# Tool Menu Contents

# Tool Menu Overview

Tool Menu is a unique application that modifies a host application, such as Visual Basic, by appending a menu entitiled "Tools."   The Tool menu can be <u>customized</u> to spawn various applications, even automatically when Tool Menu is run.   Tool Menu will even spawn its host application, once <u>initialized</u>.

Tool Menu is designed to improve productivity by adding a quick list of frequently used applications under the context of the host application.   In addition, Tool Menu can automatically launch specified applications when loaded and will automatically close spawned applications when it unloads.   This has the effect of maintaining a "desktop" of applications that is convieniently opened and closed as a single unit.

# Customizing Tool Menu

The Tool menu can be customized in terms of item order and menu text.   To change the text of each item (or any other item property) select either <u>Add or Edit</u> with the appropriate item highlighted in the item list.   To change the oder in which items are listed in the menu, simply use the arrow buttons while the item to move is highlighted in the item list.

Items can also be removed by highlighting the desired item and selecting the Remove button.   All changes can be cancelled by pressing cancel.

If the Customize Tool Menu dialog is closed via the system menu, a prompt will appear asking wether to unload the Tool Menu.   A yes answer will unload the Tool Menu application, reset the host menu, and close all spawned applications.

Tool Menu is Copyright 1995 Binary Solutions.

# Add/Edit Tool Menu Item

You can add menu items or edit the properties of existing menu items through this mechanism.   The flexibility of Tool Menu lies in these properties:

**Menu Text**

The text displayed in the menu can be editied here.   The ampersand (&) can be used to specifiy a shortcut key by placing it directly before the intended character.

**Filename**

The file to be executed is specified here as a full path (unless the file resides in the current path)   If the file is selected via the Browse button, the full path will be used. Note that a file need not be an executable, assuming the extension has a registered program associtaed with it.   See Program Manager to Associate extensions with applications.

**Parameters**

If desired, a command line can be passed along with the file to be executed.   To specify parameters at the time of launch, check the appropriate box. Note: Parameters will not be requested at time of launch if the launch is ocurring during Tool Menu load.

**Directory**

The default directory for an application can optionally be specified.   You must ensure the directory exists.   If nothing is entered, the application's directory will be the default (startup) directory.

**Auto Launch**

An application can be set to automatically run when Tool Menu is started.   This is very convenient for helper applications generally associated with the main task session such as Visual Basic.

**Run Minimized**

An application can be set to run in a minimized state.   However, if an application has its own method for determining startup state, it may be overridden.

# Special Modifications

Tool Menu can be modified for use with many Windows applications.   However, Tool Menu does not have an interface for such changes.   It is therefore necessary for the user to change Tool Menu's INI file directly, which is found in the same directory as the Tool Menu executable.   After changing these settings, it is advisable to follow the <u>initialization</u> procedure.   The following entries can be placed under the [Tool Menu] section:

**MenuText**=*Tool Menu Name*  (Default: &Tools...)
**Module**=*Application Module Name*   (Default: VB)
**TitleText**=*Caption Text for desired window in specified module* (Default: Microsoft Vusual Basic [])

Note that not all applications are suitable to Tool Menu implementation.   You should be specifically wary in MDI applications, as the menu resource displayed can be a child window's menu.

Users should also be aware that at this time, Tool Menu can only be implemented as one instance.

Tool Menu is Copyright 1995 Binary Solutions.

# Initializing Tool Menu

Tool Menu is most easily initialized by first running the intended host application and then running Tool Menu.   If the intended is not Visual Basic, <u>special changes</u> will be required in Tool Menu's INI file before initialization can take place.

Tool Menu will look for an "Option" menu in the host application to place its <u>customization</u> command.   If an option menu cannot be located, the customize command will be placed in the Tool Menu itself.

Tool Menu is Copyright 1995 Binary Solutions.

# Contacting the Author

Mark Bessett can be reached via email at the following addresses:

CompuServe: **72370,1773**
Internet: **72370.1773@Compuserve.com**