

PowerBar/Write

The Windows Write ButtonBar

Program and documentation
©1992 David Stewart, d.b.a. Argyle Softstuff

PowerBar/Write -- Making Write More Powerful

A lot of people use Write, the bundled Windows word processor, for their simpler word processing; many also use it for *all* of their word processing. While a lot of Windows users think of Write as an underpowered word processor, we who like it find Write's very lack of "power user" features a virtue--the virtue of simplicity. Write makes up in ease of use and convenience (not to mention thrift) what it lacks in power and features.

But you've probably wished that it were somewhat easier to operate--say, that it had just a bit of that toolbar technology. Toolbars, iconbars, buttonbars are among the most attractive features of Windows applications. PowerBar/Write gives you toolbar technology for Windows Write, incorporating the most useful and commonly used menu features of Write in a buttonbar that's always readily accessible. PowerBar/Write can speed your work in Write, making it more the word processor you want it to be.

In addition, PowerBar/Write includes a utility menu that includes a file finder and loader, application launching, sizing options for Write, and setup options for PowerBar/Write.

Thanks for taking a look at PowerBar/Write. I'm sure you'll find it helpful and even interesting to use.

PowerBar/Write--What It Takes

PowerBar/Write is written in Visual Basic for Windows 3.1. It requires:

- A 286 or higher CPU running in Standard or Enhanced mode.

- A mouse, because almost everything in PowerBar/Write is mouse-activated.

The Visual Basic runtime library file, VBRUN100.DLL, in a directory that is on your

Powerbar/Write

2

path. (The Windows directory is a good place for it.) If you don't have this file, you can usually download it from the same on-line service from which you got this file (assuming that was your source) or get it from a friend who uses Windows; on-line, look for VBRUN100.EXE or VBRUN100.ZIP.

Windows 3.1 and Write itself. Some of Write's menu items changed their underlined hotkeys in Windows 3.1, so while some buttons would work in Windows 3.0, more wouldn't. You might get some use out of it, but hey, if you use Windows, you should, if you can at all afford it, use Windows 3.1.

PowerBar/Write takes up some real estate on the screen--a vertical strip on the right side of your screen, just leftward enough to leave Write's scrollbar uncovered. For many margin settings in Write on standard 640x480 VGA (and on laptop computers), this may mean that you will be cutting off some of your view of the right margin area of Write. I've made PowerBar/Write as narrow as possible to minimize this problem, but it's not perfect. If you use a higher resolution--800x600 or 1024x768, e.g.--you should not have any overlap. (Future versions will, I hope, be narrower. If you like Write, but don't like the margin problem, there is a workaround: For the time you spend writing, set your margins so that they total about 3"--for example, 1.5" left, 1.5" right. This will put the right margin just to the left of PowerBar/Write, so you'll be able to work and see all your text. Then, when your editing session is done or you want to print your document the way you want it to look when you're finished, change the margins to the desired measures.)

Installation

Well, this couldn't be much easier: PowerBar/Write consists of one program file, PBWRITE.EXE, and this documentation file. You need not have the executable file in any particular directory unless you want to run it without entering a path, in which case you'd need to put it in a directory on your path. However, it would be good to have it in your Windows directory, if you don't mind having stuff that isn't "original equipment" in there; or alternatively, if you start Write up in a separate data file directory, as 3.1's Program Manager lets you, you might want to put PowerBar/Write in that directory.

Unzip the two files (unless you're sneaking a peak at this with Norton Desktop for Windows, Magellan or some other snazzy file viewing utility, you've probably done it by now) and place them in the directory in which you want them to reside. Then, if you use Program Manager as your shell, use the standard procedure for installing a new launch icon. See the *Windows User's Guide*, pp. 78-83. If you use some other shell or nonshell program manager (I highly recommend both Squeegee, by Icom Simulations, and WinEZ, a shareware program by New Generation Software), follow its app installation procedures.

Associating Your Write Files With PowerBar/Write

When you installed Windows, Setup associated your Write files with the WRITE.EXE program file, so that double-clicking on a *.wri file in File Manager or on a file icon in Program Manager starts Write with the file loaded. If you find that you use PowerBar/Write whenever you work with Write files, you can reassociate your Write files, instead, with PowerBar/Write. To do this, use File Manager's Associate... command, from the File menu. See the *Windows User's Guide*, pp. 131-132; but please note that you should Browse to find and enter the PowerBar/Write program name for association. If you set up associations with Notepad by directly editing WIN.INI rather than using File Manager, enter PowerBar/Write's entire path in

Powerbar/Write

3

all capital letters, and be sure to delete the line associating *.wri files with WRITE.EXE. You will need to restart Windows for the new association to take effect.

If you regularly use Write, you may find that you can gain from putting PowerBar/Write on the Run= line of WIN.INI or in your startup group. Since PowerBar/Write can exit Windows as well as launch any application or preassociated file on your system, it can act as a session "shell" (though, like every other Visual Basic app, it cannot be used as your Windows shell!).

How PowerBar/Write Works for You

When you use PowerBar/Write, you do not have to launch Write itself! PowerBar/Write will launch Write for you. When you double-click on the PowerBar/Write icon in Program Manager, PowerBar/Write will open up Write, maximized, and itself. (How's that for handy?) So you can get rid of that Write icon in Program Manager if you find that you like working with PowerBar/Write. If you have associated your *.wri files with PowerBar/Write, you can double click on a Write filename in File Manager (or any replacement file manager you may use), or a Write file icon in Program Manager, or enter a Write filename on a Run... command line, and PowerBar/Write will start up Write and automatically load the file. Note that you can open multiple instances of PowerBar/Write this way, but there's not much reason to; simply open your Write files through PowerBar/Write's Open button.

PowerBar/Write is almost entirely mouse-activated. If you click and drag Write's menus with your mouse, you'll find that PowerBar/Write can really speed up your work. If you use the keyboard on the menus, you may find that switching to the mouse and PowerBar/Write will speed you up, especially with editing functions where you're selecting text with the mouse.

PowerBar/Write automatically places itself on the right-hand edge of your screen, leaving just enough room for Write's vertical scrollbar, since the left visual margin is set by Write. You can, if you wish, move the bar like any other Windows window by dragging its title bar. Note, however, that you can only move and minimize it; you cannot resize the bar.

By default, PowerBar/Write also floats on top of all other application windows, so that it is always at the ready when you're working in Write. (It might sink a level if you have, in PowerBar/Write's corner of the screen, other active apps that float.) You can choose, from the menu, to sink it or float it as you prefer.

You can close PowerBar/Write by clicking the bottom button on the bar.

The Buttons on the Bar

The PowerBar/Write buttonbar is, I dare say, self-explanatory. Each button (except two--Pg Break and Exit PB/W) in version 1.0 corresponds to a menu choice in Write. The buttons are sorted according to the type of function. They could be categorized thus: File functions; Editing functions, Search functions, Type functions, Paragraph spacing functions and a Ruler toggle.

Where Write's Document menu uses the phrase "Page layout," I have used "Margins." Where Write's Character menu uses words for Bold, Italic, etc., I have used symbols. "+2" and "-2" are the Character menu's Enlarge font and Reduce font functions, each having a 2-point effect on the type selected or following. "N" removes all special type effects except size, corresponding to the Character menu's Regular (most word processors, including Microsoft's, call this Plain or Normal).

L, C, R and J are left-justify, center, right-justify and left and right-justify paragraph functions. Norm is a "Normal" paragraph as Write defines that. "Ind" is indents. "1," "1.5" and "2" are single, 1.5 and double line spacing.

The buttons will execute their appointed functions just as quickly as menu choices. But since you will be pushing a button, rather than dragging down a menu, you should, in fact, be able

Powerbar/Write

5

to work faster with the buttonbar than with Write's menus. If you are a frequent Write user, you'll have developed the menu habit, but it will pay to develop the PowerBar/Write habit. Though naturally it will be unfamiliar at first, you'll pick it up quickly.

The Options Menu

File Search/Load A file search and load utility. The search box will pop up on your screen ready to use. It consists of:

A mask setting. The default setting is *.wri, to search for Write files. But you can change it to search for any extension you wish. (Enter filespecs only here--filename and/or extensions, including wildcards; do *not* enter directories or drives.)

A drive drop-down list. You can change the drive you wish to search here.

A directory list. Here you can travel through the directory structure of the disk and drive you want to search; double-click on the target or initial subdirectory.

A matching list box. This will list all the files that match your mask.

Command buttons. These start and halt searches, load a selected file and exit the search box.

To search for a file, set up the mask, the drive and initial directory, then click the Start Search button or press AltS. The searcher will move through the initial directory and any of its subdirectories to find all the files that match the mask you set up. (To search an entire disk, specify its root directory.) Matching files will be listed in the right box. You can stop a search in progress if, say, you already see what you need; just click the Quit Search button or press AltQ.

To load a file, note the condition of the file you're working on currently, if there is a file open: Has it been saved--ever, or lately, or does it have changes you'd just as soon dispose of? PowerBar/Write can either save your most recent, unsaved changes and then load the selected file, or discard the most recent changes before it loads the selected file. If you are loading in place of an unused file, or in place of a file which has not changed since it was last saved, *either button will load the selected file without error*. Select the filename you want to load, and then click the appropriate button.

PLEASE NOTE: If the file you are using has not been saved even once, do not use the button option to save the file; PowerBar Write cannot finish its load operation by itself if you try to use it on a never-saved file. So first save the file through the buttonbar. Alternatively, you can go ahead and use the save button, but you will be prompted as to whether or not you want to overwrite the file you're trying to open; *answer no*. You'll note that that file name now appears in the Save As... dialog's filename space; simply overtype the new file's new name, and press enter; then, when the file open dialog pops up, enter the keys ShiftInsert and Enter to load the file you wanted loaded.

Runners A utility menu that lets you start several Windows applications that can be useful when working with word processor files. You can start additional instances of Write (without a second instance of PowerBar/Write; PowerBar/Write will work with whichever instance of Write was most recently active), or instances of Notepad, Cardfile, the Windows 3.1 Character Map for finding nonkeyboard characters, the Object Packager for enclosing other files, launchers and other command lines in your Write files, the Calculator, the macro Recorder, the File Manager, Program Manager, or a full-screen DOS session. In addition, Run... will pop up a Run... dialog into which you can enter a command line or associated file name. Please note that PowerBar/Write assumes that the named applications are in your Windows directory, or at least on your path. All apps are started in "normal" size except File Manager, which is opened maximized, and become the active application. If you use Program Manager or File Manager as your Windows shell, choosing the shell from the menu will restore and activate its window, not run a second instance of either. The DOS session is a full-screen session, not windowed.

Powerbar/Write

7

Size Write/Float Bar This section breaks into two more sections: one that allows you to minimize, maximize and restore the Write window, just as is you were to use Write's control menu or its sizing buttons, and one that lets you set PowerBar/Write's window to sink as a normal Windows window would do, or to float on top of other apps if it had been set to sink.

Exits ExitWrite & PowerBar/Write closes both Write and PowerBar/Write. PowerBar/Write opens and closes both itself and its "client." Is that user-friendly, or what? Exit Windows closes Windows. You will be prompted to save any changed files before Windows closes, so this is a safe exit.

PowerBar/Write: This and The Next Edition

PowerBar/Write was written late into the nights and early in the mornings of late June 1992, on a 286 machine. "Betas" were released to exactly one tester and one magazine editor.

Some features that will probably be included in PowerBar/Write 2.0:

- Paragraph styles and style sheets
- Timed autosave
- Picture buttons
- More bar configuration options
- Ability to set default fonts
- ANSI nonalphanumeric character generator--foreign letters, symbols and bullets
- Printer landscape/portrait switcher
- Brilliant ideas from users like you

Registration

PowerBar/Write is shareware; as usual, please limit yourself unregistered use to a maximum 30-day trial period. If you like and use PowerBar/Write regularly, please register it. This gets you the following: The feeling that you've gotten something for your money, which is almost always superior to feeling that you've gotten away with something. The knowledge that you're encouraging ideas and better programming in the future. Notification of updates, and perpetual license to subsequent editions of PowerBar/Write. And if you pay the higher registration price, automatic updating to the next version by mail.

Prices:

\$10 -- Standard registration license: Registers PowerBar/Write for use on any three computers, and I don't care whose--pass it to two friends, put it on a desktop at home, a mobile laptop and a desktop at work--you choose. You'll receive license by mail. Registration is perpetual; I don't expect you to pay for PowerBar/Write v. 2.0 or higher once you've paid for v. 1.0. But you're on your own as to finding future versions.

\$15 -- Same as standard registration, except that I'll send you the next major update by mail (probably version 2.0; you can specify whether you want any intermediate release instead).

Send check or money order payable to Argyle Softstuff, or, if you want to risk it, cash to:

**Argyle Softstuff
D. Stewart
2087 Edgewood
Berkley MI 48073**

Support

You can write to me, David Stewart, at the above address, or you can drop me a line on Prodigy, ID CTKJ00D (those are zeros between the J and D), or America Online, name Doc

Powerbar/Write

9

Yeah.

About the Programmer

David Stewart is an advertising copywriter, an experienced and widely published free-lance writer, reviewer and editor and now a learning programmer. He is working on a book on radically customizing Microsoft Windows--far beyond the Control Panel and WIN.INI and SYSTEM.INI settings.

If you want your programs to have a sophisticated advertising program, but can't afford an agency, or you would like to give your documentation the touch of a professional writer or editor, contact him at the address above, on Prodigy or by phone: call directory assistance at 1-313-555-1212 and ask for David Stewart in either Royal Oak or Auburn Hills (depending on whether he has moved his residence yet or not).

Acknowledgements

Thanks to:

The Waite Group, for their book *The Waite Group's Visual Basic How-To*, which gave me many ideas and lots of code for making this program work. Without this book, this program, which has been floating around in my mind for some time, could not have been carried out--at least not so quickly, and possibly not by me. The file finder started with a project in this book, and contains plenty of code therefrom, but PowerBar/ Write's file finder is greatly modified and amplified in its power.

Blake Ragsdell, editor of *Inside Visual Basic*, for making a phone call to explain just how to float PowerBar/Write, and for putting a lot of other great ideas in one neat publication.

Gary James and Richard Spurlock, Prodigy correspondents, who helped out with file association.

Tom Hoots, Windows maven and power-user, who had--and continues to have--fine insight into how to make programs useful and easy to use. Tom, eventually everything you've suggested will get into PowerBar/Write.

Disclaimer and other tiny print.

Man, I hate this part. Just like the big software companies, I disclaim all responsibility for effects of the use of PowerBar/Write. All warranties, either express or implied, including but not limited to fitness for any particular purpose other than interacting with Windows Write are disclaimed. Nor will Argyle Softstuff be liable for any damages whatsoever arising from the use of this program. Let's keep the lawyers out of this: If you have a problem, let me know.