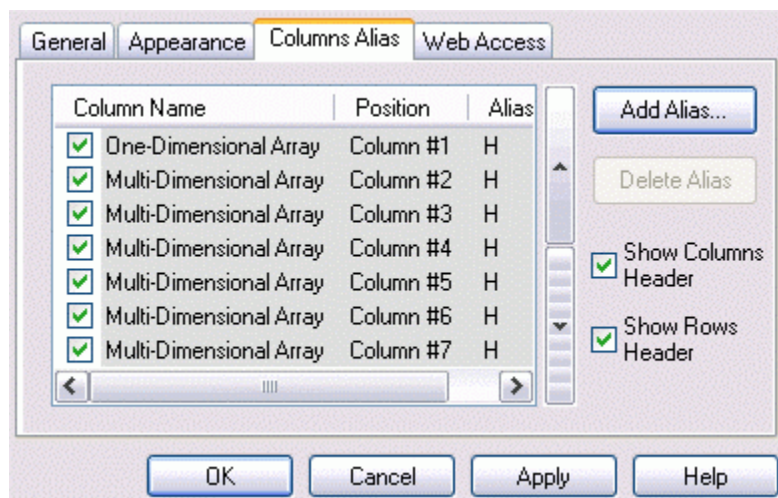
Viewing Combined Arrays

It is possible to view data from a one-dimensional array and data from a multidimensional array in the same grid display. For example, the data tags selected in the Data Mining Properties below contain two different types of arrays: Tag2 is one-dimensional, and Tag4 is multidimensional. Tag2 will yield only one column of data in the grid display, while Tag4 will produce many columns. When you enter the display into runtime, the data from both arrays will be displayed side-by-side in the Data Mining ActiveX Grid.



Data-Mining Properties: Combining Data Sources

Thus, it is recommended that you use some column aliases to distinguish between the two different sets of data. In the example configuration shown in the figure below, Column 1 represents the single column from Tag2, and the other columns (2-6) represent the columns from Tag4.



Column Names for Combined Arrays

When you enter the display into runtime mode, you can clearly distinguish between the two sets of data produced by each of the data tags, as shown in the figure below.

Data Mining Control				
	One-Dimensional Array	Multi-Dimensional Array	Multi-Dimensional Array	Multi-Dimensional Array
1	ANTON	ALFKI	Alfreds Futterkiste	Maria Anders
2	Antonio Moreno Taquería	ANATR	Ana Trujillo Emparedados y dulces	Ana Trujillo
3	Antonio Moreno	ANTON	Antonio Moreno Taquería	Antonio Moreno
4	Owner	AROUT	Around the Horn	Thomas Hardy
5	Mataderos 2312	BERGS	Berglunds snabbköp	Christina Berglund
6	México D.F.	BLAUS	Blauer See Delikatessen	Hanna Moos
7	<Empty>	BLONP	Blondel père et fils	Frédérique Citeau
8	05023	BOLID	Bólido Comidas preparadas	Martín Sommer
9	Mexico	BONAP	Bon app'	Laurence Leblond
10	(5) 555-3932	BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln
11	<Empty>	BSBEV	B's Beverages	Victoria Ashworth
12	<Empty>	CACTU	Cactus Comidas para llevar	Patricio Simpson
13	<Empty>	CENTC	Centro comercial Moctezuma	Francisco Chang
14	<Empty>	CHOPS	Chop-suey Chinese	Yang Wang
15	<Empty>	COMMI	Comércio Mineiro	Pedro Afonso
16	<Empty>	CONSH	Consolidated Holdings	Elizabeth Brown
17	<Empty>	DRACD	Drachen Delikatessen	Gerhard Koch

Viewing Combined Arrays in the Grid Display

Refreshing the Data Grid

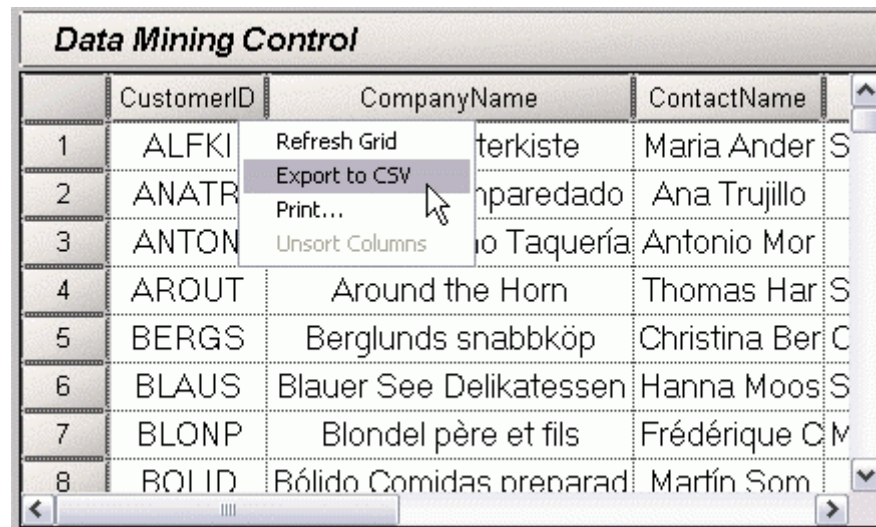
The Data Mining ActiveX automatically updates data from the connected data source (ADO data connections only) at the frequency (in seconds) you specified in the **Auto Update** field in the Data Mining ActiveX **General** tab. To manually update the data in the Grid, right-click on the data during runtime mode and select **Refresh Grid** from the pop-up menu, as shown in the figure below.

Data Mining Control				
	CustomerID	CompanyName	ContactName	
1	ALFKI	Alfreds Futterkiste	Maria Anders	S
2	ANATR	Ana Trujillo Emparedados y dulces	Ana Trujillo	
3	ANTON	Antonio Moreno Taquería	Antonio Moreno	
4	AROUT	Around the Horn	Thomas Hardy	S
5	BERGS	Berglunds snabbköp	Christina Berglund	C
6	BLAUS	Blauer See Delikatessen	Hanna Moos	S
7	BLONP	Blondel père et fils	Frédérique Citeau	M
8	BOLID	Bólido Comidas preparadas	Martín Sommer	

Refreshing Data in Grid

Exporting and Printing Data

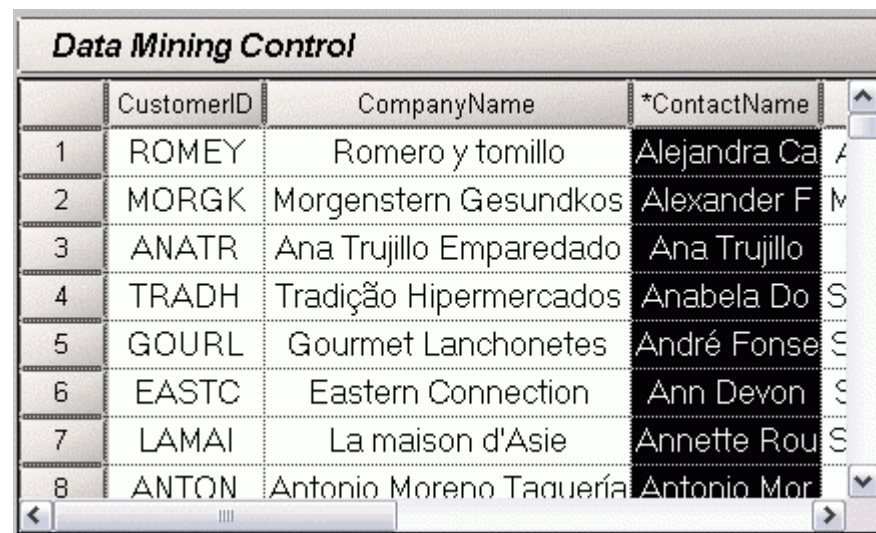
The Data Mining ActiveX allows you to export or print data during runtime mode. During runtime mode, right-click on the grid and select **Export to CSV** or **Print**, as shown in the figure below. If you select **Export to CSV**, you can specify the name of the CSV file and save the data. To view the exported data, open the CSV file in Microsoft Excel.



Exporting and Printing Data

Sorting Data Columns

The Data Mining ActiveX allows you to sort columns in runtime mode. Simply double-click on a column header to sort the column in ascending or descending order. An asterisk appears in the header of the sorted column, as shown in the figure below.



Sorting Columns

Data Mining ActiveX OLE Automation Reference

This reference describes the OLE Automation features available in the Data Mining ActiveX.

Automation Interfaces

The Data Mining ActiveX provides a COM interface that allows automation interfaces run from within the ActiveX container to manipulate the Data Mining ActiveX control as it is running. The interface is available to all programming languages that support COM, including Visual Basic (VB), Visual Basic for Applications (VBA), and Microsoft Visual C++.

To access the Automation interface from VB and VBA, the Data Mining ActiveX must be made available by choosing **Project > Components** from the main menu in the VB or VBA development environment and selecting **ICONICS DBGrid** in the list of available components.

Control Properties

AutoFitColumns

Type: Boolean

Description

If true when in runtime mode, columns will automatically adjust their width to fit the contents.

AutoSelectRow

Type: Boolean

Description

If true, when the user clicks on a cell the whole row will be highlighted.

AutoUpdateInterval

Type: Short

Description

Sets/gets the number of seconds between automatic updates.

Available only for ADO connections.

Example

ThisDocument.DBMining1.AutoUpdateInterval=60

BackColor

Type: OLE_COLOR

Description

Sets/gets the background color of the column.

Example

```
Dim column As OGridColumnWrapper  
Set column = ThisDisplay.DBMining1.GetColumn(1)  
column.ModifyBackColor = True  
column.BackColor = RGB(0, 255, 0)
```

The above example will change the background color for column 1 to green.

BlinkOnUpdate

Type: Boolean

Description

Sets/gets whether or not the Database Mining ActiveX will blink on update.

Example

```
ThisDocument.DBMining1.BlinkOnUpdate=True
```

BorderType

Type: String

Description

Sets/gets control's border appearance. Can be one of the following:

"0" - none

"1" - flat border

"2" - 3D border

Example

Sets 3D look for the control's border.

```
ThisDocument.DBMining1.BorderType="2"
```

CellHAlign

Type: Short

Description

Sets/gets the default cell alignment.

"0" – Left Alignment

"1" – Center Alignment

"2" – Right Alignment

Example

```
ThisDocument.DBMining1.CellHAlign="2"
```

ColumnHAlign**Type: Short****Description**

Sets/gets the column header alignment.

"0" – Left Alignment

"1" – Center Alignment

"2" – Right Alignment

Example

```
ThisDocument.DBMining1. ColumnHAlign ="1"
```

ConnectionString**Type: String****Description**

Sets/gets ADO connection string used by the Database Mining ActiveX control to access data.

Example

```
Dim strConnString as String
strConnString="Provider=Microsoft.Jet.OLEDB.4.0; _
              Data Source=c:\alarms\AWXLog32.mdb; _
              User ID=Admin; Password=SYSTEM"
ThisDocument.DBMining1. ConnectionString = strConnString
```

DataSourceType**Type: Short****Description**

Sets/gets the Database Mining ActiveX connection type.

"0" - ADO

"1" – Database Mining

DBMiningTitle**Type: String****Description**

Sets/gets the title shown in the window title bar of the Database Mining ActiveX control during runtime mode.

DisableResize

Type: Boolean**Description**

If true, users will not be able to adjust row or column sizes.

DisableMultiSelect**Type: Boolean****Description**

If true, users will not be able to select multiple rows, columns, or cells.

DisableResize**Type: Boolean****Description**

Sets/gets Boolean value that controls the automatic resize of the grid columns.

DisplayGridTitle**Type: Boolean****Description**

Sets/gets Boolean value that controls the visibility of the grid title bar.

Example

ThisDocument.DBMining1.DisplayGridTitle = TRUE

EnableAutoUpdate**Type: Boolean****Description**

Enables/disables automatic updates for ADO connections.

Example

ThisDocument.DBMining1.EnableAutoUpdate=True

EnableWebAccess**Type: Boolean****Description**

Enables/disables Database Mining ActiveX Web access for ADO connections.

Example

ThisDocument.DBMining1.EnableWebAccess=True

FileName**Type: BSTR**

Description

Sets/gets the name of the Database Mining ActiveX configuration file.

Example

```
ThisDocument.DBMining1.FileName="C:\DBMiningConfig.dbm"
```

Font

Type: StdFont

Description

Returns a reference to the control's default font.

Example

To set control's default font to Arial, 14 pt., use the following statements:

With ThisDocument.DBMining1.Font

.Name="Arial"

.Size=14

End With

ForeColor

Type: OLE_COLOR

Description

Sets/gets foreground color (cell's font color) property of the Database Mining ActiveX.

Example

```
ThisDocument.DBMining1. ForeColor=RGB(0,0,0)
```

GridBackColor

Type: OLE_COLOR

Description

Sets/gets color value used to paint the gaps between rightmost column and right edge of the grid, last row and bottom edge of the grid, and row headers column (the 'back area' of the grid).

Example

```
ThisDocument.DBMining1. GridBackColor =RGB(0,127,0)
```

GridDefRowHeight

Type: Integer

Description

Sets/gets the default row height for data grid.

Example

ThisDocument.DBMining1. GridDefRowHeight=16

GridHeaderShade

Type: Short

Description

Sets/gets the cell effect for the grid's title bar header.

"0" = Soft Bump

"1" = Diagonal Shade

"2" = Horizontal Shade

"3" = Vertical Shade

"4" = Horizontal Bump

"5" = Vertical Bump

"6" = Metal

GridHorizLinesColor and GridVertLinesColor

Type: OLE_COLOR

Description

Sets/gets color for horizontal or vertical grid lines.

Example

ThisDocument.DBMining1. GridHorizLinesColor =RGB(127,127,127)

ThisDocument.DBMining1. GridVertLinesColor = RGB(127,127,127)

GridHorizLineStyle and GridVertLineStyle

Type: Integer

Description

Sets/gets style for horizontal or vertical grid lines. Can be one of the following:

0 - no line

1 - solid line

2 - dot line

3 - dash line

4 - dashdot line

5 - dashdotdot line

Example

ThisDocument.DBMining1. GridHorizLineStyle =0

ThisDocument.DBMining1. GridVertLineStyle = 2

GridHorizLinesWidth and GridVertLinesWidth

Type: Integer**Description**

Sets/gets the widths for the horizontal or vertical grid lines when corresponding grid line style (property GridHorizLinesStyle or GridVertLinesStyle) is set to 1 - solid line. For all other line styles, these properties must be set to 1.

Example

Sets both horizontal and vertical grid lines to solid line style, 2 pixels wide.

```
ThisDocument.DBMining1. GridHorizLinesStyle = 1
```

```
ThisDocument.DBMining1. GridVertLinesStyle = 1
```

```
ThisDocument.DBMining1. GridHorizLinesWidth = 2
```

```
ThisDocument.DBMining1. GridVertLinesWidth = 2
```

GridShade**Type: Short****Description**

Sets/gets the cell effect for the grid column's headers.

"0" = Soft Bump

"1" = Diagonal Shade

"2" = Horizontal Shade

"3" = Vertical Shade

"4" = Horizontal Bump

"5" = Vertical Bump

"6" = Metal

GridTitleBackColor**Type: OLE_COLOR****Description**

Sets/gets the background color for the grid's title bar header.

Example

Sets title bar background color to dark blue.

```
ThisDocument.DBMining1. GridTitleBackColor = GRB(0,0,127)
```

GridTitleFont**Type: StdFont****Description**

Returns a reference to the default font for the grid's title bar header.

Example

To set control's default font to Arial, 14 pt., use the following statements:

```
With ThisDocument.DBMining1.Font  
.Name="Arial"  
.Size=14  
End With  
ThisDocument.DBMining1. GridTitleFontColor =GRB(0,0,127)
```

GridTitleHAlign**Type: Short****Description**

Sets/gets the grid's title bar header alignment.

"0" – Left Alignment

"1" – Center Alignment

"2" – Right Alignment

Example

```
ThisDocument.DBMining1. GridTitleHAlign ="1"
```

GridTitleTextColor**Type: OLE_COLOR****Description**

Sets/gets the text color for the grid's title bar header.

Example

Sets grid's title bar text color to yellow.

```
ThisDocument.DBMining1. GridTitleBackColor =GRB(255,255,0)
```

HeaderColor**Type: OLE_COLOR****Description**

Sets/gets the background color for grid's column headers.

Example

Sets color of grid's column header to dark green.

```
ThisDocument.DBMining1. GridTitleBackColor =GRB(0,127,0)
```

HeaderFont**Type: StdFont**

Description

Returns a reference to the default font for the grid's column headers.

HeaderTxtColor

Type: OLE_COLOR

Description

Sets/gets the text color for grid's column headers.

Example

Sets grid's title bar text color to yellow.

ThisDocument.DBMining1.GridTitleBackColor =RGB(255,255,0)

HideGridPopupMenu

Type: Boolean

Description

Enables/disables Database Mining ActiveX runtime popup menu.

InternalGridEdges

Type: Short

Description

Sets the amount of space between the ActiveX borders and the grid borders.

IsConnected

Type: Boolean

Description

This is read-only property, indicating that control is connected to the server.

Example

If ThisDocument.DBMining1.IsConnected Then

 ''' do something here '''

End If

MaxRecords

Type: Long

Description

Sets/gets the maximum number of records that the Database Mining ActiveX will download using a Web access RDS connection.

Example

ThisDocument.DBMining1.MaxRecords=1024

ParseHTMLTag**Type: Boolean****Description**

Enables/disables the HTML parsing inside the grid's cells.

If inside a grid cell there is a string with the format:

```
<a href="link.html">HyperText</a>
```

The Grid will show the word "HyperText" as clickable text, and when the user clicks on it an Internet Explorer window will popup opening the page "link.html".

QueryText**Type: String****Description**

Sets/gets the SQL statement used with an ADO connection to populate the data grid in runtime mode.

Example

```
Dim strQry as String
```

```
strQry=ThisDocument.DBMining1.QueryText
```

```
MsgBox strQry
```

RemoteDataServer**Type: BSTR****Description**

This is the Internet Information Server used with the Web access RDS connections.

Example

```
ThisDocument.DBMining1.RemoteDataServer = "http://localhost"
```

RemoteDataServerBackup**Type: BSTR****Description**

This is the backup URL used with Web access RDS connections if the connection with the server specified on RemoteDataServer property fails.

Example

```
ThisDocument.DBMining1.RemoteDataServerBackup = "http://localhost"
```

ShowColsHeader and ShowRowsHeader

Type: Boolean

Description

Show/hide the rows/columns headers.

URLPathName

Type: BSTR

Description

Sets/gets the URL path for a configuration file over the Web. Must set URLPathUsed to "True" for this to take effect.

Example

ThisDocument.DBMining1.URLPathName=
"http://www.iconics.com/dbminingcfg.dbm"

URLPathUsed

Type: Boolean

Description

Enables/disables the use of a URL path.

Example

ThisDocument.DBMining1.URLPathUsed=True

UseLocalSettings;

Type: Boolean

Description

When the URL path is enabled, it allows you to use or not to use the local settings configuration file.

Example

ThisDocument.DBMining1.UseLocalSettings=True

Control Methods

AboutBox

Shows **About** dialog box.

Example

```
ThisDocument.DBMining1.AboutBox
```

Connect As Boolean

It forces the control to establish connection to database, using the information set in **ConnectionString** and **QueryText**. If control is already connected, the current connection will be closed and a new connection will be established. This method can be used in VBA scripts automatically to switch the Database Mining ActiveX between different connections.

Return Value

On success, returns TRUE; if for any reason connection cannot be established, returns FALSE.

Example

```
Dim strConnString as String
strConnString="Provider=Microsoft.Jet.OLEDB.4.0; _
Data Source=c:\alarms\SomeOtherDatabase.mdb; _
User ID=Admin; Password=SYSTEM"
ThisDocument.DBMining1.ConnectionString = strConnString
ThisDocument.DBMining1.QueryText="Select * from [Table]"
ThisDocument.DBMining1.Connect
ThisDocument.DBMining1.Refresh
```

GetColumnsNumber As Integer

Returns the number of columns currently present in the grid.

Return Value

On success, returns the number of columns; if it fails or the grid doesn't have any column, returns 0.

GetGridCellValue(long ColNum,long RowNum) as VARIANT

Takes inside the grid buffer the current cell value with column number equal to *Column* and row number equal to *RowNum*.

Return Value

On success, returns the number of columns; if it fails or the grid does not have any columns, returns 0.

GetGridCellValue(ColNum As Long, RowNum As Long) as BSTR

Returns the value of the cell that has coordinates (ColNum,RowNum).

Return Value

On success, returns the value of the cell; if the cell coordinates are not valid or if the grid is not visible on the screen, it will fail returning an empty string.

GetRowsNumber As Integer

Returns the number of rows currently present in the grid.

Return Value

On success, returns the number of rows; if it fails or the grid does not have any columns, returns 0.

GetSelectedColIndex As Integer

Returns the coordinates of the currently selected column.

Return Value

On success, returns the coordinates of the currently selected column; if it fails, returns 0.

GetSelectedRange(VARIANT* TopCrd, VARIANT* LeftCrd, VARIANT* BottomCrd, VARIANT* RightCrd)

Returns the top, left, right and bottom coordinates of the currently selected area in the grid.

Return Value

On success, returns the coordinates of the currently selected area in the grid; if it fails there is no selected area, and returns all the coordinates with a negative value.

GetSelectedRowIndex As Integer

Returns the coordinates of the currently selected row.

Return Value

On success, returns the coordinates of the currently selected row; if it fails, returns 0.

GetUnselectedCellsInRange(VARIANT* UnselectedCellList)

Returns the coordinates of any unselected cells in the range returned by GetSelectedRange. The format of the cell list will be {x1, y1, x2, y2, ... }. The last element of the array will always be zero, and is not part of any coordinate.

Example:

'This example will display the coordinates one at a time in message boxes.

```
Sub getlist(o As GwxPick)
```

```
    Dim CoordList
```

```

        Dim Coordinate
        ThisDisplay.DBMining1.GetUnselectedCellsInRange CoordList
        For Each Coordinate In CoordList
            MsgBox (Coordinate)
        Next Coordinate
    End Sub

```

LoadConfigFile(BSTR newVal) as Boolean

It loads the configuration file specified in newVal.

Example

```

Dim ret As Boolean
ret = DBMining1.LoadConfigFile("C:\My Configs\cfg.dbm")

```

SaveConfigFile(BSTR newVal) as Boolean

It saves your current Database Mining ActiveX configuration to the file specified in newVal.

Example

```

Dim ret As Boolean
ret = DBMining1.SaveConfigFile("C:\My Configs\cfg.dbm")

```

LoadConfigURL(BSTR newVal) as Boolean

Loads the configuration file specified in the URL string newVal.

Example

```

Dim ret As Boolean
ret = DBMining1.LoadConfigURL("http://localhost/cfg.dbm")

```

ReplaceHost(BSTR OldHostName, BSTR NewHostName) as Long

It works over data source tags, data mining tags and path name attributes and replaces node name **substring** within URL path only. Returns 0 on success and HRESULT when something fails.

Example

'similar to ReplaceTag

'VBA example, works only in configure mode

'replaces host name in whole display

Dim Status As Long

```
Status = ThisDisplay.ReplaceHost("Host1", "Host2")
```

'if Status <> 0 then there was no replacement performed or an error occurs

```

If Status <> 0 Then
    MsgBox "No replacements"
Else
    MsgBox "Tags replaced"
End If

```

ReplaceHostEx(BSTR OldHostNameSubstring, BSTR NewHostNameSubstring, BOOL MatchCase, BOOL MatchWholeWord) as Long

It works over data source tags, data mining tags and path name attributes and replaces node name **substring** within URL path only, and supports case-sensitivity, wildcard strings and MatchWholeWord flag. Returns 0 on success and HRESULT when something fails.

Example

'similar to ReplaceTag

'VBA example, works only in configure mode

'replaces host name in whole display, regarding "case" and "whole words" options

Dim Status As Long

```
Status = ThisDisplay.ReplaceHostEx("Host1", "Host2", True, True)
```

'if Status <> 0 then there was no replacement performed or an error occurs

```

If Status <> 0 Then
    MsgBox "No replacements"
Else
    MsgBox "Tags replaced"
End If

```

SaveConfigFile(BSTR newVal) as Boolean

It saves your current Database Mining ActiveX configuration to the file specified in newVal.

Example

```

Dim ret As Boolean
ret = DBMining1.SaveConfigFile("C:\My Configs\cfg.dbm")

```

ScrollDwn

Moves the Grid content of one row down performing the same action that happens when the user scrolls down the Grid's vertical toolbar.

Example:

```
ThisDisplay.DBMining1.ScrollDwn
```

ScrollLeft

Moves the Grid content to the left performing the same action that happens when the user scrolls left the Grid's horizontal toolbar.

Example:

ThisDisplay.DBMining1.ScrollLeft

ScrollPgDwn

Moves the Grid content of one page down performing the same action that happens when the user scrolls down the Grid's vertical toolbar of one page.

Example:

ThisDisplay.DBMining1.ScrollPgDwn

ScrollPgLeft

Moves the Grid content of one page to the left performing the same action that happens when the user scrolls left the Grid's horizontal toolbar of one page.

Example:

ThisDisplay.DBMining1.ScrollPgLeft

ScrollPgRight

Moves the Grid content of one page to the right performing the same action that happens when the user scrolls right the Grid's horizontal toolbar of one page.

Example:

ThisDisplay.DBMining1.ScrollPgRight

ScrollPgUp

Moves the Grid content of one page up performing the same action that happens when the user scrolls up the Grid's vertical toolbar of one page.

Example:

ThisDisplay.DBMining1.ScrollPgUp

ScrollRight

Moves the Grid content to the right performing the same action that happens when the user scrolls right the Grid's horizontal toolbar.

Example:

ThisDisplay.DBMining1.ScrollUp

ScrollUp

Moves the Grid content of one row up performing the same action that happens when the user scrolls up the Grid's vertical toolbar.

Example:

ThisDisplay.DBMining1.ScrollUp

SetDBType(short NewDBType)

Sets the database type.

Example

DBMining1.SetDBType(1)

SortColumn(long ColID, VARIANT_BOOL bAscending)

Sorts the grid based upon a specified column in ascending or descending order.

Example:

'Sort by Column 1 ascending:

ThisDisplay.DBMining1.SortColumn 1, True

'Sort by Column 1 descending:

ThisDisplay.DBMining1.SortColumn 1, False

UnselectAllRows()

Unselects the currently selected area if present.

Example:

ThisDisplay.DBMining1.UnselectAllRows

