

DBT Grid Custom Control - Working Model Package

This file is a very basic description of the DBT Grid (Working Model) custom control. When you purchase the custom control you will receive the complete **DBT Grid Custom Control Reference Guide**.

Updates and revisions of the DBT Grid Working Model package will be made available on the DB Technologies Product support Subboard at (813) 739-2952, Subboard #53.

This Working Model VBX is not to be used in product development. It is circulated only for marketing purposes. Several properties and events have been removed from the Working Model package. Multi-Column and Multi-Row selection capabilities have been removed as well (due to the fact that DBT Grid **Actions** are not available in the Working Model package).

The Working Model does not provide for modifying edit key behavior, row and column selection **Actions/Events**, or grid cell formatting capabilities (Masks). The DBT Grid BackColor and ForeColor properties (standard properties) have also been removed from the Working Model.

Special Introductory Offer

For a limited time (before 02/01/94) you may purchase a release copy of the DBT Grid custom control for only \$29.⁹⁵ (normal retail price is \$39.⁹⁵). You will receive:

- + A fully functional DBT Grid custom control.
- + A DBT Grid Custom Control Reference Guide.
- + A DBT Grid Custom Control Programmer's Guide (with examples and source code).
- + A royalty-free license to distribute the DBT Grid VBX with your applications.
- + A discount certificate to purchase a data-aware version of the Grid for an additional \$19.⁹⁵. (Your choice of the VB3/Access Engine VBX or the Paradox Engine 3.0 VBX). The normal retail price of the data-aware controls are \$75.00 each.

(Special offer expires February 1, 1994).

To register send check or money order for \$39.95 (\$29.⁹⁵ for special introductory offer) to:

Douglas A. Bebber
2420 Briar Oak Circle
Sarasota, Florida 34232

(Make notes payable to: Douglas A. Bebber).

For product information call: (813) 378-3760.

DBT Grid (Working Model)

Description

The DBT Grid custom control provides an editable GRID control for your applications.

File Name

DBTGRID.VBX

Object Type

DBTGrid

Remarks

This control is useful for producing table objects or grid views of information. The DBT Grid is editable. A separate product is available for data-aware capabilities.

Note:

To use the Working Model you should copy the DBTGRID.VBX file into your WINDOWS/SYSTEM directory.

All of the properties, events, and methods for this control (Working Model) are listed below. Properties and events that apply only to this control, or require special consideration when used with it, are marked with an asterisk. They are documented below.

Properties

*About	FontBold	TabIndex
*CellStyle	FontItalic	TabStop
*Column	FontName	Tag
*ColumnHeading	FontSize	*Text
*ColumnHeadingStyle	FontStrikethru	Top
*ColumnLabels	FontUnderline	*Verticals
*Columns	Height	Visible
*ColumnSelection	*Horizontal	Width
*ColumnStringLength	Index	
*ColumnWidth	Left	
*DataModel	Name	
DragIcon	*Row	
DragMode	*RowLabels	
*EditMode	*Rows	
*EventMask	*RowSelection	

Events

*CellDoubleClicked	*GridHScroll
*CellFull	*GridPositionChanged
*CellValueChanged	*GridVScroll
DragDrop	LostFocus
DragOver	*MemoryError
GotFocus	

Methods

Drag	Refresh
Move	SetFocus

Properties

About

Description

Provides a Copyright notification to the user of the control.

Remarks

This property is read-only and is provided only as a copyright notification of the product.

Data Type

String

CellStyle

Description

Determines how text in a grid cell is to be aligned. The assignment is on the current column and all text in that column will have this alignment.

Usage

```
[form.]DBTGrid.CellStyle = [setting%]
```

Remarks

The column is the current column specified by DBTGrid.**Column**.

The CellStyle property settings are:

Setting	Description
0	Center
1	(Default) Left
2	Right

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()  
    ' Dimension the grid  
    DBTGrid1.Columns = 9 ' 9 columns wide  
    DBTGrid1.Rows = 30  ' by 30 rows deep  
    DBTGrid1.Column = 0 ' Set the current column  
    DBTGrid1.CellStyle = 0 ' Center all text in column #0  
    DBTGrid1.Column = 1 ' Move to next column.  
    DBTGrid1.CellStyle = 1 ' Left justify all text in column #1  
End Sub
```

Column

Description

Sets or returns the current column in the grid.

Usage

[form.]DBTGrid.Column = [setting%]

Remarks

Movement between grid columns is performed through this property. When the user moves throughout the grid this property is automatically updated to reflect the current column in the grid. Columns are numbered in sequence with the first column being 0. The maximum number of columns available in this control is 250 (0 - 249).

Data Type

Integer

Example

```
Sub Form_Load()  
    DBTGrid1.Column = 0 ' Set the current column  
    DBTGrid1.Row = 0 'Set the current row.  
    DBTGrid1.CellStyle = 0 ' Center all text in column #0  
    DBTGrid1.Column = 1 ' Move to next column.  
    DBTGrid1.CellStyle = 1 ' Left justify all text in column #1  
End Sub
```

ColumnHeading

Description

Sets or returns the column heading string for the grid.

Usage

[form.]DBTGrid.ColumnHeading = [stringexpression\$]

Remarks

Each column in the grid can have a ColumnHeading label. This label presents a description to the user concerning the information present in a grid column. The maximum size of a ColumnHeading string is 200 characters. Note that this property is only applicable when the ColumnLabels property is set to 3 (UserDefined). See the ColumnLabels description for more information.

Data Type

String

Example

```
Sub Form_Load()  
    DBTGrid1.Column = 0 ' Set the current column  
    DBTGrid1.Row = 0    'Set the current row.  
  
    ' Set the column heading.  
    DBTGrid1.ColumnHeading = "Customer No."  
  
    DBTGrid1.CellStyle = 0 ' Center all text in column #0  
    DBTGrid1.Column = 1 ' Move to next column.  
    DBTGrid1.CellStyle = 1 ' Left justify all text in column #1  
End Sub
```

ColumnHeadingStyle

Description

Sets or returns the column heading alignment for the grid.

Usage

[form.]DBTGrid.ColumnHeadingStyle = [setting%]

Remarks

Each column in the grid can have a column label (ColumnHeading string, column letters, or column numbers), this property is used to justify the column label in the grid column. The ColumnHeadingStyle property settings are:

Setting	Description
0	(Default) Center
1	Left
2	Right

Data Type

Integer (Enumerated)

Example

```

Sub Form_Load()
    DBTGrid1.Column = 0 ' Set the current column
    DBTGrid1.Row = 0    'Set the current row.

    ' Set the column heading.
    DBTGrid1.ColumnHeading = "Customer No."
    ' Now center the heading in the grid
    DBTGrid1.ColumnHeadingStyle = 0

' Do column #1 now

    DBTGrid1.Column = 1 ' Set the current column

    ' Set the column heading.
    DBTGrid1.ColumnHeading = "Last Name"
    ' Now center the heading in the grid
    DBTGrid1.ColumnHeadingStyle = 0

End Sub

```

ColumnLabels

Description

Sets or returns the column labels for the grid.

Usage

[form.]DBTGrid.ColumnLabels = [setting%]

Remarks

Each column in the grid can have a column label. Grid columns however do not need to have column labels and you can remove them by setting the ColumnLabels property to 0 (None). To label grid columns you have three choices:

Setting	Description
0	None (no column labels)
1	Letters (columns labeled with sequential letters)
2	Numbers (columns labeled with sequential numbers)
3	(Default) UserDefined (you label with ColumnHeading strings)

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30 ' Set the number of rows.  
    DBTGrid1.ColumnLabels = 3 ' User-defined column labels.  
End Sub
```

Columns

Description

Sets or returns the number of columns in the grid.

Usage

[form.]DBTGrid.Columns = [setting%]

Columns are numbered in sequence with the first column being 0. The maximum number of columns available in this control is 250 (0 - 249).

Data Type

Integer

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30 ' Set the number of rows.  
    DBTGrid1.ColumnLabels = 3 ' User-defined column labels.  
End Sub
```

ColumnSelection

Description

Sets or returns how grid columns can be selected by the user.

Usage

[form.]DBTGrid.ColumnSelection = [setting%]

Remarks

Columns can be made selectable for the user. This property determines if columns can be selected by the user, and if so, how columns can be selected. The ColumnSelection property settings are:

Setting	Description
0	None (columns can not be selected)
1	Single column selection supported
2	Multi-column selection supported (not available in Working Model!)

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30 ' Set the number of rows.  
    DBTGrid1.ColumnLabels = 3 ' User-defined column labels.  
    DBTGrid1.ColumnSelection = 0 ' Do not allow columns to be selected.  
End Sub
```

ColumnSize

Description

Sets or returns the width of a grid column.

Usage

[form.]DBTGrid.ColumnSize = [setting%]

Grid columns can be dynamically resized by the user as well as through program code. The number present in the ColumnSize property is in dialog units.

Data Type

Integer

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30 ' Set the number of rows.  
    DBTGrid1.Column = 0  
    DBTGrid1.ColumnSize = 100  
End Sub
```

ColumnStringLength

Description

Sets or returns the string length (in characters) of a grid column.

Usage

[form.]DBTGrid.ColumnStringLength = [setting%]

All data placed into the grid is of type string. Information is placed into the grid through the Text property. The grid columns accept string data via the Text property and all grid columns have a maximum string length. This string length is set through the ColumnStringLength property.

For example, if you have a Column set up to accept State information and you desire the user to enter the two letter state abbreviation, you will most likely set the column's ColumnStringLength property to 2. Once set, if the user tried to enter a third character, the grid would not accept it and would fire a **CellFull** event to handle the condition.

All grid columns are initially created with the ColumnStringLength set to 50 characters. The valid range for the ColumnStringLength property is 1 - 1000 characters.

Data Type

Integer

Example

```
Sub Form_Load()  
    DBTGrid1.Column = 0 ' Set the current column  
    DBTGrid1.Row = 0 ' Set the current row.  
  
    ' Set the column heading.  
    DBTGrid1.ColumnHeading = "Customer No."  
    ' Now center the heading in the grid  
    DBTGrid1.ColumnHeadingStyle = 0  
    ' Now set the max length of string in column #0 to 10 characters  
    DBTGrid1.ColumnStringLength = 10
```

End Sub

DataModel

Description

Sets or returns the DataModel for the control.

Usage

[form.]DBTGrid.DataModel = [setting%]

All data placed into the grid requires system memory. The grid control allocates internal data memory based on the value placed into DataModel property. The DataModel property settings are:

Setting	Description
0	Small (the grid holds data in a single 64K segment. When the segment is full a MemoryError event is generated.)
1	Large (the grid holds data in multiple 64K segments. The total grid data is limited only by the amount of global memory available to the system. When no further global memory allocated a MemoryError event is generated.)

Note:

The grid data size with DataModel set to 0 is approx 64Kb, with DataModel set to 1 is approx 64Mb.

Data Type

Integer (Enumerated)

Example

Sub Form_Load()

 'Set DataModel to Large (limited by global memory)

 DBTGrid1.DataModel = 1

 DBTGrid1.Columns = 9 ' Set the number of columns.

 DBTGrid1.Rows = 2000 ' Set the number of rows.

 ' 9 columns x 50 bytes (ColumnStringLength) x 2000 Rows = approx 879Kb

End Sub

EditMode

Description

Sets or returns the grid edit mode.

Usage

```
[form.]DBTGrid.EditMode = [setting%]
```

Remarks

The grid can be set to allow or disallow editing of information by the user. The EditMode property settings are:

Setting	Description
0	False (editing not allowed)
1	(Default) True (editing allowed)

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30 ' Set the number of rows.  
    DBTGrid1.EditMode = 1 ' Allow editing  
End Sub
```

EventMask

Description

Sets or returns the grid event mask.

Usage

```
[form.]DBTGrid.EventMask = [setting%]
```

Remarks

The grid supported events can be turned on and off at any point by the programmer. The EventMask property settings are:

Setting	Description
---------	-------------

0	(Default) False (events are allowed or are on)
1	True (events are masked off)

Note:

In the Working Model the EventMask property only has control over the **GridPositionChanged** event.

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()

    DBTGrid1.Columns = 9 ' Set the number of columns.
    DBTGrid1.Rows = 30  ' Set the number of rows.
    DBTGrid1.EditMode = 1 ' Allow editing
    ' Shut off DBTGrid events
    DBTGrid1.EventMask = 1

    ...
    ...
    " Turn DBTGrid event support back on
    DBTGrid1.EventMask = 0
End Sub
```

Horizontal

Description

Sets or returns the grids horizontal line support between grid rows.

Usage

```
[form.]DBTGrid.Horizontal = [setting%]
```

Remarks

The grid can have horizontal lines drawn to separate rows. The horizontal lines which separate rows can be turned on or off through the Horizontal property. The Horizontal property settings are:

Setting	Description
0	False (horizontal lines are off)
1	(Default) True (horizontal lines are on)

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()
```

```
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30  ' Set the number of rows.  
    DBTGrid1 Horizontals = 0 ' Turn off horizontal lines between rows.
```

```
End Sub
```

Row

Description

Sets or returns the current row in the grid.

Usage

```
[form.]DBTGrid.Row = [setting&]
```

Remarks

Movement between grid rows is performed through this property. When the user moves throughout the grid this property is automatically updated to reflect the current row in the grid. Rows are numbered in sequence with the first row being 0. The maximum number of rows available in this control is limited only by memory (see the **DataModel** property for memory specific information.) You can theoretically have more than 2 billion rows of data with the DBT Grid custom control.

Data Type

long

Example

```
Sub Form_Load()  
    DBTGrid1.Column = 0 ' Set the current column  
    DBTGrid1.Row = 0    ' Set the current row.  
    DBTGrid1.CellStyle = 0 ' Center all text in column #0  
    DBTGrid1.Row = 1    ' Move to next row.  
End Sub
```

RowLabels

Description

Sets or returns the row labels for the grid.

Usage

[form.]DBTGrid.RowLabels = [setting%]

Remarks

Rows can have labels which are controlled via the RowLabels property. The RowLabels property settings are:

Setting	Description
0	None (no row labels)
1	RowButtons (blank row buttons on each row)
2	(Default) RowNumbers (rows have buttons labeled with sequential numbers).

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30 ' Set the number of rows.  
    DBTGrid1.ColumnLabels = 3 ' User-defined column labels.  
    DBTGrid1.RowLabels = 1 ' Row buttons only  
End Sub
```

Rows

Description

Sets or returns the number of rows in the grid.

Usage

[form.]DBTGrid.Rows = [setting&]

Rows are numbered in sequence with the first row being 0. The maximum number of rows available in this control is limited only by memory (see the **DataModel** property for memory specific information.) You can theoretically have more than 2 billion rows of data with the DBT Grid custom control.

Data Type

long

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30  ' Set the number of rows.  
End Sub
```

RowSelection

Description

Sets or returns how grid rows can be selected by the user.

Usage

```
[form.]DBTGrid.RowSelection = [setting%]
```

Remarks

Rows can be made selectable for the user. This property determines if rows can be selected by the user, and if so, how rows can be selected. The RowSelection property settings are:

Setting	Description
0	None (rows can not be selected)
1	Single row selection supported
2	Multi-row selection supported (not available in Working Mode!)

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()  
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30  ' Set the number of rows.  
    DBTGrid1.ColumnLabels = 3 ' User-defined column labels.  
    DBTGrid1.RowSelection = 0 ' Do not allow columns to be selected.  
End Sub
```

Text

Description

Sets or returns the text of a grid cell.

Usage

[form.]DBTGrid.Text = [stringexpression\$]

All data placed into the grid is of type string. Information is placed into the grid through the Text property. The grid columns accept string data via the Text property and all grid columns have a maximum string length. This string length is set through the ColumnStringLength property.

For example, if you have a Column set up to accept State information and you desire the user to enter the two letter state abbreviation, you will most likely set the column's ColumnStringLength property to 2. Once set, if the user tried to enter a third character, the grid would not accept it and would fire a **CellFull** event to handle the condition.

All grid columns are initially created with the ColumnStringLength set to 50 characters. The valid range for the ColumnStringLength property is 1 - 1000 characters.

Data Type

String

Example

```
Sub Form_Load()  
    DBTGrid1.Column = 0 ' Set the current column  
    DBTGrid1.Row = 0    'Set the current row.  
  
    ' Set the column heading.  
    DBTGrid1.ColumnHeading = "Customer No."  
    ' Now center the heading in the grid  
    DBTGrid1.ColumnHeadingStyle = 0  
    ' Now set the max length of string in column #0 to 10 characters  
    DBTGrid1.ColumnStringLength = 10  
    ' Put text in the grid cell  
    DBTGrid1.Text = "123456789"  
  
End Sub
```

Verticals

Description

Sets or returns the grids vertical line support between grid columns.

Usage

[form.]DBTGrid.Verticals = [setting%]

Remarks

The grid can have vertical lines drawn to separate columns. The vertical lines which separate columns can be turned on or off through the Verticals property. The Verticals property settings are:

Setting	Description
0	False (vertical lines are off)
1	(Default) True (vertical lines are on)

Data Type

Integer (Enumerated)

Example

```
Sub Form_Load()
```

```
    DBTGrid1.Columns = 9 ' Set the number of columns.  
    DBTGrid1.Rows = 30  ' Set the number of rows.  
    DBTGrid1 Horizontals = 0 ' Turn off horizontal lines between rows.  
    DBTGrid1.Verticals = 0 ' Turn off vertical lines between columns.
```

```
End Sub
```

Events

CellDoubleClicked

Description

Occurs when a grid cell is double-clicked with the mouse.

Syntax

```
Sub DBTGrid1_CellDoubleClicked (Column As Integer, Row As Long)
```

Remarks

This event occurs every time a grid cell is double-clicked with a mouse. The cell double clicked is identified by the Column and Row arguments.

CellFull

Description

Occurs when a grid cell's text reaches the maximum value specified by the ColumnStringLength property or when the column needs to be resized.

Syntax

Sub *DBTGrid1_CellFull* (*Column As Integer, Row As Long*)

Remarks

This event notifies the program when the grid cell's limits have been met and appropriate action must be taken. The cell generating the event is identified by the arguments.

CellValueChanged

Description

Occurs when a grid cell's text has been changed.

Syntax

Sub *DBTGrid1_CellValueChanged* (*Column As Integer, Row As Long*)

Remarks

This event notifies the program when a grid cell's text has been changed. The cell generating the event is identified by the arguments.

GridHScroll

Description

Occurs when the grid has been horizontally scrolled.

Syntax

Sub *DBTGrid1_GridHScroll* ()

Remarks

This event notifies the program when the grid has been horizontally scrolled.

GridPositionChanged

Description

Occurs when the current grid cell position changes.

Syntax

Sub *DBTGrid1_GridPositionChanged* (*Column As Integer, Row As Long*)

Remarks

This event is generated every time the current grid's cell position is changed. When the user navigates through the grid this event fires every time a new cell is entered. The new grid cell position is identified by the arguments (Column, and Row).

GridVScroll

Description

Occurs when the grid has been vertically scrolled.

Syntax

Sub *DBTGrid1_GridVScroll* ()

Remarks

This event notifies the program when the grid has been vertically scrolled.

MemoryError

Description

Occurs when the grid no longer has memory space for row data.

Syntax

Sub *DBTGrid1_MemoryError* ()

Remarks

This event notifies the program when the grid control no longer has memory space for row data. (See the **DataModel** property information for memory related limits.)

Keyboard Templates

The keystrokes available for the DBT Grid (Working Model) are:

Key	Description
Up arrow	Moves up one cell
Down arrow	Moves down one cell
Right arrow	Moves right one cell
Left arrow	Moves left one cell

PgDn	Scroll down
PgUp	Scroll up
Home	Move to first cell in row
End	Move to last cell in row
Ctrl-Up	Move to top visible row
Ctrl-Down	Move to bottom visible row
Ctrl-Right	Move to right-most visible column
Ctrl-Left	Move to left-most visible column
Ctrl-PgUp	Scroll left
Ctrl-PgDn	Scroll right
Ctrl-Home	Move to top left grid cell
Ctrl-End	Move to bottom right grid cell
F2	Edit current grid cell
Enter	End cell editing
Esc	Abort cell editing

Note:

Row and Column selection keys are not fully implemented in the Working Model.