

VB Project Analyser Help



VB Project Analyser

BETA

Beta 0.3

Copyright © 1993, 1994 Mike Chapman, TechDesign
All Rights Reserved



Help File Build 0.3.00 15 February 1994



Introduction

- ▶ [What is VB Project Analyser ?](#)
- ▶ [Product Support & Contact Information](#)

VB Project Analyser is currently under development and is not yet available for general use. It is in an advanced Beta state and feedback and suggestions are very welcome. The usual stuff about "no warranty is expressed or implied" should be observed. If you want to pass on VB Project Analyser you must make sure that all the files remain in their original forms.

▶ **Guide**

- ▶ [How to use VB Project Analyser](#)
- ▶ [Analysing the Project](#)
- ▶ [Cross-Referencing](#)
- ▶ [Find Text](#)
- ▶ [Reports](#)
- ▶ [Options](#)
- ▶ [Dialogue Boxes](#)

▶ **Reference**

- ▶ [Development Wishlist](#)
- ▶ [New / Modified for Beta 0.2](#)
- ▶ [New / Modified for Beta 0.3](#)

▶ **What is VB Project Analyser ?**

VB Project Analyser is an analysis tool for Visual Basic projects. It allows you to visually see how your project is constructed along with a variety of statistics, control and property details and file version information. Interactive cross-referencing and reports of unreferenced procedures and functions are also included.

Before it can be released as Shareware there are a number of other features and enhancements to be added such as procedure calling trees and printing.

Please see the [Product Support & Contact Information](#) topic for details on how to pass on your feedback and suggestions.

▶ Product Support & Contact Information

If you have any problems using this Beta copy of VB Project Analyser, or encounter problems which you believe may be software bugs, please contact me and I will do my best to assist you in solving the problem.

I am extremely interested in feedback and suggestions. Please see the [development wishlist](#) topic for some current suggestions. Your contributions to its evolution will help ensure it meets your needs in the future.



Mike Chapman,

27 Gorse Drive,
Smallfield,
Surrey,
RH6 9GJ,
England.



CompuServe: **100030,351**

Internet:
100030.351@compuserve.com

Email is the preferred method of contact.

▶ Development Wishlist

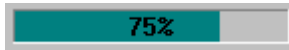
Many thanks to all those users who have provided comment, feedback and suggestions on Betas 0.1 and 0.2 !

The following is a list of suggestions for additions to *VB Project Analyser* before it is released. Please feel free to comment on these suggestions. I would be especially interested if you feel any are unnecessary, require more features, or you would like something which isn't here. Please don't assume someone else has already suggested it - I really would like to hear it.

- Variable and Constant lists with types at Global, Form and Module and Local levels. Cross-referenced as Functions and Procs are and distinguished between referenced and modified.
 - ▶ Print report(s) to disk or printer or clipboard. Selective.
 - ▶ Visual hierarchy display (calling tree) for Functions and Procs which details what calls what.
 - ▶ Full report on unreferenced Variables, Constants, and APIs.
 - ▶ Sort controls list based on number of controls. Descending order to.
 - ▶ Provide statistics on Functions and Procs as well. Parameter count. Sort on these too.
 - ▶ Allow option from property window to lookup xref for references to properties and whether they are modified or just read.
 - ▶ Display module dependencies.
 - ▶ Automatically generate documentation on project in Help file format.
 - ▶ Code Browser to navigate around the Project and display the code for any Function or Procedure in any Form or Module and also direct from the Code Window. List of Functions/Procs available globally.
 - ▶ Store statistics for review.
 - ▶ Search disk for duplicate DLLs, VBXs, etc.
 - ▶ List name of modules APIs declared in.
 - ▶ Handle comments above the subroutine declaration line.
 - ▶ Show where specific controls are used in the project.
 - ▶ Allow some sort of editing of source files ?
 - ▶ Allow customisation of the colours for the Code Window.
 - ▶ Add line totals to Event list.
 - ▶ Sort All Procedures / Unreferenced report and Procs, Events and Functions by line totals and modules.
 - ▶ Cope with control arrays.

▶ How to use VB Project Analyser

1. Open a VB project to analyse. This will complete the Project Statistics, Project Controls Listing and the Main List details. As it is analysed the percent bar will indicate its progress.




2. Use the Main List to start analysing the Project.
 - ▶ The main window can be resized and repositioned and will be saved between sessions.
 - ▶ The Status Bar at the bottom of the main window displays context-sensitive help information as the mouse pointer passes over controls.

Double click a control to display the properties and an event procedure to show code

► Opening a VB Project

There are a variety of ways to open a Visual Basic project for analysis.

1. Choose the **Open** button  or the **File|Open** menu command. From the standard open dialog box choose a Visual Basic project file (extension **.MAK**).
2. You can drag and drop a project file from a file manager onto the main window to open it. You will be asked to confirm the action.
3. You can open a project for analysis when you start VB Project Analyser by adding its fully qualified file name to the command line.



VB Project Analyser can only read files saved as **Text** but will notify you if any Forms or Modules are not saved as text. You may then cancel the analysis or continue loading the project.

▶ Analysing the Project

The Main List is used to start analysis actions on the Visual Basic project. There are four sections to the Main List which detail the file composition for the Project:

- ▶ **Forms**
- ▶ **Modules**
- ▶ **External Libraries**
- ▶ **Custom Controls**

Clicking on a **Form** or **Module** file name in the list will display statistic information for that file.

If there is a **+** symbol next to the **Form** or **Module** file name then you can expand it by clicking on the symbol to display further sub-categories:

- Declarations
- ▶ Sub Procs
- ▶ Functions

Forms can have in addition:

- ▶ Controls
- ▶ Event Procs

Clicking on each of these will display a list of those items present in that Form or Module.

Click the **=** symbol to collapse the file view. Double-clicking on the file name will also expand or collapse the file view. You can also use the buttons below the Main List to expand or collapse the view for all the Forms and Modules.

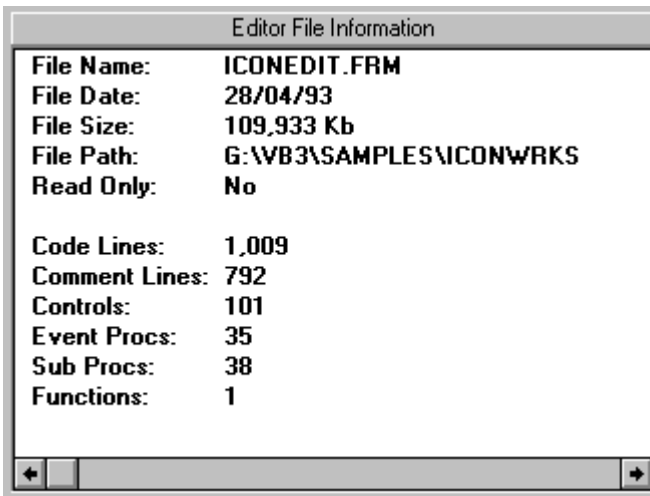


The **External Libraries** section lists the libraries for any API functions and subroutines that are declared in the project. Clicking on a file will provide version information if present and file statistics. Expanding the file using the ▶ will display the associated Declarations line which will list those API functions and subroutines declared from that particular library.

The **Custom Controls** section lists all the VBX files used in the project. Clicking on a file will provide version information if present and file statistics.

► Form or Module Statistic Information

Clicking on a Form or Module file name in the Main List will display various statistics for that file and the file length, date and path details.



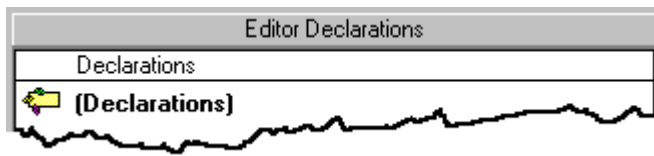
The Code Lines total is the actual coding lines in the project - it does not include comments, blank lines or any of the control definitions.

► If the

Use Form Control Names option in the Options|General menu command is not checked then the **Form Name** will appear in the above list instead of the **File Name**.

▶ Form or Module Declarations

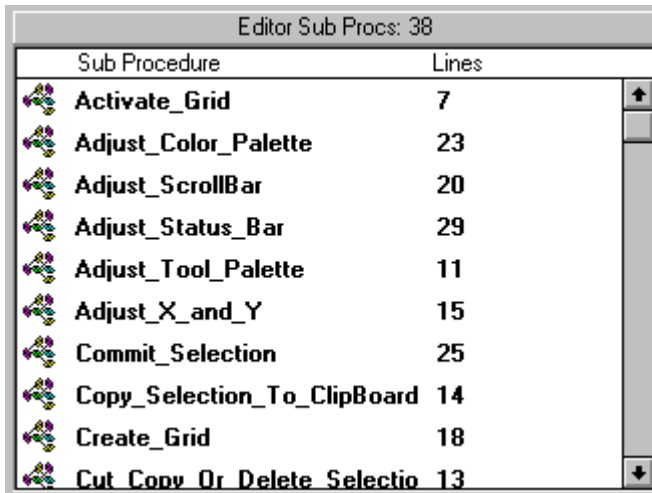
Clicking on the **Declarations** line for a particular Form or Module in the Main List will display the Declarations section in that file.













▶ Double clicking on the Declarations line will display the form or module declarations code.

▶ Sub Proc List for Form or Module

Clicking on the **Sub Procs** line for a particular Form or Module in the Main List will display those Subroutines that appear in that file. The total number present is given on the heading bar.



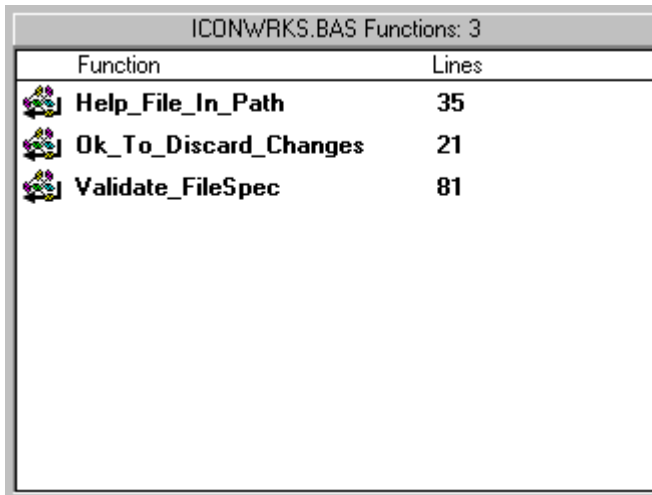
Sub Procedure	Lines
 Activate_Grid	7
 Adjust_Color_Palette	23
 Adjust_ScrollBar	20
 Adjust_Status_Bar	29
 Adjust_Tool_Palette	11
 Adjust_X_and_Y	15
 Commit_Selection	25
 Copy_Selection_To_ClipBoard	14
 Create_Grid	18
 Cut Copy Or Delete Selectio	13




The Lines column notes how many coding lines appear in that procedure.

- ▶ The Procedures can be visually identified by the image representation given.
- ▶ Double clicking on a Sub Proc line in the above list with the **left mouse button** will cross-reference that Subroutine and display which routines it is called from.
- ▶ Clicking with the **right mouse button** will display the code for that particular Subroutine.

▶ **Function List for Form or Module**

Clicking on the **Functions** line for a particular Form or Module in the Main List will display those Functions that appear in that file. The total number present is given on the heading bar.



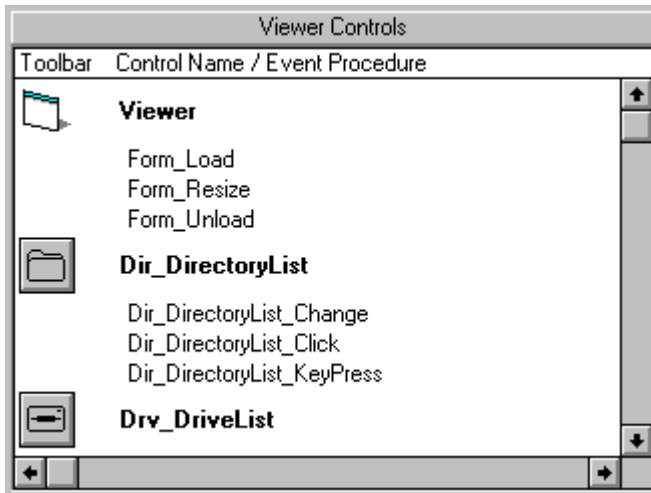
Function	Lines
 Help_File_In_Path	35
 Ok_To_Discard_Changes	21
 Validate_FileSpec	81

The Lines column notes how many coding lines appear in that procedure.

- ▶ The Functions can be visually identified by the image representation given.
- ▶ Double clicking on a Function line in the above list with the **left mouse button** will cross-reference that Function and display which routines it is called from.
- ▶ Clicking with the **right mouse button** will display the code for that particular Function.

▶ Form Controls List

Clicking on the **Controls** line for a particular Form in the Main List will display all the Controls in that Form along with their Name, toolbar image and any associated Event Procedures. Control arrays are also noted.



- ▶ Double clicking on a control will display that control's properties.
- ▶ Double clicking on an event procedure will display the code for that particular Event.
- ▶ You can resize the columns in the list and the positions will be saved for the next session.
- ▶

Note: Not all third party custom control toolbar images can be determined automatically. This is because the function used to reach the image identifier in the VBX is not required to be implemented by the third party supplier at the moment so not all do. If the toolbar image cannot be determined then the following image is displayed instead:

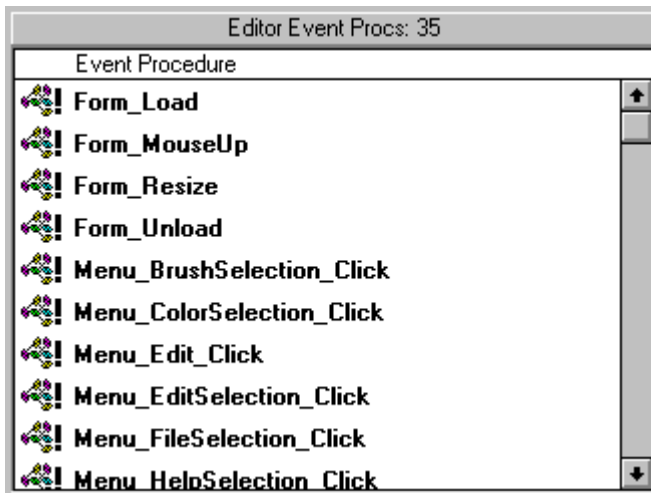
?

See Also:

Exception Database

▶ Form Event Procedure List

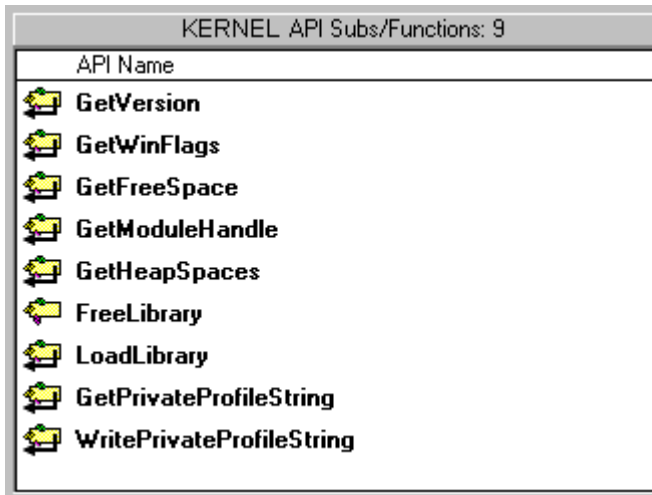
Clicking on the **Event Procs** line for a particular Form in the Main List will display all the Event Procedures in that Form. The total number present is given on the heading bar.



- ▶ The Event Procedures can be visually identified by the image representation given.
- ▶ Double clicking on an Event Procedure line in the above list with the **left mouse button** will cross-reference that Event Procedure and display which routines it is called from.
- ▶ Clicking with the **right mouse button** will display the code for that particular Event Procedure.

▶ API Declarations

Clicking on an External Library file name in the Main List will display those API functions and subroutines declared from that particular library. The total number present is given on the heading bar.



- ▶ The API Declarations can be visually identified by the image representation given.
- ▶ Double clicking on an API Declaration line in the above list with the **left mouse button** will cross-reference that API and display which routines it is called from.

▶ External Library and Custom Control Version Information

Clicking on an External Library or a Custom Control file name or in the Main List will display version information for the file if present and the file length, date and path details.



▶ For external libraries they will be located in the following order: the project directory, the Windows System directory and the DOS Path directories.

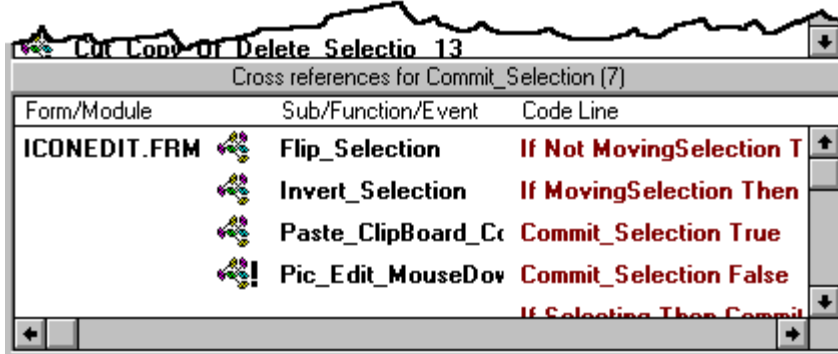
If a VBX file does not exist in the directory noted by the project MAK file then it will be located as above.

▶ Cross-Referencing

Sub Procs, Functions, Event Procs and APIs can all be cross referenced to determine what routines in the project they are called from.

Once the appropriate initial list is displayed from the Main List, double clicking on an item with the **left-mouse button** will invoke the cross-reference search.

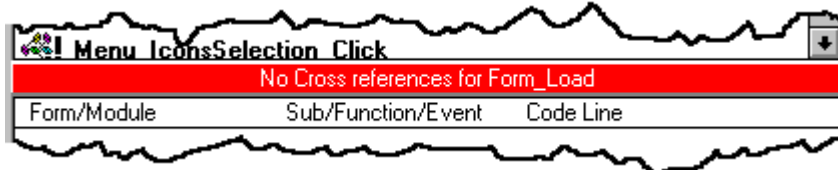
If any cross-references are found they will be displayed below that list.



All the routines that have been found to call the chosen item will be displayed along with the Form or Module that they appear in and the line of code the reference was found in.








- ▶ Double clicking on a routine in the Cross-Reference List will display the code for that particular routine.
- ▶ The procedures can be visually identified by the image representation given.
- ▶ You can resize the columns in the list and the positions will be saved for the next session.

If no cross-references have been found then the heading bar for the Cross Reference-List will note this.



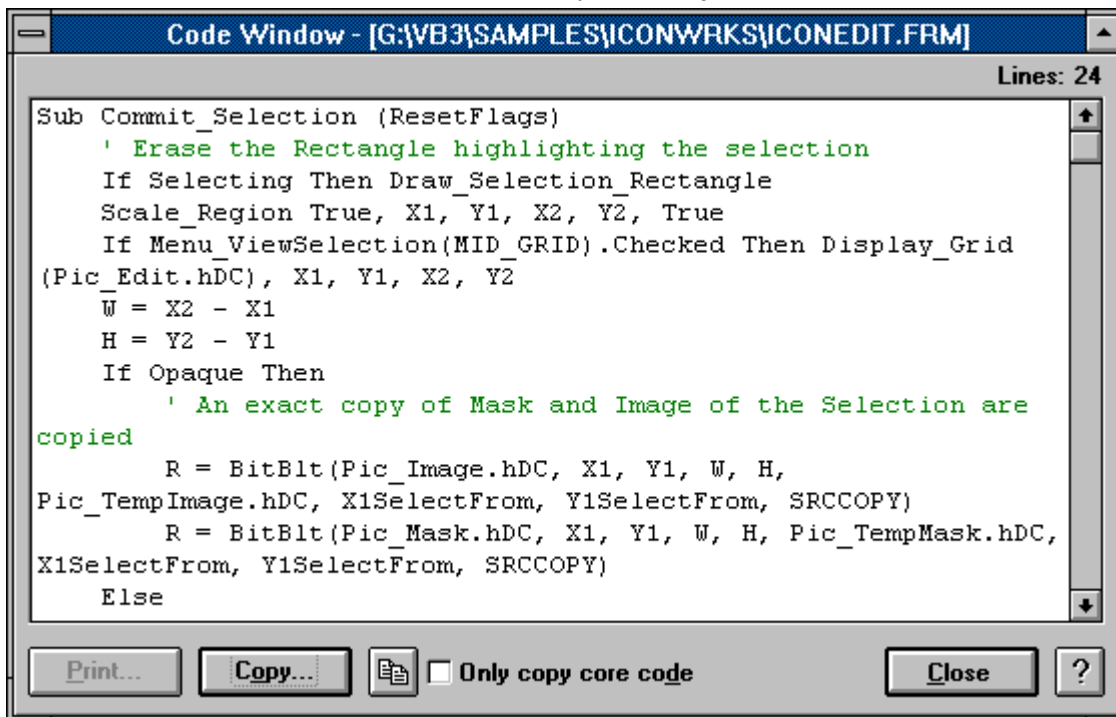
Visual Identification

The following images are used throughout VB Project Analyser to represent the various types of procedures.

-  Sub Procedure
-  Private Sub Procedure
-  Function
-  Private Function
-  Event Procedure
-  Form / API Subroutine Declaration
-  API Function Declaration

▶ Code Window



The Code Window displays the code for a particular Sub Procedure, Function, Event Procedure or Declarations Section when requested by the user.



The screenshot shows a window titled "Code Window - [G:\VB3\SAMPLES\ICONWRKS\ICONEDIT.FRM]" with a status bar indicating "Lines: 24". The code is as follows:

```
Sub Commit_Selection (ResetFlags)
    ' Erase the Rectangle highlighting the selection
    If Selecting Then Draw_Selection_Rectangle
    Scale_Region True, X1, Y1, X2, Y2, True
    If Menu_ViewSelection(MID_GRID).Checked Then Display_Grid
(Pic_Edit.hDC), X1, Y1, X2, Y2
    W = X2 - X1
    H = Y2 - Y1
    If Opaque Then
        ' An exact copy of Mask and Image of the Selection are
copied
        R = BitBlt(Pic_Image.hDC, X1, Y1, W, H,
Pic_TempImage.hDC, X1SelectFrom, Y1SelectFrom, SRCCOPY)
        R = BitBlt(Pic_Mask.hDC, X1, Y1, W, H, Pic_TempMask.hDC,
X1SelectFrom, Y1SelectFrom, SRCCOPY)
    Else
```

At the bottom of the window, there are buttons for "Print...", "Copy...", a document icon, a checkbox labeled "Only copy core code", a "Close" button, and a help button with a question mark.

- ▶ Comment lines are displayed in green.
 - ▶ If the Code Window is displayed as a result of a cross-reference or a find text command then the line that the text or procedure being referenced exists will be coloured red and the number of occurrences in this procedure will be noted.
 - ▶ The **Copy...** button will display the Copy Dialogue which allows you to selectively copy text to the clipboard.
 - ▶ The  button is used to immediately copy the entire text window to the clipboard. If the **Only copy core code** option is checked then only the code between the Sub / Function and End Sub / Function lines will be copied for easier pasting into another VB procedure. This option is not available for a Declarations Section of code.
 - ▶ The Code Window can be resized and repositioned and will be saved between sessions.
 - ▶ The  button displays this help topic.
- It is envisaged that a Code Browser will be added so that you can navigate around the Project from the Code Window.*

▶ **New / Modified for Beta 0.2**

Released 28 November 1993

1. Interactive cross-referencing for Functions and Procs.
2. Code Window to display the project code for particular Functions and Procs.
3. Statistics at Form level as well as Project level.
4. Full version information for VBX files provided.
5. Resizable window which is saved between sessions.
6. Now detects Private Sub Procs and Functions.
7. Bitmap representations in lists for easy visual identification of Functions, Sub Procs, Event Procs and API Declarations.
8. You can now double-click on a Main List Form or Module to expand or collapse it.
9. Buttons provided to expand or collapse the entire Main List.
10. Status bar context-sensitive help.
11. Fixed the problem when a referenced VBX is not in the DOS Path.
12. Cleanly detects when Forms and Modules are not saved as Text.

► **New / Modified for Beta 0.3**

Released 15 February 1994

1. Non-default property information for Controls in separate window with actual colours displayed from the controls list.
2. For image properties in the Property Window you can display the actual image (icon, bitmap, etc.) and copy to clipboard.
3. For properties which store their data in the FRX file you can display it in either ASCII or standard HEX format.
4. Option to search for particular text in a Project and display the Functions and Procs that it occurs in.
5. The Code Window text can be completely customised and the window now displays comments in green and when displayed from an xref the lines of code with references on them appear in red.
6. Report listing on all unreferenced procedures in the project.
7. VB Project Analyser can be launched with a command line for the Project to open.
8. When being notified that a file is not in text format during a load you can now cancel the load or continue.
9. Number of lines of comments now noted.
10. Project-level statistics now right aligned with thousand separators.
11. Drag and drop support from file managers for opening project files.
12. Version information provided for all DLLs declared (external libraries).
13. Private Sub Procs and Functions now visually identifiable.
14. Includes context-sensitive help with outline contents window.
15. Option provided (Fast Load) to not display bitmaps in project-level statistics list and controls list and to also start Analyser faster.
16. Copy button added to code window for copying text whole, only between the Sub and End Sub, or selectively.
17. For xref lookup and search now shows portion of line in resulting list.
18. Displays the last four projects analysed in the File menu for quicker re-analysis.
19. Separate section beneath form or module for the Declarations.
20. In the file information display for forms and modules whether they are Read Only is noted (when using version control / source management programs).
21. Report listing of all procedures in project.
22. API functions and subroutines now visually identifiable.
23. Search listbox columns can be resized and will be saved for future sessions.
24. Lists apart from main list require a double click action instead of just a single click to allow resizing of listbox columns to be made easier.
25. If a DLL or VBX file is not in the location specified by the project file it will be searched for in the project directory, the Windows System directory and all those directories in

the DOS path.

Fixes:

- a). Can now cope with huge VB projects (e.g. 50+ forms).
- b). Main window generally appears and operates cleaner.
- c). Fixed problem when cross-referencing certain projects.
- d). No longer leaves SDATA.TMP files hanging around.
- e). Screen resolutions above 800x600 now size correctly.
- f). Will not allow non-MAK files to be opened.

Main List

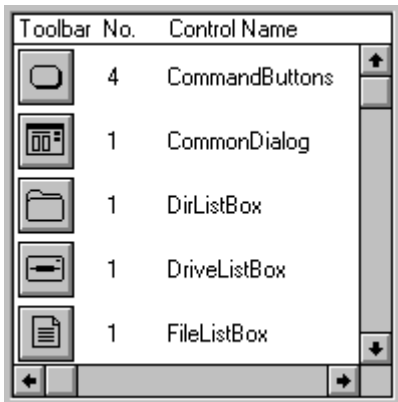
-----Forms-----
+ AboutBox
+ ColorPalette
= Editor
Controls
Declarations
Event Procs
Sub Procs
Functions
+ Viewer
-----Modules-----
+ ICONWRKS.GBL
+ ICONWRKS.BAS
--External Libraries--
+ GDI
- KERNEL
Declarations
+ USER
---Custom Controls---
CMDIALOG.VBX

Project Statistics

Statistics	
Source Code Lines:	1,965
Comment Lines:	1,661
Controls:	169
Event Procedures:	70
Sub Procedures:	64
Functions:	5
API Subs/Functions:	26

The Source Code Lines total is the actual coding lines in the project - it does not include comments, blank lines or any of the control definitions.

Project Controls Listing



Toolbar No.	Control Name
4	CommandButtons
1	CommonDialog
1	DirListBox
1	DriveListBox
1	FileListBox

All the controls used in the project are displayed along with their toolbar image and the number of occurrences.

▶ **Note:** Not all third party custom control toolbar images can be determined automatically. This is because the function used to reach the image identifier in the VBX is not required to be implemented by the third party supplier at the moment so not all do. If the toolbar image cannot be determined then the following image is displayed instead:

▶

See Also:

[Exception Database](#)

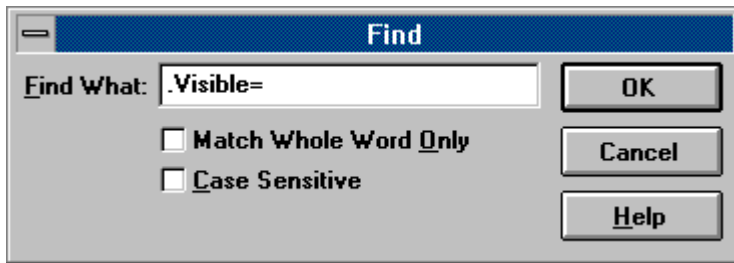
Exception Database

For some of the VBX files that have not implemented the function required to extract the toolbar image the appropriate bitmap has been manually extracted and stored in the ANALYSER.MDB file shipped with VB Project Analyser.

If you come across controls where their image cannot be determined then if you [Email me](#) a bitmap of the toolbar along with the ClassName of that control I will add it to this exception database.

► Find Dialogue

Use the Find Dialogue to search for references to specific text anywhere in the project.



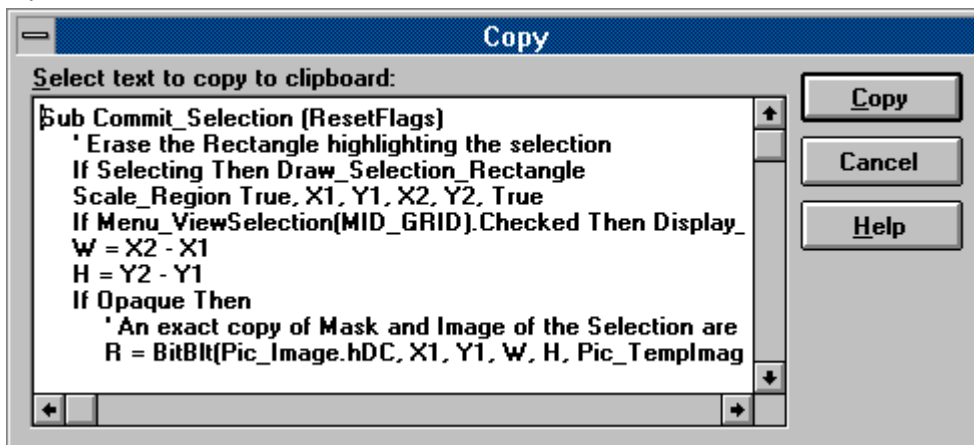
- Check the **Match Whole Word Only** option to only find references where the text entered matches completely and does not appear within another word.
- Check the **Case Sensitive** option for a case sensitive search.

See Also:

[Find Text](#)

▶ Copy Dialogue

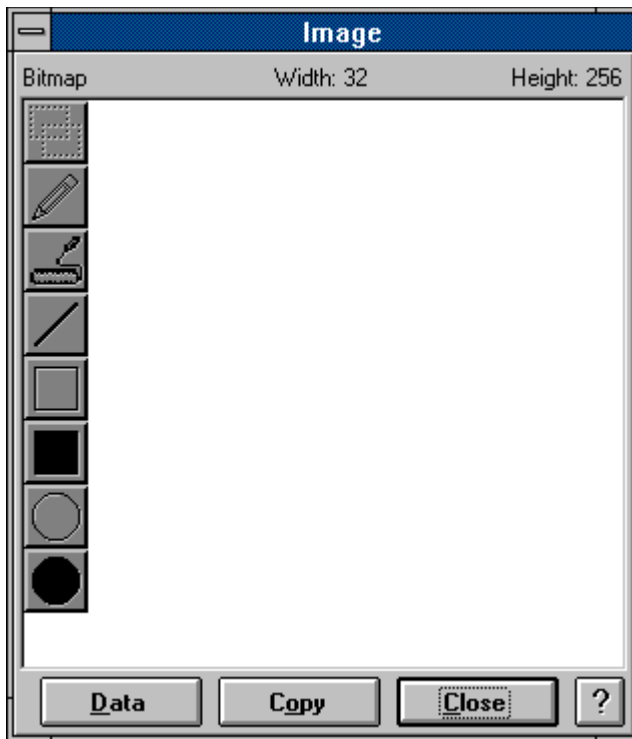
The Copy Dialogue allows you to selectively copy text from the Code Window to the clipboard.



- ▶ Highlight the text you wish to copy.
- ▶ The **Copy** button copies the highlighted text to the clipboard and closes the dialogue.

▶ Image Dialogue

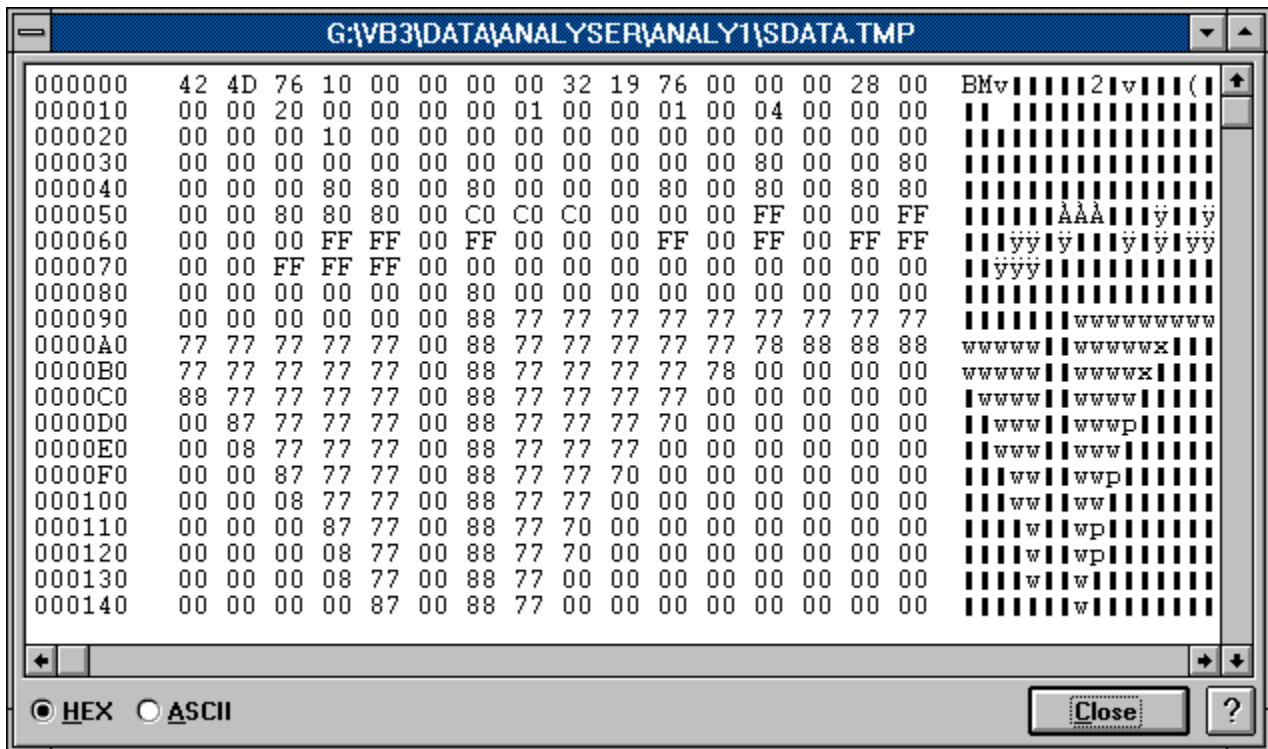
The Image Dialogue is used to display any Icons, Bitmaps or Metafiles from the [Properties Dialogue](#).



- ▶ For Bitmaps and Icons the width and height in pixels are also given.
 - ▶ The **Copy** button will copy the image to the clipboard.
 - ▶ The **Data** button will display the image in the [Data Dialogue](#) in either ASCII or HEX format.
 - ▶ The Image Dialogue can be resized and repositioned and will be saved between sessions.
- The ▶ button displays this help topic.

▶ Data Dialogue

The Data Dialogue will display information from the Properties Dialogue for an image or some raw data.



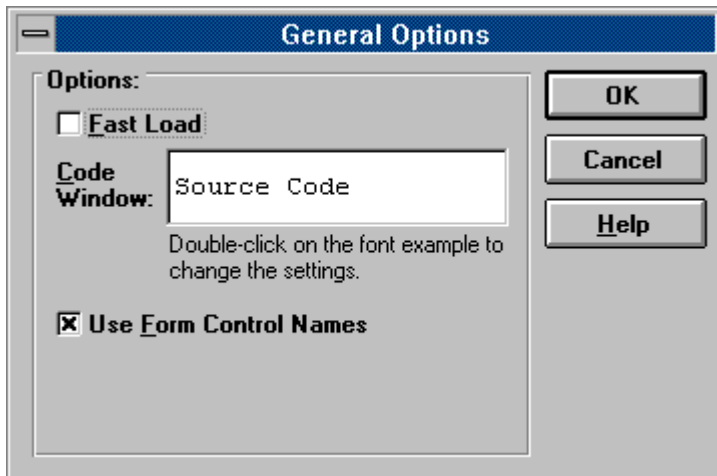
▶ Use the **HEX** and **ASCII** option buttons to specify how the data should be formatted in the window.

▶ The Data Dialogue can be resized and repositioned and will be saved between sessions.

The ▶ button displays this help topic.

▶ General Options Dialogue

Use the General Options Dialogue to specify general configuration options.



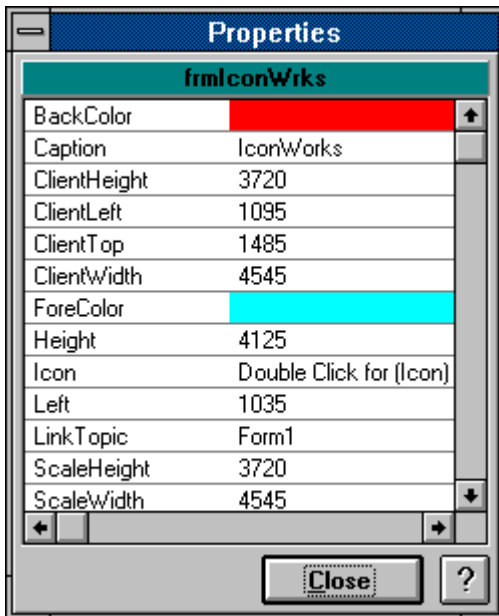
- ▶ Check the **Fast Load** option to skip display of the toolbar control images in the Project Controls Listing and the Form Controls List. This will also result in a speed increase when loading VB Project Analyser.
- ▶ Double click on the Code Window font example to set the font name and styles which will be used for the text in the Code Window.
- ▶ Check the **Use Form Control Names** option to display the Form control names in the Main List instead of the actual DOS Form file names.

Fonts

Name from available windows fonts.
Style dependent on font name (e.g. bold, italic).
Size
Strikeout
Underline
Colour

▶ Properties Dialogue

The properties dialogue is displayed when you double click on a control in the Form Controls List. It lists all the non-default properties for that control.



- ▶ Any colour properties are displayed with a block of that colour.
- ▶ The Properties Dialogue can be resized and repositioned and will be saved between sessions.
- ▶ Properties containing Icons, Bitmaps or Metafiles can be double clicked on to display the actual images.
- ▶ Some custom controls store specific property customisation information in the FRX file. The property line in the Properties Dialogue will display **Double Click for Data**. Double clicking on this property will display the actual data in either ASCII or HEX format.

How to use the Table of Contents



The Table of Contents provides a graphical view of the pages which make up this reference.



You can activate the Table of Contents at any time by pressing the *Contents* button in the toolbar of the main help window.



The *Browse* buttons will move forward and back, page by page, through this reference.



Double clicking a closed book will open it to reveal its contents.



Double clicking an open book will close it.



Double clicking on a 'page icon' will display that page.

▶ **Dialogue Boxes**

The following dialogue boxes are used in VB Project Analyser:

- ▶ Find Dialogue
- ▶ General Options Dialogue
- ▶ Code Window
- ▶ Copy Dialogue
- ▶ Properties Dialogue
- ▶ Image Dialogue
- ▶ Data Dialogue

▶ Options

Use the Options|General menu command to specify general configurations options:

- ▶ Specify not to display the toolbar control images in the Project Controls Listing and the Form Controls List.
- ▶ Set the font for the Code Window.
- ▶ Specify whether to display the Form file names or the Form control names in the Main List.




▶ Reports

The following report lists can be produced by VB Project Analyser:

- ▶ Unreferenced Procedures Report
- ▶ All Procedures Report

▶ Unreferenced Procedures Report

This report generates a list of all the unreferenced procedures or functions in the entire project and the Forms or Modules in which they appear. The total number of unreferenced procedures is given in the heading bar.

3 Unreferenced Procedures		
Sub/Function/Event	Form/Module	Lines
 TestProcUnref	VBPRJBOT.FRM	5
 cmdAllMinus_Click	VBPRJ.FRM	18
 cmdTest_Click	VBPRJ.FRM	34









The Lines column notes how many coding lines appear in that procedure. The percent bar will be displayed whilst the report is created.

- ▶ Double clicking on a routine in the list will display the code for that particular routine.
- ▶ The procedures can be visually identified by the image representation given.
- ▶ You can resize the columns in the list.

It is envisaged that you will be able to sort the list based on the number of lines in each procedure.

▶ All Procedures Report

This report lists all the Sub Procs, Functions and Event Procedures in the entire project and the Forms or Modules in which they appear. They are listed alphabetically and the total number of procedures is given in the heading bar.

139 Procedures		
Sub/Function/Event	Form/Module	Lines
 Activate_Grid	ICONEDIT.FRM	7
 Adjust_All_Controls	VIEWICON.FRM	58
 Adjust_Color_Palette	ICONEDIT.FRM	23
 Adjust_ScrollBar	ICONEDIT.FRM	20
 Adjust_Status_Bar	ICONEDIT.FRM	29
 Adjust_Tool_Palette	ICONEDIT.FRM	11
 Adjust_X_and_Y	ICONEDIT.FRM	15
 Cmd_OK_Click	ABOUTBOX.FRM	3

The Lines column notes how many coding lines appear in that procedure.

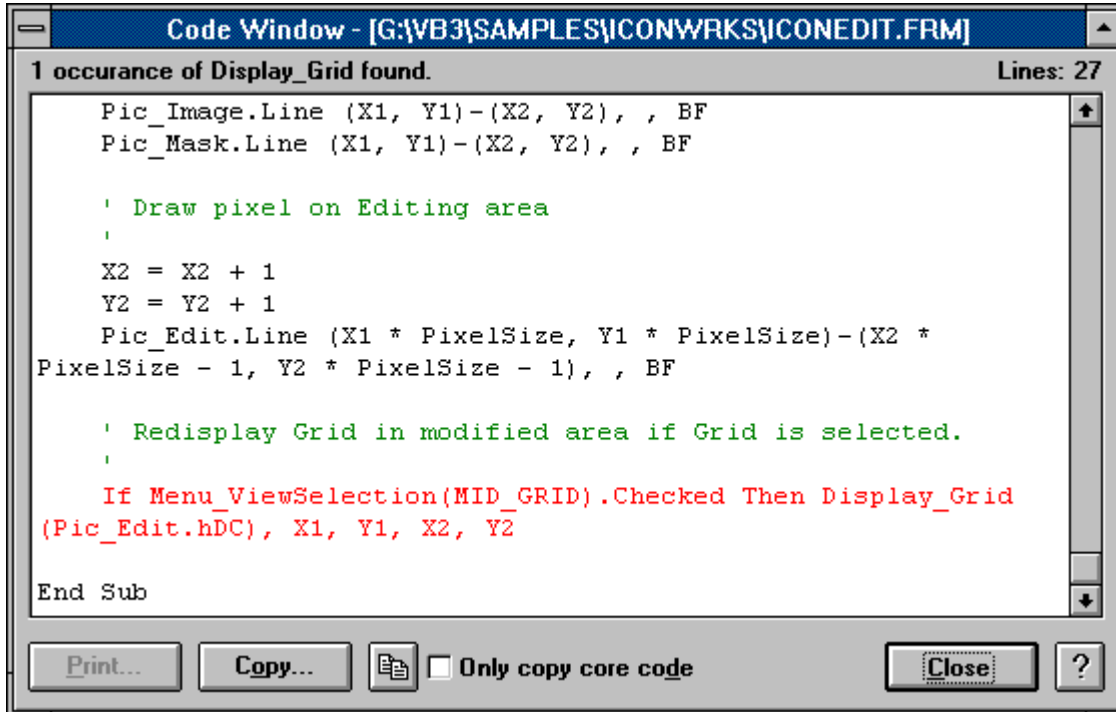
- ▶ Double clicking on a routine in the list will display the code for that particular routine.
- ▶ The procedures can be visually identified by the image representation given.
- ▶ You can resize the columns in the list.

► Find Text

Use the Edit|Find menu command to search for references to specific text anywhere in the project. You can specify the case sensitivity and whether to search for a whole word or not.

All the routines that have been found to reference the search text will be displayed along with the Form or Module that they appear in and the line of code the reference was found in.

-
- Double clicking on a routine in the list will display the code for that particular routine. The number of occurrences of the search text is noted and any lines where that text appears will be coloured red.
- The procedures can be visually identified by the image representation given.



The screenshot shows a 'Code Window' titled 'Code Window - [G:\VB3\SAMPLES\ICONWRKS\ICONEDIT.FRM]'. The window displays the results of a search for 'Display_Grid', indicating '1 occurrence of Display_Grid found.' and 'Lines: 27'. The code is as follows:

```
Pic_Image.Line (X1, Y1)-(X2, Y2), , BF
Pic_Mask.Line (X1, Y1)-(X2, Y2), , BF

' Draw pixel on Editing area
'
X2 = X2 + 1
Y2 = Y2 + 1
Pic_Edit.Line (X1 * PixelSize, Y1 * PixelSize)-(X2 *
PixelSize - 1, Y2 * PixelSize - 1), , BF

' Redisplay Grid in modified area if Grid is selected.
'
If Menu_ViewSelection(MID_GRID).Checked Then Display_Grid
(Pic_Edit.hDC), X1, Y1, X2, Y2

End Sub
```

At the bottom of the window, there are buttons for 'Print...', 'Copy...', 'Only copy core code' (with an unchecked checkbox), 'Close', and a help icon (?).

