



mcBar v2.00

multi-purpose bar activex control

all updates are on <http://www.geocities.com/SiliconValley/Way/7409>

mcBar is COMPATIBLE with VB 5.0 (any versions) or higher, for Access 97.
Normally, mcBar is compatible with all softwares supporting ActiveX.

ANY REGISTERED USERS CAN ASK ME TO ADD SOME FUNCTIONNALITIES.

IMPORTANT NOTICE

Overview

Properties

Events

Methods

Current version

Registration on CompuServe or on Internet

New features

Revision history

Acknowledgement

Installation

License agreement

Distribution note

Technical support

BlockKey

Purpose :

Normally, each block is referenced by a numeric value between 1 to .Blocks. However, it can be more interesting to reference a block with a name in place of a number. So, with this property, you can reference a block with a name.

Type :

String Property

Syntax :

```
mcBar1.BlockKey([iBlock]) = value$  
value$ = mcBar1.BlockKey([iBlock])
```

Default :

""

Comments :

Example(s) :

```
mcBar1.BlockKey(1) = "break 1"
```

See also :

@

Purpose :

Type :

Property

Syntax :

```
mcBar1.([iBlock]) = value  
value = mcBar1.([iBlock])
```

Default :

Comments :

Example(s) :

```
mcBar1.(1) =
```

See also :

BlockBarLeft

Purpose :

[BlockBarLeft](#) determines the Left edge of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockBarLeft([iBlock]) = value%  
value% = mcBar1.BlockBarLeft([iBlock])
```

Default :

-1

Comments :

Example(s) :

```
mcBar1.BlockBarLeft(1) = 25
```

See also :

BlockBarRight

Purpose :

[BlockBarRight](#) determines the Right edge of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockBarRight([iBlock]) = value%  
value% = mcBar1.BlockBarRight([iBlock])
```

Default :

-1

Comments :

Example(s) :

```
mcBar1.BlockBarRight(1) = 77
```

See also :

BlockBarTop

Purpose :

[BlockBarTop](#) determines the Top edge of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockBarTop([iBlock]) = value%  
value% = mcBar1.BlockBarTop([iBlock])
```

Default :

-1

Comments :

Example(s) :

```
mcBar1.BlockBarTop(1) = 77
```

See also :

BlockBarBottom

Purpose :

[BlockBarBottom](#) determines the Bottom edge of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockBarBottom([iBlock]) = value%  
value% = mcBar1.BlockBarBottom([iBlock])
```

Default :

-1

Comments :

Example(s) :

```
mcBar1.BlockBarBottom(1) = 25
```

See also :

BlockBorderStyle

Purpose :

[BlockBorderStyle](#) determines the Border Style of a block.

Type :

Enumerated Property









Syntax :

```
mcBar1.BlockBorderStyle([iBlock]) = value%  
value% = mcBar1.BlockBorderStyle([iBlock])
```

Default :

mcNoneBorder

Comments :

0 - mcNoneBorder	
1 - mcFlat	
2 - mcRaised	
3 - mcInset	
4 - mcEtched	
5 - mcEtchedLight	
6 - mcBump	
7 - mcBumpLight	

Example(s) :

```
mcBar1.BlockBorderStyle(1) = mcInset
```

See also : [BorderStyle](#)

BlockBorderWidth

Purpose :

[BlockBorderWidth](#) determines the Border Width of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockBorderWidth([iBlock]) = value%  
value% = mcBar1.BlockBorderWidth([iBlock])
```

Default :

1

Comments :

Only usable whether the [.BlockBorderStyle](#) is not mcNoneBorder.

Example(s) :

```
mcBar1.BlockBorderWidth(1) = 2
```

See also :

BlockShapeStyle

Purpose :

[BlockShapeStyle](#) determines the Shape Style of a block.

Type :

Enumerated Property

Syntax :

```
mcBar1.BlockShapeStyle([iBlock]) = value%  
value% = mcBar1.BlockShapeStyle([iBlock])
```

Default :

mcRectangle

Comments :

0 - mcRectangle
1 - mcCircle

Example(s) :

```
mcBar1.BlockShapeStyle(1) = mcCircle
```

See also :

BlockBrushStyle

Purpose :

[BlockBrushStyle](#) determines the Brush Style of a block.

Type :

Enumerated Property





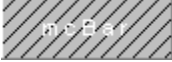


Syntax :

```
mcBar1.BlockBrushStyle([iBlock]) = value%  
value% = mcBar1.BlockBrushStyle([iBlock])
```

Default :

mcNoneBrush

Comments :

0 - mcNoneBrush	
1 - mcHorizontal	
2 - mcVertical	
3 - mcForwardDiagonal	
4 - mcBackwardDiagonal	
5 - mcCross	
6 - mcDiagonalCross	

Example(s) :

```
mcBar1.BlockBrushStyle(1) = mcDiagonalCross
```

See also : [BrushStyle](#)

BlockBrushColor

Purpose :

[BlockBrushColor](#) determines the Brush Color of a block.

Type :

Long Property

Syntax :

```
mcBar1.BlockBrushColor([iBlock]) = value&  
value& = mcBar1.BlockBrushColor([iBlock])
```

Default :

vbBlack

Comments :

Only usable whether the .BlockBrushStyle is not mcNoneBrush.

Example(s) :

```
mcBar1.BlockBrushColor(1) = vbBlue
```

See also :

BlockOnTop

Purpose :

[BlockOnTop](#) determines whether a block is on top of another.

Type :

Boolean Property

Syntax :

```
mcBar1.BlockOnTop([iBlock]) = value  
value = mcBar1.BlockOnTop([iBlock])
```

Default :

False

Comments :

Example(s) :

```
mcBar1.BlockOnTop(1) = True
```

See also :

BlockHPercentage

Purpose :

[BlockHPercentage](#) retrieves the Horizontal Percentage of a block.

Type :

Integer Property (Read-Only)

Syntax :

```
value% = mcBar1.BlockHPercentage([iBlock])
```

Default :

0

Comments :

Example(s) :

See also :

BlockVPercentage

Purpose :

[BlockVPercentage](#) retrieves the Vertical Percentage of a block.

Type :

Integer Property (Read-Only)

Syntax :

```
value% = mcBar1.BlockVPercentage([iBlock])
```

Default :

0

Comments :

Example(s) :

See also :

BlockColor

Purpose :

[BlockColor](#) determines the Color of a block.

Type :

Long Property

Syntax :

```
mcBar1.BlockColor([iBlock]) = value&  
value& = mcBar1.BlockColor([iBlock])
```

Default :

&H808080 (Gray Light)

Comments :

Example(s) :

```
mcBar1.BlockColor(1) = vbWhite
```

See also :

BlockGradientStyle

Purpose :

[BlockGradientStyle](#) determines the Gradient Style of a block.

Type :

Enumerated Property

Syntax :

```
mcBar1.BlockGradientStyle([iBlock]) = value%  
value% = mcBar1.BlockGradientStyle([iBlock])
```

Default :

mcNoneGradient

Comments :

.BlockColor = vbBlue | .BlockColorEnd = vbWhite

0 - mcNoneGradient



1 - mcLeftHGradient



2 - mcRightHGradient



3 - mcTopVGradient



4 - mcBottomVGradient



5 - mcInnerHGradient



6 - mcOuterHGradient



7 - mcInnerVGradient



8 - mcOuterVGradient



9 - mcInnerCircleGradient



10 - mcOuterCircleGradient



11 - mcInnerRectangleGradient



12 - mcOuterRectangleGradient



Example(s) :

```
mcBar1.BlockGradientStyle(1) = mcOuterHGradient
```

See also :

BlockGradientGranularity

Purpose :

[BlockGradientGranularity](#) determines the Gradient Granularity of a block.

Type :

Enumerated Property

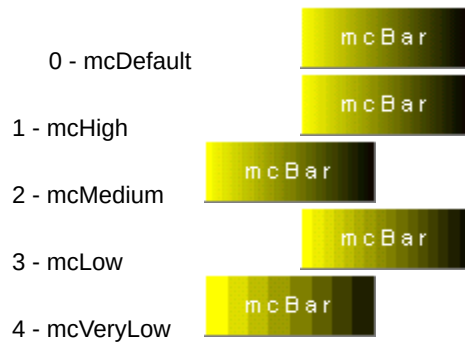
Syntax :

```
mcBar1.BlockGradientGranularity([iBlock]) = value%  
value% = mcBar1.BlockGradientGranularity([iBlock])
```

Default :

mcDefault

Comments :



Only usable whether the [.BlockGradientStyle](#) is not mcNoneGradient.

The mcDefault value is determined by the number of colors your system can handle.
Because the mcDefault value uses more colors than the mcVeryLow, the speed to draw this gradient is the slowest.
A good compromise is the mcMedium when you block is not too big.

Example(s) :

```
mcBar1.BlockGradientGranularity(1) = mcMedium
```

See also : [GradientGranularity](#)

BlockGradientColor

Purpose :

[BlockGradientColor](#) determines the Gradient Start Color of a block.

Type :

Long Property

Syntax :

```
mcBar1.BlockGradientColor([iBlock]) = value&  
value& = mcBar1.BlockGradientColor([iBlock])
```

Default :

vbBlue

Comments :

Only usable whether the [.BlockGradientStyle](#) is not mcNoneGradient.

Example(s) :

```
mcBar1.BlockGradientColor(1) = vbWhite
```

See also :

BlockText

Purpose :

[BlockText](#) determines the text in a block.

Type :

String Property

Syntax :

```
mcBar1.BlockText([iBlock]) = value$  
value$ = mcBar1.BlockText([iBlock])
```

Default :

""

Comments :

You can use the following item in your text to display some extra informations :

!short time!	: displays the short time (see windows default format)
!medium time!	: displays the medium time (see windows default format)
!long time!	: displays the long time (see windows default format)
!general date!	: displays the general date (see windows default format)
!short date!	: displays the short date (see windows default format)
!medium date!	: displays the medium date (see windows default format)
!long date!	: displays the long date (see windows default format)
!h%!	: displays the horizontal percentage of a block
!v%!	: displays the vertical percentage of a block
!disk free:X!	: displays the free disk space of the disk X
!disk used:X!	: displays the used disk space by the disk X
!disk total:X!	: displays the total disk space of the disk X
!ratio free:X!	: displays the free ratio space of the disk X
!ratio used:X!	: displays the used ratio space of the disk X
!show vertical ratio free:X!	: displays the free ratio space of the disk X and modifies the .BlockBarTop
!show vertical ratio used:X!	: displays the used ratio space of the disk X and modifies the .BlockBarTop
!show horizontal ratio free:X!	: displays the free ratio space of the disk X and modifies the .BlockBarRight
!show horizontal ratio used:X!	: displays the used ratio space of the disk X and modifies the .BlockBarRight

Example(s) :

```
mcBar1.BlockText(1) = "break 8-10"  
  
mcBar1.BlockText(2) = "process is !h%! complete"  
  
mcBar1.BlockText(3) = "disk C is used at !ratio used:C!"
```

See also :

CaptionAlignment

Purpose :

[CaptionAlignment](#) determines the Caption Alignment of the object.

Type :

Enumerated Property

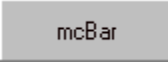
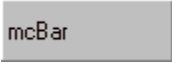
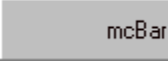
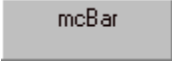
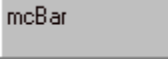
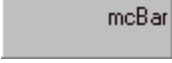
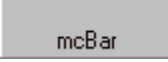
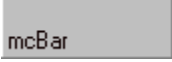
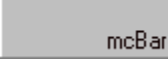
Syntax :

```
mcBar1.CaptionAlignment = value%  
value% = mcBar1.CaptionAlignment
```

Default :

mcCenterCenter

Comments :

0 - mcCenterCenter	
1 - mcCenterLeft	
2 - mcCenterRight	
3 - mcTopCenter	
4 - mcTopLeft	
5 - mcTopRight	
6 - mcBottomCenter	
7 - mcBottomLeft	
8 - mcBottomRight	

Example(s) :

```
mcBar1.CaptionAlignment = mcBottomRight
```

See also : [BlockTextAlignment](#)

BlockTextColor

Purpose :

[BlockTextColor](#) determines the Text Color of a block.

Type :

Long Property

Syntax :

```
mcBar1.BlockTextColor([iBlock]) = value&  
value& = mcBar1.BlockTextColor([iBlock])
```

Default :

vbBlack

Comments :

Example(s) :

```
mcBar1.BlockTextColor(1) = vbYellow
```

See also :

BlockTextAlignment

Purpose :

[BlockTextAlignment](#) determines the Text Alignment of a block.

Type :

Enumerated Property


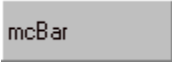
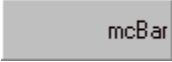
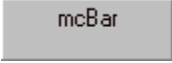
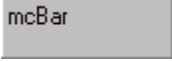
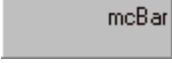
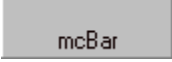
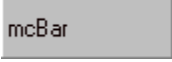
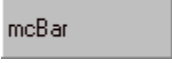
Syntax :

```
mcBar1.BlockTextAlignment([iBlock]) = value%  
value% = mcBar1.BlockTextAlignment([iBlock])
```

Default :

mcCenterCenter

Comments :

0 - mcCenterCenter	
1 - mcCenterLeft	
2 - mcCenterRight	
3 - mcTopCenter	
4 - mcTopLeft	
5 - mcTopRight	
6 - mcBottomCenter	
7 - mcBottomLeft	
8 - mcBottomRight	

Example(s) :

```
mcBar1.BlockTextAlignment(1) = mcBottomRight
```

See also : [CaptionAlignment](#)

BlockTextFont

Purpose :

[BlockTextFont](#) determines the Font of a block.

Type :

Font Property

Syntax :

```
Set mcBar1.BlockFont([iBlock]) = fFont  
Set fFont = mcBar1.BlockFont([iBlock])
```

Default :

"MS Sans Serif" 8.25

Comments :

Example(s) :

```
Set mcBar1.BlockFont(1) = mcBar1.Font
```

See also :

BlockTextFont3D

Purpose :

[BlockTextFont3D](#) determines the Text Font 3D of a block.

Type :

Enumerated Property





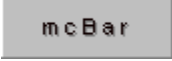
Syntax :

```
mcBar1.BlockTextFont3D([iBlock]) = value%  
value% = mcBar1.BlockTextFont3D([iBlock])
```

Default :

mcNoneFont3D

Comments :

0 - mcNoneFont3D	
1 - mcRaisedLight	
2 - mcInsetLight	
3 - mcRaisedShadow	
4 - mcInsetShadow	

Example(s) :

```
mcBar1.BlockTextFont3D(1) = mcRaisedShadow
```

See also : [CaptionFont3D](#)

BlockTextOffset

Purpose :

[BlockTextOffset](#) determines the Text Offset of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockTextOffset([iBlock]) = value%  
value% = mcBar1.BlockTextOffset([iBlock])
```

Default :

3

Comments :

Only usable whether the [.BlockTextAlignment](#) is not mcCenterCenter.

Example(s) :

```
mcBar1.BlockTextOffset(1) = 2
```

See also : [CaptionOffset](#)

BlockTextAngle

Purpose :

[BlockTextAngle](#) determines the Text Angle of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockTextAngle([iBlock]) = value%  
value% = mcBar1.BlockTextAngle([iBlock])
```

Default :

0

Comments :

Only usable whether the .BlockTextFont is a True Type Font.

The .BlockTextAngle is in degree from 0 to 360.

Example(s) :

```
mcBar1.BlockTextAngle(1) = 45
```

See also :

BlockTextAutoAngle

Purpose :

[BlockTextAutoAngle](#) determines the Text Auto Angle of a block.

Type :

Enumerated Property



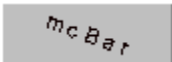
Syntax :

```
mcBar1.BlockTextAutoAngle([iBlock]) = value%  
value% = mcBar1.BlockTextAutoAngle([iBlock])
```

Default :

mcNoneAuto

Comments :

0 - mcNoneAuto	
1 - mcUp	
2 - mcDown	

This property overrides the [.BlockTextAngle](#).

Example(s) :

```
mcBar1.BlockTextAutoAngle(1) = mcDown
```

See also : [CaptionAutoAngle](#)

BlockRivetsStyle

Purpose :

[BlockRivetsStyle](#) determines the Rivets Style of a block.

Type :

Enumerated Property






Syntax :

```
mcBar1.BlockRivetsStyle([iBlock]) = value%  
value% = mcBar1.BlockRivetsStyle([iBlock])
```

Default :

mcNoneRivets

Comments :

0 - mcNoneRivets	
1 - mcRaisedRivets	
2 - mcInsetRivets	
3 - mcRaisedRivetsExt	
4 - mcInsetRivetsExt	

Example(s) :

```
mcBar1.BlockRivetsStyle(1) = mcRaisedRivetsExt
```

See also : [RivetsStyle](#)

BlockPicture

Purpose :

[BlockPicture](#) determines the Picture of a block.

Type :

Picture Property

Syntax :

```
Set mcBar1.BlockPicture([iBlock]) = pPicture  
Set pPicture = mcBar1.BlockPicture([iBlock])
```

Default :

```
LoadPicture("")
```

Comments :

The .BlockPicture can be :

```
Bitmap (*.bmp, *.rle);  
DIB (*.dib);  
GIF (*.gif);  
Icon (*.ico);  
JPG - JPEG (*.jpg, *.jpeg);  
Metafile (*.emf, *.wmf).
```

Example(s) :

```
Set mcBar1.BlockPicture(1) = LoadPicture("c:\win95\egypte.bmp")
```

See also :

BlockPictureOffset

Purpose :

[BlockPictureOffset](#) determines the Picture Offset of a block.

Type :

Integer Property

Syntax :

```
mcBar1.BlockPictureOffset([iBlock]) = value%  
value% = mcBar1.BlockPictureOffset([iBlock])
```

Default :

3

Comments :

Only usable whether the [.BlockPictureAlignment](#) is not mcCenterCenter.

Example(s) :

```
mcBar1.BlockPictureOffset(1) = 2
```

See also :

BlockPictureAlignment

Purpose :

[BlockPictureAlignment](#) determines the Picture Alignment of a block.

Type :

Enumerated Property










Syntax :

```
mcBar1.BlockPictureAlignment([iBlock]) = value%  
value% = mcBar1.BlockPictureAlignment([iBlock])
```

Default :

mcCenterCenter

Comments :

0 - mcCenterCenter	
1 - mcCenterLeft	
2 - mcCenterRight	
3 - mcTopCenter	
4 - mcTopLeft	
5 - mcTopRight	
6 - mcBottomCenter	
7 - mcBottomLeft	
8 - mcBottomRight	

Example(s) :

```
mcBar1.BlockPictureAlignment(1) = mcBottomRight
```

See also : [PictureAlignment](#)

BlockPictureTile

Purpose :

[BlockPictureTile](#) determines the Picture Tile of a block.

Type :

Enumerated Property

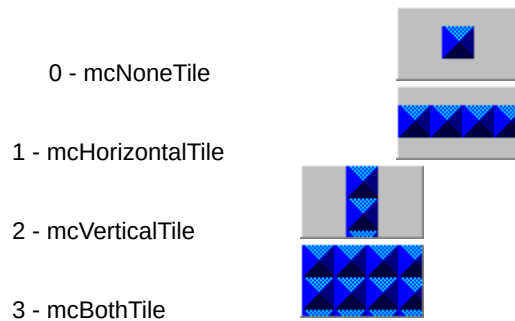
Syntax :

```
mcBar1.BlockPictureTile([iBlock]) = value%  
value% = mcBar1.BlockPictureTile([iBlock])
```

Default :

mcNoneTile

Comments :



Example(s) :

```
mcBar1.BlockPictureTile(1) = mcBothTile
```

See also : [PictureTile](#)

IMPORTANT NOTICE

In order to use correctly mcBar, you must have on your computer the following files :

file	version
MSVBVM50.DLL	05.00.4319
OLEAUT32.DLL	02.20.4118
OLEPRO32.DLL	05.00.4118
STDOLE2.TLB	02.20.4118
ASYCFILT.DLL	02.20.4118
COMCAT.DLL	04.71.0000

All of these files are included in a file called [Msvbvm50.exe](#) (size : 1307480 bytes).

The file can be download from <http://support.microsoft.com/download/support/mslfiles/Msvbvm50.exe>

Picture

Purpose :

[Picture](#) determines the Picture of the object.

Type :

Picture Property

Syntax :

```
Set mcBar1.Picture = pPicture  
Set pPicture = mcBar1.Picture
```

Default :

```
LoadPicture("")
```

Comments :

The .Picture can be :

```
Bitmap (*.bmp, *.rle);  
DIB (*.dib);  
GIF (*.gif);  
Icon (*.ico);  
JPG - JPEG (*.jpg, *.jpeg);  
Metafile (*.emf, *.wmf).
```

Example(s) :

```
Set mcBar1.Picture = LoadPicture("c:\win95\champagne.bmp")
```

See also :

PictureAlignment

Purpose :

[PictureAlignment](#) determines the Picture Alignment of the object.

Type :

Enumerated Property

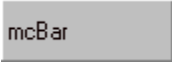
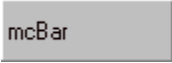
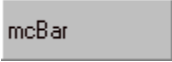
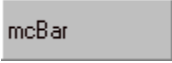
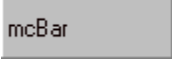
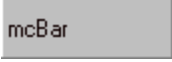
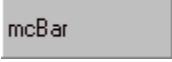
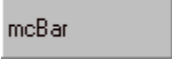

Syntax :

```
mcBar1.PictureAlignment = value%  
value% = mcBar1.PictureAlignment
```

Default :

mcCenterCenter

Comments :

0 - mcCenterCenter	
1 - mcCenterLeft	
2 - mcCenterRight	
3 - mcTopCenter	
4 - mcTopLeft	
5 - mcTopRight	
6 - mcBottomCenter	
7 - mcBottomLeft	
8 - mcBottomRight	

Example(s) :

```
mcBar1.PictureAlignment = mcBottomRight
```

See also : [BlockPictureAlignment](#)

PictureOffset

Purpose :

[PictureOffset](#) determines the Picture Offset of the object.

Type :

Integer Property

Syntax :

```
mcBar1.PictureOffset = value%  
value% = mcBar1.PictureOffset
```

Default :

3

Comments :

Only usable whether the [.PictureAlignment](#) is not mcCenterCenter.

Example(s) :

```
mcBar1.PictureOffset = 2
```

See also :

PictureAutoSize

Purpose :

[PictureAutoSize](#) determines the Picture AutoSize of the object.

Type :

Enumerated Property

Syntax :

```
mcBar1.PictureAutoSize = value%  
value% = mcBar1.PictureAutoSize
```

Default :

mcNoneAutoSize

Comments :

- 0 - mcNoneAutoSize
- 1 - mcWidthAutoSize
- 2 - mcHeightAutoSize
- 3 - mcBothAutoSize

For mcWidthAutoSize, the .Width of the object is always adapted to the width of the .Picture.

For mcHeightAutoSize, the .Height of the object is always adapted to the height of the .Picture.

For mcBothAutoSize, the .Width and .Height of the object is always adapted to the width and height of the .Picture.

Example(s) :

```
mcBar1.PictureAutoSize = mcHeightAutoSize
```

See also :

PictureTile

Purpose :

[PictureTile](#) determines the Picture Tile of the object.

Type :

Enumerated Property

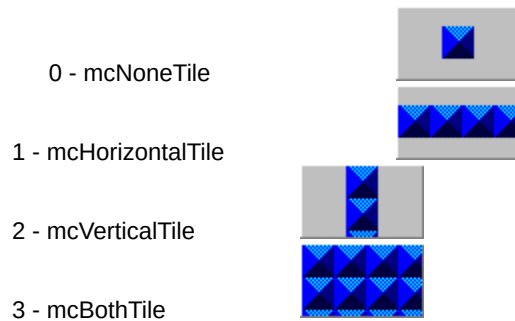
Syntax :

```
mcBar1.PictureTile = value%  
value% = mcBar1.PictureTile
```

Default :

mcNoneTile

Comments :



Example(s) :

```
mcBar1.PictureTile(1) = mcBothTile
```

See also : [BlockPictureTile](#)

Caption

Purpose :

[Caption](#) determines the caption of the object.

Type :

String Property

Syntax :

```
mcBar1.Caption = value$  
value$ = mcBar1.Caption
```

Default :

""

Comments :

Example(s) :

```
mcBar1.Caption = "mcBar Multi-Purpose Bar ActiveX control"
```

See also :

CaptionAngle

Purpose :

[CaptionAngle](#) determines the Caption Angle of the object.

Type :

Integer Property

Syntax :

```
mcBar1.CaptionAngle = value%  
value% = mcBar1.CaptionAngle
```

Default :

0

Comments :

Only usable whether the .Font is a True Type Font.

The .CaptionAngle is in degree from 0 to 360.

Example(s) :

```
mcBar1.CaptionAngle = 270
```

See also :

CaptionAutoAngle

Purpose :

[CaptionAutoAngle](#) determines the Caption Auto Angle of the object.

Type :

Enumerated Property

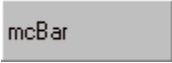
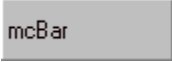
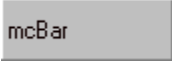
Syntax :

```
mcBar1.CaptionAutoAngle = value%  
value% = mcBar1.CaptionAutoAngle
```

Default :

mcNoneAuto

Comments :

0 - mcNoneAuto	
1 - mcUp	
2 - mcDown	

This property overrides the [.CaptionAngle](#).

Example(s) :

```
mcBar1.CaptionAutoAngle = mcUp
```

See also : [BlockTextAutoAngle](#)

ForeColor

Purpose :

[ForeColor](#) determines the ForeColor of the caption in the object.

Type :

Long Property

Syntax :

```
mcBar1.ForeColor = value&  
value& = mcBar1.ForeColor
```

Default :

vbBlack

Comments :

Example(s) :

```
mcBar1.ForeColor = vbWhite
```

See also :

CaptionFont3D

Purpose :

[CaptionFont3D](#) determines the Caption Font 3D of the object.

Type :

Enumerated Property

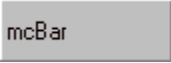
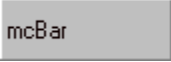
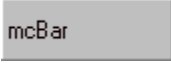
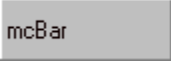
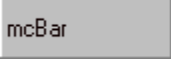
Syntax :

```
mcBar1.CaptionFont3D = value%  
value% = mcBar1.CaptionFont3D
```

Default :

mcNoneFont3D

Comments :

0 - mcNoneFont3D	
1 - mcRaisedLight	
2 - mcInsetLight	
3 - mcRaisedShadow	
4 - mcInsetShadow	

Example(s) :

```
mcBar1.CaptionFont3D = mcInsetShadow
```

See also : [BlockTextFont3D](#)

BorderStyle

Purpose :

[BorderStyle](#) determines the Border Style of the object.

Type :

Enumerated Property

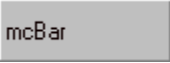
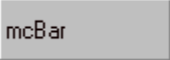
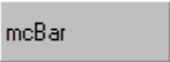
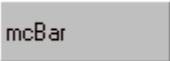
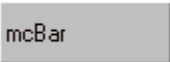
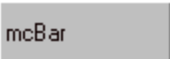
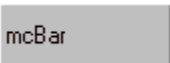
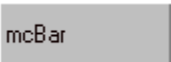
Syntax :

```
mcBar1.BorderStyle = value%  
value% = mcBar1.BorderStyle
```

Default :

mcNoneBorder

Comments :

0 - mcNoneBorder	
1 - mcFlat	
2 - mcRaised	
3 - mcInset	
4 - mcEtched	
5 - mcEtchedLight	
6 - mcBump	
7 - mcBumpLight	

Example(s) :

```
mcBar1.BorderStyle = mcBumpLight
```

See also : [BlockBorderStyle](#)

BorderWidth

Purpose :

[BorderWidth](#) determines the Border Width of the object.

Type :

Integer Property

Syntax :

```
mcBar1.BorderWidth = value%  
value% = mcBar1.BorderWidth
```

Default :

1

Comments :

Only usable whether the [.BorderStyle](#) is not mcNoneBorder.

Example(s) :

```
mcBar1.BorderWidth = 2
```

See also :

BrushStyle

Purpose :

[BrushStyle](#) determines the Brush Style of the object.

Type :

Enumerated Property

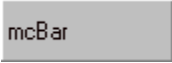
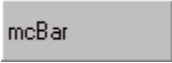
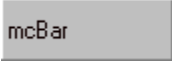
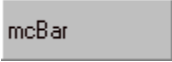
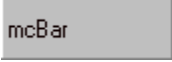
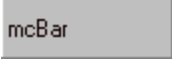
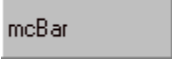
Syntax :

```
mcBar1.BrushStyle = value%  
value% = mcBar1.BrushStyle
```

Default :

mcNoneBrush

Comments :

0 - mcNoneBrush	
1 - mcHorizontal	
2 - mcVertical	
3 - mcForwardDiagonal	
4 - mcBackwardDiagonal	
5 - mcCross	
6 - mcDiagonalCross	

Example(s) :

```
mcBar1.BrushStyle = mcDiagonalCross
```

See also : [BlockBrushStyle](#)

BrushColor

Purpose :

[BrushColor](#) determines the Brush Color of the object.

Type :

Long Property

Syntax :

```
mcBar1.BrushColor = value&  
value& = mcBar1.BrushColor
```

Default :

vbBlack

Comments :

Only usable whether the [.BrushStyle](#) is not mcNoneBrush.

Example(s) :

```
mcBar1.BrushColor = vbRed
```

See also :

GradientStyle

Purpose :

[GradientStyle](#) determines the Gradient Style of the object.

Type :

Enumerated Property

Syntax :

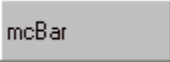
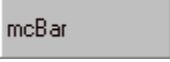
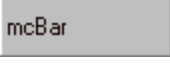
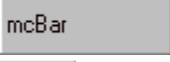
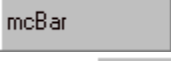
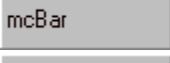
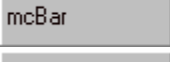
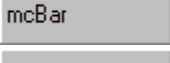
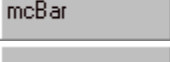
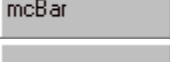

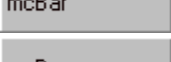
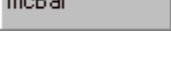
```
mcBar1.GradientStyle = value%  
value% = mcBar1.GradientStyle
```

Default :

mcNoneGradient

Comments :

.GradientColor = vbBlue | .GradientColorEnd = vbWhite

0 - mcNoneGradient	
1 - mcLeftHGradient	
2 - mcRightHGradient	
3 - mcTopVGradient	
4 - mcBottomVGradient	
5 - mcInnerHGradient	
6 - mcOuterHGradient	
7 - mcInnerVGradient	
8 - mcOuterVGradient	
9 - mcInnerCircleGradient	
10 - mcOuterCircleGradient	
11 - mcInnerRectangleGradient	
12 - mcOuterRectangleGradient	

Example(s) :

```
mcBar1.GradientStyle = mcOuterHGradient
```

See also : [GradientColor](#), [GradientColorEnd](#)

GradientGranularity

Purpose :

[GradientGranularity](#) determines the Gradient Granularity of the object.

Type :

Enumerated Property

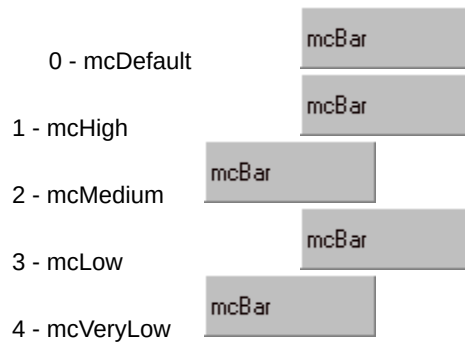
Syntax :

```
mcBar1.GradientGranularity = value%  
value% = mcBar1.GradientGranularity
```

Default :

mcDefault

Comments :



Only usable whether the [.GradientStyle](#) is not mcNoneGradient.

The mcDefault value is determined by the number of colors your system can handle.
Because the mcDefault value uses more colors than the mcVeryLow, the speed to draw this gradient is the slowest.
A good compromise is the mcMedium when you block is not too big.

Example(s) :

```
mcBar1.GradientGranularity = mcMedium
```

See also : [BlockGradientGranularity](#)

GradientColor

Purpose :

[GradientColor](#) determines the Gradient Start Color of the object.

Type :

Long Property

Syntax :

```
mcBar1.GradientColor = value&  
value& = mcBar1.GradientColor
```

Default :

vbBlue

Comments :

Only usable whether the [.GradientStyle](#) is not mcNoneGradient.

Example(s) :

```
mcBar1.GradientColor = vbWhite
```

See also : [BlockGradientColor](#)

BackColor

Purpose :

`BackColor` determines the Background Color of the object.

Type :

Long Property

Syntax :

```
mcBar1.BackColor = value&  
value& = mcBar1.BackColor
```

Default :

&HC0C0C0 (Gray)

Comments :

Example(s) :

```
mcBar1.BackColor = vbGreen
```

See also :

RivetsStyle

Purpose :

[RivetsStyle](#) determines the Rivets Style of the object.

Type :

Enumerated Property

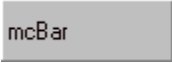
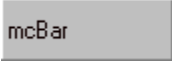
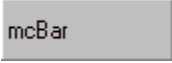
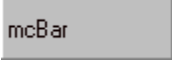
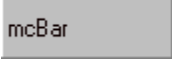
Syntax :

```
mcBar1.RivetsStyle = value%  
value% = mcBar1.RivetsStyle
```

Default :

mcNoneRivets

Comments :

0 - mcNoneRivets	
1 - mcRaisedRivets	
2 - mcInsetRivets	
3 - mcRaisedRivetsExt	
4 - mcInsetRivetsExt	

Example(s) :

```
mcBar1.RivetsStyle = mcRaisedRivetsExt
```

See also : [BlockRivetsStyle](#)

Orientation

Purpose :

[Orientation](#) determines the Orientation of all blocks in the object.

Type :

Enumerated Property

Syntax :

```
mcBar1.Orientation = value%  
value% = mcBar1.Orientation
```

Default :

mcFromLeft

Comments :

- 0 - mcFromLeft
- 1 - mcFromRight
- 2 - mcFromTop
- 3 - mcFromBottom

Example(s) :

```
mcBar1.Orientation = mcFromBottom
```

See also :

HMinValue, HMaxValue

Purpose :

[HMinValue](#) determines the Horizontal Minimum value of the object.
[HMaxValue](#) determines the Horizontal Maximum value of the object.

Type :

Integer Property

Syntax :

```
mcBar1.HMinValue = value%  
value% = mcBar1.HMinValue
```

```
mcBar1.HMaxValue = value%  
value% = mcBar1.HMaxValue
```

Default :

```
HMinValue = 0  
HMaxValue = 100
```

Comments :

Example(s) :

```
mcBar1.HMinValue = 0  
mcBar1.HMaxValue = 1440      ' minutes in a day (24 * 60)
```

See also :

VMinValue, VMaxValue

Purpose :

[VMinValue](#) determines the Vertical Minimum value of the object.
[VMaxValue](#) determines the Vertical Maximum value of the object.

Type :

Integer Property

Syntax :

```
mcBar1.VMinValue = value%  
value% = mcBar1.VMinValue
```

```
mcBar1.VMaxValue = value%  
value% = mcBar1.VMaxValue
```

Default :

```
VMinValue = 0  
VMaxValue = 256
```

Comments :

Example(s) :

```
mcBar1.VMinValue = 0  
mcBar1.VMaxValue = 256
```

See also :

BlockClickable

Purpose :

[BlockClickable](#) determines whether a block is clickable or not.

Type :

Boolean Property

Syntax :

```
mcBar1.BlockClickable([iBlock]) = value  
value = mcBar1.BlockClickable([iBlock])
```

Default :

True

Comments :

Example(s) :

```
mcBar1.BlockClickable(1) = True
```

See also :

TimerTime

Purpose :

[TimerTime](#) is used when you want to display some informations like time and date in a block of the object. The [.BlockText](#) format is automatically used.

[TimerTimeInterval](#) determines the refresh interval.

[TimerTimeEnabled](#) enables/disables the timer.

Type :

Long Property for TimerTimeInterval

Boolean Property for TimerTimeEnabled

Syntax :

```
mcBar1.TimerTimeInterval = value&  
value& = mcBar1.TimerTimeInterval
```

```
mcBar1.TimerTimeEnabled = value  
value& = mcBar1.TimerTimeEnabled
```

Default :

```
TimerTimeInterval = 1000      ' in milliseconds  
TimerTimeEnabled = False
```

Comments :

Example(s) :

```
mcBar1.TimerTimeInterval = 500  
mcBar1.TimerTimeEnabled = True
```

See also :

License Agreement

mcBar is not public domain software or free software.

mcBar is copyrighted, and all rights are reserved by its author: Michael Renard.

You are licensed to use this software on a restricted number of computers. You may copy the software to facilitate your use of it on as many computers as there are licensed users specified in the license file. Making copies for any other purpose violates international copyright laws.

You are not allowed to distribute the [mcBar.REG] file with any application that you distribute.

Disclaimer:

This software is sold AS IS without warranty of any kind, either expressed or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. The authors assume no liability for any alleged or actual damages arising from the use of this software. (Some states do not allow the exclusion of implied warranties, so the exclusion may not apply to you.)

Your use of this product indicates that you have read and agreed to these terms.

DiskDisk

Purpose :

[DiskDisk](#) is used when you want to display some informations like free and used space of a disk in a block of the object. The [.BlockText](#) format is automatically used.

[DiskDiskInterval](#) determines the refresh interval.

[DiskDiskEnabled](#) enables/disables the Disk.

Type :

Long Property for DiskDiskInterval

Boolean Property for DiskDiskEnabled

Syntax :

```
mcBar1.DiskDiskInterval = value&  
value& = mcBar1.DiskDiskInterval
```

```
mcBar1.DiskDiskEnabled = value  
value& = mcBar1.DiskDiskEnabled
```

Default :

```
DiskDiskInterval = 60000      ' in milliseconds  
DiskDiskEnabled = False
```

Comments :

Example(s) :

```
mcBar1.DiskDiskInterval = 5 * 60000&      ' 5 minutes  
mcBar1.DiskDiskEnabled = True
```

See also :

Distribution Note

When you create and distribute applications that use 'mcBar', you should install the [mcBar.OCX] in the customer's Microsoft Windows \SYSTEM or \SYSTEM32 subdirectory. The setup kit included with Visual Basic provides tools that help you write setup programs that install your applications correctly.

You are not allowed to distribute the [mcBar.REG] file with any application that you distribute.

New features

See also : [Revision History](#)

Version	Comments
-	
2.0.00	<i>no new features.</i>
1.0.52	Adds .GradientColorEnd property. Adds .BlockGradientColorEnd property. Adds .BlockClickable property. Adds .BlockAutoSize property. Adds 4 GradientStyle constant (InnerCircle, OuterCircle, InnerRectangle, OuterRectangle). Adds 4 BlockGradientStyle constant (InnerCircle, OuterCircle, InnerRectangle, OuterRectangle). Adds .SetBlockClickable method. Adds .SetBlockGradientGranularity method. Adds .SetBlockGradientColor method. Adds .SetBlockGradientColorEnd method. Adds .CaptionOffset property.
1.0.43	Initial release of mcBar .

Technical Support

Only registered users can receive support and update.

To receive support, you must specify your registration ID.

However, any report on any problem are the welcome.

The following information may be of help to you in streamlining your efforts to resolve any technical problems you may have with [mcBar](#).

GPF?

If you are getting a GPF (General Protection Fault), write down the information that is displayed when the error occurs. Also, make a note of what your code was doing (in general terms.)

ISOLATE IT

Try to isolate the cause of the error. If at all possible, step through your code with F8 and F9. Try to find the one line of code that is causing the error.

SCALE IT DOWN

If at all possible, try to reproduce the problem in a small test program that you can send in. Send your test on CompuServe.

Update

You can download the update of all of my products on the following network :

On my homepage :

<http://ourworld.compuserve.com/homepages/alpouda/homepage.htm>

On CompuServe :

MSBASIC forum

On Internet :

mcBar100.ZIP (ftp.winsite.com/pub/pc/win95/programr/vbasic OR
<http://www.geocities.com/SiliconValley/Way/7409>)

CompuServe Mail:

Name : Michael RENARD

CIS : 100042,3646

Internet : 100042.3646@compuserve.com

I'm on CompuServe one time a day.

mcBar : mcBar.OCX : 2.0.0 (03/07/1998)

Revision history

See also : [New Features](#)

Version	Comments
-	
2.0.00	Correction of problems with installation.
1.0.52	<i>no revision.</i>
1.0.43	Initial release of mcBar .

Overview

[mcBar](#) is a multi-purpose bar activex control for VB 5.0 or higher, for Access 97.

It's not a simply bar.

You can use it as :

1. a progress bar;
2. a complex progress bar;
3. a pictured progress bar;
4. a schedule bar;
5. a level indicator;
6. a status bar (with automatic date and time display)
7. a disk monitor (with automatic free and used space)
8. a puzzle game;
9. a chart;
10. and what you want.

Installation

Demonstration version :

The files mcBar.OCX and mcBar.HLP should be copied in your WIN95\SYSTEM and/or WINNT\SYSTEM32 directory.

Registered version :

The files mcBar.OCX, mcBar.HLP should be copied in your WIN95\SYSTEM and/or WINNT\SYSTEM32 directory.
The file mcBar.REG should be inserted in your registry.

Distribution note:

When you create and distribute applications that use [mcBar](#), you should install the file 'mcBar.OCX' in the customer's Microsoft Windows \SYSTEM or \SYSTEM32 subdirectory. The Visual Basic Setup Kit included with the Professional VB product provides tools to help you write setup programs that install you applications correctly.

You are not allowed to distribute [mcBar.REG] file with any application that you distribute.

Registration

'mcBar' Registration Benefits :

- Create your application easier and faster
- Create a smaller application
- Accelerate the speed of your application
- Full support for one year

Registering 'mcBar' on CompuServe

- 1) On CompuServe [GO SWREG](#)
- 2) Choose Register Shareware.
- 3) 'mcBar' SWREG ID is : #[16817](#). (price is \$[25.00](#))

Registering 'mcBar' on Internet

- 1) Use the program REGISTER.EXE or download REGISTER.ZIP from my Internet Homepage.
- 2) Select the product mcBar.
- 3) Send by e-mail or fax or postal.

As soon as I receive notification of your registration (usually 1 - 3 days) I will send you out via e-Mail a registration file for each single user license that you've asked, or site license or world license.

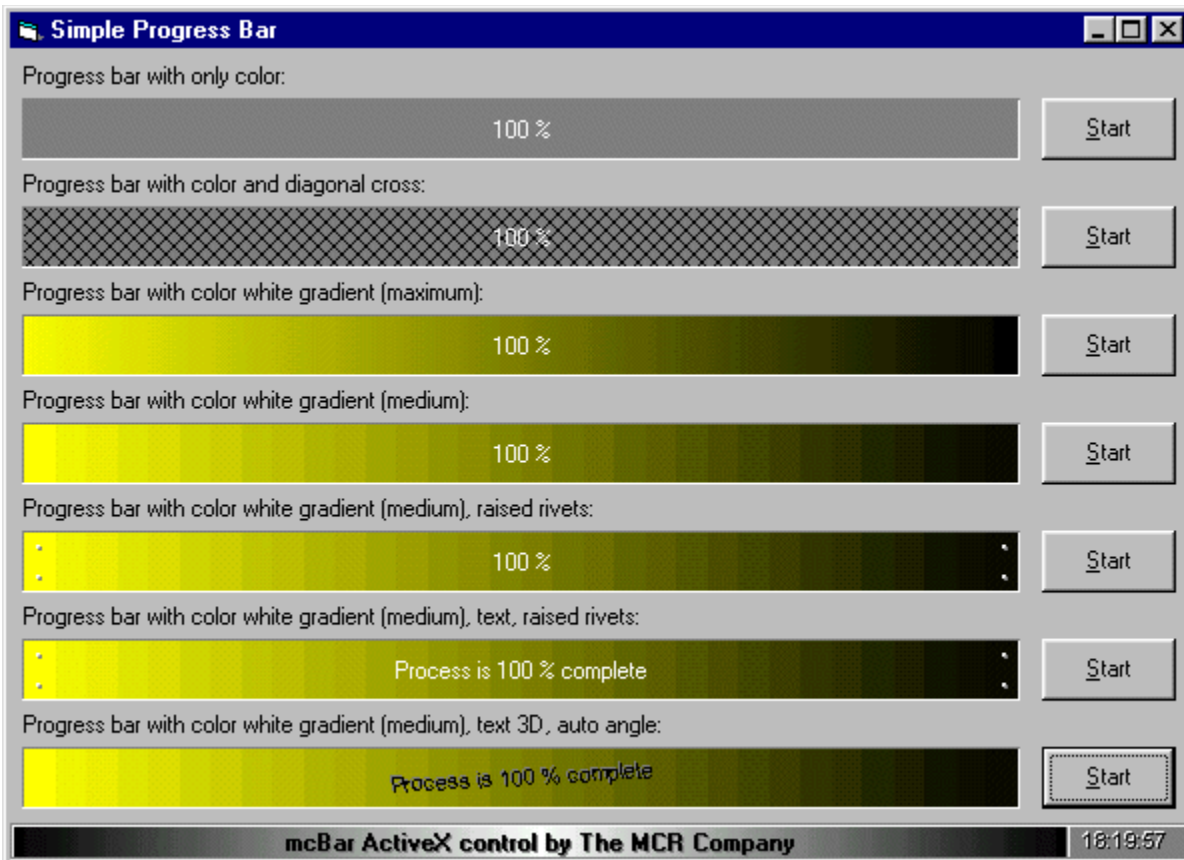
You also qualify to receive new versions of 'mcBar' during one year.

This price is much a contribution to my works that a payment.

When you register 'mcBar', you help me to develop better products and others products.

'mcBar' is written in VB 5.0 SP3.

'mcBar' can only be used with Visual Basic 5.0 or higher under Windows 95 and Windows NT.



BlockAutoSize

Purpose :

[BlockAutoSize](#) determines whether a block is resized automatically when the size of the object has been changed.

Type :

Boolean Property

Syntax :

```
mcBar1.BlockAutoSize([iBlock]) = value  
value = mcBar1.BlockAutoSize([iBlock])
```

Default :

True

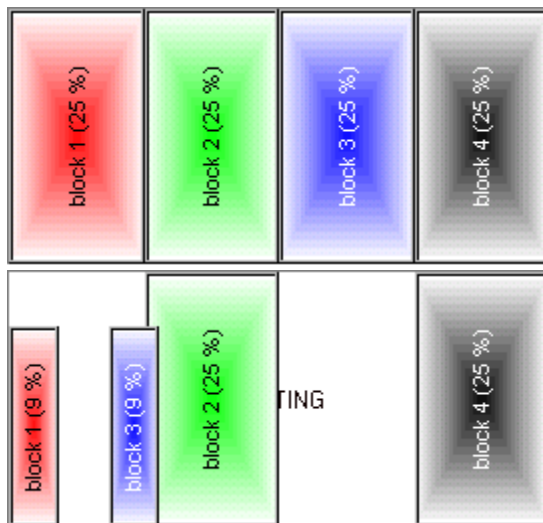
Comments :

Example(s) :

```
mcBar1.BlockAutoSize(1) = False
```

The four blocks have the .BlockAutoSize set to True
False

The blocks 1 and 3 have the .BlockAutoSize set to



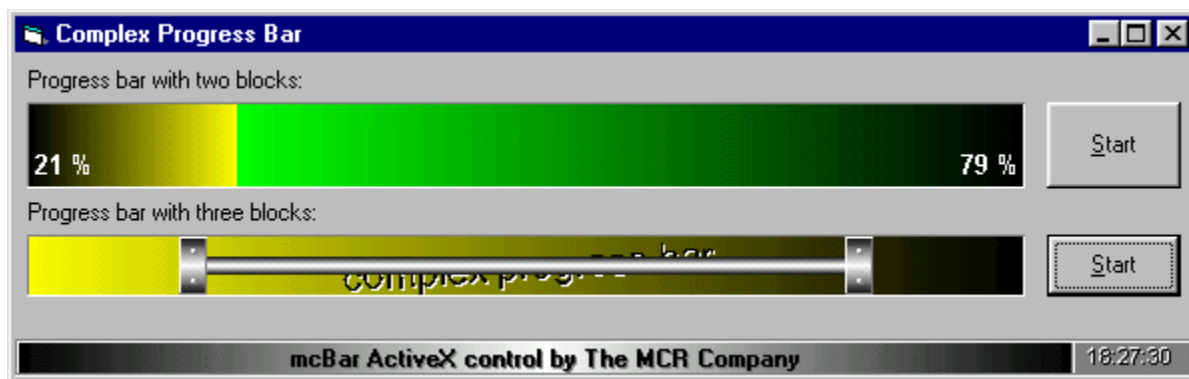
The horizontal percentages are calculated with
calculated
.HMinValue and .HMaxValue properties

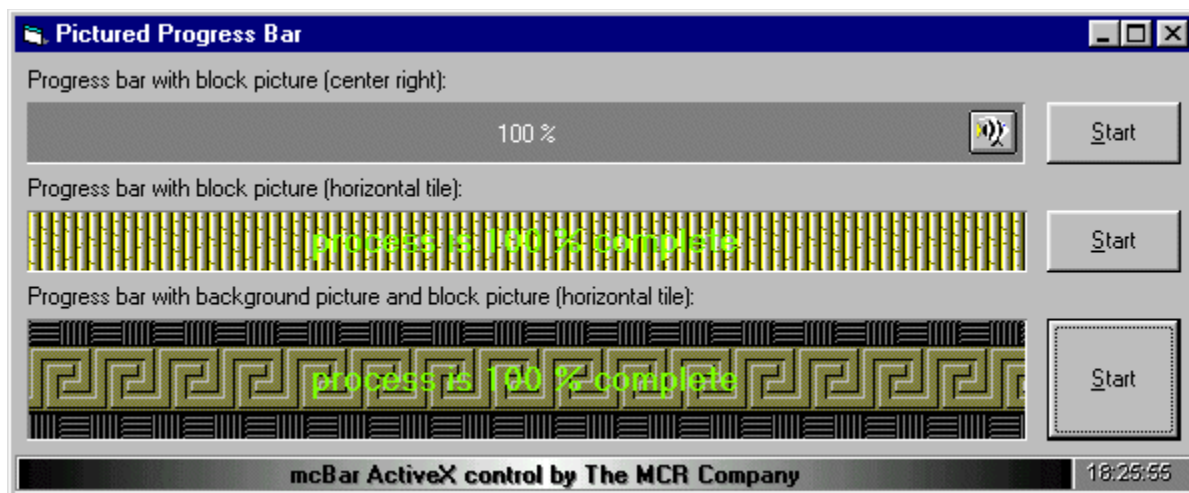
The horizontal percentages for block 1 and 3 are
with the .ScaleWidth property.

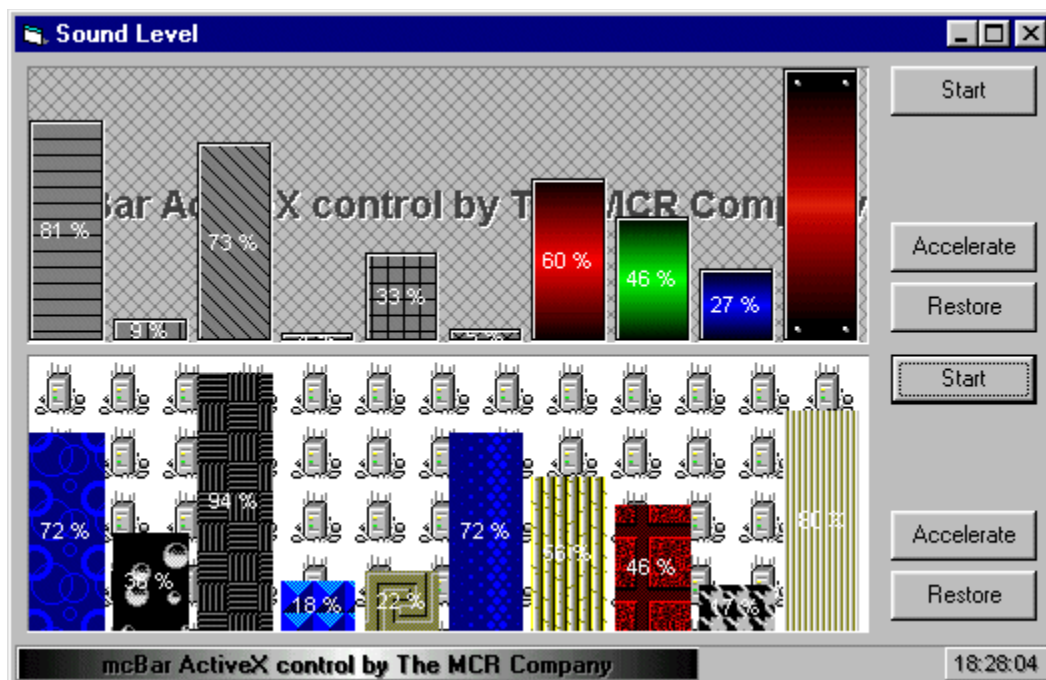
The vertical percentages are calculated with
.VMinValue and .VMaxValue properties

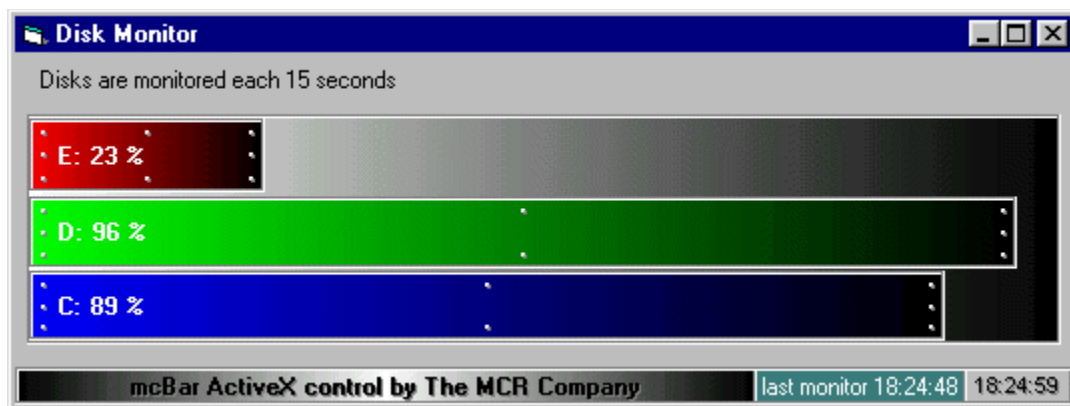
The vertical percentages for block 1 and 3 are calculated
with the .ScaleHeight property.

See also :









(About)
(Custom)
(Name)
Align
BackColor
Blocks
BorderStyle
BorderWidth
BrushColor
BrushStyle
Caption
CaptionAlignment
CaptionAngle
CaptionAutoAngle
CaptionFont3D
CaptionOffset
DisabledColor
DragIcon
DragMode
Enabled
Font
ForeColor
GradientColor
GradientColorEnd
GradientGranularity
GradientStyle
Height
HelpContextID
HMaxValue
HMinValue
Index
IndexBlock
Left
Negotiate
Orientation
Picture
PictureAlignment
PictureAutoSize
PictureOffset
PictureTile
Redraw
RivetsStyle
TabIndex
TabStop
Tag
TimerDiskEnabled
TimerDiskInterval
TimerTimeEnabled
TimerTimeInterval
ToolTipText
Top
Visible
VMaxValue
VMinValue
WhatsThisHelpID
Width

Events

BlockClicked

TimerDisk

TimerTime

Methods

About
AddBlock : adds a block in the block collection.
ClearAll : clears all blocks.
ClearBlockBar : clears all bars.
DeleteBlock : deletes the specified block.
Refresh : refresh the object.
RestoreSetting : restores all blocks saved with SaveSetting.
SaveSetting : saves all blocks.
SetBlockAutoSize : sets the .BlockAutoSize for all blocks.
SetBlockBarBottom : sets the .BlockBarBottom for all blocks.
SetBlockBarTop : sets the .BlockBarTop for all blocks.
SetBlockBorderStyle : sets the .BlockBorderStyle for all blocks.
SetBlockBrushColor : sets the .BlockBrushColor for all blocks.
SetBlockBrushStyle : sets the .BlockBrushStyle for all blocks.
SetBlockClickable : sets the .BlockClickable for all blocks.
SetBlockColor : sets the .BlockColor for all blocks.
SetBlockGradientStyle : sets the .BlockGradientStyle for all blocks.
SetBlockGradientGranularity : sets the .BlockGradientGranularity for all blocks.
SetBlockGradientColor : sets the .BlockGradientColor for all blocks.
SetBlockGradientColorEnd : sets the .BlockGradientColorEnd for all blocks.
SetBlockPicture : sets the .BlockPicture for all blocks.
SetBlockRivetsStyle : sets the .BlockRivetsStyle for all blocks.
SetBlockShapeStyle : sets the .BlockShapeStyle for all blocks.
SetBlockText : sets the .BlockText for all blocks.
SetBlockTextAlignment : sets the .BlockTextAlignment for all blocks.
SetBlockTextAngle : sets the .BlockTextAngle for all blocks.
SetBlockTextAutoAngle : sets the .BlockTextAutoAngle for all blocks.
SetBlockTextOffset : sets the .BlockTextOffset for all blocks.
SetBlockVisible : sets the .BlockVisible for all blocks.

BlockVisible

Purpose :

[BlockVisible](#) determines whether a block is visible or not.

Type :

Boolean Property

Syntax :

```
mcBar1.BlockVisible([iBlock]) = value  
value = mcBar1.BlockVisible([iBlock])
```

Default :

True

Comments :

Example(s) :

```
mcBar1.BlockVisible(1) = True
```

See also :

Blocks

Purpose :

[Blocks](#) sets/retrieves the number of blocks in the object.

Type :

Integer Property

Syntax :

```
mcBar1.Blocks = value%  
value = mcBar1.Blocks
```

Default :

0

Comments :

When you change the value of .Blocks, you create or remove blocks.

By example, if you've 10 blocks and you set the new value to 15, the object can handle 15 blocks (5 blocks are been added).

By example, if you've 10 blocks and you set the new value to 5, the object can handle 5 blocks (5 blocks are been removed).

Example(s) :

```
mcBar1.Blocks = 10
```

See also :

BlockGradientColorEnd

Purpose :

[BlockGradientColorEnd](#) determines the Gradient End Color of a block.

Type :

Long Property

Syntax :

```
mcBar1.BlockGradientColorEnd([iBlock]) = value&  
value& = mcBar1.BlockGradientColorEnd([iBlock])
```

Default :

vbBlock

Comments :

Only usable whether the [.BlockGradientStyle](#) is not mcNoneGradient.

Example(s) :

```
mcBar1.BlockGradientColorEnd(1) = vbWhite
```

See also :

IndexBlock

Purpose :

[IndexBlock](#) sets/retrieves the current block.

Type :

Integer Property

Syntax :

```
mcBar1.IndexBlock = value%  
value = mcBar1.IndexBlock
```

Default :

Comments :

The .IndexBlock can be used to reference a block.

Example(s) :

```
mcBar1.IndexBlock = 2
```

See also :

TimerTime

Purpose :

[TimerTime](#) is raised each time a .TimerTimeInterval is elapsed.

Type :

Event

Syntax :

```
Private Sub mcBar1_TimerTime()
```

Default :

Comments :

Example(s) :

```
Private Sub mcBar1_TimerTime()  
    If (Time$ = "12:00:00") Then  
        mcStatusBar.BlockText(1) = "It's time to break"  
    End If  
End Sub
```

See also :

TimerDisk

Purpose :

[TimerDisk](#) is raised each time a .TimerDiskInterval is elapsed.

Type :

Event

Syntax :

```
Private Sub mcBar1_TimerDisk()
```

Default :

Comments :

Example(s) :

```
Private Sub mcBar1_TimerDisk()  
    mcStatusBar.BlockText(3) = "last monitor " & Time$  
End Sub
```

See also :

BlockClicked

Purpose :

[BlockClicked](#) is raised each time the mouse has been clicked in a block.

Type :

Event

Syntax :

```
Private Sub mcBar1_BlockClicked(iBlock As Integer)
```

Default :

Comments :

If there is no block under the mouse or if the block is not clickable or if the block is not visible, the returned value is -1.

Example(s) :

```
Private Sub mcBar1_BlockClicked(iBlock As Integer)
    If (iBlock <> -1) Then mcBar1.BlockVisible(iBlock) = False
End Sub
```

See also :

AutoSize
BarBottom
BarLeft
BarRight
BarTop
BorderStyle
BorderWidth
BrushColor
BrushStyle
Clickable
Color
GradientColor
GradientColorEnd
GradientGranularity
GradientStyle
HPercentage
Key
OnTop
Picture
PictureAlignment
PictureOffset
PictureTile
RivetsStyle
ShapeStyle
Text
TextAlignment
TextAngle
TextAutoAngle
TextColor
TextFont
TextFont3D
TextOffset
Visible
VPercentage

Properties

Basis
Blocks

GradientColorEnd

Purpose :

[GradientColorEnd](#) determines the Gradient End Color of the object.

Type :

Long Property

Syntax :

```
mcBar1.GradientColorEnd = value&  
value& = mcBar1.GradientColorEnd
```

Default :

vbBlack

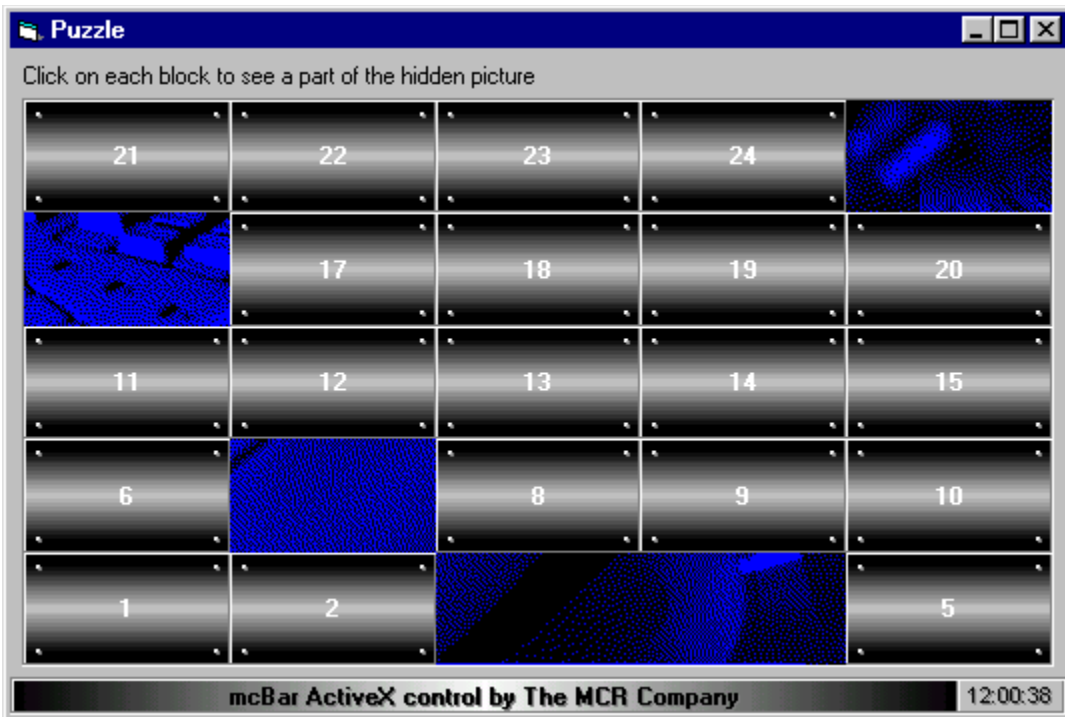
Comments :

Only usable whether the [.GradientStyle](#) is not mcNoneGradient.

Example(s) :

```
mcBar1.GradientColorEnd = vbWhite
```

See also : [BlockGradientColorEnd](#)



CaptionOffset

Purpose :

[CaptionOffset](#) determines the Caption Offset in the object.

Type :

Integer Property

Syntax :

```
mcBar1.CaptionOffset = value%  
value% = mcBar1.CaptionOffset
```

Default :

3

Comments :

Only usable whether the [.CaptionAlignment](#) is not mcCenterCenter.

Example(s) :

```
mcBar1.CaptionOffset = 7
```

See also : [BlockTextOffset](#)

Acknowledgement

Thanks to [Jim Houser](#) to help me to resolve problems with installations.

