

Installation

[Operation](#)

[Utilities](#)

Files

After unzipping Projmgr.zip, place the files as follows:

Windows\System

vbprojs.mdb	- Database file
mbar.vbx	- MeterBar VBX (Thank you, Scott Stubbs)
projmgr.hlp	- help file

Projmgrs.exe can be run from whatever directory you wish. If runtime errors are generated, it will write a log called "error.txt" in that directory.

This program was written with Visual Basic 3.0 for Windows (Standard Edition). The only other VBX referenced in it is CMDialog.VBX, which should be present on any system with VB 3.0 installed. The code in this program references msafinx.dll, which should also be present in the Windows\System directory on any system which has Access or Visual Basic on it.

This program will only work on VB projects which are saved as Text Files. I don't know how it will work, if at all with VB 2.0 or earlier as I have never used or seen that program. VB by default saves as binary, so ensure that you have Save As Text files as the default under the Options, Environment menu selection.

Purpose

Project Manager functions as a shell which loads and unloads Visual Basic with an appropriate project loaded. It tracks the files used in each project, including VBX's, external modules and DLL's called. The program will track and total the time spent on each project. The information on the projects is stored in an Access 1.1 database vbprojs.mdb. Times are written to this database as well. In addition, two utilities are included, one to compact the database, and one to write error handling and logging code to your project. A backup path can be entered and all project files are backed up to this path in one pass.

Registration and Support

Register by sending \$15.00 Canadian (\$10.00 U.S.) to:

Bud Street,
A.F. Street Consulting,
R.R.#1,
Peterborough, ON, Canada
K9J 6X2

Please make cheques, money orders payable to A.F. Street.

For support, you can contact me at 705-292-6621 or via CompuServe E-mail at 72152,333. If you have problems with the program, please have the "Error.txt" file available for me.

Any constructive comments you may have on the program are appreciated. Just go easy on me though, I first used a computer in March of 1992, and wrote my first "Hello, World" program in QBasic in September of 1993.

Disclaimer

While this program was tested on different systems, no guarantees are made as to its operation. As usual with this stuff, I assume no responsibility for anything it might do to your system or projects. Note in particular the warning before using the Error Handler utility. Try this on dummy code to see if you like it before using it on a project.

Shareware

This program remains the property of the owner and is copyrighted. You are granted a licence to use it for 30 days. If you find it useful, you are required to pay the registration fee or stop using the program. Paying the registration fee will help ensure the future availability of useful software from sources other than the big guys, who are becoming fewer by the day. It will also enable me to spend more time working on future upgrades to the program.

Operation

[Utilities](#)

[Installation](#)

Startup

With the non registered version, you will initially see a nag screen which is not present on the registered version. Clear this form by pressing enter or clicking the OK button. This is the only crippling on this version of the program.

After the nag screen is gone, you will see a LogOn Form. This is included for those developers using a secured Access database on their system. If you have a secured database on your system, you can enter your User name and Password here. Occasionally, you may have to enter the path to the System.MDA file as well, which can be done here. If you are not using a secured database, enter nothing, and press Enter or Click OK to carry on. If the program can't find your System.MDA file it will tell you and you can enter the path.

The main form now loads. From the File menu, select Add Project to add a new project. From the CmDialog box, select a .MAK file to add. The program will find the files, VBX's, and modules referenced, add them to the database and display the results in the text boxes on screen. You navigate between projects on the main form by using the Data Control as the bottom of the form.

To work on your project, click the Open Project button or select Open Project(Record Start Time) from the Times menu. Visual Basic will start with your project loaded, and record the start time for you. When done working on this project, Close VB by using the Close Project or selecting Close Project(Record Stop Time) from the Times menu. You can view the time spent on each project by selecting View Times from the Times menu. This will pop up a subform with Close and Refresh buttons on it. This form displays the times and total for each project.

To backup your project, either type a path into the Backup Path textbox or select one from the CMDialog which will pop up if you select BackupProject from the File menu and the Backup Path textbox is blank. Note the Backup Path can only contain a path, it does not need a filename.

If your project changes, select it in the database and click Update Project from the File menu. The program will add or delete files, etc., as necessary.

To delete a project, click Delete Project from the File menu. Note that there is no recovery from a delete, all files in the database, including times, are deleted. You will be asked to confirm your delete selection before the delete takes place.

Hot Keys

Quick access keys are highlighted on buttons and menu choices in the standard windows fashion. In addition, the following hotkeys will bypass the menu choices:

Add Project	-	Ctrl + A
Update Project	-	Ctrl + U
Backup Project	-	Ctrl + B
Delete Project	-	Ctrl + D

Open Project(Record Start Time)	-	Ctrl + O
Close Project(Record Stop Time)	-	Ctrl + L
View Times	-	Ctrl + T

Compact Database - Ctrl + P
Write Error Handler - Ctrl + W

Utilities

[Installation](#)

[Operation](#)

Compact Database

The database vbprojs.mdb needs to be compacted occasionally. You can do this by selecting Compact Database from the Utilities menu. You should backup the database file before compacting it, and a dialog box will remind you of this and give you the chance to do so before compacting is done.

To edit or do maintenance on the database, use the Data Manager program which comes with VB. It is not a good idea to change the structure of the tables as this will quite likely cause the program to crash.

The database was made using an unsecured version of Access, and full permissions have been enabled for all levels of users and groups. There should be no difficulty loading with either secured or unsecured systems.

For secured systems, you should be able to log on as Guest with no password in addition to using your normal User name and Password. If you wish to secure the database, import the tables into a new database which you open in Access under your User name. Save the new database and rename it vbprojs.mdb. Then copy it to your Windows\System directory. I would suggest making a backup of the original before you do this in case anything goes wrong.

Error Handling Code

To add error handling code to your project, select Write Error Handler from the Utilities menu. This selection will check the currently selected project in the database for error handling statements. If it finds none, it will go through your project and add On Error GoTo statements, with appropriate labels, to each function and sub. It will write a Constant called "Procname" to each one as well.

It will also write a function called LogError to the first module it finds in your project. This function is called by the error handler in each sub or function. When called by an error, the function is passed the error number and the Procname of the function. It then displays a message box for the user advising of the error. It writes a file called "error.txt" which contains the time, error number, error description, and the Procname of the function which caused the error. It then exits the offending procedure. It does not include Resume statements, so if you wish custom error handling in various procedures you will of course have to add these manually.

The "error.txt" text file can be very helpful in finding the cause of errant program behavior. Send it to me if you have trouble with the program. This is not an attempt at having the user debug your program. There are too many systems and setups out there to predict ahead of time what can happen to your program in someone else's hands. This just tries to find out what is going on under their usage and make it right for them.

Note that if there are any On Error statements present in your code, the function will not run as it may trash whatever you have already done. I generally write the error handlers last as a debugging aid when developing an application, so this works well for me. If there are no modules in your project you will have to add the LogError function manually. I suggest you run this function on a dummy project with a form and a module with a couple of functions first to see what it does and give you a copy of LogError() to use.

Future Plans

Future plans for the program include more choice in the error handling code generated. I intend to

allow the user to configure exactly how the error handler should write the code.

Plans also include building in the ability to check for declared variables and then search for the usage of them in procedures to find unused declared variables. Also all calls to external functions will be tracked.

Also, the ability to allow the user to store all project files in a directory (including the .exe, help files, etc) and have the program back up everything in the storage directory. While I work this way, many people likely don't, so I will try and build in an option to do this in future releases.

Provision to print out the contents of modules and forms, as well as the structure of the project is also a possibility for the future.

If you have any suggestions for additions you would like to see in the program, please advise.

Of course, your support by registering will help me devote the time necessary to upgrade and enhance the program. Registered users will get the upgrades for the cost of shipping for 6 months after registering. Upgrades after that will be available to registered users at nominal rates, while the program cost will no doubt go up as it gets bigger and better. (Isn't that how it is supposed to work?)

