#[1]$[2]K[3]+[4]axTree Control

~~Properties~~ ~~Methods~~ ~~Events~~

axTree displays a hierarchical list of **Node** objects, each of which consists of a label and a bitmap. **axTree** is typically used to display the headings in a document, the entries in an index, the files and directories on a disk, or any other kind of information that might usefully be displayed as a hierarchy. This activex control was written in VB5-SP3 and is meant as an alternative to the TreeView control which comes with Visual Basic.

File Name
axTree.OCX

Distribution
- Unzip the source code with directories intact.
- Load the sample Project1 with Visual Basic. This should register the OCX for use with other VB applications. If it does not load correctly, then from the DOS prompt, move to the install directory and type: **REGSVR32 axTree.ocx** (you may need to put **c:\windows\system\** in front of regsvr32 if you don't have a path.
- Whenever you want to add the control to a VB application, go to the Project/Components menu and select "ActiveX Tree Control".

~~Revisions~~
~~License~~
~~Tech Support~~

[1] Hlp_Contents
[2] Contents
[3] Contents
[4] Browse:0005

#[5]$[6]K[7]+[8]Revisions

1.0
- Initial release

1.1
- Conversion from array (zero based) to collection (1 based)
- Additional properties and methods added

1.2
- Added properties so user can specify alternate bitmaps to display for nodes and subitems

1.3
- Added ItemData property to match listbox, added new bitmap argument to AddNode to allow user to define bitmap for specific node

1.4
- Fixed subscript error when collapsing and expanding nodes
- Added extra border styles for inner border
- Removed extra line extending tree to top of control
- Added SelectColor and SelectTextColor properties
- Change width of highlight to text width instead of control width

1.4.6
- Changed collapse/expandall method to not move highlight to top
- Added single line border (inner and outer)
- Removed icon from about box
- Default highlight colors are system highlight and highlighttext
- Space or Enter will collapse/expand nodes and Space will checkmark nodes

1.5
- Added AutoCheckParents property
- Fixed major problem with scrolling if the number of nodes was greater than that which would display within the tree

1.6
- Don't change current node when adding new nodes
- Added DblClick, Click, MouseMove, MouseUp, MouseDown, KeyPress, KeyUp, and KeyDown events
- Removed small line at top of tree extending from plussign box to top of tree
- Text was not displaying in correct place if bitmaps, lines, and plussign were turned off
- Added Expand and Collapse methods for specific nodes
- Add NewIndex property
- If Checkmarks are turned on and you check a child node of a parent and the collapse that parent, then an error occurs
- If a tree is defined with more than 3 levels then errors would be generated if the last level was clicked on.

1.6.6
- Added Before parameter to AddNode method
- Implemented faster collection (CollectionEx) class from VBPJ to speed up adding and searching for nodes

1.7
- Added NodeClick event which only fires upon clicking on a node and returns selected item
- Changed Click event to fire whenever control is clicked upon (does not return anything)
- Fixed problem where click area for selecting/expanding a node was not correct if lines and plussigns were turned off
- Added ParentIndex property. Returns parent of selected item and 0 if item does not have a parent.

1.8
- Increased limit of nodes from 10 to 20 levels deep
- NodeClick was firing on every mouse click on control instead of just when a click was made on a node
- Added ability to collapse a node with the left arrow and expand a node with the right arrow

1.8.3

[5] Hlp_Revisions
[6] Revisions
[7] Revisions
[8] Browse:0015

- Fixed problem with bitmap argument on ADDNODE method

1.8.7
- Fixed problem where if you delete a node within the tree (not the last one) and then add a node, you will get an error stating that the index has already been used.

1.8.10
- Fixed Refresh subroutine so it will refresh the display without moving the currently selected item

1.8.12
- Alright, now I really fixed the Refresh subroutine (hehe)

1.8.14
- Changed Bookmark item property from string to variant. It was causing an error, if the bookmark was set to a database bookmark.

1.8.15
- make sure that if a parent node is deleted, then all child nodes are also deleted, otherwise errors will occur

#[9]$[10]K[11]+[12]License

This control was developed and published by Software Solution. You may use it freely for development with Microsoft Visual Basic 5.0. This product is freeware and includes source code which you may change to suit your purposes.

[9] Hlp_License

[10] License

[11] License

[12] Browse:0020

#[13]$[14]K[15]+[16]Tech Support

If you have any problems installing or using this control, please feel free to contact our technical support department at one of the following:

Internet:
kirkq@execpc.com

Telephone:
414-251-0915

Snail Mail:
N92W17053 Roger Ave
Menomonee Falls, WI 53051

HEY! Check out our world wide web page at:
HTTP://www.execpc.com/~kirkq

[13] Hlp_Tech_Support
[14] Tech Support
[15] Tech Support
[16] Browse:0025

#[17]$[18]K[19]+[20]Properties

All of the properties for this control are listed below:

Standard
Enabled
Font
Height
Index
Left
Name
TabIndex
TabStop
Tag
ToolTipText
Top
Visible
Width

Control Specific
~~AutoCheckParents~~
~~BorderStyle~~
~~BorderStyleInner~~
~~Checkmarks~~
~~Count~~
~~ParentIndex~~
~~PictureClosed~~
~~PictureClosedSome~~
~~PictureClosedAll~~
~~PictureItemNotSelected~~
~~PictureItemSelected~~
~~PictureOpen~~
~~PictureOpenSome~~
~~PictureOpenAll~~
~~SelectColor~~
~~SelectTextColor~~
~~SelectedItem~~
~~ShowBitmaps~~
~~ShowPlusSigns~~
~~ShowLines~~

[17] Hlp_Properties
[18] Properties
[19] Properties
[20] Browse:0030

#[21]$[22]K[23]+[24]Methods

All of the methods for this control are listed below:

Standard
Move
Refresh
SetFocus
ShowAbout
ShowWhatsThis
ZOrder

Control Specific
~~AddNode~~
~~Clear~~
~~Collapse~~
~~CollapseAll~~
~~Expand~~
~~ExpandAll~~
~~Find~~
~~Item~~
~~RemoveNode~~

[21] Hlp_Methods
[22] Methods
[23] Methods
[24] Browse:0035

#[25]$[26]K[27]+[28]Events

All of the events for this control are listed below:

Standard
Click
DblClick
GotFocus
LostFocus
KeyDown
KeyPress
KeyUp
MouseDown
MouseUp
MouseMove
DragDrop
DragOver

Control Specific
~~Collapse~~
~~Expand~~
~~NodeClick~~

[25] Hlp_Events
[26] Events
[27] Events
[28] Browse:0040

#[29]+[30]Standard Property/Method/Event

Depending on your host environment, this property/method/event may be referred to by a different name or may not apply to this control. Refer to your host environments documentation or help file for further information.

[29] Hlp_Standard

[30] Browse:0045

#[31]$[32]K[33]+[34]Collapse Event

Occurs when a node in the control is collapsed

Syntax:

object.**Collapse**(Index as integer)

The event syntax has these parts:

Part	**Description**
object	An object expression that evaluates to a axTree object
Index	Returns the index of the currently selected index

Remarks:

[31] Hlp_Collapse

[32] Collapse

[33] Collapse

[34] Browse:0055

#[35]$[36]K[37]+[38]Expand Event

Occurs when any node in control is expanded, that is, when its child nodes become visible

Syntax:

object.**Expand**(Index as integer)

The event syntax has these parts:

Part	**Description**
object	An object expression that evaluates to a axTree object
Index	Returns the index of the currently selected node

Remarks:

[35] Hlp_Expand
[36] Expand
[37] Expand
[38] Browse:0060

#[39]$[40]K[41]+[42]AddNode Method

Add new node to control's Nodes collection

Syntax:

*object.***AddNode**(**Caption as string, Level as integer,**
IsParent as boolean,
optional Before, optional After,
optional Bookmark, optional ItemData,
optional Bitmap, optional Visible,
optional Expanded, optional VisibleIdx,
optional Selected, optional SyncIdx)

The method syntax has these parts:

Part	Description
object	An object expression that evaluates to a axTree object
Caption	**Required. The string that appears in the node.**
Level	**Required. Integer determining from left to right (0 based) hierarchel level.**
IsParent	**Required. Boolean determing whether node is a parent or not.**
Before	**Optional. Integer to determine which current node to add new node before.**
After	**Optional. Integer to determine which current node to add new node after.**
Bookmark	**Optional. Uniquely identifies the current record in a recordset.**
ItemData	**Optional. Sets a specific number for the node object.**
Bitmap	**Optional. Picture object for alternate bitmap to display in this node**
Visible	**Optional. Boolean to determine if node is visible (collapsed child).**
Expanded	**Optional. Boolean to determine if parent node is expanded.**
VisibleIdx	**Optional**
Selected	**Optional. Integer to determine select status: 0-none, 1-all, 2-some**
SyncIdx	**Optional. Syncs this node to another array**

Example:
Example (Form_Load):

```
'clear hierarchy list
axTree1.Clear
'Add first item, level 0, child
axTree1.AddNode "Top", 0, False
'add second item, level 0, parent
axTree1.AddNode "Group", 0, True
'add third item, level 1, parent
axTree1.AddNode "SubGroup1", 1,True
'add forth item, level 2, child
axTree1.AddNode "SubItem1", 2, False
axTree1.AddNode "SubItem2", 2, False
axTree1.AddNode "Bottom", 0, False
'Collapse all parents so no children are showing, optional
axTree1.CollapseAll
```

[39] Hlp_AddNode
[40] AddNode
[41] AddNode
[42] Browse:0065

You must place the AddNode statements in the order you want to see the hierarchy list.

Level represents the position from left to right that the item will appear in the hierarchy list (zero based):

Level 0
Level 1
Level 3

#[43]$[44]K[45]+[46]Clear Method

Clears all node objects from nodes collection

Syntax:

***object*.Clear**

The method syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a axTree object**

Example:

[43] Hlp_Clear

[44] Clear

[45] Clear

[46] Browse:0070

#[47]$[48]K[49]+[50]Collapse Method

Collapse a specific node

Syntax:

***object*.Collapse(Index as integer)**

The method syntax has these parts:

Part	Description
object	**An object expression that evaluates to a axGrid object**
Index	**Node index to collapse**

Example:

[47] Hlp_Collapse1

[48] Collapse

[49] Collapse

[50] Browse:0075

#[51]$[52]K[53]+[54]CollapseAll Method

Collapse all nodes in Nodes collection so that no child nodes are visible

Syntax:

***object*.CollapseAll**

The method syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**

Example:

[51] Hlp_CollapseAll

[52] CollapseAll

[53] CollapseAll

[54] Browse:0080

#[55]$[56]K[57]+[58]Expand Method

Expand a specific node to show child nodes

Syntax:

***object*.Expand(Index as integer)**

The method syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
Index	**Index of node to expand. Must be a parent node.**

Example:

[55] Hlp_Expand1

[56] Expand

[57] Expand

[58] Browse:0085

#[59]$[60]K[61]+[62]ExpandAll Method

Expand all nodes in Nodes collection so that all child nodes are visible

Syntax:

***object*.ExpandAll**

The method syntax has these parts:

Part	Description
object	An object expression that evaluates to a control object

Example:

[59] Hlp_ExpandAll

[60] ExpandAll

[61] ExpandAll

[62] Browse:0090

#[63]$[64]K[65]+[66]Find Method

Find a node object by its caption name

Syntax:

***object*.Find(Caption as string, Mode as integer)**

The method syntax has these parts:

Part	Description
object	An object expression that evaluates to a control object
Caption	Search all nodes for this string
Mode	Indicates where to start searching from: 0=top, 1=current position

Example:

```
axTree1.Find "test",0        'find "test" in axTree1 starting from top
axTree1.Find "test",1        'find "test" in axTree1 starting from current line
```

[63] Hlp_Find

[64] Find

[65] Find

[66] Browse:0095

#[67]$[68]K[69]+[70]Item Method

Returns a node object by index or by key

Syntax:

***object*.Item(Key as variant)**

The method syntax has these parts:

Part	Description
object	An object expression that evaluates to a control object
Key	Can be a string or integer. If string, then return item based on caption, otherwise return item based on index

Item Properties:

Caption	The string that appears in the node.
Level	Integer determining from left to right (0 based) hierarchel level.
IsParent	Boolean determing whether node is a parent or not.
Bookmark	Variant: Uniquely identifies the current record in a recordset.
ItemData	Sets a specific number for the node object.
Bitmap	Picture object for alternate bitmap to display in this node
Visible	Boolean to determine if node is visible (collapsed child). *DO NOT CHANGE*
Expanded	Boolean to determine if parent node is expanded. *DO NOT CHANGE*
VisibleIdx	
Selected	Integer to determine select status: 0-none, 1-all, 2-some
SyncIdx	Syncs this node to another array

[67] Hlp_Item

[68] Item

[69] Item

[70] Browse:0100

#[71]$[72]K[73]+[74]ItemData Property

Returns/sets a specific number for each node in the control

Syntax:

***object*.ItemData(Index as integer)**

The method syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
Index	**Index of node**

Example:

[71] Hlp_ItemData
[72] ItemData
[73] ItemData
[74] Browse:0105

#[75]$[76]K[77]+[78]RemoveNode Method

Remove node object from Nodes collection

Syntax:

***object*.RemoveNode(Index as integer)**

The method syntax has these parts:

Part	Description
object	An object expression that evaluates to a control object
Index	Index of node to remove from nodes collection

Example:

[75] Hlp_RemoveNode

[76] RemoveNode

[77] RemoveNode

[78] Browse:0110

#[79]$[80]K[81]+[82]AutoCheckParents Property

Returns or sets a boolean. If the tree is setup for multiple node selection (see CheckMarks), this property is used to determine if the parent(s) of a child node are selected when the child node is selected.

Syntax:

***object*.AutoCheckParents [= boolean]**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
boolean	**A boolean expression that evaluates to True or False**

Remarks

[79] Hlp_AutoCheckParents

[80] AutoCheckParents

[81] AutoCheckParents

[82] Browse:0115

#[83]$[84]K[85]+[86]BorderStyle Property

Returns or sets the borderstyle for this control

Syntax:

***object*.BorderStyle [= integer]**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
integer	**An integer expression that evaluates to the borderstyle as listed under Settings**

Settings:

[No Border] = 0
[Single] = 1
[Thin Raised] = 2
[Thick Raised] = 3
[Thin Inset] = 4
[Thick Inset] = 5
[Etched] = 6
[Bump] = 7

[83] Hlp_BorderStyle

[84] BorderStyle

[85] BorderStyle

[86] Browse:0120

#[87]$[88]K[89]+[90]BorderStyleInner Property

Returns or sets the border style for inner area of control (none, default)

Syntax:

***object*.BorderStyleInner [= integer]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
string	**An integer expression that evaluates to the inner borderstyle for the control as listed below in Settings**

Settings:

[iNo Border] = 0
[iSingle] = 1
[iThin Raised] = 2
[iThick Raised] = 3
[iThin Inset] = 4
[iThick Inset] = 5

[87] Hlp_BorderStyleInner

[88] BorderStyleInner

[89] BorderStyleInner

[90] Browse:0125

#[91]$[92]K[93]+[94]Checkmarks Property

Returns/sets a value to determine if user has the ability to mark multiple lines in the hierarchy list

Syntax:

***object*.CheckMarks [= boolean]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
boolean	**A boolean expression that evaluates to True or False**

Remarks

[91] Hlp_Checkmarks
[92] Checkmarks
[93] Checkmarks
[94] Browse:0130

#[95]$[96]K[97]+[98]Count Property

Returns total number of node objects in Nodes collection

Syntax:

***object*.Count**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**

Remarks

[95] Hlp_Count

[96] Count

[97] Count

[98] Browse:0135

#[99]$[100]K[101]+[102]PictureClosed Property

Sets/returns the alternate bitmap displayed for all closed nodes with no subitems selected

Syntax:

***object*.PictureClosed [= picture]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[99] Hlp_PictureClosed

[100] PictureClosed

[101] PictureClosed

[102] Browse:0140

#[103]$[104]K[105]+[106]PictureClosedSome Property

Sets/returns the alternate bitmap displayed for all closed nodes with some subitems selected

Syntax:

***object*.PictureClosedSome [= picture]**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[103] Hlp_PictureClosedSome

[104] PictureClosedSome

[105] PictureClosedSome

[106] Browse:0145

#[107]$[108]K[109]+[110]PictureClosedAll Property

Sets/returns the alternate bitmap displayed for closed nodes with all subitems selected

Syntax:

***object*.PictureClosedAll [= picture]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[107] Hlp_PictureClosedAll

[108] PictureClosedAll

[109] PictureClosedAll

[110] Browse:0150

#[111]$[112]K[113]+[114]PictureItemNotSelected Property

Sets/returns the alternate bitmap displayed for node items not selected

Syntax:

***object*.PictureItemNotSelected [= picture]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[111] Hlp_PictureItemNotSelected

[112] PictureItemNotSelected

[113] PictureItemNotSelected

[114] Browse:0155

#[115]$[116]K[117]+[118]PictureItemSelected Property

Sets/returns the alternate bitmap displayed for node items selected

Syntax:

***object*.PictureItemSelected [= picture]**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[115] Hlp_PictureItemSelected

[116] PictureItemSelected

[117] PictureItemSelected

[118] Browse:0160

#[119]$[120]K[121]+[122]PictureOpen Property

Sets/returns the alternate bitmap displayed for open nodes with no subitems selected

Syntax:

***object*.PictureOpen [= picture]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[119] Hlp_PictureOpen

[120] PictureOpen

[121] PictureOpen

[122] Browse:0165

#[123]$[124]K[125]+[126]PictureOpenSome Property

Sets/returns the alternate bitmap displayed for open nodes with some subitems selected

Syntax:

***object*.PictureOpenSome [= picture]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[123] Hlp_PictureOpenSome

[124] PictureOpenSome

[125] PictureOpenSome

[126] Browse:0170

#[127]$[128]K[129]+[130]PictureOpenAll Property

Sets/returns the alternate bitmap displayed for open nodes with all subitems selected

Syntax:

***object*.PictureOpenAll [= picture]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
picture	**An expression that evaluates to a picture object**

Remarks

[127] Hlp_PictureOpenAll

[128] PictureOpenAll

[129] PictureOpenAll

[130] Browse:0175

#[131]$[132]K[133]+[134]SelectColor Property

Returns/sets the text background color of the currently selected node

Syntax:

***object*.SelectColor [= color]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
color	**An long expression to determine a color**

Remarks

[131] Hlp_SelectColor

[132] SelectColor

[133] SelectColor

[134] Browse:0180

#[135]$[136]K[137]+[138]SelectTextColor Property

Returns/sets the text forecolor of the currently selected node

Syntax:

***object*.SelectTextColor [= color]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
color	**A long expression that evaluates to a color**

Remarks

[135] Hlp_SelectTextColor

[136] SelectTextColor

[137] SelectTextColor

[138] Browse:0185

#[139]$[140]K[141]+[142]SelectedItem Property

Returns/sets the currently selected node object

Syntax:

object.SelectedItem [= integer]

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
integer	**An integer expression that evaluates to the index of a node**

Remarks

[139] Hlp_SelectedItem

[140] SelectedItem

[141] SelectedItem

[142] Browse:0195

#[143]$[144]K[145]+[146]ShowBitmaps Property

Returns/sets a value to determine whether node bitmaps are displayed on control.

Syntax:

***object*.ShowBitmaps [= boolean]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
boolean	**A boolean expression that evaluates to True or False**

Remarks

[143] Hlp_ShowBitmaps

[144] ShowBitmaps

[145] ShowBitmaps

[146] Browse:0200

#[147]$[148]K[149]+[150]ShowPlusSigns Property

Returns/sets a value to determine whether the parent indicators (plus signs) are displayed on control.

Syntax:

***object*.ShowPlusSigns [= boolean]**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
boolean	**A boolean expression that evaluates to True or False**

Remarks

[147] Hlp_ShowPlusSigns
[148] ShowPlusSigns
[149] ShowPlusSigns
[150] Browse:0205

#[151]$[152]K[153]+[154]ShowLines Property

Returns/sets a value to determine whether node lines are displayed on control.

Syntax:

***object*.ShowLines [= boolean]**

The property syntax has these parts:

Part	Description
object	**An object expression that evaluates to a control object**
boolean	**A boolean expression that evaluates to True or False**

Remarks

[151] Hlp_ShowLines
[152] ShowLines
[153] ShowLines
[154] Browse:0210

#[155]$[156]K[157]+[158]NodeClick Event

Occurs when a node is clicked upon

Syntax:

***object*.NodeClick(Index as integer)**

The event syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a axGrid object**
Index	**Index of selected node**

Remarks:

[155] Hlp_NodeClick

[156] NodeClick

[157] NodeClick

[158] Browse:0215

#[159]$[160]K[161]+[162]ParentIndex Property

Returns the parent index of the selected node. If the node does not have a parent, then it returns 0

Syntax:

***object*.ParentIndex(Index as integer)**

The property syntax has these parts:

Part	**Description**
object	**An object expression that evaluates to a control object**
Index	**An integer expression that evaluates to a specific node index**

Remarks

[159] Hlp_ParentIndex

[160] ParentIndex

[161] ParentIndex

[162] Browse:0220