Contents

Introduction to the DialogsEXE.Dialog component

The DialogsEXE.Dialog component is a 32-bit ActiveX EXE component (hosted by DialogsEXE.exe) providing a graphical user interface (GUI) for Windows Script Host (VBScript / JScript) and any other language capable of instantiating COM components (E.G.: C++, Delphi, VB6, VBA, Python, Java, Perl, Ruby, PowerShell, C#, VB.NET, etc.).

When used with script languages, it allows building fully functional Windows applications using just a text editor, like Notepad.exe.

This ActiveX EXE component does this by making some of the Visual Basic objects and procedures (the so-called VB Runtime) available for COM-aware client applications.

Main features:

- Full set of functionalities for creating window interfaces. Supports most of the standard Windows controls.
- Complete Application Programming Interface (API) to manage all aspects of the graphical user interface.
- ActiveX support: it can dynamically load ActiveX controls and provide access to their interfaces, including event interfaces.
- Language-independent component: only COM support is required.
- Supports working with popular image formats, like Bitmaps (.bmp, .dib), GIFs (.gif), JPEGs (.jpg), Metafiles (.wmf, .emf), Icons (.ico).
- Supports: Windows 98SE, ME, NT4, 2000, XP, 2003, Vista, 2008, 7, 2008 R2, 8, 8.1, 2012, 2016, 10, 11.

This ActiveX EXE component is an OOP (out-of-process) component, I.E.: it runs in its own process so it can be used by both 32-bit and 64-bit client applications.

Public events of the DialogsEXE.Dialog component

Public Events related to **ActiveX** controls:

Public Event **ActiveXControlEvent** (intIndex, arrayEventParameters, strEventName)

Occurs when an ActiveX control raises an event.

Public Event **ActiveXControlGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **ActiveXControlLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **ActiveXControlValidate** (intIndex, boolCancel)

Occurs before the focus shifts to a (second) control that has its CausesValidation property set to True.

Public Events related to the CheckBox intrinsic control

Public Event **CheckClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **CheckGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **CheckKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **CheckKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **CheckKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **CheckLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **CheckMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **CheckMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **CheckMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **CheckValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the *dropdown-list style* ComboBox intrinsic control

Public Event **ComboChange** (intIndex)

Occurs when the contents of a control have changed.

Public Event **ComboClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **ComboDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **ComboDropDown** (intIndex)

Occurs when the list portion of a ComboBox control is about to drop down.

Public Event **ComboGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **ComboKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **ComboKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **ComboKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **ComboLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **ComboScroll** (intIndex)

Occurs when you reposition the scroll box on a control.

Public Event **ComboValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the *dropdown style* ComboBox intrinsic control

Public Event **ComboTextChange** (intIndex)

Occurs when the contents of a control have changed.

Public Event **ComboTextClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **ComboTextDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **ComboTextDropDown** (intIndex)

Occurs when the list portion of a ComboBox control is about to drop down.

Public Event **ComboTextGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **ComboTextKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **ComboTextKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **ComboTextKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **ComboTextLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **ComboTextScroll** (intIndex)

Occurs when you reposition the scroll box on a control.

Public Event **ComboTextValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the CommandButton intrinsic control

Public Event **CommandClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **CommandGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **CommandKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **CommandKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **CommandKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **CommandLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **CommandMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **CommandMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **CommandMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Events related to the **Data** control

Public Event **DataCtrlClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **DataCtrlDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **DataCtrlError** (intIndex, intDataErr, intResponse)

Occurs only as the result of a data access error that takes place when no DialogsEXE / Visual Basic code is being executed.

Public Event **DataCtrlGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **DataCtrlKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus..

Public Event **DataCtrlKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **DataCtrlKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus..

Public Event **DataCtrlLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **DataCtrlMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **DataCtrlMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **DataCtrlMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **DataCtrlReposition** (intIndex)

Occurs after a record becomes the current record.

Public Event **DataCtrlValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the **Form** intrinsic class

Public Event **FormActivate** ()

Occurs when a form becomes the active window.

Public Event **FormClick** ()

Occurs when the user presses and then releases a mouse button over an object.

Public Event **FormDblClick** ()

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **FormDeactivate** ()

Occurs when a form is no longer the active window.

Public Event **FormGotFocus** ()

Occurs when an object receives the focus.

Public Event **FormInitialize** ()

Occurs when an application creates an instance of a Form, MDIForm, or class.

Public Event **FormKeyDown** (intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **FormKeyPress** (intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **FormKeyUp** (intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **FormLoad** ()

Occurs when a form is loaded.

Public Event **FormLostFocus** ()

Occurs when an object loses the focus.

Public Event **FormMouseDown** (intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **FormMouseMove** (intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Events related to the **Form** intrinsic class (continued)

Public Event **FormMouseUp** (intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **FormPaint** ()

Occurs when any part of a form or PictureBox control is moved, enlarged, or exposed.

Public Event **FormQueryUnload** (intCancel, intUnloadMode)

Occurs before a form closes.

Public Event **FormResize** ()

Occurs when a form is first displayed or the size of an object changes.

Public Event **FormTerminate** ()

Occurs when all references to an instance of a Form, MDIForm, or class are removed from memory.

Public Event **FormUnload** (intCancel)

Occurs when a form is about to be removed from the screen.

Public Events related to the **Frame** intrinsic control

Public Event **FrameClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **FrameDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **FrameMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **FrameMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **FrameMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Events related to the HScrollBar intrinsic control

Public Event **HScrollBarChange** (intIndex)

Occurs when the scroll box portion of the scroll bar is moved.

Public Event **HScrollBarGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **HScrollBarKeyDown** (intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **HScrollBarKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **HScrollBarKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **HScrollBarLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **HScrollBarScroll** (intIndex)

Occurs when the scroll box on a ScrollBar control is repositioned or scrolled horizontally.

Public Event **HScrollBarValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the **Image** intrinsic control

Public Event **ImageClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **ImageDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **ImageMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **ImageMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **ImageMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Events related to the *check-box style* **ListBox** intrinsic control

Public Event **ListCheckClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **ListCheckDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **ListCheckGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **ListCheckItemCheck** (intIndex, intItem)

Occurs when a ListBox control's Style property is set to 1 (checkboxes) and an item's checkbox in the ListBox control is selected or cleared.

Public Event **ListCheckKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **ListCheckKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **ListCheckKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **ListCheckLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **ListCheckMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **ListCheckMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse

Public Event **ListCheckMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **ListCheckScroll** (intIndex)

Occurs when you reposition the scroll box on a control

Public Event **ListCheckValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the ListBox intrinsic control

Public Event **ListClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **ListDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **ListGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **ListKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **ListKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **ListKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **ListLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **ListMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **ListMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **ListMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **ListScroll** (intIndex)

Occurs when you reposition the scroll box on a control.

Public Event **ListValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the *multi-line* TextBox intrinsic control

Public Event **MultiTextChange** (intIndex)

Occurs when the contents of a control have changed.

Public Event **MultiTextClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **MultiTextDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **MultiTextGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **MultiTextKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **MultiTextKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **MultiTextKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **MultiTextLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **MultiTextMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **MultiTextMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **MultiTextMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **MultiTextValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the **OptionButton** intrinsic control

Public Event **OptionClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **OptionDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **OptionGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **OptionKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **OptionKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **OptionKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **OptionLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **OptionMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **OptionMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **OptionMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **OptionValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the **PictureBox** intrinsic control

Public Event **PictureChange** (intIndex)

Occurs when the contents of a control have changed.

Public Event **PictureClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **PictureDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **PictureGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **PictureKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **PictureKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **PictureKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **PictureLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **PictureMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **PictureMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **PictureMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **PicturePaint** (intIndex)

Occurs when any part of a form or PictureBox control is moved, enlarged, or exposed.

Public Event **PictureResize** (intIndex)

Occurs when a form is first displayed or the size of an object changes.

Public Events related to the PictureBox intrinsic control (continued)

Public Event **PictureValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the **TextBox** intrinsic control

Public Event **TextChange** (intIndex)

Occurs when the contents of a control have changed.

Public Event **TextClick** (intIndex)

Occurs when the user presses and then releases a mouse button over an object.

Public Event **TextDblClick** (intIndex)

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Public Event **TextGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **TextKeyDown** (intIndex, intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **TextKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **TextKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **TextLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **TextMouseDown** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user presses the mouse button while an object has the focus.

Public Event **TextMouseMove** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user moves the mouse.

Public Event **TextMouseUp** (intIndex, intButton, intShift, sngX, sngY)

Occurs when the user releases the mouse button while an object has the focus.

Public Event **TextValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public Events related to the Timer control

Public Event **TimerCtrlTimer** (intIndex)

Occurs when a preset interval for a Timer control has elapsed.

Public Events related to the VScrollBar intrinsic control

Public Event **VScrollBarChange** (intIndex)

Occurs when the scroll box portion of the scroll bar is moved.

Public Event **VScrollBarGotFocus** (intIndex)

Occurs when an object receives the focus.

Public Event **VScrollBarKeyDown** (intKeyCode, intShift)

Occurs when the user presses a key while an object has the focus.

Public Event **VScrollBarKeyPress** (intIndex, intKeyAscii)

Occurs when the user presses and releases an ANSI key.

Public Event **VScrollBarKeyUp** (intIndex, intKeyCode, intShift)

Occurs when the user releases a key while an object has the focus.

Public Event **VScrollBarLostFocus** (intIndex)

Occurs when an object loses the focus.

Public Event **VScrollBarScroll** (intIndex)

Occurs when the scroll box on a ScrollBar control is repositioned or scrolled vertically.

Public Event **VScrollBarValidate** (intIndex, boolCancel)

Occurs when a control loses focus to a control that causes validation.

Public properties and methods of the DialogsEXE.Dialog component

Public properties and methods related to the **Form** intrinsic class

Public Property **Caption**

Returns/sets the text displayed in form's title bar.

Public Property **Height**

Returns/sets the height of an object (in twips).

Public Property **Left**

Returns/sets the distance between the internal left edge of an object and the left edge of its container (in twips).

Public Property **ScaleHeight**

Returns the number of units for the vertical measurement of an object's interior (in twips).

Public Property **ScaleWidth**

Returns the horizontal coordinates for the left edges of an object (in twips).

Public Property **Top**

Returns/sets the distance between the internal top edge of an object and the top edge of its container (in twips).

Public Property **Topmost**

A form whose Topmost property is set to true appears above all windows whose Topmost properties are set to false.

Public Property **Width**

Returns/sets the width of an object (in twips).

Public Sub **GetForm** (Optional strCaption, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some attributes of the Form object.

Public properties and methods related to the **Form** intrinsic class (continued)

Public Sub **SetForm** (Optional ByVal strCaption, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional lngErrNumber, Optional strErrDescription)

Sets some attributes of the Form object.

Public Sub **Hide** (Optional lngErrNumber, Optional strErrDescription)

Hides a Form object but doesn't unload it.

Public Sub **ShowModal** (Optional lngErrNumber, Optional strErrDescription)

Displays the Form object modally (no subsequent code is executed until the form is hidden or unloaded).

Public Function **FormObject** (Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Form object.

Public Sub **FormCircleAbsolute** (ByVal sngX, ByVal sngY, ByVal sngRadius, Optional ByVal lngOLEColor, Optional ByVal sngStart, Optional ByVal sngEnd, Optional ByVal sngAspect, Optional lngErrNumber, Optional strErrDescription)

Draws a circle, ellipse, or arc on the Form object.

Public Sub **FormLineAbsolute** (ByVal sngX1, ByVal sngY1, ByVal sngX2, ByVal sngY2, Optional ByVal lngOLEColor, Optional ByVal boolBox, Optional ByVal boolFill, Optional lngErrNumber, Optional strErrDescription)

Draws lines and rectangles on the Form object.

Public Sub **FormPaintPicture** (ByVal byteStdPictureArray, ByVal sngX1, ByVal sngY1, Optional ByVal sngWidth1, Optional ByVal sngHeight1, Optional ByVal sngX2, Optional ByVal sngY2, Optional ByVal sngWidth2, Optional ByVal sngHeight2, Optional ByVal lngOpcode, Optional lngErrNumber, Optional strErrDescription)

Draws the contents of a graphics file on the Form object.

Public Sub **FormPrintText** (ByVal strText, Optional lngErrNumber, Optional strErrDescription)

Prints text on the Form object.

Public Sub **FormPSetAbsolute** (ByVal sngX, ByVal sngY, Optional ByVal lngOLEColor, Optional lngErrNumber, Optional strErrDescription)

Sets a point on the Form object to a specified color.

*See the **Members of the VB.Form** class paragraph, at the page 67 of this document, for a list of members to which this **Form object** reference provides access.*

Public properties and methods related to the **DialogsEXE** ActiveX EXE itself

Public Property **LogMode**

Returns a value that indicates the target (event log or log file) and other log attributes.

Public Property **LogPath**

Returns the filename for an application's log (if logging to a file).

Public Property **OLERequestPendingMsgText**

Returns/sets text of 'busy' message displayed while an Automation request is pending.

Public Property **OLERequestPendingMsgTitle**

Returns/sets title of 'busy' message displayed while an Automation request is pending.

Public Property **OLERequestPendingTimeoutMSecs**

Returns/sets milliseconds Automation requests will run before user actions trigger a 'busy' message.

Public Property **OLEServerBusyMsgText**

Returns/sets text of 'busy' message displayed if an ActiveX component rejects a request.

Public Property **OLEServerBusyMsgTitle**

Returns/sets title of 'busy' message displayed when an ActiveX component rejects a request.

Public Property **OLEServerBusyRaiseError**

Determines whether a rejected Automation request raises an error, instead of displaying a 'busy' message.

Public Property **OLEServerBusyTimeoutMSecs**

Returns/sets milliseconds during which an Automation request will continue to be retried.

Public Property **ProcessID**

Returns the ID of the executing process.

Public Property **ThreadID**

Returns the ID of the executing thread.

Public Sub **LogEvent** (ByVal strInfoToLog, Optional ByVal lngEventType, Optional lngErrNumber, Optional strErrDescription)

Logs an event in the application's log target.

Public properties and methods related to the DialogsEXE ActiveX EXE itself (continued)

Public Sub **LogInfo** (ByVal strInfoToLog, Optional lngErrNumber, Optional strErrDescription)

Logs an information in the DialogsEXE.txt log file, in the user's Documents folder.

Public Sub **StartLogging** (ByVal strLogTarget, ByVal lngLogModes, Optional lngErrNumber, Optional strErrDescription)

Sets the log target and log mode for an application.

Public properties and methods related to **ActiveX** controls

Public Function **AddActiveXControlWithEvents** (ByVal strProgID, Optional ByVal strLicenseKey, Optional ByVal objContainerRef, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Adds an ActiveX control to the ActiveX Controls WithEvents collection and returns the index of the added ActiveX control.*

Public Function **GetActiveXControlWithEvents** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the ActiveX control from the ActiveX Controls WithEvents collection based on the index of the control.*

Public Function **RemoveActiveXControlWithEvents** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Removes the specified ActiveX control from the ActiveX Controls WithEvents collection.*

Public Function **AddActiveXControlWithoutEvents** (ByVal strProgID, Optional ByVal strLicenseKey, Optional ByVal objContainerRef, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Adds an ActiveX control to the ActiveX Controls WithoutEvents collection and returns the index of the added ActiveX control.*

Public Function **GetActiveXControlWithoutEvents** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the ActiveX control from the ActiveX Controls WithoutEvents collection based on the index of the ActiveX control.*

Public Function **RemoveActiveXControlWithoutEvents** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Removes the specified ActiveX control from the ActiveX Controls WithoutEvents collection.*

Public methods related to the **CheckBox** intrinsic control

Public Function **LoadCheck** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a CheckBox control array element into memory and returns its index value.*

Public Function **UnloadCheck** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a CheckBox control array element from memory.*

Public Sub **GetCheck** (ByVal intIndex, Optional strCaption, Optional intTabIndex, Optional boolVisible, Optional intValue, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a CheckBox control array element, from the CheckBox controls array, based on the index of the control.*

Public Sub **SetCheck** (ByVal intIndex, Optional ByVal strCaption, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intValue, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a CheckBox control array element, from the CheckBox controls array, based on the index of the control.*

Public Function **CheckObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Check ListBox control array element, from the Check ListBox controls array, based on the index of the control.*

*See the **Members of the VB.CheckBox** class paragraph, at the page 77 of this document, for a list of members to which this **Check object** reference provides access.*

Public methods related to the *dropdown-list style* ComboBox intrinsic control

Public Function **LoadCombo** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a dropdown-list ComboBox control array element into memory and returns its index value.*

Public Function **UnloadCombo** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a dropdown-list ComboBox control array element from memory.*

Public Sub **GetCombo** (ByVal intIndex, Optional arrayItemsList, Optional intTabIndex, Optional boolVisible, Optional intListIndex, Optional strListValue, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Dropdown list ComboBox control array element, from the Dropdown list ComboBox controls array, based on the index of the control.*

Public Sub **SetCombo** (ByVal intIndex, Optional ByVal arrayItemsList, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intListIndex, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Dropdown list ComboBox control array element, from the Dropdown list ComboBox controls array, based on the index of the control.*

Public Function **ComboObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Dropdown list ComboBox control array element, from the Dropdown list ComboBox controls array, based on the index of the control.*

*See the **Members of the VB.ComboBox** class paragraph, at the page 82 of this document, for a list of members to which this **Combo object** reference provides access.*

Public methods related to the *dropdown style* ComboBox intrinsic control

Public Function **LoadComboText** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a Dropdown combo ComboBox control array element into memory and returns its index value.*

Public Function **UnloadComboText** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a Dropdown combo ComboBox control array element from memory.*

Public Sub **GetComboText** (ByVal intIndex, Optional arrayItemsList, Optional intTabIndex, Optional boolVisible, Optional intListIndex, Optional strListValue, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Dropdown combo ComboBox control array element, from the Dropdown combo ComboBox controls array, based on the index of the control.*

Public Sub **SetComboText** (ByVal intIndex, Optional ByVal strText, Optional ByVal arrayItemsList, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intListIndex, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Dropdown combo ComboBox control array element, from the Dropdown combo ComboBox controls array, based on the index of the control.*

Public Function **ComboTextObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Dropdown combo ComboBox control array element, from the Dropdown combo ComboBox controls array, based on the index of the control.*

*See the **Members of the VB.ComboBox** class paragraph, at the page 82 of this document, for a list of members to which this **ComboText object** reference provides access.*

Public methods related to the **CommandButton** intrinsic control

Public Function **LoadCommand** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a CommandButton control array element into memory and returns its index value.*

Public Function **UnloadCommand** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a CommandButton control array element from memory.*

Public Sub **GetCommand** (ByVal intIndex, Optional strCaption, Optional boolDefault, Optional boolCancel, Optional intTabIndex, Optional boolVisible, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a CommandButton control array element, from the CommandButton controls array, based on the index of the control.*

Public Sub **SetCommand** (ByVal intIndex, Optional ByVal strCaption, Optional ByVal boolDefault, Optional ByVal boolCancel, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a CommandButton control array element, from the CommandButton controls array, based on the index of the control.*

Public Function ***CommandObject*** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the CommandButton control array element, from the CommandButton controls array, based on the index of the control.*

*See the **Members of the VB.CommandButton** class paragraph, at the page 88 of this document, for a list of members to which this **Command object** reference provides access.*

Public methods related to the **Data** control

Use the Data control carefully as some computers don't have all the Microsoft libraries required by this control. When such libraries are missing, you should not use the Data control for your database-related scripts or programs; instead, you should use ADO with an adequate OLE DB Provider.

Public Function **LoadDataCtrl** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of a Data control array element into memory and returns its index value.

Public Function **UnloadDataCtrl** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads a Data control array element from memory.

Public Sub **GetDataCtrl** (ByVal intIndex, Optional intTabIndex, Optional boolVisible, Optional strConnect, Optional strDatabaseName, Optional intDefaultCursorType, Optional intDefaultType, Optional boolExclusive, Optional intOptions, Optional objRecordset, Optional intRecordsetType, Optional strRecordSource, Optional intEditMode, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of a Data control array element, from the Data controls array, based on the index of the control.

Public Sub **SetDataCtrl** (ByVal intIndex, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal strConnect, Optional ByVal strDatabaseName, Optional ByVal intDefaultCursorType, Optional ByVal intDefaultType, Optional ByVal boolExclusive, Optional ByVal intOptions, Optional ByVal objRecordset, Optional ByVal intRecordsetType, Optional ByVal strRecordSource, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of a Data control array element, from the Data controls array, based on the index of the control.

Public Function **DataCtrlObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Data control array element, from the Data controls array, based on the index of the control.

Public Function **DataCtrlOpenData** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Closes and rebuilds the Recordset object or data structures created by the Data control array element, from the Data controls array, based on the index of the control.

Properties related to the **Data** control, accessible through late-binding

The reference to the Data control array element, returned by the **DataCtrlObject** function, provides access (through late-binding mechanism) to the following group of properties:

Appearance

Returns/sets whether or not an object is painted at run time with 3-D effects (possible values 0 or 1).

BackColor

Returns/sets the background color used to display text and graphics in an object.

BackStyle

Indicates whether the background of the Data control is transparent or opaque (possible values 0 or 1).

BorderStyle

Returns/sets the border style for an object (possible values 0 or 1).

CaptionFirst

Returns/sets the caption for the "navigate first" command button.

CaptionLast

Returns/sets the caption for the "navigate last" command button.

CaptionNext

Returns/sets the caption for the "navigate next" command button.

CaptionPrevious

Returns/sets the caption for the navigate previous command button.

Connect

The string value indicating the source of an open database, a database used in a pass-through query, or an attached table (default value: Access 2000;). For base tables in a Micorosoft Jet database (.mdb), the Connect property setting could be a zero-length string ("").

The following list shows possible database types and their corresponding database specifiers and paths for the Connect property setting. In an ODBCDirect workspace, only the "ODBC" specifier can be used.

Properties related to the **Data** control, accessible through late-binding (continued)

Database type: **Microsoft Jet Database**

Specifier: [database];

Example: Example: drive:\path\filename.mdb

Database type: **dBASE III**

Specifier: dBASE III;

Example: drive:\path

Database type: **dBASE IV**

Specifier: dBASE IV;

Example: drive:\path

Database type: **dBASE 5**

Specifier: dBASE 5.0;

Example: drive:\path

Database type: **Paradox 3.x**

Specifier: Paradox 3.x;

Example: drive:\path

Database type: **Paradox 4.x**

Specifier: Paradox 4.x;

Example: drive:\path

Database type: **Paradox 5.x**

Specifier: Paradox 5.x;

Example: drive:\path

Database type: **Microsoft Excel 3.0**

Specifier: Excel 3.0;

Example: drive:\path\filename.xls

Database type: **Microsoft Excel 4.0**

Specifier: Excel 4.0;

Example: drive:\path\filename.xls

Properties related to the **Data** control, accessible through late-binding (continued)

Database type: **Microsoft Excel 5.0 or Excel 95**

Specifier: Excel 5.0;

Example: drive:\path\filename.xls

Database type: **Microsoft Excel 97**

Specifier: Excel 8.0;

Example: drive:\path\filename.xls

Database type: **Lotus 1-2-3 WKS and WK1**

Specifier: Lotus WK1;

Example: drive:\path\filename.wk1

Database type: **Lotus 1-2-3 WK3**

Specifier: Lotus WK3;

Example: drive:\path\filename.wk3

Database type: **Lotus 1-2-3 WK4**

Specifier: Lotus WK4;

Example: drive:\path\filename.wk4

Database type: **HTML Import**

Specifier: HTML Import;

Example: drive:\path\filename

Database type: **HTML Export**

Specifier: HTML Export;

Example: drive:\path

Database type: **Text**

Specifier: Text;

Example: drive:\path

Properties related to the **Data** control, accessible through late-binding (continued)

Database type: **ODBC**

Specifier: ODBC;

DATABASE=database;

UID=user;

PWD=password;

DSN= datasourcename;

[LOGINTIMEOUT=seconds;]

Example: None

Database type: **Microsoft Exchange**

Specifier: Exchange 4.0;

MAPILEVEL=folderpath;

[TABLETYPE={ 0 | 1}];

[PROFILE=profile;]

[PWD=password;]

[DATABASE=database;]

Example: drive:\path\filename.mdb

Remarks

If the specifier is only "ODBC;", the ODBC driver displays a dialog box listing all registered ODBC data source names so that the user can select a database.

DatabaseName

The string value containing the name and location of the source of data for a Data control.

Properties related to the **Data** control, accessible through late-binding (continued)

DefaultCursorType

Returns/sets an integer value which indicates the cursor library used by the Data control (for "ODBCDirect" only; not for "Jet-connected ODBC data sources").

0: The ODBC driver will choose the appropriate style of cursors. Server-side cursors are used if they are available.

1: Use the ODBC cursor library. This option gives better performance for small result sets, but degrades quickly for larger result sets.

2: Use server-side cursors. For most large operations this gives better performance, but might cause more network traffic.

DefaultType

The integer value which determines the type of data source (Jet or ODBCDirect) that is used by the Data control.

1: Use ODBC

2: Use Jet

EditMode (read only)

Returns a value that indicates the state of editing for the current record.

0: No editing operation is in progress.

1: The Edit method has been invoked, and the current record is in the copy buffer.

2: The AddNew method has been invoked, and the current record in the copy buffer is a record row that hasn't been saved in the database.

Exclusive

A boolean value indicating whether the underlying database for a Data control is opened for single-user or multiuser access.

Font

Returns / sets a Font object.

Properties related to the **Data** control, accessible through late-binding (continued)

Options

The integer value that specifies one or more characteristics of the Recordset object in the control's Recordset property.

Settings

Use one or more of the following values to set the Options property. If you use more than one option, you must add their values:

1: In a multi-user environment, other users can't make changes to records in the Recordset.

2: In a multi-user environment, other users can't read records (table-type Recordset only).

4: You can't make changes to records in the Recordset.

8: You can add new records to the Recordset, but you can't read existing records.

16: Updates can apply to all fields of the Recordset, even if they violate the join condition.

32: (Default) Updates apply only to those fields that don't violate the join condition.

64: When using Data controls with an SQL statement in the RecordSource property, sends the SQL statement to an ODBC database, such as a SQL Server or Oracle database, for processing.

256: The Recordset object supports forward-only scrolling. The only move method allowed is MoveNext. This option cannot be used on Recordset objects manipulated with the Data control.

512: Generate a trappable error if another user is changing data you are editing.

ReadOnly

Returns or sets a value that determines whether the control's Database is opened for read-only access.

Recordset

The Recordset object defined by a Data control's properties or by an existing Recordset.

Properties related to the **Data** control, accessible through late-binding (continued)

RecordsetType

The integer value indicating the type of Recordset object the Data control will create.

0: A table-type Recordset

1: (Default) A dynaset-type Recordset

2: A snapshot-type Recordset

RecordSource

The string representing the underlying table, SQL statement, or QueryDef object for a Data control.

Methods related to the Data control, accessible through late-binding (continued)

OpenData

Closes and rebuilds the Recordset object or data structures created by the Data control.

Public methods related to the **Frame** intrinsic control

Public Function **LoadFrame** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a Frame control array element into memory and returns its index value.*

Public Function **UnloadFrame** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a Frame control array element from memory.*

Public Sub **GetFrame** (ByVal intIndex, Optional strCaption, Optional intTabIndex, Optional boolVisible, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Frame control array element, from the Frame controls array, based on the index of the control.*

Public Sub **SetFrame** (ByVal intIndex, Optional ByVal strCaption, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Frame control array element, from the Frame controls array, based on the index of the control.*

Public Function **FrameObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Frame control array element, from the Frame controls array, based on the index of the control.*

*See the **Members of the VB.Frame** class paragraph, at the page 93 of this document, for a list of members to which this **Frame object** reference provides access.*

Public methods related to the HScrollBar intrinsic control

Public Function **LoadHScrollBar** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a Horizontal ScrollBar control array element into memory and returns its index value.*

Public Function **UnloadHScrollBar** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a Horizontal ScrollBar control array element from memory.*

Public Sub **GetHScrollBar** (ByVal intIndex, Optional intTabIndex, Optional boolVisible, Optional intValue, Optional intMin, Optional intMax, Optional intSmallChange, Optional intLargeChange, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Horizontal ScrollBar control array element, from the Horizontal ScrollBar controls array, based on the index of the control.*

Public Sub **SetHScrollBar** (ByVal intIndex, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intValue, Optional ByVal intMin, Optional ByVal intMax, Optional ByVal intSmallChange, Optional ByVal intLargeChange, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Horizontal ScrollBar control array element, from the Horizontal ScrollBar controls array, based on the index of the control.*

Public Function **HScrollBarObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Horizontal ScrollBar control array element, from the Horizontal ScrollBar controls array, based on the index of the control.*

*See the **Members of the VB.HScrollBar** class paragraph, at the page 97 of this document, for a list of members to which this **HScrollBar object** reference provides access.*

Public methods related to the **Image** intrinsic control

Public Function **LoadImage** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of an Image control array element into memory and returns its index value.

Public Function **UnloadImage** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads an Image control array element from memory.

Public Sub **GetImageCtrl** (ByVal intIndex, Optional byteStdPictureArray, Optional intBorderStyle, Optional boolStretch, Optional boolVisible, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of an Image control array element, from the Image controls array, based on the index of the control.

Public Function **GetImageCtrlPictureArray** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns, as an array of bytes, the Picture property of the Image control array element, from the Image controls array, identified by its index (for JScript clients).

Public Sub **SetImageCtrl** (ByVal intIndex, Optional ByVal byteStdPictureArray, Optional ByVal intBorderStyle, Optional ByVal strPicturePath, Optional ByVal boolStretch, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of an Image control array element, from the Image controls array, based on the index of the control.

Public Sub **LoadPictureIntoImageCtrl** (ByVal intIndex, Optional ByVal strFileName, Optional ByVal lngSize, Optional ByVal lngColorDepth, Optional ByVal lngX, Optional ByVal lngY, Optional lngErrNumber, Optional strErrDescription)

Loads a graphic, from a file, into an Image control array element identified by the index of the control.

Public Sub **SavePictureFromImageCtrl** (ByVal intIndex, ByVal strFileName, Optional lngErrNumber, Optional strErrDescription)

Saves a graphic, from an Image control array element identified by the index of the control, to a file.

Public methods related to the Image intrinsic control (continued)

Public Function **ImageObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Image control array element, from the Image controls array, based on the index of the control.*

*See the **Members of the VB.Image** class paragraph, at the page 101 of this document, for a list of members to which this **Image object** reference provides access.*

Public methods related to the **Label** intrinsic control

Public Function **LoadLabel** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of a Label control array element into memory and returns its index value.

Public Function **UnloadLabel** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads a Label control array element from memory.

Public Sub **GetLabel** (ByVal intIndex, Optional strCaption, Optional intTabIndex, Optional intBorderStyle, Optional boolVisible, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of a Label control array element, from the Label controls array, based on the index of the control.

Public Sub **SetLabel** (ByVal intIndex, Optional ByVal strCaption, Optional ByVal intTabIndex, Optional ByVal intBorderStyle, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of a Label control array element, from the Label controls array, based on the index of the control.

Public Function **LabelObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Label control array element, from the Label controls array, based on the index of the control.

*See the **Members of the VB.Label** class paragraph, at the page 104 of this document, for a list of members to which this **Label object** reference provides access.*

Public methods related to the **Line** intrinsic control

Public Function **LoadLine** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of a Line control array element into memory and returns its index value.

Public Function **UnloadLine** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads a Line control array element from memory.

Public Sub **GetLine** (ByVal intIndex, Optional lngBorderColor, Optional intBorderStyle, Optional boolVisible, Optional intBorderWidth, Optional sngX1, Optional sngX2, Optional sngY1, Optional sngY2, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of a Line control array element, from the Line controls array, based on the index of the control.

Public Sub **SetLine** (ByVal intIndex, Optional ByVal lngBorderColor, Optional ByVal intBorderStyle, Optional ByVal boolVisible, Optional ByVal intBorderWidth, Optional ByVal sngX1, Optional ByVal sngX2, Optional ByVal sngY1, Optional ByVal sngY2, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of a Line control array element, from the Line controls array, based on the index of the control.

Public Function **LineObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Line control array element, from the Line controls array, based on the index of the control.

*See the **Members of the VB.Line** class paragraph, at the page 108 of this document, for a list of members to which this **Line object** reference provides access.*

Public methods related to the *standard style* ListBox intrinsic control

Public Function **LoadList** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a standard ListBox control array element into memory and returns its index value.*

Public Function **UnloadList** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a standard ListBox control array element from memory.*

Public Sub **GetList** (ByVal intIndex, Optional arrayItemsList, Optional intTabIndex, Optional boolVisible, Optional intListIndex, Optional strListValue, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Standard ListBox control array element, from the Standard ListBox controls array, based on the index of the control.*

Public Sub **SetList** (ByVal intIndex, Optional ByVal arrayItemsList, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intListIndex, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Standard ListBox control array element, from the Standard ListBox controls array, based on the index of the control.*

Public Function **ListObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Standard ListBox control array element, from the Standard ListBox controls array, based on the index of the control.*

Public Function **ListShowHScrollBar** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Displays a horizontal scroll bar on a standard ListBox control array element, from the standard ListBox controls array, based on the index of the control.*

*See the **Members of the VB.ListBox** class paragraph, at the page 110 of this document, for a list of members to which this **List object** reference provides access.*

Public methods related to the *checkbox style* ListBox intrinsic control

Public Function **LoadListCheck** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of a check ListBox control array element into memory and returns its index value.

Public Function **UnloadListCheck** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads a check ListBox control array element from memory.

Public Sub **GetListCheck** (ByVal intIndex, Optional arrayItemsList, Optional arraySelectionsList, Optional intTabIndex, Optional boolVisible, Optional intListIndex, Optional strListValue, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of a Check ListBox control array element, from the Check ListBox controls array, based on the index of the control.

Public Sub **SetListCheck** (ByVal intIndex, Optional ByVal arrayItemsList, Optional ByVal arraySelectionsList, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intListIndex, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of a Check ListBox control array element, from the Check ListBox controls array, based on the index of the control.

Public Function **ListCheckObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Check ListBox control array element, from the Check ListBox controls array, based on the index of the control.

Public Function **ListCheckShowHScrollBar** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Displays a horizontal scroll bar on a check ListBox control array element, from the check ListBox controls array, based on the index of the control.

*See the **Members of the VB.ListBox** class paragraph, at the page 110 of this document, for a list of members to which this **ListCheck object** reference provides access.*

Public methods related to the *multiline* TextBox intrinsic control

Public Function **LoadMultiText** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a multiline TextBox control array element into memory and returns its index value.*

Public Function **UnloadMultiText** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a multiline TextBox control array element from memory.*

Public Sub **GetMultiText** (ByVal intIndex, Optional strText, Optional lngMaxLength, Optional intTabIndex, Optional boolVisible, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Multiline TextBox control array element, from the Multiline TextBox controls array, based on the index of the control.*

Public Sub **SetMultiText** (ByVal intIndex, Optional ByVal strText, Optional ByVal lngMaxLength, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Multiline TextBox control array element, from the Multiline TextBox controls array, based on the index of the control.*

Public Function **MultiTextObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Multiline TextBox control array element, from the Multiline TextBox controls array, based on the index of the control.*

*See the **Members of the VB.TextBox** class paragraph, at the page 116 of this document, for a list of members to which this **MultiText object** reference provides access.*

Public methods related to the **OptionButton** intrinsic control

Public Function **LoadOption** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of an OptionButton control array element into memory and returns its index value.

Public Function **UnloadOption** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads an OptionButton control array element from memory.

Public Sub **GetOption** (ByVal intIndex, Optional strCaption, Optional intTabIndex, Optional boolVisible, Optional boolValue, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of an OptionButton control array element, from the OptionButton controls array, based on the index of the control.

Public Sub **SetOption** (ByVal intIndex, Optional ByVal strCaption, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal boolValue, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of an OptionButton control array element, from the OptionButton controls array, based on the index of the control.

Public Function **OptionObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the OptionButton control array element, from the OptionButton controls array, based on the index of the control.

*See the **Members of the VB.OptionButton** class paragraph, at the page 122 of this document, for a list of members to which this **Option object** reference provides access.*

Public methods related to the **PictureBox** intrinsic control

Public Function **LoadPictureCtrl** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a PictureBox control array element into memory and returns its index value.*

Public Function **UnloadPictureCtrl** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a PictureBox control array element from memory.*

Public Sub **GetPictureCtrl** (ByVal intIndex, Optional byteStdPictureArray, Optional intBorderStyle, Optional boolAutoSize, Optional boolVisible, Optional intTabIndex, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a PictureBox control array element, from the PictureBox controls array, based on the index of the control.*

Public Function **GetPictureCtrlPictureArray** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns, as an array of bytes, the Picture property of the PictureBox control array element, from the PictureBox controls array, identified by its index (for JScript clients).*

Public Sub **SetPictureCtrl** (ByVal intIndex, Optional ByVal byteStdPictureArray, Optional ByVal intBorderStyle, Optional ByVal strPicturePath, Optional ByVal boolAutoSize, Optional ByVal boolVisible, Optional ByVal intTabIndex, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a PictureBox control array element, from the PictureBox controls array, based on the index of the control.*

Public Sub **LoadPictureIntoPictureCtrl** (ByVal intIndex, Optional ByVal strFileName, Optional ByVal lngSize, Optional ByVal lngColorDepth, Optional ByVal lngX, Optional ByVal lngY, Optional lngErrNumber, Optional strErrDescription)

> *Loads a graphic, from a file, into a PictureBox control array element identified by the index of the control.*

Public Sub **SavePictureFromPictureCtrl** (ByVal intIndex, ByVal strFileName, Optional lngErrNumber, Optional strErrDescription)

> *Saves a graphic, from a PictureBox control array element identified by the index of the control, to a file.*

Public methods related to the **PictureBox** intrinsic control (continued)

Public Function **PictureCtrlObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the PictureBox control array element, from the PictureBox controls array, based on the index of the control.

Public methods related to the PictureBox intrinsic control (continued)

Public Sub **PictureCtrlCircleAbsolute** (ByVal intIndex, ByVal sngX, ByVal sngY, ByVal sngRadius, Optional ByVal lngOLEColor, Optional ByVal sngStart, Optional ByVal sngEnd, Optional ByVal sngAspect, Optional lngErrNumber, Optional strErrDescription)

> *Draws a circle, ellipse, or arc on the PictureBox control array element, from the PictureBox controls array, identified by the index of the control..*

Public Sub **PictureCtrlLineAbsolute** (ByVal intIndex, ByVal sngX1, ByVal sngY1, ByVal sngX2, ByVal sngY2, Optional ByVal lngOLEColor, Optional ByVal boolBox, Optional ByVal boolFill, Optional lngErrNumber, Optional strErrDescription)

> *Draws lines and rectangles on the PictureBox control array element, from the PictureBox controls array, identified by the index of the control.*

Public Sub **PictureCtrlPaintPicture** (ByVal intIndex, ByVal byteStdPictureArray, ByVal sngX1, ByVal sngY1, Optional ByVal sngWidth1, Optional ByVal sngHeight1, Optional ByVal sngX2, Optional ByVal sngY2, Optional ByVal sngWidth2, Optional ByVal sngHeight2, Optional ByVal lngOpcode, Optional lngErrNumber, Optional strErrDescription)

> *Draws the contents of a graphics file, represented by the byte array, on the PictureBox control array element, from the PictureBox controls array, identified by the index of the control.*

Public Sub **PictureCtrlPrintText** (ByVal intIndex, strText, Optional lngErrNumber, Optional strErrDescription)

> *Prints text on the PictureBox control array element, from the PictureBox controls array, identified by the index of the control.*

Public Sub **PictureCtrlPSetAbsolute** (ByVal intIndex, ByVal sngX, ByVal sngY, Optional ByVal lngOLEColor, Optional lngErrNumber, Optional strErrDescription)

> *Sets a point on the PictureBox control array element, from the PictureBox controls array, identified by the index of the control, to a specified color.*

*See the **Members of the VB.PictureBox** class paragraph, at the page 128 of this document, for a list of members to which this **PictureCtrl object** reference provides access.*

Public methods related to the **Shape** intrinsic control

Public Function **LoadShape** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of a Shape control array element into memory and returns its index value.

Public Function **UnloadShape** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads a Shape control array element from memory.

Public Sub **GetShape** (ByVal intIndex, Optional intShape, Optional intFillStyle, Optional Visible, Optional intBorderStyle, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

Returns some of the attributes of a Shape control array element, from the Shape controls array, based on the index of the control.

Public Sub **SetShape** (ByVal intIndex, Optional ByVal intShape, Optional ByVal intFillStyle, Optional ByVal boolVisible, Optional ByVal intBorderStyle, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

Sets some of the attributes of a Shape control array element, from the Shape controls array, based on the index of the control.

Public Function **ShapeObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Shape control array element, from the Shape controls array, based on the index of the control.

*See the **Members of the VB.Shape** class paragraph, at the page 136 of this document, for a list of members to which this **Shape object** reference provides access.*

Public methods related to the TextBox intrinsic control

Public Function **LoadText** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a TextBox control array element into memory and returns its index value.*

Public Function **UnloadText** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a TextBox control array element from memory.*

Public Sub **GetText** (ByVal intIndex, Optional strText, Optional lngMaxLength, Optional strPasswordChar, Optional intTabIndex, Optional boolVisible, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a TextBox control array element, from the TextBox controls array, based on the index of the control.*

Public Sub **SetText** (ByVal intIndex, Optional ByVal strText, Optional ByVal lngMaxLength, Optional ByVal strPasswordChar, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a TextBox control array element, from the TextBox controls array, based on the index of the control.*

Public Function **TextObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the TextBox control array element, from the TextBox controls array, based on the index of the control.*

*See the **Members of the VB.TextBox** class paragraph, at the page 116 of this document, for a list of members to which this **Text object** reference provides access.*

Public methods related to the Timer control

Public Function **LoadTimerCtrl** (Optional lngErrNumber, Optional strErrDescription)

Loads a new instance of a Timer control array element into memory and returns its index value.

Public Function **UnloadTimerCtrl** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

Unloads a Timer control array element from memory.

Public Sub **GetTimerCtrl** (ByVal intIndex, Optional lngInterval, Optional lngErrNumber, Optional strErrDescription)

Returns the attributes of a Timer control array element, from the Timer controls array, based on the index of the control.

Public Sub **SetTimerCtrl** (ByVal intIndex, Optional ByVal lngInterval, Optional ByVal boolEnabled, Optional lngErrNumber, Optional strErrDescription)

Sets the attributes of a Timer control array element, from the Timer controls array, based on the index of the control.

Public methods related to the VScrollBar intrinsic control

Public Function **LoadVScrollBar** (Optional ByVal intPictureContainerIndex, Optional lngErrNumber, Optional strErrDescription)

> *Loads a new instance of a Vertical ScrollBar control array element into memory and returns its index value.*

Public Function **UnloadVScrollBar** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Unloads a Vertical ScrollBar control array element from memory.*

Public Sub **GetVScrollBar** (ByVal intIndex, Optional intTabIndex, Optional boolVisible, Optional intValue, Optional intMin, Optional intMax, Optional intSmallChange, Optional intLargeChange, Optional sngLeft, Optional sngTop, Optional sngWidth, Optional sngHeight, Optional lngErrNumber, Optional strErrDescription)

> *Returns some of the attributes of a Vertical ScrollBar control array element, from the Vertical ScrollBar controls array, based on the index of the control.*

Public Sub **SetVScrollBar** (ByVal intIndex, Optional ByVal intTabIndex, Optional ByVal boolVisible, Optional ByVal intValue, Optional ByVal intMin, Optional ByVal intMax, Optional ByVal intSmallChange, Optional ByVal intLargeChange, Optional ByVal sngLeft, Optional ByVal sngTop, Optional ByVal sngWidth, Optional ByVal sngHeight, Optional ByVal intZOrder, Optional lngErrNumber, Optional strErrDescription)

> *Sets some of the attributes of a Vertical ScrollBar control array element, from the Vertical ScrollBar controls array, based on the index of the control.*

Public Function **VScrollBarObject** (ByVal intIndex, Optional lngErrNumber, Optional strErrDescription)

> *Returns a reference to the Vertical ScrollBar control array element, from the Vertical ScrollBar controls array, based on the index of the control.*

*See the **Members of the VB.VScrollBar** class paragraph, at the page 138 of this document, for a list of members to which this **VScrollBar object** reference provides access.*

Clipboard-related public methods

Public Sub **ClipboardClear** (Optional lngErrNumber, Optional strErrDescription)

Clears the contents of the system Clipboard.

Public Sub **ClipboardGetFormat** (ByVal intFormat, Optional lngErrNumber, Optional strErrDescription)

Returns a value indicating whether an item on the Clipboard object matches a specified format.

Public Sub **ClipboardGetText** (Optional ByVal intFormat, Optional lngErrNumber, Optional strErrDescription)

Returns a text string from the Clipboard object.

Public Sub **ClipboardSetText** (ByVal strText, Optional ByVal intFormat, Optional lngErrNumber, Optional strErrDescription)

Puts a text string on the Clipboard object using the specified Clipboard object format.

Public Sub **ClipboardGetData** (byteStdPictureArray, Optional intFormat, Optional lngErrNumber, Optional strErrDescription)

Returns the byte array representing the graphic from the Clipboard object.

Public Sub **ClipboardSetData** (ByVal byteStdPictureArray, Optional ByVal intFormat, Optional lngErrNumber, Optional strErrDescription)

Puts a picture represented by the byte array on the Clipboard object using the specified graphic format.

Printer-related public methods

Public Function **ListOfInstalledPrinters** (Optional lngErrNumber, Optional strErrDescription)

Returns an array containing the device name for the installed printers.

Public Sub **PrinterCircleAbsolute** (ByVal sngX, ByVal sngY, ByVal sngRadius, Optional ByVal lngOLEColor, Optional ByVal sngStart, Optional ByVal sngEnd, Optional ByVal sngAspect, Optional lngErrNumber, Optional strErrDescription)

Draws a circle, ellipse, or arc on the default Printer object.

Public Sub **PrinterLineAbsolute** (ByVal sngX1, ByVal sngY1, ByVal sngX2, ByVal sngY2, Optional ByVal lngOLEColor, Optional ByVal boolBox, Optional ByVal boolFill, Optional lngErrNumber, Optional strErrDescription)

Draws lines and rectangles on the default Printer object.

Public Function **PrinterObject** (Optional lngErrNumber, Optional strErrDescription)

Enables you to communicate with the default system printer.

Public Sub **PrinterPaintPicture** (ByVal byteStdPictureArray, ByVal sngX1, ByVal sngY1, Optional ByVal sngWidth1, Optional ByVal sngHeight1, Optional ByVal sngX2, Optional ByVal sngY2, Optional ByVal sngWidth2, Optional ByVal sngHeight2, Optional ByVal lngOpcode, Optional lngErrNumber, Optional strErrDescription)

Draws the contents of a graphics file on the default Printer object.

Public Sub **PrinterPrintTabbedText** (ByVal varTabbedInfoArray, Optional ByVal boolSuppresCRLFAfterPrint, Optional lngErrNumber, Optional strErrDescription)

Prints text on the default Printer object; the text to print is passed through the varTabbedInfoArray array with an arbitrary number of rows and two columns: first column represents the position of the text; second column indicates the text to print.

Public Sub **PrinterPrintText** (ByVal strText, Optional lngErrNumber, Optional strErrDescription)

Prints text on the default Printer object.

Public Sub **PrinterPSetAbsolute** (ByVal sngX, ByVal sngY, Optional ByVal lngOLEColor, Optional lngErrNumber, Optional strErrDescription)

Sets a point to a specified color on the default Printer object.

Public Sub **SetPrinterObject** (ByVal strDeviceName, Optional lngErrNumber, Optional strErrDescription)

Changes the particular printer that the Printer object points to.

*See the **Members of the VB.Printer** class paragraph, at the page 142 of this document, for a list of members to which this **Printer object** reference provides access.*

Screen-related public properties and methods

Public Property **ScreenHeight**

Returns the height (in twips) of the screen (read-only property).

Public Property **ScreenMousePointer**

Returns/sets the type of mouse pointer displayed when over part of an object.

Public Property **ScreenTwipsPerPixelX**

Returns the number of twips per pixel for an object measured horizontally (read-only property).

Public Property **ScreenTwipsPerPixelY**

Returns the number of twips per pixel for an object measured vertically (read-only property).

Public Property **ScreenWidth**

Returns the width (in twips) of the screen (read-only property).

Public Sub **ScreenGetMouseIcon** (byteStdPictureArray, Optional lngErrNumber, Optional strErrDescription)

Gets the picture used as a custom mouse icon as a byte array.

Public Sub **ScreenSetMouseIcon** (ByVal byteStdPictureArray, Optional lngErrNumber, Optional strErrDescription)

Sets a custom mouse icon using a picture represented by the byte array.

Public Function **ScreenObject** (Optional lngErrNumber, Optional strErrDescription)

Returns a reference to the Screen object.

*See the **Members of the VB.Screen** class paragraph, at the page 147 of this document, for a list of members to which this **Screen object** reference provides access.*

General public methods

Public Sub **AboutBox**

Displays a window containing information about the Dialogs ActiveX EXE component.

Public Sub **Beep**

Sounds a tone through the computer's speaker..

Public Sub **BlobToArray** (ByVal varSource, byteArrayDestination, Optional lngErrNumber, Optional strErrDescription)

Extracts, if possible, the content of the varSource binary large object parameter, as an array of bytes, and places it in the byteArrayDestination parameter.

Public Function **BrowseForFolder** (Optional ByVal lnghWnd, Optional ByVal strMessage, Optional ByVal strStartDir, Optional lngErrNumber, Optional strErrDescription)

Creates a dialog box that allows the user to select a folder and then returns the selected folder's name.

Public Sub **CommonDialogShowColor** (lngSelectedColor, Optional ByVal lnghWnd, Optional ByVal boolFullOpen, Optional ByVal boolPreventFullOpen, Optional ByVal boolAnyColor, Optional ByVal lngInitialColor, Optional ByVal lngCustomColorsArray, Optional lngErrNumber, Optional strErrDescription)

Displays the CommonDialog's Color dialog box.

Public Sub **CommonDialogShowFont** (Optional ByVal lnghWnd, Optional ByVal boolUsePrinterFonts, Optional ByVal boolUseScreenFonts, Optional ByVal boolUseFixedPitchOnly, Optional ByVal boolUseNoVectorFonts, Optional ByVal boolUseTrueTypeOnly, Optional ByVal boolUseEffects, Optional ByVal boolUseLogFont, Optional lngFontColor, Optional boolFontItalic, Optional boolFontBold, Optional boolFontStrikethru, Optional boolFontUnderline, Optional strFontName, Optional byteFontSize, Optional lngErrNumber, Optional strErrDescription)

Displays the CommonDialog's Font dialog box.

Public Sub **CommonDialogShowOpen** (ByVal strInitDir, ByVal strFileFilter, ByVal strDialogTitle, strFileName, Optional ByVal lnghWnd, Optional ByVal boolMultiSelect, Optional lngErrNumber, Optional strErrDescription)

Displays the CommonDialog control's Open dialog box.

General public methods (continued)

Public Sub **CommonDialogShowPrinter** (Optional lnghDC, Optional strDeviceName, Optional bytePrintRange, Optional ByVal boolDisablePageNumbers, Optional lngFromPage, Optional lngToPage, Optional ByVal boolDisableSelection, Optional intCopies, Optional ByVal boolShowPrintToFile, Optional ByVal boolDisablePrintToFile, Optional boolPrintToFile, Optional boolCollate, Optional ByVal boolPreventWarning, Optional ByVal lnghWnd, Optional lngErrNumber, Optional strErrDescription)

Displays the CommonDialog's Printer dialog box.

Public Sub **CommonDialogShowSave** (ByVal strInitDir, ByVal strFileFilter, ByVal strDialogTitle, strFileName, Optional ByVal lnghWnd, Optional lngErrNumber, Optional strErrDescription)

Displays the CommonDialog control's Save As dialog box.

Public Sub **GetMyDocumentsFolder**

Get the location of the user's Document folder.

Public Sub **Help**

Displays the properties, methods and events exposed by the Dialog class.

Public Function **InBox** (ByVal strPrompt, Optional ByVal strTitle, Optional ByVal strDefault, Optional ByVal sngXPos, Optional ByVal sngYPos)

Displays a prompt in a dialog box and returns user input text.

Public Function **MessageBox** (ByVal strPrompt, Optional ByVal lngButtons, Optional ByVal strTitle)

Displays a dialog box and returns the value of the user's selection.

Public Sub **ProcessMessages**

Yields execution so that the operating system can process other events.

Public Sub **QBColor** (ByVal intColor, lngQBColor)

Returns the RGB color code for a specified color number from a list of predefined standard colors.

Public Sub **RGBFromOLEColor** (ByVal byteRed, ByVal byteGreen, ByVal byteBlue, lngOLEColor)

Translates a whole number into RGB color values.

Public Sub **RGBToOleColor** (ByVal lngOLEColor, byteRed, byteGreen, byteBlue)

Returns a whole number representing an RGB color value.

JScript-related public properties and methods

Public Function **JS_BlobToArray** (ByVal varSource)

> *Returns – if possible – the content of the varSource binary large object parameter, as an array of bytes (for JScript clients).*

Public Sub **JS_CancelEvent**

> *Signals, in an event handling routine for an event that can be canceled, the cancellation of that event (for JScript clients).*

Public Property **JS_CDSCSelectedColor**

> *Returns the color that has been selected using the CommonDialogShowColor method (for JScript clients).*

Public Property **JS_CDSFSelectedFont**

> *Returns the Font object that has been selected using the CommonDialogShowFont method (for JScript clients).*

Public Property **JS_CDSFSelectedFontColor**

> *Returns the color of the Font object that has been selected using the CommonDialogShowFont method (for JScript clients).*

Public Property **JS_CDSOSelectedFileName**

> *Returns the filename that has been selected using the CommonDialogShowOpen method (for JScript clients).*

Public Property **JS_CDSOSelectedFileNames**

> *Returns an array containing the filenames that have been selected using the CommonDialogShowOpen method, having the MultiSelect parameter set to True (for JScript clients).*

JScript-related public properties and methods (continued)

Public Property **JS_CDSPSelectedPrinter**

Returns a Scripting.Dictionary object, containing some properties of the Printer object that has been selected using the CommonDialogShowPrinter method (for JScript clients). This dictionary object contains the following names (items):

- *lnghDC*
- *strDeviceName*
- *bytePrintRange*
- *lngFromPage*
- *lngToPage*
- *intCopies*
- *boolPrintToFile*
- *boolCollate*
- *lngErrNumber*
- *strErrDescription*

Public Property **JS_CDSSSpecifiedFileName**

Returns the filename that has been specified using the CommonDialogShowSave method (for JScript clients).

Public Function **JS_ClipboardGetData** (Optional ByVal intFormat)

Returns the byte array representing the graphic from the Clipboard object (for JScript clients).

Public Function **JS_GetLastErrDescription**

Returns the error description for the error generated by the last method call or statement execution (for JScript clients).

Public Function **JS_GetLastErrNumber**

Returns the error number for the error generated by the last method call or statement execution (for JScript clients).

JScript-related public properties and methods (continued)

Public Function **JS_QBColor** (ByVal intColor)

Returns the RGB color code for a specified color number from a list of predefined standard colors (for JScript clients).

Public Function **JS_RGBFromOLEColor** (ByVal lngOLEColor)

Returns a Scripting.Dictionary object containing the R, G, B values extracted from the OLEColor parameter (for JScript clients). This dictionary object contains the following names (items):

- *byteRed*
- *byteGreen*
- *byteBlue*

Public Function **JS_RGBToOleColor** (ByVal byteRed, ByVal byteGreen, ByVal byteBlue)

Returns a whole number representing an RGB color value (for JScript clients).

Public Function **JS_ScreenGetMouseIcon**

Returns the picture used as a custom mouse icon as a byte array (for JScript clients).

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method

A window or dialog box that makes up part of an application's user interface.

Event **Activate**

*Occurs when a form becomes the active window. Available as the Dialog's **FormActivate** event.*

Property **ActiveControl** As Control

Returns the control that has focus.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **AutoRedraw** As Boolean

Returns/sets the output from a graphics method to a persistent bitmap.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **Caption** As String

Returns/sets the text displayed in an object's title bar or below an object's icon.

Sub **Circle** (ByVal Step As Integer, ByVal X As Single, ByVal Y As Single, ByVal Radius As Single, ByVal Color As Long, ByVal Start As Single, ByVal End As Single, ByVal Aspect As Single)

*Draws a circle, ellipse, or arc on an object. Don't use it; instead, use the Dialog.**FormCircleAbsolute** method.*

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **FormClick** event.*

Property **ClipControls** As Boolean

Determines whether graphics methods in Paint events repaint an entire object or newly exposed areas.

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Sub **Cls**

Clears graphics and text generated at run time from a Form, Image, or PictureBox.

Property **ControlBox** As Boolean

Returns a value indicating whether a Control-menu box is displayed on a form at run time.

Property **Controls** As Object

A collection whose elements represent each control on a form, including elements of control arrays.

Property **Count** As Integer

Returns the number of objects in a collection.

Property **CurrentX** As Single

Returns/sets the horizontal coordinates for next print or draw method.

Property **CurrentY** As Single

Returns/sets the vertical coordinates for next print or draw method.

Event **DblClick**

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's ***FormDblClick*** *event.*

Event **Deactivate**

Occurs when a form is no longer the active window. Available as the Dialog's ***FormDeactivate*** *event.*

Property **DrawMode** As Integer

Sets the appearance of output from graphics methods or of a Shape or Line control.

Property **DrawStyle** As Integer

Determines the line style for output from graphics methods.

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Property **DrawWidth** As Integer

Returns/sets the line width for output from graphics methods.

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **FillColor** As Long

Returns/sets the color used to fill in shapes, circles, and boxes.

Property **FillStyle** As Integer

Returns/sets the fill style of a shape.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontTransparent** As Boolean

Returns/sets a value that determines whether background text/graphics on a Form, Printer or PictureBox are displayed.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog’s **FormGotFocus** event.*

Property **HasDC** As Boolean

Determines whether a unique display context is allocated for the control.

Property **hDC** As Long

Returns a handle (from Microsoft Windows) to the object's device context.

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Sub **Hide**

Hides an MDIForm or Form object but doesn't unload it.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Icon** As stdole.Picture

Returns the icon displayed when a form is minimized at run time. Don’t use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **Image** As stdole.Picture

Returns a handle, provided by Microsoft Windows, to a persistent bitmap. Don’t use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **Initialize**

*Occurs when an application creates an instance of a Form, MDIForm, or class. Available as the Dialog’s **FormInitialize** event.*

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **FormKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **FormKeyPress** event.*

Property **KeyPreview** As Boolean

Returns/sets whether keyboard events for an object are invoked before keyboard events for controls on that object.

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **FormKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Sub **Line** (ByVal Flags As Integer, ByVal X1 As Single, ByVal Y1 As Single, ByVal X2 As Single, ByVal Y2 As Single, ByVal Color As Long)

*Draws lines and rectangles on an object. Don't use it; instead, use the Dialog. **FormLineAbsolute** method.*

Event **Load**

*Occurs when a form is loaded. Available as the Dialog's **FormLoad** event.*

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **FormLostFocus** event.*

Property **MaxButton** As Boolean

Determines whether a form has a Maximize button.

Property **MDIChild** As Boolean

Returns/sets a value that determines if a form is displayed as an MDI child form.

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Property **MinButton** As Boolean

Determines whether a form has a Minimize button.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's ***FormMouseDown*** *event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user moves the mouse. Available as the Dialog's ***FormMouseMove*** *event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's ***FormMouseUp*** *event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Moveable** As Boolean

Returns/sets a value that determines whether a form can be moved.

Property **Name** As String

Returns the name used in code to identify an object.

Event **Paint**

Occurs when any part of a form or PictureBox control is moved, enlarged, or exposed. Available as the Dialog's ***FormPaint*** *event.*

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Sub **PaintPicture** (ByVal Picture As stdole.Picture, ByVal X1 As Single, ByVal Y1 As Single, [ByVal Width1], [ByVal Height1], [ByVal X2], [ByVal Y2], [ByVal Width2], [ByVal Height2], [ByVal Opcode])

Draws the contents of a graphics file on a Form, PictureBox, or Printer object. Don't use it; instead, use the Dialog. ***FormPaintPicture*** *method.*

Property **Palette** As stdole.Picture

Returns/sets an image that contains the palette to use on an object when PaletteMode is set to Custom. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **PaletteMode** As Integer

Returns/sets a value that determines which palette to use for the controls on a object.

Property **Picture** As stdole.Picture

Returns/sets a graphic to be displayed in a control. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Function **Point** (ByVal X As Single, ByVal Y As Single) As Long

Returns, as an integer of type Long, the RGB color of the specified point on a Form or PictureBox object.

Sub **PrintForm**

Sends a bit-by-bit image of a Form object to the printer.

Sub **PSet** (ByVal Step As Integer, ByVal X As Single, ByVal Y As Single, ByVal Color As Long)

*Sets a point on an object to a specified color. Don't use it; instead, use the Dialog.****FormPSetAbsolute*** *method.*

Event **QueryUnload** (Cancel As Integer, UnloadMode As Integer)

Occurs before a form or application closes. Available as the Dialog's ***FormQueryUnload*** *event.*

Sub **Refresh**

Forces a complete repaint of a object.

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Event **Resize**

*Occurs when a form is first displayed or the size of an object changes. Available as the Dialog's **FormResize** event.*

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **Scale** (ByVal Flags As Integer, [ByVal X1], [ByVal Y1], [ByVal X2], [ByVal Y2])

Defines the coordinate system for a Form, PictureBox, or Printer.

Property **ScaleHeight** As Single

Returns/sets the number of units for the vertical measurement of an object's interior.

Property **ScaleLeft** As Single

Returns/sets the horizontal coordinates for the left edges of an object.

Property **ScaleMode** As Integer

Returns/sets a value indicating measurement units for object coordinates when using graphics methods or positioning controls.

Property **ScaleTop** As Single

Returns/sets the vertical coordinates for the top edges of an object.

Property **ScaleWidth** As Single

Returns/sets the number of units for the horizontal measurement of an object's interior.

Function **ScaleX** (ByVal Width As Single, [ByVal FromScale], [ByVal ToScale]) As Single

Converts the value for the width of a Form, PictureBox, or Printer from one unit of measure to another.

Function **ScaleY** (ByVal Height As Single, [ByVal FromScale], [ByVal ToScale]) As Single

Converts the value for the height of a Form, PictureBox, or Printer from one unit of measure to another.

Sub **SetFocus**

Moves the focus to the specified object.

Members of the VB.**Form** class, made available through the object reference returned by the Dialog.**FormObject** method (continued)

Sub **Show** ([ByVal Modal], [ByVal OwnerForm])

*Displays an MDIForm or Form object. Don't use it; instead, use the Dialog.**ShowModal** method.*

Property **ShowInTaskbar** As Boolean

Determines whether a Form or MDIForm object appears in the Windows taskbar.

Property **StartUpPosition** As Integer

Returns or sets a value specifying the position of a Form when it first appears.

Property **Tag** As String

Stores any extra data needed for your program.

Event **Terminate**

*Occurs when all references to an instance of a Form, MDIForm, or class are removed from memory. Available as the Dialog's **FormTerminate** event.*

Function **TextHeight** (ByVal Str As String) As Single

Returns the height of a text string as it would be printed in the current font.

Function **TextWidth** (ByVal Str As String) As Single

Returns the width of a text string as it would be printed in the current font.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Event **Unload** (Cancel As Integer)

Occurs when a form is about to be removed from the screen.

Sub **ValidateControls**

Validate contents of the last control on the form before exiting the form

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisButton** As Boolean

Returns or sets whether the What's This button appears in the title bar of a Form or MDIForm.

Members of the VB.Form class, made available through the object reference returned by the Dialog.FormObject method (continued)

Property **WhatsThisHelp** As Boolean

Sets or returns whether context-sensitive Help uses the What's This popup provided by Windows Help or the main Help window.

Sub **WhatsThisMode**

Prepares the application to display What's This Help on a selected object.

Property **Width** As Single

Returns/sets the width of an object.

Property **WindowState** As Integer

Returns/sets the visual state of a form window at run time.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**CheckBox** class, made available through the object reference returned by the Dialog.**CheckObject** method

Displays an X when selected; the X disappears when the CheckBox is cleared.

Property **Alignment** As Integer

Returns/sets the alignment of a CheckBox or OptionButton, or a control's text.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **Caption** As String

Returns/sets the text displayed in an object's title bar or below an object's icon.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **CheckClick** event.*

Property **Container** As Object

Returns the container of an object.

Property **DisabledPicture** As stdole.Picture

Returns/sets a graphic to be displayed when the button is disabled, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **DownPicture** As stdole.Picture

Returns/sets a graphic to be displayed when the button is in the down position, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Members of the VB.**CheckBox** class, made available through the object reference returned by the Dialog.**CheckObject** method (continued)

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **CheckGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Members of the VB.**CheckBox** class, made available through the object reference returned by the Dialog.**CheckObject** method (continued)

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **CheckKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **CheckKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **CheckKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **CheckLostFocus** event.*

Property **MaskColor** As Long

Returns or sets a color in a button's picture to be a 'mask' (that is, transparent), if Style is set to 1.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **CheckMouseDown** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **CheckMouseMove** event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Members of the VB.**CheckBox** class, made available through the object reference returned by the Dialog.**CheckObject** method (continued)

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's ***CheckMouseUp*** *event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Property **Picture** As stdole.Picture

Returns/sets a graphic to be displayed in a CommandButton, OptionButton or CheckBox control, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **Style** As Integer

Returns/sets the appearance of the control, whether standard (standard Windows style) or graphical (with a custom picture).

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Members of the VB.**CheckBox** class, made available through the object reference returned by the Dialog.**CheckObject** method (continued)

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **UseMaskColor** As Boolean

Returns or sets a value that determines whether the color assigned in the MaskColor property is used as a 'mask'. (That is, used to create transparent regions.) Applies only if Style is set to 1.

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **CheckValidate** event.*

Property **Value** As Integer

Returns/sets the value of an object.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**ComboBox** class, made available through the object reference returned by the Dialog.**ComboObject** or the Dialog.**ComboTextObject** method

Combines the features of a TextBox control and a ListBox control.

Sub **AddItem** (ByVal Item As String, [ByVal Index])

Adds an item to a Listbox or ComboBox control or a row to a Grid control.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Change**

*Occurs when the contents of a control have changed. Available as the Dialog's **ComboChange** event.*

Sub **Clear**

Clears the contents of a control or the system Clipboard.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **ComboClick** event.*

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **ComboDblClick** event.*

Event **DropDown**

*Occurs when the list portion of a ComboBox control is about to drop down. Available as the Dialog's **ComboDropDown** event.*

Members of the VB.**ComboBox** class, made available through the object reference returned by the Dialog.**ComboObject** or the Dialog.**ComboTextObject** method (continued)

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **ComboGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Members of the VB.**ComboBox** class, made available through the object reference returned by the Dialog.**ComboObject** or the Dialog.**ComboTextObject** method (continued)

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **IntegralHeight** As Boolean

Returns/Sets a value indicating whether the control displays partial items.

Property **ItemData** (ByVal As Integer) As Long

Returns/sets a specific number for each item in a ComboBox or ListBox control.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **ComboKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **ComboKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **ComboKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Property **List** (ByVal As Integer) As String

Returns/sets the items contained in a control's list portion.

Property **ListCount** As Integer

Returns the number of items in the list portion of a control.

Property **ListIndex** As Integer

Returns/sets the index of the currently selected item in the control.

Members of the VB.**ComboBox** class, made available through the object reference returned by the Dialog.**ComboObject** or the Dialog.**ComboTextObject** method (continued)

Property **Locked** As Boolean

Determines whether a control can be edited.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **ComboLostFocus** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon.

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **NewIndex** As Integer

Returns the index of the item most recently added to a control.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Sub **RemoveItem** (ByVal Index As Integer)

Removes an item from a ListBox or ComboBox control or a row from a Grid control.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Event **Scroll**

*Occurs when you reposition the scroll box on a control. Available as the Dialog's **ComboScroll** event.*

Members of the VB.**ComboBox** class, made available through the object reference returned by the Dialog.**ComboObject** or the Dialog.**ComboTextObject** method (continued)

Property **SelLength** As Long

Returns/sets the number of characters selected.

Property **SelStart** As Long

Returns/sets the starting point of text selected.

Property **SelText** As String

Returns/sets the string containing the currently selected text.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **Sorted** As Boolean

Indicates whether the elements of a control are automatically sorted alphabetically.

Property **Style** As Integer

Returns/sets a value that determines the type of control and the behavior of its list box portion.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **Text** As String

Returns/sets the text contained in the control.

Members of the VB.ComboBox class, made available through the object reference returned by the Dialog.ComboObject or the Dialog.ComboTextObject method (continued)

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **TopIndex** As Integer

Returns/sets which item in a control is displayed in the topmost position.

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **ComboValidate** event.*

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**CommandButton** class, made available through the object reference returned by the Dialog.**CommandObject** method

Looks like a push button and is used to begin, interrupt, or end a process.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **Cancel** As Boolean

Indicates whether a command button is the Cancel button on a form.

Property **Caption** As String

Returns/sets the text displayed in an object's title bar or below an object's icon.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **CommandClick** event.*

Property **Container** As Object

Returns the container of an object.

Property **Default** As Boolean

Determines which CommandButton control is the default command button on a form.

Property **DisabledPicture** As stdole.Picture

Returns/sets a graphic to be displayed when the button is disabled, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **DownPicture** As stdole.Picture

Returns/sets a graphic to be displayed when the button is in the down position, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Members of the VB.**CommandButton** class, made available through the object reference returned by the Dialog.**CommandObject** method (continued)

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **CommandGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Members of the VB.**CommandButton** class, made available through the object reference returned by the Dialog.**CommandObject** method (continued)

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **CommandKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **CommandKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **CommandKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **CommandLostFocus** event.*

Property **MaskColor** As Long

Returns or sets a color in a button's picture to be a 'mask' (that is, transparent), if Style is set to 1.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **CommandMouseDown** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **CommandMouseMove** event.*

Members of the VB.**CommandButton** class, made available through the object reference returned by the Dialog.**CommandObject** method (continued)

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's **CommandMouseUp** event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Property **Picture** As stdole.Picture

Returns/sets a graphic to be displayed in a CommandButton, OptionButton or CheckBox control, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **Style** As Integer

Returns/sets the appearance of the control, whether standard (standard Windows style) or graphical (with a custom picture).

Members of the VB.**CommandButton** class, made available through the object reference returned by the Dialog.**CommandObject** method (continued)

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **UseMaskColor** As Boolean

Returns or sets a value that determines whether the color assigned in the MaskColor property is used as a 'mask'. (That is, used to create transparent regions.) Applies only if Style is set to 1.

Property **Value** As Boolean

Returns/sets the value of an object.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**Frame** class, made available through the object reference returned by the Dialog.**FrameObject** method

Provides an identifiable grouping for controls.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **Caption** As String

Returns/sets the text displayed in an object's title bar or below an object's icon.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **FrameClick** event.*

Property **ClipControls** As Boolean

Determines whether graphics methods in Paint events repaint an entire object or newly exposed areas.

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **FrameDblClick** event.*

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Members of the VB.**Frame** class, made available through the object reference returned by the Dialog.**FrameObject** method (continued)

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **FrameMouseDown** event.*

Members of the VB.**Frame** class, made available through the object reference returned by the Dialog.**FrameObject** method (continued)

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don’t use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user moves the mouse. Available as the Dialog’s ***FrameMouseMove*** *event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog’s ***FrameMouseUp*** *event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Members of the VB.**Frame** class, made available through the object reference returned by the Dialog.**FrameObject** method (continued)

Property **Tag** As String

Stores any extra data needed for your program.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.HScrollBar class, made available through the object reference returned by the Dialog.HScrollBarObject method

Provides a horizontal scroll bar for easy navigation through long lists of items.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Change**

*Occurs when the contents of a control have changed. Available as the Dialog's **HScrollBarChange** event.*

Property **Container** As Object

Returns the container of an object.

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **HScrollBarGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **HScrollBarKeyDown** event.*

Members of the VB.HScrollBar class, made available through the object reference returned by the Dialog.HScrollBarObject method (continued)

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **HScrollBarKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **HScrollBarKeyUp** event.*

Property **LargeChange** As Integer

Returns/sets amount of change to Value property in a scroll bar when user clicks the scroll bar area.

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **HScrollBarLostFocus** event.*

Property **Max** As Integer

Returns/sets a scroll bar position's maximum Value property setting.

Property **Min** As Integer

Returns/sets a scroll bar position's maximum Value property setting.

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Members of the VB.HScrollBar class, made available through the object reference returned by the Dialog.HScrollBarObject method (continued)

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Event **Scroll**

*Occurs when you reposition the scroll box on a control. Available as the Dialog's **HScrollBarScroll** event.*

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **SmallChange** As Integer

Returns/sets amount of change to Value property in a scroll bar when user clicks a scroll arrow.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Members of the VB.**HScrollBar** class, made available through the object reference returned by the Dialog.**HScrollBarObject** method (continued)

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **HScrollBarValidate** event.*

Property **Value** As Integer

Returns/sets the value of an object.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**Image** class, made available through the object reference returned by the Dialog.**ImageObject** method

Displays a graphic.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **ImageClick** event.*

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **ImageDblClick** event.*

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **Height** As Single

Returns/sets the height of an object.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **ImageMouseDown** event.*

Members of the VB.**Image** class, made available through the object reference returned by the Dialog.**ImageObject** method (continued)

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **ImageMouseMove** event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's **ImageMouseUp** event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Property **Picture** As stdole.Picture

Returns/sets a graphic to be displayed in a control. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Sub **Refresh**

Forces a complete repaint of a object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Members of the VB.**Image** class, made available through the object reference returned by the Dialog.**ImageObject** method (continued)

Property **Stretch** As Boolean

Returns/sets a value that determines whether a graphic resizes to fit the size of an Image control.

Property **Tag** As String

Stores any extra data needed for your program.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**Label** class, made available through the object reference returned by the Dialog.**LabelObject** method

Displays text that a user can't change directly.

Property **Alignment** As Integer

Returns/sets the alignment of a CheckBox or OptionButton, or a control's text.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **AutoSize** As Boolean

Determines whether a control is automatically resized to display its entire contents.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **BackStyle** As Integer

Indicates whether a Label or the background of a Shape is transparent or opaque.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **Caption** As String

Returns/sets the text displayed in an object's title bar or below an object's icon.

Event **Change**

*Occurs when the contents of a control have changed. Available as the Dialog's **LabelChange** event.*

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **LabelClick** event.*

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **LabelDblClick** event.*

Members of the VB.**Label** class, made available through the object reference returned by the Dialog.**LabelObject** method (continued)

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Property **Height** As Single

Returns/sets the height of an object.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Members of the VB.**Label** class, made available through the object reference returned by the Dialog.**LabelObject** method (continued)

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **LabelMouseDown** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **LabelMouseMove** event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's **LabelMouseUp** event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Members of the VB.**Label** class, made available through the object reference returned by the Dialog.**LabelObject** method (continued)

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **Tag** As String

Stores any extra data needed for your program.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **UseMnemonic** As Boolean

Returns/sets a value that specifies whether an & in a Label's Caption property defines an access key.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Property **WordWrap** As Boolean

Returns/sets a value that determines whether a control expands to fit the text in its Caption.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**Line** class, made available through the object reference returned by the Dialog.**LineObject** method

A graphical control displayed as a horizontal, vertical or diagonal line.

Property **BorderColor** As Long

Returns/sets the color of an object's border.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **BorderWidth** As Integer

Returns or sets the width of a control's border.

Property **Container** As Object

Returns the container of an object.

Property **DrawMode** As Integer

Sets the appearance of output from graphics methods or of a Shape or Line control.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **X1** As Single

Returns/sets the X coordinate of the starting point of a Line control.

Members of the VB.**Line** class, made available through the object reference returned by the Dialog.**LineObject** method (continued)

Property **X2** As Single

Returns/sets the X coordinate of the ending point of a Line control.

Property **Y1** As Single

Returns/sets the Y coordinate of the starting point of a Line control.

Property **Y2** As Single

Returns/sets the Y coordinate of the ending point of a Line control.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**ListBox** class, made available through the object reference returned by the Dialog.**ListCheckObject** or the Dialog.**ListObject** method

Displays a list of items from which the user can select one or more.

Sub **AddItem** (ByVal Item As String, [ByVal Index])

Adds an item to a Listbox or ComboBox control or a row to a Grid control.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Sub **Clear**

Clears the contents of a control or the system Clipboard.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **ListCheckClick** or **ListClick** event.*

Property **Columns** As Integer

Returns/sets a value that determines whether a ListBox scrolls vertically in a single column (value of 0) or horizontally in snaking columns (values greater than 0).

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **ListCheckDblClick** or **ListDblClick** event.*

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Members of the VB.**ListBox** class, made available through the object reference returned by the Dialog.**ListCheckObject** or the Dialog.**ListObject** method (continued)

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **ListCheckGotFocus** or **ListGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Members of the VB.**ListBox** class, made available through the object reference returned by the Dialog.**ListCheckObject** or the Dialog.**ListObject** method (continued)

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **IntegralHeight** As Boolean

Returns/Sets a value indicating whether the control displays partial items.

Event **ItemCheck** (Item As Integer)

*Occurs when a ListBox control's Style property is set to 1 (checkboxes) and an item's checkbox in the ListBox control is selected or cleared. Available as the Dialog's **ListCheckItemCheck** event.*

Property **ItemData** (ByVal As Integer) As Long

Returns/sets a specific number for each item in a ComboBox or ListBox control.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **ListCheckKeyDown** or **ListKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **ListCheckKeyPress** or **ListKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **ListCheckKeyUp** or **ListKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Property **List** (ByVal As Integer) As String

Returns/sets the items contained in a control's list portion.

Property **ListCount** As Integer

Returns the number of items in the list portion of a control.

Property **ListIndex** As Integer

Returns/sets the index of the currently selected item in the control.

Members of the VB.**ListBox** class, made available through the object reference returned by the Dialog.**ListCheckObject** or the Dialog.**ListObject** method (continued)

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **ListCheckLostFocus** or **ListLostFocus** event.*

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **ListCheckMouseDown** or **ListMouseDown** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **ListCheckMouseMove** or **ListMouseMove** event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's **ListCheckMouseUp** or **ListMouseUp** event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **MultiSelect** As Integer

Returns/sets a value that determines whether a user can make multiple selections in a control.

Property **Name** As String

Returns the name used in code to identify an object.

Property **NewIndex** As Integer

Returns the index of the item most recently added to a control.

Members of the VB.ListBox class, made available through the object reference returned by the Dialog.ListCheckObject or the Dialog.ListObject method (continued)

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Sub **RemoveItem** (ByVal Index As Integer)

Removes an item from a ListBox or ComboBox control or a row from a Grid control.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Event **Scroll**

*Occurs when you reposition the scroll box on a control. Available as the Dialog's **ListCheckScroll** or **ListScroll** event.*

Property **SelCount** As Integer

Returns the number of selected items in a ListBox control.

Property **Selected** (ByVal As Integer) As Boolean

Returns/sets the selection status of an item in a control.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **Sorted** As Boolean

Indicates whether the elements of a control are automatically sorted alphabetically.

Property **Style** As Integer

Returns/sets a value that determines whether checkboxes are displayed inside a ListBox control.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Members of the VB.**ListBox** class, made available through the object reference returned by the Dialog.**ListCheckObject** or the Dialog.**ListObject** method (continued)

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **Text** As String

Returns/sets the text contained in the control.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **TopIndex** As Integer

Returns/sets which item in a control is displayed in the topmost position.

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **ListCheckValidate** or **ListValidate** event.*

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**TextBox** class, made available through the object reference returned by the Dialog.**MultiTextObject** or the Dialog.**TextObject** method

Displays information entered at design time by the user, or in code at run time.

Property **Alignment** As Integer

Returns/sets the alignment of a CheckBox or OptionButton, or a control's text.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Change**

*Occurs when the contents of a control have changed. Available as the Dialog's **MultiTextChange** or **TextChange** event.*

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **MultiTextClick** or **TextClick** event.*

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **MultiTextDblClick** or **TextDblClick** event.*

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Members of the VB.**TextBox** class, made available through the object reference returned by the Dialog.**MultiTextObject** or the Dialog.**TextObject** method (continued)

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **MultiTextGotFocus** or **TextGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **HideSelection** As Boolean

Specifies whether the selection in a Masked edit control is hidden when the control loses focus.

Members of the VB.**TextBox** class, made available through the object reference returned by the Dialog.**MultiTextObject** or the Dialog.**TextObject** method (continued)

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **MultiTextKeyDown** or **TextKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **MultiTextKeyPress** or **TextKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **MultiTextKeyUp** or **TextKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Property **Locked** As Boolean

Determines whether a control can be edited.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **MultiTextLostFocus** or **TextLostFocus** event.*

Property **MaxLength** As Long

Returns/sets the maximum number of characters that can be entered in a control.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **MultiTextMouseDown** or **TextMouseDown** event.*

Members of the VB.**TextBox** class, made available through the object reference returned by the Dialog.**MultiTextObject** or the Dialog.**TextObject** method (continued)

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **MultiTextMouseMove** or **TextMouseMove** event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's **MultiTextMouseUp** or **TextMouseUp** event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **MultiLine** As Boolean

Returns/sets a value that determines whether a control can accept multiple lines of text.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Property **PasswordChar** As String

Returns/sets a value that determines whether characters typed by a user or placeholder characters are displayed in a control.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Members of the VB.**TextBox** class, made available through the object reference returned by the Dialog.**MultiTextObject** or the Dialog.**TextObject** method (continued)

Property **ScrollBars** As Integer

Returns/sets a value indicating whether an object has vertical or horizontal scroll bars.

Property **SelLength** As Long

Returns/sets the number of characters selected.

Property **SelStart** As Long

Returns/sets the starting point of text selected.

Property **SelText** As String

Returns/sets the string containing the currently selected text.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **Text** As String

Returns/sets the text contained in the control.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Members of the VB.**TextBox** class, made available through the object reference returned by the Dialog.**MultiTextObject** or the Dialog.**TextObject** method (continued)

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **MultiTextValidate** or **TextValidate** event.*

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**OptionButton** class, made available through the object reference returned by the Dialog.**OptionObject** method

Displays an option that can be turned on or off.

Property **Alignment** As Integer

Returns/sets the alignment of a CheckBox or OptionButton, or a control's text.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **Caption** As String

Returns/sets the text displayed in an object's title bar or below an object's icon.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **OptionClick** event.*

Property **Container** As Object

Returns the container of an object.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **OptionDblClick** event.*

Property **DisabledPicture** As stdole.Picture

Returns/sets a graphic to be displayed when the button is disabled, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **DownPicture** As stdole.Picture

Returns/sets a graphic to be displayed when the button is in the down position, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Members of the VB.**OptionButton** class, made available through the object reference returned by the Dialog.**OptionObject** method (continued)

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog’s **OptionGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Members of the VB.**OptionButton** class, made available through the object reference returned by the Dialog.**OptionObject** method (continued)

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **OptionKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **OptionKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **OptionKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **OptionLostFocus** event.*

Property **MaskColor** As Long

Returns or sets a color in a button's picture to be a 'mask' (that is, transparent), if Style is set to 1.

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **OptionMouseDown** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Members of the VB.**OptionButton** class, made available through the object reference returned by the Dialog.**OptionObject** method (continued)

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user moves the mouse. Available as the Dialog's ***OptionMouseMove*** *event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's ***OptionMouseUp*** *event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Property **Picture** As stdole.Picture

Returns/sets a graphic to be displayed in a CommandButton, OptionButton or CheckBox control, if Style is set to 1. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Members of the VB.**OptionButton** class, made available through the object reference returned by the Dialog.**OptionObject** method (continued)

Property **Style** As Integer

Returns/sets the appearance of the control, whether standard (standard Windows style) or graphical (with a custom picture).

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **UseMaskColor** As Boolean

Returns or sets a value that determines whether the color assigned in the MaskColor property is used as a 'mask'. (That is, used to create transparent regions.) Applies only if Style is set to 1.

Event **Validate** (Cancel As Boolean)

Occurs when a control loses focus to a control that causes validation. Available as the Dialog's ***OptionValidate*** *event.*

Property **Value** As Boolean

Returns/sets the value of an object.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Members of the VB.OptionButton class, made available through the object reference returned by the Dialog.OptionObject method (continued)

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method

Displays a graphic from a bitmap, icon or metafile.

Property **Align** As Integer

Returns/sets a value that determines where an object is displayed on a form.

Property **Appearance** As Integer

Returns/sets whether or not an object is painted at run time with 3-D effects.

Property **AutoRedraw** As Boolean

Returns/sets the output from a graphics method to a persistent bitmap.

Property **AutoSize** As Boolean

Determines whether a control is automatically resized to display its entire contents.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Change**

*Occurs when the contents of a control have changed. Available as the Dialog's **PictureChange** event.*

Sub **Circle** (ByVal Step As Integer, ByVal X As Single, ByVal Y As Single, ByVal Radius As Single, ByVal Color As Long, ByVal Start As Single, ByVal End As Single, ByVal Aspect As Single)

*Draws a circle, ellipse, or arc on an object. Don't use it; instead, use the Dialog.**PictureCtrlCircleAbsolute** method.*

Event **Click**

*Occurs when the user presses and then releases a mouse button over an object. Available as the Dialog's **PictureClick** event.*

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method (continued)

Property **ClipControls** As Boolean

Determines whether graphics methods in Paint events repaint an entire object or newly exposed areas.

Sub **Cls**

Clears graphics and text generated at run time from a Form, Image, or PictureBox.

Property **Container** As Object

Returns the container of an object.

Property **CurrentX** As Single

Returns/sets the horizontal coordinates for next print or draw method.

Property **CurrentY** As Single

Returns/sets the vertical coordinates for next print or draw method.

Event **DblClick**

*Occurs when the user presses and releases a mouse button and then presses and releases it again over an object. Available as the Dialog's **PictureDblClick** event.*

Property **DrawMode** As Integer

Sets the appearance of output from graphics methods or of a Shape or Line control.

Property **DrawStyle** As Integer

Determines the line style for output from graphics methods.

Property **DrawWidth** As Integer

Returns/sets the line width for output from graphics methods.

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Property **FillColor** As Long

Returns/sets the color used to fill in shapes, circles, and boxes.

Property **FillStyle** As Integer

Returns/sets the fill style of a shape.

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method (continued)

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontTransparent** As Boolean

Returns/sets a value that determines whether background text/graphics on a Form, Printer or PictureBox are displayed.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Event **GotFocus**

Occurs when an object receives the focus. Available as the Dialog's ***PictureGotFocus*** *event.*

Property **HasDC** As Boolean

Determines whether a unique display context is allocated for the control.

Property **hDC** As Long

Returns a handle (from Microsoft Windows) to the object's device context.

Members of the VB.PictureBox class, made available through the object reference returned by the Dialog.PictureCtrlObject method (continued)

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Image** As stdole.Picture

Returns a handle, provided by Microsoft Windows, to a persistent bitmap. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **PictureKeyDown** event.*

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **PictureKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **PictureKeyUp** event.*

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Sub **Line** (ByVal Flags As Integer, ByVal X1 As Single, ByVal Y1 As Single, ByVal X2 As Single, ByVal Y2 As Single, ByVal Color As Long)

*Draws lines and rectangles on an object. Don't use it; instead, use the Dialog.**PictureCtrlLineAbsolute** method.*

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method (continued)

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **PictureLostFocus** event.*

Event **MouseDown** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user presses the mouse button while an object has the focus. Available as the Dialog's **PictureMouseDown** event.*

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Event **MouseMove** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user moves the mouse. Available as the Dialog's **PictureMouseMove** event.*

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Event **MouseUp** (Button As Integer, Shift As Integer, X As Single, Y As Single)

*Occurs when the user releases the mouse button while an object has the focus. Available as the Dialog's **PictureMouseUp** event.*

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Event **Paint**

*Occurs when any part of a form or PictureBox control is moved, enlarged, or exposed. Available as the Dialog's **PicturePaint** event.*

Sub **PaintPicture** (ByVal Picture As stdole.Picture, ByVal X1 As Single, ByVal Y1 As Single, [ByVal Width1], [ByVal Height1], [ByVal X2], [ByVal Y2], [ByVal Width2], [ByVal Height2], [ByVal Opcode])

*Draws the contents of a graphics file on a Form, PictureBox, or Printer object. Don't use it; instead, use the Dialog.**PictureCtrlPaintPicture** method.*

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method (continued)

Property **Parent** As Form

Returns the object on which this object is located.

Property **Picture** As stdole.Picture

Returns/sets a graphic to be displayed in a control. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Function **Point** (ByVal X As Single, ByVal Y As Single) As Long

Returns, as an integer of type Long, the RGB color of the specified point on a Form or PictureBox object.

Sub **PSet** (ByVal Step As Integer, ByVal X As Single, ByVal Y As Single, ByVal Color As Long)

*Sets a point on an object to a specified color. Don't use it; instead, use the Dialog.**PictureCtrlPSetAbsolute** method.*

Sub **Refresh**

Forces a complete repaint of a object.

Event **Resize**

*Occurs when a form is first displayed or the size of an object changes. Available as the Dialog's **PictureResize** event.*

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **Scale** (ByVal Flags As Integer, [ByVal X1], [ByVal Y1], [ByVal X2], [ByVal Y2])

Defines the coordinate system for a Form, PictureBox, or Printer.

Property **ScaleHeight** As Single

Returns/sets the number of units for the vertical measurement of an object's interior.

Property **ScaleLeft** As Single

Returns/sets the horizontal coordinates for the left edges of an object.

Property **ScaleMode** As Integer

Returns/sets a value indicating measurement units for object coordinates when using graphics methods or positioning controls.

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method (continued)

Property **ScaleTop** As Single

Returns/sets the vertical coordinates for the top edges of an object.

Property **ScaleWidth** As Single

Returns/sets the number of units for the horizontal measurement of an object's interior.

Function **ScaleX** (ByVal Width As Single, [ByVal FromScale], [ByVal ToScale]) As Single

Converts the value for the width of a Form, PictureBox, or Printer from one unit of measure to another.

Function **ScaleY** (ByVal Height As Single, [ByVal FromScale], [ByVal ToScale]) As Single

Converts the value for the height of a Form, PictureBox, or Printer from one unit of measure to another.

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Function **TextHeight** (ByVal Str As String) As Single

Returns the height of a text string as it would be printed in the current font.

Function **TextWidth** (ByVal Str As String) As Single

Returns the width of a text string as it would be printed in the current font.

Property **ToolTipText** As String

Returns/sets the text displayed when the mouse is paused over the control.

Members of the VB.**PictureBox** class, made available through the object reference returned by the Dialog.**PictureCtrlObject** method (continued)

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **PictureValidate** event.*

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**Shape** class, made available through the object reference returned by the Dialog.**ShapeObject** method

A graphical control displayed as a rectangle, square, oval, circle or rounded rectangle or square.

Property **BackColor** As Long

Returns/sets the background color used to display text and graphics in an object.

Property **BackStyle** As Integer

Indicates whether a Label or the background of a Shape is transparent or opaque.

Property **BorderColor** As Long

Returns/sets the color of an object's border.

Property **BorderStyle** As Integer

Returns/sets the border style for an object.

Property **BorderWidth** As Integer

Returns or sets the width of a control's border.

Property **Container** As Object

Returns the container of an object.

Property **DrawMode** As Integer

Sets the appearance of output from graphics methods or of a Shape or Line control.

Property **FillColor** As Long

Returns/sets the color used to fill in shapes, circles, and boxes.

Property **FillStyle** As Integer

Returns/sets the fill style of a shape.

Property **Height** As Single

Returns/sets the height of an object.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Members of the VB.**Shape** class, made available through the object reference returned by the Dialog.**ShapeObject** method (continued)

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Property **Shape** As Integer

Returns/sets a value indicating the appearance of a control.

Property **Tag** As String

Stores any extra data needed for your program.

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**VScrollBar** class, made available through the object reference returned by the Dialog.**VScrollBarObject** method

Provides a vertical scroll bar.

Property **CausesValidation** As Boolean

Returns/sets whether validation occurs on the control which lost focus.

Event **Change**

*Occurs when the contents of a control have changed. Available as the Dialog's **VScrollBarChange** event.*

Property **Container** As Object

Returns the container of an object.

Property **Enabled** As Boolean

Returns/sets a value that determines whether an object can respond to user-generated events.

Event **GotFocus**

*Occurs when an object receives the focus. Available as the Dialog's **VScrollBarGotFocus** event.*

Property **Height** As Single

Returns/sets the height of an object.

Property **HelpContextID** As Long

Specifies the default Help file context ID for an object.

Property **hWnd** As Long

Returns a handle (from Microsoft Windows) to an object's window.

Property **Index** As Integer

Returns/sets the number identifying a control in a control array.

Event **KeyDown** (KeyCode As Integer, Shift As Integer)

*Occurs when the user presses a key while an object has the focus. Available as the Dialog's **VScrollBarKeyDown** event.*

Members of the VB.**VScrollBar** class, made available through the object reference returned by the Dialog.**VScrollBarObject** method (continued)

Event **KeyPress** (KeyAscii As Integer)

*Occurs when the user presses and releases an ANSI key. Available as the Dialog's **VScrollBarKeyPress** event.*

Event **KeyUp** (KeyCode As Integer, Shift As Integer)

*Occurs when the user releases a key while an object has the focus. Available as the Dialog's **VScrollBarKeyUp** event.*

Property **LargeChange** As Integer

Returns/sets amount of change to Value property in a scroll bar when user clicks the scroll bar area.

Property **Left** As Single

Returns/sets the distance between the internal left edge of an object and the left edge of its container.

Event **LostFocus**

*Occurs when an object loses the focus. Available as the Dialog's **VScrollBarLostFocus** event.*

Property **Max** As Integer

Returns/sets a scroll bar position's maximum Value property setting.

Property **Min** As Integer

Returns/sets a scroll bar position's maximum Value property setting.

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Sub **Move** (ByVal Left As Single, [ByVal Top], [ByVal Width], [ByVal Height])

Moves an object.

Members of the VB.**VScrollBar** class, made available through the object reference returned by the Dialog.**VScrollBarObject** method (continued)

Property **Name** As String

Returns the name used in code to identify an object.

Property **Parent** As Form

Returns the object on which this object is located.

Sub **Refresh**

Forces a complete repaint of a object.

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Event **Scroll**

*Occurs when you reposition the scroll box on a control. Available as the Dialog's **VScrollBarScroll** event.*

Sub **SetFocus**

Moves the focus to the specified object.

Sub **ShowWhatsThis**

Displays a selected topic in a Help file using the What's This popup provided by Windows Help.

Property **SmallChange** As Integer

Returns/sets amount of change to Value property in a scroll bar when user clicks a scroll arrow.

Property **TabIndex** As Integer

Returns/sets the tab order of an object within its parent form.

Property **TabStop** As Boolean

Returns/sets a value indicating whether a user can use the TAB key to give the focus to an object.

Property **Tag** As String

Stores any extra data needed for your program.

Members of the VB.**VScrollBar** class, made available through the object reference returned by the Dialog.**VScrollBarObject** method (continued)

Property **Top** As Single

Returns/sets the distance between the internal top edge of an object and the top edge of its container.

Event **Validate** (Cancel As Boolean)

*Occurs when a control loses focus to a control that causes validation. Available as the Dialog's **VScrollBarValidate** event.*

Property **Value** As Integer

Returns/sets the value of an object.

Property **Visible** As Boolean

Returns/sets a value that determines whether an object is visible or hidden.

Property **WhatsThisHelpID** As Long

Returns/sets an associated context number for an object.

Property **Width** As Single

Returns/sets the width of an object.

Sub **ZOrder** ([ByVal Position])

Places a specified object at the front or back of the z-order within its graphical level.

Members of the VB.**Printer** class, made available through the object reference returned by the Dialog.**PrinterObject** method

Enables you to communicate with a system printer (initially the default printer).

Sub **Circle** (ByVal Step As Integer, ByVal X As Single, ByVal Y As Single, ByVal Radius As Single, ByVal Color As Long, ByVal Start As Single, ByVal End As Single, ByVal Aspect As Single)

*Draws a circle, ellipse, or arc on an object. Don't use it; instead, use the Dialog.**PrinterCircleAbsolute** method.*

Property **ColorMode** As Integer

Returns/sets the selected color.

Property **Copies** As Integer

Returns/sets a value that determines the number of copies to be printed.

Property **CurrentX** As Single

Returns/sets the horizontal coordinates for next print or draw method.

Property **CurrentY** As Single

Returns/sets the vertical coordinates for next print or draw method.

Property **DeviceName** As String

Returns the name of the device a driver supports.

Property **DrawMode** As Integer

Sets the appearance of output from graphics methods or of a Shape or Line control.

Property **DrawStyle** As Integer

Determines the line style for output from graphics methods.

Property **DrawWidth** As Integer

Returns/sets the line width for output from graphics methods.

Property **DriverName** As String

Returns the name of the driver for a Printer object.

Property **Duplex** As Integer

Determines whether a page is printed on both sides.

Members of the VB.**Printer** class, made available through the object reference returned by the Dialog.**PrinterObject** method (continued)

Sub **EndDoc**

Terminates a print operation sent to the Printer object.

Property **FillColor** As Long

Returns/sets the color used to fill in shapes, circles, and boxes.

Property **FillStyle** As Integer

Returns/sets the fill style of a shape.

Property **Font** As stdole.Font

Returns a Font object.

Property **FontBold** As Boolean

Returns/sets bold font styles.

Property **FontCount** As Integer

Returns the number of fonts available for the current display device or active printer.

Property **FontItalic** As Boolean

Returns/sets italic font styles.

Property **FontName** As String

Specifies the name of the font that appears in each row for the given level.

Property **Fonts** (ByVal As Integer) As String

Returns all font names available for the current display device or active printer.

Property **FontSize** As Single

Specifies the size (in points) of the font that appears in each row for the given level.

Property **FontStrikethru** As Boolean

Returns/sets strikethrough font styles.

Property **FontTransparent** As Boolean

Returns/sets a value that determines whether background text/graphics on a Form, Printer or PictureBox are displayed.

Property **FontUnderline** As Boolean

Returns/sets underline font styles.

Members of the VB.**Printer** class, made available through the object reference returned by the Dialog.**PrinterObject** method (continued)

Property **ForeColor** As Long

Returns/sets the foreground color used to display text and graphics in an object.

Property **hDC** As Long

Returns a handle (from Microsoft Windows) to the object's device context.

Property **Height** As Long

Returns/sets the height of an object.

Sub **KillDoc**

Immediately terminates the current print job.

Sub **Line** (ByVal Flags As Integer, ByVal X1 As Single, ByVal Y1 As Single, ByVal X2 As Single, ByVal Y2 As Single, ByVal Color As Long)

*Draws lines and rectangles on an object. Don’t use it; instead, use the Dialog.**PrinterLineAbsolute** method.*

Sub **NewPage**

Ends the current page and advances to the next page on the Printer object.

Property **Orientation** As Integer

Returns/sets a value indicating whether documents are printed in portrait or landscape mode.

Property **Page** As Integer

Returns the current page number.

Sub **PaintPicture** (ByVal Picture As stdole.Picture, ByVal X1 As Single, ByVal Y1 As Single, [ByVal Width1], [ByVal Height1], [ByVal X2], [ByVal Y2], [ByVal Width2], [ByVal Height2], [ByVal Opcode])

*Draws the contents of a graphics file on a Form, PictureBox, or Printer object. Don’t use it; instead, use the Dialog.**PrinterPaintPicture** method.*

Property **PaperBin** As Integer

Returns/sets the default paper bin on the printer from which paper is fed when printing.

Property **PaperSize** As Integer

Returns/sets the paper size for the current printer.

Members of the VB.**Printer** class, made available through the object reference returned by the Dialog.**PrinterObject** method (continued)

Property **Port** As String

Returns the name of the port through which a document is sent to a printer.

Property **PrintQuality** As Integer

Returns/sets a value indicating the printer resolution.

Sub **PSet** (ByVal Step As Integer, ByVal X As Single, ByVal Y As Single, ByVal Color As Long)

*Sets a point on an object to a specified color. Don't use it; instead, use the Dialog.**PrinterPSetAbsolute** method.*

Property **RightToLeft** As Boolean

Determines text display direction and control visual appearance on a bidirectional system.

Sub **Scale** (ByVal Flags As Integer, [ByVal X1], [ByVal Y1], [ByVal X2], [ByVal Y2])

Defines the coordinate system for a Form, PictureBox, or Printer.

Property **ScaleHeight** As Single

Returns/sets the number of units for the vertical measurement of an object's interior.

Property **ScaleLeft** As Single

Returns/sets the horizontal coordinates for the left edges of an object.

Property **ScaleMode** As Integer

Returns/sets a value indicating measurement units for object coordinates when using graphics methods or positioning controls.

Property **ScaleTop** As Single

Returns/sets the vertical coordinates for the top edges of an object.

Property **ScaleWidth** As Single

Returns/sets the number of units for the horizontal measurement of an object's interior.

Function **ScaleX** (ByVal Width As Single, [ByVal FromScale], [ByVal ToScale]) As Single

Converts the value for the width of a Form, PictureBox, or Printer from one unit of measure to another.

Members of the VB.**Printer** class, made available through the object reference returned by the Dialog.**PrinterObject** method (continued)

Function **ScaleY** (ByVal Height As Single, [ByVal FromScale], [ByVal ToScale]) As Single

Converts the value for the height of a Form, PictureBox, or Printer from one unit of measure to another.

Function **TextHeight** (ByVal Str As String) As Single

Returns the height of a text string as it would be printed in the current font.

Function **TextWidth** (ByVal Str As String) As Single

Returns the width of a text string as it would be printed in the current font.

Property **TrackDefault** As Boolean

Returns/sets a value that determines if the Printer object considers the default printer setting in the Control Panel.

Property **TwipsPerPixelX** As Single

Returns the number of twips per pixel for an object measured horizontally.

Property **TwipsPerPixelY** As Single

Returns the number of twips per pixel for an object measured vertically.

Property **Width** As Long

Returns/sets the width of an object.

Property **Zoom** As Long

Returns/sets the percentage by which printed output is to be scaled up or down.

Members of the VB.**Screen** class, made available through the object reference returned by the Dialog.**ScreenObject** method

Manipulates forms according to their placement on the screen and controls the mouse pointer.

Property **ActiveControl** As Control

Returns the control that has focus.

Property **ActiveForm** As Form

Returns the form that is the active window.

Property **FontCount** As Integer

Returns the number of fonts available for the current display device or active printer.

Property **Fonts** (ByVal As Integer) As String

Returns all font names available for the current display device or active printer.

Property **Height** As Single

Returns/sets the height of an object.

Property **MouseIcon** As stdole.Picture

Sets a custom mouse icon. Don't use the stdole.Picture type as parameters / return values for methods in out-of-process ActiveX EXEs because, by design, they generate catastrophic failures.

Property **MousePointer** As Integer

Returns/sets the type of mouse pointer displayed when over part of an object.

Property **TwipsPerPixelX** As Single

Returns the number of twips per pixel for an object measured horizontally.

Property **TwipsPerPixelY** As Single

Returns the number of twips per pixel for an object measured vertically.

Property **Width** As Single

Returns/sets the width of an object.

Some aspects to consider when programming in VBScript/Jscript with the DialogsEXE.Dialog component

When programming in VBScript / Jscript, please note that:

- Syntax errors or runtime errors may generate error entries in Windows Application Log. If you don't want this to happen then make sure to address all these errors.
- Missing third-party libraries may generate error entries in Windows Application Log. Make sure to provide these libraries if you intend to use their features or features based on these libraries.
- The StdPicture / IPictureDisp objects used as arguments or return values generate catastrophic failures (this is by Microsoft design). Use an array of bytes as a substitute for these objects (see the DB-related samples).
- The Object-related functions (E.G.: CheckObject, ComboObject, etc.) should be used carefully; not all their properties or methods can be used freely (E.G.: see the StdPicture / IPictureDisp properties / methods).
- This DialogsEXE.Dialog component makes some of the Visual Basic objects and procedures available for scripting; do not expect the limitations / anomalies / weird behavior of these Visual Basic objects or procedures to be fixed by the DialogsEXE.Dialog component! If a Visual Basic object or procedure is affected by a certain particular problem, then that object or procedure will continue to be affected by that problem, even after having been made available by way of the DialogsEXE.Dialog component.