

VBMax 3D Effects DLL

Copyright © 1995 - 1996 Mike Stanley. All Rights Reserved

VBMax—Visual Basic to the Max

Table of Contents

- Description
- Version Information
 - Version 1.2
 - Version 1.1
- Registration
- Important—Use at Your Own Risk!
- Installation
- Frequently Asked Questions
- Technical Support
- Gronked Files
- Packing List
- How to use the DLL
- Methods Summary
- Properties Summary

Description

Add the 3D effects that Microsoft forgot to your VB 4.0 applications.

Written entirely in VB 4.0, VBMax3D.dll is a fully functional in-process OLE server that provides you with a variety of 3D effects without the overhead of third-party controls.

VBMax3D.dll is Windows 95 color-scheme aware and contains methods and properties for creating panels, borders, frames, lines, drop shadows, status bars, progress meters and many different types of text effects.

When you register, you get the complete, fully-commented source code so you can see how it works and modify it any way you want to suit your own needs.

Shareware: \$5.

The archive file is freely distributable.

Version Information

Version 1.2

Updated this documentation to include a Frequently Asked Questions section.

Version 1.1

Changed the demo program by replacing the Windows common slider control with scroll bars because users of VB Standard Edition did not have COMCTL32.OCX.

Registration

This is a shareware utility which you can register for US \$5 per copy. Send payment to:

Mike Stanley

25 Lansdowne Street
Manchester
NH 03103
USA

In return, I will give you a registration ID and an encryption key which will allow you to unlock the source code from the file 'VBMax3D Source Code.grk'—***don't forget to include your Internet e-mail address***. Better still, also send me an e-mail to let me know that your registration is on the way.

Any future updates to the software will use the same encryption key, so you can always get them at no additional cost.

If you don't have an e-mail address, you can receive the registration ID and encryption key via snail-mail by sending me a postage-paid self-addressed envelope.

Important—Use at Your Own Risk!

While every effort has been made to ensure a reliable, high-quality product, you should note that this software is provided 'as is' without any kind of warranty whatsoever, neither explicit nor implied.

In order to use this software, you must agree 100% that I will not be held liable in any way, shape or form (not even a little bit) for anything untoward that befalls anyone or anything, either directly or indirectly, as a result of using this software no matter how calamitous, disastrous or inauspicious the occasion may be.

Your use of this software constitutes acceptance of these terms.

Installation

Copy the files to a folder of your choice and edit Reg.bat so that it contains the correct path for the regsvr32.exe program that comes with VB 4.0. Please note that regsvr32.exe is not copied to your hard-drive automatically when you install VB – you can find it in the Tools\Pss folder on your VB CD.

Run the edited Reg.bat which will update the Windows registry so that it knows about VBMax3D.dll.

Go into VB, load the demo program and run it.

If you want to use the DLL in your own programs, you have to add a reference to it as follows:

From VB, select Tools from the menu.

Then select References...

Locate VBMax 3D Effects in the list and click the check box so that an X appears in it. Click OK.

The OLE server is now visible in the Object Browser.

Frequently Asked Questions

- Q.** I try to run your demo program but it keeps telling me that OLE cannot create the object. Why?
A. You have not registered VBMax3D.dll with Windows. Please refer to the installation instructions above.
- Q.** I have searched my hard-drive but cannot find regsvr32.exe – what's the scoop?
A. Regsvr32.exe is not copied to your hard-drive automatically when you install VB – you can find it in the Tools\Pss folder on your VB CD.

Q. Why don't you just create a regular installation program?

A. One thing I don't like about VB 4 is the amount of extra baggage that must be included with the installation of even the simplest programs—this translates to long download times from online services. Seeing as VBMax3D.dll is meant for programmers who already have all the necessary software installed on their computers, I decided that manually installing the DLL was a fair tradeoff to racking up your online connect time.

Technical Support

Technical support is available via e-mail. Please send questions, comments, criticisms, etc. to 74632.2227@compuserve.com.

Gronked Files

These are encrypted files, created using the VBMax Gronk Meister, that contain one or more other files. In this case, the file 'VBMax3D Source Code.grk' contains the complete, commented VB 4.0 source code for VBMax3D.dll.

The source files may be extracted using the included VBMax Degronker utility. Before you can do that, however, you need to know the encryption key. This key will be given to you when you register VBMax3D.dll.

Packing List

Read Me.rtf/txt	-	The document you are reading.
3Ddemo.vbp (and related files)	-	Source code for demo program showing the DLL in action.
Reg.bat	-	Registers the DLL with Windows (you have to edit the path).
Unreg.bat	-	Unregisters the DLL with Windows (you have to edit the path).
VBMax3D Source Code.grk	-	Source code for the DLL in gronked format.
Degronker.exe	-	Utility to extract the source code from the gronked file.

How to use the DLL

When you include a reference to VBMax3D.dll in VB as described earlier, you will have access to a class called cls3D. When you select this class using the Object Browser, you will see the methods and properties available to you. You can see a brief description of what each one does by clicking on them.

In your application, you can create one or more objects from this class module. How many you choose depends on how many effects you want and whether or not they share the same attributes. For example, if you set the style property to gnRAISED, all effects that use this property will use the same style.

If you want to use different styles you have two options:

1. Change the style property each time you use it.
2. Create different objects with different styles.

It's your call—yer pays yer money and yer takes yer choice. Multiple objects all use the same code but keep their own copy of the variables.

Objects are created from this class by using code like this:

```
Dim mo3D As New cls3D.
```

You create 3D effects by drawing regular 2D controls on a form and then passing them to methods in the object created by cls3D.

These methods then create the desired 3D effect by drawing on the form at the location of the 2D control. The 2D

controls act as place-holders for the resulting 3D effect and allow you to see what you are doing as you design your form.

You can get a good idea of how the DLL works by examining the source code for the demo program. More detailed information is contained within the DLL source code.

Methods Summary

Object.Border Control

Draws a border around a control.

Control: Control around which border is to be drawn.

Example: mo3D.Border Text1

Object.Caption Control [, FontSwitch]

Draws text using a variety of 3D effects.

Control: Label control used as place-holder and source for text.

FontSwitch: Optional. If set, uses the font attributes of the Label control
Settings: gnUSE_CONTROL_FONT

Example 1: mo3D.Caption Label1

Example 2: mo3D.Caption Label1, gnUSE_CONTROL_FONT

Object.DropShadow Control

Draws a drop-shadow behind a control.

Control: The control behind which the shadow is to be drawn.

Example: mo3D.DropShadow Text1

Object.FormBorder Form

Draws a border around the inside edge of a form.

Form: The form on which to draw the border.

Example: mo3D.FormBorder Me

Object.Frame Control, Caption

Draws a frame around a control.

Control: Control around which frame is to be drawn.

Caption: Frame caption.

Example: mo3D.Frame picFrameColor, "Color"

Object.Line3D Control

Draws a 3D line.

Control: Line control used as a place-holder.

Example: mo3D.Line3D Line1

Object.Panel Control

Converts a control to a panel.

Control: Control used as place-holder for panel.

Example: mo3D.Panel Shape1

Object.PercentMeter Control, Percent

Converts a Label control to a percentage meter.

Control: Label control used as a place-holder

Percent: The percentage to display.

Example: mo3D.PercentMeter lblStatus, 25

Object.StatusBar Control, Status

Displays a message in a status bar.

Control: Place-holder for status bar.

Status: The message to be displayed.

Example: mo3D.StatusBar lblStatus, "Connecting to network..."

Properties Summary

Object.Alignment = Alignment

Specifies the justification of the caption in a frame.

Alignment: One of the following:
gnLEFT_JUSTIFY
gnRIGHT_JUSTIFY
gnCENTER

Example: mo3D.Alignment = gnLEFT_JUSTIFY

Object.Depth = Depth

Object.Width = Width

Control the appearance of panels and borders.

Example 1: mo3D.Width = 5
Example 2: mo3D.Depth = 2

Object.FillColor = Color

The color of the bar in the percentage meter.

Example: mo3D.FillColor = vbBlue

Object.FontBold = Boolean

Object.FontItalic = Boolean

Object.FontName = FontName

Object.FontSize = FontSize

Object.FontStrikethru = Boolean

Object.FontUnderline = Boolean

Specifies the font attributes for 3D text effects.

Example 1: mo3D.FontName = cboFontNames.Text
Example 2: mo3D.FontSize = 10
Example 3: mo3D.FontBold = True
Example 4: mo3D.FontItalic = True
Example 5: mo3D.FontStrikethru = False
Example 6: mo3D.FontUnderline = False

Object.ForeColor = ForeColor

Specifies the color to be used for the text.

ForeColor: Color value

Example: mo3D.ForeColor = vbBlue

Object.Offset = Offset

The distance between a control and its drop shadow.

Example: mo3D.Offset = 5

Object.Shadow = Shadow

Specifies whether to use light or dark shadows on various 3D effects.

Shadow: One of the following:

gnLIGHT_SHADOW	Use light shadows
gnDARK_SHADOW	Use dark shadows

Example: mo3D.Shadow = gnLIGHT_SHADOW

Object.ShowPercent = Boolean

Specifies whether or not to show the numbers on the percentage meter.

Example: `mo3D.ShowPercent = True`

Object.SolidMeter = Boolean

Specifies whether or not to show the percentage meter as a solid bar.

Example: `mo3D.SolidMeter = True`

Object.Style = Effect

Specifies the type of 3D effect.

Effect:	One of the following :	No 3D effect.
	<code>gnNORMAL</code>	Creates a raised 3D look.
	<code>gnRAISED</code>	Gives a recessed 3D look.
	<code>gnRECESSED</code>	Gives text a raised embossed look.
	<code>gnEMBOSSSED_RAISED</code>	Gives text a recessed embossed look.
	<code>gnEMBOSSSED_RECESSED</code>	Makes text look like its floating over the form.
	<code>gnFLOATING</code>	Shows only the right edges of the text.
	<code>gnOUTLINE_RIGHT</code>	Shows only the left edges of the text
	<code>gnOUTLINE_LEFT</code>	Displays raised text from a different perspective.
	<code>gnPROJECTED</code>	

Example: `mo3D.Style = gnRAISED`

Mike Stanley
25 Lansdowne Street
Manchester
New Hampshire
USA
E-Mail: 74632.2227@compuserve.com