

VB Backup for Visual Basic 2.0 (and later)

[What is VB Backup?](#)

[Features](#)

[Getting Started](#)

[Using VB Backup](#)

[Command-Line Arguments](#)

[Multiple Backup Generations](#)

[vbbackup.ini Explained](#)

[Restoring Projects](#)

[Comments](#)

What is VB Backup?

VB Backup is a utility for conveniently backing up the files belonging to your Visual Basic projects.

Backing up a project with **VB Backup** is as easy as double-clicking or dragging a file.

Features

VB Backup will remember a unique backup path for each project.

VB Backup can make a backup directory if it doesn't already exist.

Multiple Backup Generations of backups can be maintained - automatically.

VB Backup works with VB's .mak files. However, VB 1.0 .mak files are in a binary format and cannot be read by VB Backup. If you try to backup a VB 1.0 file, VB Backup will display an error message and bailout.

You can specify which files VB Backup will skip when backing up projects. For example *.vbx would specify any custom controls such as grid.vbx. See vbackup.ini Explained.

VB Backup is for VB 2.0 and later (I hope). It requires vbrun200.dll.

VB Backup is freeware.

vbbackup.ini Explained

```
[Defaults]
MainProject=c:\windows\vb\kiddraw.mak
DefaultDestination=b:\vb_proj
ExcludeFromBackup=*.vbx constant.txt c:\windows\vb\reusable\*.*

[c:\windows\vb\kiddraw.mak]
CopyTo=b:\vb_proj
Generations=3
LastWritten=2
```

`MainProject` is the project to backup when no command-line arguments are used.

`DefaultDestination` is for your convenience when backing up a new project. **VB Backup** uses it to fill in the new project dialog so you don't have to type the destination path.

`ExcludeFromBackup` lists the files to skip when backing up the project. VB's "Like" operator is used to compare the file pattern with each line in your project's .mak file. 'Regular expression' matching is used rather than DOS file matching. See your VB manual under "Like Operator" in the "Operators" appendix for more details.

Each file pattern should be separated by a space. Wildcards '*' and '?' are accepted. Examples of possible file patterns are:

<code>*.vbx</code>	'excludes all files with .vbx extension
<code>constant.txt</code>	'excludes files called constant.txt
<code>e:*.*</code>	'excludes all files on drive e:
<code>c:\windows\vb\reusable*.*</code>	'excludes all files in dir c:\windows\vb\reusable ' and its subdirs

This entry example is for the project `c:\windows\vb\kiddraw.mak`, which is backed up to `b:\vb_proj`. Three generations of backups are maintained in addition to the main backup:

```
b:\vb_proj
├── 1
├── 2
└── 3
```

The last directory used was `b:\vb_proj\2`.

Getting Started

1. Open [vbbackup.ini](#) for editing. Put your main VB project's name after `MainProject` under `[Defaults]`.

For example: `MainProject=c:\windows\vb\kiddraw.mak`

Also put the path of the default backup directory path after `DefaultDestination`.

For example: `DefaultDestination=b:\vb_proj`

VB Backup will always confirm the destination for a project it hasn't handled, but the `DefaultDestination` setting will give it something to put on the dialog box so you don't have to type a path for each new project.

2. Then run **VB Backup**. A dialog box will appear to confirm the entry to be written to `vbbackup.ini`. See [Using VB Backup](#).

Using VB Backup

Double click on **VB Backup** to backup the project listed after `MainProject` in [vbbackup.ini](#).

If you use Norton Desktop, place **VB Backup** on the desktop or in a group window. Dragging .mak files on top of **VB Backup** will backup the project.

Or run VB Backup from the [command-line](#).

If **VB Backup** hasn't handled the project before, a dialog box will appear to collect information for the `vbbackup.ini` entry.

VB Backup can currently backup one project at a time.

The backup can be interrupted while in-progress by hitting the escape key. However, **VB Backup** will wait until in-between files to pause the backup and verify the abort.

Command-Line Arguments

Specify a project to backup as a command-line argument:

`vbackup source`

For example: "vbackup c:\windows\vb\kiddraw.mak"

The destination will be read from the [c:\windows\vb\kiddraw.mak] entry in vbackup.ini. If no entry exists then a dialog will appear to collect information. Just a .mak file name without a path will cause VB Backup to concatenate the name of the directory it is being run from with the name of the project. This is convenient if you keep VB Backup in the same directory as your projects.

Or use both a project and a destination path as command-line arguments:

`vbackup source destination`

For example: "vbackup kiddraw.mak a:\".

The project will be backed up to the destination path on the command-line, regardless of the destination path in the vbackup.ini entry. In addition, the multiple backup generation option will be ignored if a destination is specified on the command-line.

In all cases, the name of the project to backup can be with or without the .mak extension, and the path of the destination can be with or without an ending '\.

Multiple Backup Generations

VB Backup can maintain several generations of backups so you can go back to previous versions of your project if necessary.

Multiple generations of backups can be made by setting `Generations` in `vbbbackup.ini` to something larger than zero (0).

Using `Generations=0` will cause **VB Backup** to use only the destination you specify, overwriting the last backup each time. Setting `Generations` to greater than zero will cause **VB Backup** to copy files to the directory you specify, then during the next backup to a subdirectory `\1`, and so on until it uses the last directory as specified by `Generations`. Then it will begin again with the main backup directory. `LastWritten` is used by **VB Backup** to record the last directory it used.

You can set `Generations` on the dialog that **VB Backup** brings up for new projects, or by directly editing a project's `vbbbackup.ini` entry.

Comments

[VB Backup](#) is copyrighted 1993 by Stephen Kearney.

If you use [VB Backup](#), please drop me a note. I'd love some feedback!

Please send comments, questions, money, etc. to:

Stephen Kearney

Internet email: neff123@garnet.berkeley.edu

4215 Terrace Street, #2
Oakland, CA 94611
USA

Phone: (510) 547-3189

.mak files

Visual Basic uses .mak files to store information about your project such as the files it uses, the name of the .exe, etc.

Restoring Projects

To restore the projects you have backed up with [VB Backup](#), use [VB Restore](#) by Ismo Kärkkäinen. It is distributed in the same zipfile as [VB Backup](#).

